Universitat Politècnica de Catalunya

PhD Thesis

# Dynamic OD Matrix Estimation exploiting ICT traffic measurements

Xavier Ros Roca

Advisor: Lídia Montero Mercadé
Co-advisor: Jaume Barceló Bugeda

Departament d'Estadística i Investigació Operativa

Als meus pares

# Abstract

During the last decades, urban mobility has become the main concern for city councils and transportation operators. The main problem is the traffic congestion that easily appears in urban networks, producing negative economic impacts for the associated cost and, what is becoming more relevant from the sustainability point of view, pollution and noise that affect negatively not only the environment of the city but also to the public health of the citizens and also of our planet. In this context, the transportation operators and planners make use of traffic simulation models that assist their strategic decisions aiming at improving the mentioned problems.

The dynamic OD matrices estimation problem is a crucial step in transportation modeling and simulation because they contain the total number of vehicles that are circulating throughout the city, including their origins, destinations, and their departing time and describe the associated mobility patterns in terms of trip distributions. As this information is not directly observable in reality, this problem has been widely studied and many different methodologies have been proposed in order to obtain the suitable OD matrices that reflect the urban mobility of the studied area. The common approach is to use the counting stations data sets to estimate, using a minimization problem, the OD matrices that produce them. This is called the bi-level optimization approach. However, the main problem of this approach is that it is mathematically underdetermined, because many different OD matrices can produce the same traffic counts on certain links of the urban network, but presenting totally different trip distributions that could not correspond to the socio-demographic structure originating them. Many researchers addressed this problem with different types of conventional data, such as link speeds and densities with the intention of reducing the degrees of freedom of the problem.

The structure of an OD matrix describes how trips are distributed from the different origins to the different destinations and then represents the demand pattern of the area of study. Since two different OD matrices can generate the same traffic counts, the study of the structural similarity between OD matrices is completely necessary. In this thesis, we address the different studies measuring the structural similarity between the estimated OD matrix and the reliable OD matrices, which are the ground truth OD matrix in synthetic experiments or the historical OD matrix in the real ones.

The appearance of new sources of traffic data from the growth of the information and communication technologies (ICT) appeals to the researchers to use it for reducing such underdetermination, adding it to the OD estimation problem. GPS devices are increasingly used by vehicles and a huge volume of data is generated every day that, implicitly, contains information of the traffic state under real conditions. These data can be analyzed and processed in order to clean, filter and extract this information and can

be then introduced into the OD estimation problem. Most of the theoretical research since the ICT technologies are available assume implicitly or explicitly that GPS tracking data can be done through a controlled collection process. However, in the practical world, GPS data are supplied by companies that use different data collection policies and constraints imposed by privacy policies, which invalidate some of these theoretical hypotheses. One of the main research aspects of this thesis is to investigate how these commercial data can be used for the OD estimation problem.

However, the introduction of such information in the bi-level optimization problem is not direct and many alternatives arise. The extensively used analytical techniques do not permit an easy addition of such data, because it is not clear how to analytically relate it with the OD flows. On the other hand, the versatility of the simulation-based optimization methods permit such incorporation but its computational burden is an inherent drawback. This thesis proposes a data-driven estimation of the dynamic assignment matrix to introduce the GPS data information to an analytical model, reducing the underdetermination of the problem. Moreover, such estimation replaces the dynamic traffic assignment reducing also the computational effort of the OD estimation problem.

As this thesis results from the collaboration between the simulation software company PTV Group and the Universitat Politècnica de Catalunya, all the experiments of this thesis have been carried out in PTV Visum and using the already existing products. Moreover, the results have been analyzed both from the computational performance and from the quality aspect. The latter, as mentioned above, aims to analyze and find the structurally most appropriate OD matrix in the OD matrix estimation problem.

# Resum

Durant les últimes dècades, les externalitats que es deriven de la mobilitat urbana han estat una de les principals preocupacions dels ajuntaments, gestors metropolitans i operadors de transport. El principal problema és la congestió, que fàcilment apareix en infraestructures urbanes i que impacta negativament en la nostra economia i, el que és més greu, en la sostenibilitat del planeta en que vivim. La contaminació i el soroll provocats per la congestió no només afecten nocivament a la qualitat de l'aire, sinó que també afecten la salut ciutadana i mediambiental. En aquest context, els operadors i planificadors de trànsit utilitzen models de planificació i simulació de trànsit que els aporten coneixement per dur a terme decisions estratègiques i operatives que mitiguin els problemes associats a la mobilitat urbana.

El problema d'estimació de les matrius origen-destinació (OD) és un tema crucial en la modelització i simulació del trànsit. Aquestes contenen el nombre total de vehicles que circulen per la ciutat, incloent informació sobre els l'origen, destinació i temps de sortida de cadascun en un horitzó temporal. D'aquesta manera, la distribució de viatges definida en les matrius OD descriu el patró de mobilitat de la xarxa. No obstant això, aquesta informació no és directament observable en un cas pràctic real i, per aquest motiu, es tracta d'un problema profundament estudiat. S'han desenvolupat diferents metodologies que procuren obtenir matrius OD apropiades, és a dir, que reprodueixin correctament la mobilitat de la zona estudiada. L'enfoc més comú consisteix en usar dades recollides per sensors de trànsit que compten vehicles en certs punts de la xarxa per estimar les matrius OD mitjançant la resolució d'un problema de minimització. De tota manera, aquest problema complex és altament indeterminat i diferents matrius OD, que representen realitats sociodemogràfiques i patrons de mobilitat diferents, poden reproduir els mateixos comptatges de vehicles en les vies de la xarxa dotades de sensors. Per tant, moltes línies de recerca han usat diferents tipus de dades de transport addicionals, com ara velocitats mitjanes i densitats de flux, per reduir els graus de llibertat del problema.

L'estructura d'una matriu OD descriu el nombre de viatges i la forma com es distribueixen espaialment en la xarxa urbana, des del seu origen a la seva destinació, traçant així el patró de mobilitat global de la xarxa d'estudi. Com que dues matrius OD poden generar els mateixos comptatges, és absolutament necessari fer un estudi exhaustiu de la similaritat de les seves estructures. En aquesta tesi, enfoquem les diferents propostes mesurant sempre el grau de similaritat estructural entre la matriu OD estimada i una matriu OD de referència, sent aquesta la matriu OD històrica en casos reals o la matriu fonamental en el cas dels experiments sintètics.

L'aparició de noves fonts de dades de trànsit degut al creixement de les tecnologies de la informació i comunicació (TIC) obre noves línies de recerca adreçades a reduir la indeterminació del problema

d'estimació de les matrius OD. L'ús d'aparells GPS en vehicles va en augment, fet que contribueix a la generació diària de grans volums de dades. Aquestes contenen, de manera implícita, informació de l'estat del trànsit en condicions reals. Mitjançant un procés de neteja, filtratge i extracció es pot derivar informació del trànsit per a després introduir-la al problema de l'estimació de matrius OD. El conjunt de dades GPS de tipus comercials no permet conèixer el procediment de recol·lecció de dades i, sovint, està subjectes a polítiques de protecció i privacitat que no permeten assumir certes hipòtesis de qualitat i control en relació als orígens i destinacions. En aquesta tesi, investiguem el valor que poden afegir aquests conjunts de dades comercials per a l'estimació de matrius OD.

La introducció d'aquestes dades al problema d'optimització binivell no és directa i existeixen diverses alternatives. Els enfocs analítics no permeten introduir directament aquestes dades perquè la relació entres les dades GPS i els fluxos OD no és elemental. Per altra banda, la versatilitat dels mètodes de simulació-optimització permeten usar-los directament, però l'inconvenient és l'esforç computacional associat. Aquesta tesi proposa un model de la matriu dinàmica d'assignacions basat en dades (*data-driven*) per aprofitar la informació implícita de les dades GPS i reduir, així, la indeterminació del problema. A més, aquesta tècnica substitueix la necessitat de recórrer a un model de simulació y redueix l'esforç computacional del problema.

Aquesta tesi és fruit de la col·laboració entre l'empresa de software de simulació PTV Group i la Universitat Politècnica de Catalunya. Tots els experiments d'aquesta tesi han estat implementats en PTV Visum i usant els productes existents. A més, els resultats de la tesi han estat sempre analitzats des d'una doble perspectiva: computacional i de la qualitat. Aquesta última té com a objectiu analitzar la matriu OD pel que fa a la seva similaritat estructural amb la matriu de referència.

# Resumen

Durante las últimas décadas, las externalidades que se derivan de la movilidad urbana han sido una de las principales preocupaciones de los ayuntamientos, gestores metropolitanos, y operadores de tráfico. El principal problema es la congestión, que fácilmente aparece en infraestructuras urbanas y que impacta de forma negativa en nuestra economía y, lo que es más grave, en la sostenibilidad del planeta que habitamos. La contaminación y el ruido provocados por la congestión no solo afectan nocivamente a la calidad del aire, sino que también perjudican la salud ciudadana y medioambiental. En este contexto, los operadores y planificadores de transporte usan modelos de planificación y simulación de tráfico que les aportan conocimiento para tomar decisiones estratégicas y operativas que mitiguen los problemas asociados a la movilidad urbana.

El problema de la estimación de las matrices origen-destino (OD) es un tema crucial en la modelización y simulación de tráfico. Éstas contienen el número total de vehículos que circulan por la ciudad, incluyendo información sobre el origen, destino y tiempo de salida de cada uno de los vehículos en un horizonte temporal. De esta manera, la distribución de viajes definida en las matrices OD describe el patrón de movilidad de la red. Aún así, esta información no es directamente observable en un caso práctico real y, por este motivo, se trata de un problema extensamente estudiado. Se han desarrollado diferentes metodologías con el fin de obtener las matrices OD más apropiadas, es decir, aquellas que reproducen adecuadamente la movilidad de la zona estudiada. El enfoque más común consiste en usar datos recogidos por sensores de tráfico que cuentan vehículos en ciertos puntos de la red para estimar las matrices OD mediante la resolución de un problema de minimización. Aún así, este complejo problema es altamente indeterminado y diferentes matrices OD, que representan realidades sociodemográficas y patrones de movilidad distintos, pueden reproducir los mismos conteos de vehículos en las vías de la red dotadas de sensores. Por consiguiente, muchas líneas de investigación han utilizado de forma adicional diferentes tipos de datos de tráfico, como velocidades medias y densidades de flujo, para reducir los grados de libertad del problema.

La estructura de una matriz OD describe el número de viajes y la forma como se distribuyen espacialmente en la red urbana, desde su origen hasta su destino, trazando, así, el patrón de movilidad global de la red de estudio. Como dos matrices OD pueden reproducir los mismos conteos, es absolutamente necesario hacer un análisis exhaustivo de la similitud de sus estructuras. En esta tesis, abordamos las diferentes propuestas midiendo siempre el grado de similitud estructural entre la matriz OD estimada y una matriz OD de referencia, siendo ésta la matriz OD histórica en casos reales o la matriz fundamental en el caso de los experimentos sintéticos.

La aparición de nuevas fuentes de datos de tráfico debido al crecimiento de las tecnologías de la información y comunicación (TIC) abre nuevas líneas de investigación dirigidas a reducir la indeterminación del problema de estimación de las matrices OD. El uso de aparatos GPS en vehículos va en aumento, hecho que contribuye a la generación diaria de grandes volúmenes de datos. Éstos contienen, de manera implícita, información del estado del tráfico en condiciones reales. Mediante un proceso de limpieza, filtrado, y extracción se puede derivar información del tráfico para luego introducirla en el problema de estimación de matrices OD. El conjunto de datos GPS de tipo comercial no permite conocer el procedimiento de recolecta de datos y, a menudo, está sujeto a políticas de protección y privacidad que no permiten asumir ciertas hipótesis de calidad y control en relación a los orígenes y destinos. En esta tesis, investigamos el valor que pueden añadir estos conjuntos de datos comerciales para la estimación de matrices OD.

La introducción de estos datos en el problema de optimización binivel no es directa y existen diferentes alternativas. Los enfoques analíticos no permiten incorporar directamente estos datos puesto que la relación entre los datos GPS y los flujos OD no es elemental. Por otro lado, la versatilidad de los métodos de simulación-optimización permiten usarlos directamente, pero el inconveniente es el esfuerzo computacional asociado. Esta tesis propone un modelo de la matriz dinámica de asignaciones basado en datos (*data-driven*) para aprovechar la información implícita de los datos GPS y reducir, así, la indeterminación del problema de estimación. Además, esta técnica reemplaza la necesidad de recurrir a un modelo de simulación y reduce el esfuerzo computacional del problema.

Esta tesis es fruto de la colaboración entre la empresa de software de simulación PTV Group y la Universitat Politècnica de Catalunya. Todos los experimentos de la tesis han sido implementados en PTV Visum y usando los productos existentes. Además, los resultados de la tesis han sido siempre analizados desde una doble perspectiva: computacional y de calidad. Esta última tiene como objetivo analizar la matriz OD estimada respeto la similitud estructural con la matriz de referencia.

# Acknowledgements

grup són els companys de despatx a l'UPC: Gracias Carlos y Dani, Paula, Vicky, Glòria, Diana... por simplemente compartir cafés, comidas y despacho.

Gràcies també als amics i amigues que m'acompanyen des de fa anys. De vegades es fa complicat ajudar a una persona que fa un doctorat, però les vostres mostres de suport i interés en tot moment i en moments més intensos són molt valuoses. En aquest sentit, vull agrair especialment la capacitat de la Glòria per escoltar el problema, reduir-ne la importància i sintetitzar la solució en unes quantes paraules que podrien resumir-se: *No li donguis tantes voltes, relativitza, sigues feliç!* Amics i amigues que no tan sols han estat aprop si no que s'han implicat en la tesi, aportant un gra de sorra. L'Anna, la Paula i la Sandra com a correctores; el Raül i el Víctor per acompanyar-me a donar voltes i a córrer quan necessitava sortir i distreure'm. Al grup de mates sencer, que fa 10 anys que ens coneixem i ens seguim veient. A ex-companys de feina com l'Alberto, l'Erne i l'Ana i a persones molt especials com la Margot i la Clara, l'Annabel i el Pepe, l'Arnau i la Laura i els companys i companyes de DataForGoodBCN. Finalment, gràcies a molta més gent que he anat coneixent aquests anys.

I finalment, moltes gràcies a la meva família. A la meva germana Anna i el Joel, a l'àvia i a les famílies Ros i Roca al complet. A l'Albert per acompanyar-me sempre i, a més, treballar intensament per a la tesi. I en darrer lloc, als meus pares. Ells han estat sempre al meu costat. Com em coneixen i saben que dono pocs detalls, han sabut llegir cada etapa, felicitar-me quan aconseguia fites importants, preocupar-se en moments complicats i acompanyar-me tots aquests anys (i els anteriors). Han sabut donar consells basats en l'experiència, a relativitzar i sobretot, han sabut estimar-me.

# Contents

<div align="right">

# 1

</div>

# Introduction: The role of origin-destination matrices in transport models

*This chapter introduces the OD matrix estimation problem and presents the different approaches that have been studied from its origins. It also shows the context of this thesis, the outline, the contributions and the related publications of the author.*

Transportation analysis attempts to understand traffic patterns in a given geographic area, most frequently an urban or metropolitan area spanned by a transportation network. Such analysis represents the transport network under certain conditions in order to provide insight into how the transport infrastructure is used by transport demand, that is, the trips in the area. Transport demand is commonly defined in terms of an origin-to-destination (OD) matrix, $\mathbf{X}$, whose entries $(i, j)$ represent the number of trips from a certain origin $i$ to a certain destination $j$. From a practical point of view, the area object of study is partitioned into a number of transport analysis zones (TAZ) by following well established criteria that balance land use and socioeconomic information provided by various sources such as censuses and city planners (Ortúzar & Willumsen (2011)). Traffic demand is defined in terms of an OD matrix, and the traffic assignment is the process determining how that demand loads onto the network. In this way, it is possible to compute the path flows and thus the traffic flows on the network links in order to explain trip behavior and accessibility to specific locations (Figure 1.1).



Traffic analysis zones (TAZ)    Demand model OD matrix

Supply model: Transport network

Traffic assignment
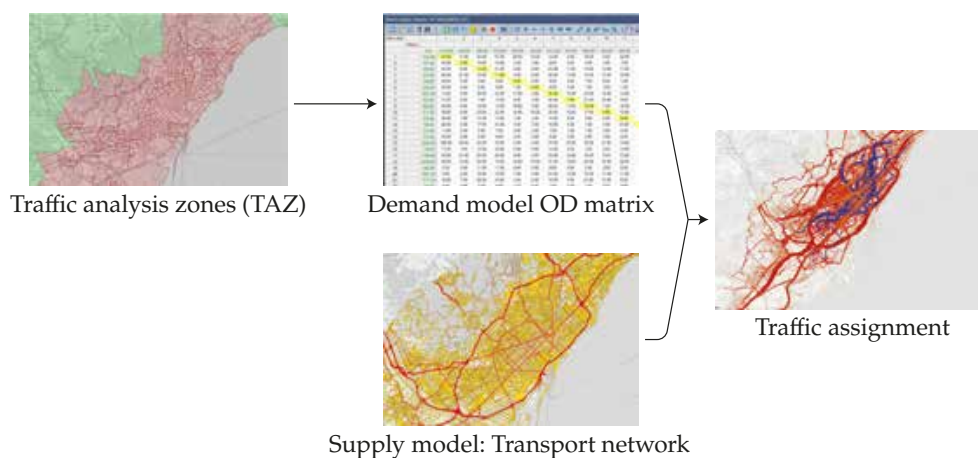
Figure 1.1: Transport analysis supported by traffic assignment models

The underlying modeling hypothesis predicts that vehicles travel from origins to destinations in the network along the available routes connecting them, in accordance with rules explaining their behavioral choices. The characteristics of a traffic assignment procedure are determined by hypotheses on how travelers use the routes. The main modeling hypothesis is based on the concept of *user equilibrium*, which assumes that travelers try to minimize their individual travel times, that is, travelers choose the routes that they perceive to be the shortest under the prevailing traffic conditions. This modeling hypothesis is formulated in terms of *Wardrop's first principle* (Wardrop (1952)): the journey times on all the routes actually used between an origin and a destination are equal to and less than those which would be experienced by a single vehicle on any unused route.

Traffic assignment models based on this principle are known as *user equilibrium models*, as opposed to models in which the objective is to optimize the total system travel time independently of individual preferences (see Sheffi (1985), Florian & Chen (1995), Patriksson (1994)). Florian & Hearn (1995) stated the equations to be satisfied for a user equilibrium model. Given the following: an origin $i$; a destination $j$ with flow $x_{ij}$; all the different paths between them $p \in \mathcal{P}_{ij}$; a certain flow $x_{ijp}$; travel time $tt_{ijp}$; and the shortest path between this origin and destination represented by $tt_{ij}^*$; then:

$$
\begin{cases}
(tt_{ijp} - tt_{ij}^*)x_{ijp} = 0 & \forall p \in \mathcal{P}_{ij} \ \ \forall (i,j) \in N \\[2mm]
tt_{ijp} - tt_{ij}^* \geq 0 & \forall p \in \mathcal{P}_{ij} \ \ \forall (i,j) \in N \\[2mm]
tt_{ijp}, tt_{ij}^*, x_{ijk} \geq 0 & \forall p \in \mathcal{P}_{ij} \ \ \forall (i,j) \in N \\[2mm]
\sum_{p \in \mathcal{P}_{ij}} x_{ijp} = x_{ij} & \forall (i,j) \in N
\end{cases}
\tag{1.1}
$$

If all the equations are satisfied, then these flows are in an equilibrium that satisfies Wardrop's principle. Effectively, if path $p$ from origin $i$ to destination $j$ carries a flow $x_{ijp} > 0$, then the first equation is satisfied only if the path cost $tt_{ijp}$ is equal to the minimum path cost $tt_{ij}^*$ for all paths from $i$ to $j$, as required by Wardrop's principle. Reciprocally, to satisfy the first equation if the path cost $tt_{ijp}$ is greater than the minimum path cost, then the flow on path $p$ must be zero. In other words, it is an unused path, according to Wardrop's principle. This formulation is usually applied in the static traffic assignment models that are widely used in strategic transportation analysis.

New technology has given rise to intelligent transport systems (ITS), advanced traffic management systems (ATMS), and advanced traffic information systems (ATIS), for which there exists a need for models that account for flow changes with time. More specifically, these dynamic models must be able to appropriately describe the time dependencies of traffic demand and the corresponding induced traffic flows. The *dynamic traffic assignment* (DTA) problem can thus be considered an extension of the traffic assignment problem described above. DTA can determine such time-varying link or path flows, meaning that it describes how the network's traffic flow patterns evolve in time and space (Mahmassani (2001)). The problem can be formulated as a dynamic user equilibrium problem using the dynamic version of Wardrop's principle (Friesz et al. (1993), Smith (1993), Ran & Boyce (1996)), which states that if, for each OD pair at each instant of time, the actual travel times experienced by travelers departing at the same time are equal and minimal, the dynamic traffic flow over the network is in a travel-time-based *dynamic user equilibrium* (DUE) state.

Wu (1998) shows that the DUE approach can be implemented by solving the following mathematical model:

$$
\begin{cases}
(tt_{ijp}(t) - tt_{ij}^*(t))x_{ijp}(t) = 0 & \forall p \in \mathcal{P}_{ij}(t) \ \ \forall (i,j) \in N \ \ \forall t \in [0,T] \\[2mm]
tt_{ijp}(t) - tt_{ij}^*(t) \geq 0 & \forall p \in \mathcal{P}_{ij}(t) \ \ \forall (i,j) \in N \ \ \forall t \in [0,T] \\[2mm]
tt_{ijp}(t), tt_{ij}^*(t), x_{ijp}(t) \geq 0 & \forall p \in \mathcal{P}_{ij}(t) \ \ \forall (i,j) \in N \ \ \forall t \in [0,T] \\[2mm]
\displaystyle\sum_{p \in \mathcal{P}_{ij}(t)} x_{ijp}(t) = x_{ij}(t) & \forall (i,j) \in N \ \ \forall t \in [0,T]
\end{cases}
\tag{1.2}
$$

where, as before, $x_{ijp}(t)$ is the flow on path $p$ from $i$ to $j$, departing from origin at time interval $t$; $tt_{ijp}(t)$ is the actual path cost from $i$ to $j$ on route $p$, departing from origin at time interval $t$; $tt_{ij}^*(t)$ is the cost of the shortest path from $i$ to $j$, departing from origin at time interval $t$; and $\mathcal{P}_{ij}(t)$ is the set of all available paths from $i$ to $j$ at time interval $t$.

The formulation above is equivalent to solving a finite-dimensional variational inequality problem consisting of finding a vector of path flows $\mathbf{x}^*$ and a vector of path travel times $\tau$, such that

$$
[\mathbf{x} - \mathbf{x}^*]^\top \tau \geq 0 \qquad\qquad\qquad \forall \mathbf{x} \in \mathcal{N} \tag{1.3}
$$

where $\mathcal{N}$ is the set of feasible flows defined by

$$
\mathcal{N} = \left\{ \mathbf{x} = \{x_{ijp}(t)\}, \ \ x_{ijp}(t) \geq 0 \ \ \middle| \ \ \sum_{p \in \mathcal{P}_{ij}(t)} x_{ijp}(t) = x_{ij}(t) \qquad \forall (i,j) \in N \ \ \forall t \in [0,T] \right\} \tag{1.4}
$$

Wu (1991), Wu et al. (1998) proved that this is equivalent to solving the discretized variational inequality:

$$
\sum_{t \in [0,T]} \sum_{p \in \cup \mathcal{P}_{ij}} tt_{ijp}(t)[x_{ijp}(t) - x_{ijp}^*(t)] \geq 0 \tag{1.5}
$$

The objective of *traffic assignment models* is to assign a trip OD matrix onto a network by incorporating a route choice mechanism, in order to estimate the traffic flows in the network. Therefore, they all use OD trip matrices as major data input to describe the patterns of traffic behavior across the network. All formulations of static (Florian & Hearn (1995)) and dynamic traffic assignment models (Ben-Akiva et al. (2001)) assume the availability of a reliable OD estimate. However, since neither static OD matrices nor time dependent OD matrices are directly observable, but they are key inputs to traffic assignment models, namely to the dynamic ones, the problem of how to estimate them becomes crucial for the use of these models.

## 1.1   Formulating the origin-destination matrix estimation problem using traffic counts

Historically, the processes used to construct OD matrices for simulating traffic in large transportation networks were direct OD estimation methods, such as sampling surveys and home interviews. These

methods are usually expensive, complex, and they usually lack any acceptable precision. On the other hand, the late 1970s gave rise to researchers taking more interest in indirect OD estimation methods because they infer OD matrices from real traffic measurements that are collected from the network.

These indirect estimation methods are the so-called matrix adjustment methods, whose main modeling hypothesis is stated in Section 8.5 of Cascetta (2001). Traffic flows in the links of a network are the consequence of assigning an OD matrix onto a network. Then, if it is possible to measure such link flows, estimating the OD matrix that generates the network flows becomes the inverse of the assignment problem.



(a) Direct problem of traffic assignment          (b) Reciprocal problem of OD estimation
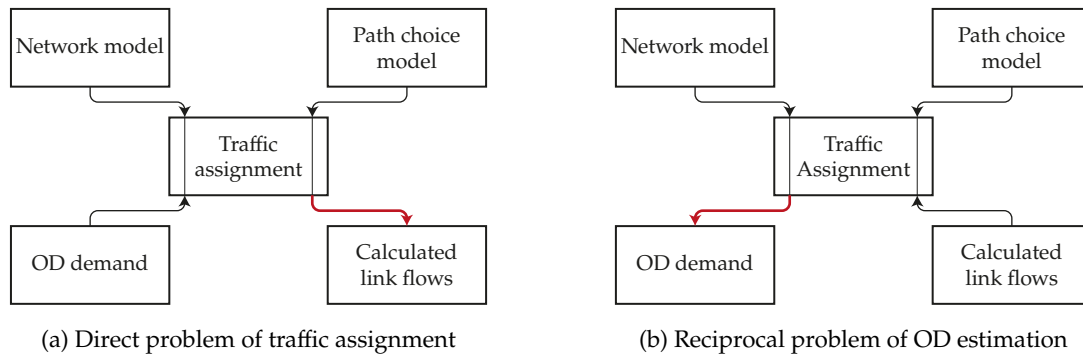
Figure 1.2: Traffic assignment and OD estimation problems

The assignment problem shown in Figure 1.2a uses the OD matrix, the cost, and the network conditions to estimate the user equilibrium flows on the road network's links. That is, $\mathbf{Y} = \mathtt{Assignment}(\mathbf{X})$, where $\mathbf{Y}$ is the set of all link flows, $\mathbf{X}$ is the OD matrix, and *Assignment* is an equilibrium assignment algorithm assigning the OD matrix $\mathbf{X}$ onto the network. Then, the reciprocal problem (Figure 1.2b) would be the inverse, which is namely estimating from the observed link flows $\hat{\mathbf{Y}} = [\hat{y}_l]_l$ the OD matrix $\mathbf{X}$ that gave origin to them. In other words, the reciprocal problem (as defined by Cascetta (2001)) consists of assuming that the observed flows $\hat{y}_l$ in a subset $\hat{L} \subseteq L$ of network links constitute a Wardrop user equilibrium flow pattern. Therefore, the link flows determine the OD matrix X whose assignment would produce such observed flows $\hat{y}_l$. This approximation is called origin–destination count-based estimation (ODCBE).

Using traffic counts for OD matrix estimation originated with Van Zuylen & Willumsen (1980), who used maximum entropy models and emphasized that the main advantages of traffic counts are their low cost and availability. Cascetta (1984), Cascetta & Nguyen (1988) followed in their steps but used a generalized least squares (GLS) approach to estimate OD trips. In the following years, many approaches were developed by Spiess (1990), Yang et al. (1992), Florian & Chen (1995), who were the first to treat the OD estimation problem as a bi-level optimization problem in which one optimization problem needs to solve another optimization problem. The aforementioned authors presented algorithmic methods to solve them.

OD estimation methods fall under broad classifications because they have been intensively studied under different approaches due to the importance of model calibration. They are classified according to various factors such as demand profile (*static* or *dynamic*); the traffic data defining the input to the problem (*direct* traffic measurements like traffic counts and/or *indirect* data like prior OD matrices induced from surveys); the calibration model's purpose (*offline* for designing traffic management strategies and *online* for predicting real-time traffic situations); and how the problem is solved (*analytical* or *simulation-based optimization* techniques). This thesis focuses on offline models for both static and dynamic demand, and

it uses direct traffic measurements such as traffic counts, as well as both analytical and simulation-based optimization techniques.

## 1.2 The bi-level optimization problem

Van Zuylen & Willumsen (1980) were pioneers in proposing a model for estimating OD matrices based on knowledge of paths used and the traffic counts of certain links in the network. However, these models consider the path proportions to be constant and independent of the congestion produced by changing demand. Therefore, that initial approach had to be reformulated to overcome these drawbacks, this was the bi-level optimization, Spiess (1990), Yang et al. (1992), usually considered as the most appropriate adjustment for combining available data sources when estimating OD matrices, because it explicitly takes into account congestion effects that influence the paths used between OD pairs. The problem is then formulated as an optimization problem that minimizes some discrepancy functions between observed measurements and their corresponding simulation measurements, which must be computed using a traffic assignment procedure. The problem can be written as

$$
\begin{aligned}
\min \quad & Z\left(\mathbf{X}\right) = w_1 F_1\left(\mathbf{Y}, \hat{\mathbf{Y}}\right) + w_2 F_2\left(\mathbf{X}, \mathbf{X}^H\right) \\
\text{s. to:} \quad & \mathbf{Y} = \text{Assignment}(\mathbf{X}) \\
& \mathbf{X} \geq \mathbf{0}
\end{aligned}
\tag{1.6}
$$

where $F_1$ and $F_2$ are distance functions between estimated and observed values, and $w_1$ and $w_2$ are weighting factors reflecting the uncertainty and importance of the information contained in $\hat{\mathbf{Y}}$ and $\mathbf{X}^H$, respectively. The underlying hypothesis is that $\mathbf{Y} = \{y_l\}$ are the link flows predicted by assigning the demand matrix $\mathbf{X} = \{x_{ijr}\}$. As shown, the traffic volumes on certain links constitute the core of the problem, but an available historical OD matrix $\mathbf{X}^H$ can be added to the formulation of the objective function, which is usually provided by either a household survey or a former demand model.

From a mathematical point of view, the problem is highly underdetermined, because there are usually more variables, the OD flows, than number of observations, the traffic counts. Moreover, as traffic counts are the combination of different OD flow proportions, there are multiple combinations of such proportions that can provide the same sum on a certain link. Finally, it is notorious the work of Bierlaire (2002) proving that the problem is also underdetermined in the extreme case of a full coverage, when a sensor is placed at each link. That facts imply that there are infinitely many solutions that with different OD flows can supply the same traffic counts.

Moreover, the complexity of the assignment problem shows that it is generally a non-convex and non-differentiable function; therefore, analytical approaches are limited because they are constrained to simple uncongested cases. Consequently, other formulations have been proposed, which is the reason for moving from the minimization problem of Equation 1.6 to the bi-level OD matrix estimation problem shown in Figure 1.3.

The problem is split into two different parts. At the lower level of the algorithmic scheme shown above, a traffic assignment is made at each iteration of the minimization algorithm in order to obtain the assignment outputs $\mathbf{Y}(\mathbf{X}^{(k)})$, which are the traffic counts on certain links in the network. The role of the lower level is to update the assignment matrix at each iteration and to build it using the new OD

$$\begin{array}{|c|}
\hline
\min_{X} \quad Z(\mathbf{X}) = w_1 F_1(\mathbf{Y}, \hat{\mathbf{Y}}) + w_2 F_2(\mathbf{X}, \mathbf{X^H}) \\
\text{s. to} \quad \hat{\mathbf{Y}} = \text{Assignment}(\mathbf{X}) \\
\mathbf{X} \geq 0 \\
\hline
\end{array}$$
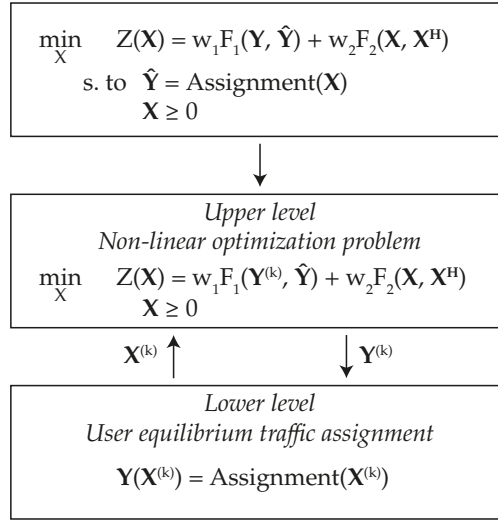
Figure 1.3: The bi-level optimization problem

proportions in the path search, due to changing congestion effects. The upper level calculates a sequence of OD matrices using the assignment outputs and the measured traffic counts in order to minimize the objective function. Therefore, solving the OD estimation problem consists of conducting a sequence of iterations between the lower and upper level in order to find, at each step, a better OD matrix **X**. Together with the corresponding outputs of the traffic assignment, this improved matrix decreases the objective function until a suitable solution is reached. Many approaches can be taken to solving the upper level of the problem, but a user equilibrium traffic assignment is always required for the lower level. Originally, the analytical approaches were often used to solve the offline static OD estimation (Spiess (1990), Florian & Chen (1995), Lundgren & Peterson (2008)).

### 1.2.1   Analytical Formulations

The reference formulation of the OD matrix estimation problem is Equation 1.6. However, in order to solve it analytically, the static traffic assignment can be approached with a linear function. The underlying hypothesis is that the link flows are predicted by assigning the demand matrix **X** onto the network, which can be expressed by a proportion of the OD demand flows passing through the count location at a certain link. In terms of the assignment matrix $\mathbf{A} = [a_{ij}^l]_{ijl}$, which is the proportion of an OD flow, $x_{ij}$, that contributes to a certain link's traffic counts, which can therefore be calculated as the sum of all these contributions as follows:

$$y_l = \sum_{(i,j) \in N} a_{ij}^l x_{ij} \quad , \quad \forall l \in \hat{L} \subseteq L \quad \Rightarrow \quad \mathbf{Y} = [y_l]_l = \mathbf{A}(\mathbf{X})\mathbf{X} \quad , \quad \mathbf{A}(\mathbf{X}) = \left[ a_{ij}^l \right]_{ijl} \tag{1.7}$$

where $\mathbf{A}(\mathbf{X})$ indicates that the assignment matrix depends on the OD matrix **X**. This approach is indeed an approach of the traffic assignment function. The bi-level optimization problem therefore consists of a nonlinear optimization problem at the upper level, and it requires the lower level assignment calculation

in order to obtain the corresponding assignment matrix that will be introduced into the following objective function:

$$\min \quad Z\left(\mathbf{X}\right) = w_1 F_1\left(\mathbf{A}(\mathbf{X})\mathbf{X}, \hat{\mathbf{Y}}\right) + w_2 F_2\left(\mathbf{X}, \mathbf{X}^H\right)$$
$$\text{s. to:} \quad \mathbf{X} \geq \mathbf{0}$$

(1.8)

When $F_1$ and $F_2$ are differentiable functions, this assumption converts the objective function into a differentiable function in terms of the minimization variables, which are the OD values $\mathbf{X}$. This is because all the objective function's elements depend directly on the OD values.

The analysis of the existing analytical approaches that are used to solve the offline static OD estimation without additional constraints, reveals that, either are based on Spiess (Spiess (1990), Florian & Chen (1995)), or on extensions of Spiess (Lundgren & Peterson (2008), Toledo & Kolechkina (2013)), while other approaches modify the formulation adding constraints (Codina & Barceló (2004), Doblas & Benitez (2005)). For the case without additional constraints this thesis explores further extensions to Spiess (1990), which basic formulation is outlined below, while the addition of constraints will be considered in a different context.

**Spiess Method (1990)**

Spiess (1990) proposes a classic gradient descent method to solve the static OD matrix estimation problem (Equation 1.8) once the assignment matrix is calculated. This approach relates linearly traffic counts to OD flows. Moreover, the second term of the objective function is not considered (that is, $w_2 = 0$), and $F_1$ is a quadratic difference function between the traffic count measurements and their corresponding simulated values. Hence, the formulation of the Spiess method is

$$\min \quad Z\left(\mathbf{X}\right) = \sum_{l \in \hat{L}} (y_l - \hat{y}_l)^2$$
$$\text{s. to:} \quad y_l = \sum_{(i,j) \in N} a_{ij}^l x_{ij}$$
$$\mathbf{X} \geq \mathbf{0}$$

(1.9)

The iterative procedure for solving the bi-level optimization problem is the following.

1) Assign $\mathbf{X}$ using simulation. Obtain $\mathbf{A} = \left[a_{ij}^l\right]_{ijl}$.

2) Calculate the traffic counts and the objective function:

$$y_l = \sum_{(i,j) \in N} a_{ij}^l x_{ij}$$

(1.10)

$$Z(\mathbf{X}) = \sum_{l \in \hat{L}} (y_l - \hat{y}_l)^2$$

(1.11)

3) Calculate the gradient $Z(\mathbf{X})$:

$$\frac{\partial Z}{\partial x_{ij}} = \sum_{l \in \hat{L}} 2a_{ij}^l (y_l - \hat{y}_l) \tag{1.12}$$

4) Calculate the optimal step for the gradient method, $\lambda^*$:

$$y_l' = \sum_{(i,j) \in N} -x_{ij} \frac{\partial Z}{\partial x_{ij}} a_{ij}^l \tag{1.13}$$

$$\tag{1.14}$$

$$\lambda^* = \frac{\displaystyle\sum_{l \in \hat{L}} y_l' (\hat{y}_l - y_l)}{\displaystyle\sum_{l \in \hat{L}} y_l'^2} \tag{1.15}$$

5) Calculate the next iteration of the OD matrix, $\mathbf{X}_{k+1}$:

$$\mathbf{X}^{(k+1)} = \mathbf{X}^{(k)} \left( 1 - \lambda^{*(k)} \nabla Z \left( \mathbf{X}^{(k)} \right) \right) \tag{1.16}$$

6) Go back to 1.

In practice, the Spiess method for the static OD matrix estimation problem is among the most robust for solving the bi-level optimization problem. Moreover, its simplicity and ease of implementation make it one of the most used methods for demand calibration. The problem shown in Equation 1.9 is a non-negative constrained problem, meaning that the OD flows can take a wide range of values. Codina & Barceló (2004) and Doblas & Benitez (2005) aimed at reducing the feasible set adding constraints and using an augmented Lagrangian function in Frank and Wolfe method.

The multiplicative formulation in Equation 1.16 of the gradient formulation preserves the zeroes of the initial OD matrix. This has been object of controversy because the usual additive formulation in iterative procedures could make that null cells of the initial OD become no null through the iterative process. The question is whether this is realistic or consequence of the numerical procedure with no relation with the underlying reality. If the initial OD matrix is structurally reliable then null cells will correspond to OD pairs for which no trips have been observed, and then, generating trip between these OD pairs just because the numerical method does, it would not be realistic. Since one of the hypothesis in this thesis is that structurally reliable historical OD matrices are available in most cases of interest, then we will use this assumption directly or indirectly whenever it could be possible.

### 1.2.2   Simulation-based optimization approaches

Another stream of research takes into account that traffic phenomena are usually stochastic and, thus, analytical approaches cannot capture all variable interactions or their effects on the objective function. Simulation-based optimization (SO) algorithms combine the simulator and the optimization algorithm to obtain the next OD matrix with a lower objective function value without any explicit analytical or numerical gradient. Different simulation-optimization techniques have been successfully proven, and these are heuristic optimization methods for stochastic systems (Bierlaire (2015)).

One powerful advantage of the SO approaches is that the objective function can include other traffic measurements such as travel times and link speeds, since these approaches need no analytical expression to obtain the gradient direction.

Some relevant alternatives are the Nelder–Mead simplex algorithm (Nelder & Mead (1965)); SNOBFIT (Huyer & Neumaier (2008)); genetic algorithms (Ma & Abdulhai (2002), Kim & Rilett (2004), Ma et al. (2006)); metamodels embedded into the simulation-based optimization approaches (Osorio & Chong (2015), Osorio (2019)); and simultaneous perturbation stochastic approximation (SPSA), (Spall (1998, 2003)), which is one of the most used in demand adjustment.

**Simultaneous perturbation stochastic approximation**

The simultaneous perturbation stochastic approximation (SPSA) is an iterative optimization algorithm that does not depend on the objective function's direct gradient information. It computes an approximation to the gradient after measuring the objective function in perturbed points on the neighborhood (further details on this are in Chapter 4). This gradient approximation is calculated without requiring an explicit functional relationship between the variables and the objective function, which makes it suitable for the OD estimation problem (Equation 1.6). Moreover, the gradient approximation is calculated with only a few evaluations of the objective function, which makes it computationally more efficient than a numerical gradient approximation.

Because of the characteristics described above and the possibility of easily including additional measurements like link speeds and travel times into the objective function (Antoniou et al. (2016), Carrese et al. (2017), Nigro et al. (2018)), many researchers have chosen to use SPSA (Balakrishna (2006), Cipriani et al. (2011), Antoniou et al. (2015), Cantelmo, Cipriani, Gemma & Nigro (2014), Lu et al. (2015), Kostic et al. (2015), Tympakianaki (2018)).

## 1.3   From static to dynamic: the dynamic origin-destination matrix estimation problem

Traffic modeling and simulation evolved together with the increase in computational power, allowing to address the new paradigm known as dynamic traffic assignment models. These models are able to capture the dynamics and evolution of a traffic network over time, thus increasing the level of detail and closeness to the reality represented by the model, although also its complexity. Moreover, the demand profile must be dynamic when used as a model input, thus requiring OD matrices that evolve over time and are usually split into time periods, so they can be understood as a time series of OD matrices. The transition from static OD matrices (Figure 1.4a) to dynamic OD matrices (Figure 1.4b) opened up the field for studying indirect estimation methods for dynamic OD matrices. The problem of obtaining a suitable dynamic OD matrix that is associated with the dynamic traffic assignment is called the dynamic OD matrix estimation (DODME) problem.

(a) Static OD matrix                                    (b) Dynamic OD matrix
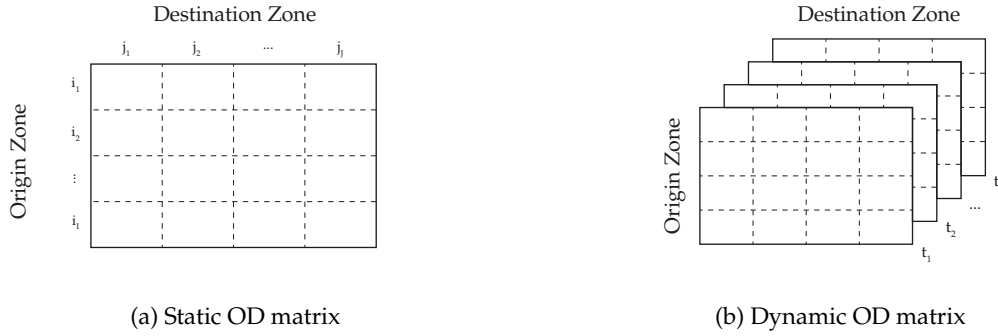
Figure 1.4: OD matrices for traffic simulation

The OD estimation problem formulated in Equation 1.6 indeed holds for both demand profiles. A static demand means a unique OD matrix for the whole time period under study, and therefore it is also expected that the traffic counts outputs of the static user equilibrium will be static.. On the other hand, a dynamic demand profile means that OD matrices evolving over time capture the traffic situation's dynamicity over time, and thus the traffic counts are also dynamic over time. The only thing that changes in the formulation is the problem's dimension of vectors and variables. Therefore, given $\mathcal{T} = \{t_1, \ldots, t_T\}$, which are the time periods dividing the simulation time, the OD matrices are $\mathbf{X} = \left[x_{ijr}\right]_{ijr}$, where $x_{ijr}$ is the flow from origin $i$ to destination $j$, departing at time period $r$, and the traffic counts are $\mathbf{Y} = [y_{lt}]_{lt}$, where $y_{lt}$ stands for the flow crossing link $l$ at time period $t$.

As mentioned above, dynamic traffic assignment models imply a dynamic demand profile, which in turn leads to the emergence of complex dynamic phenomena derived from traffic propagation over time such as congestion over time periods and different used paths for each time period. This incremental complexity has led researchers to explore new and different approaches that take into account the temporal behavior of the transport system as a complex one. The analytical approach, which will be further described in this thesis, has its origins on Frederix et al. (2010, 2011), Toledo & Kolechkina (2013), Cantelmo, Viti, Tampère, Cipriani & Nigro (2014). Many other approaches have been successfully tested, such as Kalman filter approaches for online dynamic OD problems (Ashok (1996), Ashok & Ben-Akiva (2002), Antoniou et al. (2004), Bierlaire & Crittin (2004), Lin & Chang (2007), Barceló et al. (2013)). Others include simulation-based optimization approaches, which have been proposed mainly by Balakrishna (2006), Cipriani et al. (2011), Djukic (2014), Antoniou et al. (2015), Tympakianaki (2018).

### 1.3.1   Extension of the analytical formulations

Similarly to the approach described in Section 1.2.1, the DODME problem can use the dynamic assignment matrix as an approximation of the DTA, which is $\mathbf{A} = \mathbf{A}(X) = \left[a_{ijr}^{lt}\right]$, and $a_{ijr}^{lt}$ represents the proportion of $(i, j)$ OD flow departing at time $r$, $x_{ijr}$ and passing through link $l$ at time $t$, $y_{lt}$. Therefore, the linear relationship between traffic counts and OD flows is

$$y_{lt} = \sum_{(i,j) \in N} \sum_{r=1}^{t} a_{ijr}^{lt} x_{ijr} \Rightarrow \mathbf{Y}_t = \sum_{r=1}^{t} \mathbf{A}_r^t \cdot \mathbf{X}_r \tag{1.17}$$

which takes into account flows departing in past time periods $r \leq t$ that can feed the sensor due to congestion and long travel times. Similarly, $\mathbf{A}_r^t$ is the partial assignment matrix that relates the OD flows

departing at time $r$ and which are detected by all the sensors at time $t$. Finally, the global information can be summarized as a matrix product between the dynamic assignment matrix and the OD flows for each time interval. This is expressed as

$$
\mathbf{Y} = \mathbf{A}(\mathbf{X}) \cdot \mathbf{X} \text{ with } \mathbf{A} = \begin{pmatrix} \mathbf{A}_1^1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{A}_1^2 & \mathbf{A}_2^2 & \mathbf{0} & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{A}_1^T & \cdots & \mathbf{A}_{T-1}^T & \mathbf{A}_T^T \end{pmatrix} \text{ where } \mathbf{A}_r^t = \begin{pmatrix} a_{i_1 j_1 r}^{l_1 t} & \cdots & a_{i_1 j_1 r}^{l_1 t} \\ \vdots & \ddots & \vdots \\ a_{i_1 j_1 r}^{l_L t} & \cdots & a_{i_I j_J r}^{l_L t} \end{pmatrix} \tag{1.18}
$$

where the vector of detected flows is $\mathbf{Y} = (\mathbf{Y}_1, \ldots, \mathbf{Y}_T) = (y_{11}, \ldots, y_{L1}, \ldots, y_{1T}, \ldots, y_{LT})$, and the vector of OD flows is $\mathbf{X} = (\mathbf{X}_1, \ldots \mathbf{X}_T) = (x_{i_1 j_1 1}, \ldots, x_{i_1 j_J 1}, \ldots, x_{i_1 j_1 T}, \ldots, x_{i_1 j_J T})$. This linear mapping between the link flows and the OD flows is indeed the first term in the Taylor expansion of the relationship between link flows and OD flows, where additional terms capture the assignment matrix's sensitivity to changes in the OD flows, path choice and congestion propagation effects, as shown in Frederix et al. (2011, 2013). Let $\tilde{x}_{ijr}$ be in the neighborhood of $x_{ijr}$. Then, the Taylor expansion is

$$
y_{lt} = \sum_{(i,j) \in N} \sum_{r=1}^{t} a_{ijr}^{lt}(\tilde{\mathbf{X}}) \tilde{x}_{ijr} + \sum_{(i,j) \in N} \sum_{r=1}^{t} \frac{\partial y_{lt}(\tilde{\mathbf{X}})}{\partial x_{ijr}} (x_{ijr} - \tilde{x}_{ijr}) + \cdots = \tag{1.19}
$$

$$
= \sum_{(i,j) \in N} \sum_{r=1}^{t} a_{ijr}^{lt}(\tilde{\mathbf{X}}) \tilde{x}_{ijr} + \sum_{(i,j) \in N} \sum_{r=1}^{t} \left[ \frac{\partial \left[ \sum_{(i,j) \in N} \sum_{r=1}^{t} a_{ijr}^{lt}(\tilde{\mathbf{X}}) x_{ijr} \right]}{\partial x_{ijr}} \right] (\tilde{\mathbf{X}}) (x_{ijr} - \tilde{x}_{ijr}) + \cdots
$$

One second-order approach for solving DODME is Toledo & Kolechkina (2013). However, although theoretically the second order terms apparently could bring more details to the dynamic aspects, they require a more complex numerical optimization procedures (e.g., Armijo rules to compute the step length) that require more computational effort, while not showing a significant improvement in the quality of the results.

A GLS-based quasi-dynamic OD flows estimator was proposed by Cascetta et al. (2013). It uses traffic counts, under the assumption that OD matrices shares are constant across a reference period, whilst total flows leaving each origin vary for each sub-period within the reference period. The advantage of this approach over conventional within-day dynamic estimators is that of reducing drastically the number of unknowns given the same set of observed time-varying traffic counts.

### 1.3.2   Extensions to dynamic formulations: Simulation-based optimization approaches

The extension to the dynamic traffic assignment naturally increases the complexity of the traffic assignment. Due to the increase in the number of variables, stochasticity, congestion, and the MSA iterative procedure that is used to launch the assignment, the analytical objective function becomes unavailable. Therefore, unless some additional assumptions are considered as shown in previous Section 1.3.1, the

gradient cannot be calculated analytically, and the significant increase in variables makes it computationally inefficient to calculate it numerically. This has led to the growth in recent years of using stochastic algorithms and simulation-based optimization methods for dynamic traffic models.

Moreover, and as already mentioned, the possibility of including new traffic measurements makes these approaches very suitable for emerging *information and communication techonlogies* (ICT) methods for collecting data. To see some examples of dynamic traffic assignment models, the use of other traffic measurements, and SO approaches, these can be found in Cipriani et al. (2011), Cantelmo, Cipriani, Gemma & Nigro (2014), Bullejos et al. (2014), Antoniou et al. (2016), Kostic et al. (2017).

## 1.4  Thesis environment

The work of this thesis is a collaboration between the German company PTV Group and the Universitat Politècnica de Catalunya, with an agreement under the program *Doctorats Industrials* of the Catalan Government. This is a research program that partially funds the three participants (PhD student, University and Company) in order to motivate innovation and research in the Catalan industry environment. The collaboration should disruptively solve a problem that the company has and the university must address and validate the solution, that must be innovative and fulfill the requirements of a PhD thesis.

### 1.4.1  PTV Group

PTV Group[1] is a simulation software company founded in 1979 in Karlsruhe, Germany. His main product is a macroscopic and mesoscopic simulator called Visum, PTV AG (2020). In the context of the increase of use of the dynamic models, Visum offers a Simulation-based Assignment (SBA) based on Mahut (2000). Transportation planning and simulation models are widely used by traffic authorities and planners as a decision support tool and scenario assessment and the inputs of such models are crucial to accurately reflect reality. This is the main reason why traffic simulation software companies are investing in research on procedures to improve the generation of suitable data inputs for a proper operation of the simulation models, which are the appropriate OD matrices, that identify the dynamic mobility patterns.

The Innovation department of PTV is based in the headquarters of Karlsruhe and directed by Dr. Klaus Nökel that, together with the collaboration of Dr. Jaume Barceló, co-advisor of this thesis, opened in 2017 an Innovation and Research Center in Barcelona. The main objective of this delocalized subsidiary is to develop solutions for the most appealing long-term objectives of the Visum platform.

The first problem to handle is the one of this thesis, which consists of finding ways to improve the OD estimation process usually supplied by conventional technologies (e.g. inductive loop detectors, radar...) and now in combination with traffic-related measurement supplied by ICT applications (e.g. Bluetooth, GPS, Smartphones...) and its main objectives will be to enhance the knowledge in the mathematical models to successfully develop new methods for the estimation of dynamic Origin-Destination (OD) matrices that the new technologies make available.

From the perspective of a simulation modeling software company, the OD matrices are crucial for the transportation analysis models, both in the strategic planning models and operational ones. The OD estimation problem is an appealing and complex problem that, due to its underdetermination can lead to different solutions, depending on the initial OD matrix. Moreover, the complexity of the dynamic traffic

---

[1] www.ptvgroup.com

assignment produces a non-analytical objective function. The addition of ICT traffic measurements aims to reduce the underdetermination of the problem. This company objective implies the study of the current methods that are used in both static and dynamic assignment, learning from them and checking their computational and convergence properties, the quality of the solution that they provide and advantages and disadvantages of these methods when new ICT data is included in the statement of the problem formulation.

### 1.4.2   Universitat Politècnica de Catalunya

This thesis has been carried out in the Department of Statistics and Operations Research[2] at Universitat Politècnica de Catalunya, in Barcelona, Spain. This department is a multi-focus research department related to the mathematics fields of statistics and operations research and is internationally recognized for their works in statistical clinical and trial analysis, supply chain processes, simulation and optimization. Among these areas, there is a group of professors that has wide and recognized expertise in transportation problems related to traffic data analysis and simulation.

The line of research followed by this thesis began with the works in Bullejos et al. (2014) related to Bluetooth data and Kalman filtering to address the OD estimation by Barceló et al. (2013). Followingly, the research group was involved in the MULTITUDE cost action study, Antoniou et al. (2016), of the simulation-based optimization methods, relying on the use of the SPSA method. From the university department, this thesis is under the supervision of Dr. Lídia Montero, associate professor in transportation analysis and modeling for more than 20 years.

### 1.4.3   Previous work of this thesis

The PhD candidate developed his Master's thesis within the same research group and coadvisored by the same professors, Dr. Jaume Barceló and Dr. Lídia Montero. The master's thesis was a first approach to the simulation-based optimization approach to address calibration methods for traffic simulation models. In this work, we used the SPSA algorithm to properly calibrate the car following parameters of a microscopic model of a Bluetooth and radar sensorized motorway in Sweden. The successful results were presented in EWGT 2017 and published in Ros-Roca et al. (2017) and Ros-Roca (2017).

This Master's thesis was also awarded as the *Best Master's thesis for Transport Infrastructure in Spain 2017* by the Abertis Chair of Transportation[3].

## 1.5   Thesis motivations and objectives

The main objective of this thesis is to use ICT traffic measurements provided by commercial vendors for improving the OD estimation process in Visum software not only from the computational point of view but also from the quality of the resulting estimated OD matrix.

As this thesis is under the Industrial Doctorate program, the results of this thesis are highly oriented towards producing a methodology that can be used by the customers that use Visum. Nevertheless, the theoretical aspects and the contributions to knowledge have been also considered and validated

---

[2]www.eio.upc.edu
[3]Abertis Press Note (www.abertis.com/en/press-room/press-releases/992)

under the supervision of the University. In this sense, we are focused on using real ICT data, such as the (pre-processed or not) GPS data provided by commercial vendors and that the customers can have access to them. Differently from the Bluetooth and WiFi data that require a detection layout design involving also infrastructure costs, the privacy policies of the commercial vendors that collect and distribute the GPS data do not permit to infer the origin and destination for each GPS trace (to preserve anonymity) nor the data set information (how is it collected, vehicle types...). Therefore, the high volume of these GPS data sets and the ease of generation make them appealing and interesting, aiming at accessing the implicit information regarding the traffic state. Moreover, the computational aspect of the OD estimation problem-solving process must be taken into account, analyzing each step and studying alternatives to reduce the computational burden.

PTV Group has 40 years of experience in the deployment of software solutions that are now commercially available. PTV Group was born from academic research groups and these deployed solutions are based always on the trend lines of research and updated with the current advancements. The Industrial Doctorate environment permits also to use of such deployed solutions and takes advantage of its availability to produce more rapidly and efficiently. In this sense, this thesis uses the SBA assignment for dynamic mesoscopic models and also the k shortest paths algorithm implemented in Visum.

Last but not least, it is well known that the existing OD estimation procedures are great in increasing the fitting between simulated and observed traffic counts. However, the drawback of this increase is usually the demand pattern structure. In this tessitura, we also focus on providing an OD estimation procedure that maintains the structure of the reference OD matrix. Since the customers of PTV Group are transport planners that aim to re-estimate past years' OD matrices that must be updated, we assume that a reliable reference OD matrix is available and we focus particularly on preserving such structure, which is the demand pattern of the network. In this sense, we always analyze the similarity between them in all experiments of the thesis.

## 1.6   Thesis outline

This thesis is organized into 7 chapters.

Chapter 1 introduces the OD matrix estimation problem and its importance in traffic simulation and presents how this problem has been historically addressed. Here, the motivation, objectives and contributions of this thesis are also explained. Chapter 2 explains the traffic data and its evolution from conventional traffic data from counting stations equipped with the usual sensors to the emergence of ICT traffic measurements. Chapter 3 discusses how to measure the quality of an estimated OD matrix after the OD estimation process since it is crucial to obtain estimates that are reflecting a realistic mobility pattern. Chapter 4 shows the advances, findings and contributions made regarding the simulation-based optimization procedures to solve the OD matrix estimation problem, using or not travel times coming from ICT data. In Chapter 5, we analyze carefully the analytical approaches for the dynamic models. In Chapter 6, we propose an alternative analytical approach derived from the GPS data, which is the base for an original approach in the data-driven estimation procedures of the dynamic assignment matrix. In Chapter 4, 5 and 6 we show a complete set of experiments that support the conclusions of such chapters. Finally, in Chapter 7 we summarize the major contributions and outcomes from this thesis, concluding with further research and some final remarks.
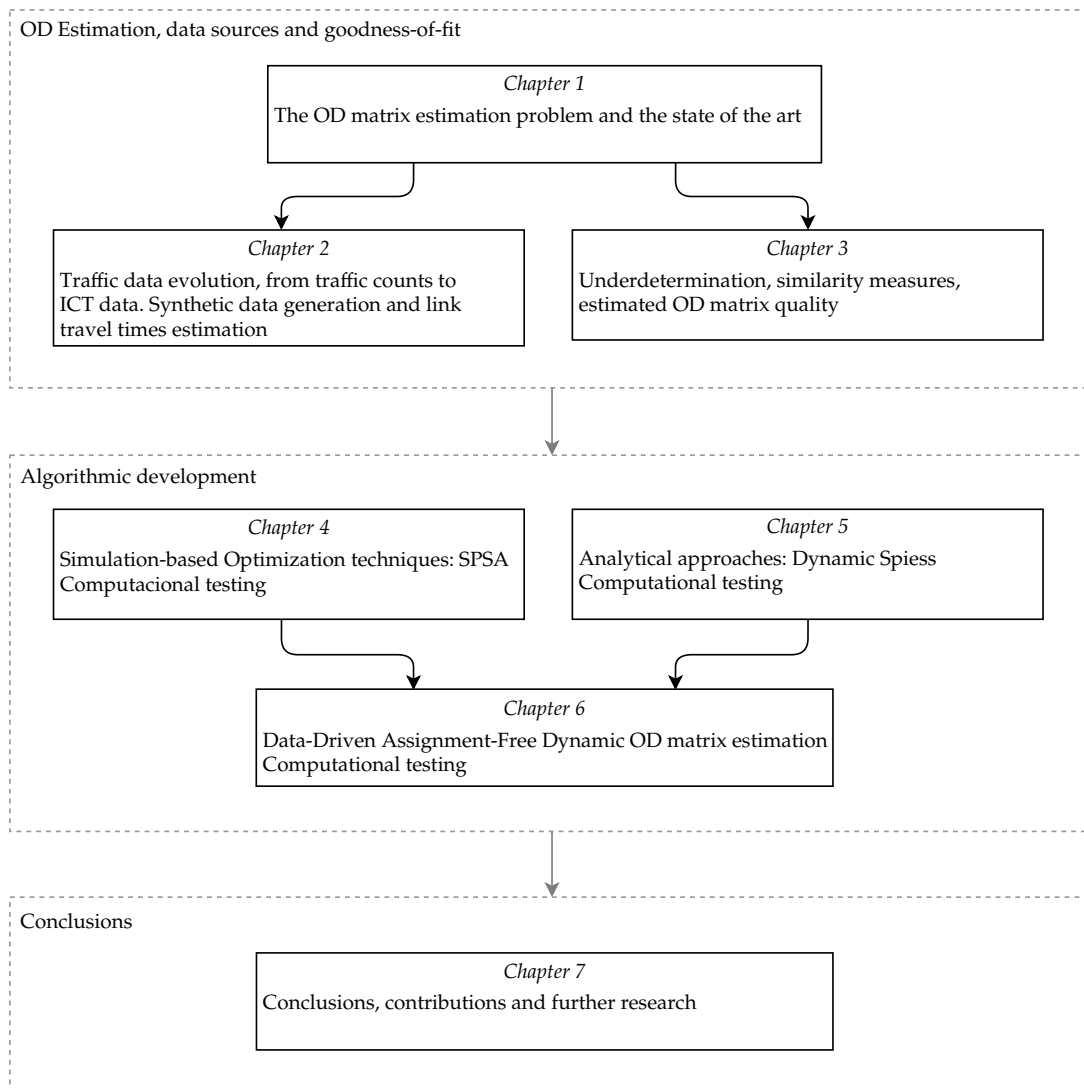
Figure 1.5: Thesis outline

## 1.7   Thesis Contributions

The main contributions of this thesis are listed below:

- In this thesis and all the related works, we always analyze and validate the obtained results using a goodness-of-fit measure based on matrix similarity to guarantee the reliable OD travel pattern, according to the demographics of the study area.

- The GPS data acquired from commercial vendors present some problems of cleansing, filtering and quality. Consequently, in order to be able of conducting a sensitivity-like analysis, we had to design in Chapter 2, a combined mesoscopic and microscopic framework to consistently generate synthetic data to test under controlled conditions the OD estimation process, producing traffic counts, GPS data and a historical OD matrix. This procedure emulates very closely the physical GPS data collection process.

- In Chapter 3, we propose a weighted MSSIM similarity measure based on the contribution of each origin and destination regarding the number of vehicles.

- In Chapter 4, we propose an enhanced SPSA procedure, with normalization of variables and an automatic parameters selection procedure.

- We also propose a new heuristic methodology to include maximal subpaths travel times in SPSA, that are extracted from GPS traces.

- In Chapter 5, we state the analytical formulation and implementation of the dynamic variant of the Spiess (1990) proposal to solve the dynamic OD estimation problem.

- We propose a data-driven assignment-free dynamic OD matrix estimation in Chapter 6. This new proposal combines GPS data, traffic counts and the historical OD matrix to find an estimate of the OD matrix. In this proposal, we replace the dynamic traffic assignment by a process of the GPS data, to reduce the computational burden.

- At the end of Chapter 6, we propose a set of stopping criteria that aimed to get the best quality of the estimated OD matrix, using a measure of similarity, rather than objective function convergence criteria.

## 1.8   Thesis Publications

**Journal articles**

Ros-Roca, Xavier; Montero, Lídia; Barceló, Jaume; Nökel, Klaus; Gentile, Guido (2020). "A practical approach to Assignment-Free Dynamic Origin-Destination Matrix Estimation problem". *Submitted to Transport Research Part C: Emerging Technologies*.

Ros-Roca, Xavier; Montero, Lídia; Barceló, Jaume (2020). "Investigating the Quality of Spiess-Like and SPSA approaches for Dynamic OD Matrix Estimation". *Transportmetrica A: Transport Science, Vol. 17(3), pp 235–257*.

**Peer reviewed conference papers**

Ros-Roca, Xavier; Montero, Lídia; Barceló, Jaume and Nökel, Klaus (2021). "Dynamic Origin-Destination Matrix Estimation with ICT Traffic Measurements using SPSA". To be published in *IEEE Proceedings for the 7th International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS 2021)*. $21^{st} - 24^{th}$ June 2021. Virtual due to COVID-19.

Montero, Lídia and Ros-Roca, Xavier (2020). "Using GPS tracking data to validate route choice in OD trips in dense urban networks". *Transportation Research Procedia, Vol. 47, pp 593–600*. 22nd Euro Working Group of Transportation, EWGT 2019, $18^{th} - 20^{th}$ September 2019, Barcelona, Spain.

Montero, Lídia; Ros-Roca, Xavier; Herranz, Ricardo and Barceló, Jaume (2019). "Fusing mobile phone data with other data sources to generate input OD matrices for transport models". *Transportation Research Procedia, Vol. 37, pp 417–424*. 21st Euro Working Group of Transportation, EWGT 2018, $17^{th} - 19^{th}$ September 2018, Braunschweig, Germany.

Ros-Roca, Xavier; Montero, Lídia; Schneck, Arne and Barceló, Jaume (2018). "Investigating the Performance of SPSA in Simulation-Optimization Approaches to Transportation Problems". *Transportation*

*Research Procedia, Vol. 34, pp 83–90*. 6th International Symposium of Transport Simulation and 5th International Workshop on Traffic Data Collection and its Standardization, ISTS-IWTDCS 2017, $3^{rd} - 6^{th}$ August 2018, Ehime, Japan.

Ros-Roca, Xavier; Montero, Lídia and Barceló, Jaume (2017). "Notes on Using Simulation-Optimization Techniques in Traffic Simulation". *Transportation Research Procedia, Vol. 27, pp 881–888*. 20th Euro Working Group of Transportation, EWGT 2017, $4^{th} - 6^{th}$ September 2017, Budapest, Hungary.

**Peer-reviewed presentation at Conferences**

Ros-Roca, Xavier (2021). "A Data Driven Approach to Dynamic Origin-Destination Matrix Estimation". In *2021 International Symposium on Transportation Data and Modelling (ISTDM 2021)*, $21^{st} - 24^{th}$ June 2021. Virtual due to COVID-19.

Ros-Roca, Xavier; Montero, Lídia; Barceló, Jaume; Nökel, Klaus (2021). "Dynamic Origin-Destination Matrix Estimation with ICT Traffic Measurements using SPSA". In *7th International IEEE Conference on Models and Technolgies for Intelligent Transportation Systems (MT-ITS 2021)*, $16^{th} - 17^{th}$ June 2021. Virtual due to COVID-19.

Ros-Roca, Xavier; Montero, Lídia; Barceló, Jaume; Nökel, Klaus; Gentile, Guido (2020). "Transport Analytics approaches to the Dynamic Origin-Destination Estimation Problem". In *3rd Symposium on Management of Future Motorway and Urban Traffic Systems (MFTS 2020)*, $6^{th} - 8^{th}$ July 2020. Virtual due to COVID-19.

Ros-Roca, Xavier; Montero, Lídia and Barceló, Jaume. "Transport Analytics approaches to the Dynamic Origin-Destination Estimation Problem". In *IV Campus Científico del Foro de Ingeniería del Transporte (FIT)*, $24^{th} - 26^{th}$ June 2020. Virtual due to COVID-19.

Ros-Roca, Xavier; Montero, Lídia and Barceló, Jaume. "Exploiting ICT Measurements in the Dynamic Origin-Destination Estimation Problems". In *III Campus Científico del Foro de Ingeniería del Transporte (FIT)*, $4^{th} - 5^{th}$ April 2019. Cercedilla, Madrid, Spain.

Ros-Roca, Xavier; Montero, Lídia; Schneck, Arne and Barceló, Jaume. "Investigating the quality of SPSA and Spiess-like approaches for Dynamic OD Estimation". In *Mathematics Applied in Transport and Traffic Systems (MATTS 2018)*, $17^{th} - 19^{th}$ October 2018. Technical University of Delft, The Netherlands.

Ros-Roca, Xavier; Montero, Lídia and Barceló, Jaume. "Avances en la investigación de PTV IBERIA en materia de estimación dinámica de matrices OD". In *XII Congreso de Ingeniería del Transporte Spain (CIT)*, $6^{th} - 8^{th}$ June 2018. Gijón, Spain.

# 2

# Traffic data: From traffic counts to ICT traffic measurements

*This chapter focuses on the evolution of the traffic data from conventional traffic counts to the new paradigm of ICT devices, that permit to collect information about the vehicles at different places through time. These new devices open the range of traffic data to path reconstruction, and estimation of link and path travel times. At the same time, we propose a framework to synthetically generate consistent traffic counts, GPS and historical OD matrix data sets to build a synthetic experiment to test the OD matrix estimation procedures. At the end of the chapter, we also design and test a methodology to extract the information from GPS data and transform it into estimates of link travel times.*

Motorized vehicles have become a key transportation mode that supports individual mobility, which has led to a continuous increase in traffic flows on road networks. Thus, it is necessary to develop applications that increase the efficiency of road networks and diagnose problems while trying to avoid, alleviate, and solve these problems in a variety of circumstances. Planning and designing road networks have been the strategic tools for achieving these objectives since the beginning, and traffic control and management systems are the operational tools for managing traffic systems in real-time. Acting on a system requires acquisition of knowledge about the system. Such knowledge is usually generated from the observational data of variables that characterize the state of the system, which means that many sensors are needed to capture these data. The advent of *information and communication technologies* (ICT) and their pervasive dissemination make it possible to acquire new and unprecedented amounts of data, thus enabling the generation of new, richer knowledge. This in turn makes more efficient ways of controlling and managing the traffic network possible, as well as the possibility of translating the knowledge into information that, when properly conveyed to users, can help them use the network better and change their behaviors. These are the objectives of the intelligent transport systems that are made possible by ICT applications.

## 2.1 Conventional traffic data

In the case of transportation systems, traffic data is usually collected by sensors that gather different measures describing the traffic and vehicle conditions in certain places on the network. There are many

sensors that measure different physical variables, and they can be placed in vehicles or installed in the pavement and other infrastructures in the urban network.

In traffic management, one useful measure is the traffic volume on different roads in the network. Installing these sensors allows monitoring the traffic conditions in specific zones. Depending on how they are installed in the infrastructure, the sensors can be divided into:

- *Intrusive sensors*: These are installed across the pavement surface. Installing road sensors thus implies traffic disruption during their installation as well as high maintenance and repair costs. On the other hand, their advantage is that they are highly accurate—when well calibrated—at detecting vehicles, which is why they have been widely implemented over the years. These high costs and implications require that their installation usually be strategic, not only for the specific purpose of OD estimation, but also for other traffic management purposes.

- *Non-intrusive sensors*: These are installed in other places on the roads, such as atop a mast. These sensors are also expensive, but their installation does not imply disruption. They are less accurate and sensitive to weather, environmental conditions, and interference from other objects.
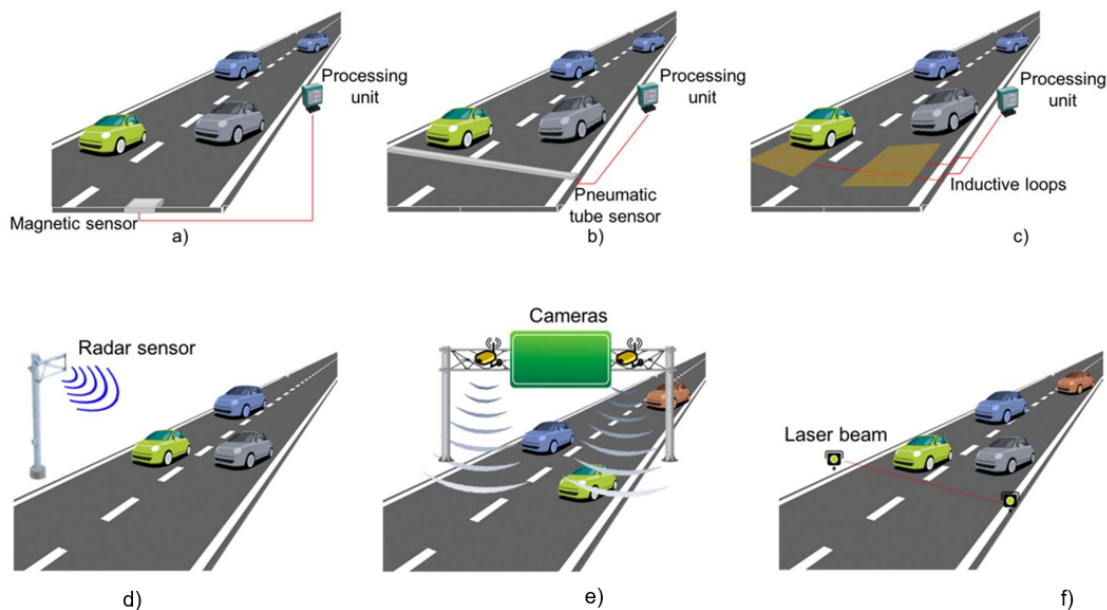


Figure 2.1: (a), (b) and (c) are intrusive sensors and (d), (e) and (f) correspond to non-intrusive sensors. Image taken from Guerrero-Ibáñez et al. (2018)

Figure 2.1 and Table 2.1 are taken from Guerrero-Ibáñez et al. (2018), and they describe the different intrusive and non-intrusive sensors that measure traffic counts on the network. All these sensors detect vehicles crossing the studied road in the network, and they aggregate the counts for a certain time period. The most used sensor is the inductive loop detector (ILD), which consists of a long wire coiled into a loop installed under the road surface. It measures changes in the electrical properties of the circuit when a vehicle passes over it, producing an electrical current due to the magnetic field perturbation. However, they need to be recalibrated regularly, a policy that is not always possible. Furthermore, according to the experience shared by most practitioners, a significant amount of detectors (i.e., about 20% of all installed) are either out of service or require recalibration.

Table 2.1: Categories of sensors currently used for traffic management. Table taken from Guerrero-Ibáñez et al. (2018)

| Category | Sensor Type | Application and Use |
| --- | --- | --- |
| Intrusive | Pneumatic road tube | Used for keeping track of the number of vehicles, vehicle classification and vehicle count. |
| | Inductive Loop Detector (ILD) | Used for vehicle's movement, presence, count and occupancy. The signals generated are recorded in a device at the roadside. |
| | Magnetic sensors | Used for detection of presence of vehicle, identifying stopped and moving vehicles. |
| | Piezoelectric | Classification of vehicles, count vehicles and measuring vehicle's weight and speed. |
| Non-intrusive | Video cameras | Detection of vehicles across several lanes and can classify them by their length and report vehicle presence, flow rate, occupancy and speed for each class. |
| | Radar sensors | Vehicular volume and speed measurement, detection of direction of motion of vehicles and used by applications for managing traffic lights. |
| | Infrared | Application for speed measurement, vehicle length, volume and lane occupancy. |
| | Ultrasonic | Tracking the number of vehicles, vehicle's presence and occupancy. |
| | Acoustic array sensors | Used in the development of applications for measuring vehicle's passage, presence and speed. |
| | RFID (Radio-Frequency identification) | Used to track vehicles mainly for toll management. |

However, these conventional sensors are not capable of capturing the network's mobility patterns, which therefore must be inferred. According to Cascetta (2001), the data available from traffic counts is suitable for estimating OD matrices. Assuming that traffic flows on the links of a road network are the result of assigning an OD matrix to it and that these traffic flows are measurable by traffic counting stations, then the problem of finding the OD matrix where they originated can be considered the inverse of the traffic assignment problem. For instance, one can appreciate in Problem 1.6 that traffic counts on certain links in the network play a decisive role in the OD estimation problem.

All these previously mentioned sensors must be installed on the network. Furthermore, they also need calibration, maintenance, and repairs in order to provide precise measurements of traffic counts. All these costs require deciding on the location and number of traffic measurement stations in order to place them strategically. Several studies have been conducted on the optimal placement of such sensors for maximizing network coverage and observability while also minimizing the number of stations in order to reduce the cost of the whole detection layout. The detection layout is strongly related to the

observability of the system (Castillo et al. (2008)). Some heuristic techniques for finding a suboptimal detection layout can be found in Barceló et al. (2012).

## 2.2   The emergence of ICT and traffic measurements

*Information and communications technologies* (ICT) have opened the range of traffic measurements that can be used in intelligent transportation systems, because the communication between sensors provides new insights and information about the traffic phenomenon on the studied network. The main advantage that should be highlighted is vehicle identification, by which one vehicle can be detected in two different places using sensors connected in the network infrastructure. Moreover, connected devices in vehicles, such as GPS devices or mobile phones, provide geolocation data, which must be post-processed to obtain different measures of their trip. These data are different from conventional traffic data, which measures a certain variable at a fixed point in the network. This new technology allows collecting spatio-temporal information about the traffic conditions.

A large class of ICT detection devices is known as *space-based*. These are devices that capture the identity of a vehicle or an on-board device in the vehicle. Some examples are license plate recognition, Bluetooth/Wi-Fi antennas, and tag readers. The vehicle or device is identified at the point where the detector is located, and reidentified downstream by another detection device. Moreover, the forthcoming road side units (RSU) also dialogue with the equipped vehicle passing through its detection range. However, these are different from inductive loop detectors that capture all the vehicles, as some studies have shown that only about 30% of all vehicles are visible (Daamen et al. (2014)). Another example can be found in public transport in the form of smart cards that record the origin and destination of their user's trips.

Despite the promise of this non-conventional data to supply much information and new analyses that gain a better understanding of the transportation system, different methods and techniques for processing and modeling are needed to obtain the desired information. Those techniques are different from those used in the case of the previous sensor data, that is gathered with an intended purpose. The ICT sensors usually do not require active and controlled solicitation. Moreover, the large size and continuous generation makes such data challenging. Some examples of working with these data for OD estimation are Mo et al. (2020), who use license plate recognition technology, and Barceló et al. (2013), who use Bluetooth pairing information when a vehicle is sequentially captured by two antennas.

The following section defines another class of ICT detection devices, the *time-based* devices, which capture the evolution of the same vehicle's location instead of the visible vehicles at a certain location. This thesis will focus on this class of data. This classification between space-based and time-based devices was proposed by Nanthawichit et al. (2003).

## 2.3   Time-based ICT devices

Time-based devices are able to sequentially capture the location of a vehicle over time. These are usually external devices inside a vehicle that track the device's geoposition and thus that of the vehicle. The main examples of this class are probe vehicles designed for tracking their own trajectory, GPS devices, and most recently drivers' smartphones that passively record their locations. In recent years, probe vehicles (or floating car data) have been poorly used because of the growth of smartphones and GPS devices that

spontaneously capture vehicle locations. As in the space-based class, vehicles captured by these devices are obviously only a sample of the total population circulating on the network. Nevertheless, when a huge amount of data is gathered on these vehicles circulating on similar days (for example, during rush hour on workdays), the traffic state can be inferred using data analysis techniques that overcome the drawbacks and limitations of conventional transportation data collection methods, such as household surveys and conventional traffic measurement stations.

Nowadays, the main producers and owners of these data sets are the Netherlands-based commercial vendor TomTom in the case of GPS units and, in the case of probe vehicles, INRIX from the United States and HERE from the Netherlands. In the case of smartphones, the data come from the phone operators and commercial providers who buy and process these large sets of geopositioning data, Antoniou et al. (2019).

These data sets provide information on the mobility of many individuals throughout the network and at high temporal resolution, which in some cases allows either tracking the trajectory of the vehicle or at least correctly inferring the route choice. However, the time latency of each individual depends on the particular device and commercial policy, which implies a great amount of heterogeneity in the latency (Chen et al. (2016)). Therefore, from a generic perspective, one can neither assume that the positioning frequency is uniform in the data set nor that there is a unique route between two consecutive waypoints. Moreover, in the case of mobile phone data (also called *call detail record*; CDR), positioning the device can be imprecise, since it is uniquely determined by triangulation between nearby cell towers. In this case, at least three cell towers are required for capturing distance to the device, which makes this more precise in urban networks. However, many factors related to the antenna's technology significantly affect the precision of the CDR data.

These difficulties in finding the exact vehicle locations based on the phone's data limit the use of such data sources for traffic analysis. In fact, the CDR data allow locating the devices within the range of the antennas, which allows obtaining global information (meaning all transport modes and purposes) regarding the mobility of the devices at a certain time of the day, but they do not allow reconstructing the trajectories of such mobile phones. Moreover, they do not inform us of the purpose or mode of transport. Thus, mobile devices can be used for analyzing global mobility patterns and for calculating scaling factors that convert the obtained OD matrix into a population-level count, thereby providing a final estimate. Chen et al. (2016), Calabrese et al. (2011) and Alexander et al. (2015) used a census-based ratio for each different zone, but they observed biases in zones with low penetration rates. Ma et al. (2012) and Iqbal et al. (2014) took other interesting approaches that scale the obtained OD matrices using external procedures, such as traffic assignment and optimization steps.

Although both mobile phone and GPS data sets track vehicles circulating through the network, the devices do not report the purpose of the trip or mode of transport. Some inference can be made by studying each single user and their different days. For example, Phithakkitnukoon et al. (2010) and Alexander et al. (2015) infer that the most frequently visited weekday locations during the day and at night are home and work. Moreover, one can use CDR data to infer the purpose of the trip through contextual information such as activity locations and service hours at points of interest (POIs) in the network, as done in Xie et al. (2009), Huang et al. (2010) and Chen et al. (2010). Another interesting approach when using CDR data for OD estimation is the work on Caceres et al. (2013), that uses the cell changing of the devices as traffic counts on the link that crosses the boundary of two adjacent cells.

On the other hand, inferring the mode of transport from tracking requires more sophisticated methods. For example, by projecting the points obtained from the mobile phone data onto the supply network of different transport modes, one can therefore decide which mode of transport is used, although this has

been tested successfully only for long distances and in low density areas. Moreover, some studies have analyzed travel speeds and compared them to those of each transport mode, as well as whether there is a station at the beginning or end of the trajectory (Wang et al. (2010)). Montero et al. (2019) combines the information with other data sources to extract the mode of transport.

In summary, GPS data and the former use of probe vehicles produce more precise data sets by accurately locating ICT devices. An example of a data set produced by this class of devices is shown in Table 2.2, where the ID defines each different device and its latitudinal and longitudinal position at each timestamp. Each row of the data set is usually called a waypoint, which takes the form $(ID_k, ts_{k,l}, lat_{k,l}, long_{k,l})$, where $k$ denotes the trip and $l$ is the ordered $l$-th waypoint of the trip $k$. From now on, we will refer to these data sets as either a *waypoint data set* or *waypoint database*.

Table 2.2: An example of a waypoint data set

| ID | Date | Timestamp | Latitude | Longitude |
|---|---|---|---|---|
| 4261353 | 2019-11-30 | 07:43:58 | 45.445988 | 9.1244048 |
| 4261353 | 2019-11-30 | 07:44:11 | 45.445496 | 9.1241952 |
| ............. | ................. | ............. | ................. | ................. |
| 4261353 | 2019-11-30 | 07:45:08 | 45.444767 | 9.1192517 |
| 4261355 | 2019-11-30 | 07:45:02 | 45.445980 | 9.1247048 |
| 4261355 | 2019-11-30 | 07:45:23 | 45.445574 | 9.1192821 |
| ............. | ................. | ............. | ................. | ................. |
| 4261355 | 2019-11-30 | 07:46:56 | 45.444767 | 9.1197541 |

In this thesis, we will assume that the available ICT traffic data is like what is shown in Table 2.2, which is typical of what comes from GPS devices that are more precise reporting the exact latitudinal and longitudinal position.

### 2.3.1   Time-based devices for OD estimation

The OD matrix estimation problem is a bi-level minimization problem based on traffic counts, as shown in Equation 1.6. It is a highly underdetermined problem, mainly because the number of variables (i.e., the OD flows) is much larger than the number of available link traffic counts in urban networks. Furthermore, a determined problem cannot be ensured even by a full detection layout that acquires traffic counts for all links in the network. Therefore, many researchers have studied alternative methodologies and how to exploit different data sources.

Van Aerde et al. (1993) and Eisenman & List (2004) made use of the data generated by probe vehicles circulating through the network. As Eisenman & List (2004) stated, the question that arises is how the available probe data can help overcome the underdetermination of the OD estimation problem. In the case of probe vehicles, the assumption of full visibility is widely accepted, which means that the origins, destinations, time departures, and paths of all tracked vehicles are known. In all these cases, a simple count of probe vehicles and traffic counts for all the links provide estimates of the detection technology's penetration rates, which in turn can be used intelligently to improve the OD matrix estimation problem. However, biases can appear if the probe vehicles are not distributed similarly to the overall population of vehicles.

On the other hand, the tracking data sets of GPS devices and smartphones present more problems in terms of the previously mentioned assumptions. In these cases, the waypoint data sets contain millions of trips taken by anonymized users throughout the network every day over a long time period that is usually more than a month and around a year. In the case of GPS devices, it is acceptable to assume that the waypoint sequences pertain to on-road vehicles, but there is no reliable information regarding their origin, destination, or travel purpose. For instance, last-mile fleet vehicles are usually equipped with GPS devices, but they do not reflect real origin-to-destination trips. And smartphone tracking cannot determine whether these trips pertain to private transport, pedestrians, or public transport. For such cases, advanced analytics processes have been designed to infer the key aspects, origin, destination, time departure, purpose, and mode.

Despite the drawbacks, a global OD matrix can be obtained by map-matching these millions of waypoints and constructing paths between the different OD pairs using GPS or, as already mentioned, by observing the change of regions in a mobile phone data set. Added value can be used to differentiate these trips according to purpose and type of day, as in Alexander et al. (2015).

By agreement with data providers, many researchers have designed a data collection process that gathers only private vehicle tracking from GPS devices or mobile phones during time periods on days with similar traffic conditions, for example by analyzing the morning rush hour for each weekday. Although these trips are only a sample of the total trips in the network on a large scale, they can represent mobility patterns for the selected time period on days with similar characteristics. In such cases, the assumptions gathered from probe vehicles about origins, destinations, and purposes are also accepted, by which a simple procedure can map the initial and ending waypoints of each trajectory onto the zonification of the network, thus obtaining the origin and destination of each.

However, all the cited researchers agree on the fact that these OD matrices are initial approaches that must be compared with other data sources like traffic counts to validate their accuracy, even when the data collecting process has been designed. This requires resorting to very specific data analysis techniques, given the huge amount of recorded data. Gundlegård et al. (2013) and Jiang et al. (2016) provide good examples of this data processing to extract OD matrices. Ma et al. (2012) suggest that the obtained OD matrix is a good candidate for use as an initial OD matrix in classic OD estimation procedures, such as those described in Chapter 1.

It is a fact that waypoint data sets provide more information than a household survey and its posterior processing, because they capture the network movement of many vehicles, human mobility patterns, and the traffic state of the studied area. However, commercial data sets have an uncontrolled data collection process whose accuracy and quality cannot be assumed. In fact, biases could be present in data sets on anonymous vehicles due to detecting different samples of vehicles for different OD pairs. Other problems with anonymized data are poor quality (i.e., low latencies that impede unequivocal reconstruction of trajectories) and different vehicle purposes (such as fleet vehicles) that must be filtered. These characteristics must be taken into account for specific uses because, if not, they can lead to wrong conclusions. Moreover, given the commercial nature of these data sets that are gathered and sold by private companies, as well as privacy rules, the validation and comparative analysis of the data set quality is usually limited (Antoniou et al. (2019)).

To conclude, waypoint data sets have proven to be increasingly interesting to researchers and practitioners seeking to obtain new and different information on the traffic conditions in a study area. Using these sets of data requires that one at least clearly understand the data collection process, which is responsible for the quality, accuracy, and possible problems in the data set. Once the data has been analyzed, the data must be pre-processed, cleansed, and filtered for its specific purpose and use.

## 2.4    A procedure for generating synthetic GPS data

In most papers that deal with DODME by adding richer ICT traffic measurements to the conventional link flow counts (Yang et al. (2017), Krishnakumari et al. (2020), Mitra et al. (2020)), the authors assume specific conditions for controlling the data collection processes proposed by Lopez, Krishnakumari, Leclercq, Chiabaut & van Lint (2017), Lopez, Leclercq, Krishnakumari, Chiabaut & van Lint (2017). These conditions for collecting the data ensure their quality and allow making assumptions that form the basis of the approaches. This is not always possible with commercial data, because the researcher has no access to it, the fleet size is very limited, or the only available data are supplied by commercial companies who prohibit access to the raw data that they instead pre-process, depending on their business model. Therefore, it is common to conduct simulation experiments that emulate reality, which is then mimicked by generating synthetic data. Antoniou et al. (2016) provide an experimental framework that has been widely used by researchers. However, they use specific microscopic simulation models and numerical software (i.e., MATLAB). This is why we propose a synthetic and agnostic software data generation process that fulfills the functional requirements.

This thesis proposes a procedure for synthetically generating data sets in order to computationally test any network. The scheme in Figure 2.2 represents the synthetic data generation. As it is explained below, the two data sets have been produced emulating the real process of collection in real word, so it validates the procedure for computationally generating the different traffic measures. From a ground truth OD matrix, $\mathbf{X}^{\mathrm{GT}}$, and using both a mesoscopic and microscopic model of the same network, the three needed data objects are generated:
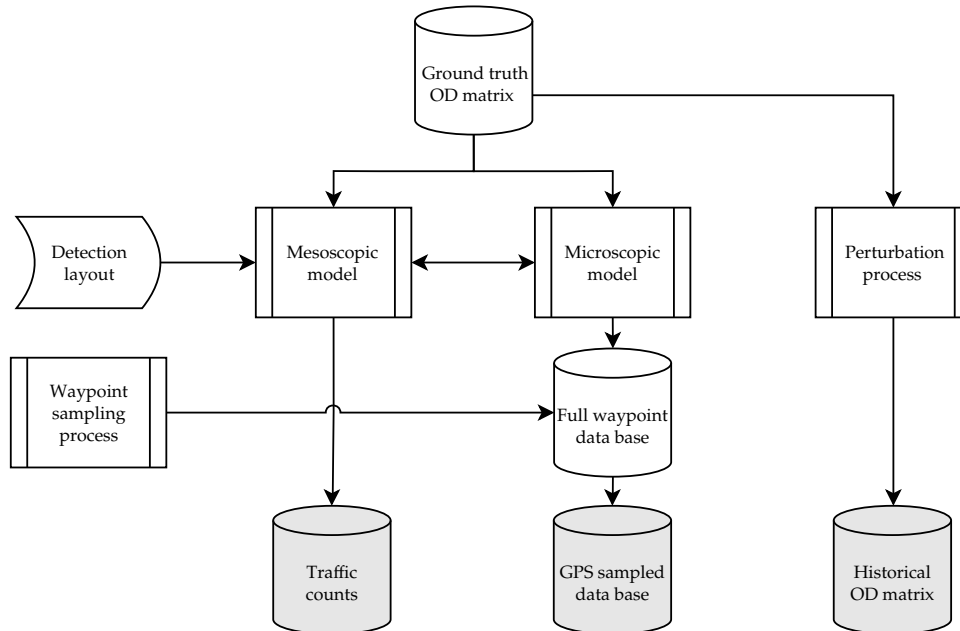


Figure 2.2: Methodological scheme of the synthetic data generation for computational testing

- The traffic count data set, $y_{lt}$, is generated by inductive loop detectors installed in the network, which tally the total number of vehicles that cross them in a fixed time period.

- The GPS data set comprises the trajectories of a sample of vehicles circulating in the network.

■ The historical OD matrix, $\mathbf{X}^H$, which acts as a reliable OD matrix that could be obtained from a household survey or from a past project that must be updated.

The three data source outputs in this methodological scheme are the data sets this thesis needs to investigate how to incorporate the GPS data set into the OD matrix estimation problem.

### Traffic count data set

In real life, the counting stations are inductive loop detectors in the network. In practice, traffic counts are commonly obtained by averaging traffic counts from a few days under the same traffic conditions, since these are fairly stable counts. Therefore, to obtain the traffic count data set, a dynamic traffic assignment uses the mesoscopic model with the ground truth OD matrix to produce link flow values through simulation, measured in vehicles per hour (veh/h). The data set contains the ID code of the detector and the flow captured for a certain time period. An example is shown in Table 2.3.

Table 2.3:  An example of a traffic count data set

| ID | Initial Time | Ending Time | Flow |
|---|---|---|---|
| 35 | 07:30:00 | 07:45:00 | 80.125 |
| 41 | 07:30:00 | 07:45:00 | 75.852 |
| 47 | 07:30:00 | 07:45:00 | 83.147 |
| 53 | 07:30:00 | 07:45:00 | 52.761 |
| ...... | .............. | .............. | ............ |
| 35 | 07:45:00 | 08:00:00 | 124.547 |
| 41 | 07:45:00 | 08:00:00 | 112.365 |
| 47 | 07:45:00 | 08:00:00 | 183.787 |
| 53 | 07:45:00 | 08:00:00 | 102.643 |
| ...... | .............. | .............. | ............ |
| 35 | 08:00:00 | 08:15:00 | 127.741 |
| 41 | 08:00:00 | 08:15:00 | 102.654 |
| 47 | 08:00:00 | 08:15:00 | 164.512 |
| 53 | 08:00:00 | 08:15:00 | 174.102 |
| ...... | .............. | .............. | ............ |

In a real network, suitably placed counting stations collect the observed traffic counts in the network; therefore, the detection layout (i.e., the detector placement) is another aspect that must be considered when generating the synthetic data. In this case, we propose using any heuristic methodology that satisfies some desired criteria. An optimal scenario would maximize the observability of the network. Thus, we use the first phase of Barceló et al. (2012)'s detection layout procedure, which proposes an easily implementation greedy algorithm that finds a suboptimal solution for maximizing coverage of the OD demand in terms of link and path flows. This heuristic is an iterative process, as follows:

1) After a traffic assignment, compute the total flows on each link in the network.

2) While detectors remain for placement:

    a) Find the link with greatest flow and add a detector to it.

    b) For each path that used the link, set its flow to zero.

    c) Sum up and update the link flows, after (b).

3) Return to step 2.

This greedy algorithm finds a suboptimal detection layout because it is always placing a detector on the link that captures the most after removing the already captured flow. One can stop the procedure once a desired number of sensors have been placed or after capturing a percentage of total flow or of OD.

## GPS data set

The properly calibrated dynamic traffic assignment model is usually supported by a mesoscopic simulation platform that emulates flow propagation with a given OD matrix, which will be used as the ground truth OD matrix, $\mathbf{X}^{GT}$. This can be imported into microscopic modeling software that allows tracking individual vehicles circulating throughout the network. This software uses the Vehicle Tracking Procedure tool to generate vehicle-tracking data that are similar to those that are physically collected from GPS devices. They are in the same format shown in Table 2.1.

In our case, we opted to generate a sample with a uniform penetration rate, meaning that each OD pair has the same penetration rate for the GPS technology. On the other hand, we use an empirical distribution of latencies for our synthetic experiments. In other words, we used GPS data from a past project from which we obtain the empirical distribution of latencies using all the vehicles in the sample and then assigned a uniform latency to each vehicle, following the mentioned distribution.

## Historical OD matrix

The historical OD matrix can be generated with multiple different perturbations of the ground truth OD matrix, depending on the desired degree of similarity. This thesis obtains it from the ground truth OD matrix by following Antoniou et al. (2016)'s MULTITUDE procedure, that is:

$$x_{ijr}^{H} = x_{ijr}^{GT}(p + q \cdot \varepsilon_{ijr}) \ , \ \forall i \in I, j \in J, r \in T \tag{2.1}$$

where $p, q$ are two parameters and $\varepsilon_{ijr} \sim N(0, t), t \in (0, 1/2)$. These parameters $p, q \in (0, 1)$ and $p > q$ decrement the historical OD matrix relative to the ground truth value using a random perturbation. This perturbation tries to emulate a realistic historical OD matrix from surveys and past projects with similar traffic conditions, although the emulation is certainly not identical to the ground truth OD matrix.

Taking an overall perspective of the methodology, the consistency between the three generated data sources is ensured. Traffic counts are generated using a mesoscopic model; a microscopic model generates GPS tracks through simulation replicas; and vehicles are randomly sampled on different days under similar traffic conditions. This realistically emulates the actual data collection process. Finally, the historical OD matrix is generated by perturbing the ground truth OD matrix in order to preserve a high degree of reliability. Therefore, the proposed methodology generates these data sources in such a way that the synthetically generated data are indistinguishable from the physically measured data, making

them usable for computationally testing the proposed methods for estimating OD matrices when the physical data are neither available nor good enough for these purposes.

### 2.4.1 Applying the methodology to a synthetic network

The proposed methodology has been tested on many different networks, but we present here the results from applying it to the network shown in Figure 2.3, which is further analyzed to check the consistency of the synthetically generated data sets.
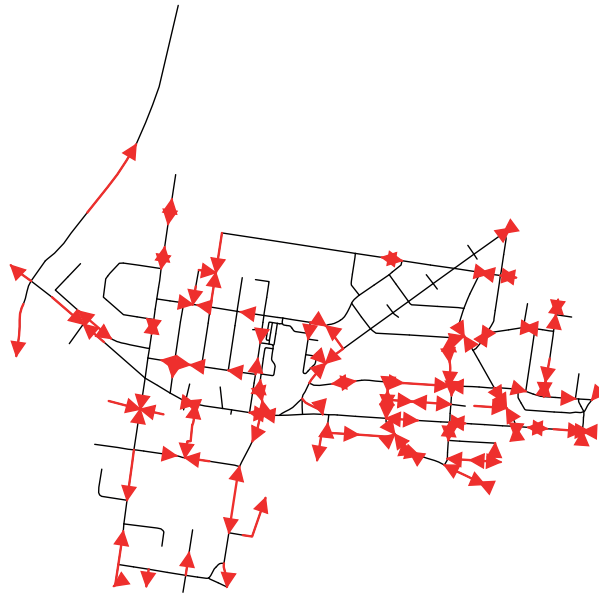
Figure 2.3: The network used to test the proposed synthetic data generation process. Red arrows indicate the installed counting sensors

This network was prepared using a ground truth OD matrix divided into time periods of 15 minutes. This OD matrix contains a total of 8300 trips, divided into the studied time periods. Both the mesoscopic and microscopic simulations run from 07:00 AM to 09:00 AM in order to have data from 07:30 AM to 08:30 AM (8 time periods) and from the preceding and posterior loading and unloading periods, respectively. With the preceding and posterior time periods, the total number of trips increased to 15126 trips.

The microscopic simulator generated GPS data for around 14870 simulated trips each day, which is similar to the ground truth OD matrix. Losses (around 2%) are attributed to the decimal numbers of the cells that cannot be generated in a microscopic simulation. That amounts to almost 3 million trips and almost 700 million waypoints for all 200 simulated days. With a 5% uniform penetration rate for all OD pairs, the number of trips decreases to 750 trips per day, which correspond to around 150k trips and 37 million waypoints. At this point, each trip is represented by waypoints every 0.1 seconds, so latency must be assigned to each vehicle.

As already mentioned, the empirical distribution of latencies was obtained from Montero & Ros-Roca (2020), and it corresponds to the distribution of latencies in the GPS sample of the city of Barcelona (Catalonia, Spain), as shown in Figure 2.4. This distribution shows that nearly 75% of the tracked vehicles have latencies below 6 seconds. Once all sampled vehicles are assigned to one of these intervals,

the specific latency is assigned uniformly. For example, if one vehicle is assigned to the interval [6, 15], then, a latency of between 6 and 15 seconds is assigned with equal probability. Therefore, once each latency is applied and the waypoints are filtered, the resulting database is reduced to around 44460 waypoints per day, meaning a total of around 9 million waypoints for all 200 days, with a penetration rate of 5%.
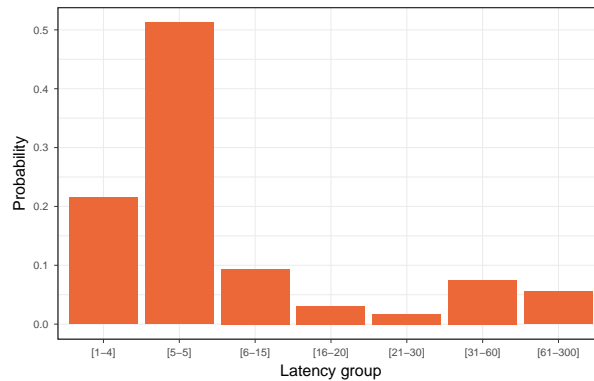


Figure 2.4: Empirical distribution of latencies (in seconds) of the GPS data from Barcelona

The detection layout was generated as explained before, using the heuristic of Barceló et al. (2012). A total of 40 sensors were placed according to a certain level of desired coverage. They cover 97.27% of the total ground truth flow, which is 69.64% of the OD pairs totally captured and 13.07% of those partially captured.

Finally, regarding the historical OD matrix, we have applied Equation 2.1, with $p = 0.75$ and $q = 0.15$. The historical OD matrix is decremented by an average of 25% with a random perturbation. This perturbation emulates a realistic historical OD matrix from surveys and past projects that represent highly similar traffic conditions. Effectively, the ground truth OD matrix has 8300 trips and the historical OD matrix 6232 trips.

## 2.5   A practical map-matching procedure for estimating link travel times from commercial GPS data

As it is, the GPS data (Table 2.2) cannot be used directly for transportation analysis. Nevertheless, it shows individuals circulating through the network that contain implicitly valid information regarding the traffic conditions. Moreover, as already mentioned, we assume that the commercial nature of the used data do not ensure reliability regarding neither the origin-destination information nor the overall demand pattern. In order to bring to light the information on the traffic conditions hidden in the GPS data, we propose map-matching these trajectories onto the network. Map-matching (explained in detail below) reconstructs the defined trajectories from the waypoint sequence and estimates the travel times at the link level by using the timestamps of each waypoint. In this thesis, we will consider that the GPS data allows estimating link travel times by following the described procedure.

The map-matching process transforms waypoint sequences to paths in the network (PTV AG (2020)). First, waypoints are projected onto the network using a map-matching approach (Kubicka et al. (2018)), assigning them to an appropriate point on the nearest link in the network and, secondly, the trajectory

is reconstructed. Figure 2.5 shows an example of how this process works, where the red stars are the waypoints and the red numbers near the links are the relative position of the waypoint projection onto the target link. Timestamps for waypoints are depicted in green. The first and last links are not fully covered by time information and are thus dropped from the link sequence.



$$tt_3 = \frac{l_3}{0.2l_2 + l_3 + 0.7l_4} 20s \quad \text{(a)} \qquad\qquad tt_4 = \frac{l_3}{0.2l_2 + l_3 + 0.7l_4} 20s + \frac{0.3l_4}{0.3l_4 + 0.5l_5} 7s \quad \text{(b)}$$
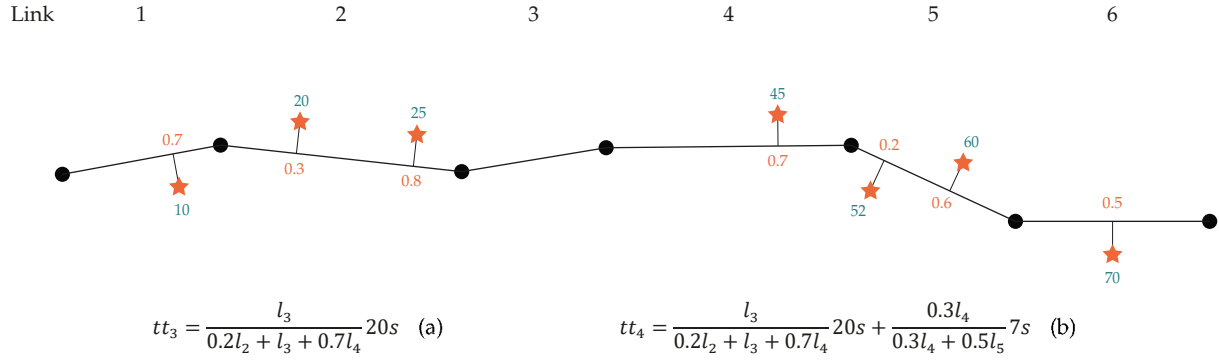
Figure 2.5: Example of the interpolation of travel times according to the waypoints sequence

This map-matching process estimates the link travel times from the sequences in the waypoint timestamps. For all links in the sequence, their interpolated travel time is the sum of the timestamp differences between two consecutive waypoints mapped on the target link. In the case of two consecutive waypoints that are not wholly projected within one link, the distance-based fraction within the link is taken ($l_k$ is the length of link $k$ in Figure 2.5).

For instance, the travel time for link $l_3$ can be estimated by taking into account that the travel time for a trip between the $3^{\text{rd}}$ and $4^{\text{th}}$ waypoints is 20 seconds, and this time is the estimated travel time of the whole link $l_3$ plus a 0.2 fraction of $l_2$ and a 0.7 fraction of $l_4$ (Equation (a) in Figure 2.5). The estimated travel time of link $l_4$ is obtained by adding two parts (Equation (b)). The first part is the travel time proportion between the $3^{\text{rd}}$ and $4^{\text{th}}$ timestamps in link $l_4$ (adding 0.7 of $l_4$ to 0.2 of the length of link $l_2$ plus the entire length of link $l_3$). The second part is estimated directly from the proportion of link $l_4$ lying between the $4^{\text{th}}$ and $5^{\text{th}}$ timestamps (a fraction of 7 seconds calculated as 0.3 of the $l_4$ distance within the total distance between the $4^{\text{th}}$ and $5^{\text{th}}$ waypoints: $0.3l_4 + 0.2l_5$).

The map-matching process transforms each trajectory from a waypoint sequence $\{WP_1, \ldots, WP_k\}$ (see Table 2.2) into a sequence of links in the network $\{L_1, \ldots, L_n\}$ with an extrapolated travel time for each link; that is, into a sequence $(tt_1, \ldots, tt_n)$, such that the sum of all these travel times is the difference between the first and last waypoint of the trajectory. In this process, it is useful to store the departure time of each path, which is indeed the timestamp of the first waypoint. Therefore, one can estimate the timestamp when crossing each link by using the estimated travel time. Table 2.4 shows an example of the map-matched floating car database.

Finally, once all the waypoint sequences are converted to several paths with full details at the link level, link travel times can be estimated using an average of all the extrapolated link travel times for the same time period. The outcome of this process is the set of estimated link travel times in each time period t: $\overline{tt}_{lt}, \forall l \in L, \forall t \in \mathcal{T}$ for all links in the network that are used by the GPS tracking. This is the data set of estimated link travel times.

Table 2.4: An example of the map-matched database

| Path ID | Link Sequence | Link ID | Starting Time | Travel Time [sec] | Cumulated Travel Time [sec] |
|---------|---------------|---------|---------------|-------------------|------------------------------|
| 4261353 | 1 | 1542 | 07:43:58 | 2.5 | 2.5 |
| 4261353 | 2 | 1574 | - | 3.7 | 6.2 |
| 4261353 | 3 | 1562 | - | 8.1 | 14.3 |
| .............. | ...... | ........ | ............... | ........ | ........ |
| 4261353 | 12 | 1598 | - | 6.5 | 65.8 |
| 4261353 | 13 | 1602 | - | 3.9 | 69.7 |
| .............. | ...... | ........ | ............... | ........ | ........ |
| 4261355 | 1 | 1841 | 07:45:02 | 10.4 | 10.4 |
| 4261355 | 2 | 1714 | - | 7.9 | 18.3 |
| 4261355 | 3 | 1542 | - | 4.7 | 23.0 |
| .............. | ...... | ........ | ............... | ........ | ........ |
| 4261355 | 24 | 1562 | - | 5.4 | 114.1 |
| .............. | ...... | ........ | ............... | ........ | ........ |

Despite the huge quantity of trajectories introduced into the network, the GPS sample may uncover some links, depending on the GPS data's penetration rate among the population. Moreover, the procedure that infers link travel times can produce non-feasible values when link travel times are below free-flow link travel times. In these situations, scaled travel time is used:

$$\overline{tt}_{l't} = R \cdot tt_{0l'} \; , \; \; R = mean_{l \in GPS}\left(\frac{\overline{tt}_{lt}}{tt_{0l'}}\right) \tag{2.2}$$

where $tt_{0l}$ is the free-flow travel time at each link and $R$ is computed using all observed link travel times and their corresponding free-flow travel times. $R$ is then the arithmetic mean of the expanding factors found for each link, which can be understood as a global expanding factor that implicitly accounts for the congestion effects. The methodological process for generating the observed link travel times data set is summarized in Figure 2.6.
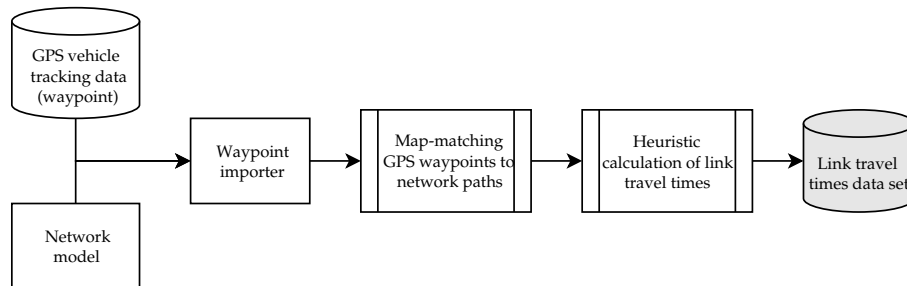


Figure 2.6: Conceptual methodological approach to importing waypoints into a mesoscopic model and using them to estimate link travel times

Algorithm 2.1 explains how the estimated link travel times are obtained.

---

**Algorithm 2.1:** From GPS data to estimated travel times

---

**Input:** A waypoint data set (WP), like the one shown in Table 2.2.
**Input:** The link travel times at free flow, $tt_{0l}$.

// Phase 1:  Importing GPS trips to links;
**foreach** *ID in unique(WP.ID)* **do**

> **Import:**  Waypoints into Visum;
> **Map-match:**  Trajectories to sequence of links;
> **Estimate:**  Travel Times at link level ;                                    // Figure 2.5
> **Calculate:**  Starting time at each link;
> **Obtain:**  $\overline{tt}_{lt}(WP)$;

// Phase 2:  Estimate travel times;
**foreach** *t in Time Periods* **do**

> **foreach** *l in Links* **do**
>
> > **Set:**  $\overline{tt}_{lt} \leftarrow \text{mean}(tt_{lt}(WP))$;

// Phase 3:  Missing values;
**Calculate:**  $R \leftarrow \text{mean}(\overline{tt}_{lt}/tt_{0l})$;
**foreach** *missing* $\overline{tt}_{lt}$ **do**

> **Set:**  $\overline{tt}_{lt} = R \cdot tt_{0l}$ ;                                    // Equation 2.2

**Output:** Return $\overline{\mathbf{TT}} = \left[\overline{tt}_{lt}\right]_{lt}$

---

### 2.5.1   Applying the methodology to a synthetic network (Part II)

Building on the synthetic network based on generated synthetic data in Section 2.4, our intention at this point is to convert the GPS data set (with 150k trips and 9 million waypoints) into reliable information on the traffic conditions. Since the data generation procedure is synthetic, we assume, first, that the GPS data set contains exclusively private vehicles that circulate with a fixed origin and destination and, second, that the GPS data is highly accurate.

As this thesis is a collaboration with the company PTV Group (see Section 1.4.1), we make use of the Visum tool GPX Import (PTV AG (2020); Section 29.14), which map-matches a GPS data base onto the network by means of the procedure described previously. After this internal Visum process, we obtain a large data set of trajectories at the link level with interpolated travel times across the network. These results provide the input for the heuristic calculation of the time-dependent link travel times, because they indicate the precise time when the vehicle used the link. Therefore, these extrapolated link travel times can be split into different time periods. At the end of the process and after imputing the missing value by means of Equation 2.2, we have obtained estimates of link travel times for each time period.

Figure 2.7 shows in black the links where the averaged travel times are available, which represent the 96% of the links of the network, which represents a sufficiently large coverage. The ones in red were estimated as show in Equation 2.2, either because they were not available or because they were under the free flow travel time, as explained. One can see that this last set of links belongs mostly to ending links, which could happen that do not have traffic flow in simulation.
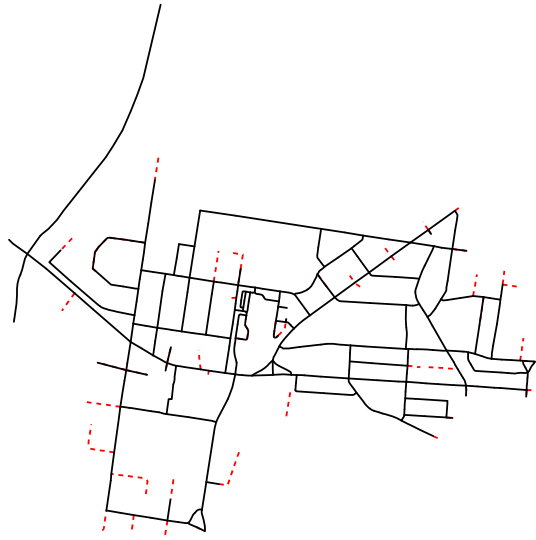
Figure 2.7: Links of the network covered by the GPS data set.

As stated before, a large amount of data should ensure reliable estimates by averaging those travel times at each link and time interval. In order to check their reliability, they must be compared with the path travel times obtained from the ground truth OD matrix. All the path travel times for each OD pair at each different time interval are collected, and the mean has been calculated by OD pair and time interval in order to obtain the OD travel times. The comparison is shown in Figure 2.8.
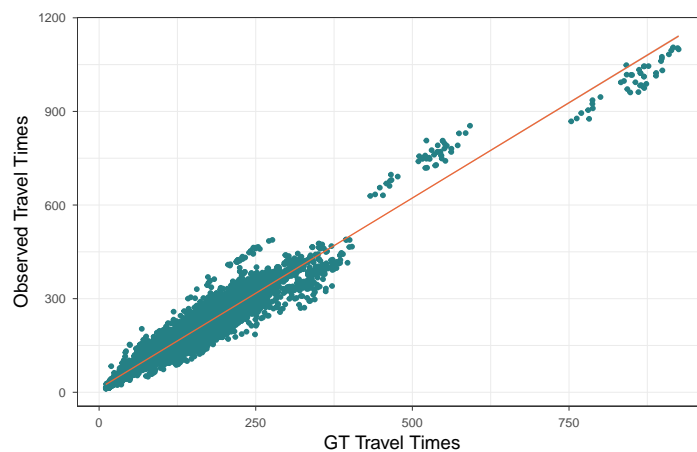


Figure 2.8: Comparison of observed OD travel times with ground truth OD travel times

The correspondence of both measurements is high, because the fit is $R^2 = 0.9445$ after removing some outliers that represent only 0.31% of total OD flow. These results ensure that when the GPS sample is appropriately filtered, cleansed, and free of bias in the OD pairs, the proposed methodology for estimating link travel times is reliable, since using them to construct different OD paths presents travel times that are similar to those under ground truth conditions.

# 3

# Quality Measures for the Estimated OD Matrix

*This chapter deals with the underdetermination of the OD matrix. After justifying that the conventional measures are not able to compare different OD matrices from the traffic demand perspective, we study a measure that permits to evaluate the structural similarity between OD matrices.*

## 3.1 The underdetermined DODME problem

The DODME bi-level optimization problem (Equation 1.6) is complex and can lead to solutions that are far from the real mobility pattern of the network. Frederix et al. (2013) carefully analyzed the congestion behavior of a dynamic traffic assignment and highlighted many factors that influence the dynamic OD matrix estimation. The main reasons for it being difficult are that the OD estimation problem's objective function is inherently non-convex and the assignment matrix has a nonlinear dependence on the OD flows. Congestion in the network produces indirect spatial and temporal effects on the link traffic counts, and these effects are usually not captured by the analytical methods that use the dynamic assignment matrix to approximate it. Moreover, adding a Taylor expansion (Equation 1.19) to capture the mentioned congestion effects are not computationally feasible, because it requires a large number of evaluations for the dynamic traffic assignment.

The non-convexity of the objective function makes it feasible to converge to a local minimum of the function and, therefore, as Tavana (2001) suggested, one requirement for tackling this is starting with a reliable reference OD matrix. Moreover, Frederix et al. (2013) highly recommend throughout their paper the use of a reference OD matrix that presents the correct traffic regime, despite the fact that it still does not ensure convergence to the global minimum of the function.

Moreover, the detection layout coverage has an effect on the underdetermination of the problem. Due to the fact that there are always fewer sensors (equations) than OD pairs (variables), the problem is always underdetermined. Moreover, due to the low number of detectors and their locations, the detection layout is usually not able to capture 100% of the OD flows circulating throughout the network; thus, their effects are not reflected in the objective function. The underdetermination and, of course, the lack of coverage imply that one can reach different estimated OD matrices that reproduce the target traffic counts on the covered links, depending on which seed OD matrix is used. Bierlaire (2002) studied this phenomenon in the static case, defining the *Total Demand Scale* (TDS). This measure shows how large underdetermination gap is for a specific assignment matrix and for specific link coverage. It measures the difference between the maximum and minimum OD matrix (in terms of number of trips) that can satisfy the assignment

equations and provide the correct estimation of traffic counts. When the detection layout cannot cover at all all the OD flows, then $TDS = \infty$, even in the cases when there is full link coverage, meaning one sensor at each link of the network. As Bierlaire (2002) points out, TDS could then be equal to 0, meaning that "the OD estimation algorithm has captured correctly the total level of demand in the network. The underdetermination relates only to the repartition of that demand across OD pair". From a practical point of view, a larger TDS requires a better a priori matrix. Although measuring the TDS allows one adjust the position and number of detectors and see its effect on the coverage, the full link coverage does not guarantee a unique solution for the OD estimation problem.

## 3.2   $R^2$ as measure of goodness of fit for DODME

As already mentioned, the OD matrices of an urban network indicate the trips that circulate throughout the network, but they are not directly observable. Therefore, there is no solution to which can be compared an estimated OD matrix obtained by the DODME problem. On the other hand, the objective function of the OD matrix estimation problem is commonly stated as a discrepancy function between traffic measurements and the corresponding simulated values using either a macroscopic or a mesoscopic model. Therefore, the objective function is not analytical because of its dependence on the dynamic traffic assignment and congestion effects, and it can thus be evaluated only by simulation.

So the question is: How do we measure the quality of the estimated OD matrix? Since research began on OD matrix estimation (Van Zuylen & Willumsen (1980), Spiess (1990), Florian & Chen (1995)), the most common approach is to check if the estimated OD matrix is capable of reproducing the same traffic counts on the available links. This is measured by fitting a linear regression between observed counts and estimated counts while also using $R^2$ as a goodness-of-fit indicator. Even though there are many other goodness of fit measures, as shown in Hollander & Liu (2008), that are used regarding the traffic counts in the dynamic OD matrix estimation problem (such as the root mean square error (RMSE), its normalized version (nRMSE) and, in practical environments, the GEH and the Theil's proportions), we consider the most common metric, the $R^2$ and new similarity measures that are presented below.

The linear regression effectively shows how the estimated OD matrix produces the simulated traffic counts, once it has been assigned to the simulation model. However, as already explained, there are many OD matrices that can reproduce similar traffic counts on certain links, but they can be totally different in terms of mobility patterns. Because the objective function is non-analytical, it is difficult to understand how the optimization problem works. Moreover, the minimization problem tries to fit these values and reaches the global minimum once the simulated traffic counts are identical to their corresponding real values. The optimization problem can be viewed as merely a meta-regression problem that adjusts the OD flows in order to minimize the discrepancies. However, the problem as defined in Equation 1.6 does not take into account the socio-economic demographics and land use of the study area. Furthermore, it can produce unrealistic OD flows in this regard, although it provides a good fit on the linear regression of the traffic counts.

As a matter of fact, despite $R^2$ being a good indicator of the optimization problem's performance, it can produce misleading results in which a high regression value is achieved but the resulting estimated OD matrix can be far from the reality of the demand pattern and the internal mobility of the study area. Therefore, some other indicators are needed to evaluate the mobility pattern of the OD matrices.

## 3.3   Comparing OD matrices

Some examples of a reference OD matrix are a historical OD matrix from a previous project that needs to be updated or a reference OD matrix obtained by surveys and socio-demographic studies. Using such a reference OD matrix is usually the best approach for obtaining the mobility pattern of the network, because it is not possible to obtain the ground truth OD matrix, which represents the real mobility of the network. If this reference OD matrix is available, one can naturally compare the estimated OD matrix to it and obtain another measure of goodness of fit for the procedure. Moreover, in the case of a synthetic OD matrix estimation problem where the ground truth OD matrix is available, the estimates can be directly compared to it and, therefore, it makes sense to consider good measures for comparing OD matrices.

Typical measures between OD matrices are those that are extrapolated from classical vector distances by considering both matrices $\mathbf{A} = \left[ A_{ij} \right]_{ij}, \mathbf{B} = \left[ B_{ij} \right]_{ij} \in \mathcal{M}_{m \times n}(\mathbb{R})$ as vectors of $\mathbb{R}^{m \times n}$. However, the measures inspired by the Euclidean distance ($\mathrm{RMSE}(\mathbf{A}, \mathbf{B})$), Manhattan distance ($\mathrm{RMAE}(\mathbf{A}, \mathbf{B})$), and other vector distances fail to capture the differences and similarities in many aspects, such as the structure of the OD matrix. One example inspired by Djukic (2014) is shown in Figure 3.1, where a reference OD matrix can produce two matrices ($\mathbf{A}$ and $\mathbf{B}$) with clearly different structures, although they are indistinguishable in terms of the RMSE, which is calculated as:

$$\mathrm{RMSE}(\mathbf{A}, \mathbf{B}) = \sqrt{\frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \left( A_{ij} - B_{ij} \right)^2} \tag{3.1}$$

Graphically, one can see that the OD matrix $\mathbf{A}$ is structurally similar to the reference OD matrix, despite the values being greater. On the other hand, the OD matrix $\mathbf{B}$ has similar values but is clearly structurally different.

Table 3.1 quantifies this phenomenon. Both OD matrices have the same RMSE with respect to the ground truth OD matrix, but another measure also exists for capturing the structural similarity ($\mathrm{MSSIM}$), and this will be described in Section 3.4.

Table 3.1: RMSE and $\mathrm{MSSIM}$ values for comparing both OD matrices to the reference OD matrix

|          | RMSE   | MSSIM  |
|----------|--------|--------|
| GT-$M_1$ | 14.125 | 0.7629 |
| GT-$M_2$ | 14.125 | 0.5543 |

From a traffic point of view, the structure of an OD matrix represents the network's mobility pattern. For example, a row of an OD matrix represents all the trips departing from the same origin zone to all the other destination zones. Analogously, a column of an OD matrix counts all the trips that arrive at a certain destination zone from all other zones in the network. Therefore, it is important to use a measure that compares not only the values of both matrices but also their structure while taking into account the meaning of row and column totals in the context of urban mobility.

For this purpose, Djukic et al. (2013), Djukic (2014), Behara et al. (2018) propose and use a similarity measure called the *Mean Structural Similarity Index* (MSSIM), which was created for image quality assessment by Wang et al. (2004) in order to compare two different images. This is the measure that is studied in this thesis, and we propose some modifications that provide transportation meaning.
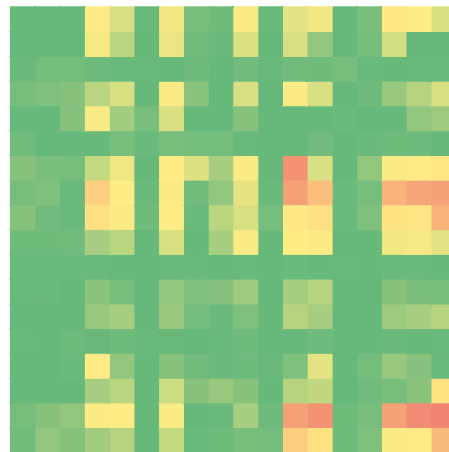
(a) Ground truth OD matrix
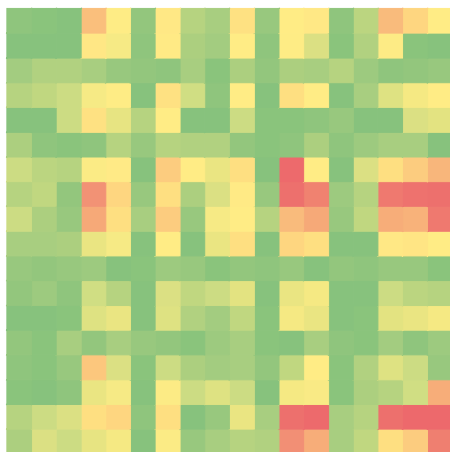


(b) OD matrix $M_1$
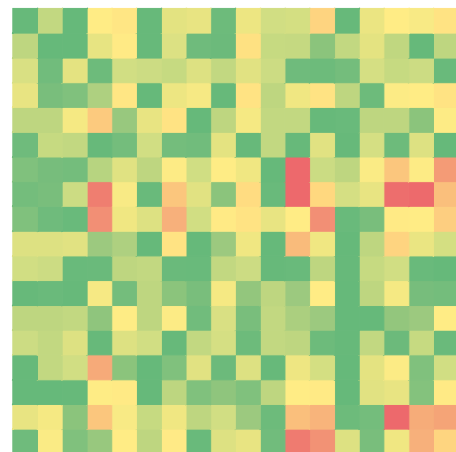


(c) OD matrix $M_2$

Figure 3.1: Heatmaps of a ground truth OD matrix and 2 different OD matrices

Other researchers also studied how to compare two OD matrices with a structure perspective. Ruiz de Villa et al. (2014) and Behara et al. (2020a) proposed the Wasserstein and Levenshtein. However, as Behara (2019) thesis studied exhaustively, these distances, designed to account for similarities and dissimilarities among string of characters, do not have a clear intermediation in terms of what an OD matrix physically means, while the analogy between pixels and OD cells is more clear, and its physical interpretation more suitable, so there were discarded.

Therefore it has finally the one selected by Djukic (2014) or Behara (2019), Behara et al. (2020b), providing explicit procedures based on that physical interpretation, and this has been the reason for proposing and using in this thesis a refined MSSIM and a windowing procedure explicitly based on the physical interpretation of the meaning of an OD.

## 3.4   Mean Structural Similarity Index (MSSIM)

In the context of an image quality assessment process for comparing two different images, Wang et al. (2004) present SSIM – the structural similarity index – for a matrix of pixels that is the product of three different comparison components: luminance, contrast, and structure. It is calculated as follows:

$$\text{SSIM}(\mathbf{a}, \mathbf{b}) = L(\mathbf{a}, \mathbf{b})^{\alpha} C(\mathbf{a}, \mathbf{b})^{\beta} S(\mathbf{a}, \mathbf{b})^{\gamma} \tag{3.2}$$

where

$$\begin{cases} L(\mathbf{a}, \mathbf{b}) = \dfrac{2\mu_a \mu_b + C_1}{\mu_a^2 + \mu_b^2 + C_1} \\[2mm] C(\mathbf{a}, \mathbf{b}) = \dfrac{2\sigma_a \sigma_b + C_2}{\sigma_a^2 + \sigma_b^2 + C_2} \\[2mm] S(\mathbf{a}, \mathbf{b}) = \dfrac{\sigma_{ab} + C_3}{\sigma_a \sigma_b + C_3} \end{cases} \tag{3.3}$$

Here, $\mu_a, \sigma_a, \mu_b, \sigma_b, \sigma_{ab}$ are the mean, standard deviation, and covariance of the vectors $\mathbf{a}$ and $\mathbf{b}$, while $C_1, C_2, C_3$, are small stability constants for avoiding numerical problems. $\alpha, \beta, \gamma$ are weighting coefficients typically set to 1, Wang et al. (2004). Luminance (L) corresponds to the intensity of illumination, which is the mean of the different pixels in a sub-matrix. Contrast (C) is the root of the squared average between pixels once the luminance is removed from the sub-matrix, making it the standard deviation. Structure (S) is the comparison between the two sub-matrices using the covariance. As follows, these three measures are first transformed in order to adjust them to the interval $[-1, 1]$, where 1 means a perfect match and 0 means they are totally different. Negative values on S mean a negative correlation between the vectors. They are then placed together in Equation 3.2.

The SSIM index has a symmetric and bounded construction that presents a unique maximum ($\text{SSIM}(\mathbf{a}, \mathbf{b}) = 1$), which means a perfect match between the two vectors. These three properties are very powerful in helping us understand the similarity of two matrices. However, SSIM is not a distance in a metric space, because $\text{SSIM}(\mathbf{a}, \mathbf{b}) \neq 0$ when $\mathbf{a} = \mathbf{b}$. In cases where $C_3 = C_2/2$ are set, SSIM can be equivalently rewritten as follows, specifically by equivalently defining $\text{SSIM}(\mathbf{a}, \mathbf{b})$ as:

$$\text{SSIM}(\mathbf{a}, \mathbf{b}) = S_1(\mathbf{a}, \mathbf{b}) \cdot S_2(\mathbf{a}, \mathbf{b}) \text{ where } \begin{cases} S_1(\mathbf{a}, \mathbf{b}) = \dfrac{2\mu_a\mu_b + C_1}{\mu_a^2 + \mu_b^2 + C_1} \\ S_2(\mathbf{a}, \mathbf{b}) = \dfrac{2\sigma_{ab} + C_2}{\sigma_a^2 + \sigma_b^2 + C_2} \end{cases} \tag{3.4}$$

Moreover, Brunet et al. (2012) prove that the quantities in Equation 3.5 are bounded metrics in the corresponding space, aside from SSIM not being a distance:

$$\sqrt{1 - S_i(\mathbf{a}, \mathbf{b})} \text{ and } D_2(\mathbf{a}, \mathbf{b}) = \sqrt{2 - S_1(\mathbf{a}, \mathbf{b}) - S_2(\mathbf{a}, \mathbf{b})} \tag{3.5}$$

For instance, $D_2$ is the associated distance to SSIM because it satisfies the three axioms:

- *Identity*: $D_2(\mathbf{a}, \mathbf{b}) = 0 \iff \mathbf{a} = \mathbf{b}$

- *Symmetry*: $D_2(\mathbf{a}, \mathbf{b}) = D_2(\mathbf{b}, \mathbf{a})$

- *Triangle Inequality*: $D_2(\mathbf{a}, \mathbf{b}) \leq D_2(\mathbf{a}, \mathbf{c}) + D_2(\mathbf{c}, \mathbf{b})$

In the original problem, Wang et al. (2004) obtain the MSSIM by averaging the SSIM using sliding windows, which are submatrices of size $N_s$. In extrapolating the sliding windows to OD matrices, Djukic (2014) proposes reordering the OD matrix by volume, both by rows and by columns in order to obtain the MSSIM using the same sliding submatrices, because the term S of SSIM is highly sensitive to the order of the OD pairs. Another open question is the dimension of the submatrices $N_s$, which should be fixed, and it does affect the final measure. Behara et al. (2018) also reorder the OD pairs, but in this case they propose clustering them in greater regional areas. In the same article, they show how the dimension of the submatrices affects the MSSIM measure, so this new proposal solves the problem of tuning $N_s$, which is fixed automatically by the dimension of these regional areas.

We propose here a more meaningful variant that is easy to apply in practice and considers the physical meaning of the OD matrices. This variant consists of calculating the averages of SSIM according to rows and columns rather than submatrices, that is, by using rectangular sliding rules that correspond to either rows or columns in the OD matrix. One row in an OD matrix represents the distribution of trips departing from a single origin zone while, analogously, one column is the distribution of trips arriving at a single destination zone. This therefore corresponds to a physical interpretation of patterns in the underlying transport system. Thus, SSIM will capture the similarity between these described distributions by considering the mean, the variance, and the structure of departure and arrival distributions, all of which correspond to the structural property of the trip patterns described by the OD. Moreover, this proposal also fixes $N_s$ to the number of origins and destinations in the network.

Then, if MSSIM is averaged over $N_s$ sliding windows, a key question arises in regard to whether all windows have the same weight or if their role in the total demand requires that they have different weights. In the case of OD Matrices, it is obvious that not all origins and destinations are equivalent in a transport network. Therefore, a weighted MSSIM (as in Wang & Simoncelli (2008)) prioritizes those origins and destinations with more impact on the network. This proposed weighting average is defined as follows:

$$MSSIM(\mathbf{A}, \mathbf{B}) = \frac{\sum_{i=1}^{N_s} W(\mathbf{a}_i, \mathbf{b}_i) SSIM(\mathbf{a}_i, \mathbf{b}_i)}{\sum_{i=1}^{N_s} W(\mathbf{a}_i, \mathbf{b}_i)} \tag{3.6}$$

where $\mathbf{a}_i, \mathbf{b}_i$ are, respectively, the $i$-th windows of $\mathbf{A}, \mathbf{B}$, while the weight $W(\mathbf{a}_i, \mathbf{b}_i)$ is given by:

$$W(\mathbf{a}_i, \mathbf{b}_i) = \log\left[\left(1 + \frac{\sigma_{a_i}^2}{C_2}\right)\left(1 + \frac{\sigma_{b_i}^2}{C_2}\right)\right] \tag{3.7}$$

In terms of the variance of the selected windows in OD matrices, these weighting factors correspond to the variance of the total generated trips from an origin (or total attracted trips to a destination) to all destinations (from all origins); thus, weighting is increased as the origin is distributed to all destinations (or the destination attracts from origins) becomes more non-uniform. Moreover, these weights also take into account the magnitude of their contribution (implicitly in the variance value), so the contribution of each origin or destination to the overall demand pattern is well balanced.

The same weights are used to calculate the weighted $D_2$ for the entire OD matrix, that is:

$$MD_2(\mathbf{A}, \mathbf{B}) = \frac{\sum_{i=1}^{N_s} W(\mathbf{a}_i, \mathbf{b}_i) D_2(\mathbf{a}_i, \mathbf{b}_i)}{\sum_{i=1}^{N_s} W(\mathbf{a}_i, \mathbf{b}_i)} \tag{3.8}$$

Brunet et al. (2012) proved that $MD_2$ is a distance measure, and as that it has the properties making it suitable to be included in the distance minimization term of the objective function, Equation 1.6. However, the way MSSIM and $MD_2$ are formulated leads to mathematical models that are analytically and numerically hard to deal with, unless simplifications based on very restrictive assumptions. An example it the proposal of Behara (2019), Behara et al. (2020b) that seems valid only when ICT data are collected by Bluetooth antennas under very specific layouts that cannot be replicated in GPS scenarios.

### 3.4.1   MSSIM computation with a synthetic network

Let us imagine that a fictitious network with 18 zones is available. In order to calculate the different approaches of MSSIM, let us imagine that the ground truth OD matrix, $\mathbf{X}^{GT}$, and an estimated OD matrix, $\mathbf{X}^*$, are also available. These OD matrices can be arranged by proximity using wider zonification by regions (from I to V), as in Behara et al. (2018). The regions we created for this purpose are summarized in Table 3.2. Regions IV and V generate more than 60% of the trips, while regions II and IV attract more than 80% of the trips.

Figure 3.2 shows the heatmaps of both OD matrices when different arrangements are applied. In the first row (Figures 3.2a and 3.2b), they are ordered by the zonification code for origins and destinations, which is the original OD matrix. In the second row (Figures 3.2e and 3.2f), the rows and columns are

in increasing order according to the total number of trips generated or attracted, which we have done in order to calculate the measurement as suggested by Djukic et al. (2013). Finally, in the third row (Figures 3.2e and 3.2f), we can see it is ordered by the graphical divisions so they are grouped by regions, as suggested by Behara et al. (2018).

Table 3.2: Fictitious ground truth network demand by regions and zones

| Region | Zone | Generated Trips | Attracted Trips | Region Generated Trips | Region Attracted Trips |
|---|---|---|---|---|---|
| I | 1 | 393 | 102 | | |
| | 2 | 235 | 103 | 830 | 324 |
| | 15 | 202 | 119 | | |
| II | 5 | 173 | 413 | | |
| | 12 | 177 | 572 | 807 | 2006 |
| | 13 | 196 | 532 | | |
| | 16 | 261 | 489 | | |
| III | 3 | 98 | 114 | | |
| | 6 | 90 | 41 | 303 | 252 |
| | 11 | 50 | 48 | | |
| | 14 | 65 | 49 | | |
| IV | 4 | 331 | 496 | | |
| | 7 | 501 | 474 | | |
| | 10 | 355 | 368 | 2202 | 2382 |
| | 17 | 588 | 495 | | |
| | 18 | 427 | 549 | | |
| V | 8 | 595 | 145 | 1137 | 315 |
| | 9 | 542 | 170 | | |
| **TOTAL** | | **5279** | **5279** | **5279** | **5279** |
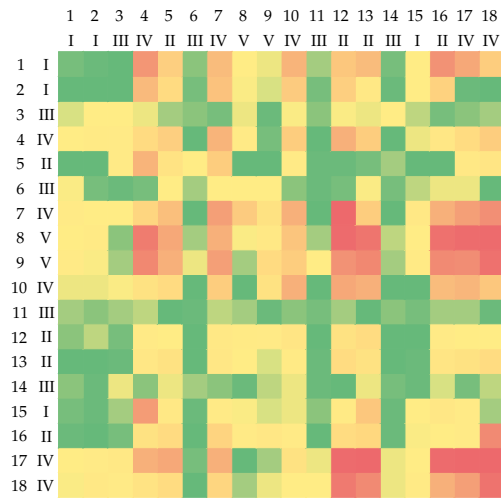
(a) Ground truth OD matrix
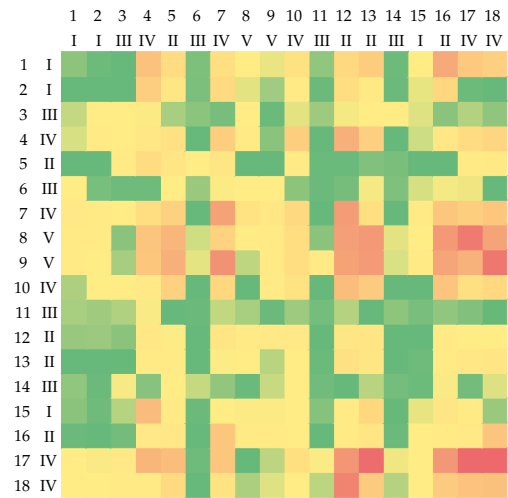
(b) Estimated OD matrix
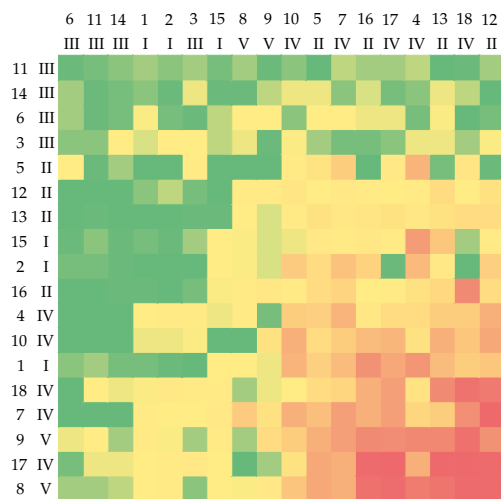
(c) Ground truth OD matrix by flows
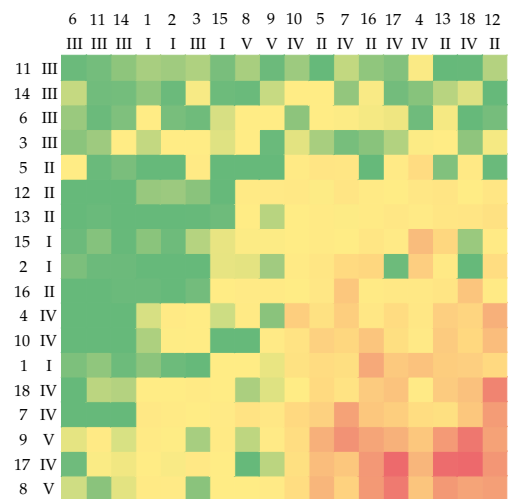
(d) Estimated OD matrix by flows
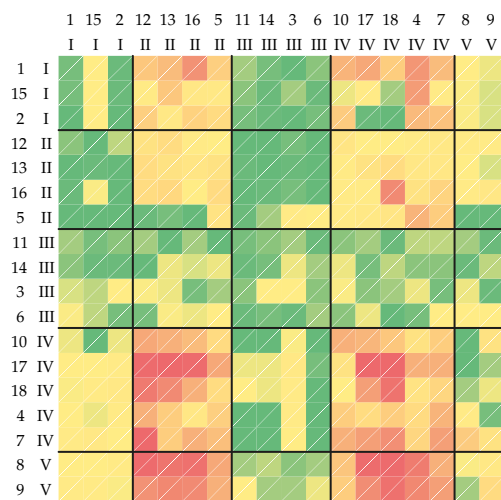
(e) Ground truth OD matrix by regions
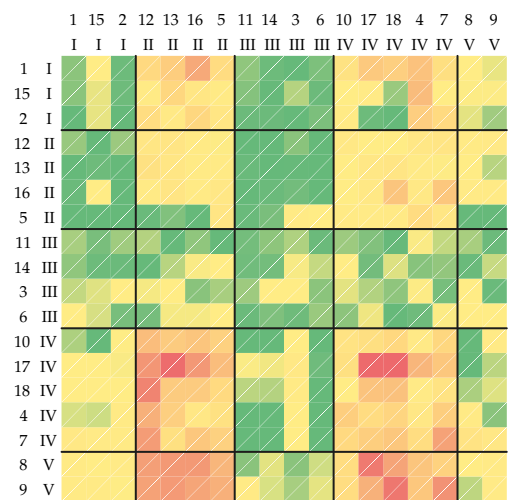
(f) Estimated OD matrix by regions

Figure 3.2: The ground truth and estimated OD matrices

The RMSE calculation is independent of the arrangement, as it does not depend on the position of each value in the matrix and its result is $\text{RMSE}(\mathbf{X}^{\text{GT}}, \mathbf{X}^*) = 12.072$. In any event, the MSSIM calculation is influenced by the OD flow arrangement, as can be seen in Table 3.3. The last row indicates the average of both calculations, by origins and by destinations, which leads to a global MSSIM indicator.

Table 3.3: Different MSSIM calculations

|  | $N_s$ | Not weighted | Weighted |
|---|---|---|---|
| | 3 | 0.64651 | 0.64654 |
| Original | 5 | 0.65161 | 0.65205 |
| OD matrix | 7 | 0.65264 | 0.65276 |
| | 9 | 0.65048 | 0.65055 |
| | 3 | 0.58501 | 0.58471 |
| Arranged by | 5 | 0.59273 | 0.59241 |
| flows | 7 | 0.59145 | 0.59414 |
| | 9 | 0.59458 | 0.59553 |
| Arranged by Regions | - | 0.62342 | 0.62773 |
| MSSIM by rows | - | 0.64582 | 0.64538 |
| MSSIM by columns | - | 0.63963 | 0.64026 |
| average MSSIM | - | 0.64273 | 0.64282 |

Although the values are quite similar and lead to the same conclusions, the arrangements result in different MSSIM values, depending on the dimensions of the submatrices, $N_s$.

Calculating MSSIM by rows and by columns does not depend on the $N_s$ nor on the regional zonification, which indeed provides more information about the network but is not always available. Moreover, it has a physical meaning, as already mentioned. Finally, depending on their magnitude and dispersion, adding weights prioritizes those that contribute more to the mobility pattern and gives more detail to the similarity measure. Table 3.4 shows $\mu_x, \mu_y, \sigma_x, \sigma_y$, and $\sigma_{xy}$ for each origin. Moreover, it shows the calculations of all the measures that comprise the MSSIM (luminance, contrast, and structure) and the weights.

As shown in this example, the weights are higher when the magnitudes $\mu_x$ and $\mu_y$ are higher, because they contribute more to the mobility pattern. However, when comparing the origins that generate the most trips, their weights take into account not only total trips generated, but also their standard deviation, which means their capacity of being more widely distributed throughout the network.

Table 3.4: Calculation of L, C, S, W, MSSIM, $MD_2$ for the specific example

| Origin | $\mu_x$ | $\mu_y$ | $\sigma_x$ | $\sigma_y$ | $\sigma_{xy}$ | L | C | S | W | MSSIM | $MD_2$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 21.83 | 10.77 | 18.86 | 9.66 | 175.84 | 0.79338 | 0.81151 | 0.96488 | 38.04169 | 0.62123 | 0.65085 |
| 2 | 13.06 | 6.01 | 14.29 | 7.07 | 95.92 | 0.75979 | 0.79529 | 0.94883 | 36.86310 | 0.57333 | 0.69686 |
| 3 | 5.44 | 2.91 | 2.79 | 1.55 | 4.09 | 0.83152 | 0.84785 | 0.94798 | 30.55599 | 0.66833 | 0.60393 |
| 4 | 18.39 | 10.26 | 13.76 | 8.57 | 114.00 | 0.85101 | 0.89729 | 0.96672 | 37.17102 | 0.73820 | 0.53062 |
| 5 | 9.61 | 4.50 | 12.03 | 5.25 | 61.47 | 0.76757 | 0.73264 | 0.97358 | 35.92177 | 0.54750 | 0.72052 |
| 6 | 5.00 | 2.63 | 3.48 | 1.97 | 6.57 | 0.82375 | 0.85721 | 0.95827 | 31.48140 | 0.67666 | 0.59566 |
| 7 | 27.83 | 13.71 | 20.54 | 9.89 | 194.30 | 0.79280 | 0.78155 | 0.95681 | 38.25809 | 0.59285 | 0.67779 |
| 8 | 33.06 | 16.62 | 27.69 | 13.63 | 366.63 | 0.80267 | 0.79235 | 0.97162 | 39.49733 | 0.61795 | 0.65381 |
| 9 | 30.11 | 17.36 | 22.52 | 14.02 | 303.14 | 0.86537 | 0.89746 | 0.95978 | 39.14151 | 0.74540 | 0.52275 |
| 10 | 19.72 | 9.82 | 16.99 | 8.86 | 140.20 | 0.79822 | 0.82032 | 0.93103 | 37.66015 | 0.60963 | 0.66184 |
| 11 | 2.78 | 1.44 | 1.63 | 0.99 | 1.48 | 0.81706 | 0.88942 | 0.91517 | 28.59560 | 0.66506 | 0.60743 |
| 12 | 9.83 | 5.51 | 7.85 | 4.57 | 34.74 | 0.85308 | 0.87008 | 0.96845 | 34.79120 | 0.71882 | 0.55163 |
| 13 | 10.89 | 4.89 | 9.74 | 4.74 | 43.43 | 0.74740 | 0.78664 | 0.94074 | 35.29563 | 0.55309 | 0.71595 |
| 14 | 3.61 | 1.95 | 2.45 | 1.43 | 3.32 | 0.83535 | 0.86946 | 0.94763 | 30.13729 | 0.68827 | 0.58372 |
| 15 | 11.22 | 5.90 | 12.25 | 6.47 | 78.39 | 0.82396 | 0.82541 | 0.98938 | 36.37582 | 0.67289 | 0.59949 |
| 16 | 14.50 | 7.13 | 14.03 | 6.88 | 88.65 | 0.79209 | 0.79108 | 0.91801 | 36.77142 | 0.57523 | 0.69404 |
| 17 | 32.67 | 19.12 | 29.46 | 18.69 | 536.49 | 0.87186 | 0.90479 | 0.97429 | 40.25321 | 0.76857 | 0.49660 |
| 18 | 23.72 | 11.37 | 20.74 | 10.51 | 205.27 | 0.77970 | 0.80648 | 0.94116 | 38.40094 | 0.59182 | 0.67917 |

# 4

# Simulation-based Optimization Approaches

*This chapter focuses on the simulation-based optimization approaches. We study the existing SPSA studies for OD estimation and propose enhancements that permit to add more information to the problem. We also present a heuristic methodology that allows the addition of estimated travel times from GPS data. All the proposed enhancements are presented with a complete set of experiments and results.*

The simulation-based optimization approaches to solving DODME are heuristic techniques that, as the name states, combine simulation engines and optimization algorithms in order to estimate the optimal solution without computing the objective function. Moreover, contrary to the analytical approaches that linearize the objective function, these approaches are able to capture the traffic phenomena changes and congestion effects that occur when performing dynamic traffic assignment with different OD matrices.

Using new ICT traffic measurements that complement the traffic counts used could help reduce under-determination in the DODME problem. Furthermore, it is no straightforward task to introduce them into the analytical version of the problem in Equation 1.8, since there is no clear relationship between the problem variables, the OD flows, and these new traffic measurements, such as travel times between arbitrary points in the network or certain link speeds. The simulation-based optimization techniques can overcome this problem by using non-analytical optimization methods with stochastic gradients that make them computationally fast. However, these usually obtain only an estimate of an optimal solution.

Of all the different approaches mentioned in Chapter 1, the simultaneous perturbation stochastic approximation (SPSA) is one of the most commonly used for solving the OD estimation problem, principally because of its versatility and easiness of implementation.

## 4.1   A specific approach based on SPSA

SPSA, originally proposed by Spall (1992), is a heuristic optimization method that estimates the direction of descent by calculating a stochastic gradient and evaluating the objective function only twice instead of N times, as in the case of a finite-difference gradient approach. This is appropriate for cases where the objective function cannot be analytically expressed as a function of the parameters and when the evaluation is costly, because it requires a simulation engine to produce the data involved in the evaluation. This is the case of the DODME problem, where a gradient approximation with only two dynamic traffic

assignments is desirable, because it is the most time-consuming step of the optimization, especially in the case of large networks.

The SPSA procedure, as in many iterative procedures, begins with an initial OD matrix (usually a historical OD matrix), and the next OD matrix is computed using the first order Taylor development:

$$\mathbf{X}^{(k+1)} = \mathbf{X}^{(k)} - a_k \hat{\mathbf{g}}(\mathbf{X}^{(k)}) \tag{4.1}$$

which is modified with the term $a_k \hat{\mathbf{g}}\left(\mathbf{X}^{(k)}\right)$, compounded by a gain sequence $a_k$ and the estimated gradient $\hat{\mathbf{g}}\left(\mathbf{X}^{(k)}\right)$. Moreover, with only the two evaluations, $Z\left(\mathbf{X}^{(k)} + c_k \mathbf{\Delta}^{(k)}\right)$ and $Z\left(\mathbf{X}^{(k)} - c_k \mathbf{\Delta}^{(k)}\right)$, the gradient estimate is calculated as follows:

$$\hat{\mathbf{g}}\left(\mathbf{X}^{(k)}\right) = \frac{Z\left(\mathbf{X}^{(k)} + c_k \mathbf{\Delta}^{(k)}\right) - Z\left(\mathbf{X}^{(k)} - c_k \mathbf{\Delta}^{(k)}\right)}{2c_k} \cdot \begin{pmatrix} \Delta_1^{-1} \\ \vdots \\ \Delta_N^{-1} \end{pmatrix}_{(k)} = \tag{4.2}$$

$$= \begin{pmatrix} \dfrac{Z\left(\mathbf{X}^{(k)} + c_k \mathbf{\Delta}^{(k)}\right) - Z\left(\mathbf{X}^{(k)} - c_k \mathbf{\Delta}^{(k)}\right)}{2c_k \Delta_1^{(k)}} \\ \vdots \\ \dfrac{Z\left(\mathbf{X}^{(k)} + c_k \mathbf{\Delta}^{(k)}\right) - Z\left(\mathbf{X}^{(k)} - c_k \mathbf{\Delta}^{(k)}\right)}{2c_k \Delta_N^{(k)}} \end{pmatrix}$$

where $\mathbf{\Delta}^{(k)}$ is a random N-dimensional vector (N corresponds to the number of problem variables) with $\Delta_i = \Delta_i^{(k)}, \forall i$ independent identically distributed random variables that satisfy $\mathbb{E}\left(\Delta_i\right) = 0$ and $\left|\mathbb{E}((\Delta_i^{-1})^n)\right| < \infty, \forall n$. The gain sequences $a_k$ and $c_k$ are named *step size* and *spacing coefficient*, and they are decreasing sequences of positive real numbers. Moreover, they must accomplish some regularity conditions (described in Spall (1992)) to ensure the almost certain convergence of Z to a local minimum.

Typically, the two sequences of step size $a_k$ and spacing coefficient $c_k$ are set as

$$a_k = \frac{a}{(A + k + 1)^\alpha}, \quad c_k = \frac{c}{(k + 1)^\gamma} \tag{4.3}$$

which satisfies the regularity conditions. $a, A, c$ are fixed and depend on the specific problem. On the other hand, Spall (1992) proved that $\alpha = 0.602$ and $\gamma = 0.101$ are optimal for the success of the procedure. Finally, the commonly used perturbation random variable is $\Delta_i \sim Be(1/2, \pm 1)$, which is a Bernoulli distribution with probability of $1/2$ for each outcome, $\pm 1$.

Algorithm 4.1 summarises the SPSA procedure.

---

**Algorithm 4.1:** Original SPSA

---

**Input:** A seed OD matrix, $\mathbf{X}^0$
**Input:** A stopping criterion, `thrsh_stop`
**Input:** A maximum number of iterations, `IterMax`

**Evaluate:** DTA $\left(\mathbf{X}^0\right)$ and obtain $\mathbf{Y}$;
**Set:** $\mathbf{X} \leftarrow \mathbf{X}^0$;
**Set:** $a, A, c$;
**Set:** $\alpha = 0.602$;
**Set:** $\gamma = 0.101$;

**foreach** $k = 1, \ldots, $`IterMax` **do**

    **Set:** $c_k \leftarrow c/(k+1)^\gamma$;
    **Set:** $a_k \leftarrow a/(A + k + 1)^\alpha$;

    **Generate:** $\Delta^{(k)} \sim Be(1/2, \pm 1, N)$;

    **Set:** $\mathbf{X}^+ = \mathbf{X} + c_k \Delta^{(k)}$ ; $\mathbf{X}^- = \mathbf{X} - c_k \Delta^{(k)}$;

    **Evaluate:** $Z^+ \leftarrow Z\left(\mathbf{X}^+\right)$ ; $Z^- \leftarrow Z\left(\mathbf{X}^-\right)$ ;

    **Calculate:** $\hat{\mathbf{g}}\left(\mathbf{X}\right) \leftarrow (Z^+ - Z^-)/\left(2c_k \Delta^{(k)}\right)$;

    **Set:** $\mathbf{X} \leftarrow \mathbf{X} - a_k \hat{\mathbf{g}}(\mathbf{X})$;

    **Evaluate:** DTA $(\mathbf{X})$ and obtain $\mathbf{Y}$;

    **if** `rel_diff` < `thrsh_stop` **then**
        **Stop**;

**Set:** $\mathbf{X}^* \leftarrow \mathbf{X}$ ;                          `// X* represents the estimated OD matrix`

**Output:** Return $\mathbf{X}^*$

---

### 4.1.1 Variants of SPSA

The previous description of SPSA is the original one given by Spall (1992). However, many improvements and alternatives have been developed and used for years. Below is a list of the most relevant variants of the DODME problem, especially for this thesis.

**Average of independent estimates**

Spall (1992) shows that averaging many independent estimates of the Equation 4.2 gradient contributes to a more stable and quicker convergence of the SPSA method. Therefore, the gradient estimate is finally calculated as:

$$\hat{\mathbf{g}}\left(\mathbf{X}^{(k)}\right) = \frac{1}{n_g} \sum_{j=1}^{n_g} \hat{\mathbf{g}}_j \left(\mathbf{X}^{(k)}\right) \tag{4.4}$$

where $\hat{\mathbf{g}}_j\left(\mathbf{X}^{(k)}\right)$ is calculated as in Equation 4.2. $n_g$ represents the number of independent perturbations used to calculate different estimated directions of descent before averaging them.

**Asymmetric Design**

As with numerically calculating the finite-differences gradient, it is possible to build the estimated asymmetric stochastic gradient. That means using $Z\left(\mathbf{X}^{(k)}\right)$ and $Z\left(\mathbf{X}^{(k)} + c_k\Delta^{(k)}\right)$ in Equation 4.2:

$$
\hat{\mathbf{g}}(\mathbf{X}) = \frac{Z\left(\mathbf{X}^{(k)} + c_k\Delta^{(k)}\right) - Z\left(\mathbf{X}^{(k)}\right)}{c_k} \cdot \begin{pmatrix} \Delta_1^{-1} \\ \vdots \\ \Delta_N^{-1} \end{pmatrix}_{(k)} = \begin{pmatrix} \dfrac{Z\left(\mathbf{X}^{(k)} + c_k\Delta^{(k)}\right) - Z\left(\mathbf{X}^{(k)}\right)}{c_k\Delta_1^{(k)}} \\ \vdots \\ \dfrac{Z\left(\mathbf{X}^{(k)} + c_k\Delta^{(k)}\right) - Z\left(\mathbf{X}^{(k)}\right)}{c_k\Delta_N^{(k)}} \end{pmatrix} \tag{4.5}
$$

In the case of combining the averaged estimated gradient, a large number of objective function evaluations are saved because all $\hat{\mathbf{g}}_j\left(\mathbf{X}^{(k)}\right)$ share the mid-point $\mathbf{X}^{(k)}$ evaluation and, therefore, the combination is computationally more efficient. The example in Figure 4.1 graphically shows the number of evaluations that must be made with multiple perturbations in the cases of a symmetric $(2 \cdot n_g)$ and asymmetric design $(n_g + 1)$. In the latter, all the gradients share the midpoint evaluation $Z(\mathbf{X})$.



$2n_g$

(a) Symmetric Design

$n_g + 1$

(b) Asymmetric Design
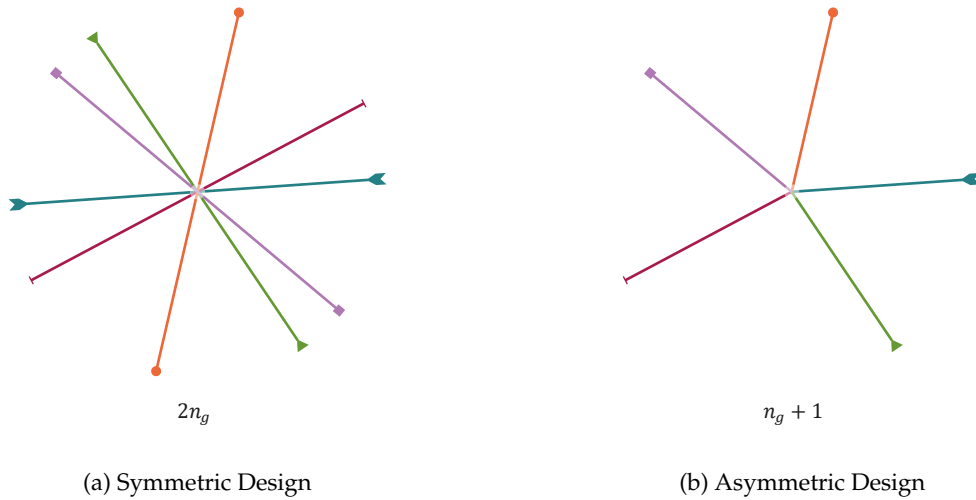
Figure 4.1: The symmetric and asymmetric design with averaging gradient and the corresponding number of evaluations

**Conjugate gradient (CG) descent version**

Some researchers in past works (Bullejos et al. (2014), Cantelmo, Cipriani, Gemma & Nigro (2014)) used the conjugate gradient (CG) (Luenberger & Ye (1973)) descent method for the optimization procedures in the OD estimation problem. This technique modifies the direction of descent in the iterative procedure by means of a linear combination of the current estimated gradient and the previous iteration direction.

It can be understood as an intermediate direction between the direction of maximum descent and the direction obtained from Newton's method. Moreover, it can easily be incorporated into SPSA by replacing Equation 4.1 with:

$$\mathbf{X}^{(k+1)} = \mathbf{X}^{(k)} + a_k \mathbf{d}^{(k)}$$
$$\mathbf{d}^{(k)} = -\hat{\mathbf{g}}^{(k)}\left(\mathbf{X}^{(k)}\right) + \beta_k \hat{\mathbf{g}}^{(k-1)}\left(\mathbf{X}^{(k-1)}\right)$$
$$\beta_k = \frac{\hat{\mathbf{g}}^{(k)}\left(\mathbf{X}^{(k)}\right) \cdot \mathbf{d}^{(k-1)}}{\left\|\mathbf{d}^{(k-1)}\right\|^2}$$

(4.6)

The main purpose of this is to avoid a slow convergence, which is typically associated with the gradient method's behavior. The implementation consists of incorporating more information but at an acceptable level of computation and storage. With this modification to the direction of descent, each succeeding step moves in a direction that is orthogonal to the previous iteration, as proven in Luenberger & Ye (1973).

**W-SPSA, C-SPSA and Hyb-SPSA**

Since SPSA is a method that estimates the gradient direction with few evaluations, many correlations between the problem variables are underestimated. In the case of the DODME problem, correlations between OD flows are evident, since one change in one OD flow can produce congestion that affects the others. The proposed W-SPSA in Antoniou et al. (2015) is a weighted SPSA that estimates these correlations and introduces them as weights into the estimated gradient. However, these weights must be calibrated, and they can introduce some bias into the OD estimation process.

Another variant that originates from applying SPSA to OD estimation tries to solve one of its other limitations. As it can be easily deduced from developing the SPSA, the perturbation induced at each iteration is the same for all system variables of the system, independently of their magnitudes. In the specific problem of OD estimation, a wide range of magnitudes exists for the OD flows, and the same perturbation for all them can produce strange effects or long convergences. Tympakianaki et al. (2015, 2018) propose a cluster-wise SPSA (c-SPSA) that groups the OD flows by magnitude (and by spatial correlation in the second paper), with the aim of intelligently perturbing the OD flows according to the clusters. However, these approaches need to calibrate more parameters, since there are different $c, a, A$ for each cluster; and the number of clusters must also be set. Moreover, the second clustering strategy is based on a complex analysis of the spatial correlations of the OD flows and their influence on sensors, which depends on the congestion-building process and the location of the sensors. Although it improves the results, such an analysis implies a substantial increase in the computational burden.

The Hybrid SPSA (Hyb-SPSA) variant was proposed for solving any potential source of instability induced by approximating the gradient for SPSA. Tympakianaki et al. (2018) propose calculating a hybrid form of the estimated gradient when some parts of the objective function are analytical and easily differentiable. For instance, the perturbation of the SPSA is used for computing only part of the gradient, and then the other part is provided by means of its analytical derivative. For example, if the objective function is $Z(\mathbf{X}) = w_1 \left\|\mathbf{Y}(\mathbf{X}) - \hat{\mathbf{Y}}\right\|_2^2 + w_2 \left\|\mathbf{X} - \mathbf{X}^H\right\|_2^2$, its hybrid gradient is calculated as:

$$\hat{\mathbf{g}}_{\text{hyb}}\left(\mathbf{X}^{(k)}\right) = w_1 \hat{\mathbf{g}}_{F_1}\left(\mathbf{X}^{(k)}\right) + 2w_2\left(\mathbf{X}^{(k)} - \mathbf{X}^H\right) = \tag{4.7}$$

$$= w_1 \frac{\left\|\mathbf{Y}\left(\mathbf{X}^{(k)} + c_k\boldsymbol{\Delta}^{(k)}\right) - \hat{\mathbf{Y}}\right\|_2^2 - \left\|\mathbf{Y}\left(\mathbf{X}^{(k)} - c_k\boldsymbol{\Delta}^{(k)}\right) - \hat{\mathbf{Y}}\right\|_2^2}{2c_k} \cdot \boldsymbol{\Delta}^{(k)-1} + 2w_2\left(\mathbf{X}^{(k)} - \mathbf{X}^H\right)$$

## 4.2   Advantages and limitations of SPSA

Since it is a method that computes the gradient by few perturbed simulation replications, SPSA is very versatile because it allows minimizing any objective function, even when it can not be analytically formulated and its evaluation must be made by other means, like, for instance, simulation. Moreover, another advantage is the simplicity of its implementation and, unlike the finite-differences gradient, the gradient is estimated by evaluating the objective function only twice, which is a nice property when the objective function is costly, which is the case for the OD estimation problem because a full dynamic traffic assignment is executed for each evaluation.

However, this method has pros and cons. Despite the advantages indicated above, due to the few number of evaluations and the way in which the gradient is estimated, it may omit some correlation effects between the variables and it can not guarantee a monotonous sequence of descent directions that can enlarge notably the computational times. Besides, the same property also allows modifying the objective function without modifying the algorithm. Furthermore, adding constraints requires a minor modification to the implementation, which is detailed in Sadegh (1997).

On the other hand, the sensitivity of the parameter settings is the main disadvantage of the SPSA, as these parameters have a large effect on the performance of the optimization procedure. One can easily deduce that a large step size $a_k$ implies undesirable large jumps at each iteration. On the other hand, however, a small step size naturally leads to slow convergence. The same happens with the spacing coefficient $c_k$, which must be adjusted well in order to have a suitable estimated gradient.

Another disadvantage of SPSA is that it uses an approximated and stochastic gradient. Therefore, it is recommended to provide an appropriate seed value in order to remain near the desired minimum. Finally, the same perturbation is applied to each variable at each iteration, which seems inconvenient in cases where variables have different magnitudes because they cannot be perturbed identically.

Table 4.1 summarizes the different advantages and limitations of SPSA.

Table 4.1: Advantages and disadvantages of SPSA

| Advantages | Disadvantages |
|---|---|
| Versatility | Parameters settings |
| Easy implementation | Approximated and stochastic |
| Few evaluations | Same perturbation at each variable |
| Additional information | |

### 4.2.1 Advantages and limitations of SPSA for DODME

The OD estimation problem has a complex objective function, so SPSA's strongest feature is its ability to deal with unknown objective functions. Moreover, because it allows adding new traffic measurements to the objective function as new summands, SPSA is suitable for OD estimation using diverse types of data that analytical models are unable to incorporate. Finally, the low number of evaluations needed for SPSA is a strong reason to use it in OD estimation, since evaluating the objective function requires simulating the full traffic assignment of the corresponding OD matrix.

On the other hand, OD matrices are typically sparse and, moreover, they usually present few cells with very high values, while many of them actually have low values. Therefore, SPSA performs badly by producing the same perturbation to all the different magnitudes of OD values. Furthermore, as already mentioned, SPSA's estimated gradient can fail on capturing the correlations between OD values and they are important as the effects that a specific OD value increasing produce to the other are notorious due to the availability of alternative routes.

It is known that the OD estimation problem is underdetermined, so many different OD matrices can produce the same traffic counts (that is, the same objective function value, thus leading to many local minima), even though they are different in terms of the network's socio-demographic information. Therefore, the heuristic nature of the algorithm requires a sound seed OD matrix to conduct to a proper local minimum. Finally, the setting of SPSA parameters is also crucial in the case of OD estimation, as it is always for the SPSA performance.

## 4.3 SPSA improvements

After some experimentation with different networks to explore the performance and sensitivity of SPSA, we have implemented some improvements and tested them for the OD estimation problem, always with the aim of overcoming the limitations presented above.

### 4.3.1 Normalization of variables

The OD flows in an OD matrix usually have different magnitudes, depending on the network's socio-demographic information. As already mentioned, the original version of SPSA applies the same perturbation to all the different OD flows, which can produce two effects:

- If the magnitude of the perturbation is linked to a greater magnitude of OD flows, very low OD flows are greatly perturbed, thus significantly changing them and the OD matrix structure.

- If the magnitude of the perturbation is linked to a lower magnitude of OD flows, very high OD flows will experience an insignificant perturbation, which implies that the method produces small changes to the most important flows of the system and longer convergences.

Tympakianaki et al. (2015, 2018) approached this phenomenon by clustering the variables according to their magnitudes. This thesis proposes the alternative of normalizing the OD values to the interval $[0, 1]$ before obtaining the perturbation. This is, indeed, a transformation of the perturbation according to the magnitude of each OD value. To do this, some lower and upper bounds must be set for each OD value. For example, one can use some additional information from the network, such as socioeconomic

data, to set a reasonable interval for each OD value or to perhaps set a relative percentage of acceptable change relative to a reliable reference OD matrix.  This normalization can be performed easily with a linear application for each OD flow:

$$\varphi_{ijr} : [a_{ijr}, b_{ijr}] \to [0, 1] \tag{4.8}$$
$$x_{ijr} \mapsto \tilde{x}_{ijr} = \frac{x_{ijr} - a_{ijr}}{b_{ijr} - a_{ijr}}$$

where $a_{ijr}, b_{ijr}$ are the lower and upper bounds corresponding to the OD value $x_{ijr}$. Once the transformation is set, the perturbation is applied to the normalized OD value and, therefore, the inverse of the linear transformation is needed:

$$\varphi_{ijr}^{-1} : [0, 1] \to [a_{ijr}, b_{ijr}] \tag{4.9}$$
$$\tilde{x}_{ijr} \mapsto x_{ijr} = a_{ijr} + \tilde{x}_{ijr}(b_{ijr} - a_{ijr})$$

Note that the entire transformation must be applied to the objective function and to the estimated gradient in order to be consistent.  That is, if the joint linear transformation is built as $\boldsymbol{\Phi} = (\varphi_{i_1 j_1 r_1}, \dots, \varphi_{i_t j_T r_T})$, the objective function must be composed with the inverse linear function $\tilde{Z}(\tilde{\boldsymbol{X}}) = Z(\boldsymbol{\Phi}^{-1}(\tilde{\boldsymbol{X}})) = Z(\boldsymbol{X})$, and the gradient should be calculated according to the set of normalized OD flows.

Therefore, when using the normalized variables in the SPSA procedure, each variable will be perturbed according to its magnitude.  It should be highlighted that the optimization problem remains free of constraints when adding the normalization, because the intervals are used only to equate the magnitude and thus, they are not constraints added to the problem. So, the equivalent OD estimation problem is as follows:

$$\begin{aligned} \min \quad & \tilde{Z}\left(\tilde{\boldsymbol{X}}\right) = w_1 F_1\left(\boldsymbol{Y}, \hat{\boldsymbol{Y}}\right) + w_2 F_2\left(\boldsymbol{\Phi}^{-1}(\tilde{\boldsymbol{X}}), \boldsymbol{X}^H\right) \\ \text{s. to:} \quad & \boldsymbol{Y} = \mathrm{Assignment}(\boldsymbol{\Phi}^{-1}(\tilde{\boldsymbol{X}})) \\ & \boldsymbol{\Phi}^{-1}(\tilde{\boldsymbol{X}}) \geq \boldsymbol{0} \end{aligned} \tag{4.10}$$

### 4.3.2  Selection of SPSA gain sequences

Spall (1992, 1998) and his experiments with SPSA in optimization problems show that the selection of SPSA gain sequences, $a_k, c_k$, is crucial for the algorithm's convergence and performance. The sequences in the form of Equation 4.3 are widely used, as they satisfy the conditions of convergence that were proved in Spall (1992). This reduces the problem to a matter of selecting appropriate values for $a, A, \alpha, c$ and $\gamma$. Moreover, Kostic et al. (2017) show the sensitivity of SPSA with respect to these parameters, but no research has been conducted on finding a criterion to select them in an automatized procedure. Based on the guidelines in Spall (2003), we propose an automated selection of the parameters $a, A$ and $c$. The choice of these parameters is based on the objective function's variability, which itself results from the simulation and the desired perturbation steps in the early iterations. The schema is detailed below:

1) Fix $\alpha = 0.602, \gamma = 0.101$, as Spall (1998) determines, where it is stated that they are optimal values for the convergence.

2) Compute several evaluations of $Z(\mathbf{X}^H)$ in order to capture the variability of the objective function. Since the variables have been normalized, as mentioned above, it seems natural to use the coefficient of variation $(CoV(Z) = \sigma_Z / \mu_Z)$. The parameter $c$ is set at $c = CoV$.

3) Set $A$ as 10% of the maximum number of iterations $(A = 0.1 \cdot iter_{max})$.

4) Simulate $n_g$ experiments using the SPSA logic $\mathbf{X}_i = \mathbf{X}^H + c\Delta_N$ and find the respective gradients, $\hat{\mathbf{g}}^{(k)}$, as in the SPSA procedure.

5) Determine the desired iterative modification of the first iteration:

$$\mathbf{X}^{(k+1)} = \mathbf{X}^{(k)} - a_k \hat{\mathbf{g}}^{(k)} \rightarrow \mathbf{X}^{(k+1)} - \mathbf{X}^{(k)} = \left| a_k \hat{\mathbf{g}}^{(k)} \right| \tag{4.11}$$

For example, the desired iterative perturbation could be set to $\left| a_k \hat{\mathbf{g}}^{(k)} \right| = 0.2$, which in the case of normalized variables would mean allowing a 20% change.

6) Therefore, one can compute the corresponding parameter $a$ for the desired change in the initial iteration, when $k = 1$:

$$\left| a_k \hat{\mathbf{g}}^{(k)} \right| = \frac{a}{(1 + A + k)^\alpha} |\hat{\mathbf{g}}^{(k)}| \rightarrow a = \frac{\left| a_k \hat{\mathbf{g}}^{(k)} \right| (1 + A + k)^\alpha}{|\hat{\mathbf{g}}^{(k)}|} \tag{4.12}$$

7) Since $n_g$ experiments have been performed, one has to choose the minimum, which is calculated as above, that is:

$$a = \min \left\{ a_{\{i=1\}}, \dots, a_{\{i=N_g\}} \right\} \tag{4.13}$$

Using this methodology for selecting the SPSA parameters reduces the problem to setting a desired perturbation modification for the first iteration, which is easier to contemplate, especially when the variables are normalized. Algorithm 4.2 summarises the proposed methodology.

### 4.3.3   SPSA variants: Reducing the feasible set

The versatility of simulation optimization techniques – especially when using SPSA – allows us to include additional information in a newer form, such as the constraints in the OD estimation problem.

As mentioned, the underdetermination of the DODME problem from Equation 1.6 can lead to different adjusted OD matrices that show the same traffic counts as the sensors, even though they are different. Furthermore, the adjusted OD matrix can also be inconsistent with the socioeconomic factors of the area under study. In traffic studies, practitioners usually have access to historical data in the form of an OD matrix, $\mathbf{X}^H$, which provides prior information about the mobility patterns of the study area. Therefore, by including constraints that account for this information in the SPSA formulation, more realistic results can be provided. Cipriani et al. (2011) were pioneers in introducing a total generation constraint into the minimization problem when solving it with SPSA:

$$\sum_{r=1}^{t_T} G_i^r \leq G_i^* \ , \ \forall i \in I \tag{4.14}$$

---

**Algorithm 4.2:** Selection of gain sequences for SPSA

---

**Input:** A seed OD matrix, $\mathbf{X}^0$
**Input:** Number of independent evaluations for step 2, `N_eval`
**Input:** A maximum number of iterations, `IterMax`
**Input:** Number of independent gradients, `ng`
**Input:** Admissible initial change, `rel_change`

**Set:** $\mathbf{X} \leftarrow \mathbf{X}^0$;

// STEP 1;
**Set:** $\alpha = 0.602$;
**Set:** $\gamma = 0.101$;

// STEP 2;
**foreach** $i = 1, \ldots, N\_eval$ **do**

> **Change:** Simulation seed;
> **Evaluate:** DTA $\left(\mathbf{X}^0\right)$ and obtain $\mathbf{Y}$;
> **Set:** $Z_i \leftarrow Z(\mathbf{X})$;

**Evaluate:** $\mu_Z \leftarrow \text{mean}(Z_i)$ ; $\sigma_Z = \text{std}(Z_i)$;
**Set:** $c \leftarrow \sigma_Z/\mu_Z$;

// STEP 3;
**Set:** $A \leftarrow \text{rel\_change} \cdot \text{IterMax}$;

// STEP 4;
**foreach** $k = 1, \ldots, ng$ **do**

> **Generate:** $\Delta^{(k)} \sim \text{Be}(1/2, \pm 1, N)$;
>
> **Set:** $\mathbf{X}_i^+ = \mathbf{X}_i + c\Delta^{(k)}$;
>
> **Evaluate:** $Z_i^+ \leftarrow Z\left(\mathbf{X}_i^+\right)$ ; $Z_i \leftarrow Z\left(\mathbf{X}_i\right)$;
>
> **Calculate:** $\hat{\mathbf{g}}_i(\mathbf{X}) \leftarrow \left(Z_i^+ - Z_i\right)/(c\Delta^{(k)})$;                    // Asymmetric Design, Equation 4.5
>
> // STEPS 5 AND 6;
> **Calculate:** $a_i \leftarrow \text{rel\_change} \cdot (1 + A + 1)^\alpha/\text{norm}\left(\hat{\mathbf{g}}_i(\mathbf{X})\right)$;         // Equation 4.12

// STEP 7;
**Set:** $a \leftarrow \min\{a_i\}$;                                                      // Equation 4.13

**Output:** Return $a, A, c$

---

with $G_i^*$ being the a priori generation value for the origin zone $i$; and $T$ is the number of time periods. Indeed, many different constraints can be added to the OD estimation problem in order to give more physical meaning to the feasible set. However, the process of building constraints can be tedious, depending on the granularity and the information being introduced, such as when adding information on the demographics of each zone.

Assuming that the historical OD matrix $\mathbf{X}^H$ is reliable, we propose a global constraint approach that assumes an admissible percentage of change relative to this reference OD value. More formally, let $x_{ijr}^H$ be the historical OD value for a certain origin $i$, destination $j$, and departure time $r$. Therefore, the estimated OD value is expected to be nearby, such as $x_{ijr} \in \left[ x_{ijr}^H(1 - \beta), x_{ijr}^H(1 + \beta) \right]$, where $\beta$ represents the admissible relative discrepancy between the historical and estimated OD flows.

The proposed relative change $\beta$ can be applied to all the OD values of the minimization problem, leading to the following feasible rectangle in the variable space. We consider a value of $\beta \in (0, 1)$ in order to obtain a solution in an acceptable neighborhood around the historical OD matrix:

$$G = \left\{ \mathbf{X} \mid x_{ijr}^H(1 - \beta) \leq x_{ijr} \leq x_{ijr}^H(1 + \beta) , \ x_{ijr} \in \mathbf{X} , \ \beta \in (0, 1) \right\} \subset \mathbb{R}^{|I| \times |J| \times |T|} \tag{4.15}$$

Note that the single constraint of Equation 4.14 is the constraint that results from summing all the upper bounds defined in $G$ for each origin. The summation of all the constraints makes the feasible region larger and thus allows greater values among some variables, which is compensated by others having low values. The proposal of Equation 4.15 for constraining the OD estimation problem defines a smaller feasible region that accounts for further information for each OD pair.

**Constrained SPSA**

The feasible set can be added as constraints to the minimization problem of OD estimation, as seen in Equation 1.6. This is done as follows:

$$
\begin{aligned}
\min \quad & Z\left(\mathbf{X}\right) = w_1 F_1\left(\mathbf{Y}, \hat{\mathbf{Y}}\right) + w_2 F_2\left(\mathbf{X}, \mathbf{X}^H\right) \\
\text{s. to:} \quad & \mathbf{Y} = \text{Assignment}(\mathbf{X}) \\
& (1 - \beta)\mathbf{X}^H \leq \mathbf{X} \leq (1 + \beta)\mathbf{X}^H \\
& \mathbf{X} \geq \mathbf{0}
\end{aligned}
\tag{4.16}
$$

The natural way that SPSA interacts with any feasible set is to orthogonally project the next calculated vector of variables onto the border of $G$, as proposed in Sadegh (1997). This projection can be difficult to implement in cases of strange constraints. However, in the case of separable constraints for each variable, as in the case of Equation 4.16, it consists of projecting each component onto the bounding values $(1 - \beta)x_{ijr}^H$ and $(1 + \beta)x_{ijr}^H$ when they go under or over the value limits during the iterative procedure, Equation 4.1:

$$
x_{ijr}^{(k+1)} = \begin{cases} (1-\beta)x_{ijr}^{H} & \text{if} \quad x_{ijr}^{(k)} - a_k \hat{g}\left(\mathbf{X}_{ijr}^{(k)}\right) < (1-\beta)x_{ijr}^{H} \\ x_{ijr}^{(k)} - a_k \hat{g}\left(\mathbf{X}_{ijr}^{(k)}\right) & \text{if} \quad (1-\beta)x_{ijr}^{H} \leq x_{ijr}^{(k)} - a_k \hat{g}\left(\mathbf{X}_{ijr}^{(k)}\right) \leq (1+\beta)x_{ijr}^{H} \\ (1+\beta)x_{ijr}^{H} & \text{if} \quad (1+\beta)x_{ijr}^{H} < x_{ijr}^{(k)} - a_k \hat{g}\left(\mathbf{X}_{ijr}^{(k)}\right) \end{cases} \tag{4.17}
$$

Algortihm 4.3 shows the addition of Equation 4.17 to the original SPSA.

---

**Algorithm 4.3:** Constrained SPSA

---

**Input:** A seed OD matrix, $\mathbf{X}^0$
**Input:** A stopping criterion, `thrsh_stop`
**Input:** A maximum number of iterations, `IterMax`
**Input:** Admissible relative change for $\mathbf{X}$, $\beta$

**Evaluate:** DTA $\left(\mathbf{X}^0\right)$ and obtain $\mathbf{Y}$;

**Set:** $\mathbf{X} \leftarrow \mathbf{X}^0$;
**Set:** $a, A, c$;
**Set:** $\alpha = 0.602$;
**Set:** $\gamma = 0.101$;
**Set:** $\mathbf{X}_{low} = (1-\beta)\mathbf{X}^0$ ; $\mathbf{X}_{up} = (1+\beta)\mathbf{X}^0$;

**foreach** k = 1,...,*IterMax* **do**

    **Set:** $c_k \leftarrow c/(k+1)^\gamma$;
    **Set:** $a_k \leftarrow a/(A+k+1)^\alpha$;

    **Generate:** $\Delta^{(k)} \sim Be(1/2, \pm 1, N)$;

    **Set:** $\mathbf{X}^+ = \mathbf{X} + c_k\Delta^{(k)}$ ; $\mathbf{X}^- = \mathbf{X} - c_k\Delta^{(k)}$;

    **Evaluate:** $Z^+ \leftarrow Z\left(\mathbf{X}^+\right)$ ; $Z^- \leftarrow Z\left(\mathbf{X}^-\right)$;

    **Calculate:** $\hat{\mathbf{g}}(\mathbf{X}) \leftarrow (Z^+ - Z^-)/\left(2c_k\Delta^{(k)}\right)$;

    **Set:** $\mathbf{X} \leftarrow \mathbf{X} - a_k\hat{\mathbf{g}}(\mathbf{X})$;
    **Set:** $\mathbf{X} \leftarrow \min\{\max\{\mathbf{X}, \mathbf{X}_{low}\}, \mathbf{X}_{up}\}$;             // Equation 4.17

    **Evaluate:** DTA $(\mathbf{X})$ and obtain $\mathbf{Y}$;

    **if** `rel_diff` < `thrsh_stop` **then**
        **Stop**;

**Set:** $\mathbf{X}^* \leftarrow \mathbf{X}$;             // $\mathbf{X}^*$ represents the estimated OD matrix

**Output:** Return $\mathbf{X}^*$

---

**Penalized SPSA**

Alternatively, Wang & Spall (1999) propose another way to add the constraints defined by the trust region G, which is based on adding penalty functions to the objective function, that is:

$$\min \quad Z\left(\mathbf{X}\right) = w_1 F_1\left(\mathbf{Y}, \hat{\mathbf{Y}}\right) + w_2 F_2\left(\mathbf{X}, \mathbf{X}^H\right) + r_k P(\mathbf{X}, \mathbf{X}^H)$$

$$\text{s. to:} \quad \mathbf{Y} = \text{Assignment}(\mathbf{X}) \tag{4.18}$$

$$\mathbf{X} \geq \mathbf{0}$$

where $r_k$ is an increasing sequence of the form $r_k = (1+k)^\rho$, and $P(\mathbf{X}, \mathbf{X}^H)$ is a set of penalization functions for the set of constraints that delimit the constraints of set G. Formally:

$$G \triangleq \left\{ \mathbf{q}_{ijr}(\mathbf{X}, \mathbf{X}^H) \leq 0, \; q^U_{ijr}(\mathbf{X}, \mathbf{X}^H) \leq 0, \forall i, j, r \right\} = \tag{4.19}$$
$$= \left\{ x_{ijr} - (1+\beta)x^H_{ijr} \leq 0, \; (1-\beta)x^H_{ijr} - x_{ijr} \leq 0, \; \forall i, j, r \right\}$$

The penalty function $P(\mathbf{X}, \mathbf{X}^H)$ has to be differentiable, non-negative, and an increasing function, such that propels the iterative procedure to values where $P(\mathbf{X}, \mathbf{X}^H) = 0$. Wang & Spall (1999) propose a sum positive functions, $p(x)$, that penalize when each constraint is violated. These positive functions satisfy $p(x) = 0$ if and only if $x \geq 0$, such as:

$$P(\mathbf{X}, \mathbf{X}^H) = \sum_{i \in I} \sum_{j \in J} \sum_{r=1}^{T} w_{ijr} p(q_{ijr}(\mathbf{X}, \mathbf{X}^H)) = \sum_{i \in I} \sum_{j \in J} \sum_{r=1}^{T} w_{ijr} \max\left\{0, q_{ijr}(\mathbf{X}, \mathbf{X}^H)\right\}^2 \tag{4.20}$$

As in the previous variant, the iterative procedure is also modified in order to incorporate the gradient of the penalization function. In this case, it affects the objective function and, consequently, the gradient. Since the penalty function is analytical, one can use its gradient to update the iterative process of Equation 4.1 to:

$$\mathbf{X}^{(k+1)} = \mathbf{X}^{(k)} - a_k \hat{\mathbf{g}}\left(\mathbf{X}^{(k)}\right) - a_k r_k \nabla P\left(\mathbf{X}^{(k)}, \mathbf{X}^H\right) \tag{4.21}$$

where the gradient can be calculated:

$$\nabla P(\mathbf{X}, \mathbf{X}^H) = \nabla\left(\sum_{i \in I} \sum_{j \in J} \sum_{r=1}^{T} w_{ijr} \max\left\{0, q_{ijr}(\mathbf{X}, \mathbf{X}^H)\right\}^2\right) = \tag{4.22}$$
$$= \sum_{i \in I} \sum_{j \in J} \sum_{r=1}^{T} w_{ijr} \nabla\left(\max\left\{0, q_{ijr}(\mathbf{X}, \mathbf{X}^H)\right\}^2\right) =$$
$$= \sum_{i \in I} \sum_{j \in J} \sum_{r=1}^{T} 2 w_{ijr} \max\left\{0, q_{ijr}(\mathbf{X}, \mathbf{X}^H)\right\} \cdot \nabla q_{ijr}(\mathbf{X}, \mathbf{X}^H)$$

Algortihm 4.4 shows how the Penalized SPSA is proposed for DODME.

---

**Algorithm 4.4:** Penalized SPSA

---

**Input:** A seed OD matrix, $\mathbf{X}^0$
**Input:** A stopping criterion, `thrsh_stop`
**Input:** A maximum number of iterations, `IterMax`
**Input:** Admissible relative change for $\mathbf{X}$, $\beta$

**Evaluate:**  DTA $\left(\mathbf{X}^0\right)$ and obtain $\mathbf{Y}$;

**Set:**  $\mathbf{X} \leftarrow \mathbf{X}^0$;
**Set:**  $a, A, c$;
**Set:**  $\alpha = 0.602$;
**Set:**  $\gamma = 0.101$;
**Set:**  $\mathbf{X}_{low} = (1 - \beta)\mathbf{X}^0$  ;  $\mathbf{X}_{up} = (1 + \beta)\mathbf{X}^0$;

**foreach** $k = 1, \ldots, $ `IterMax` **do**

    **Set:**  $c_k \leftarrow c/(k+1)^\gamma$;
    **Set:**  $a_k \leftarrow a/(A + k + 1)^\alpha$;

    **Generate:**  $\mathbf{\Delta}^{(k)} \sim Be(1/2, \pm 1, N)$;

    **Set:**  $\mathbf{X}^+ = \mathbf{X} + c_k\mathbf{\Delta}^{(k)}$  ;  $\mathbf{X}^- = \mathbf{X} - c_k\mathbf{\Delta}^{(k)}$;

    **Evaluate:**  $Z^+ \leftarrow Z\left(\mathbf{X}^+\right) + P(\mathbf{X}^+, \mathbf{X}^0)$  ;  $Z^- \leftarrow Z\left(\mathbf{X}^-\right) + P(\mathbf{X}^-, \mathbf{X}^0)$;    // Equation 4.20

    **Calculate:**  $\hat{\mathbf{g}}(\mathbf{X}) \leftarrow (Z^+ - Z^-)/\left(2c_k\mathbf{\Delta}^{(k)}\right)$;

    **Set:**  $\mathbf{X} \leftarrow \mathbf{X} - a_k\hat{\mathbf{g}}(\mathbf{X}) - a_k r_k \nabla P(\mathbf{X}, \mathbf{X}^0)$;    // Equation 4.21

    **Evaluate:**  DTA $(\mathbf{X})$ and obtain $\mathbf{Y}$;

    **if** `rel_diff` < `thrsh_stop` **then**
        **Stop**;

**Set:**  $\mathbf{X}^* \leftarrow \mathbf{X}$;    // $\mathbf{X}^*$ represents the estimated OD matrix
**Output:** Return $\mathbf{X}^*$

---

# 4.4   Case study: Results of SPSA without travel times

All SPSA proposed variants were tested on the downtown area of the city of Hillsboro (Oregon, United States). The network available in PTV Visum is calibrated for using the simulation-based assignment (SBA), which is a dynamic traffic assignment based on Mahut (2000).

The middle-size network consists of 618 links and 58 zones, with the simulation running over a time horizon from 08:00 AM to 09:00 AM in 3 periods of 20 minutes. This replicates the traffic conditions during rush hour on weekday mornings. The network is represented in Figure 4.2.
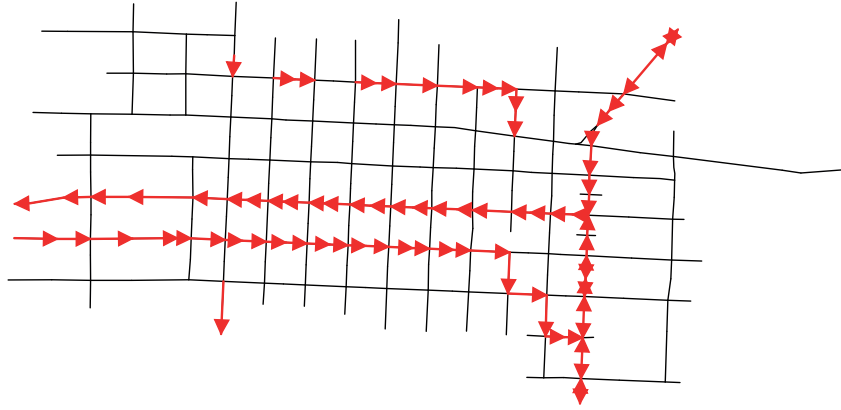
Figure 4.2: The network of the city of Hillsboro. Red arrows indicate the installed counting sensors

Following Antoniou et al. (2016), we prepared a set of synthetic experiments to test the different versions and variants proposed for SPSA before adding new terms to the objective function. In this sense, a detection layout of 80 sensors was set and traffic counts were generated for each time period. The historical OD matrix is a design factor for the set of experiments, because it is well known that the SPSA procedure is highly sensitive to the given seed.

Below, the factors for full factor experiments are listed:

- **Historical OD matrix**: This is also used as a seed for the SPSA procedure. Six different initializations were used to contemplate different situations, from similarly structured matrices with different numbers of trips to non-similar-structure matrices:

  - Incremental+: Incrementing all the OD values of the ground truth matrix by a fixed percentage: $\mathbf{Inc}^{+} = \mathbf{X}^{\mathsf{GT}}(1 + \delta)$ , $\delta = 0.25$.
  - Incremental-: Decrementing all the OD values of the ground truth matrix by a fixed percentage: $\mathbf{Inc}^{-} = \mathbf{X}^{\mathsf{GT}}(1 - \delta)$ , $\delta = 0.25$.
  - Chaos: Equidistributing all the OD values of the ground truth matrix by rows while fixing generated trips by rows.
  - Chaos+Inc+: Equidistributing all the OD values of the ground truth matrix by rows and incrementing all of them by the same $\delta = 0.25$ fixed proportion.
  - Chaos+Inc-: Equidistributing all the OD values of the ground truth matrix by rows and decrementing all of them by the same $\delta = 0.25$ fixed proportion.
  - Multitude: Adapted from Antoniou et al.'s (2016) low demand initialization: $\mathbf{X}^{\mathsf{M}} = \mathbf{X}^{\mathsf{GT}}(r + q\varepsilon)$, with $r = 0.75$, $q = 0.15$ and $\varepsilon \sim N(0, 1/3)$.

  Note that the Incremental+ and Incremental– OD matrices have the same spatial distribution structure and mobility pattern as the ground truth OD matrix, because only a regular increment or decrement has been applied; while the Chaos matrices present a complete perturbation that changes the structure of the OD matrix. In other words, the mobility patterns of the ground truth are different from those of the historical OD matrix, which is used as the seed for the estimation procedure. Figure 4.3 visualizes the differences between three of those OD matrices using heatmaps in red, orange, yellow, and white. As can be seen, the Incremental+ and ground truth OD matrices have the same structural similarity (the one following the MULTITUDE approach also has a similar structure). Furthermore, the Chaos family of OD matrices presents a totally different structure.

- **SPSA variant**: The three described variants used were the Free SPSA (Equation 4.10), the Constrained SPSA (Equation 4.16) and the Penalized SPSA (Equation 4.18). All three methods were used using the improvements: variable normalization, and the automatic selection of the gain sequence parameters $a, A, c$. We generically tested the SPSA variants that add transport information for the test site. Based on the historical OD matrix for each experiment, we used $\beta = 0.25$ constraints on the form $(1 - \beta)\mathbf{X}^H \geq \mathbf{X} \geq (1 + \beta)\mathbf{X}^H$. This means that the OD values can vary between $-25\%$ and $+25\%$ of their original values, which come from a historical and reliable OD matrix.

- **Second term objective function**: In the bi-level problem of Equation 1.6, the second term of differences can be added to the historical OD matrix. This factor is equivalent in mathematical terms to changing $w_2 = 0$ to $w_2 = 1$. Toledo & Kolechkina (2013) studied the effect of this term with respect to the optmization procedure, and concluded that the overall optimization process can be significantly affected even by a small coefficient. We use $w_2 = 1$ because we want to make use of the historical OD matrix and analyze the effect on the quality of the estimated OD matrix.

- **Conjugate gradient approach**: This approach refines the estimated descent direction using Equation 4.6, and we also tested it by analyzing its effects on the estimated OD matrix.



(a) Ground truth OD matrix ($\mathbf{X}^{GT}$)        (b) Incremental+ ($\mathbf{Inc}^+$)        (c) Chaos+Inc+
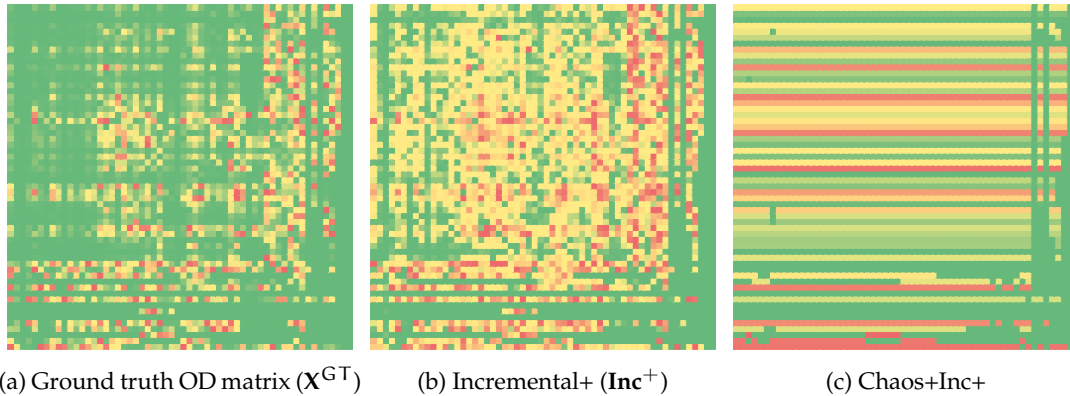
Figure 4.3: Heatmaps of the OD matrices

Therefore, having a full design raises the total number of experiments to 72, which are fully detailed in Appendix B.1. Below, we present some remarkable results and conclusions.

In the next figures, and also in Chapters 5 and 6, the four KPIs that are plotted are, from one side the objective function to verify the descent nature of the optimization procedure; the $R^2$ value as indicator of the fitting between the simulated and observed traffic counts; the total number of trips to check if the approach is able to reach the ground truth number of trips; and the MSSIM to compare the demand pattern with respect to the ground truth OD matrix.

The results from the first set of experiments are shown in Figure 4.4, and these experiments were performed using the historical OD matrix built by the Multitude proposal based on the different variants of SPSA. This initial OD matrix serves as a reliable OD matrix that has a high degree of structural similarity to the ground truth OD matrix, although its demand is lower than the ground truth OD total number of trips.
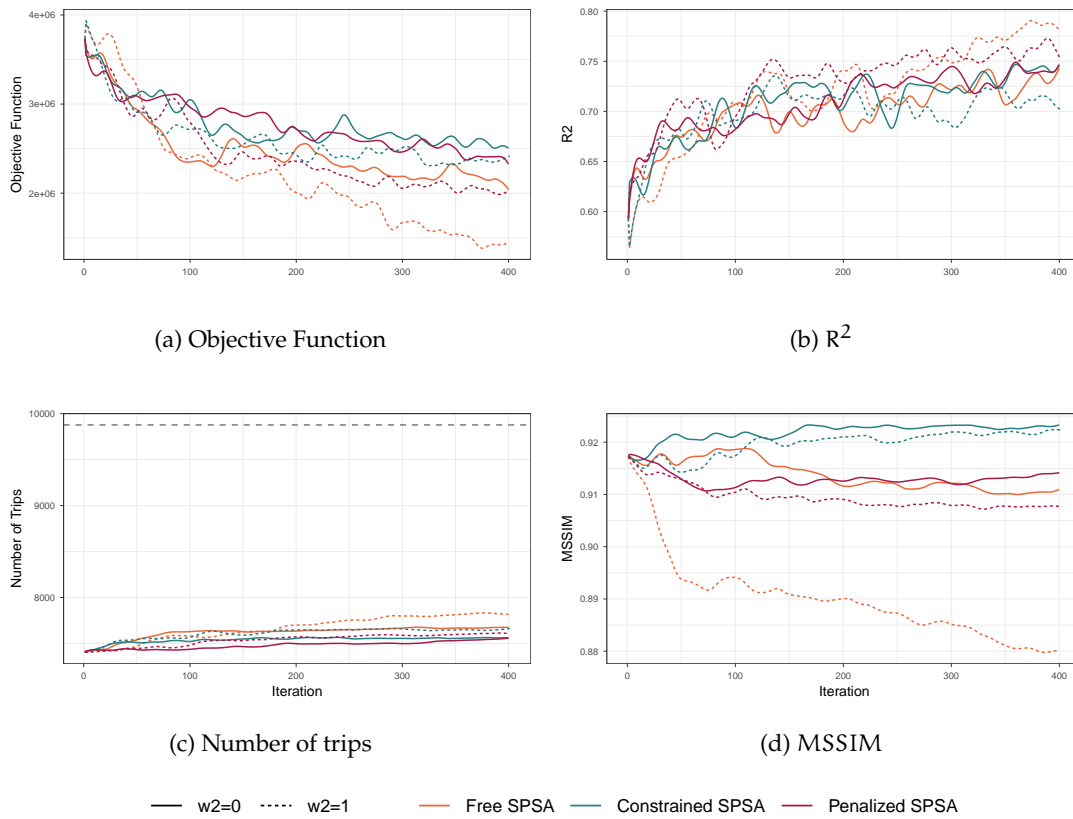
Figure 4.4: Results for different variants of SPSA (without CG) and the Multitude initial OD matrix

Figure 4.4a and 4.4b indicate an acceptable performance of all SPSA variants as a minimization procedure and as a meta-regressor for traffic flow counts, specifically when seeking to increment the fitting between measures. In these cases, the descent is clear, despite the oscillations resulting from the stochastic nature of the algorithm. The total number of trips (Figure 4.4d) cannot significantly increase to where it reaches the ground truth total number of trips, which is $NT(\mathbf{X}^{GT}) = 9878$ trips.

Naturally, the Free SPSA presents a better descent, followed by Penalized SPSA and, finally, Constrained SPSA, which has the worst descent. A similar phenomenon occurs in the evolution of total number of trips, where Free SPSA most approximates the ground truth OD matrix while Penalized and Constrained SPSA approximate it less so, because of the constraints added to the problem. On the other hand, the constraints are reflected in the evolution of MSSIM, where these constraints clearly help maintain the structure of the OD matrix, more similar to the ground truth OD matrix structure. Constrained SPSA seems to perform better. Of the four indicators in Figure 4.4, the second term of the objective function (when $w_2 = 1$) performs better.

On the other hand, Figure 4.5 shows the SPSA performance for the initial OD matrices that are structurally different from the ground truth OD matrix (Chaos, Chaos+Inc+, Chaos+Inc-) when $w_2 = 1$.

(a) Objective Function

(b) R$^2$

(c) Number of trips

(d) MSSIM

Free SPSA — Constrained SPSA — Penalized SPSA — Chaos ⋯⋯ Chaos+Inc– --- Chaos+Inc+
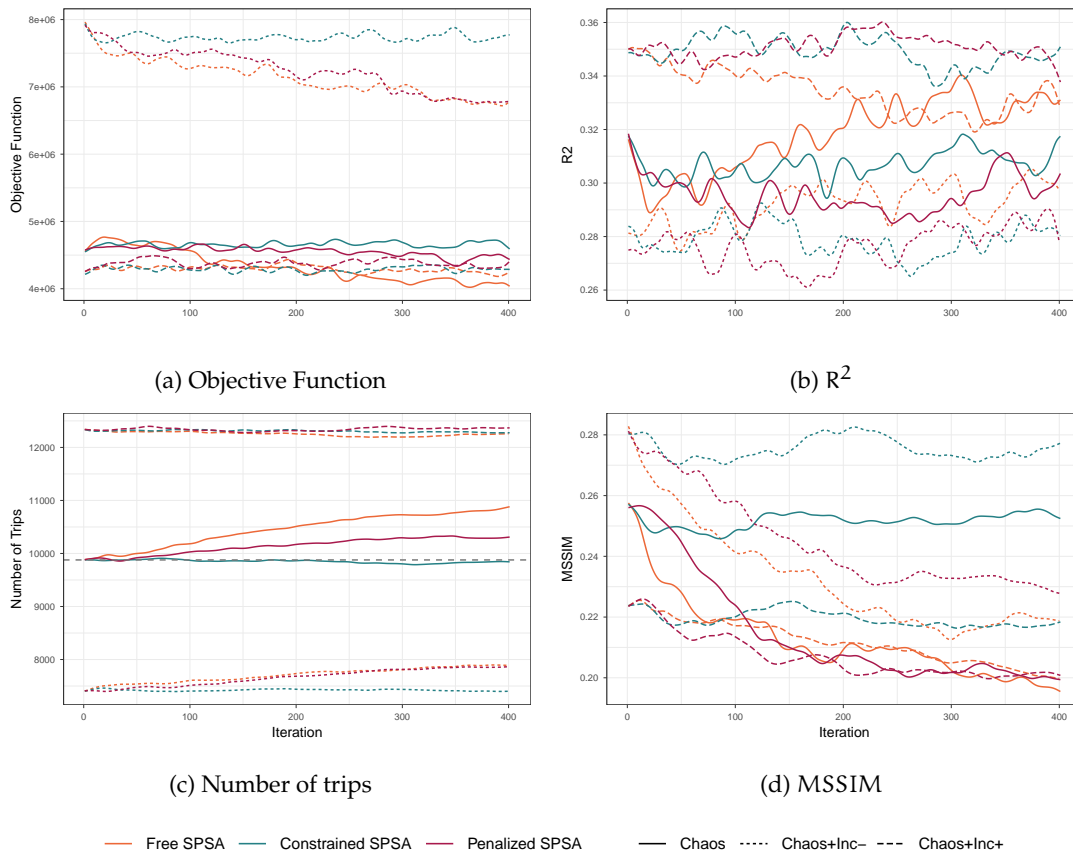
Figure 4.5: Results for different variants of SPSA (without CG and $w_2 = 1$) for different initial OD matrices

Comparing all plots in Figure 4.4 and Figure 4.5 clearly shows that SPSA variants work properly only for those initial OD matrices that are structurally similar to the ground truth OD matrix. When the seed OD matrix is one of the Chaos initializations (Figure 4.5), the SPSA variants do not act as a minimization procedure; they do not increase values; and they do not perturb the matrices at all. The variants are therefore useless in these cases, indicating that the SPSA procedures are in no way robust in terms of the initial OD matrix. This is a drawback to using SPSA when one is not confident about the structure of the seed OD matrices being used.

Figure 4.6 studies the conjugate gradient variant. In this case, the Multitude initial OD matrix was used, and $w_2 = 1$ and Constrained SPSA were selected, since these seem to be the best choice, based on past performance. We can see here that the conjugate gradient technique does not perform better in SPSA and, moreover, the classic stochastic gradient version is better in terms of similarity, as shown in Figure 4.6d.

After studying the performance of these experiments, we consider that SPSA has been significantly studied as an optimization technique for solving the bi-level OD estimation problem in the dynamic case. Its versatility and ease of implementation allow adding many variants for enhancing the results. For instance, adding constraints to the problem helps obtain an OD matrix estimate that remains structurally similar to the initial OD matrix. Moreover, adding the second term in the objective function helps obtain better results, not only in terms of the total number of trips, but also in terms of similarity. On the other hand, the conjugate gradient variation adds no significant improvements, while it uses

more computational memory.  In summary, we can conclude that the initial OD matrix is essential to SPSA-based methods, because SPSA has no effect when the initialization lacks a reliable mobility pattern.
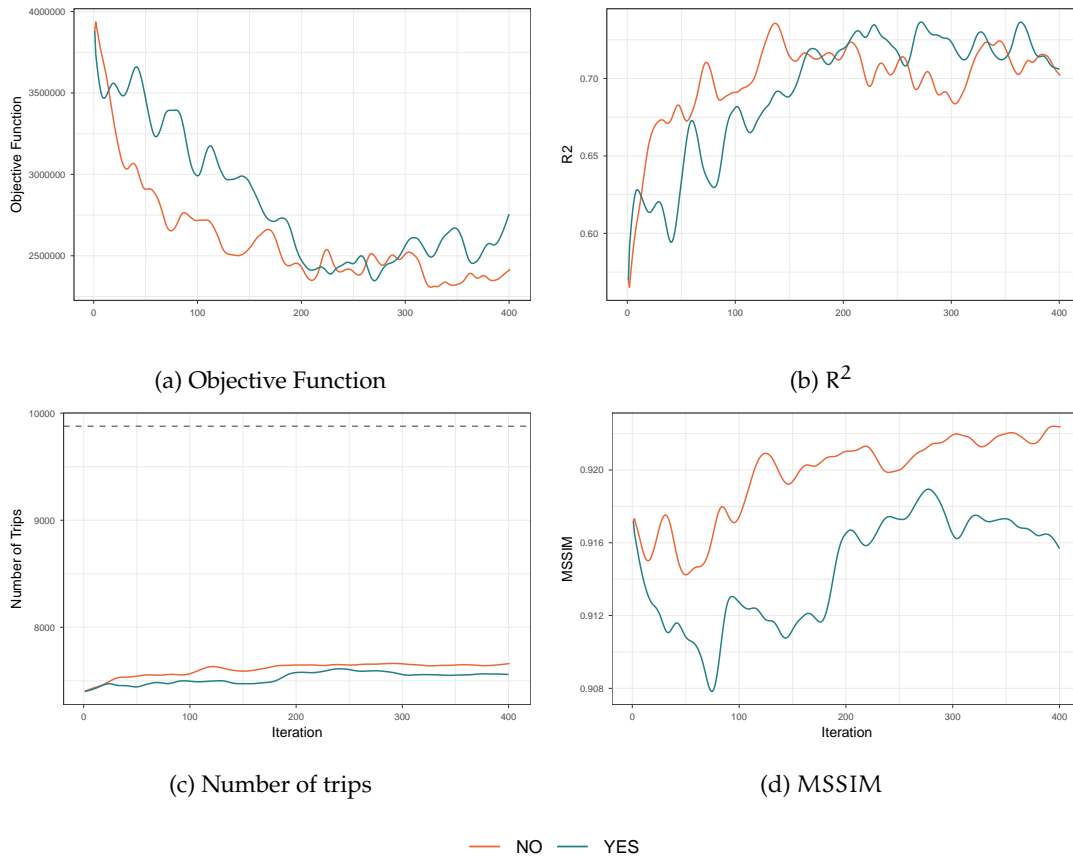


(a) Objective Function

(b) $R^2$

(c) Number of trips

(d) MSSIM

Figure 4.6: Results for the Constrained SPSA with and without the conjugate gradient for Multitude initialization

## 4.5   SPSA with travel times obtained from GPS data

As already mentioned, one of the main advantages of the SPSA approach for solving the OD estimation problem is that it is a method that does not require an analytical gradient calculation, which means that there is no need to know the analytical form of the objective function.  This allows adding different terms in the objective function, such as the discrepancy functions of different traffic measurements, which have been widely explored with link speeds and travel times.  Some examples are Cantelmo, Cipriani, Gemma & Nigro (2014), Kostic et al. (2015), Carrese et al. (2017) and Nigro et al. (2018).

A GPS data set contains many waypoints that describe the trajectories of vehicles circulating through the network. These data sets are usually huge and contain several days' worth of tracking under similar traffic conditions. The proposed methodology aims to gain insights from the data set, which can then be translated to traffic measurements and incorporated into the SPSA procedure as another term in the objective function.
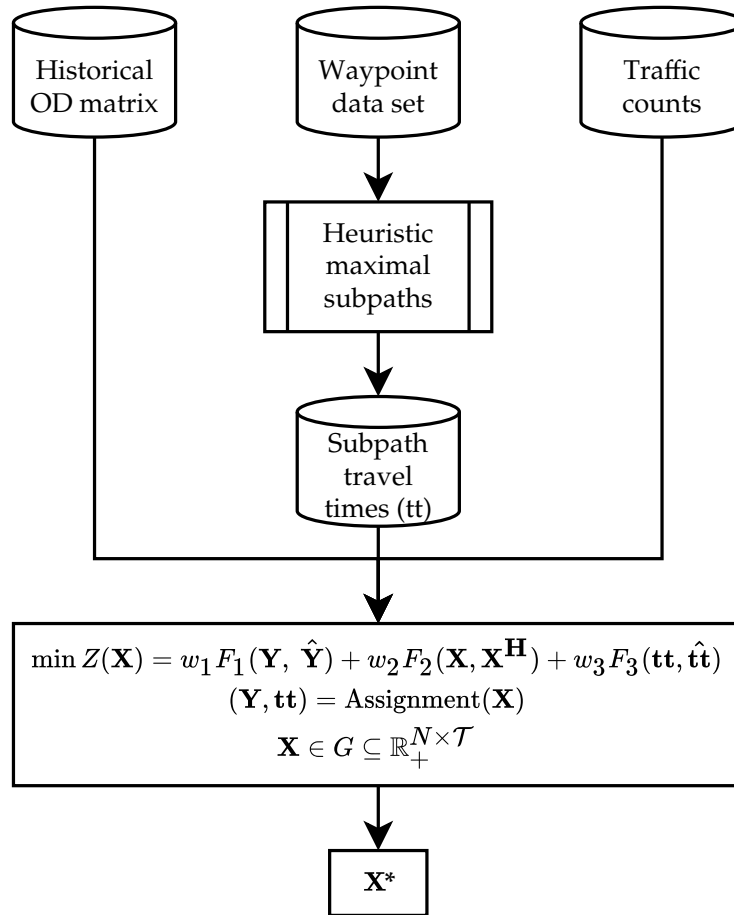
Figure 4.7: Methodological scheme for GPS processing in SPSA

A common option would be to use these trajectories to estimate average OD travel times, which can easily be compared to the outputs of a dynamic traffic assignment model. However, as already mentioned in Section 2.3.1, a commercial GPS data set from an uncontrolled data collection process could carry biases in the OD pairs, as well as poor quality and different vehicle purposes being grouped together, all of which must be taken into account. For this reason, we discarded the first and last waypoint of each trajectory as origin and destination points, and instead opted to extract the travel times of a common subpath with other vehicles.

The methodological approach is described in the logical diagram in Figure 4.7. The GPS data is processed in order to obtain the most used paths in the network and their observed path travel times, $\hat{tt}$. These are map-matched to the dynamic traffic assignment's transport model, thus allowing us to estimate the corresponding travel times, tt.

## 4.5.1   From GPS data to travel times on subpaths

The processing of the waypoint database starts with map-matching the waypoints onto the links in the network (see details in Section 2.4). The output of this process is a data set of trajectories at the link level, such as the one shown in Table 2.4.

The network links in this data set can appear repeatedly when they have been used by tracked vehicles in the real network. Other links appear only once because only one vehicle in the sample passed through it, and some do not appear at all because they were not used by the tracked vehicles.

The presented methodology aims to extract the most reliable information from the sample by selecting the maximal subpaths that are captured most in the floating car data set. Therefore, they have more information on the observed travel times under the given traffic conditions. The heuristic procedure is the following:



Figure 4.8: OD path flows intercepted by the purple link

1) From Table 2.4, the most used link is selected. One example is highlighted in purple in Figure 4.8, with orange representing all the different paths using this link.

2) As shown in the same figure, all the paths that use the selected link are selected. Therefore, simultaneously searching forward and backward adds the previous and subsequent links that maximize the number of paths using this path. In the first iteration, the result is a sequence of three links, with the most used link being in the middle. A fictitious example is shown in Figure 4.9 in order to provide a better understanding of the heuristic procedure for selecting the maximal subpath. Let us imagine that the incoming path (blue part) is shared by 110 trajectories in the data set. In order to add more links, a forward and backward search is conducted to find the distribution of these 110 paths. Therefore, the subset of 110 trajectories is reduced to the 66 trajectories that enlarge the subpath.
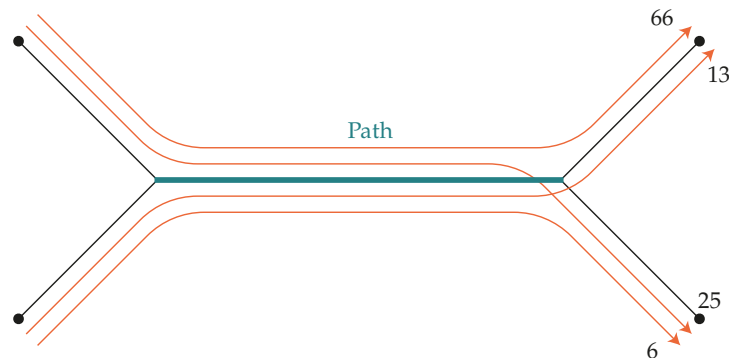


Figure 4.9: Fictitious example to select the subsequent links

3) Step 2 is repeated until the number of paths used by the found sequence is reduced from the number of paths using the initial link to a threshold (for example 20%). This threshold is a design parameter. At the end of this step, subpath $\mathcal{P}_1 = \{L_1^1, \ldots, L_{m_1}^1\}$ is found. The travel time $tt_{\mathcal{P}_1}$ is the average of all the trajectory times circulating over this subpath.

4) Once the previous subpath ($\mathcal{P}_{k-1}$) has been selected, the links forming it are removed from the list of paths, and the remaining paths are split into two new paths.

5) Go back to step 1 to find the next maximal subpath. Stop when there are no available links.

Algorithm 4.5 shows the implementation of the suggested heuristic.

---

**Algorithm 4.5:** From GPS data to estimated travel times

---

**Input: P** Paths described by a sequence of links and link travel times, like the one shown in Table 2.4.
**Input:** Percentual threshold, thrsh.

**Set:** $k \leftarrow 1$;                                                                          // Counter
**while** *more subpaths needed* **do**

> // Step 1:  Select the most used link;
> **Select:** $L_k \leftarrow$ Most used link in **P**;
> **Calculate:** $N_k^L \leftarrow$ number of paths using $L_k$;
>
> **Set:** $\mathcal{P}_k \leftarrow \{L_1\}$;
> **Calculate:** $N_k^{\mathcal{P}} \leftarrow$ number of paths using $\mathcal{P}_k$;
>
> **while** $N_k^{\mathcal{P}} > \text{thrsh} \cdot N_k^L$ **do**
>
>> // Step 2 and 3:  Find Backward and Forward;
>> **Find:**  previous and next links $L_p^i$ from the path $L_n^j$;
>> **foreach** *Combination* $\{L_p^i, L_n^j\}$ **do**
>>
>>> **Count:**  number of paths using it;
>>
>> **Select:**  the maximum used path, $\{L_p^i, L_n^j\}$;
>> **Set:** $\mathcal{P}_k \leftarrow \left\{ L_p^i, \mathcal{P}_k, L_p^j \right\}$;
>> **Calculate:** $N_k^{\mathcal{P}} \leftarrow$ number of paths using $\mathcal{P}_k$;
>
> // Step 4:  Update **P**;
> **Remove:**  Links in $\mathcal{P}_k$ from **P**;
> **Divide:**  Affected paths in **P** in two paths;
> **Set:** $k \leftarrow k + 1$;

**Output:** Return $\{\mathcal{P}_1, \ldots, \mathcal{P}_p\}$

---

The heuristic is valid for static and dynamic OD estimation. In the case of dynamic OD estimation, as in this thesis, the heuristic must be run separately for each time period of the simulation, thereby capturing the changing dynamics of the congestion on the different subpaths found for each time period. From now on, the set of maximal subpaths of a certain network is denoted as $\mathcal{P} = \mathcal{P}_{T_1} \cup \mathcal{P}_{T_2} \cup \cdots \cup \mathcal{P}_{T_\mathcal{T}}$, which is the union of the subpaths found for each of the time periods. Therefore, if the path $k$ is obtained for the time period $T_t$, the corresponding simulated travel time $tt_p$ must also be calculated by simulation for the same time period.

Once the iterative heuristic is performed, the resulting travel times for all the maximal subpaths are calculated as the mean of the different trajectories that use it. If the trajectories map-matched onto the network are sufficient and cover the network, it is expected that these maximal subpaths will have a

reliable estimated travel time and that the entire set will cover the majority of the links in the network, thus providing more information about the traffic dynamics of the network in the time period under study.

### 4.5.2 SPSA with travel times

The extension of the bi-level formulation accounting for measured $\hat{\mathbf{tt}}$ and estimated travel times ($\mathbf{tt} = (tt_1, \ldots, tt_{\mathcal{P}})$) is straightforward, as it simply expands the objective function by adding a third term $F_3(\mathbf{tt}, \hat{\mathbf{tt}})$ to minimize the distance between the measured and estimated travel times of the paths, which are calculated with the heuristic procedure that uses the GPS tracking of vehicles. The formulation is, therefore, as follows:

$$\min \quad Z(\mathbf{X}) = w_1 F_1(\mathbf{Y}, \hat{\mathbf{Y}}) + w_2 F_2(\mathbf{X}, \mathbf{X}^H) + w_3 F_3(\mathbf{tt}, \hat{\mathbf{tt}})$$
$$\text{s. to:} \quad (\mathbf{Y}, \mathbf{tt}) = \text{Assignment}(\mathbf{X}) \tag{4.23}$$
$$\mathbf{X} \geq \mathbf{0}$$

This is also a bi-level problem that solves a dynamic traffic assignment at the lower level to obtain the estimated traffic counts ($\mathbf{Y}$) and travel times ($\mathbf{tt}$). Since SPSA is a estimated gradient method and does not need a specific form of the objective function, adding the new term $F_3$ does not modify the procedure for solving the problem. Moreover, all the proposed variants can also be applied in order to find an estimated OD matrix that improves the fitting of the objective function.

### 4.5.3 Hybridization of SPSA with travel times

Tympakianaki (2018) propose a hybrid SPSA gradient that takes advantage of the differentiable analytical part of the objective function in order to obtain a better estimate of the maximum descent direction in the bi-level problem (Equation 4.7). This research shows that the SPSA procedure's convergence and robustness is improved, which thus inspires the following formulation.

Given the problem shown in Equation 4.23, the dynamic assignment matrix $\mathbf{A}$ states a linear relationship between the traffic counts and the OD flows, which are the optimization problem's variables. Furthermore, the three distance functions ($F_1, F_2, \text{and} F_3$) are the quadratic distance, by which the optimization problem is as follows:

$$\min \quad Z(\mathbf{X}) = w_1 \sum_{l \in \hat{L}} \sum_{t \in \mathcal{T}} \left( \left( \sum_{(i,j) \in N} \sum_{r=1}^{t} a_{ijr}^{lt} x_{ijr} \right) - \hat{y}_{lt} \right)^2 + w_2 \sum_{(i,j) \in N} \sum_{r \in \mathcal{T}} \left( x_{ijr} - x_{ijr}^H \right)^2 + w_3 \sum_{p \in \mathcal{P}} \left( tt_p - \hat{tt}_p \right)^2$$
$$\text{s. to:} \quad \mathbf{tt} = (tt_1, \ldots, tt_{\mathcal{P}}) = \text{Assignment}(\mathbf{X})$$
$$\mathbf{X} \geq \mathbf{0}$$

$$\tag{4.24}$$

In this case, if $a_{jir}^{lt}$ is assumed to be constant in a neighborhood of $x_{ijr}$, the partial derivative of the terms $F_1$ and $F_2$ can be calculated. On the other hand, travel times are not directly related as a function of

the variables of the problem. Therefore, a formal partial derivative cannot be calculated and it must be estimated using the SPSA approach. Using derivative rules of the sum:

$$
\left[\frac{\partial Z}{\partial x_{ijr}}\right]_{hyb} = w_1 \frac{\partial F_1}{\partial x_{ijr}} + w_2 \frac{\partial F_2}{\partial x_{ijr}} + w_3 \frac{\partial F_3}{\partial x_{ijr}} = \tag{4.25}
$$
$$
= 2w_1 \sum_{lt} \left( \sum_{(i,j)\in N} \sum_{r=1}^{t} a_{ijr}^{lt} x_{ijr} - \hat{y}_{lt} \right) a_{ijr}^{lt} + 2w_2(x_{ijr} - x_{ijr}^{H}) + w_3 \frac{\hat{\partial}F_3}{\partial x_{ijr}}
$$

The hybrid gradient approach is expected to be a better approximation of the maximum descent direction, so the method should present better results on convergence and stability. However, as the problem is highly underdetermined, the hybrid gradient could not lead to a better local minimum tan other approaches.

## 4.6   Case Study: Results of SPSA with travel times

We have used the same network in Section 4.4, namely the city of Hillsboro, USA. In this case, the experiments aim to show the effect of adding new data to the SPSA.

We used the methodology described in Section 2.4 to consistently generate the three data sources. We used the same detection layout for this network and simulated 120 days using different random seeds in Vissim. This allowed us to obtain enough GPS trips and time latencies based on the learning process from the physical data. The full GPS data set contains 9.1M waypoints, which equals approximately 109k trips. This data set was transformed to PrT Paths in Visum using the GPX import tool and its map-matching techniques to obtain, from the waypoint sequence, the interpolated travel times at the link level, as explained in Section 2.5. All the experiments used the historical OD matrix that was proposed for benchmarking purposes in MULTITUDE Cost Action (Antoniou et al. (2016)), as in Section 4.4. We also normalized the variables in the objective function without taking the conjugate gradient approach, which also corresponds to the conclusions extracted from the experiments in Section 4.4.

The iterative procedure described in Section 4.5.1 was executed to find the maximal subpaths used by the GPS data in this network. We also considered the departing time period of each path in order to find subpaths at each time period in the dynamic traffic assignment procedure, thus allowing us to capture its dynamicity. The heuristic stops when no more paths are found, which results in 379 subpaths (127 on $T_1$, 123 on $T_2$, and 129 on $T_3$), whose observed path travel times ($\hat{tt}$) are calculated as the sum of their observed link travel times.
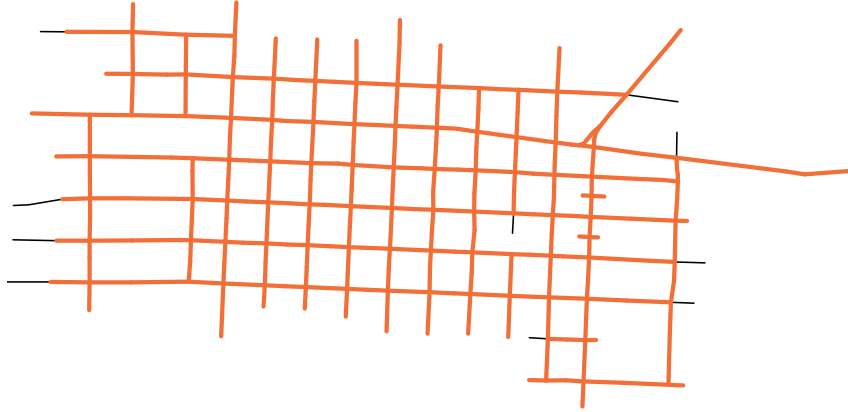
Figure 4.10: The subpaths found on the Hillsboro network

These subpaths and their observed travel times cover the whole network, as shown in Figure 4.10 (orange color means that the link is covered), and they are used by many GPS paths (158 for each, on average), which instill confidence in their use and observed travel times. The average length of these subpaths is 911m. In this case, we consider only subpaths with more than 5 links.

The minimization problem used for the experiments is shown in Equation 4.23. The two methods used in the computational experiments were:

- SPSA with travel times, with the objective function defined by:

$$Z\left(\mathbf{X}\right) = w_1 \sum_{l \in \hat{L}} \sum_{t \in \mathcal{T}} (y_{lt} - \hat{y}_{lt})^2 + w_2 \sum_{(i,j) \in N} \sum_{r \in \mathcal{T}} \left(x_{ijr} - x_{ijr}^H\right)^2 + w_3 \sum_{p \in \mathcal{P}} \left(tt_p - \hat{tt}_p\right)^2 \qquad (4.26)$$

  where $y_{lt}$ and $tt_p$ are calculated at the lower level by performing a dynamic traffic assignment.

- Hybrid SPSA with travel times, with the objective function defined in Equation 4.24.

Following the computational experiments in Section 4.4, we used the Constrained SPSA as the SPSA variant in all cases. Here, the feasible set G is determined by the bounding constraints, as shown in Equation 4.15, and it preserves the structural similarity of the historical OD matrix. A total of 8 experiments were performed in order to see the effects of each proposed innovation. Three main factors were combined to generate the set of experiments:

- The term of the reference historical OD matrix in the objective function ($w_2 = 0$ and $w_2 = 1$).

- The term of travel times on the objective function ($w_3 = 0$ and $w_3 = 1$).

- Hybridization or non-hybridization of the SPSA gradient.

The weight $w_3$ was set as $w_3 = F_1(\mathbf{X}^0)/F3(\mathbf{X}_0)$ in order to equalize the importance of traffic counts and travel times.

Figure 4.11 summarizes the computational results. In terms of the structural similarity index MSSIM, and the total number of trips, when adding both the terms $F_2$ and $F_3$, the total number of trips increases and we get closer to the ground truth $NT(\mathbf{X}^{GT}) = 9878$. Furthermore, the estimated OD matrices are more

reliable, from a structural point of view. The hybrid approach therefore demonstrates more consistent behavior. In the case of the Hybrid SPSA without travel times, the effect of adding the historical OD matrix ($w_2 = 0$) is negligible because the curves overlap. That is because the optimization process is no longer stochastic and the term's contribution is not significant.



(a) Objective Function

(b) $R^2$

(c) Number of trips

(d) MSSIM

Figure 4.11: Results for SPSA with travel times variants

On the other hand, we have experimentally proven that the Hybrid SPSA gradient (Equation 4.25) outperforms the SPSA gradient. As expected, the analytical part of the linearized objective function's gradient is a better approximation of the maximum descent direction. This effect is notably appreciated in the case of $w_3 = 0$ (curves in yellow). However, when $w_3 \neq 0$, the effects of the hybrid SPSA vanish due to the SPSA gradient part (curves in green).

Figure 4.12 shows the simulated and real traffic measurements in the objective function for the case of $w_2 = 1$. Table 4.2 summarizes the $R^2$ adjustment between the measurements and the MSSIM between the ground truth OD matrix used to generate the synthetic data and the estimated OD matrix.

Regarding traffic counts, including travel times in the objective function decreases the $R^2$. On the other hand, the two variants of SPSA with travel times properly adjust the path travel times with respect to the real measurements, but SPSA with travel times shows greater $R^2$ values. The MSSIM always increases when $w_2 = 1$, because the objective function considers the historical OD matrix. On the other hand, including travel times in SPSA improves the structural similarity with respect to the ground truth OD matrix. In the case of Hybrid SPSA, the MSSIM value decreases because adding the travel times converts the algorithm from deterministic to stochastic, as seen. In Figure 4.12, a remarkable group of outliers is

(a) Traffic counts        (b) Travel Times

SPSA without TT    SPSA with TT    HybSPSA without TT    HybSPSA with TT
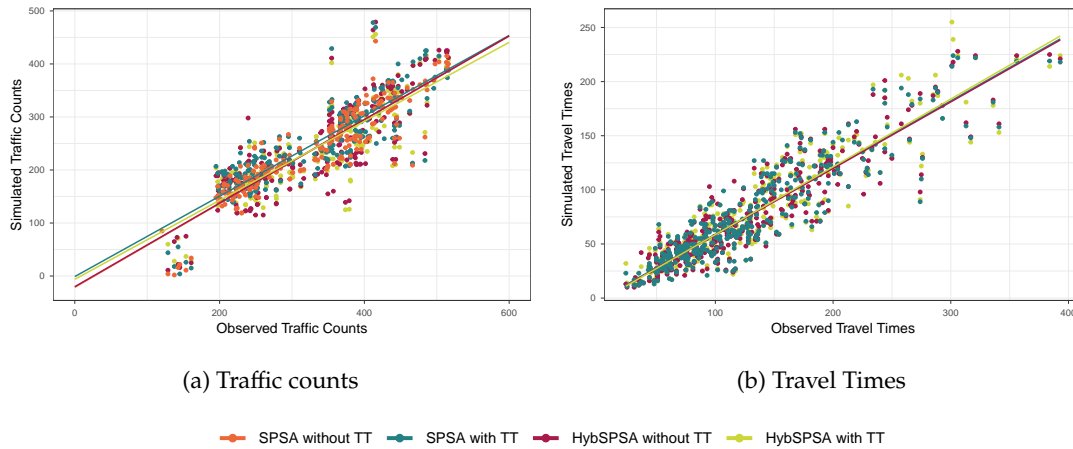
Figure 4.12: $R^2$ between simulated and real measurements

Table 4.2: $R^2$ between traffic measurements and MSSIM between the estimated and the ground truth OD matrix

|  |  | SPSA | | Hybrid SPSA | |
|---|---|---|---|---|---|
|  |  | no tt | tt | no tt | tt |
| $R^2$ Traffic | $w_2 = 0$ | 0.8297 | 0.7614 | 0.7598 | 0.6485 |
| Counts | $w_2 = 1$ | 0.8319 | 0.7488 | 0.7684 | 0.7243 |
| $R^2$ Travel | $w_2 = 0$ | - | 0.8312 | - | 0.8371 |
| Times | $w_2 = 1$ | - | 0.8426 | - | 0.8297 |
| MSSIM | $w_2 = 0$ | 0.9189 | 0.9282 | 0.9371 | 0.9283 |
|  | $w_2 = 1$ | 0.9207 | 0.9318 | 0.9371 | 0.9278 |

shown for traffic count measurements. All these points pertain to the same detector in the outer limits of the study area and they appear in all methods, thus indicating that traffic counts cannot be satisfied without perturbing the global fit. A general trend can be highlighted: the SPSA without travel times method has a greater $R^2$ because the estimation procedure behaves as a metamodel that modifies the output matrix in such a way that traffic counts are forced. Nevertheless, as travel times are included, a compromise must be made between traffic counts and travel times and a jointly enhancement requires to relax traffic counts fit for obtaining a global benefit in the estimation of the output matrix.

## 4.7 Conclusions of SPSA

The SPSA method has been widely used for solving the OD estimation problem, mainly when the use of new traffic measurements is desired, because its versatility allows easily adding new discrepancies terms to the objective function. This chapter has studied and analyzed the SPSA in depth in order to understand its behavior when solving the OD estimation problem, as well as to improve it by modifying the SPSA with a focus on its performance in estimating the specific OD matrix.

We conducted a complete study of previous research works in order to compile proposed modifications to SPSA for OD estimation. This study ends with an enhanced SPSA methodology that specifically

works for OD estimation while tackling its inherent weaknesses when used for OD estimation. Two contributions of this thesis are the normalization of variables and the automatization of selecting the SPSA parameters, which have been used in all the experiments of this chapter.

The proposed improvements to SPSA aimed at limiting the total number of trips of the OD matrix and improving the structural similitude between the historical and the resulting OD matrix, led to two alternative formulations. First, a Constrained SPSA in terms of bounding constraints on the changes in the entries of $\mathbf{X}$. Second, a Penalized SPSA, adding a penalty term to the objective function to limit the changes in the OD matrix values. Both versions provided similar results in term of better convergence and quality of the estimated OD matrix when compared to the ground truth OD matrix.

In this sense, Section 4.4 first studied using SPSA without adding travel times. The objective of this was to find the best settings for estimating OD matrices. Since SPSA is a stochastic estimated gradient based method, the initial OD matrix is crucial for successfully obtaining a structurally similar estimate, as proven. The reliability of such an initial OD matrix is also helpful as a second term in the bi-level optimization form of the OD estimation problem (Equation 1.6). The SPSA procedure performs better in terms of similarity when adding some constraints, both when using constraints to define a feasible set and when adding the penalizing term in the objective function. Moreover, adding the second term that refers to the historical OD matrix helps maintain the structure of the estimation, which is desirable for mobility patterns.

On the other hand, SPSA is not a suitable method when the reliability of the OD matrix is not certain. As shown in Figure 4.4, the experiments performed using structurally different OD matrices show bad performance of SPSA, which acts as a mere random perturbation method without finding an optimal direction for improving any of the indicators.

Adding ICT traffic measurements is the main objective of this thesis, with the aim of taking advantage of ICT technologies and Big Data. Section 4.5 presents an innovative and specific approach for transforming a GPS tracking data set into traffic measurements that define the traffic phenomena in the network during the studied time periods. With this heuristic procedure, travel times on the most used subpaths are estimated and added to the objective function as a third term of discrepancies between estimated measurements and its corresponding simulated measurements. This methodology has been tested and validated using a synthetic data generation process and, finally, by observing its performance.

In terms of similarity, adding new data sources contributes positively to the SPSA performance, which increases the final value of $MSSIM$ and also leads to the total number of trips reaching the ground truth value. Thus, we have proved that the proposed methodology is suitable for adding information from GPS tracking.

# 5

# Analytical approaches to solve DODME

*In this chapter, we analyze the analytical approaches based on the assignment matrix. We formulate the dynamic version of the Spiess (1990) method and present some variants. We show and test its performance with some synthetic experiments.*

As we introduced in Chapter 1, the OD estimation problem of Equation 1.6 is the reference to estimate OD matrices, both in the static and in the dynamic case. In the first case, the researchers have made wide use of the analytical approaches, as described in Section 1.2.1, to solve the bi-level optimization problem. The key object in these approaches is the static assignment matrix, Equation 1.7, that converts the objective function to a nonlinear optimization problem once assumed constant on a reduced neighborhood. Selecting appropriate differentiable distance functions, one can design computationally efficient algorithms that reevaluate the static assignment matrix at each iteration and present nice properties for convergence and stability, Codina & Montero (2006), Lundgren & Peterson (2008), Spiess (1990).

On the step moving to the dynamic traffic assignment case that we described in Section 1.3, the congestion effects and traffic dynamics are taken into consideration and the assignment matrix involves the time dimension twice. The terms of the dynamic assignment matrix, Equation 1.17, are $a_{ijr}^{lt}$ and represent the proportion of the OD flow $x_{ijr}$ that departs at time period $r$ from origin $i$ to destination $j$ and is detected on link $l$ at time period $t \geq r$. The dynamic assignment matrix acts as a linear function that relates the OD flows with the traffic counts detected on the sensors, and therefore replaces the dynamic traffic assignment.

The linearization of the relationship between OD flows and the traffic counts is indeed the first term of the Taylor expansion of the unknown specific dynamic traffic assignment function, Equation 1.19. In the dynamic case, where temporal evolution is important, this linearization cannot properly account for the impacts of traffic dynamics, time dependencies and route choice alternatives induced by congestions. As Frederix et al. (2011, 2013) highlight, counts in and downstream of congestion are not informative of demand, but of discharge capacity. Additional terms of such Taylor expansion would help to capture the assignment matrix's sensitivity to spatial and temporal changes and its effects. However, Frederix et al. (2013) state that the use of a reference OD matrix that presents the correct traffic regime is a right starting point, despite it cannot ensure convergence to an estimated OD matrix that represents correctly the demographics of the network.

Some researchers (Frederix et al. (2013), Toledo & Kolechkina (2013), Yang et al. (2017)) drew attention to the role played by the quality of the assignment matrix, which results from the lower level assignment

process when estimating the flows used in the upper level. Therefore, those researchers have proposed either analytical or empirical approaches for improving it. While Frederix et al. (2013) offers a relevant theoretical contribution, Toledo & Kolechkina (2013) provides more insight into applying it in larger networks. However, their paper mentions Spiess (1990) as one of the tested alternatives, and yet they do not provide further details about its extension to dynamic scenarios. It seems to be a particular case of their approach, which uses second-order derivatives and complex numerical optimization procedures (e.g., Armijo rules to compute the step length), which require higher computational effort. Therefore, in this chapter, we seek to validate the modification of the Spiess procedure using, on the one hand, a first-order approach to the assignment matrix that is provided by a Dynamic Traffic Assignment; and, on the other, an ad hoc reformulation of the analytical calculation of the gradient that is suitable for a straightforward calculation of the step length at each iteration.

The problem studied in this chapter is the one shown in Equation 1.8, but in the dynamic case. In order to state it clearly, we think necessary formulating again the problem in this chapter.

Given a network with a set of links $L$, a set $I$ of OD pairs, and the set of time periods $\mathcal{T}$. The goal of the dynamic OD-matrix estimation problem is to find a feasible vector (OD-matrix) $\mathbf{X}^* \in G \subseteq \mathbb{R}_+^{I \times J \times \mathcal{T}}$, where $\mathbf{X}^* = (x^*_{ijr}), i \in I, j \in J, r \in \mathcal{T}$, consists of the demands for all OD pairs. It can be assumed that, when assigning the time-sliced OD matrices onto the links of the network, it should be done according to an assignment proportion matrix $\mathbf{A} = \left[ a^{lt}_{ijr} \right]$, where each element in the matrix is defined as the proportion of the OD demand $x_{ijr}$ that uses link $l$ at time period $t$. The notation $\mathbf{A} = \mathbf{A}(\mathbf{X})$ is used to indicate that, in general, these proportions depend on the demand but the main assumption on the analytical approaches is that the assignment matrix is constant on a reduced neighborhood, as mentioned. The linear relationship between the flow count on a link and the given OD pair is in matrix form, which thus sets the vector of detected flows as $\mathbf{Y} = (\mathbf{Y}_1, \ldots, \mathbf{Y}_T) = (y_{11}, \ldots, y_{L1}, \ldots, y_{1T}, \ldots, y_{LT})$ and the vector of OD flows as $\mathbf{X} = (\mathbf{X}_1, \ldots \mathbf{X}_T) = (x_{i_1 j_1 1}, \ldots, x_{i_1 j_J 1}, \ldots, x_{i_1 j_1 T}, \ldots, x_{i_1 j_J T})$. The relationship can be expressed as a matrix product, that is

$$\mathbf{Y} = \mathbf{A}(\mathbf{X}) \cdot \mathbf{X} \text{ with } \mathbf{A} = \begin{pmatrix} \mathbf{A}_1^1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{A}_1^2 & \mathbf{A}_2^2 & \mathbf{0} & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{A}_1^T & \cdots & \mathbf{A}_{T-1}^T & \mathbf{A}_T^T \end{pmatrix} \text{ where } \mathbf{A}_r^t = \begin{pmatrix} a^{l_1 t}_{i_1 j_1 r} & \cdots & a^{l_1 t}_{i_I j_1 r} \\ \vdots & \ddots & \vdots \\ a^{l_L t}_{i_1 j_1 r} & \cdots & a^{l_L t}_{i_I j_J r} \end{pmatrix} \tag{5.1}$$

where $\mathbf{A}^{rt}$ represents the assignment matrix for the departing flows at time window $r$ detected at time window $t$ and, therefore, $\mathbf{A}$ is a lower-diagonal matrix, because OD flow departing at time $r$ cannot pass through link $l$ at time $t < r$. The resulting bi-level optimization problem, once replacing the static assignment by the dynamic assignment matrix, is as follows:

$$\min \quad Z(\mathbf{X}) = w_1 F_1 \left( \mathbf{A}(\mathbf{X})\mathbf{X}, \hat{\mathbf{Y}} \right) + w_2 F_2 \left( \mathbf{X}, \mathbf{X}^H \right)$$
$$\text{s. to:} \quad \mathbf{X} \geq \mathbf{0} \tag{5.2}$$

## 5.1    Analytical approaches and ICT traffic measurements

The main objective of this thesis is to add ICT traffic measurements to the OD estimation problem, aiming to reduce the underdetermination of the problem supplementing it with new data sources, different to

the used traffic counts. The simulation optimization-based procedures, Chapter 4, are very suitable for introducing different traffic measurements, since they do not necessarily need an analytical objective function to find a sound estimated OD matrix. Indeed, we propose a methodology to use GPS data in the SPSA procedure, after processing it and transforming to reliable travel times.

However, the case of analytical approaches is different. As it is already shown, the analytical approaches base their functionality on finding relationships between the variables of the system, that are the OD flows, and the traffic measurements, which are the traffic counts. The traffic counts are algebraically relationable to the OD variables, because they are just sums of proportions of them, as explained. The other traffic measurements, as link speeds or travel times are not that easy, since more complex dynamics relate these measurements to the OD flows, such as a traffic assignment. In the dynamic case, the traffic assignment and the consequences of reflecting congestion effects, makes it infeasible to find a suitable, but simple, relationship between OD flows and other new ICT traffic measurements.

In this chapter, we analyze in depth the analytical approaches to solve the dynamic version of the bi-level optimization problem. Therefore, we explore the Spiess approach, aiming at achieving a deeper understanding of its robust and deep functionality, to explore the possibilities of finding ways of relating them with the ICT data.

## 5.2 Dynamic Spiess approach: Spiess gradient based method

In this section, we propose the extension of the Spiess' static approach (Spiess (1990)), described in Section 1.2.1, to the dynamic case because it is one of the most robust approaches in practice. That means to make the calculations but accounting too for the time dependencies. Therefore, the proposed approach, that we name it *the Dynamic Spiess*, does not account for the propagation effects, but it explicitly considers the time dependencies. Moreover, after the mentioned experiences with the inclusion of a second term in the objective function, we include also this term to the formulation, that compares the OD matrix to the historical OD matrix, that is assumed to be a reference reliable OD demand.

As mentioned earlier, the extrapolation of the Spiess method to a dynamic traffic assignment (DTA) is possible by including the time windows to the entire formulation, as follows:

$$
\begin{aligned}
\min \quad & Z\left(\mathbf{X}\right) = w_1 \sum_{l \in \hat{L}} \sum_{t \in \mathcal{T}} \left(y_{lt} - \hat{y}_{lt}\right)^2 + w_2 \sum_{(i,j) \in N} \sum_{r \in \mathcal{T}} \left(x_{ijr} - x_{ijr}^{H}\right)^2 \\
\text{s. to:} \quad & y_{lt} = \sum_{(i,j) \in N} \sum_{r=1}^{t} a_{ijr}^{lt} x_{ijr} \\
& \mathbf{X} \geq \mathbf{0}
\end{aligned}
\tag{5.3}
$$

As in Spiess (1990), the proposed iterative procedure is a more realistic approach of the gradient, based on a relative change of the demand. And the gradient and the optimal step must be calculated following the original calculations but including the time dependencies and the second term of the objective function. That is:

$$x_{ijr}^{(k+1)} = x_{ijr}^{(k)} \left( 1 - \lambda \frac{\partial Z}{\partial x_{ijr}} \right) = \tag{5.4}$$

$$= x_{ijr}^{(k)} - \lambda x_{ijr}^{(k)} \frac{\partial Z}{\partial x_{ijr}}$$

where the partial derivative of the objective function is calculated using the chain rule:

$$\frac{\partial Z}{\partial x_{ijr}} = 2w_1 \sum_{l \in \hat{L}} \sum_{t=1}^{r} (y_{lt} - \hat{y}_{lt}) \frac{\partial y_{lt}}{\partial x_{ijr}} = 2w_1 \sum_{l \in \hat{L}} \sum_{t=1}^{r} (y_{lt} - \hat{y}_{lt}) a_{ijr}^{lt} \tag{5.5}$$

The optimal length, $\lambda^*$, is that one larger enough to avoid small steps and long convergences but smaller enough to avoid increasing the value of the objective value on the next point. Mathematically, it is desired to use a value $\lambda^*$ such that $Z\left(X^{(k+1)}\right)$ is minimum. That is:

$$\min_{\lambda} \quad h(\lambda) = Z\left(X^{(k+1)}\right) = w_1 \sum_{l \in \hat{L}} \sum_{t \in \mathcal{T}} (y_{lt} - \hat{y}_{lt})^2 + w_2 \sum_{(i,j) \in N} \sum_{r \in \mathcal{T}} \left( x_{ijr}^{(k+1)} - x_{ijr}^H \right)^2$$

$$\text{s. to:} \quad y_{lt} = \sum_{(i,j) \in N} \sum_{r=1}^{t} a_{ijr}^{lt} x_{ijr}^{(k+1)} \tag{5.6}$$

$$\lambda \frac{\partial Z}{\partial x_{ijr}^{(k)}} \leq 1$$

In order to solve this 1-dimensional optimization problem, we must compute $h'(\lambda) = 0$ and therefore, we first calculate:

$$y'_{lt} = \frac{dy_{lt}}{d\lambda} = \sum_{(i,j) \in N} \sum_{r=1}^{t} \frac{\partial y_{lt}}{\partial x_{ijr}^{(k+1)}} \frac{dx_{ijr}^{(k+1)}}{d\lambda} = -\sum_{(i,j) \in N} \sum_{r=1}^{t} a_{ijr}^{lt} x_{ijr}^{(k)} \frac{\partial Z}{\partial x_{ijr}} \tag{5.7}$$

where we used the chain rule and Equation 5.4 to measure the change of the traffic counts along the gradient direction. And therefore, to find the minimum:

$$h'(\lambda) = \sum_{l \in \hat{L}} \sum_{r \in \mathcal{T}} \frac{\partial h}{\partial y_{lt}} y'_{lt} = 2w_1 \sum_{l \in \hat{L}} \sum_{r \in \mathcal{T}} \left( \sum_{(i,j) \in N} \sum_{r=1}^{t} a_{ijr}^{lt} x_{ijr}^{(k+1)} - \hat{y}_{lt} \right) = \tag{5.8}$$

$$= 2w_1 \sum_{l \in \hat{L}} \sum_{r \in \mathcal{T}} (y_{lt} + \lambda y'_{lt} - \hat{y}_{lt}) = 0$$

and results to determine the optimal value of $\lambda^*$:

$$\lambda^* = \frac{\displaystyle\sum_{l \in \hat{L}} \sum_{r \in \mathcal{T}} (y_{lt} - \hat{y}_{lt}) y'_{lt}}{\displaystyle\sum_{l \in \hat{L}} \sum_{r \in \mathcal{T}} y'_{lt}{}^2} \tag{5.9}$$

Algorithm 5.1 summarizes the procedure, where as it can be seen, a full dynamic traffic assignment at each iteration is needed to update the dynamic assignment matrix, $\mathbf{A}$ at the lower level of the bi-level optimization problem.

---

**Algorithm 5.1:** Dynamic Spiess algorithm

---

**Input:** A seed OD matrix, $\mathbf{X}^0$
**Input:** A stopping criterion, `thrsh_stop`
**Input:** A maximum number of iterations, `IterMax`

**Evaluate:** DTA $\left(\mathbf{X}^0\right)$ and obtain $\mathbf{A} = \mathbf{A}(\mathbf{X}^0)$;
**Set:** $\mathbf{X} \leftarrow \mathbf{X}^0$;

**foreach** $\mathtt{i} = 1, \ldots, \mathtt{IterMax}$ **do**

    **Evaluate:** $Z(\mathbf{X})$ ;                                                       // Equation 5.3
    **Evaluate:** $\nabla Z(\mathbf{X})$;                                                    // Equation 5.5
    **Evaluate:** $\lambda^*$ ;                                                             // Equation 5.9

    **if** $any(\lambda^* * \nabla Z(\mathbf{X})) > 1$ **then**
        $\lambda^* \leftarrow 1/\max(\nabla Z(\mathbf{X})) - \varepsilon$ ;                  // $\varepsilon$ is to avoid numerical problems

    **Set:** $\mathbf{X} \leftarrow \mathbf{X}(1 - \lambda^* \nabla Z(\mathbf{X}))$ ;      // Equation 5.4

    **if** `rel_diff` < `thrsh_stop` **then**
        **Stop**;

    **Evaluate:** DTA $(\mathbf{X})$ and obtain $\mathbf{A} = \mathbf{A}(\mathbf{X})$;

**Set:** $\mathbf{X}^* \leftarrow \mathbf{X}$ ;                                          // $\mathbf{X}^*$ represents the estimated OD matrix
**Output:** Return $\mathbf{X}^*$

---

### 5.2.1  Dynamic Spiess variants

With the aim of improving the computational performance, an analysis of the computational burden of the algorithm has been done. In this sense, the most expensive step is the DTA procedure to update the dynamic assignment matrix. Assuming again that small changes on the OD matrix imply small changes to the dynamic assignment matrix, we reduce the number of assignments to improve the computational performance. On the other hand, we propose and analyze an alternative distance term between OD matrices in the objective function.

**Reassigning at convergence**

As in Spiess (1990), the iterative procedure used to solve the minimization problem on Equations 5.3 employs gradient methods, Equation 5.4, 5.5 and 5.9, and this requires one full assignment of the OD matrices at each single iteration of the minimization procedure. The assignment matrix, $\mathbf{A} = \left[a_{ijr}^{lt}\right]$, depends directly on $\mathbf{X}$; so each iteration of the minimization problem requires a single assignment of $\mathbf{X}$ onto the network, which increases the computational time. Let us assume that the assignment matrix does not change significantly at each iteration. Then, reassigning at every single iteration is not needed and the computational time would significantly be reduced, because DTA assignment is a time-consuming procedure. In this case, we propose to distinguish the iterations between major iterations (when an assignment is required at the lower level) and minor iterations (when only the minimization

iterations at the upper level are considered). The jump from the upper level to the lower level is made only when a convergence criterion is satisfied. The assignment matrix is then recalculated, and more minor iterations are launched. At the end, a dynamic traffic assignment is needed to update the resulting traffic counts according to an updated assignment matrix. Algorithm 5.2 shows the new design, where two thresholds determine if it is needed to launch a DTA and update the assignment matrix or the stopping criterion is reached. Both thresholds consider the relative decrease of the objective function.

---

**Algorithm 5.2:** Dynamic Spiess algorithm innovation: Reassigning at convergence

---

**Input:** A seed OD matrix, $\mathbf{X}^0$
**Input:** A DTA criterion, `thrsh_DTA`
**Input:** A stopping criterion, `thrsh_stop`
**Input:** A maximum number of iterations, `IterMax`

**Evaluate:** DTA $\left(\mathbf{X}^0\right)$ and obtain $\mathbf{A} = \mathbf{A}(\mathbf{X}^0)$;
**Set:** $\mathbf{X} \leftarrow \mathbf{X}^0$;

**foreach** $\mathtt{i} = 1, \ldots, \mathtt{IterMax}$ **do**

    **Evaluate:** $Z(\mathbf{X})$ ;                                                        // Equation 5.3
    **Evaluate:** $\nabla Z(\mathbf{X})$;                                                    // Equation 5.5
    **Evaluate:** $\lambda^*$ ;                                                              // Equation 5.9

    **if** *any*$(\lambda^* * \nabla Z(\mathbf{X})) > 1$ **then**
        $\lambda^* \leftarrow 1/\max(\nabla Z(\mathbf{X})) - \varepsilon$ ;                   // $\varepsilon$ is to avoid numerical problems

    **Set:** $\mathbf{X} \leftarrow \mathbf{X}(1 - \lambda^*\nabla Z(\mathbf{X}))$ ;                  // Equation 5.4

    **if** *rel_diff* < `thrsh_stop` ;                                                        // Stopping criterion
    **then**
        **Stop**;

    **if** *rel_diff* < `thrsh_DTA` ;                                                         // Reassignment criterion
    **then**
        **Evaluate:** DTA $(\mathbf{X})$ and obtain $\mathbf{A} = \mathbf{A}(\mathbf{X})$;

**Set:** $\mathbf{X}^* \leftarrow \mathbf{X}$ ;                                                            // $\mathbf{X}^*$ represents the estimated OD matrix
**Evaluate:** DTA $(\mathbf{X})$ ;                                                                         // Final Assignment

**Output:** Return $\mathbf{X}^*$

---

#### Entropy function as a distance function between OD matrices

Due to criticism of using Euclidean distances of Frederix et al. (2013), using a different distance function is another alternative for the objective function. Furthermore, as already mentioned and shown in Chapter 3, Djukic (2014) also shows that using a Euclidean distance term can result in two matrices that have very different structures but maintain the same distance value with respect to the reference matrix. However, MSSIM has been discarded as we seek for an easy differentiable distance function that permits us to obtain the gradient and use the maximum descent iterative procedure.

In this sense, the classical entropy function, Ortúzar & Willumsen (2011), has been chosen because of its structural meaning and ease to derive. Then, the objective function is still similar to the one in Equation 5.3, but the differences function $F_2$ is replaced by the entropy function, that is:

$$Z(\mathbf{X}) = w_1 \sum_{l \in \hat{L}} \sum_{t \in \mathcal{T}} (y_{lt} - \hat{y}_{lt})^2 + w_2 \sum_{(i,j) \in N} \sum_{r \in \mathcal{T}} x_{ijr} \log \left( \frac{x_{ijr}}{x_{ijr}^H} \right) \tag{5.10}$$

And therefore, the partial derivatives are also affected, so it results:

$$\frac{\partial Z}{\partial x_{ijr}} = 2w_1 \sum_{l \in \hat{L}} \sum_{t \in \mathcal{T}} (y_{lt} - \hat{y}_{lt})^2 \, a_{ijr}^{lt} + w_2 \left( \log \left( \frac{x_{ijr}}{x_{ijr}^H} \right) + 1 \right) \tag{5.11}$$

## 5.3 Case Study: Results of Dynamic Spiess different versions

All Dynamic Spiess proposed variants have been tested using the same test site than in the experiments of SPSA, which is the downtown of the city of Hillsboro (Oregon, United States), see details in Section 4.4. Using the same network permits us to analyze and compare both methodologies. As in the previous experiments with SPSA, the convergence threshold has been deactivated to study also the effects of the methodology on the late iterations.

As in the previous chapter, following the framework of Antoniou et al. (2016), we prepared a set of synthetic experiments to test the different versions and variants proposed for Dynamics Spiess. As in the previous chapter, the historical OD matrix and the addition of the second term of the objective function are design factors.

Therefore, the factors for a full factor experiments are listed, which are similar to those in Chapter 4:

- **Historical OD matrix**: 6 different initializations have been used in order to contemplate different situations, from similar-structure matrices with different number of trips to non-similar-structure matrices. These are the same initializations as in SPSA testing experiments, we refer to Section 4.4 to see the details of each initial OD matrix.

- **Dynamic Spiess variant**: The three variants described have been used. That are:
  - **Original Dynamic Spiess**: The one presented in Algorithm 5.1. A major iteration assigns the OD matrix at each iteration, $\mathbf{X}^{(k)}$, in order to update the assignment matrix $\mathbf{A}^{(k)}$. This is done after a minor iteration at the upper level that updates the OD matrix iteration by iteration.
  - **Dynamic Spiess reassigning at convergence**: Major iterations are done after the upper level optimization procedure converges, when the relative error is lower than $\mathtt{thrsh\_DTA} = 10^{-4}$.
  - **Dynamic Spiess with entropy**: The Original Dynamic Spiess in Algorithm 5.1, but using the entropy function as a second term in the objective function, as shown in Equation 5.10.

- **Second term objective function**: As in SPSA experiments, we will check the effect of using or not the second term of the objective function, that is between $w_2 = 0$ or $w_2 = 1$.

The full design accounts for 30 different experiments, which are fully detailed in the Appendix B.3. Following these lines, we present some remarkable results and conclusions.

### 5.3.1   Original Dynamic Spiess results

The results of the 6 different initial OD matrices and for $w_2 = 0$ and $w_2 = 1$ are shown in Figure 5.1, where it can be seen that in all cases, the Dynamic Spiess approaches show an excellent descent of the objective function and convergence behavior.

The same KPIs as in Chapter 4 are plotted in these experiments, which are the objective function, the $R^2$, the total number of trips and the $MSSIM$.
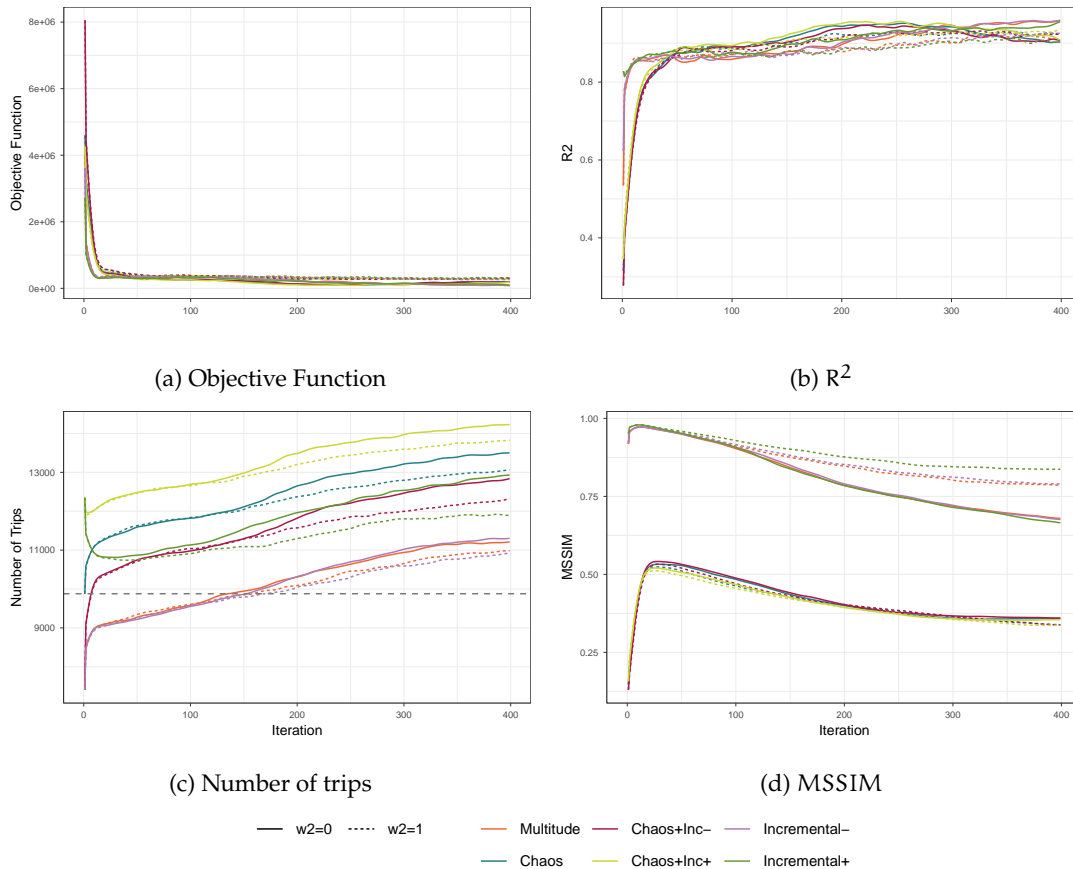


(a) Objective Function                                     (b) $R^2$

(c) Number of trips                                        (d) $MSSIM$

Figure 5.1: Results for Dynamic Spiess for different initial matrices

Dynamic Spiess behaves very well on all the different variants, working as a meta-regression procedure in which the fitting between traffic counts, $R^2$ becomes high no matter the initial matrix. However, the number of trips presents in all cases an uncontrolled increase far from the objective ground truth number of trips. A deeper study on this behavior is shown in Figure 5.2, where two boxplots of the growth of all the different OD pairs is graphed. The growth has been calculated as a quotient between the final value and the initial value for each OD pair.

(a) Boxplot by quantiles on the initial value

(b) Boxplot by quantiles on the final value
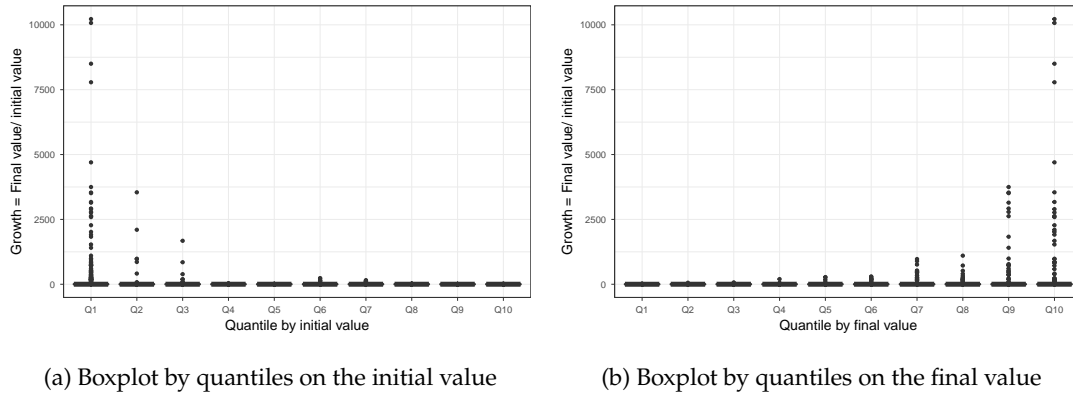
Figure 5.2: Boxplots showing the growth of the OD flows after the Dynamic Spiess procedure

We see in Figure 5.2a that the OD pairs that increase at most their flow after the optimization procedure are those that in the historical OD matrix were very low. Moreover, the OD pairs that increase their flow become the greatest, once the algorithm is finished, as seen in Figure 5.2b.
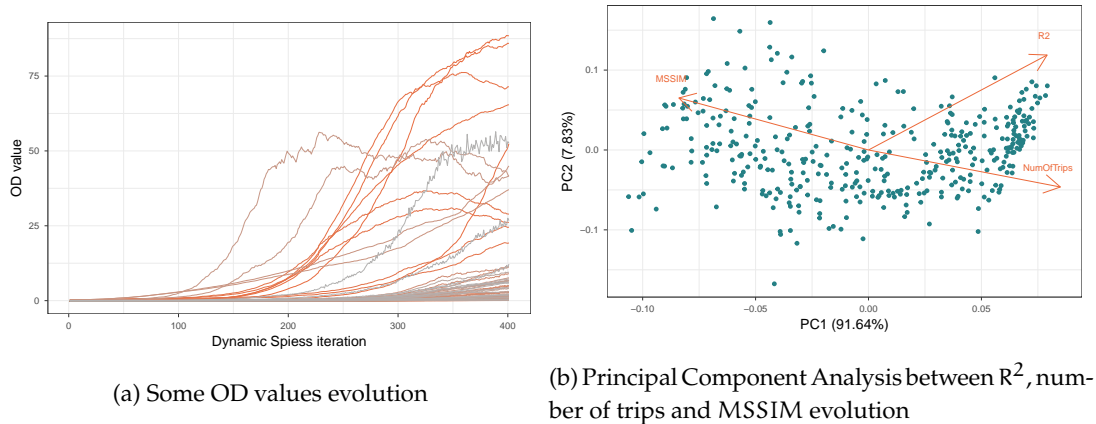


(a) Some OD values evolution

(b) Principal Component Analysis between $R^2$, number of trips and MSSIM evolution

Figure 5.3: Analysis of $R^2$, number of trips and MSSIM evolution

In order to show better the phenomenon, we plotted in Figure 5.3a the OD flows evolution iteration by iteration of those that increase their value with a growth factor greater than 100. Note that, again, these OD pairs were the ones that had low values in the seed OD matrix. In Figure 5.3b we show a principal components analysis between the $R^2$, the number of trips and the MSSIM at each iteration. In this case, we can see that the number of trips and $R^2$ are positively correlated but MSSIM is inversely correlated as detected.

Moreover, coming back to Figure 5.1, the descent structural similarity indicator shown can be understood after the analysis of Figures 5.2 and 5.3. The seed OD matrix has a high structural similarity with the ground truth OD matrix, and once the Dynamic Spiess method increase the OD flows for those that were very small, it is completely modifying its structure and therefore, affecting negatively the MSSIM with respect to the ground truth, both in the term L of the SSIM that describes the similarity of magnitudes and also in the case of S (Equation 3.2), because it is affecting clearly to the correlation between rows. However, on the early iterations, when the objective function decreases at most, the MSSIM value increases significantly, before starting to decrease on a second phase. That confirms that the Dynamic

Spiess procedure acts as a mere regressor of the traffic counts and pulls up the OD matrices to fulfill these traffic counts, without considering the demand pattern nor controlling the total number of trips.

In all cases, the addition of the second term to the objective function, that is $w_2 = 1$, helps to behave better, in terms of both number of trips and similarity to the ground truth OD matrix.

### 5.3.2    Dynamic Spiess reassigning at convergence results

In Figure 5.4, we compare the Original Dynamic Spiess with the Dynamic Spiess reassigning at convergence. The last one is much faster because it only launches a dynamic traffic assignment once a threshold of convergence is reached. In the figure, we show the experiments for the Multitude-based initial OD matrix, which is the most realistic one.



(a) Objective Function                                                  (b) $R^2$

(c) Number of trips                                                      (d) MSSIM

——— w2=0    ······ w2=1          ——— Dynamic Spiess    ——— Dynamic Spiess on convergence
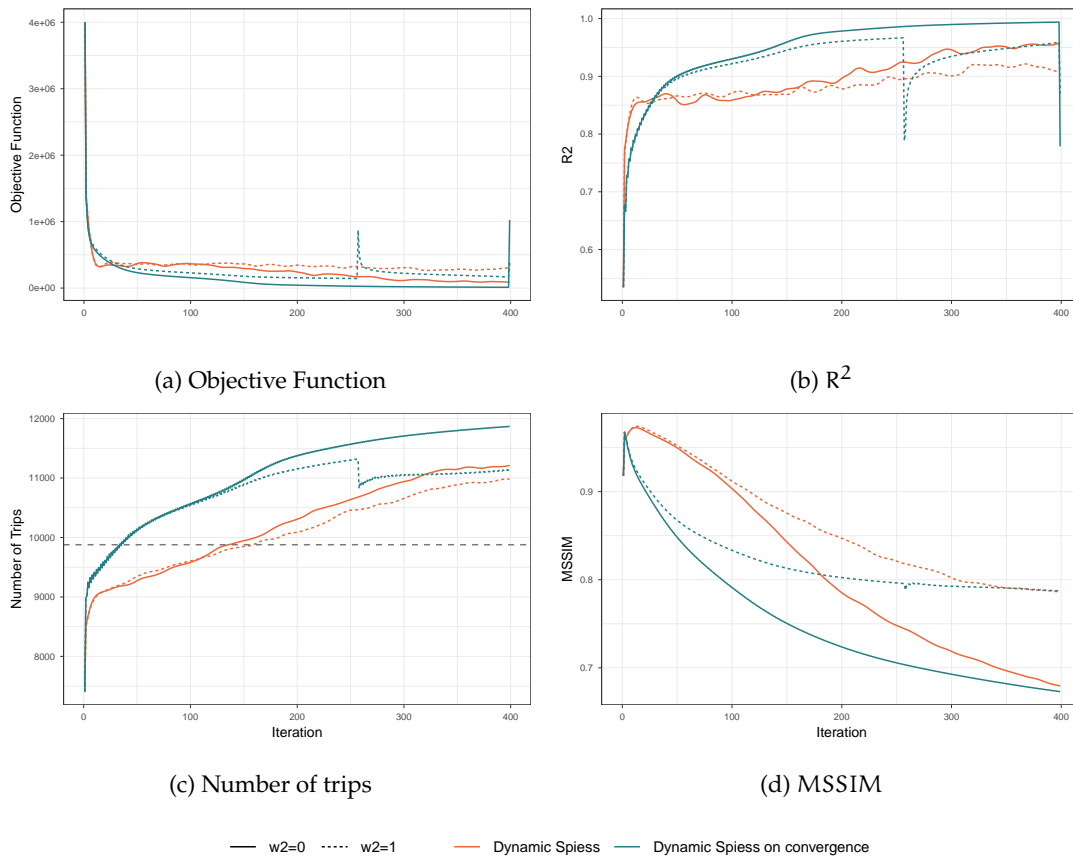
Figure 5.4: Results for Dynamic Spiess reassigning always or only on convergence, for the same initial OD matrix, the Multitude approach

From the graphs shown, we want to highlight that in the case when $w_2 = 0$, there is no reassignment in 400 minor iterations, meaning that the algorithm does not converge according to the threshold. The previous results with Dynamic Spiess algorithm in Section 5.3.1 showed the behavior in the later iterations, where there is a continuous degradation of the structure of the resulting matrix. In the case of Dynamic Spiess reassigning only on the convergence, this degradation occurs immediately, as shown in Figure 5.4d. Therefore, it is not a suitable alternative when reducing the computational time is desired. Moreover, the total number of trips increases more than when launching the DTA at each iteration.

### 5.3.3   Original Dynamic Spiess with entropy function

In Figure 5.5, we represent the same four indicators for 6 different experiments. These experiments correspond to the three decremented initial matrices (Multitude, Incremental- and Chaos+Inc-) with the different second term, whether the Euclidean distance or the entropy function.
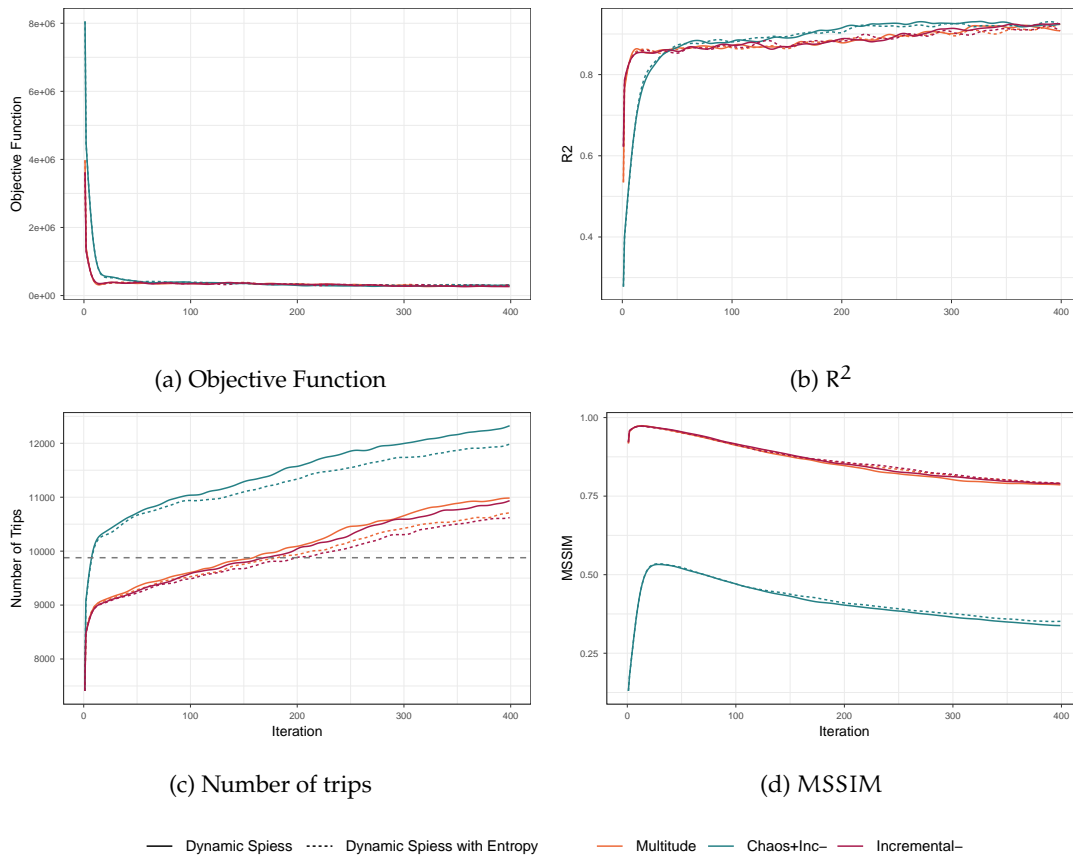


(a) Objective Function                                                         (b) $R^2$

(c) Number of trips                                                                (d) MSSIM

Dynamic Spiess ····· Dynamic Spiess with Entropy          Multitude      Chaos+Inc–      Incremental–

Figure 5.5: Results for Dynamic Spiess with the second term, using Euclidean distance and entropy and three different decremented initial OD matrices

The results obtained with the entropy function are similar to those using the Euclidean distance and they present no evidence of better performance. The `Chaos+Inc-` initialization shows an excellent $R^2$ and convergence properties, by adding many trips, but being unable to discover the ground truth demand pattern (as indicated in MSSIM plot). Either Euclidean, or entropy distance produce similar results, with slightly better results when entropy distance is set.

### 5.3.4   Comparison between Dynamic Spiess and SPSA procedures (without travel times)

As we used the same network and the same initial OD matrices, we can compare the KPIs of SPSA, from Chapter 4, and Dynamic Spiess approaches. In Figure 5.6, we show the evolution of the same four KPIs for the different approaches. In this set, we only present the results with the Multitude-based historical OD matrix. In the case of SPSA algorithms, only the experiments without the conjugate

gradient variant are shown.  We show here the experiments referred to the different variants of SPSA (Free, Constrained, Penalized, Hybrid) without travel times.  As a reminder, all the set of experiments are plotted in Appendices B.1 and B.3.



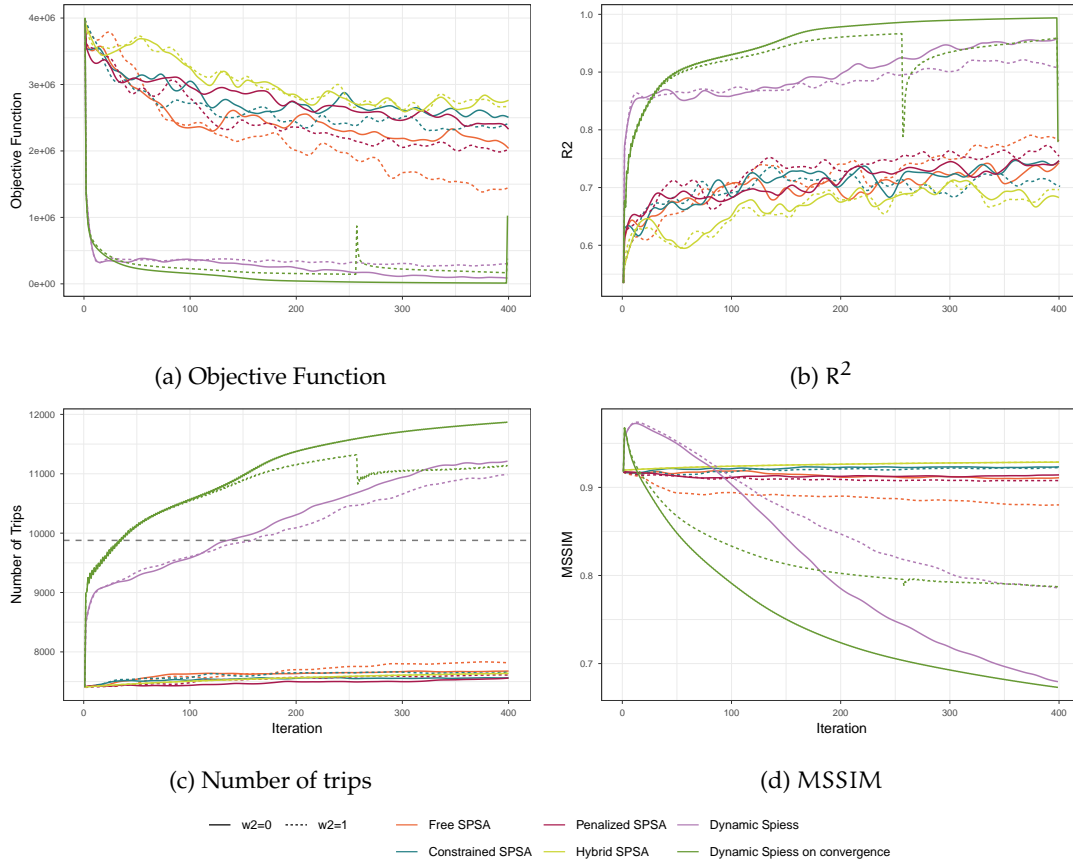(a) Objective Function

(b) $R^2$

(c) Number of trips

(d) MSSIM

Figure 5.6:  Results for Dynamic Spiess and SPSA approaches using the Multitude-based initial OD matrix and $w_2 = 1$

The Dynamic Spiess procedures outperform clearly the SPSA approaches in terms of objective function descent and convergence which makes sense because the SPSA approaches are stochastic optimization approaches and therefore, use an estimation of the gradient descent direction.  In terms of $R^2$, the Dynamic Spiess approaches do also reach greater values than SPSA. However, Figures 5.6c and 5.6d show that, as mentioned, the good performance on the two first indicators for Dynamic Spiess approaches is at the cost of the total number of trips and MSSIM on the latter iterations.  The Hybrid SPSA without travel times, which is indeed an iterative procedure with analytical gradient shows a very slow increase of the MSSIM indicator, because its step size is smaller than the one in the dynamic Spiess approach.

## 5.4   Conclusions for the comparison between Dynamic Spiess and SPSA

In previous Section 5.3.4, we have compared the four KPIs evolution of the different proposed methodologies of Chapter 4 and 5, that are the SPSA variants and the Dynamic Spiess.  Both methodologies are

interesting and present nice properties to be used and they have a common stochastic component as far as both use DTA in the lower level, but the long convergence of SPSA variant may be due to the additional intrinsic stochastic nature of SPSA when calculating the perturbations for the stochastic gradient. On the other hand, Dynamic Spiess method is faster and improves significantly on the early iterations, which make it very usable.

However, Dynamic Spiess approaches present a significant anomalous increase of the total number of trips that affects negatively to the structure of the OD matrix, while SPSA shows stability regarding the number of trips and the $MSSIM$ as indicator of structural similarity. Moreover, the SPSA algorithmics permits to add bounding constraints that adds more stability and present better results, while Dynamic Spiess methodology can not add constraints.

As a summary, from a practical point of view and effectivity, the Dynamic Spiess method is an efficient method that properly calibrates the OD flows for a network, also in the cases where the starting point is a bad estimation of the OD matrix. However, longer iterations of the method carry to undesirable effects on the structure of the matrix. On the other hand, the slowness of SPSA makes it infeasible to be used in practice for large networks, even though its versatility and ease at the moment of adding new traffic measurements or bounding constraints.

## 5.5   Conclusions of Analytical Approaches

It is very well known that analytical approaches, such as the Spiess (1990) algorithm, are suitable for the static OD matrix estimation problem. Moreover, these approaches present nice convergence, stability and robustness characteristics, so they are widely used to solve the bi-level optimization problem.

As mentioned, the step to the dynamic traffic assignment models aims to include the complexity of congestion effects propagation and makes the OD estimation problem more complex, because the OD flows changes on a certain time period do affect the same but also posterior traffic counts. In this sense, Frederix et al. (2011, 2013) made a significant theoretical study about these effects on the assignment matrix and also on the underdetermination of the bi-level estimation problem. Moreover, these approaches, based on finding relationships between the variables (the OD flows) and the traffic measurements do not fulfill the objective of this thesis, of adding different traffic measurements to the OD estimation problem.

However, we studied in depth these analytical approaches on the static version and proposed an extrapolation of the Spiess method to the dynamic version of the bi-level approach, following the aim of understanding better the dynamics of these approaches and what are the key aspects in these cases.

After using the called Dynamic Spiess on a synthetic case study, with the city of Hillsboro, we have understood how this approach works and what are its advantages and inconvenients. On one hand, its main advantage is the ease of implementation and fast convergence. In a few iterations, the proposed methodology obtains a great descent of the objective function, reaching a very nice fitting between the simulated and observed traffic counts, no matter how bad the initial OD matrix is. However, the impossibility of adding bounding constraints, as we proposed in Chapter 4 for SPSA, shows an important limitation of the methodology. The Dynamic Spiess works as a perfect fitter between simulated and observed counts but it presents an uncontrolled increase of the number of trips that affects completely the structure of the resulting OD matrix, that is the demand pattern of the network. Therefore, in a real case, it is very important to properly set a threshold that stops the procedure at the early iterations in order to avoid the uncontrolled second phase that is presented in all the studies performed.

Furthermore, in general aspects, all the proposed Dynamic Spiess variants that aimed to find alternatives that improve the results obtained with the classic bi-level problem do not present significant improvements regarding the quality of the estimated OD matrix.

The main information for the objectives of this thesis is that brought into the mathematical model by the dynamic assignment matrix, which is a key component of the OD estimation process, because it is the way that we can relate the variables of the problem that we want to estimate with the observed data of the network, that are the traffic counts through certain links of the network. Furthermore, the assignment matrix appears in all the steps of the algorithm, that are the gradient calculation and iterative procedure. The objective of adding analytically new ICT traffic measurements directly as a new term of the objective functions is not possible because there is not an analytical relationship between them and the estimated OD matrices. However, the GPS data is a collection of vehicles circulating through the network and contains information about the traffic dynamics and congestion at each studied time period. A deeper analysis of the structure of the assignment matrix, in terms of the use of a link by OD paths raises the question of whether empirical evidence of that use could be exploited to estimate the assignment matrix. Therefore, GPS traces could be an interesting data source to generate such information that would have the additional interest of corresponding to the current state of the network at the time period considered. Consequently, this is the next step of our research.

# 6

# A Data-Driven Assignment-Free DODME methodology

*In this chapter, based on the current information of the traffic state from the GPS Data, we estimate a dynamic assignment matrix without executing a DTA and it could be considered a data-driven assignment-free method. This procedure is validated and tested on a synthetic network and also with two real networks of Turin and Barcelona. At the end, we propose some stopping criteria based on the structural similarity.*

The dynamic assignment OD has been proven to be the key element in analytical approaches and ICT data is a new source that opens the possibility of empirically estimating this matrix and developing assignment free approaches for DODME that is the aim of this chapter. Among the most efficient approaches to solving the OD estimation problem, the one that formulates the problem in terms of a bi-level optimization problem has been widely used. This formulation solves at the upper level a nonlinear optimization problem that minimizes some distance measures between observed and estimated link flow counts at certain counting stations located in a subset of links in the network, and at the lower level a traffic assignment that estimates these link flow counts assigning the current estimated matrix. The variants of this formulation differ in the analytical approaches that estimate the link flows in terms of the traffic assignment and their time dependencies. Since these estimations are based on a traffic assignment at the lower level, these analytical approaches, although numerically efficient, imply a high computational cost. The advent of ICT applications has made available new sets of traffic related measurements enabling new approaches; under certain conditions, the data collected allows to estimate the most likely used paths, from which a de facto assignment matrix can be computed. This allows extracting empirically similar information to that provided by the dynamic traffic assignment that is used in the analytical approaches.

## 6.1   Setting the foundations: a review of the previous approaches

The OD estimation problem in terms of the bi-level optimization problem is shown in Equation 1.6, aimed at adjusting an initial target OD matrix, $\mathbf{X}^{\mathrm{H}}$, so that it could explain the observed link flow counts $\mathbf{Y}$ at counting stations in the network. The underlying hypothesis is that $\mathbf{Y}(\mathbf{X})$ are the link flows predicted by assigning the demand matrix $\mathbf{X}$ onto the network. This mathematical model is highly undetermined

since the number of variables of the problem, the OD flows, is much larger than the number of the available link traffic counts. Therefore, the resolution of the optimization problem can lead to different solutions, even in the case when the seed OD matrix is proper to the solution, Yang et al. (1992). As already mentioned in Chapter 3, it is well known that even a full covered network with traffic sensors on each link does not ensure a determined problem, Bierlaire (2002), therefore, an appealing research topic has been to explore new approaches including further information, such as link speeds or travel times (Cantelmo, Viti, Tampère, Cipriani & Nigro (2014), Nigro et al. (2018), Kostic et al. (2015), aimed at reducing such underdetermination.

Traffic modeling for transportation analysis has evolved to the dynamic traffic assignment models, which are able to include the time dependencies on the traffic system, overcoming in this way the main drawbacks of static assignment of not accounting for the congestion generation and its dynamic propagation across the network. Dynamic models require then dynamic inputs, which means dynamic OD matrices that are represented as a time series of sequentially OD matrices. Therefore, the dynamic OD matrix estimation problem (DODME) becomes more complex, with more variables and time dependencies across the time periods of the traffic simulation, Frederix et al. (2010, 2011).

Cantelmo (2018) have considered the DODME problem based on the bi-level approach and include utility-based DTA models in the lower level relying on activity location and trip duration information. They demonstrate that, extending the bi-level approach by taking into account such information, the number of free parameters in the DODME problem systematically decreases, improving the reliability of the estimated dynamic OD matrices by reducing the underdetermination of the solution.

Recent literature is addressed to include ICT measures into the DODME problem to reduce the underdetermination of the underlying problem. Mo et al. (2020) propose a two-step ordinary least squares (OLS) OD estimation model, which incorporates the output from a Bayesian path reconstruction model developed to cope with insufficient coverage rate of ICT data from Licence Plate Recognition and estimates both the dynamic OD demand and assignment matrix without any historical matrix need. Finally, Yang et al. (2017) and Krishnakumari et al. (2020) use the geopositioning data of probe vehicles from an ad hoc experiment designed by the authors to obtain an a priori dynamic OD matrix and the reconstructed paths are included into the OD estimation process.

As explained in Chapter 5, the linearization of the relationship between traffic counts and OD flows is one way of solving these problems in a more computationally efficient way. This can be achieved by using the proportion of the OD demand flows passing through the count location at a certain link. In these terms, the dynamic assignment matrix $\mathbf{A}(\mathbf{X}) = \left[ a_{ijr}^{lt} \right]$ is the result of the mapping and $a_{ijr}^{lt}$ represents the proportion of the OD flow that departs from origin $i$ at time period $r$ and goes to destination $j$ that crosses link $l \in \hat{L} \subseteq L$ at time period $t \geq r$.

These analytical approaches to DODME problem show that all of them rely on the availability of the assignment matrix $\mathbf{A} = \left[ a_{ijr}^{lt} \right]$ for the various time intervals, calculated at the lower level of the bi-level problem (Equation 5.2) by the dynamic traffic assignment at each time interval.

The availability of the GPS generated data enables us to assume that, after a suitable data processing to find the empirical paths and the inference of path choice proportions, it is possible to estimate a dynamic assignment matrix that relies on the information regarding traffic conditions. Since it would play a similar role to that of the analytical assignment matrix obtained by a DTA, we focus our attention on how to efficiently estimate that assignment matrix, in practical terms, from available commercial data as discussed above.
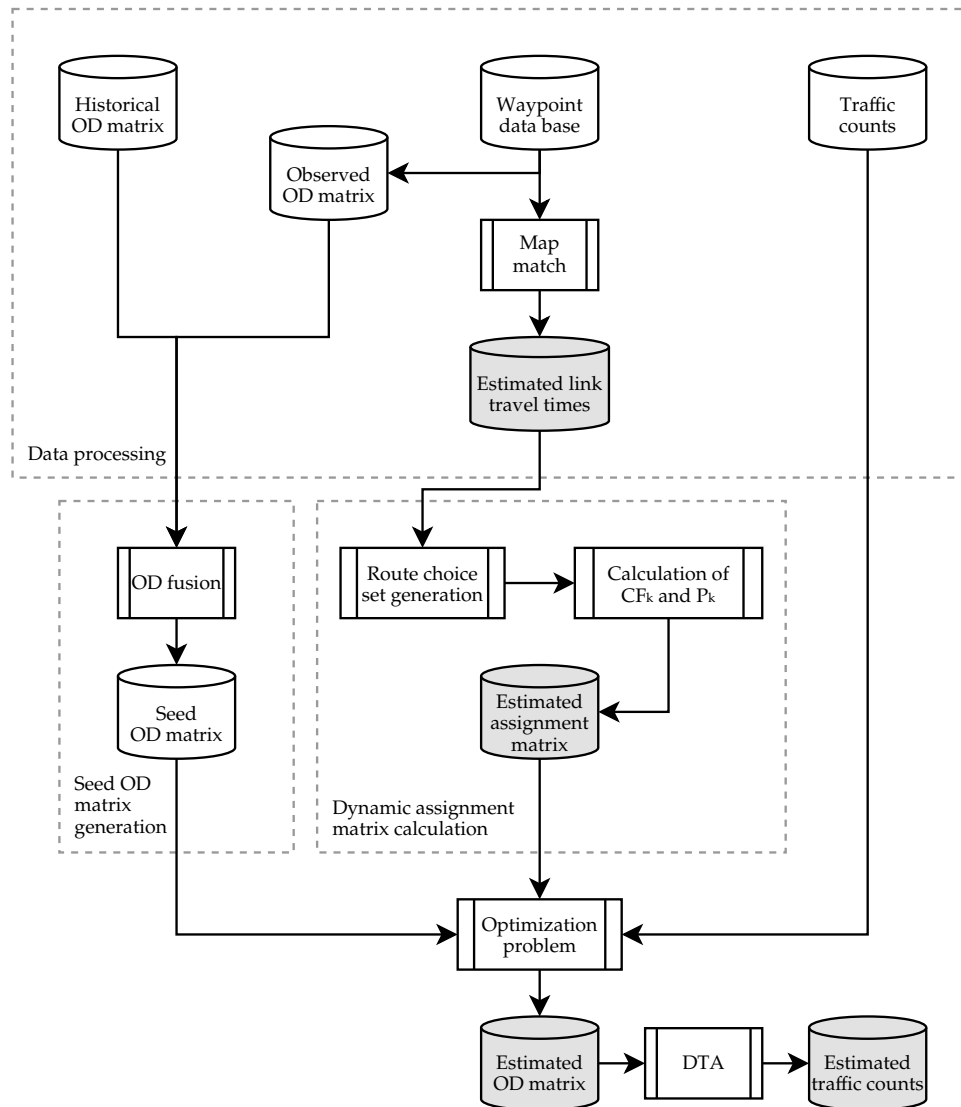
Figure 6.1: The Data-driven Assignment-free DODME methodology

## 6.2   A Data-Driven Assignment-Free DODME

In Section 2.3, we discuss the use of empirical data for the intended purpose. As mentioned, a key aspect is whether one can control the data collection process or, on the contrary, one depends on the commercial GPS traces as supplied by data providers. The first situation ensures the quality and reliability of the data collected and also its adequacy to make the necessary estimations in terms of complete reliable trajectories from origins to destinations (Yang et al. (2017) and Krishnakumari et al. (2020)).

In the second case, data are usually of two types, either non-processed waypoints or in-house processed information as, for instance, speed profiles. Non-processed waypoints are not directly usable for transportation analysis, and they must be processed before: they must be filtered, cleansed to remove outliers and correct errors, and suitably map matched to fit the transport network. Alternatively, data supplier companies also provide references to tools to extract additional information like speed profiles at link level from the waypoints, which can be used for transportation analysis to infer OD travel times among other applications. An example of such a tool would be OpenLR, OpenLR (2020).

In this thesis, we want to work with available reliable data, coming from GPS traces and transformed to link travel times, either using the proposed map-matching methodology of Section 2.4, obtaining it directly from the provider or generated by the analyst using other available tools. In all cases, this paper assumes that the available data have already been filtered, cleansed and processed, and therefore, we focus the work on what can be done with the available link travel times estimations.

Therefore, assuming that a set of estimated link travel times and a set of traffic counts are available for a selected period of time; a specific purpose designed process produces route choice paths and proportions for generating an estimated dynamic assignment matrix, $\overline{\mathbf{A}}$. Then, the research question addressed in the following sections is to investigate whether it is possible to use such information to state a different formulation of the DODME problem, in terms of an optimization model, not requiring the execution of any dynamic traffic assignment procedure. The conceptual computational scheme of the proposed data-driven assignment-free DODME approach, powered by the ICT applications capturing GPS data trajectories and providing estimated link travel times, is summarized in Figure 6.1.

### 6.2.1   Calculation of the dynamic assignment matrix

According to assignment-based methods, the paths used to travel between origins and destinations are provided by a user equilibrium assignment. In an assignment-free approach, we propose an alternative method relying on the available estimated link travel times to generate a plausible Route Choice Set, $\mathcal{P} = \{P_{ijr}, \forall i \in I, \forall j \in J, \forall r \in \mathcal{T}\}$, specifically from among the most likely used paths between each origin and each destination at each departure time.

The estimated link travel times for each link of the network $l$ at each time period $t$, $\hat{t}t_{lt}$ are the main inputs for generating the route choice set. That is the set of most likely alternatives. This is usually done through a selective approach that identifies the routes based on some previously mentioned criteria ($k$ shortest paths, path flows computation, etc.).

Many alternative approaches can be used for this, and all them are essentially based on variants of $k$ shortest paths. Alternatives based on iteratively applying Dijkstra-based algorithms for similar purposes while explicitly accounting for overlapping penalties have been analyzed by Janmyr & Wadell (2018) and Nassir et al. (2014). Other alternative procedures based on Chabini (1998) time dependent shortest paths or path search algorithms can be found in the literature. In this thesis, as we are in collaboration with PTV Group, we use a specific tool in PTV Visum, which is a path search algorithm available in Visum (PTV AG (2020), 6.18) that calculates specified sets of $k$ shortest paths by perturbing the link's impedances with a normal distribution perturbation. The parameter $k$ must be stated differently according to the network characteristics in order to generate sufficient number of paths but limiting the number of irrelevant paths.

Once the candidate routes in the route choice set are specified, a key question in the route choice model is how to address the problem that the alternatives are usually not independent but correlated due to overlapping paths. From a theoretical point of view, Probit models are likely those who better account for these correlations, but the difficulties in practically implementing them led to search for other approaches. Cascetta et al. (1996) and Ben-Akiva & Bierlaire (1999) propose alternative models for capturing the correlation among alternatives by modifying the logit-based choice, specifically by measuring the degree of similarity between the alternatives and adding it to the utility's deterministic component in the corresponding discrete choice model. This term is usually called the *commonality factor*, and its main role, Cascetta (2001), is to overcome the problems deriving from the basic hypothesis of independence of irrelevant alternatives, which discrete choice logit models assume and could otherwise lead to unrealistic results. This term reduces the systematic utility of a path in proportion to its level

of overlapping with other alternative paths. The formulation adopted in this thesis is Janmyr & Wadell (2018)'s proposed modification of the formulation of Bovy et al. (2008).

The paths in $\mathcal{P}_{ijr}$ are $p = p(i,j,r) \in \mathcal{P}_{ijr}$ which explicitly show the dependence on $(i,j,r)$. For simplicity, we denote only $p$ but always refer to a path of a certain OD pair, $(i,j,r)$. For a certain path $p$, the sequence of links that compound it is the set $\Gamma_p = \{e_1, \ldots, e_{m_p}\}$. Then, the proportion of path choice for each path, $P_p$, in the set $\mathcal{P}_{ijr}$ is calculated as a modified discrete logit-based choice model that uses the commonality factor within the OD pair and time period, $CF_p$. It further acts as an additive penalization factor on current travel times Bovy et al. (2008). The commonality factor penalization for a certain path $p \in \mathcal{P}_{ijr}$ is defined as follows:

$$CF_p = \frac{1}{\mu_{CF(\mathcal{P})}} \sum_{a \in \Gamma_p} \left( \frac{l_a}{L_p} \log \left( \sum_{h \in \mathcal{P}_{ijr}} (\delta_{ahr} + 1) \right) \right) \tag{6.1}$$

where $\delta_{ahr}$ indicates whether path $h \in \mathcal{P}_{ijr}$ uses link $a$; $l_a$ is the length of link $a \in \Gamma_p$; and $L_p$ is the total length of path $p \in \mathcal{P}_{ijr}$. $\mu_{CF(\mathcal{P})}$ is a parameter that depends on the set $\mathcal{P}_{ijr}$ and normalizes the effect of the penalization.

Once the penalization term is calculated for each path of the set of $k$ shortest paths, $\mathcal{P}_{ijr}$, the proportions of flow assigned to each path are calculated:

$$P_p = \frac{\exp[\mu_{P(\mathcal{P})}(-\overline{tt}_p - CF_p)]}{\displaystyle\sum_{h \in \mathcal{P}_{ijr}} \exp[\mu_{P(\mathcal{P})}(-\overline{tt}_h - CF_h)]} \tag{6.2}$$

where $\overline{tt}_p$ is the estimated travel time for path $p \in \mathcal{P}_{ijr}$, that can be obtained either estimating it from a GPS data set using the methodology in Section 2.4, or obtaining it from the commercial provider, and summating for all the links that conform the path, that is:

$$\overline{tt}_p = \sum_{a \in \Gamma_p} \overline{tt}_a \tag{6.3}$$

Note that, the sum of the proportions for all the paths of a set $\mathcal{P}_{ijr}$ is, naturally, up to 1. In order to adapt magnitudes for the discrete choice summation, $\mu_{CF(\mathcal{P})}$ and $\mu_{P(\mathcal{P})}$ are parameters that are fixed as follows:

$$\mu_{CF(\mathcal{P})} = \mu_{P(\mathcal{P})} = \left( \frac{1}{|\mathcal{P}_{ijr}|} \sum_{p \in \mathcal{P}_{ijr}} \overline{tt}_p \right)^{-1} \tag{6.4}$$

which aims at helping to discriminate between paths when the average travel time of the OD pair is low.

These calculations obtain the flow distribution for each path on the basis of observed path travel times, that as long as they are estimated from the actual traffic conditions from GPS data, are proportions based on the given traffic state. Once $\mathbf{P}_p = [P_p]$ is determined, we can then calculate the estimated

time-dependent assignment matrix $\bar{\mathbf{A}} = \left[ \bar{a}_{ijr}^{lt} \right]$, that must be calculated for all the OD pairs and for all the equipped sensors at each time period:

$$\bar{a}_{ijr}^{lt} = \sum_{p \in \mathscr{P}_{ijr}} \delta_p^{lt} P_p \ , \quad \forall i, j, r, l, t \tag{6.5}$$

where $\delta_p^{lt}$ is the estimated incidence indicator:

$$\delta_p^{lt} = \begin{cases} 1 & \text{if path } p \text{ uses link } l \text{ at time } t \\ 0 & \text{otherwise} \end{cases} \tag{6.6}$$

This estimated assignment matrix is the outcome of the Dynamic Assignment Matrix Calculation box included in Figure 6.1 and constitutes an estimate of the dynamic assignment matrix that would be obtained by a DTA assignment based on the ground truth matrix. As a consequence, the estimated assignment matrix derived by the proposed process does not depend on a perfect calibrated model allowing to split the calibration process into the network supply calibration and the demand calibration stages.

## 6.2.2   Optimization Procedure

The possibility of estimating an assignment matrix, Equation 6.5, allows reformulating DODME by relating the estimated traffic counts with the OD flows using the estimated assignment matrix:

$$\bar{y}_{lt} = \sum_{(i,j) \in N} \sum_{r=1}^{t} \bar{a}_{ijr}^{lt} x_{ijr} \tag{6.7}$$

where $\bar{y}_{lt}$ is the estimated flow in link $l$ at time period $t$; $x_{ijr}$ is the flow departing origin $i \in I$, with destination $j \in J$, at time interval $r \in \mathcal{T}$; and $\bar{a}_{ijr}^{lt}$ is the estimated assignment matrix, which is the fraction of trips from origin $i$ with destination $j$, departing at time $r$ reaching link $l$ at time $t$ estimated by the use of the GPS traces.

In Equation 6.7, the estimated assignment matrix is applied to the OD matrix to obtain the estimated link flows. In the bi-level optimization problem, the OD flows are always the variables that must be adjusted to minimize the objective function. However, from the experience with analytical approaches in Chapter 5, we observed in our experiments that OD flows may be modified to fit traffic counts implying high volumes to certain OD pairs. We propose a data-driven approach aiming to preserve the OD pattern included in the seed OD matrix.

The proposal is inspired by gravity models, that set bi-dimensional constraints for rows and columns, as in the double-constrained models that are common for updating gravity distribution models, Ortúzar & Willumsen (2011). Therefore, the OD flow appearing in Equation 6.7 can be decoupled into independent scaling factors different for origins and destinations. That is:

$$x_{ijr} = \alpha_i \beta_j x_{ijr}^0 \ , \quad \forall i \in I, \forall j \in J, \forall r \in \mathcal{T} \tag{6.8}$$

where $x_{ijr}^0$ represents any reference OD matrix, that can be used as a starting point for the OD estimation, as in the Furness process to calculate the OD matrix using gravity models. The inclusion of a third scaling factor $\gamma_r$, depending on the sliding time windows, could make sense for larger time periods, when the time variability of the demand can be influenced by other structural aspects, but not in the short term investigated in this thesis.

As in all optimization methods that aim to find a solution, a seed OD matrix must be provided to the OD estimation process as a feasible starting point. In this described methodology, many alternatives arise. One common option is to use a reliable historical OD matrix, $\mathbf{X}^H$, as a suitable seed for the optimization algorithm (Cascetta et al. (2013), Kostic et al. (2015), Cantelmo, Cipriani, Gemma & Nigro (2014), Nigro et al. (2018)), specially, in those cases in which DODME is applied to not very long time periods to support dynamic traffic models in conditions where surveillance systems likely provide reliable historical OD estimates containing a wealth of structural information (Ashok & Ben-Akiva (1993), Ben-Akiva et al. (2001)). Moreover, in real life applications of these approaches for traffic management, the assumption of having a reliable reference OD matrix also holds, Djukic et al. (2018), Aimsun (2017). Another alternative, if data collected from a sample of GPS-tracked vehicles is available, is to create a discrete time estimate of the target OD matrix from it, which is the observed OD matrix, $\hat{\mathbf{X}} = [\hat{x}_{ijr}]$, which is a simple counting of the GPS traces. With the aim of adding both information sources, a seed matrix $x_{ijr}^0$ can be generated combining the Historical OD matrix $x_{ijr}^H$ with the observed OD matrix $\hat{x}_{ijr}$, which is obtained from GPS tracked trips. This possibility consists of generating a proper seed OD matrix as a combination of the two different sources.

$$
x_{ijr}^0 = \begin{cases}
\hat{x}_{ijr} & \text{when only } \hat{x}_{ijr} \text{ is available} \\
f(\hat{x}_{ijr}, x_{ijr}^H) & \text{when both matrices are available} \\
x_{ijr}^H & \text{when only } x_{ijr}^H \text{ is available}
\end{cases}
\tag{6.9}
$$

If $\hat{y}_{lt}$, $l \in \hat{L} \subseteq L$, $t \in \mathcal{T}$ are the link flows measured at the counting stations, the dynamic data-driven assignment-free OD matrix estimation problem can be formulated as an optimization problem for finding the values of the scaling factors $\alpha_i, i \in I$ and $\beta_j, j \in J$, without any need to conduct the traffic assignment at the lower level of Problem 1.6. This is done by exploiting the estimated assignment matrix $\bar{a}_{ijr}^{lt}$. The proposed new formulation of the DODME problem including the estimated assignment matrix and using a seed OD matrix of the form is:

$$
\min_{\alpha_i, \beta_j} \left[ \sum_{l \in \hat{L}} \sum_{t \in \mathcal{T}} \left( \hat{y}_{lt} - \sum_{(i,j) \in N} \sum_{r=1}^{t} \alpha_i \beta_j \bar{a}_{ijr}^{lt} x_{ijr}^0 \right)^2 + w_2 \left( \sum_{(i,j) \in N} \sum_{r=1}^{t} \left( x_{ijr}^H - \alpha_i \beta_j x_{ijr}^0 \right)^2 \right) \right]
\tag{6.10}
$$

$$
\text{s. to:} \quad \alpha_i, \beta_j \geq LB > 0
$$

Now, the problem variables are the multiplicative scaling factors for each origin $\alpha_i$ and destination $\beta_j$, that have been chosen to drastically reduce the number of variables from $|I| \cdot |J| \cdot |\mathcal{T}|$ to $|I| + |J|$. Moreover, using the scaling factors as variables aims to preserve the structure of the seed OD matrix, as gravity models do. The minimization problem is solved iteratively by means of the L-BFGS-B method, Morales & Nocedal (2011). It is a quasi-Newton method solved for constrained non-linear problems with a high

number of variables that efficiently reduces the memory requirements and the computational burden. The available version in python package *scipy.optimize* has been used in this case.

Theoretically, LB should be a non-negativity constraint for all the scaling factors $\alpha_i, \beta_j$. However, from a practical point of view, $\alpha_i = 0$ or $\beta_j = 0$ implies that a positive OD flow of the seed OD matrix from a certain origin or certain destination must be converted to 0. Therefore, considering that the seed OD matrix, in Equation 6.9, comes from reliable information on mobility, the scaling factors cannot be 0 and the lower bound should therefore be LB $> 0$.

By the end of the optimization procedure, an estimation of the OD matrix, $\mathbf{X}^* = [x^*_{ijr}]$, is obtained. Therefore, as shown in Figure 6.1, a dynamic traffic assignment is launched to obtain the corresponding simulated values for traffic counts, that are $\hat{\mathbf{Y}}^* = [y^*_{lt}]$.

Algorithm 6.1 summarizes the Data-Driven Assignment-Free DODME methodology.

---

**Algorithm 6.1:** Data-Driven Assignment-Free DODME

---

**Input:** A historical OD matrix, $\mathbf{X}^H$
**Input:** A data set of traffic counts, $\hat{\mathbf{Y}}$
**Input:** A data set of GPS traces, **GPS**
**Input:** A stopping criterion, `thrsh_stop`
**Input:** A maximum number of iterations, `IterMax`

// Estimate Link Travel Times from GPS;
**Calculate:** $\overline{\mathbf{TT}} = \mathbf{TT}(\mathbf{GPS})$ ;                                    // Algorithm 2.1

// Set Initial OD Matrix;
**Calculate:** $\hat{\mathbf{X}} \leftarrow \text{count}(\mathbf{GPS})$;
**Set:** $\mathbf{X}^0 \leftarrow f\left(\mathbf{X}^H, \hat{\mathbf{X}}\right)$;                                    // Equation 6.9

// Estimate the Dynamic Assignment Matrix;
**foreach** $(i, j, r)$ *in ODpairs* **do**

    **Compute:** $\mathcal{P}_{ijr}$;                                    // Set of k shortest paths
    **foreach** p *in* $\mathcal{P}_{ijr}$ **do**

        **Calculate:** $CF_p$;                                    // Equation 6.1
        **Calculate:** $P_p$;                                    // Equation 6.2

**Calculate:** $\overline{\mathbf{A}}$;                                    // Equation 6.5

// Optimization Procedure;
**Set:** $\{\alpha_i, \beta_j\}, \forall i, j$;
**Optimize:** $\left\{\alpha^*_i, \beta^*_j\right\} = \texttt{L-BFGS-B}\left(\texttt{init} = \{\alpha_i, \beta_j\}, \texttt{fun} = Z, \texttt{params} = \left(\mathbf{X}^0, \overline{\mathbf{A}}, \hat{\mathbf{Y}}\right)\right)$

**Set:** $x^*_{ijr} \leftarrow \alpha_i \beta_j x_{ijr}$ ;                                    // Equation 6.8
**Evaluate:** DTA $(\mathbf{X}^*)$ ;                                    // Final Assignment

**Output:** Return $\mathbf{X}^*$

---

## 6.3   Case Study: Results of the DDAF DODME procedure

In Section 2.3, we propose a synthetic data generation procedure to be applied to any network and synthetically generate the needed consistent data sources. In the case of this algorithmic methodology, these data sources are GPS traces, traffic counts and a historical OD matrix and it is used to evaluate the

robustness of the presented method and validate its effectiveness on a potential real case under certain goodness conditions of the available data.

The network used in this case study is the same as in Section 2.3.1, where we show the synthetic data generation of the different data sets. It should be highlighted that, since the synthetic generation allows to control the traffic conditions and data gathering, the final GPS data set is a data set without biases, filtered and cleansed to obtain reliable estimations of travel times at link level.

As a summary and reminder, the test network is shown in Figure 2.3 in Section 2.4.1 and its main characteristics are detailed in Table 6.1. The simulation time interval is from 07:30 AM to 08:30 AM sliced into 15-minutes time periods.

Table 6.1: Network and OD characteristics

| | |
|---|---|
| Time periods | 4 |
| Zones | 114 |
| Detectors | 40 |
| OD pairs X Time | ≈52k |
| Ground Truth Trips | 8300 |
| Ground Truth Positive OD | ≈41k (78.49%) |
| Average number of paths per OD | 7.27 |

As shown, the average number of alternative meaningful paths for each OD pair is approximately 7, which makes it a suitable network to study the assignment matrix calculation methodology. Further than the size of the network, what is relevant is its structure and the average number of meaningful routes between each OD pair.

By construction, the historical OD matrix is on average decremented by 25% with a random perturbation, see Section 2.4.1 for further details. This perturbation tries to emulate a realistic historical OD matrix from surveys and past projects that represent similar traffic conditions.

In the case of the estimated link travel times, they are calculated after generating the GPS traces of different vehicles of the network. In this case, we emulated a controlled data collection, which consists of recording the waypoints sequences of vehicles on different days but on similar traffic conditions. We collected data during 200 days in similar conditions (independent replications by microsimulation), equivalent to an annual average working days, and fixed different penetration rates, which represent the number of random vehicles (among the total) that are recorded each day. In this study, a uniform penetration rate has been used for every day and also for all the OD pairs. Depending on the selected penetration rate, 5, 10 or 15%, these samples contain between 4.7M and 14M waypoints, which represent between 106k and 318k different vehicle trips circulating on the network. The frequency of recording waypoints has been assigned differently to each vehicle, following an empirical distribution of latencies from an INRIX real GPS data set of another network, see Section 2.4.1 for further details. Improvements would be expected as penetration rates increase, since a higher penetration rate provides better information about the traffic conditions.

As the thesis is a collaboration with the company PTV Group, we make use of already implemented tools. For example, in order to process the simulated GPS data, we used the tool *GPX Import*, which

transforms the GPS data set into paths using Visum links and interpolating travel times at the link level, as explained in Section 2.5. The estimated time-dependent link travel times are used to generate a route choice set (see Section 6.2.1) by using an independent tool, namely a path search algorithm available in Visum (PTV AG (2020), 6.18) that calculates specified sets of k shortest paths by perturbing the link's impedances with a normal distribution perturbation. The initial link costs are the estimated link travel times, and the maximum number of paths between connectors for each OD pair are set to $N_{max} = 5$, meaning that the tool calculates at least 1 but at maximum 5 paths between the connectors of the origin and destination zones.

As in all the synthetic experiments of this thesis, the optimization procedure in Equation 6.10 is set without the stopping criteria. Therefore, one can observe its behavior along all the iterations of the procedure.

### 6.3.1   Validation

In order to computationally test the consistency and quality of the algorithmic framework that is defined conceptually in Figure 6.1 and to analyze the quality of the partial results at each step, we have conducted a set of computational experiments based on the synthetic data generated by simulation. This allows further analysis of the quality of the methodology described above. Following Figure 6.1, all the different sub-results of the steps of the algorithm are analyzed.

**Observed OD matrix analysis**

The observed OD matrix is the mere counting of how many trips depart from each origin and arrive at each destination at each time period on the synthetic generated data. Based on the building process of the GPS tracking data, Section 2.4.1, it is expected to obtain an observed OD matrix similar in the OD pattern structure to the ground truth OD matrix, given that the penetration rate of the GPS technology has been set homogeneous to all the OD pairs.

As explained in Chapter 3, instead of using the conventional MSE or similar indicators to compare OD matrices the MSSIM and related metrics are used. The measure used to compare both OD matrices is the MSSIM and its components L, C and S. In order to use L and C, the observed OD matrix must be scaled in order to have the same magnitude for both matrices in terms of total OD trips.

Table 6.2: MSSIM values for the observed OD matrix

| Reference OD Matrix | penetration rate | L | C | S | MSSIM |
|---|---|---|---|---|---|
| $\mathbf{X}^{GT}$ | 5% | 0.9753 | 0.9595 | 0.9046 | 0.8654 |
| | 10% | 0.9856 | 0.9773 | 0.9357 | 0.9113 |
| | 15% | 0.9894 | 0.9844 | 0.9548 | 0.9355 |
| $\mathbf{X}^{H}$ | 5% | 0.9396 | 0.9059 | 0.9047 | 0.7884 |
| | 10% | 0.9472 | 0.9284 | 0.9351 | 0.8320 |
| | 15% | 0.9501 | 0.9375 | 0.9537 | 0.8550 |

Table 6.2 shows the MSSIM values for the observed OD matrix with respect to the ground truth and historical OD matrices. The L values, which correspond to magnitude, are very high because of the previously mentioned scaling; therefore, no further analysis has to be made. High C values indicate similar dispersion of the values, and high S means that the observed OD matrix has a similar pattern, thus indicating that the GPS sample is not biased and the penetration rate is homogeneous for origins and destinations, as expected. Globally, MSSIM values are high, and they increase as the penetration rate increases, meaning that the built sample of GPS data presents the appropriate goodness to be used to estimate the link travel times. Furthermore, these results are a proof to validate the use of the observed OD matrix as a seed OD matrix for the DODME method's optimization procedure.

**The effect of the Commonality Factor**

As mentioned in Section 6.2.1, the term $CF_p$ acts as a penalization for the discrete choice. $CF_p$, $P_p$ and $\bar{a}_{ijr}^{lt}$ are calculated by applying Equations 6.1, 6.2 and 6.5. The commonality factor, $CF_p$, penalizes those paths that are similar to others in the same set $\mathcal{P}_{ijr}$. Then, it reallocates the flows accordingly by increasing or decreasing the corresponding flow. In order to visualize the effect of $CF_p$ on the path flow distribution, a positive OD flow is selected. Its corresponding route choice set is depicted in Figure 6.2, where eight paths resulted from the route choice set calculations. Since there are three paths that are very similar, they were clustered by similarity into three sets (I, II, III) in order to better understand the role played by the commonality factor. Although link lengths are not large, network complexity in terms of number of OD paths per OD pair is high making the network suitable to address the consistency of the estimated assignment matrix appearing in the methodological proposal.



Figure 6.2: Path set ($\mathcal{P}_{ijr}$) for a selected origin and destination in the network

As shown in Figure 6.2, Table 6.3 and Table 6.4, set I is compounded by four very similar paths and

without the penalization (using only travel times) results in more than one-half of the flow being assigned to I. On the other hand, when the penalization is applied, 48.50% of the flow is assigned to these four paths. This is because the four paths are very similar and capture the majority of the flow from the beginning.

Table 6.3: Results of path distribution for a selected OD pair in the network

| Path | Length | Observed Time | $P_p$ without $CF_p$ | $P_p$ with $CF_p$ | % of gain | Set |
|------|--------|---------------|----------------------|-------------------|-----------|-----|
| 1 | 1.20 km | 2 min 11 s | 15.21% | 15.18% | -0.03% | I |
| 2 | 1.61 km | 2 min 57 s | 12.64% | 11.63% | -1.01% | I |
| 3 | 1.55 km | 3 min 13 s | 11.68% | 13.33% | +1.65% | II |
| 4 | 1.52 km | 2 min 59 s | 12.45% | 13.29% | +0.84% | III |
| 5 | 1.38 km | 3 min 02 s | 12.24% | 11.15% | -1.09% | I |
| 6 | 1.26 km | 3 min 06 s | 12.02% | 10.54% | -1.48% | I |
| 7 | 1.32 km | 3 min 18 s | 11.54% | 12.69% | +1.15% | II |
| 8 | 1.29 km | 3 min 03 s | 12.22% | 12.19% | -0.03% | III |

On the other hand, this flow is assigned to the other sets, which are different and do not share very much with the other sets in Table 6.4. For instance, set II receives more flow after the commonality factor penalization, because it is the one that shares less with the other sets.

Table 6.4: Results of path distribution by sets for a selected OD pair in the network

| Set | Mean Length | Mean Observed Time | $P_p$ without $CF_p$ | $P_p$ with $CF_p$ | % of gain |
|-----|-------------|--------------------|----------------------|-------------------|-----------|
| I | 1.36 km | 2 min 49 s | 52.11% | 48.50% | -3.61% |
| II | 1.43 km | 3 min 15 s | 23.22% | 26.02% | +2.80% |
| III | 1.41 km | 3 min 01 s | 24.67% | 25.48% | +0.81% |

**Qualitative analysis of the Estimated assignment matrix**

The estimated assignment matrix resulting from this methodology is used to capture the real mobility of the network, doing so by processing a large amount of GPS waypoints. While solving the optimization problem, in which an objective function minimizes the traffic flow differences detected by sensors, this assignment matrix remains invariant for each time interval, thus confirming the stability hypothesis formulated by Cascetta et al. (2013). Therefore, it is important to check the consistency between paths in the SBA (which is used to generate the reference data) and to identify the paths used to analyze the route choice set of paths (which are generated from the empirical data). Stated briefly, one should confirm that the theoretical assignment matrix from SBA is consistent with the estimated assignment matrix. Our qualitative proposal is to use the most relevant link in the network to compare OD flows.

(a) using SBA assignment                    (b) using estimated OD matrix for a selected experiment
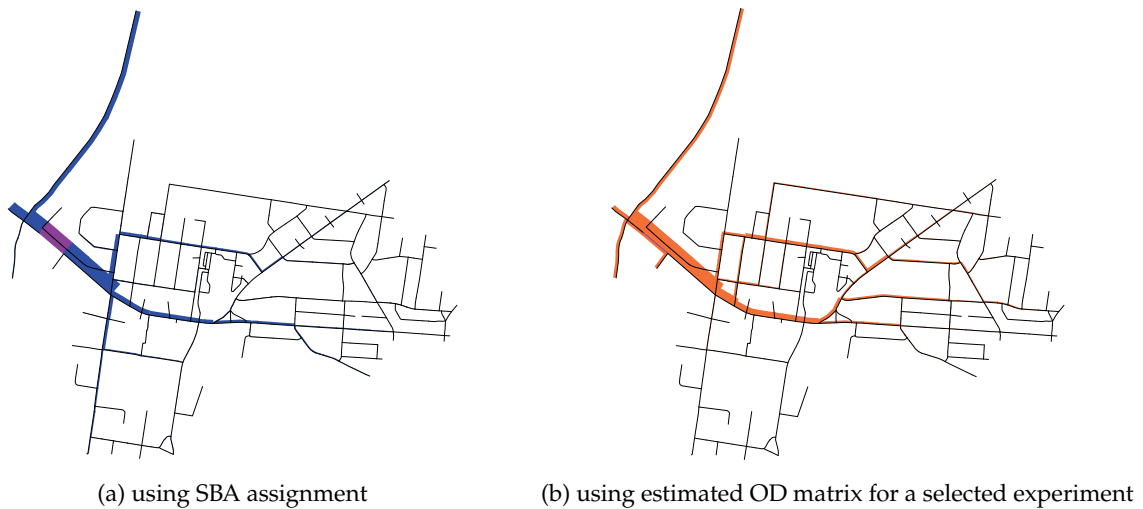
Figure 6.3: OD flows for the most used link

The left-hand side of Figure 6.3 shows the OD flows using a singular link (the most used link) in the network, according to the ground truth SBA assignment. On the right, the same picture is plotted by means of the estimated assignment matrix and the historical OD values. Both graphs are qualitatively similar, such that the downstream and upstream propagation of the flows circulating on this link indicate a similar assignment matrix.

**Optimization Procedure**

As seen in Figure 6.4, the convergence of the method is clear and fast.



(a) Objective Function                    (b) Relative Difference
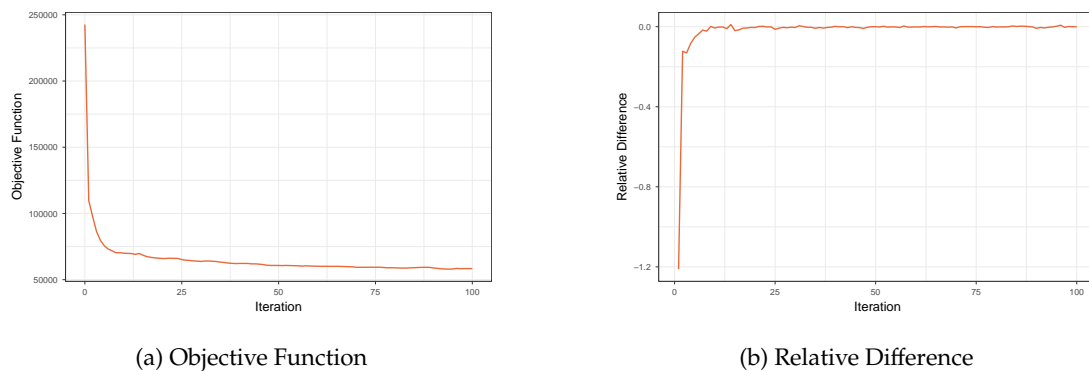
Figure 6.4: Optimization performance for a selected experiment

Figure 6.5 shows two versions of the linear regression between the traffic counts measured by sensors and their corresponding simulated values. On the left is the regression before optimization, when the estimated assignment matrix is already calculated but the OD values are not yet calibrated. On the right is the linear regression after computing the convergence of the minimization method and using Visum-SBA to incorporate the resulting assignment matrix from the assignment of the estimated OD matrix, $\boldsymbol{X}^*$.

(a) $\mathbf{Y} = \overline{\mathbf{A}} \cdot \mathbf{X}^0$          (b) After a DTA($\mathbf{X}^*$)
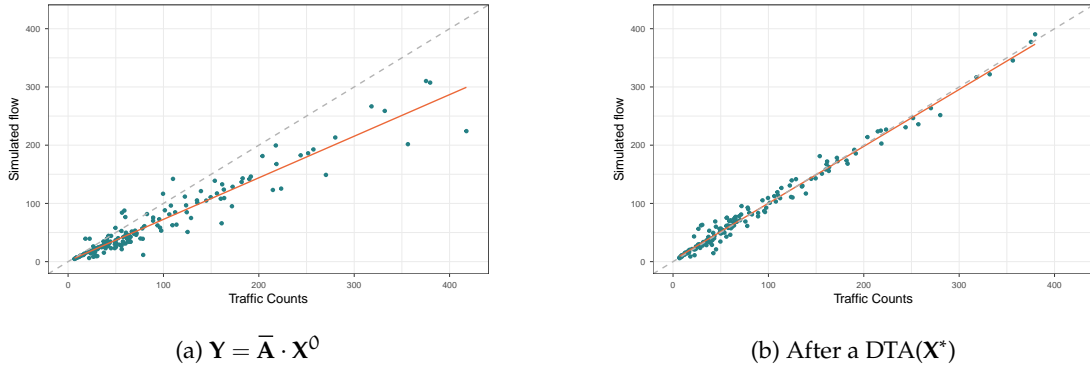
Figure 6.5: $R^2$ improvement in a selected experiment

The evolution shown in the figure validates the optimization method, which uses scaling factors instead of OD values as variables. The estimated scaling factors correctly adjust the flows in order to replicate the traffic counts measured in the network.

Finally, the use of scaling factors as optimization variables has also been studied. It has been done by comparing the estimated OD matrix with the one obtained by using the OD variables as variables on Equation 6.10. That is, firstly scaling the OD flows to the Historical magnitude and then using a maximum descent method to solve the optimization problem.

Concretely, the experiment performed used the historical OD matrix as seed, with a penetration rate of 10% and $w_2 = 0$. The results of convergence and fitting are high, presenting a final $R^2 = 0.974$. However, and as expected, the MSSIM values with respect the ground truth OD matrix is $MSSIM(\mathbf{X}^{GT}, \mathbf{X}^*) = 0.5479$, significantly lower than the ones obtained by using the scaling factors, $MSSIM(\mathbf{X}^{GT}, \mathbf{X}^*) = 0.7980$. Therefore, we consider that the use of scaling factors not only reduces the number of variables but also permits us to obtain a more stable solution in terms of similarity to the given seed.

### 6.3.2   Experimental Design

Finally, a set of experiments using the synthetic network and data generated is used to assess the robustness and the sensitivity of the methodology described with regard to some factors. This design factors are:

- The penetration rate of the GPS technology, as a percentage of vehicles that are captured by the GPS sample: 5%, 10% and 15%.

- The initial OD matrix for the minimization procedure $\left( \mathbf{X}^0 = [x^0_{ijr}] \right)$. As stated in Equation 6.9, the seed OD matrix can be the historical ($\mathbf{X}^0 = \mathbf{X}^H$), the observed ($\mathbf{X}^0 = \hat{\mathbf{X}}$), or both OD matrices combined. In this case, the combination tries to fill in the empty cells of the observed OD matrix with information from a reliable historical OD matrix. The two tested combinations are the following:

$$x^0_{ijr} = f_1(\hat{x}_{ijr}, x^H_{ijr}) = \begin{cases} k \cdot \hat{x}_{ijr} & \text{when } \hat{x}_{ijr} > 0 \\ x^H_{ijr} & \text{otherwise} \end{cases} \tag{6.11}$$

$$x^0_{ijr} = f_2(\hat{x}_{ijr}, x^H_{ijr}) = \begin{cases} \hat{x}_{ijr} & \text{when } \hat{x}_{ijr} > 0 \\ x^H_{ijr}/k & \text{otherwise} \end{cases} \tag{6.12}$$

where k is a factor that increases or reduces the number of trips in the seed OD matrix in order to approximate both magnitudes:

$$k = \frac{NT(\mathbf{X}^{H}(\hat{\mathbf{X}} > 0))}{NT(\hat{\mathbf{X}}(\hat{\mathbf{X}} > 0))} = \frac{NT(\mathbf{X}^{H}(\hat{\mathbf{X}} > 0))}{NT(\hat{\mathbf{X}})} \tag{6.13}$$

These four matrices are named, respectively, *Hist, Obs, Comb1* and *Comb2*.

- The objective function for the minimization procedure. By using $w_2 = 0$ or $w_2 = 1$ in Equation 6.10, the objective function may or may not include the discrepancy term regarding the historical OD matrix.

These design factors result with 24 different experiments. Some results are shown in the next Section, and they are fully detailed in Appendix B.4.

### 6.3.3 Results

Similarly to the other synthetic experiments of this thesis, the KPIs used to analyze the experiments are the objective function, the $R^2$, the total number of trips and the MSSIM. Figure 6.6 show the results for the three different penetration rates and $w_2 = 0$ and $w_2 = 1$, resulting in six different experiments:



(a) Objective Function

(b) $R^2$

(c) Number of Trips

(d) MSSIM
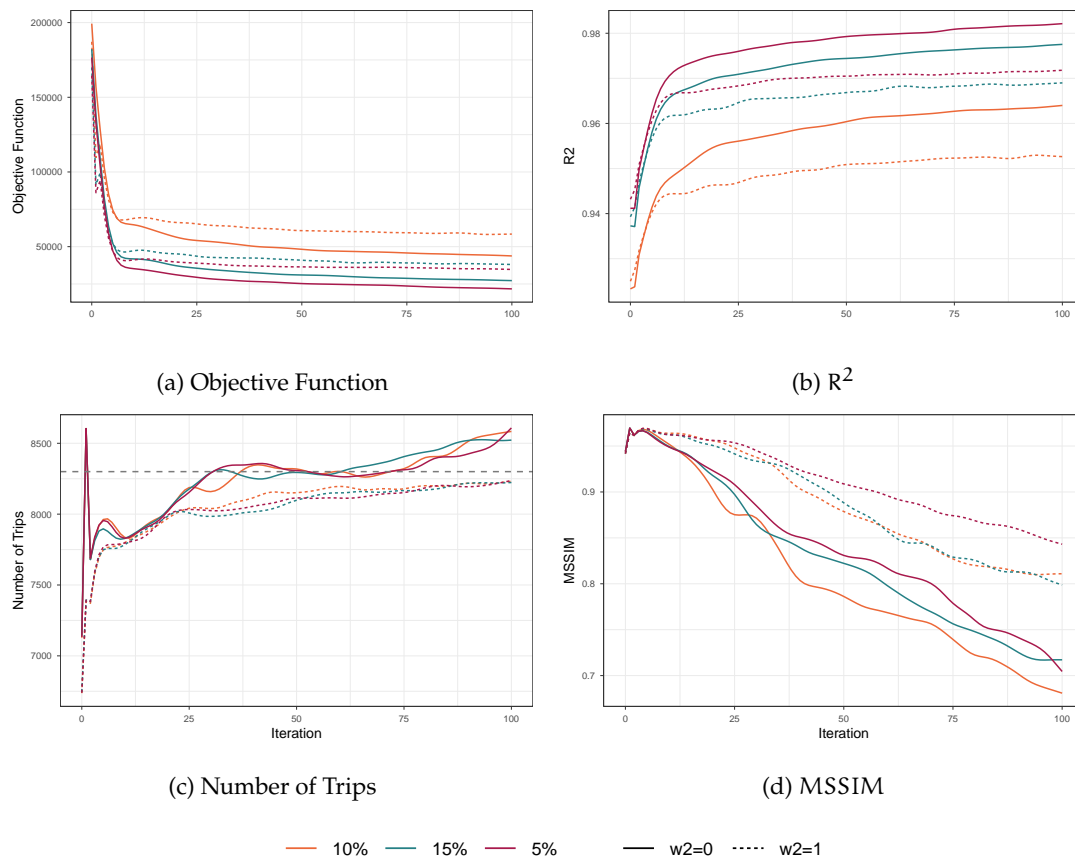
Figure 6.6: Results for data-driven assignment-free with different penetration rates and the Hist seed OD matrix

As seen in Figure 6.6, the objective function presents a fast and nice convergence in all cases. $R^2$ reaches very high values in all experiments, which means that the DODME procedure works very well as an optimization problem for adjusting traffic count measurements. In particular, $R^2$ is lower when $w_2 = 1$ as the second term of the objective function compensates the first term as a fitting regressor of the traffic counts.

The total number of trips (NT) in the estimated OD matrices is always near the ground truth total number of trips, which is $NT_{GT} = 8300$ vehicles. There are some initial anomalies in the case of $w_2 = 0$, but they are rapidly fixed. What is more, these anomalies do not occur when there is the second term in the objective function.

In terms of similarity, the contribution of $w_2 = 1$ in obtaining better MSSIM results is well known and clearly reflected on the graph. Generally, the estimated OD matrix presents a higher MSSIM when comparing to the ground truth OD matrix, than when comparing to the historical. However, as the procedure is actually an analytical-based methodology, the MSSIM evolution is similar to those shown in Chapter 5, where after an initial increase, there is an erratic behavior, decreasing at the same time as the $R^2$ increase. The suggestion in this case is, as in previous experiments, to find a manner of stopping the procedure in the early iterations, in order to preserve the structure of the matrix.



(a) Objective Function

(b) $R^2$

(c) Number of Trips

(d) MSSIM

———  w2=0    ······  w2=1    ———  Hist    ———  Obs    ———  Comb1    ———  Comb2
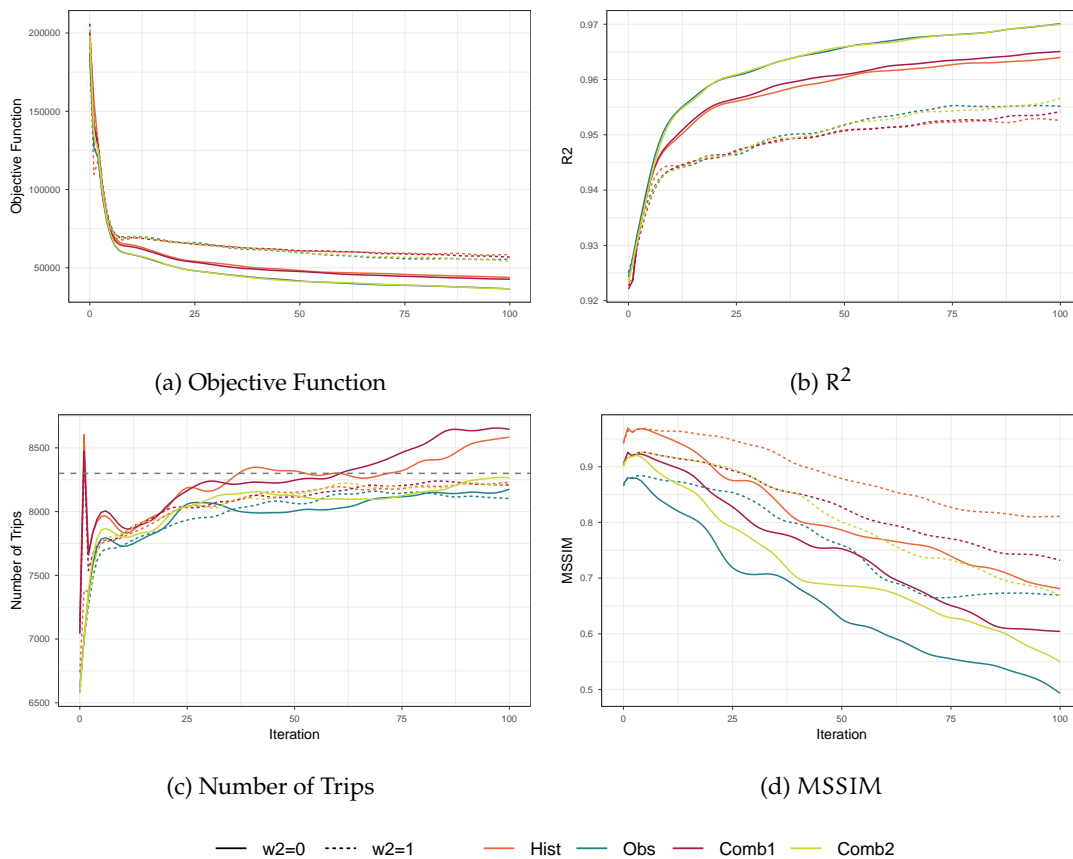
Figure 6.7: Results for data-driven assignment-free with different initial OD matrices and penetration rate 10%

In Figure 6.7, we show the resulting experiments for the different initial OD matrices at fixed mid penetration rate, that is 10%. The best choice is the historical OD matrix, since it is the OD matrix with

the largest MSSIM compared to the ground truth. However, *Comb1* and *Comb2*, that contain information about the GPS traces, do not significantly improve these indicators. That means, especially when the seed is the historical OD matrix, since the resulting OD matrix has adapted its structure to the ground truth traffic conditions, by using the proposed methodology.

### 6.3.4    Comparison to the Dynamic Spiess

In Chapter 5, a dynamic version of the Spiess method was built and studied. This method is also applied to the currently discussed network, using the same historical OD matrix as the seed OD matrix in the optimization procedure. The main difference is that Dynamic Spiess requires a dynamic traffic assignment at each iteration, while the supplementary data, link travel times, can not be included in the process. The quality of the results is compared to those obtained by DDAF in Figure 6.8.



(a) Objective Function

(b) $R^2$

(c) Number of Trips

(d) MSSIM
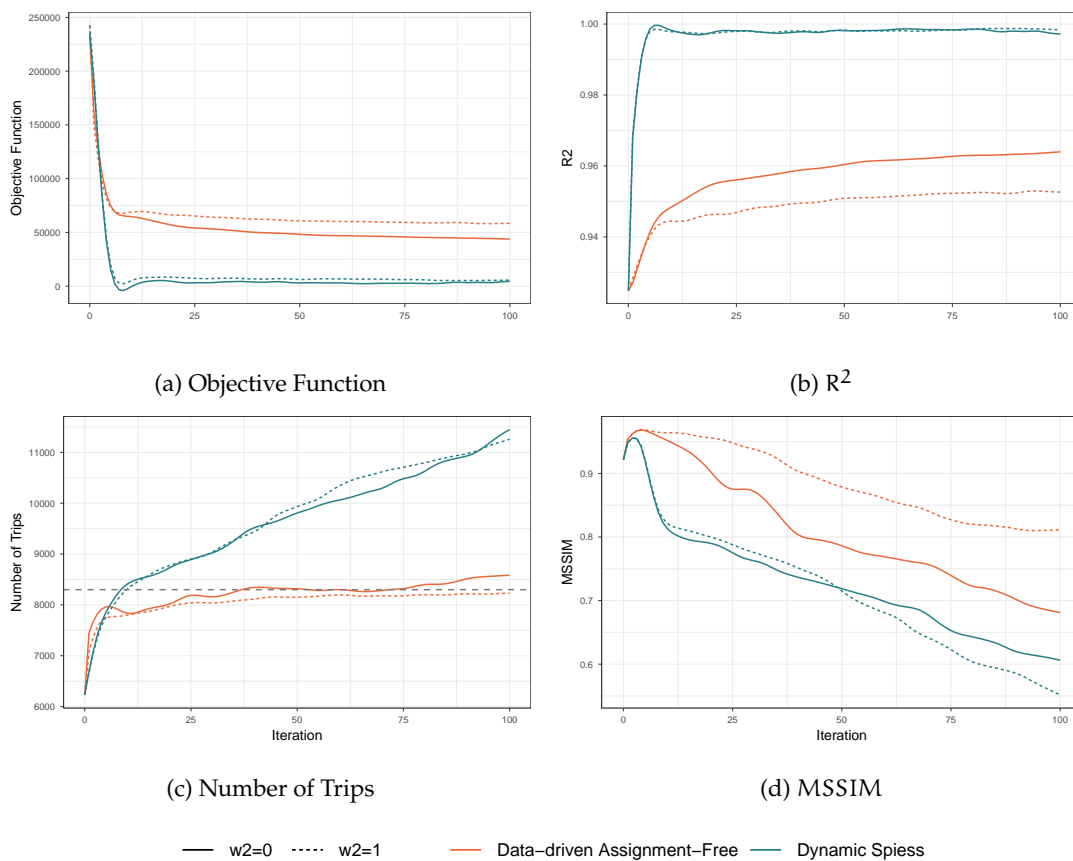
Figure 6.8: Results for data-driven assignment-free with penetration rate 10% and Dynamic Spiess

The results of dynamic Spiess and DDAF at 10% of penetration rate confirm that both analytical approaches behave similarly. While the objective function and $R^2$ are clearly showing better performance on the dynamic Spiess approach, the quality shown by the number of trips and the MSSIM confirms that the additional information of the estimated dynamic assignment matrix from the GPS traces helps to maintain the structure of the resulting OD matrix.

As previously noted, the DDAF method improves the similarity of KPIs when $w_2 = 1$. Moreover, all the procedures have similar computational times, although DDAF takes advantage of available data while

dynamic Spiess cannot include it in the OD matrix estimation.

Furthermore, it has not been taken into account that the main requirement to use a dynamic Spiess is that the network must be previously calibrated for launching a dynamic traffic assignment, from which the assignment matrix is calculated and is always a very time-consuming task. In contrast, the proposed DDAF methodology does not require a fully calibrated model since the necessary information, link travel times, can be estimated from the data.

## 6.4    Stopping criteria to preserve the structure of the OD matrix

As already seen, the analytical approaches in Chapter 5 but also the data-driven analytical approach proposed in this chapter do excellently fit the traffic counts, as it is the goal of the corresponding term of the objective function. However, they make it at the cost of increasing or decreasing certain OD flows and, therefore, altering the structure of the resulting OD matrix.

As mentioned in the objectives of this thesis, we take a special interest in the structure of the resulting OD matrix, since it reflects the demand pattern of the network. That is the reason why we address a chapter for the discussion of the structural similarity between matrices, through the use of MSSIM, and applied it in all the different case studies of this thesis. After all the results on the synthetic analysis, we, therefore, state some stopping criteria to properly stop the proposed iterative methodologies, based on the structural similarity.

In order to show and test these proposals, we will use the previous case study with the synthetic network with $w_2 = 1$ and the Historical OD matrix used as the initial OD matrix and penetration rate of the GPS technology set at 10%. This experiment is the same used in Section 6.3.1. These proposed stopping criteria are applied to the different cases and results are shown in Appendix C.2.

### 6.4.1    Based on the ground truth OD matrix



(a) Evolution by iteration

(b) Relative change
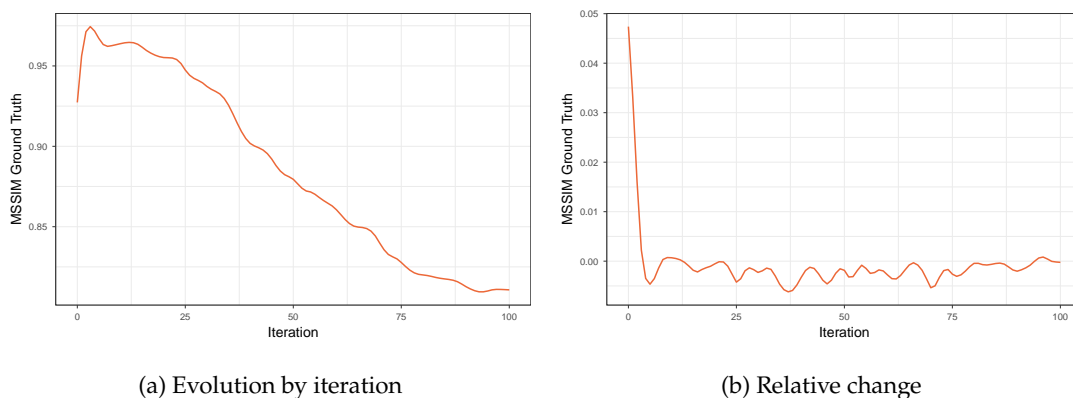
Figure 6.9: MSSIM $\left( \mathbf{X}^{(\mathrm{k})}, \mathbf{X}^{\mathrm{GT}} \right)$

In Figure 6.9, we show the MSSIM evolution between the ground truth OD matrix and the OD matrix found at iteration k. This is, indeed, one of the indicators that have been plotted at each synthetic experiment of this thesis. As already detected, there is an increase at the early iterations, which means

that the estimated OD matrix is structurally more similar to the ground truth than the one used at the beginning of the iterative procedure.

Theoretically, it would then be more appropriate to use a different stopping criterion accounting for other quality indicators, as for instance the maximum of of the *MSSIM* evolution of Figure 6.9a. Let us call it $M_1$. In this experiment, that maximum corresponds to the third iteration ($k = 3$), which corresponds to the following performance indicators of Table 6.5:

Table 6.5: Results with the proposed stopping criterion, $M_1$

|         | OF        | $R^2$  | Number of trips | MSSIM to GT |
|---------|-----------|--------|-----------------|-------------|
| Initial | 242637.70 | 0.9248 | 6232.14         | 0.9219      |
| $k = 3$ | 86417.38  | 0.9353 | 7752.56         | 0.9743      |
| Improve | -64.38%   | +1.13% | +24.40%         | +5.68%      |

However, this criterion is only available for those experiments that are synthetically designed, and therefore the ground truth OD matrix is available.

### 6.4.2   Based on the historical OD matrix

In a real network with real data, the ground truth OD matrix is not known. Therefore, in these cases, the *MSSIM* cannot be applied with respect to that OD matrix. The most reliable OD matrix that one can obtain is what we called the Historical OD matrix, which represents a household survey OD matrix that may have been used in the past and needs an estimate. These OD matrices are accepted to be structurally similar to the ground truth, representing the real demand pattern of the studied network.

Therefore, we propose to use as a stopping criterion for the analytical approaches of this thesis the *MSSIM* calculation between the OD matrix at iteration ($k$) and the historical OD matrix. That is, $MSSIM\left(\mathbf{X}^{(k)}, \mathbf{X}^H\right)$.

It is expected that the behavior of this new indicator to be a decreasing function, since the historical OD matrix is the initial OD matrix, for this experiment, and the iterative procedure perturbs it to become another function, aiming at improving the objective function.
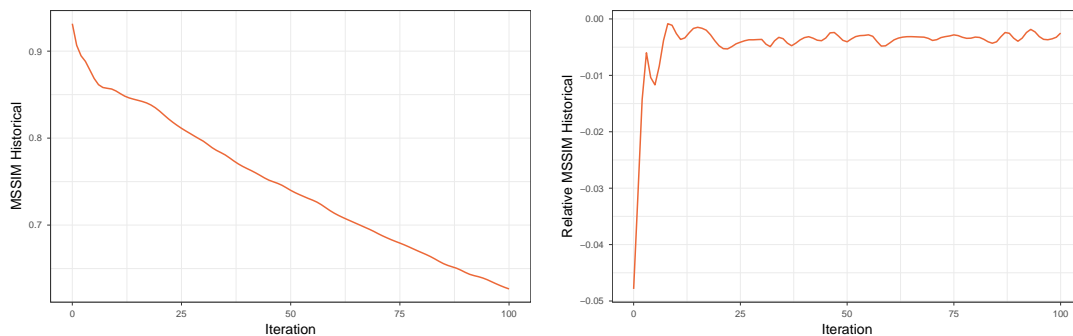


(a) Evolution by iteration                                        (b) Relative change

Figure 6.10: $MSSIM\left(\mathbf{X}^{(k)}, \mathbf{X}^H\right)$

In Figure 6.10 we show how the MSSIM evolves at each iteration. The relative change shown in Figure 6.10b shows that after a notable increase at the beginning, the evolution is more or less constant. Therefore, a threshold on the relative change can be applied:

$$M_2 = \left| \frac{\text{MSSIM}(\mathbf{X}^{(k)}, \mathbf{X}^H) - \text{MSSIM}(\mathbf{X}^{(k-1)}, \mathbf{X}^H)}{\text{MSSIM}(\mathbf{X}^{(k-1)}, \mathbf{X}^H)} \right| < \varepsilon \qquad (6.14)$$

This threshold is denoted by $M_2$. In this experiment, this stopping condition occurs at iteration $k = 7$, with $\varepsilon = 10^{-3}$. In this case, the performance indicators are shown in the next Table:

Table 6.6: Results with the proposed stopping criterion, $M_2$

|  | OF | $R^2$ | Number of trips | MSSIM to GT |
|---|---|---|---|---|
| Initial | 242637.70 | 0.9248 | 6232.14 | 0.9219 |
| $k = 7$ | 71899.04 | 0.9438 | 7694.62 | 0.9626 |
| Improve | -70.37% | +2.05% | +23.47% | +4.41% |

### 6.4.3 Based on the $MSSIM$ change at each iteration

The latter proposal is thought for the cases when a reliable historical OD matrix is not available for a real application. In this situation, nor the ground truth nor the historical OD matrix serve as a reference. Therefore, we propose a stopping criterion based on the change at successive iterations, that is calculating the $\text{MSSIM}\left(\mathbf{X}^{(k)}, \mathbf{X}^{(k-1)}\right)$. Figure 6.11 shows the corresponding graph for the selected experiment. On the left, the relative change is also shown.



(a) Evolution by iteration        (b) Relative change
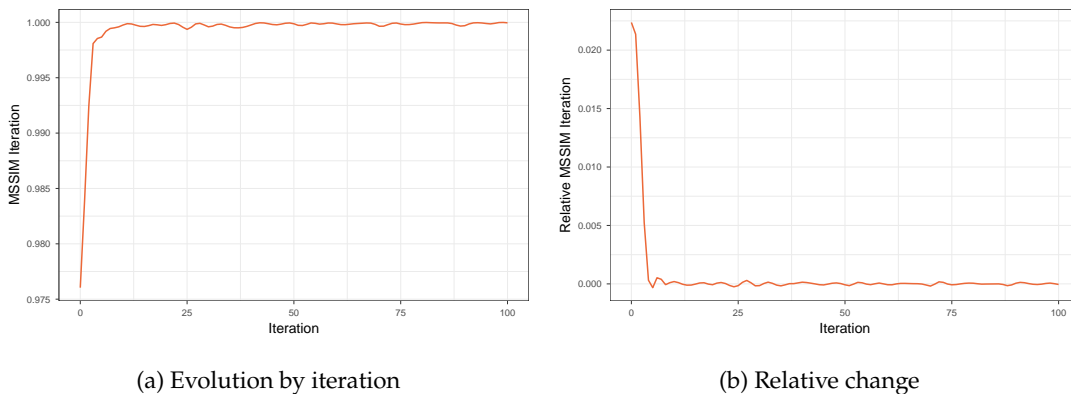
Figure 6.11: $\text{MSSIM}\left(\mathbf{X}^{(k)}, \mathbf{X}^{(k-1)}\right)$

As before, after a fast increase at the beginning of the iterative procedure, the MSSIM becomes stable around 1, meaning that the changes between iterations are not very remarkable. In this sense, a similar threshold criterion can be applied, based on the relative change:

$$M_3 = \left| \frac{\text{MSSIM}\left(\mathbf{X}^{(k)}, \mathbf{X}^{(k-1)}\right) - \text{MSSIM}\left(\mathbf{X}^{(k-1)}, \mathbf{X}^{(k-2)}\right)}{\text{MSSIM}\left(\mathbf{X}^{(k-1)}, \mathbf{X}^{(k-2)}\right)} \right| < \varepsilon \tag{6.15}$$

This criterion is denoted by $M_3$. In this example, $\varepsilon = 10^{-3}$ stops the procedure at iteration $k = 5$, which results in the following indicators.

Table 6.7: Results with the proposed stopping criterion, $M_3$

|         | OF        | $R^2$   | Number of trips | MSSIM to GT |
|---------|-----------|---------|-----------------|-------------|
| Initial | 242637.70 | 0.9248  | 6232.14         | 0.9219      |
| $k = 5$ | 75694.07  | 0.9409  | 7681.72         | 0.9651      |
| Improve | -68.80%   | +1.74%  | +23.26%         | +4.69%      |

### 6.4.4   A more robust criterion relying on the threshold band

There is a wide range of possibilities analyzing different MSSIM between OD matrices. In this case, we represent In Figure 6.12 the relative MSSIM with respect to the historical OD matrix, Equation 6.14, and also the same with respect to the previous iteration OD matrix, Equation 6.15, with the threshold band around zero, also using $\varepsilon = 10^{-3}$.



(a) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{H}\right)$    (b) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{(k-1)}\right)$
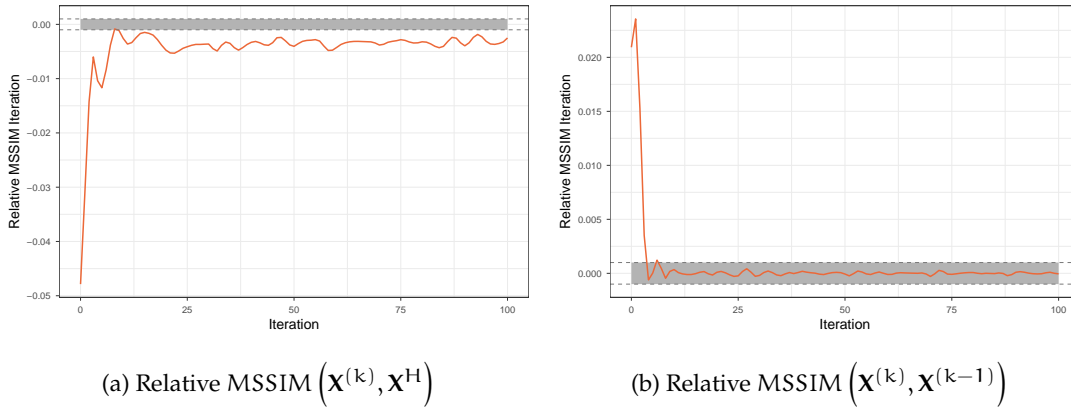
Figure 6.12: Stopping criteria with threshold $\varepsilon = 10^{-3}$

It can be seen that naturally the criterion with respect to the historical OD matrix, Figure 6.12a, is more unstable than the same criterion but between the successive iterative OD matrices, Figure 6.12b. For the sake of completeness, it has been checked for different experiments of the set, and the behavior is similar in all cases.

In this specific case, the threshold is satisfied once in the left figure, and after that, the relative MSSIM does not satisfy the stopping criterion again. It could be an effect of a particular iteration where the descent direction and step size are changing slightly in this specific measure. In general, a stopping criterion is intended to be satisfied continuously after the first reaching.

The likely conjecture that the objective function could be very plate could explain why when the stopping criterion is not the number of iterations, and the algorithms continue iterating trying to reach the convergence threshold for the objective function, if that threshold is relatively small, the iterations oscillate between values very close to the threshold. The key role is then that of the second term, fitting the simulated link flow values to the observed ones, and then the observed effect in other cases is reproduced, while the OD values are changed but keeping the value of the objective function under control, the flows fitting is improved, as the $R^2$ shows, at the price of structurally degrading the estimated OD. This suggests that possibly a more significant convergence criterion could be looking at the stability of the relative changes for a number $r$ of iterations. We denote this criterion by $M_4$.

Therefore, after studying carefully the behavior of the suggested criteria, we suggest a more robust stopping criterion, based on the successive iterative OD matrices $M_3$, Equation 6.14, but when is satisfied on the $r$ successive iterations. This last criterion, based on $r$ satisfied in consecutive iterations is denoted by $M_4$. In this example, Figure 6.12b shows that this occurs at iteration $k = 9$ when $r$ has been set to $r = 3$, showing the results in the following table.

Table 6.8: Results with the proposed stopping criterion, $M_4$

|         | OF        | $R^2$  | Number of trips | MSSIM to GT |
|---------|-----------|--------|-----------------|-------------|
| Initial | 242637.70 | 0.9248 | 6232.14         | 0.9219      |
| $k = 9$ | 70423.20  | 0.9441 | 7779.71         | 0.9634      |
| Improve | -70.97%   | +2.09% | +24.83%         | +4.50%      |

### 6.4.5   Summary of the suggested criteria

Table 6.9 summarizes the different criteria for the example used along this section, ordered by the iteration when it is first satisfied. In the end, we have added the ground truth OD matrix indicators. From the results shown in Section 6.3.3, we already know that $R^2$ increases iteration by iteration and the number of trips approximate the ground truth reference number of trips. Conversely, the similarity indicator initially increases and after reaching a maximum starts to decrease.

Table 6.9: Results with all the proposed stopping criterion

|         | Criteria Satisfied | OF        |         | $R^2$  |        | Number of trips |         | MSSIM to GT |        |
|---------|--------------------|-----------|---------|--------|--------|-----------------|---------|-------------|--------|
| Initial |                    | 242637.70 |         | 0.9248 |        | 6232.14         |         | 0.9219      |        |
| $k = 3$ | $M_1$              | 86417.38  | -64.38% | 0.9353 | +1.13% | 7752.56         | +24.40% | 0.9743      | +5.68% |
| $k = 5$ | $M_3$              | 75694.07  | -68.80% | 0.9409 | +1.74% | 7681.72         | +23.26% | 0.9651      | +4.69% |
| $k = 7$ | $M_2$              | 71899.04  | -70.37% | 0.9438 | +2.05% | 7694.62         | +23.47% | 0.9626      | +4.41% |
| $k = 9$ | $M_4$              | 70423.20  | -70.97% | 0.9441 | +2.09% | 7779.71         | +24.83% | 0.9634      | +4.50% |
| GT      |                    | 0.00      |         | 1.000  |        | 8300.00         |         | 1.000       |        |

As already stated, the best criterion is $M_1$ because it is comparing directly to the ground truth OD matrix. However, from a practical point of view, where the ground truth is not always available. The relative change of the MSSIM between the estimated OD matrix and the historical OD matrix shows difficulties in converging in the band of the threshold, as seen in Figure 6.12a. On the other hand, the relative

MSSIM between successive iterations, $M_3$ is more stable, as the minimization procedures normally generate smaller steps iteration by iteration. In this sense, $M_3$ and $M_4$ are the more suitable criteria for stopping the iterative procedure, as the stopping criterion because it controls the structural variability at the early iterations, near the maximum of the MSSIM and stops the uncontrolled decrease at the next iterations. It is then an equilibrium between the desired increase of the number of trips and fitting of traffic counts with the undesired decrease of the structural similarity.

## 6.5 Real networks with real data

Despite the already mentioned drawbacks of the currently available physical measurements in Chapter 2 and considering our experience with the synthetic data, we, for the sake of completeness, conducted a further test on two real networks using commercial GPS data to infer estimated link travel times. The selected exercise is useful for the practitioner's point of view since a ground truth OD matrix is not available, but only traffic counts, a historical OD matrix and estimated link travel times obtained from GPS commercial data.

### 6.5.1 Case Study 2: Real Network of Turin

The network used is shown in Figure 6.13, with its characteristics presented in Table 6.10. It corresponds to the downtown of the capital of Piemonte region, Turin, in Italy. The detection layout comprises 302 counting stations, situated in the network as shown and prioritizes traffic counts over main streets. Furthermore, there is a historical OD matrix with a high level of confidence, since it is the estimated OD matrix of a previous study.



Figure 6.13: The Turin network used with the detection layout

The available mobility data for this network is a sample of GPS tracking data provided by INRIX that contains one year of indistinguishable private and fleet vehicles circulating on labor days during the peak period in the morning. The sample contains 3.76M waypoints, which represents 232k different trips (partially) circulating on the network. The penetration rate is 1.32% if we compare yearly GPS trips to the historical OD matrices for the selected period in terms of the number of trips. Moreover, this incompleteness reflects a mobility pattern that does not correspond to the network's OD pattern, as shown by further analysis of the matrix structure.

Table 6.10: Network and OD characteristics

| | |
|---|---|
| Time periods | 4 |
| Zones | 221 |
| Detectors | 302 |
| OD pairs X Time | ≈195k |
| Historical Trips | 129k |
| Historical Positive OD | ≈77.5k (39.71%) |

Since the data set does not provide information regarding the vehicle types, we cannot distinguish between fleet and regular vehicles nor filter and cleansing the sample to obtain estimations of the link travel times with a controlled degree of confidence. However, an estimation of link travel times for each time period of study has been made using the methodology described in Section 2.5.

Because this is a real network experiment, the ground truth conditions are unknown and it is, therefore, impossible to compare the resulting OD matrix to the ground truth OD matrix. The reference OD matrix, in this case, is the available historical OD matrix, as it is a reliable OD matrix from a previous study.

The observed OD matrix is built from the GPS data set by aggregating the trips according to their origin and destination zones, as well as their departure times. The obtained OD matrix, $\hat{\mathbf{X}}$, has 35k positive OD values, which are 17.76% of the OD values. The term $S$ of the $MSSIM$ index of Equation 3.3 between the observed and the historical OD matrices is an appropriate indicator of whether the observed OD matrix is a good seed choice for the optimization procedure since it indicates how similar these matrices are. In this case, the term $S(\hat{\mathbf{X}}, \mathbf{X}^{\mathrm{H}}) = 0.0287$, thus indicating it is not suitable as a seed OD matrix, since the captured mobility pattern is not similar to that of the more reliable historical OD matrix. The presence of fleet vehicles and the unknown percentage of them is the main cause for discarding this observed OD matrix as a seed, since fleet vehicles have neither a fixed origin nor destination.

In Figure 6.14, we show the objective function, $R^2$, number of trips and $MSSIM$ evolution. In this case, as it is a real network with unknown ground truth OD matrix, the $MSSIM$ is performed between the estimated OD matrix and the Historical one, which also serves as the starting OD matrix for the optimization procedure. The behavior showed of the objective function and $R^2$ evolution are similar to those obtained in the synthetic exercise. However, regarding the number of trips, the evolution is different. In Figure 6.14c we see an initial descent, followed by an increasing return to the historical OD number of trips. It can be understood saying that initially the number of trips are decreased to globally match traffic counts, but fine tuning the fitting to traffic counts requires modifying the OD matrix by increasing some OD flows (thus increasing the trips across iterations within the prescribed limits for some OD pairs to enhance $R^2$, while decreasing the similarity with respect to $MSSIM$ indicator). The $MSSIM$ starts at 1 because the historical OD matrix is used as a seed for optimization.

(a) Objective Function

(b) $R^2$

(c) Number of Trips

(d) MSSIM
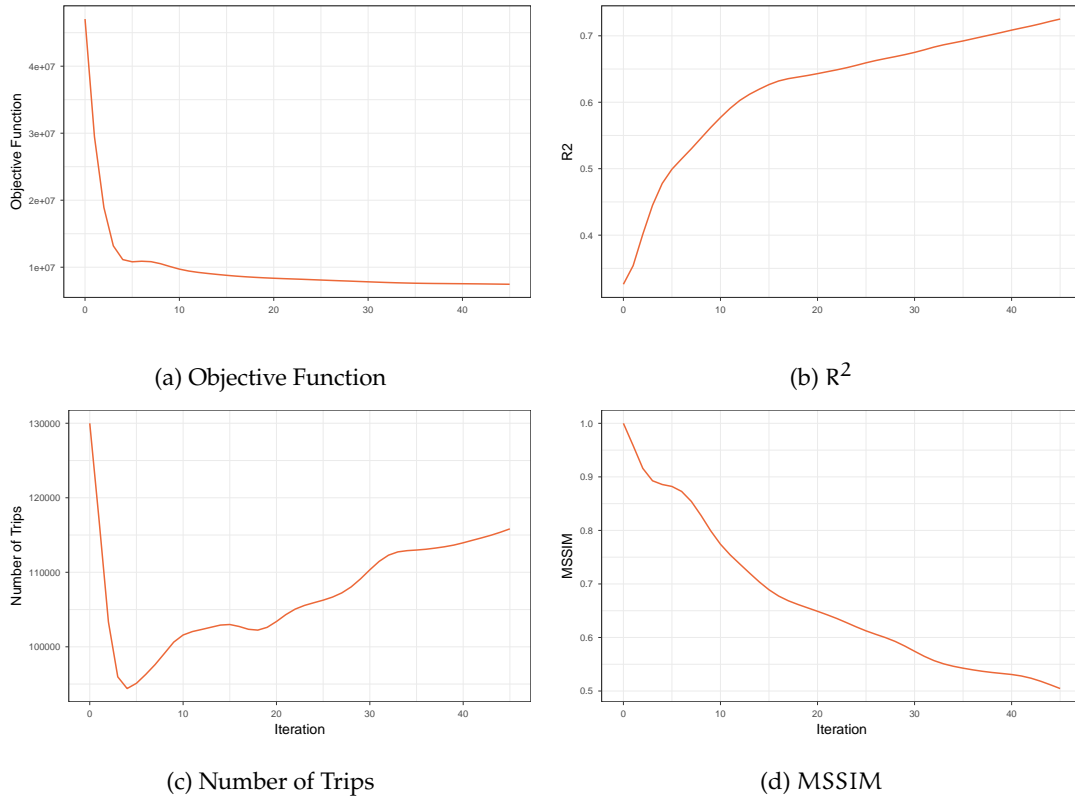
Figure 6.14: Results for data-driven assignment-free with Turin's network

The DDAF method converged after 45 iterations using a threshold on the objective function relative change at each iteration, and the linear regression between the traffic counts of sensors and the corresponding simulated values is included in Figure 6.15. The corresponding simulated traffic counts have been obtained by launching a DTA with the resulting estimated OD matrix. As shown, the optimization procedure increases the fitting of these measurements, from $R_0^2 = 0.3261$ to $R_f^2 = 0.7266$.



(a) $\mathbf{Y} = \overline{\mathbf{A}} \cdot \mathbf{X}^0$

(b) After a DTA($\mathbf{X}^*$)
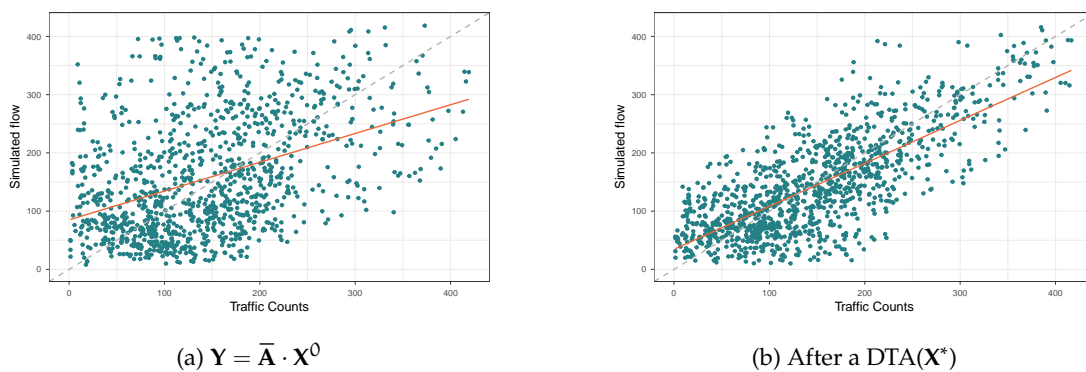
Figure 6.15: $R^2$ improvement after DDAF method in Turin

The total number of trips resulting from the estimated OD matrix are $NT = 120974$ and the positive OD values are 71983 (36.85%). The similarity between the historical and the estimated OD matrix measured with MSSIM is $MSSIM\left(\mathbf{X}^*, \mathbf{X}^H\right) = 0.5167$.

In Section 6.4, we suggested 4 different stopping criteria based on the structural similarity. In this case, we use three of them (since the first one does not apply because the ground truth OD matrix is unknown) in a posteriori process to see their effect in relation to the final results. Figure 6.16 shows the relative $MSSIM$ with respect to the Historical OD matrix, left, and the relative $MSSIM$ between successive iterations, right. In the two cases, the threshold is indicated, which has been set to $\varepsilon = 10^3$.



(a) Relative $MSSIM \left( \mathbf{X}^{(k)}, \mathbf{X}^{H} \right)$           (b) Relative $MSSIM \left( \mathbf{X}^{(k)}, \mathbf{X}^{(k-1)} \right)$
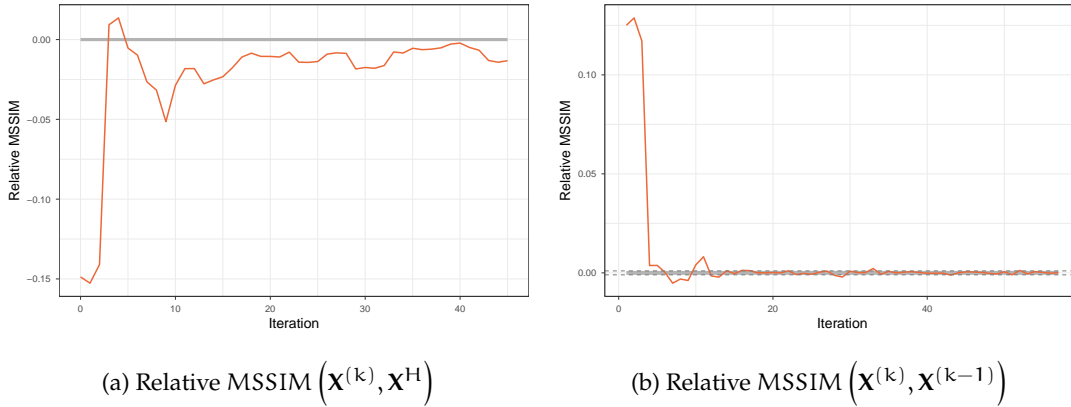
Figure 6.16: Relative $MSSIM$ with threshold $\varepsilon = 10^{-3}$

On the left graph, the relative $MSSIM$ does not fit into the band, meaning that it never reaches a value in $[-\varepsilon, \varepsilon]$. On the right, 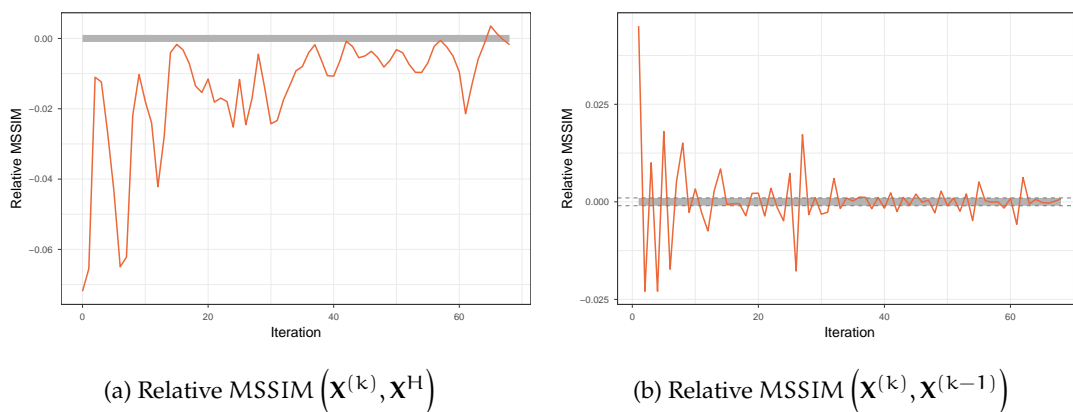the first time it occurs is at iteration $k = 6$. The more sophisticated criterion, when the relative $MSSIM$ remains 3 successive iterations inside the band stops the algorithm at iteration $k = 19$. In Table 6.11, we show the KPIs, for the stopping points, at iterations $k = 6, 19, 45$.

Table 6.11: Results with the proposed stopping criteria

|         | Criteria satisfied | OF       | $R^2$  | Number of trips | MSSIM to Hist |
|---------|--------------------|----------|--------|-----------------|---------------|
| Initial |                    | 47035189 | 0.3261 | 129990          | 1.0000        |
| $k = 6$ | $M_3$              | 11516217 | 0.4975 | 96761           | 0.8809        |
| $k = 19$| $M_4$              | 8423369  | 0.6034 | 102338          | 0.6557        |
| $k = 45$| ObjFun             | 7458148  | 0.6393 | 115899          | 0.5167        |

The proposed criteria set the stop at different iterations. However, $M_2$ does not reach the fixed threshold. In this case, $M_4$ is the best stopping criterion because, as shown in Figure 6.16b, $M_3$ would stop the criterion at an early iteration, because it suddenly reached the threshold band, but there are some iterations after that. Therefore, $k = 19$ is the suggested stopping iteration, which represents a compromise between a good fit to traffic counts (indicated by $R^2$), limiting the increase of OD flows in some OD pairs (and thus the total number of trips) while preserving the Historical OD pattern ($MSSIM$).

## 6.5.2   Case Study 3: Real Network Barcelona

The last network used is shown in Figure 6.17, with its characteristics presented in Table 6.12. It corresponds to the downtown of my birth city, Barcelona, in Catalonia, Spain. The detection layout comprises 70 counting stations, situated in the network as shown and prioritizes traffic counts over main

streets. Furthermore, there is a historical OD matrix estimated OD matrix of a previous study by using a submatrix extraction procedure based on a partially calibrated network.



Figure 6.17: The Barcelona network used with the detection layout

The available mobility data for this network is a sample of GPS tracking data provided by INRIX that contains only 1 month of distinguishable private and fleet vehicles circulating on labor days during the peak period in the morning. The sample contains 566k waypoints, which represent 20k different trips (partially) circulating on the network. The penetration rate, including the two classes of vehicles, is around 2% if we compare the available GPS trips for one month to the historical OD matrices for the selected period in terms of the number of trips.

Table 6.12: Network and OD characteristics

| | |
|---|---|
| Time periods | 4 |
| Zones | 202 |
| Detectors | 70 |
| OD pairs X Time | ≈163k |
| Historical Trips | 60k |
| Historical Positive OD | ≈44k (26.77%) |

Different from the GPS data from Turin, in this data set, we have information about the nature of the trip, distinguishing between fleet and private vehicles. We use this information to build the observed OD matrix. Since the fleet vehicles have neither an origin nor destination, they will not be included in the observed OD matrix. On the other hand, they were included in the estimation of link travel times because they use the same network with the same traffic conditions, so their travel times are reliable. Effectively, both scenarios were initially considered and the effects of removing the fleet vehicle were insignificant. The methodology used to obtain the link travel times estimates is the one described in Section 2.5 of this thesis.

Because this is a real network experiment, the ground truth conditions are unknown and it is, therefore,

impossible to compare the resulting OD matrix to the ground truth OD matrix. The reference OD matrix in this case is the available historical OD matrix.

The observed OD matrix is built from the GPS data set by aggregating only the private trips according to their origin and destination zones, as well as their departure times. The obtained OD matrix, $\hat{\mathbf{X}}$, has around 23k positive OD values, which are 13.91% of the OD values. The term S of the MSSIM index of Equation 3.3 between the observed and the historical OD matrices is an appropriate indicator of whether the observed OD matrix is a good seed choice for the optimization procedure, since it indicates how similar these matrices are. In this case, the term $S(\hat{\mathbf{X}}, \mathbf{X}^{\text{H}}) = 0.1814$, thus indicating it is not suitable as a seed OD matrix, since the captured mobility pattern is not similar to that of the more reliable historical OD matrix. Even in this case, that the fleet cars are removed to build the observed OD matrix, the S factor is still low, showing that the observed OD matrix does not reflect the mobility pattern of the network, which is another detected problem of the commercial GPS data sets.

In Figure 6.18, we show the same indicators as in the other networks. In this case, as it is a real network with unknown ground truth OD matrix, the MSSIM is performed between the estimated OD matrix and the Historical one, which also serves as the starting OD matrix for the optimization procedure.

The behavior showed of the objective function and $R^2$ evolution are similar to those obtained in the synthetic exercise. However, regarding the number of trips, the evolution is similar to those in Turin. In Figure 6.18c we see an initial descent, followed by an increasing return to the historical OD number of trips. The MSSIM starts at 1, because the historical OD matrix is used as a seed for the optimization.



(a) Objective Function

(b) $R^2$

(c) Number of Trips

(d) MSSIM
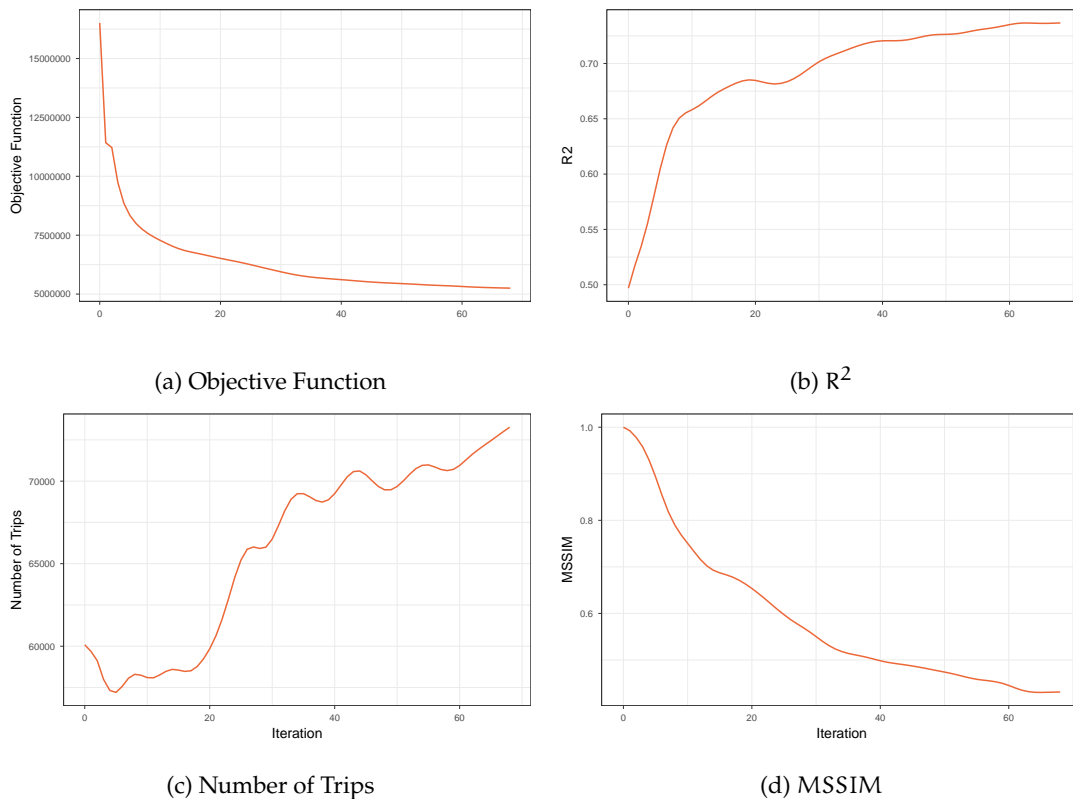
Figure 6.18: Results for data-driven assignment-free with Barcelona's network

The DDAF method converged after 68 iterations using a threshold on the objective function relative

change at each iteration, and the linear regression between the traffic counts of sensors and the corresponding simulated values is included in Figure 6.19. The corresponding simulated traffic counts have been obtained by launching a DTA with the resulting estimated OD matrix. As shown, the optimization procedure increases the fitting of these measurements, from $R_0^2 = 0.4969$ to $R_f^2 = 0.7367$.



(a) $\mathbf{Y} = \overline{\mathbf{A}} \cdot \mathbf{X}^0$          (b) After a DTA($\mathbf{X}^*$)
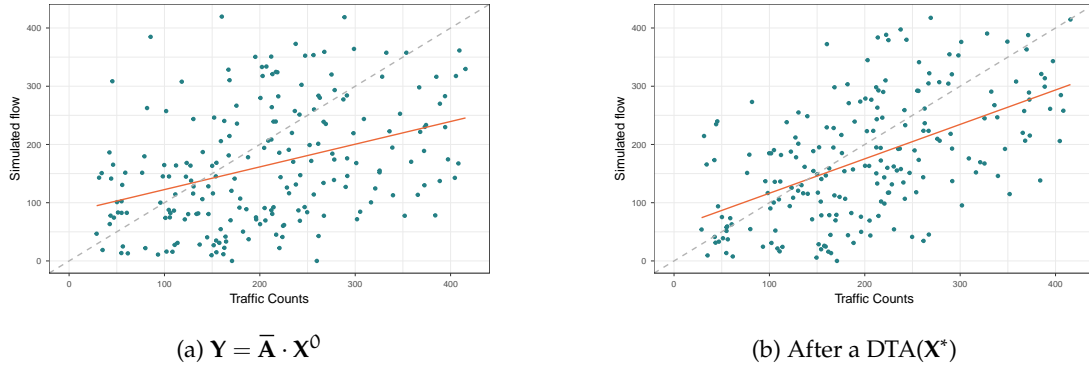
Figure 6.19: $R^2$ improvement in a selected experiment

The total number of trips resulting from the estimated OD matrix are $NT = 73190$, $+21\%$ higher than the historical number of trips ($NT(\mathbf{X}^H) = 60083$), and the positive OD values are 43215 (26.47% of the total OD pairs). The similarity between the historical and the estimated OD matrix measured with MSSIM is $MSSIM\left(\mathbf{X}^*, \mathbf{X}^H\right) = 0.4308$.

As done in Turin, we use the four stopping criteria proposed in Section 6.4 in a posteriori process to see their effect concerning the final results. Figure 6.20 shows the relative MSSIM with respect to the Historical OD matrix, left, and the relative MSSIM between successive iterations, right. In the two cases, the threshold is indicated, which has been set to $\varepsilon = 10^{-3}$.



(a) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^H\right)$        (b) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{(k-1)}\right)$

Figure 6.20: Relative MSSIM with threshold $\varepsilon = 10^{-3}$

In the left graph, the relative MSSIM fits into the band only at iteration $k = 42$ and $k = 67$. On the right, the first time it occurs is at iteration $k = 15$. The $M_4$ sophisticated criterion, when the relative MSSIM remains $r = 3$ successive iterations inside the band stops the algorithm at iteration $k = 17$. In Table 6.13, we show the KPIs, for the stopping points, at iterations $k = 15, 17, 42, 68$.

Table 6.13: Results with the proposed stopping criteria

| | Criteria Satisfied | OF | $R^2$ | Number of trips | MSSIM to Hist |
|---|---|---|---|---|---|
| Initial | | 18849753 | 0.4969 | 60083 | 1.0000 |
| $k = 15$ | $M_3$ | 6794112 | 0.6762 | 58676 | 0.6884 |
| $k = 17$ | $M_4$ | 6707685 | 0.6813 | 58042 | 0.6823 |
| $k = 42$ | $M_2$ | 5466618 | 0.7272 | 69243 | 0.4778 |
| $k = 68$ | ObjFun | 5252524 | 0.7367 | 73190 | 0.4308 |

The analysis of the different criteria shows similar behavior. On one hand, $M_2$ reaches the threshold at iteration 42, because it oscillates during the early iterations. On the other hand, $M_3$ and $M_4$ stop the optimization algorithm at earlier iterations, preserving better the structure of the matrix, but reaching lower fittings values.

### 6.5.3   Revision for Dynamic Spiess with Stopping criteria



(a) MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{GT}\right)$



(b) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{H}\right)$

(c) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{(k-1)}\right)$
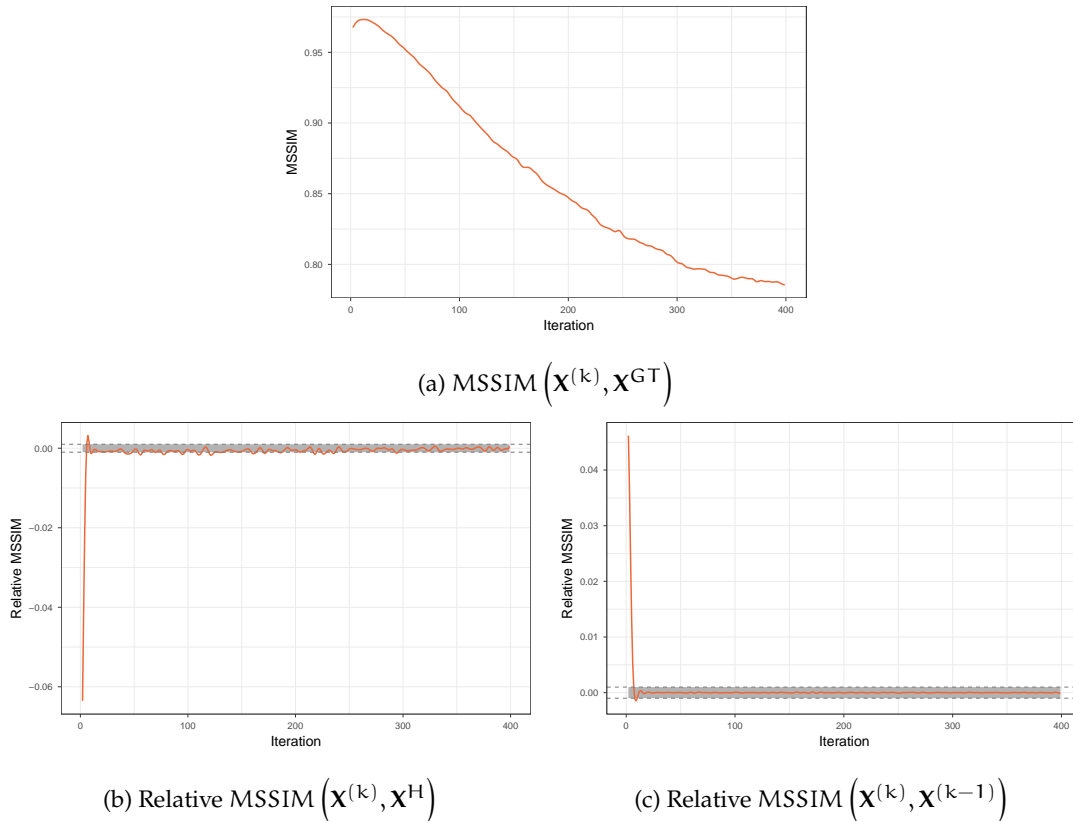
Figure 6.21: MSSIM and Relative MSSIM with threshold $\varepsilon = 10^{-3}$

After testing this proposal with the data-driven assignment-free DODME experiments in this chapter, we wanted to test these criteria with the experiments in Chapter 5, using the Dynamic Spiess algorithm. In that case, there was the same phenomenon, where after an increase of the similarity between the

estimated OD matrix and the ground truth of the synthetic experiment, there was an uncontrolled decrease of such similarity.

Consistently with the criteria proposed in Section 6.4, we show in Figure 6.21 an experiment of the case study shown in Section 5.3. It consists of the experiment with the Dynamic Spiess algorithm applied to the Hillsboro network, with $w_2 = 1$ and using the Multitude OD matrix as historical. The different MSSIM and relative MSSIM are shown to determine, in each case, when the algorithm would be stopped. The rest of the experiments of the Dynamic with the criteria are graphed in Appendix C.1.

In Figure 6.21a, we show the MSSIM with respect to the ground truth, because it is again a synthetic experiment. In this case, the algorithm should be stopped when the maximum is reached, that is at iteration $k = 11$. In Figure 6.21b, the relative MSSIM to the historical OD matrix is shown, and it satisfies the threshold criteria at iteration $k = 5$. Finally, in Figure 6.21c, we show the MSSIM between successive iterations, which reaches the threshold also at iteration $k = 5$. In the latter, the threshold is satisfied from $k = 5$ to the last iteration, so the last suggestion would stop the algorithm at iteration $k = 7$. In summary, the results of the four KPIs are shown in Table 6.14, where at the end the ground truth indicators are included.

Table 6.14: Results with the proposed stopping criteria

|  | Criteria Satisfied | OF | $R^2$ | Number of trips | MSSIM to GT |
|---|---|---|---|---|---|
| Initial |  | 6358827 | 0.5345 | 7408 | 0.9169 |
| $k = 5$ | $M_2, M_3$ | 540415 | 0.8423 | 8883 | 0.9709 |
| $k = 7$ | $M_4$ | 409375 | 0.8534 | 8966 | 0.9725 |
| $k = 11$ | $M_1$ | 303638 | 0.8889 | 8953 | 0.9736 |
| GT |  | 0 | 1.0000 | 9878 | 1.000 |

As we see, the synthetically designed experiments of Chapter 5 show a similar behavior regarding the stopping criteria. The one that is structurally more similar to the ground truth is $M_1$ followed by $M_4$. As before, the use of $M_4$ has a high MSSIM value but worsens the fitting and the total number of trips.

## 6.6 Conclusions of Data-driven Assignment-Free DODME

The components shared by most of the DODME approaches are: an assignment matrix $\mathbf{A} = [a_{ijr}^{lt}]$, whose elements represent the proportion of the OD demand; $x_{ijr}$ travelling from origin $i$ to destination $j$, departing the trip at time period $r$; and reaching the counting station on link $l$ at time $t$; a historical OD matrix $\mathbf{X}^H$ that provides additional information on the mobility patterns, namely their space-time structure; and link flow counts, $\hat{\mathbf{Y}} = [\hat{y}_{lt}]$, for a subset of links $l \in \hat{L} \subseteq L$ where counting stations are located. In analytical methods, the assignment matrix describing the dynamic structure of the temporal use of the network is usually costly to obtain using dynamic traffic assignment procedures that depend on an initial OD matrix. This initial matrix is not always reliable and in addition, the OD estimation applications are limited by the significant computational effort that they require to rely on a good network calibration.

In this chapter, we investigated whether the data provided by new ICT sources (namely GPS data) could empirically provide better estimates in DODME practical applications. In other words, our general hypothesis assumes that the data contain information about the generating phenomenon, by which we

aim to specifically find a suitable mean to process the data and incorporate the information into the DODME process.

As already mentioned in this thesis, the exploratory analysis of the GPS tracking from the commercial providers reveals that data collection methods are tailored to commercial goals and that nowadays this data is not able to generate a reliable initial OD matrix for a traditional DODME process. The process of synthetic data generation described in Chapter 2 permitted us to generate a framework to validate each step of the assignment matrix building process from the information about the real mobility enhanced in the GPS data. From this information, we can estimate link travel times from the traffic state revealed from the ICT data.

The experimental design based on synthetic data allows conducting a sensitivity analysis of the equipped vehicles' penetration rate. Furthermore, this analysis could be difficult to perform when using only purely empirical data. The sensitivity analysis shows the robustness of the assignment-free approach, and, as expected, an increase in the quality of the results when the percentage of equipped vehicles increases.

The computational results prove that the proposed methodology is reliable for estimating the key component: the assignment matrix. The estimated assignment matrix is consistent, as well as the empirical route choice set approach to identify the most likely used paths, their travel times, and the path flow proportions from which the assignment matrix is derived.

The computational performance is very good, converging very quickly in a few iterations to achieve good results. A relevant point to highlight is the consistent performance regarding the total number of trips and the structure of the final estimated matrix. Most of the approaches in the literature also exhibit good performance in terms of the converging objective function and the quality of the results when using only $R^2$ as a KPI to measure the performance, but no analysis is made on the total number of trips in the final estimated OD matrix or its structure.

Quite frequently, $R^2$ is a very good indicator for a simple meta-regression model, but at the price of increasing or decreasing the number of trips or destroying historical OD trip matrix patterns. In order to fit the observed flow count in a link, it pulls forward to and backward from the OD pairs whose paths use that link. However, considering the underlying physical system (i.e., the transportation system), the resulting estimated OD matrix may not be realistic because some affected OD pairs are forced to generate or attract an unrealistic number of trips. The developed assignment-free DODME outperforms on matrix similarly, nevertheless is also based on obtaining a good quality of the $R^2$ that explains the link flows, but it exhibits remarkable stability in the total number of vehicles when comparing the ground truth OD to the resulting estimated OD. A high degree of structural similarity also exists between both matrices. Therefore, we can conclude that the estimated OD is more reliable than those obtained by other approaches.

The computational results reported using real networks and real data in Turin and Barcelona case studies show clearly that, as with all data-driven approaches, the quality of the results strongly depends on the quality of the data used, achieving in these cases worse results than in the synthetic case. Although the observed OD matrices from commercial providers data do not directly reflect OD pattern, there is a significant improvement when the data-driven OD estimation procedure is applied. Using only measured data to define an initial OD matrix is recommended when the data quality ensures observed OD matrices, which could perhaps be achieved when using purpose-oriented commercial data (per agreement with the data providers). This would thus overcome the mentioned drawbacks.

Even in analytic assignment-based oriented methods, using a stopping criterion based only on the objective function convergence is not a guarantee of quality, since the $R^2$ fitting dominates the convergence process, and good flow fits do not ensure good structural quality. Alternative stopping criteria that

orchestrate between $R^2$ (traffic counts fit) and $MSSIM$ (structure similarity of the travel pattern in the estimated matrix) has been found to perform better and therefore has been proposed. These criteria are useful for Spiess-like methods and assignment-free DODME as shown.

# 7
# Conclusions

*This final chapter summarizes the main contributions and conclusions of the thesis. Moreover, some further research is highlighted.*

This thesis deals with the Dynamic OD Matrix Estimation problem, which is an appealing and crucial problem for traffic analysis modeling and simulation. The OD matrices are key inputs for the traffic simulation models and therefore, they must be properly calibrated in order to reproduce the correct traffic mobility. This problem has been studied for decades and the bi-level optimization formulation is the reference approach as it permits to constantly compare to the observed traffic data from the real network to correct the OD matrix accordingly. However, the problem is highly underdetermined and computationally costly, increasing the computational burden as the size of the network does.

This thesis studied carefully and systematically the dynamic OD estimation problem, aiming at proposing a methodology that permits to add the GPS data information about the traffic state and thus reduces the inherent underdetermination of the problem. For this purpose, we worked on different lines of research that are related to how the problem is solved. Moreover, we always focused the analyzes not only on the computational and convergence point of view but also on the quality of the obtained results. In this sense, we have studied the computational experiments using four different indicators: the objective function, the fitting between traffic counts, the total number of trips of the OD matrices and the measure of similarity between OD matrices, that takes into account the structure of the OD matrices, which represent indeed the demand pattern of the network.

The emergence of information and communication technologies (ICT) generated different types of traffic data. Before it, the traffic data was collected by sensors placed onto the network that permitted to count the number of vehicles that circulate through it. The new paradigm of sensors, that are connected due to the ICT technology, permitted reidentification of the same vehicle downstream, which opens a wide range of information about the traffic state, as a form of travel times, path reconstruction, etc. We took a special interest in commercial GPS data, which thanks to the emergence of GPS devices, collects continuously big amounts of data from the vehicles and permits us to obtain vehicle location information evolving in time. This data is usually subject to many privacy policies and the origin and destination information is not available because it has been pre-processed before the acquisition of it. In Chapter 2, we propose a methodology to estimate the link travel times for the different time periods, based on the big amount of vehicles using the network during many days with similar conditions. This extracted information of the form of link travel times permits us to obtain further information regarding the traffic

state and it can be included in the OD estimation procedures. In the same chapter, we also propose a methodological framework to synthetically generate consistent traffic counts, GPS and historical OD matrix databases.

In Chapter 3, we follow the studies from Djukic (2014) and Behara et al. (2018) about the measures of quality of the solution. In this sense, we adapt the $MSSIM$ measure of similarity to the OD estimation problem, proposing a weighting formula that takes into account the total origin or destination vehicles contributing to the travel demand pattern. With this, we suppress the degree of freedom of selecting the sliding window size and we do not depend on further information regarding the regional aspect of the zonification.

In the exhaustive study of the existing methodologies to solve the OD estimation problem, on one hand, we studied the simulation-based optimization approaches, which reference is the SPSA procedure. In Chapter 4, we stated an automatic procedure for selecting the gain sequences parameters, which are key inputs for the good behavior of the stochastic process. Moreover, we proposed many alternative methods and variants that permit us to add further information, as constraints to the problem, and to overcome the drawbacks of SPSA for solving the OD estimation problem. By the end of the chapter, we propose an alternative of introducing the travel times information coming from commercial GPS data. As these data do not provide reliable information regarding the origin-to-destination travel times, we proposed a maximal subpaths heuristic to introduce the travel times to the objective function of the bi-level problem.

For the SPSA contributions, we completed a full set of experiments testing the different variants and the improvement of such inclusion. We finally concluded that the computational times of SPSA and the high sensitivity that the quality of the estimated OD matrix has with respect to the initial OD matrix are two aspects that make SPSA unsuitable to find a proper estimated OD matrix in a computationally reasonable time.

On the other hand, we studied the analytical approaches in Chapter 5. In this chapter, we stated the formulation of the dynamic version of Spiess (1990) approach and tested it with many different experiments. With the experimental results, we concluded that the analytical approaches, despite their impossibility to add new traffic data to the objective function, are more robust to the initial OD matrix, present fast convergence properties and are computationally more efficient. Moreover, we noticed the important role of the dynamic assignment matrix in all the different steps of the iterative procedure.

After the knowledge acquired, in Chapter 6, we designed carefully a data-driven assignment-free methodology to obtain from the GPS data information an estimate of the dynamic assignment matrix on the traffic conditions of the traffic data. Therefore, we reduce notably the computational effort of the OD estimation problem, because a traffic assignment is not needed anymore, since the assignment matrix is obtained from the GPS data processing. To wrap it up, we propose an optimization problem with scaling factors that both reduces the number of variables of the problem (and thus reduces the computational effort and the underdetermination) and also permits to add constraints to the OD flows.

This methodology has been validated and tested with a synthetically generated database to study its consistency, robustness and sensitivity. It has been also compared with the Dynamic Spiess to see the effect of the assignment-free nature of the procedure. By the end, it has been tested with real networks (Turin and Barcelona) showing promising improvements, even though the difficulties found in working with real commercial GPS data.

At the end of Chapter 6, we also designed a set of stopping criteria based on several measures of performance including the $MSSIM$ matrix similarity of the estimated matrices over the iterations. This is a proposal that aims to find an equilibrium between the regression nature of the bi-level optimization

problem and the quality of the obtained solution, that could be deeply studied thanks to the set of synthetic experiments, where the ground truth OD matrix is available.

After the experience gained in this thesis with the dynamic OD matrix estimation, we analyzed carefully different methodologies that tackle the OD estimation. In the case when the available traffic data are only from traffic counts on links, we strongly suggest using an analytical approach such as the dynamic Spiess proposed in Chapter 5. However, these approaches need a DTA engine and a properly calibrated model. On the other hand, the data-driven approaches are alternative methods that use a GPS trace data set to estimate the dynamic assignment matrix and reduce the underdetermination of the problem. The proposed alternative takes advantage of this new data set and the DTA is not required anymore.

## 7.1 Further Research

The ICT devices are generating more and more data about mobility in urban networks. Therefore, the traffic models, analysis and simulation must evolve together with these data, making use of them to improve the likelihood of their outputs. However, the huge volume of data can become a problem if these data are not well collected, meaning subject to a data gathering methodology; processed, filtered and cleansed. The quality of these data sets is a key point that must be guaranteed.

The data-driven assignment-free DODME proposal of Chapter 6 is a methodology that uses the current commercial GPS data, with the already mentioned problems. In this sense, the assumptions that are taken about the quality and certainty of the data set are very few. One supplementary line of research is to analyze in detail the quality of the GPS data and how to measure it. If the quality of the data becomes higher, then, we can then use the information to build a more sophisticated observed OD matrix, including specific scaling factors for each OD pair, that can be used also as a seed OD matrix for the data-driven assignment-free OD estimation approach.

Another enhancement that must be studied in the future is the addition of more complex constraints to the data-driven proposal. At this moment, the unique constraint is to maintain the positivity of the OD flows with respect to the seed OD matrix. In SPSA, we also added bounding constraints for each OD flow that permit to control of the total number of trips. As the non-linear optimization method used, the L-BFGS-B, permits to add other constraints, one immediate step is to add such constraints to the data-driven formulation. In the future, other different constraints, related to the production and attraction based on land-use models, can also be considered.

Last but not least, it would be of interest to build a bi-level approach for the data-driven approach where at the upper level the OD flows are optimized, based on a lower level optimization of the scaling factors despite this approach is more demanding on the computational point of view.

# Notation

## Sets

I       Set of origin zones

J       Set of destination zones

N       Set of OD pairs ($N = I \times J$)

L       Set of links of the network

$\hat{L}$       Set of equipped links of the network, so $\hat{L} \subseteq L$

$\mathcal{T}$       Set of time periods of the simulation

$\mathcal{P}_{ijr}$       Set of paths for a certain OD pair $(i, j)$ departing at certain time interval $r$

## Objects

$\mathbf{X}$       is an OD matrix (and can be represented as a vector of length $|I| \cdot |J| \cdot |T|$), that is $\mathbf{X} = \left[ x_{ijr} \right]$.

$x_{ijr}$       is an OD flow for $(i, j)$ OD pair departing at Time Interval $r$.

$\mathbf{Y}$       is a vector of traffic counts on $\hat{L}$ over time, that is $\mathbf{Y} = [y_{lt}]$.

$y_{lt}$       is a traffic count measurement on link $l \in \hat{L}$ at Time Interval $t$.

$\mathbf{A}$       is the Assignment Matrix, that is $\mathbf{A} = \left[ a_{ijr}^{lt} \right]$.

$a_{ijr}^{lt}$       is the proportion of $x_{ijr}$ that uses link $l$ at time $t$.

## Subindexes

i       stands for Origins

j       stands for Destinations

n       stands for OD pairs, normally $n \equiv (i, j)$

| l | stands for Links |
|---|---|
| r | stands for departing time period (from an origin) |
| t | stands for arriving time interval (to a link) |
| p | stands for a path of the set $\mathcal{P}_{ijr}$. In fact, $p = p(i, j, r)$ |

## Superindexes

| H | means Historical, reference. |
|---|---|
| 0 | means seed for an optimization problem |
| * | means obtained as a solution of an optimization problem |
| (k) | means at iteration $k$ |

## Accents

| $\wedge$ | means measured or observed measure |
|---|---|
| $-$ | means estimated after a suitable process of a traffic measurement |

# List of Figures

# List of Tables

# List of Algorithms

# References

Aimsun (2017), *Aimsun Next 8.2. User's Manual*, Barcelona, Spain.

Alexander, L., Jiang, S., Murga, M. & González, M. (2015), 'Origin-destination trips by purpose and time of day inferred from mobile phone data', *Transportation Research Part C: Emerging Technologies* **58**(B), 240–250.

Antoniou, C., Azevedo, C. L., Lu, L., Pereira, F. & Ben-Akiva, M. (2015), 'W-SPSA in Practice: Approximation of Weight Matrices and Calibration of Traffic Simulation Models', *Transportation Research Procedia* **59**, 129–146.

Antoniou, C., Barceló, J., Breen, M., Bullejos, M., Casas, J., Cipriani, E., Ciuffo, B., Djukic, T., Hoogendoorn, S. P., Marzano, V., Montero, L., Nigro, M., Perarnau, J., Punzo, V., Toledo, T. & van Lint, H. (2016), 'Towards a generic benchmarking platform for origin-destination flows estimation/updating algorithms: Design, demonstration and validation', *Transportation Research Part C: Emerging Technologies* **66**, 79–98.

Antoniou, C., Ben-Akiva, M. & Koutsopoulos, H. (2004), 'Incorporating automated vehicle identification data into origin-destination estimation', *Transportation Research Record: Journal of the Transportation Research Board* **1882**(1), 37–44.

Antoniou, C., Dimitriou, L. & Pereira, F. (2019), *Mobility Patterns, Big Data and Transport Analytics Tools and Applications for Modeling*, Elsevier.

Ashok, K. (1996), Dynamic origin-destination matrix estimation and prediction, PhD thesis, Massachusetts Institute of Technology (MIT), Boston, United States of America.

Ashok, K. & Ben-Akiva, M. (1993), 'Dynamic Origin-Destination Matrix Estimation for Prediction for Real-time Traffic Management Systems', *International Symposium on the Theory of Traffic Flow and Transportation* pp. 465–484.

Ashok, K. & Ben-Akiva, M. (2002), 'Estimation and Prediction of Time-Dependent Origin-Destination Flows with a Stochastic Mapping to Path Flows and Link Flows', *Transportation Science* **36**(2), 184–198.

Balakrishna, R. (2006), Off-line Calibration of Dynamic Traffic Assignment Models, PhD thesis, Massachusetts Institute of Technology (MIT), Boston, United States of America.

Barceló, J., Gilliéron, F., Linares, M., Serch, O. & Montero, L. (2012), 'Exploring Link Covering and Node Covering Formulations of Detection Layout Problem', *Transportation Research Record: Journal of the Transportation Research Board* **2308**, 17–26.

Barceló, J., Montero, L., Bullejos, M., Serch, O. & Carmona, C. (2013), 'A kalman filter approach for exploiting bluetooth traffic data when estimating time-dependent od matrices', *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations* **17**(2), 123–141.

Behara, K. N. S. (2019), Origin-Destination matrix estimation using big traffic data: a structural perspective, PhD thesis, School of Civil Engineering and Built Environment Science and Engineering Faculty Queensland University of Technology.

Behara, K. N. S., Bhaskar, A. & Chung, E. (2018), 'Classification of typical Bluetooth OD matrices based on structural similarity of travel patterns: Case study on Brisbane city', *Proceedings of the Annual Meeting The Transportation Research Board (TRB) 97th Annual Meeting* .

Behara, K. N. S., Bhaskar, A. & Chung, E. (2020*a*), 'A novel approach for the structural comparison of origin-destination matrices: Levenshtein distance', *Transportation Research Part C: Emerging Technologies* **111**.

Behara, K. N. S., Bhaskar, A. & Chung, E. (2020*b*), 'A Novel Methodology to Assimilate Sub-Path Flows in Bi-Level OD Matrix Estimation Process', *IEEE Transactions on Intelligent Transportation Systems* pp. 1–11.

Ben-Akiva, M. & Bierlaire, M. (1999), Discrete Choice Methods and their Applications to Short Term Travel Decisions, *in* 'Handbook of Transportation Science', pp. 5–33.

Ben-Akiva, M., Bierlaire, M., Burton, D., Koutsopoulos, H. & Mishalani, R. (2001), 'Network State Estimation and Prediction for Real-Time Traffic Management', *Networks and Spatial Economics* **1**(3-4), 293–318.

Bierlaire, M. (2002), 'The total demand scale: A new measure of quality for static and dynamic origin-destination trip tables', *Transportation Research Part B: Methodological* **36**(9), 837–850.

Bierlaire, M. (2015), 'Simulation and optimization: A short review', *Transportation Research Part C: Emerging Technologies* **55**, 4–13.

Bierlaire, M. & Crittin, F. (2004), 'An efficient algorithm for real-time estimation and prediction of dynamic od tables', *Operations Research* **52**(1), 116–127.

Bovy, P. H. L., Bekhor, S. & Prato, C. G. (2008), 'The Factor of Revisited Path Size', *Transportation Research Record: Journal of the Transportation Research Board* **2076**(1), 132–140.

Brunet, D., Vrscay, E. R. & Wang, Z. (2012), 'On the mathematical properties of the structural similarity index', *IEEE Transactions on Image Processing* **21**(4), 1488–1495.

Bullejos, M., Barceló, J. & Montero, L. (2014), A DUE based bilevel optimization approach for the estimation of time sliced OD matrices, *in* 'International Symposium of Transport Simulation 2014'.

Caceres, N., Romero, L. M. & Benitez, F. G. (2013), 'Inferring origin-destination trip matrices from aggregate volumes on groups of links: a case study using volumes inferred from mobile phone data', *Journal of Advanced Transportation* **47**(7), 650–666.
**URL:** *http://doi.wiley.com/10.1002/atr.187*

Calabrese, F., Di Lorenzo, G. & Liu, L. (2011), 'Estimating origin-destination flows using mobile phone location data', *IEEE Pervasive Computation* **10**, 36–44.

Cantelmo, G. (2018), Dynamic Origin-Destination Matrix Estimation with Interacting Demand Patterns, PhD thesis, Université du Luxembourg, Luxembourg, Luxembourg.

Cantelmo, G., Cipriani, E., Gemma, A. & Nigro, M. (2014), 'An adaptive Bi-level gradient procedure for the estimation of dynamic traffic demand', *IEEE Transactions on Intelligent Transportation Systems* **15**(3), 1348–1361.

Cantelmo, G., Viti, F., Tampère, C. M. J., Cipriani, E. & Nigro, M. (2014), 'Two-Step Approach for Correction of Seed Matrix in Dynamic Demand Estimation', *Transportation Research Record: Journal of the Transportation Research Board* **2466**, 125–133.

Carrese, S., Cipriani, E., Mannini, L. & Nigro, M. (2017), 'Dynamic demand estimation and prediction for traffic urban networks adopting new data sources', *Transportation Research Part C: Emerging Technologies* **81**, 83–98.

Cascetta, E. (1984), 'Estimation of trip matrices from traffic counts and survey data: A generalized least squares estimator', *Transportation Research Part B: Methodological* **18**(4-5), 289–299.

Cascetta, E. (2001), *Transportation Systems Analysis*, New York, United States of America.

Cascetta, E. & Nguyen, S. (1988), 'A unified framework for estimating or updating origin/destination matrices form traffic counts', *Transportation Research Part B: Methodological* **22**(6), 437–455.

Cascetta, E., Nuzzolo, A., Russo, F. & Vitetta, A. (1996), 'A modified Logit route choice model overcoming path overlapping problems. Specification and some calibration results for interurban networks', *Proceedings of the 13th International Symposium on the Theory of Road Traffic Flow* .

Cascetta, E., Papola, A., Marzano, V., Simonelli, F. & Vitiello, I. (2013), 'Quasi-dynamic estimation of o–d flows from traffic counts: Formulation, statistical validation and performance analysis on real data', *Transportation Research Part B: Methodological* **55**, 171–187.

Castillo, E., Jimenez, P., Menéndez, J. M. & Conejo, A. J. (2008), 'The Observability Problem in Traffic Models: Algebraic and Topological Methods', *IEEE Transactions on Intelligent Transportation Systems* **9**(2), 275–287.

Chabini, I. (1998), 'Discrete Dynamic Shortest Path Problems in Transportation Applications: Complexity and Algorithms with Optimal Run Time', *Transportation Research Record: Journal of the Transportation Research Board* **1645**(1), 170–175.

Chen, C., Gong, H., Lawson, C. & Bialostozky, E. (2010), 'Evaluating the feasibility of a passive travel survey collection in a complex urban environment: Lessons learned from the New York City case study', *Transportation Research Part A: Policy and Practice* **44**(10), 830–840.

Chen, C., Ma, J., Susilo, Y., Liu, Y. & Wang, M. (2016), 'The promises of big data and small data for travel behavior (aka human mobility) analysis', *Transportation Research Part C: Emerging Technologies* **68**, 285–299.

Cipriani, E., Florian, M., Mahut, M. & Nigro, M. (2011), 'A gradient approximation approach for adjusting temporal origin-destination matrices', *Transportation Research Part C: Emerging Technologies* **19**(2), 270–282.

Codina, E. & Barceló, J. (2004), 'Adjustment of O–D trip matrices from observed volumes: An algorithmic approach based on conjugate directions', *European Journal of Operational Research* **155**(3), 535–557.

Codina, E. & Montero, L. (2006), Approximation of the steepest descent direction for the O-D matrix adjustment problem, *in* 'Annals of Operations Research', Vol. 144, pp. 329–362.

Daamen, W., Buisson, C. & Hoogendoorn, S. P. (2014), *Traffic Simulation and Data: Validation Methods and Applications*, CRC Press, Boca Raton, Florida, United States of America.

Djukic, T. (2014), Dynamic OD demand estimation and prediction for dynamic traffic management, PhD thesis, TU Delft, Delft, The Netherlands.

Djukic, T., Lint, H. V. & Hoogendoorn, S. P. (2013), 'Incorporating the structural information into the estimation of OD matrices', *Triennial symposium on transportation analysis* (June), 1–5.

Djukic, T., Masip, D., Breen, M. & Perarnau, J. (2018), 'Heuristic-based framework for dynamic OD demand estimation in the congested networks', *Transportation Research Board- 97th Annual Meeting, 2018 - Washington D.C.* .

Doblas, J. & Benitez, F. G. (2005), 'An approach to estimating and updating origin–destination matrices based upon traffic counts preserving the prior structure of a survey matrix', *Transportation Research Part B: Methodological* **39**(7), 565–591.

Eisenman, S. & List, G. (2004), Using probe data to estimate OD matrices, *in* 'Proceedings. The 7th International IEEE Conference on Intelligent Transportation Systems (IEEE Cat. No.04TH8749)', IEEE, pp. 291–296.

Florian, M. & Chen, Y. (1995), 'A coordinate descent method for the bi-level O-D matrix adjustment problem', *International Transactions in Operational Research* **2**(2), 165–179.

Florian, M. & Hearn, D. (1995), 'Network Equilibrium Models and Algorithms', *Handbooks in Operations Research and Management Science* **8**, 485–550.

Frederix, R., Viti, F., Corthout, R. & Tampère, C. M. J. (2011), 'New Gradient Approximation Method for Dynamic Origin-Destination Matrix Estimation on Congested Networks', *Transportation Research Record: Journal of the Transportation Research Board* **2263**(1), 19–25.

Frederix, R., Viti, F. & Tampère, C. M. J. (2010), 'A density-based dynamic OD estimation method that reproduces within-day congestion dynamics', *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC* (1), 694–699.

Frederix, R., Viti, F. & Tampère, C. M. J. (2013), 'Dynamic origin-destination estimation in congested networks: Theoretical findings and implications in practice', *Transportmetrica A: Transport Science* **9**(6), 494–513.

Friesz, T. L., Bernstein, D., Smith, T. E., Tobin, R. L. & Wie, B. W. (1993), 'Variational inequality formulation of the dynamic network user equilibrium problem', *Operations Research* **41**(1), 179–191.

Guerrero-Ibáñez, J., Zeadally, S. & Contreras-Castillo, J. (2018), 'Sensor Technologies for Intelligent Transportation Systems', *Sensors* **18**(4), 1212.

Gundlegård, D., Rydergren, C., Barceló, J. & Dokoohaki, N. (2013), 'Travel demand analysis with differentially private releases', *D4D Challenge Senegal 2014 Netmob 2015, November 2015, MIT, Boston* .

Hollander, Y. & Liu, R. (2008), 'The principles of calibrating traffic microsimulation models', *Transportation* **35**, 347–362.

Huang, L., Li, Q. & Yue, Y. (2010), Activity identification from GPS trajectories using spatial temporal POIs' attractiveness, *in* 'Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Location Based Social Networks - LBSN '10', ACM Press, New York, New York, USA, p. 27.

Huyer, W. & Neumaier, A. (2008), 'SNOBFIT - Stable Noisy Optimization by Branch and Fit', *Transactions on Mathematical Software* **35**(2).

Iqbal, M. S., Choudhury, C., Wang, P. & González, M. (2014), 'Development of origin–destination matrices using mobile phone call data', *Transportation Research Part C: Emerging Technologies* **40**, 63–74.

Janmyr, J. & Wadell, D. (2018), *Analysis of vehicle route choice during incidents*, Linköping, Sweden.

Jiang, S., Yang, Y., Gupta, S., Veneziano, D., Athavale, S. & González, M. (2016), 'The TimeGeo modeling framework for urban mobility without travel surveys', *Proceedings of the National Academy of Sciences* **113**(37), E5370–E5378.

Kim, K. & Rilett, L. R. (2004), 'A genetic algorithm based approach to traffic micro-simulation calibration using ITS data', *83rd TRB annual meeting, Washington D.C.* .

Kostic, B., Gentile, G. & Antoniou, C. (2015), An Aggregate Approach for the Calibration of Time-Dependent Demand in Dynamic Traffic Assignment Models using SPSA Algorithm, *in* '4th Symposium of the European Association for Research in Transportation (hEART 2015)', pp. 1–1.

Kostic, B., Meschini, L. & Gentile, G. (2017), 'Calibration of the demand structure for dynamic traffic assignment using flow and speed data: Exploiting the advantage of distributed computing in derivative-free optimization algorithms', *Transportation Research Procedia* **27**, 993–1000.

Krishnakumari, P., van Lint, H., Djukic, T. & Cats, O. (2020), 'A data driven method for OD matrix estimation', *Transportation Research Part C: Emerging Technologies* **113**, 38–56.

Kubicka, M., Cela, A., Mounier, H. & Niculescu, S.-I. (2018), 'Comparative Study and Application-Oriented Classification of Vehicular Map-Matching Methods', *IEEE Intelligent Transportation Systems Magazine* **10**(2), 150–166.

Lin, P.-W. & Chang, G.-L. (2007), 'A generalized model and solution algorithm for estimation of the dynamic freeway origin-destination matrix', *Transportation Research Part B: Methodological* **41**(5), 554–572.

Lopez, C., Krishnakumari, P., Leclercq, L., Chiabaut, N. & van Lint, H. (2017), 'Spatiotemporal Partitioning of Transportation Network Using Travel Time Data', *Transportation Research Record: Journal of the Transportation Research Board* **2623**(1), 98–107.

Lopez, C., Leclercq, L., Krishnakumari, P., Chiabaut, N. & van Lint, H. (2017), 'Revealing the day-to-day regularity of urban congestion patterns with 3D speed maps', *Scientific Reports* **7**(1), 14029.

Lu, L., Xu, Y., Antoniou, C. & Ben-Akiva, M. (2015), 'An enhanced SPSA algorithm for the calibration of Dynamic Traffic Assignment models', *Transportation Research Part C: Emerging Technologies* **51**, 149–166.

Luenberger, D. G. & Ye, Y. (1973), *Linear and nonlinear programming*, Vol. 228.

Lundgren, J. T. & Peterson, A. (2008), 'A heuristic for the bilevel origin-destination-matrix estimation problem', *Transportation Research Part B: Methodological* **42**(4), 339–354.

Ma, J., Yuan, F., Joshi, C., Li, H. & Bauer, T. (2012), 'A New Framework for Development of Time Varying OD Matrices Based on Cellular Phone Data', *4th TRB Innovations in Travel Modeling (ITM) Conference, Tampa, Florida, USA* .

Ma, J., Zhang, M. & Dong, H. (2006), 'Calibration of departure time and route choice parameters in micro-simulation with macro measurements and genetic algorithm', *85th TRB annual meeting, paper, Washington D.C.* **2376**.

Ma, T. & Abdulhai, B. (2002), 'A genetic algorithm-based optimization approach and generic tool for calibrating traffic microscopic simulation parameters', *Transportation Research Record* **1806**, 6–15.

Mahmassani, H. (2001), 'Dynamic Network Traffic Assignment and Simulation Methodology for Advanced System Management Applications', *Networks and Spatial Economics* **1**(3/4), 267–292.

Mahut, M. (2000), A discrete flow model for dynamic network loading, PhD thesis, Université de Montréal, Montréal, Canada.

Mitra, A., Attanasi, A., Meschini, L. & Gentile, G. (2020), 'Methodology for O-D matrix estimation using the revealed paths of floating car data on large-scale networks', *IET Intelligent Transport Systems* **14**(12), 1704–1711.

Mo, B., Li, R. & Dai, J. (2020), 'Estimating dynamic origin-destination demand: A hybrid framework using license plate recognition', *Computer-Aided Civil and Infrastructure Engineering* **35**(7), 734–752.

Montero, L. & Ros-Roca, X. (2020), 'Using GPS tracking data to validate route choice in OD trips within dense urban networks', *Transportation Research Procedia* **47**, 593–600.

Montero, L., Ros-Roca, X., Herranz, R. & Barceló, J. (2019), 'Fusing mobile phone data with other data sources to generate input OD matrices for transport models', *Transportation Research Procedia* **37**, 417–424.

Morales, J. L. & Nocedal, J. (2011), 'Remark on "algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound constrained optimization"', *ACM Transactions on Mathematical Software* **38**(1), 1–4.

Nanthawichit, C., Nakatsuji, T. & Suzuki, H. (2003), 'Application of Probe-Vehicle Data for Real-Time Traffic-State Estimation and Short-Term Travel-Time Prediction on a Freeway', *Transporation Research Record Journal of the TRB* **1855**(1).

Nassir, N., Ziebarth, J., Sall, E. & Zorn, L. (2014), 'Choice Set Generation Algorithm Suitable for Measuring Route Choice Accessibility', *Transportation Research Record: Journal of the Transportation Research Board* **2430**(1), 170–181.

Nelder, J. A. & Mead, R. (1965), 'A simplex method for function minimization', *The Computer Journal* **7**(5), 308–313.

Nigro, M., Abdelfatah, A., Cipriani, E., Colombaroni, C., Fusco, G. & Gemma, A. (2018), 'Dynamic O-D demand estimation: Application of SPSA AD-PI method in conjunction with different assignment strategies', *Journal of Advanced Transportation* **2018**.

OpenLR (2020), 'OpenLR White Paper. Version 1.5, revision 2'.
**URL:** *https://www.openlr-association.com/fileadmin/user_upload/openlr-whitepaper_v1.5.pdf*

Ortúzar, J. d. D. & Willumsen, L. G. (2011), *Modelling Transport*.

Osorio, C. (2019), 'Dynamic origin-destination matrix calibration for large-scale network simulators', *Transportation Research Part C: Emerging Technologies* **98**(April 2018), 186–206.

Osorio, C. & Chong, L. (2015), 'A Computationally Efficient Simulation-Based Optimization Algorithm for Large-Scale Urban Transportation Problems', *Transportation Science* **49**(3), 623–636.

Patriksson, M. (1994), *The Traffic Assignment Problem: Models and Methods*, Nörkoping, Sweden.

Phithakkitnukoon, S., Horanont, T., Di Lorenzo, G., Shibasaki, R. & Ratti, C. (2010), 'Activity-Aware Map: Identifying Human Daily Activity Pattern Using Mobile Phone Data', *International Workshop on Human Behavior Understanding* .

PTV AG (2020), *PTV Visum - User's manual*, Karlsruhe, Germany.

Ran, B. & Boyce, D. (1996), *Modeling Dynamic Transportation Networks*, Springer Vorlag.

Ros-Roca, X. (2017), *Analyzing SPSA approaches to solve the non-linear non-differentiable problems arising in the assisted calibration of traffic simulation models*.
**URL:** *http://hdl.handle.net/2117/100675*

Ros-Roca, X., Montero, L. & Barceló, J. (2017), 'Notes on Using Simulation-Optimization Techniques in Traffic Simulation', *Transportation Research Procedia* **27**, 881–888.

Ruiz de Villa, A., Casas, J. & Breen, M. (2014), 'OD matrix structural similarity: Wasserstein metric', *TRB 93rd Annual Meeting Compendium of Papers* .

Sadegh, P. (1997), 'Constrained Optimization via Stochastic Approximation with a Simultaneous Perturbation Gradient Approximation', *Automatica* **33**(5), 332–341.

Sheffi, Y. (1985), *Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods*, Prentice Hall.

Smith, M. J. (1993), 'A new dynamic traffic model and the existence and calculation of dynamic user equilibria on congested capacity-constrained road networks', *Transportation Research Part B: Methodological* **27**(1), 49–63.

Spall, J. C. (1992), 'Multivariate Stochastic Approximation Using a Simultaneous Perturbation Gradient Approximation', *IEEE Transactions on Automatic Control* **37**(3), 332–341.

Spall, J. C. (1998), 'An Overview of the Simultaneous Perturbation Method for Efficient Optimization', *Johns Hopkins Apl Technical Digest* **19**(4), 482–492.

Spall, J. C. (2003), *Introduction to Stochastic Search and Optimization: Estimation, Simulation and Control*, Wiley-Interscience.

Spiess, H. (1990), 'A gradient approach for the O-D matrix adjustment problem', *Centre for research on transportation, University of Montreal, Canada* **693**, 1–11.

Tavana, H. (2001), Internally-Consistent Estimation of Dynamic Network Origin-Destination Flows from Intelligent Transportation Systems Data using Bi-Level Optimization, PhD thesis, University of Texas, Austin, United States of America.

Toledo, T. & Kolechkina, T. (2013), 'Estimation of dynamic origin-destination matrices using linear assignment matrix approximations', *IEEE Transactions on Intelligent Transportation Systems* **14**(2), 618–626.

Tympakianaki, A. (2018), Demand Estimation and Bottleneck Management Using Heterogeneous Traffic Data, PhD thesis, KTH Royal Institute of Technology, Stockholm, Sweden.

Tympakianaki, A., Koutsopoulos, H. & Jenelius, E. (2015), 'c-SPSA: Cluster-wise simultaneous perturbation stochastic approximation algorithm and its application to dynamic origin-destination matrix estimation', *Transportation Research Part C: Emerging Technologies* **55**, 231–245.

Tympakianaki, A., Koutsopoulos, H. & Jenelius, E. (2018), 'Robust SPSA algorithms for dynamic OD matrix estimation', *Procedia Computer Science* **130**, 57–64.

Van Aerde, M., Hellinga, B., Yu, L. & Rakha, H. (1993), 'Vehicle Probes as Real-Time ATMS Sources of Dynamic O-D and Travel Time Data', *Proc. of the Advanced Traffic Management Conference, FHWA* pp. 207–230.

Van Zuylen, H. J. & Willumsen, L. G. (1980), 'The most likely trip matrix estimated from traffic counts', *Transportation Research Part B: Methodological* **14**(3), 281–293.

Wang, H., Calabrese, F., Di Lorenzo, G. & Ratti, C. (2010), Transportation mode inference from anonymized and aggregated mobile phone call detail records, *in* '13th International IEEE Conference on Intelligent Transportation Systems', IEEE, pp. 318–323.

Wang, I.-J. J. & Spall, J. C. (1999), A constrained simultaneous perturbation stochastic approximation algorithm based on penalty functions, *in* 'Proceedings of the 1999 American Control Conference (Cat. No. 99CH36251)', IEEE, pp. 393–399 vol.1.

Wang, Z., Bovik, A. C., Sheikh, H. R. & Simoncelli, E. P. (2004), 'Image quality assessment: From error visibility to structural similarity', *IEEE Transactions on Image Processing* **13**(4), 600–612.

Wang, Z. & Simoncelli, E. P. (2008), 'Maximum differentiation (MAD) competition: A methodology for comparing computational models of perceptual quantities', *Journal of Vision* **8**(12), 1–13.

Wardrop, J. G. (1952), 'Some theoretical aspects of road traffic research communication networks', *ICE Proceedings: Engineering Divisions* **2**, 325–362.

Wu, J., Chen, Y. & Florian, M. (1998), 'The continuous dynamic network loading problem: a mathematical formulation and solution method', *Transportation Research Part B: Methodological* **32**(3), 173–187.

Wu, J. H. (1991), A Study of Monotone Variational Inequalities and their Application to Network Equilibrium Problems, PhD thesis, Université de Montréal, Montréal, Canada.

Wu, J. H. (1998), 'A projection algorithm for the dynamic network equilibrium problem', *Traffic and Transportation Studies, Proceedings of the ICTTS'98* pp. 379–390.

Xie, K., Deng, K. & Zhou, X. (2009), 'From trajectories to activities: a spatio-temporal join approach', *LBSN '09: Proceedings of the 2009 International Workshop on Location Based Social Networks* pp. 25–32.

Yang, H., Sasaki, T., Iida, Y. & Asakura, Y. (1992), 'Estimation of origin-destination matrices from link traffic counts on congested networks', *Transportation Research Part B: Methodological* **26**(6), 417–434.

Yang, X., Lu, Y. & Hao, W. (2017), 'Origin-Destination Estimation Using Probe Vehicle Trajectory and Link Counts', *Journal of Advanced Transportation* **2017**(1), 1–18.

# Appendices

# A
# Urban networks used

## A.1 Fictitious network

The fictitious network is provided by PTV Group[1], built in Visum and Vissim and calibrated to be used with the Simulation-based Assignment (SBA), the mesoscopic dynamic traffic assignment available in Visum.

Table A.1: Fictitious network and OD characteristics

| | |
|---|---|
| Time periods | 4 |
| Zones | 114 |
| Detectors | 40 |
| OD pairs X Time | 51984 |
| Ground Truth Trips | 8300 |
| Ground Truth | 40802 |
| Positive OD | (78.49%) |
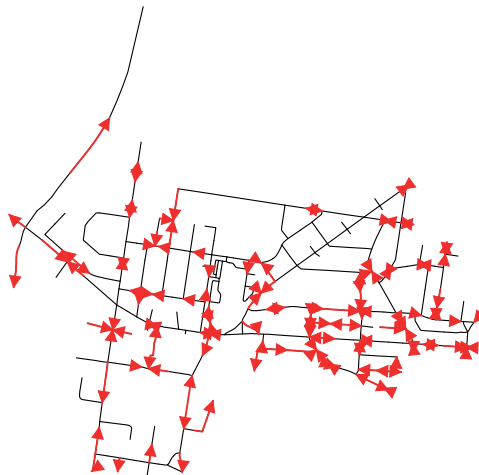


Figure A.1: The fictitious network

---

[1]www.company.ptvgroup.com/en

## A.2   Hillsboro (Oregon, United States)

The Hillsboro network is provided by PTV Group, built in Visum and Vissim and calibrated to be used with the Simulation-based Assignment (SBA), the mesoscopic dynamic traffic assignment available in Visum.

Table A.2:  Hillsboro network and OD characteristics

| | |
|---|---|
| Time periods | 3 |
| Zones | 58 |
| Detectors | 80 |
| OD pairs X Time | 10092 |
| Ground Truth Trips | 9878 |
| Ground Truth | 8100 |
| Positive OD | (80.26%) |
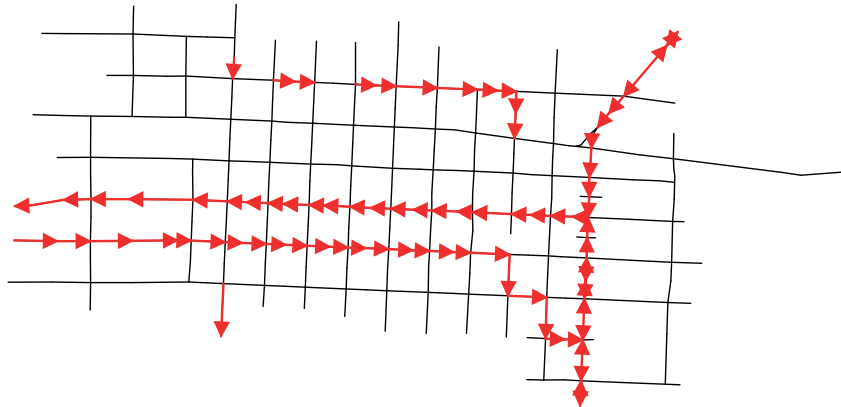


Figure A.2:  Network of Hillsboro

## A.3   Turin (Piemonte, Italy)

The Turin network is provided by PTV Sistema (part of PTV Group), built in Visum and calibrated to be used with the Traffic Realtime Equilibrium (TRE) Assignment[2].

Table A.3: Turin network and OD and GPS data characteristics

| | |
|---|---|
| Time periods | 4 |
| Zones | 221 |
| Detectors | 302 |
| OD pairs X Time | 195364 |
| Historical Trips | 129990 |
| Historical | 77579 |
| Positive OD | (39.71%) |
| GPS trips | 232418 |
| GPS waypoints | 3758445 |



Figure A.3: Network of Turin

---

[2]Traffic Realtime Equilibrium (TRE)

## A.4   Barcelona (Catalonia, Spain)

The Barcelona Visum model comes from the Virtual Mobility Lab, a CARNET project[3]. It is calibrated to be used with the Simulation-based Assignment (SBA), the mesoscopic dynamic traffic assignment of Visum.

Table A.4: Barcelona network and OD and GPS data characteristics

| | |
|---|---|
| Time periods | 4 |
| Zones | 202 |
| Detectors | 70 |
| OD pairs X Time | 163216 |
| Historical Trips | 60083 |
| Historical | 43693 |
| Positive OD | (26.77%) |
| GPS trips | 19974 |
| GPS waypoints | 566359 |



Figure A.4: Network of Barcelona

*Acknowledgements*: The Virtual Mobility Lab project and CARNET.

# B
# Full results of the Case Studies in this thesis

## B.1 Results of SPSA without Travel Times

Table B.1: Initial values of the KPIs

| KPI | Chaos | Chaos+Inc$^-$ | Chaos+Inc$^+$ | Inc$^-$ | Inc$^+$ | Multitude |
|---|---|---|---|---|---|---|
| $R_0^2$ | 0.3149 | 0.2775 | 0.3453 | 0.6226 | 0.8278 | 0.5345 |
| NT | 9878 | 7408 | 12347 | 7408 | 12347 | 7408 |
| $MSSIM_0$ | 0.5538 | 0.6301 | 0.4684 | 0.9808 | 0.9898 | 0.9769 |

Since the experiments were launched for 400 iterations without a stopping criterion, the final indicators show the values at the iteration when $R^2$ reached the maximum value.

Table B.2: Final $R^2$ between traffic counts

| Method | $w_2$ | CG | Chaos | Chaos+Inc$^-$ | Chaos+Inc$^+$ | Inc$^-$ | Inc$^+$ | Multitude |
|---|---|---|---|---|---|---|---|---|
| Free SPSA | $w_2 = 0$ | NO | 0.3229 | 0.3197 | 0.3673 | 0.8303 | 0.7832 | 0.7655 |
| | | YES | 0.3208 | 0.3017 | 0.3682 | 0.7776 | 0.8541 | 0.7784 |
| | $w_2 = 1$ | NO | 0.3630 | 0.3005 | 0.3445 | 0.7471 | 0.8640 | 0.8110 |
| | | YES | 0.3221 | 0.2949 | 0.3459 | 0.8130 | 0.8631 | 0.7177 |
| Constrained SPSA | $w_2 = 0$ | NO | 0.3292 | 0.2760 | 0.3428 | 0.7486 | 0.8423 | 0.7704 |
| | | YES | 0.3175 | 0.3046 | 0.3604 | 0.7428 | 0.8351 | 0.7395 |
| | $w_2 = 1$ | NO | 0.3134 | 0.2928 | 0.3642 | 0.7346 | 0.8702 | 0.7447 |
| | | YES | 0.3122 | 0.3123 | 0.3616 | 0.7221 | 0.8693 | 0.8016 |
| Penalized SPSA | $w_2 = 0$ | NO | 0.3202 | 0.3061 | 0.3667 | 0.8287 | 0.8242 | 0.7759 |
| | | YES | 0.3358 | 0.2758 | 0.3538 | 0.7852 | 0.8374 | 0.7564 |
| | $w_2 = 1$ | NO | 0.3326 | 0.2675 | 0.3489 | 0.8069 | 0.8614 | 0.7822 |
| | | YES | 0.3071 | 0.2836 | 0.3647 | 0.7682 | 0.8216 | 0.8350 |

Table B.3: Final total number of trips, NT

| Method | $w_2$ | CG | Chaos | Chaos+Inc$^-$ | Chaos+Inc$^+$ | Inc$^-$ | Inc$^+$ | Multitude |
|---|---|---|---|---|---|---|---|---|
| Free | $w_2 = 0$ | NO | 9891 | 7420 | 12307 | 7869 | 12314 | 7638 |
| | | YES | 9873 | 7430 | 12273 | 7520 | 12110 | 7502 |
| SPSA | $w_2 = 1$ | NO | 10722 | 7589 | 12289 | 8325 | 12108 | 7819 |
| | | YES | 9826 | 7384 | 12365 | 8149 | 11329 | 7521 |
| Constrained | $w_2 = 0$ | NO | 9852 | 7410 | 12302 | 7597 | 11779 | 7561 |
| | | YES | 9916 | 7436 | 12314 | 7448 | 12166 | 7454 |
| SPSA | $w_2 = 1$ | NO | 9877 | 7412 | 12300 | 7603 | 12013 | 7601 |
| | | YES | 9853 | 7433 | 12354 | 7422 | 11877 | 7593 |
| Penalized | $w_2 = 0$ | NO | 9896 | 7476 | 12294 | 7679 | 11929 | 7535 |
| | | YES | 9980 | 7402 | 12247 | 7593 | 12341 | 7531 |
| SPSA | $w_2 = 1$ | NO | 10336 | 7847 | 12311 | 7805 | 11399 | 7568 |
| | | YES | 9952 | 7407 | 12322 | 7905 | 11853 | 7716 |

Table B.4: Final MSSIM between $\mathbf{X}^*$ and $\mathbf{X}^{GT}$

| Method | $w_2$ | CG | Chaos | Chaos+Inc$^-$ | Chaos+Inc$^+$ | Inc$^-$ | Inc$^+$ | Multitude |
|---|---|---|---|---|---|---|---|---|
| Free | $w_2 = 0$ | NO | 0.2492 | 0.2758 | 0.2084 | 0.8844 | 0.9386 | 0.9116 |
| | | YES | 0.2368 | 0.2744 | 0.2174 | 0.9011 | 0.9411 | 0.9024 |
| SPSA | $w_2 = 1$ | NO | 0.2008 | 0.2430 | 0.2228 | 0.7859 | 0.9310 | 0.8801 |
| | | YES | 0.2475 | 0.2728 | 0.2176 | 0.8305 | 0.8680 | 0.8984 |
| Constrained | $w_2 = 0$ | NO | 0.2553 | 0.2779 | 0.2292 | 0.9145 | 0.9346 | 0.9227 |
| | | YES | 0.2534 | 0.2801 | 0.2204 | 0.9175 | 0.9474 | 0.9126 |
| SPSA | $w_2 = 1$ | NO | 0.2553 | 0.2725 | 0.2183 | 0.9166 | 0.9468 | 0.9198 |
| | | YES | 0.2549 | 0.2786 | 0.2189 | 0.9101 | 0.9477 | 0.9169 |
| Penalized | $w_2 = 0$ | NO | 0.2533 | 0.2619 | 0.2198 | 0.9014 | 0.9186 | 0.9130 |
| | | YES | 0.2372 | 0.2803 | 0.2184 | 0.9024 | 0.9520 | 0.8976 |
| SPSA | $w_2 = 1$ | NO | 0.2031 | 0.2336 | 0.2011 | 0.9066 | 0.8940 | 0.9080 |
| | | YES | 0.2417 | 0.2790 | 0.2201 | 0.7988 | 0.8794 | 0.8849 |
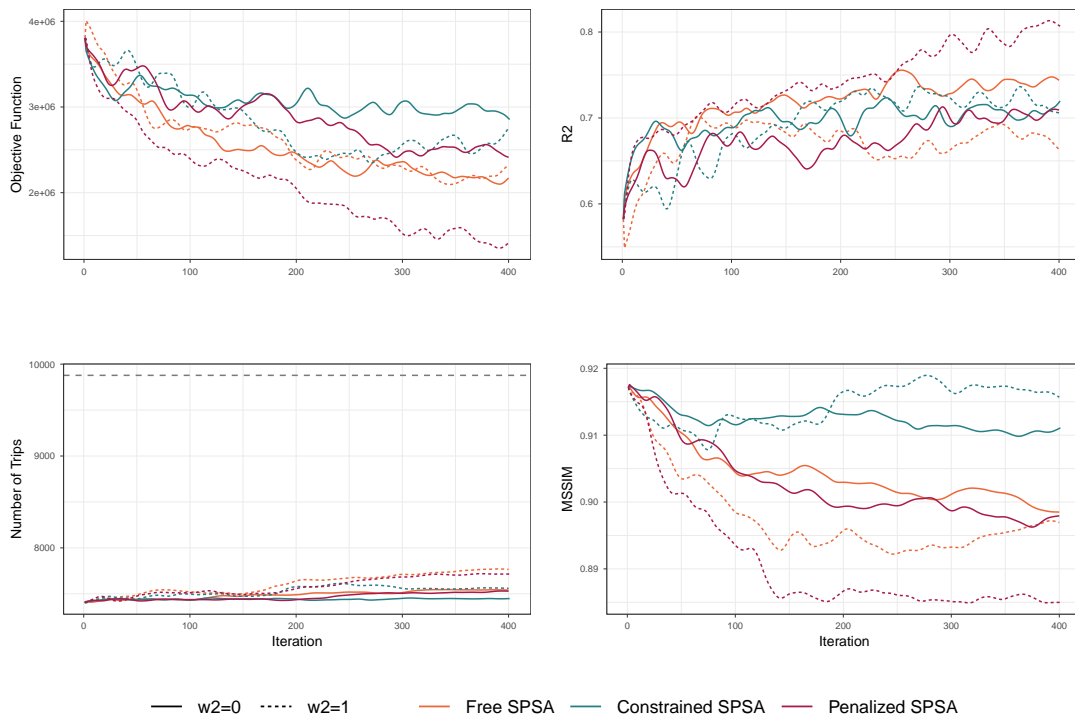
Figure B.1: Multitude initial matrix, with YES conjugate gradient
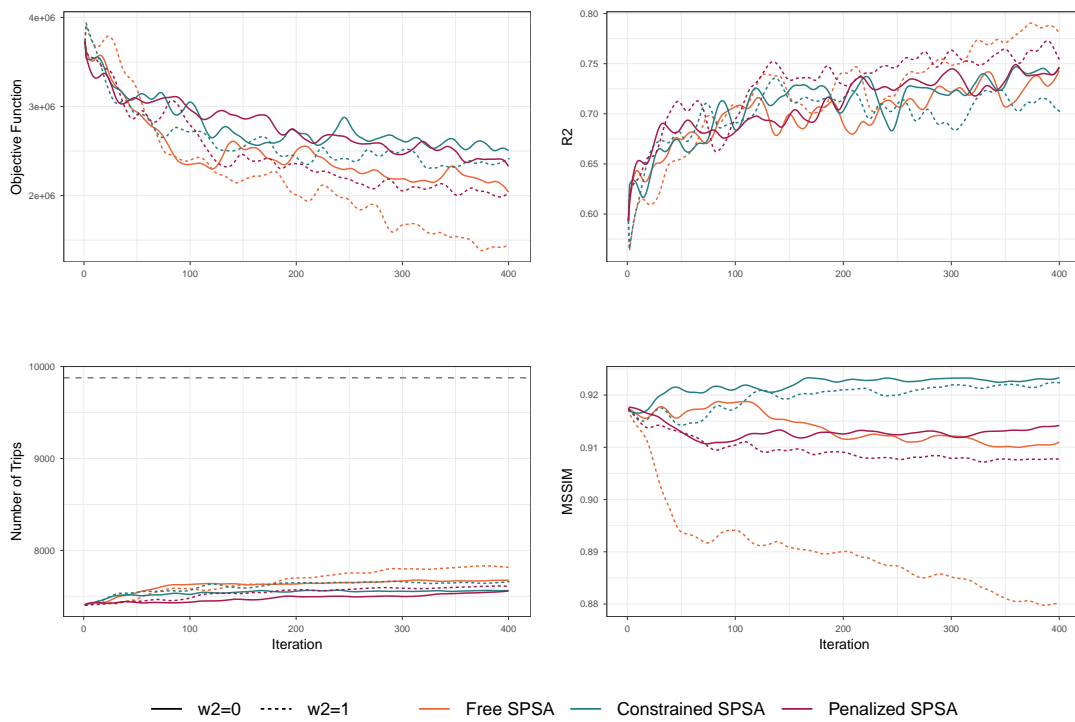


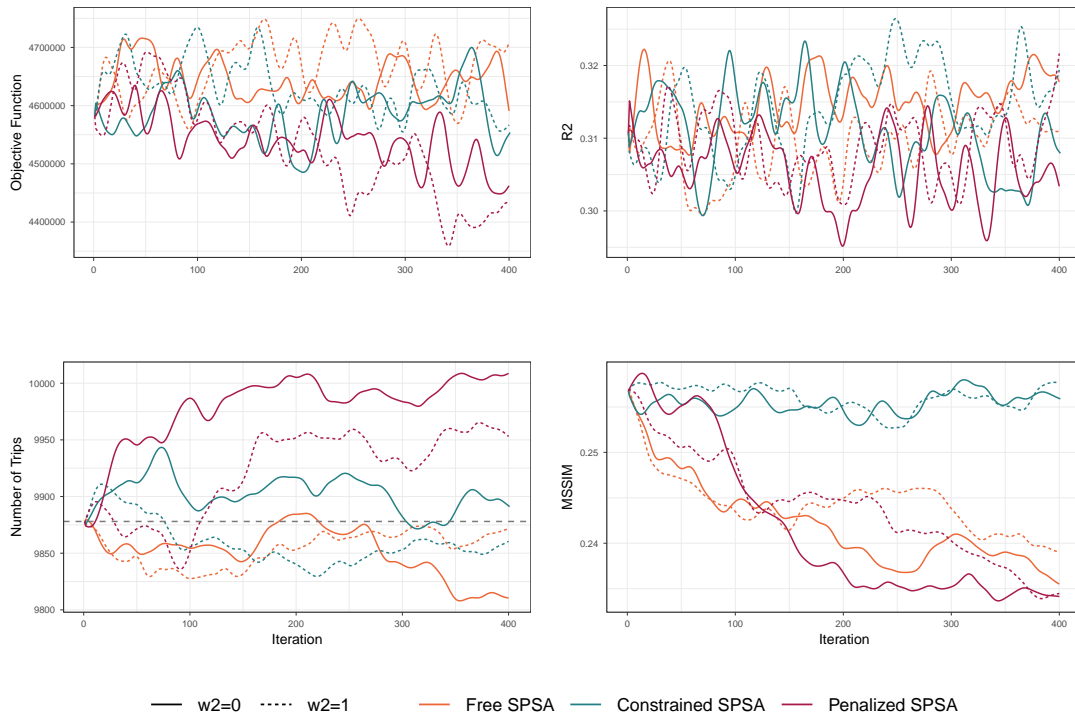Figure B.2: Multitude initial matrix, with NO conjugate gradient

Figure B.3: Chaos initial matrix, with YES conjugate gradient
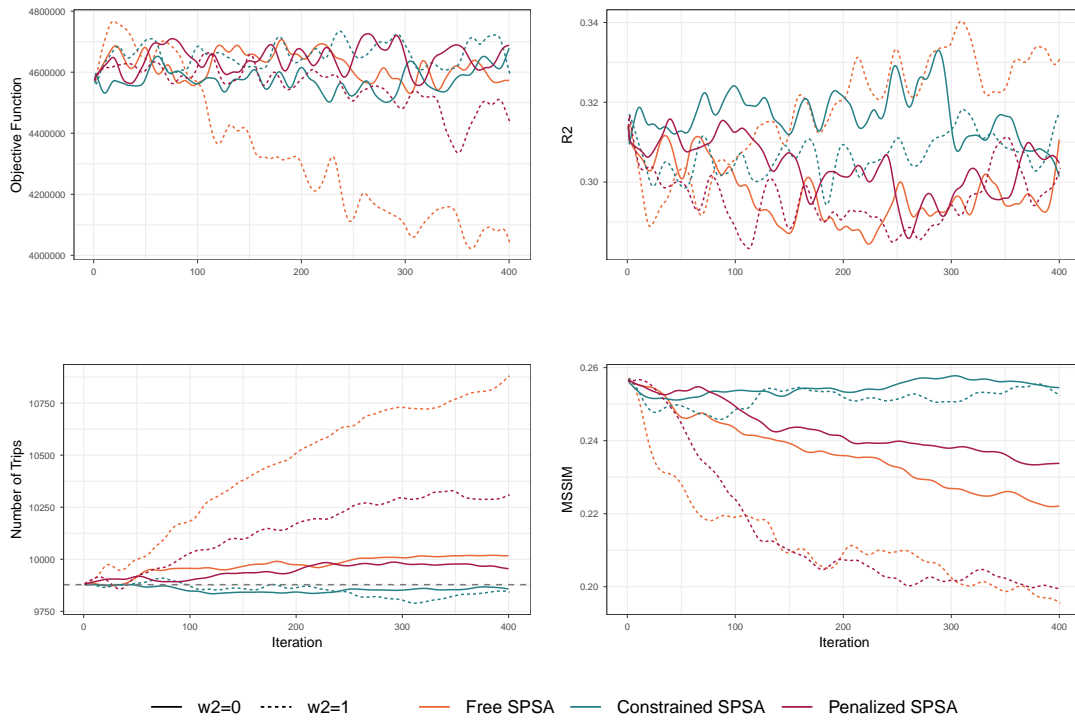


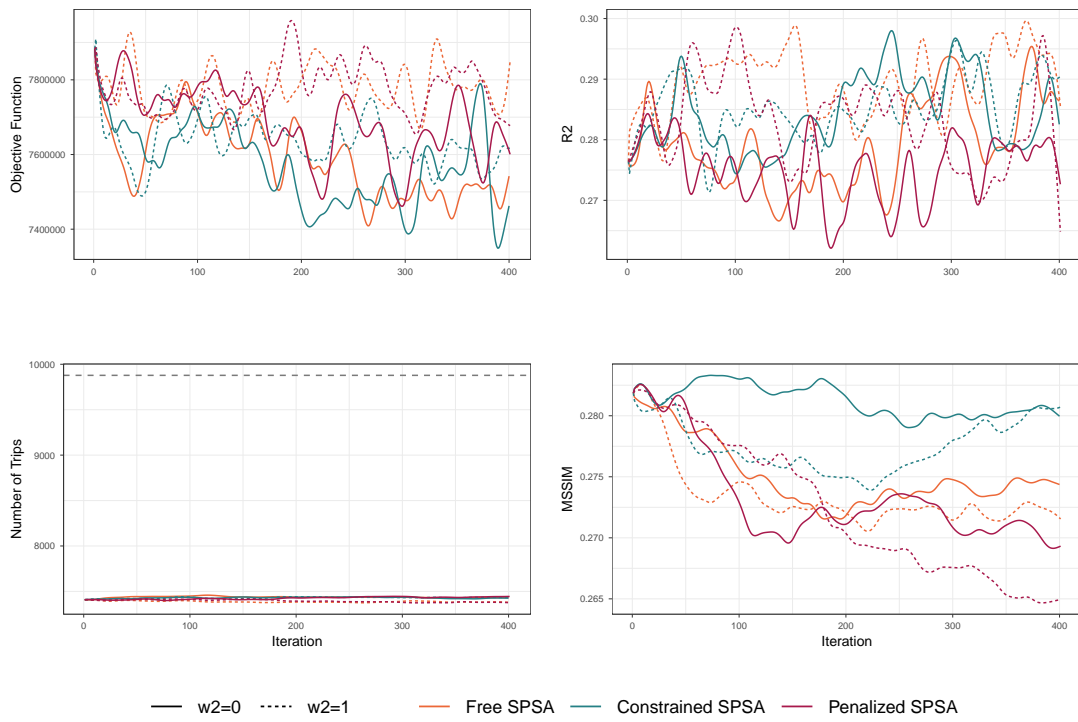Figure B.4: Chaos initial matrix, with NO conjugate gradient

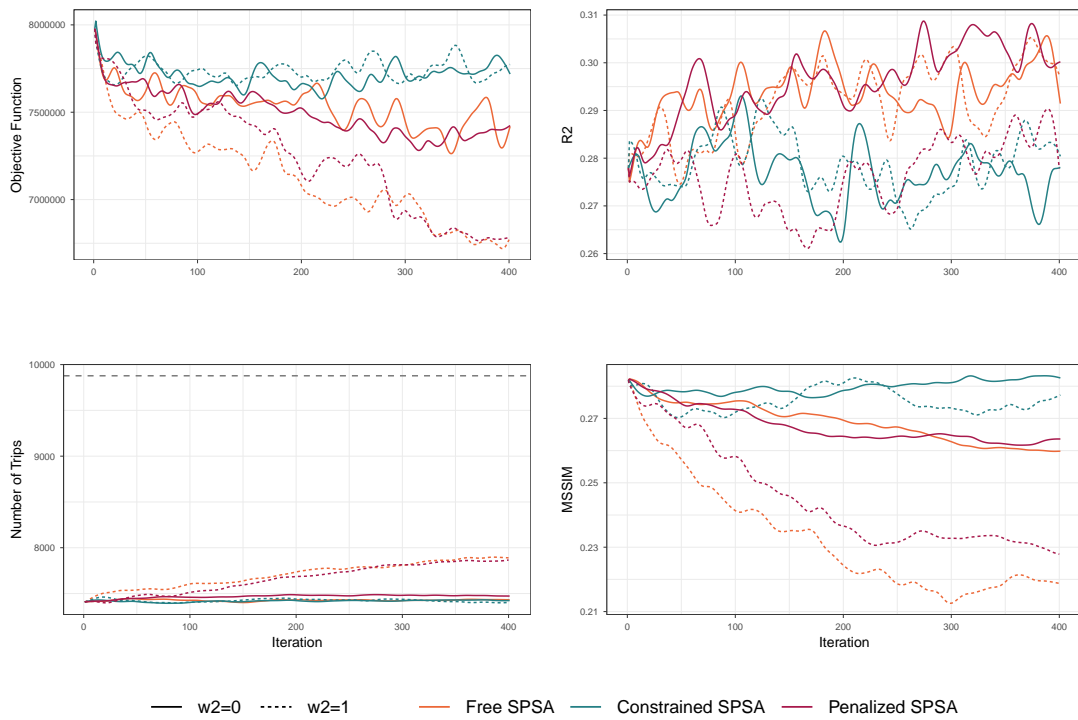Figure B.5: Chaos+Inc- initial matrix, with YES conjugate gradient



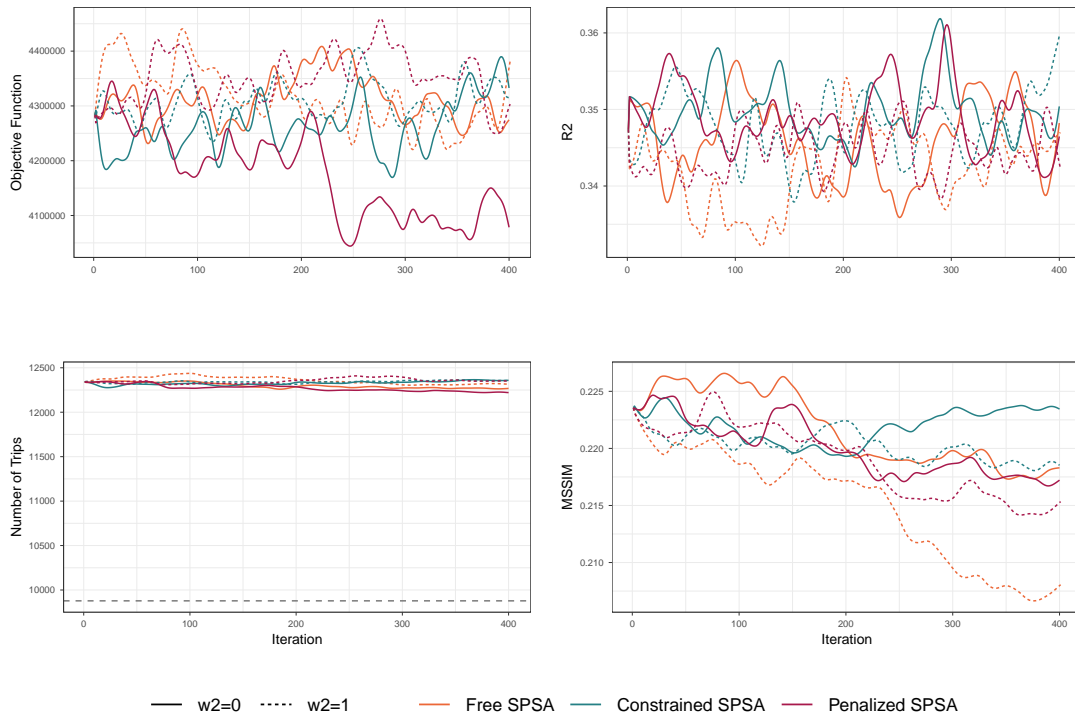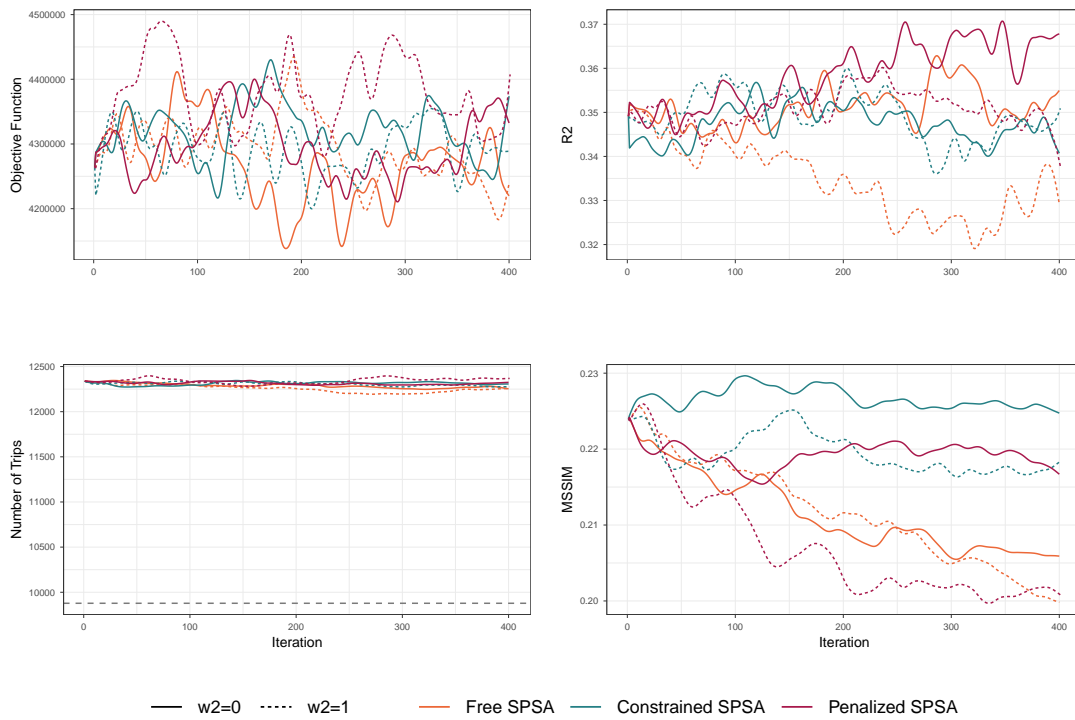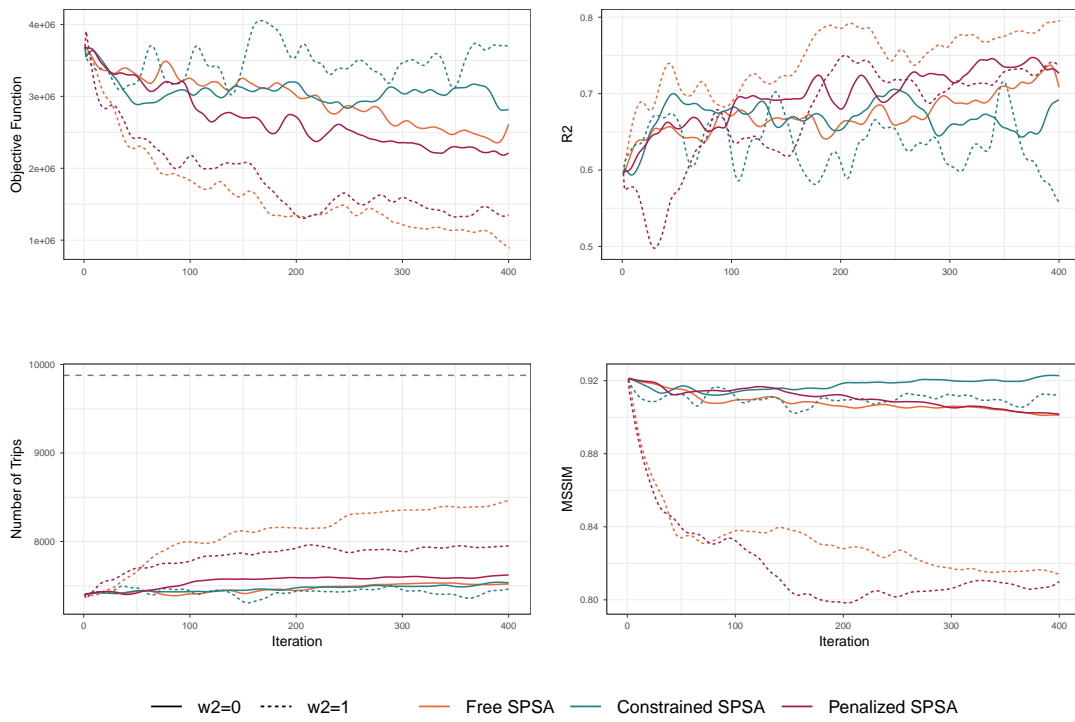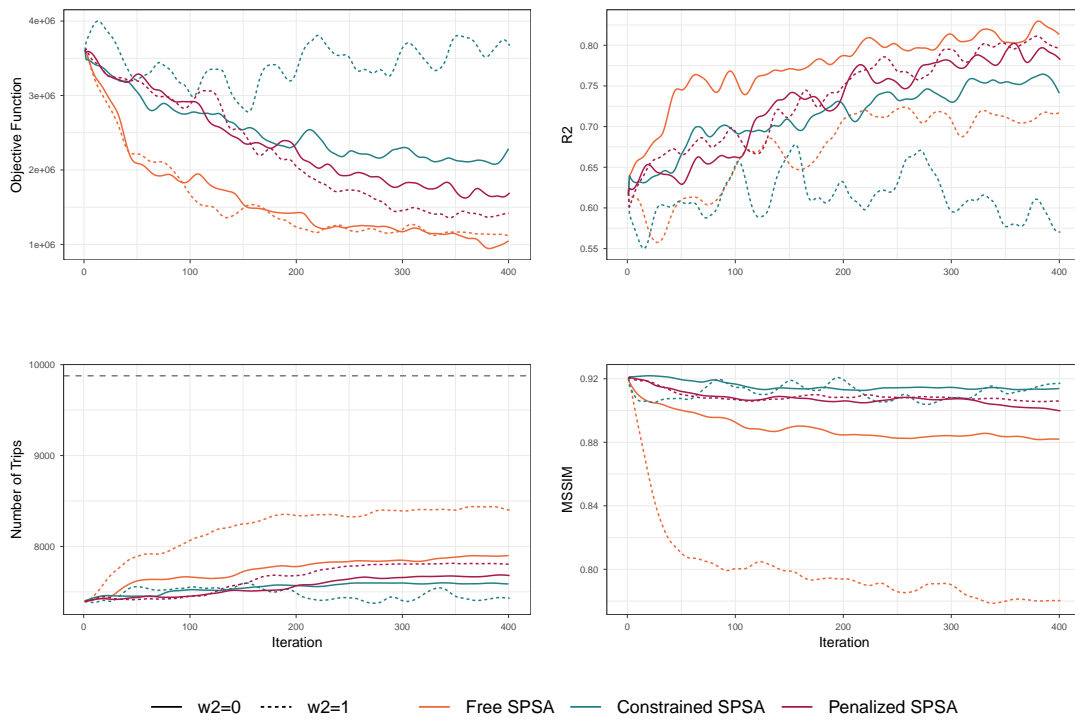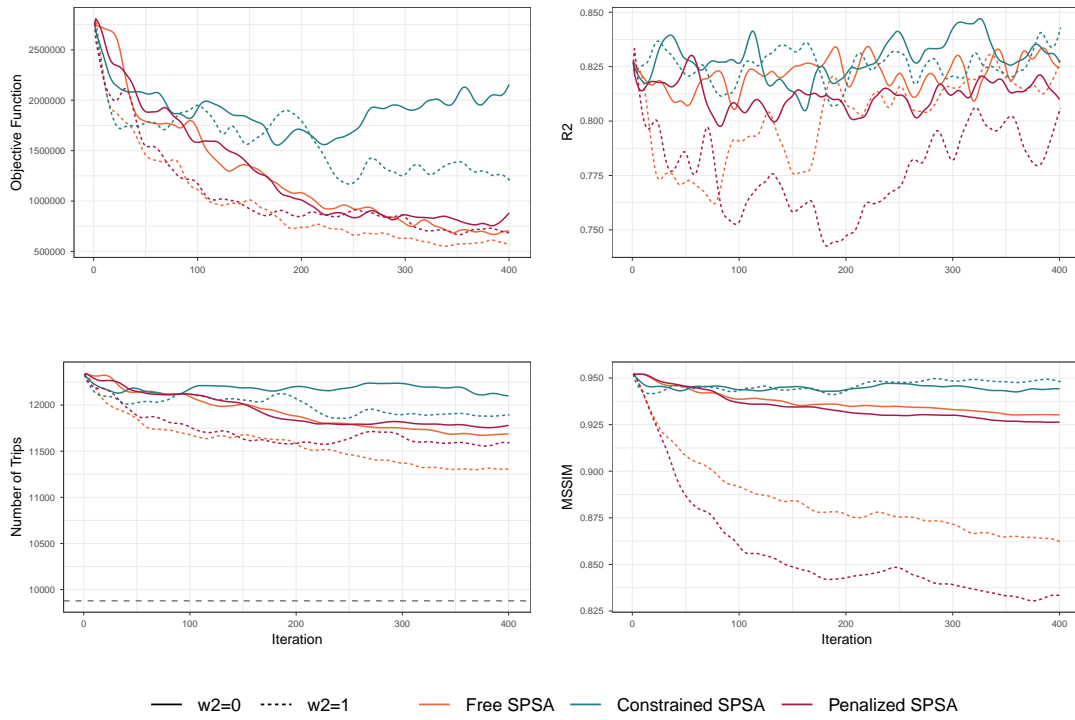Figure B.6: Chaos+Inc- initial matrix, with NO conjugate gradient

Figure B.7: Chaos+Inc+ initial matrix, with YES conjugate gradient



Figure B.8: Chaos+Inc+ initial matrix, with NO conjugate gradient

Figure B.9: Incremental- initial matrix, with YES conjugate gradient



Figure B.10: Incremental- initial matrix, with NO conjugate gradient

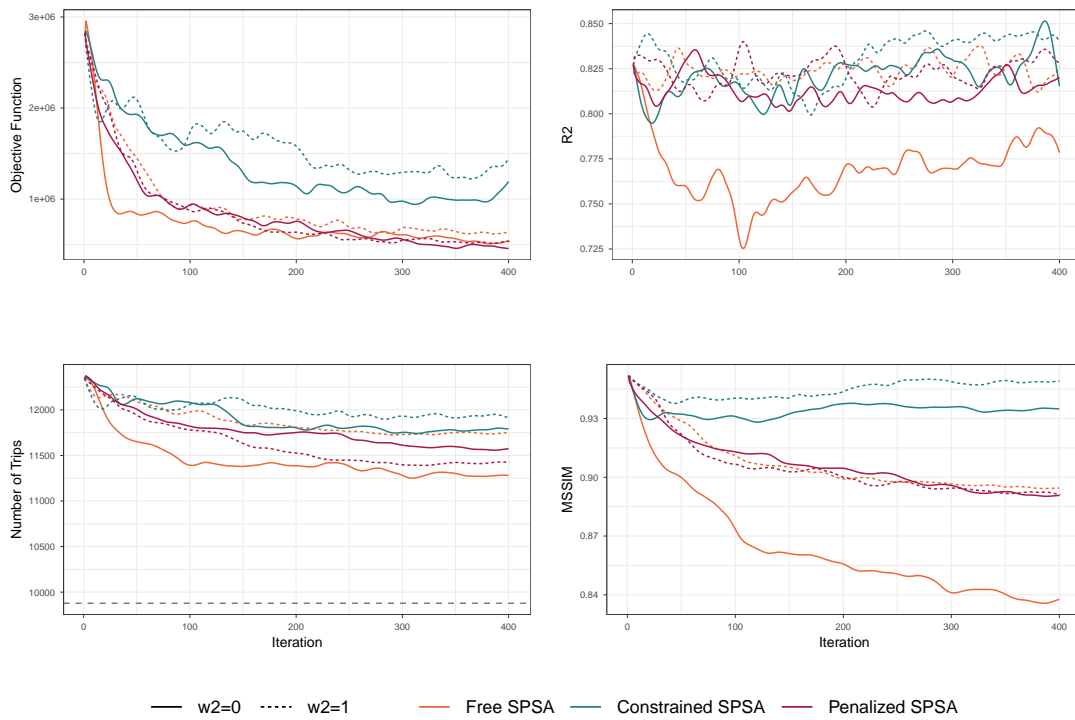Figure B.11: Incremental+ initial matrix, with YES conjugate gradient



Figure B.12: Incremental+ initial matrix, with NO conjugate gradient

## B.2   Results of SPSA with Travel Times

Table B.5: Initial values of the KPIs

| KPI | Multitude |
|---|---|
| $R_0^2$ Traffic counts | 0.5345 |
| $R_0^2$ Travel times | 0.4714 |
| NT | 7408 |
| $MSSIM_0$ | 0.9769 |

Since the experiments were launched for 400 iterations without a stopping criterion, the final indicators show the values at the iteration when $R^2$ reached the maximum value.

Table B.6: Final $R^2$ between traffic counts

| Method | $w_2 = 0$ | $w_2 = 1$ |
|---|---|---|
| SPSA without TT | 0.8297 | 0.8319 |
| SPSA with TT | 0.7614 | 0.7488 |
| HybSPSA without TT | 0.7598 | 0.7684 |
| HybSPSA with TT | 0.6485 | 0.7243 |

Table B.7: Final $R^2$ between travel times

| Method | $w_2 = 0$ | $w_2 = 1$ |
|---|---|---|
| SPSA without TT | - | - |
| SPSA with TT | 0.8312 | 0.8426 |
| HybSPSA without TT | - | - |
| HybSPSA with TT | 0.8371 | 0.8297 |

Table B.8: Final total number of trips, NT

| Method | $w_2 = 0$ | $w_2 = 1$ |
|---|---|---|
| SPSA without TT | 7415 | 7458 |
| SPSA with TT | 7572 | 7529 |
| HybSPSA without TT | 7612 | 7598 |
| HybSPSA with TT | 7416 | 7385 |

Table B.9: Final MSSIM between $\mathbf{X}^*$ and $\mathbf{X}^{GT}$

| Method | $w_2 = 0$ | $w_2 = 1$ |
|---|---|---|
| SPSA without TT | 0.9189 | 0.9207 |
| SPSA with TT | 0.9282 | 0.9318 |
| HybSPSA without TT | 0.9371 | 0.9371 |
| HybSPSA with TT | 0.9283 | 0.9278 |

(a) Objective Function                              (b) $R^2$

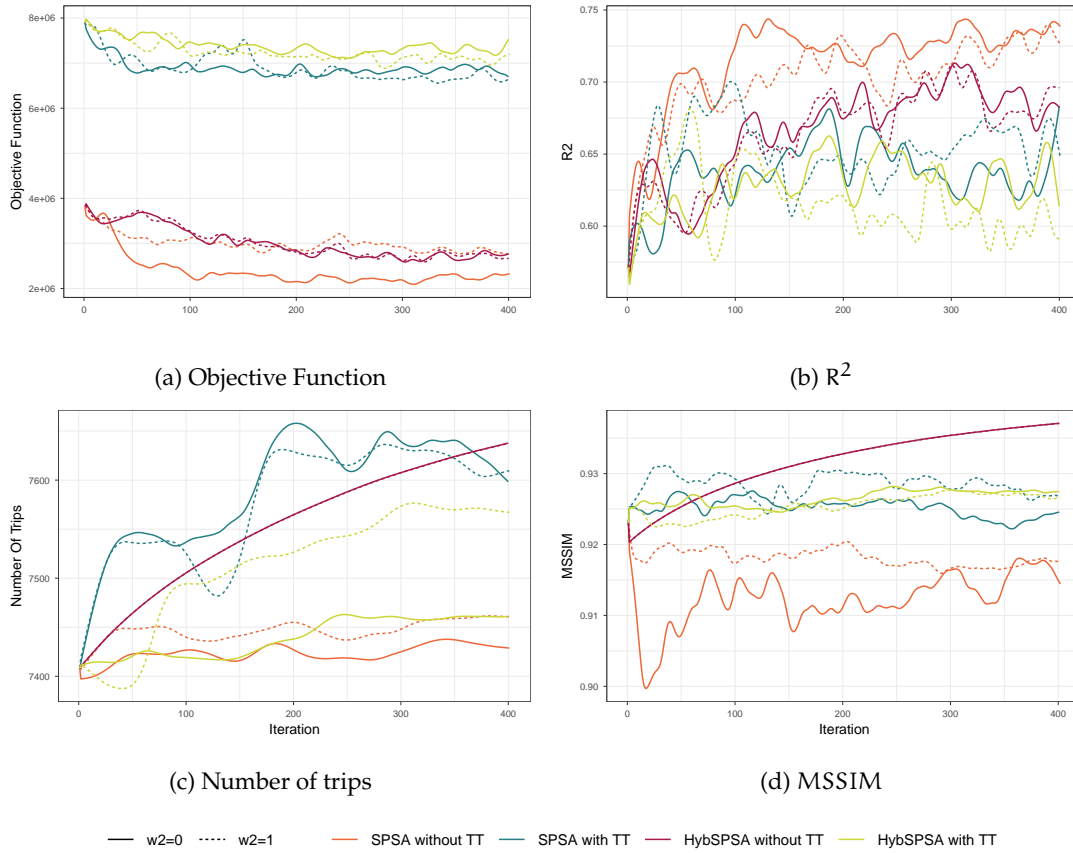(c) Number of trips                                 (d) MSSIM

Figure B.13: Results for SPSA with travel times variants

## B.3   Results of Dynamic Spiess

Table B.10: Initial values of the KPIs

| KPI | Chaos | Chaos+Inc$^-$ | Chaos+Inc$^+$ | Inc$^-$ | Inc$^+$ | Multitude |
|---|---|---|---|---|---|---|
| $R^2_0$ | 0.3149 | 0.2775 | 0.3453 | 0.6226 | 0.8278 | 0.5345 |
| $NT_0$ | 9878 | 7408 | 12347 | 7408 | 12347 | 7408 |
| $MSSIM_0$ | 0.5538 | 0.6301 | 0.4684 | 0.9808 | 0.9898 | 0.9769 |

Since the experiments were launched for 400 iterations without a stopping criterion, the final indicators show the values at the iteration when $R^2$ reached the maximum value.

Table B.11: Final R$^2$ between traffic counts

| Method | $w_2$ | Chaos | Chaos+Inc$^-$ | Chaos+Inc$^+$ | Inc$^-$ | Inc$^+$ | Multitude |
|---|---|---|---|---|---|---|---|
| Dynamic | $w_2 = 0$ | 0.9670 | 0.9616 | 0.9660 | 0.9667 | 0.9583 | 0.9658 |
| Spiess | $w_2 = 1$ | 0.9476 | 0.9450 | 0.9461 | 0.9404 | 0.9316 | 0.9371 |
| Dynamic Spiess | $w_2 = 0$ | 0.9994 | 0.9996 | 0.9988 | 0.9953 | 0.9937 | 0.9939 |
| on convergence | $w_2 = 1$ | 0.9818 | 0.9792 | 0.9728 | 0.9747 | 0.9655 | 0.9666 |
| Dynamic Spiess with Entropy | $w_2 = 0$ | 0.9436 | 0.9490 | 0.9440 | 0.9406 | 0.9362 | 0.9410 |

Table B.12: Final total number of trips, NT

| Method | $w_2$ | Chaos | Chaos+Inc$^-$ | Chaos+Inc$^+$ | Inc$^-$ | Inc$^+$ | Multitude |
|---|---|---|---|---|---|---|---|
| Dynamic | $w_2 = 0$ | 12906 | 12140 | 13847 | 11219 | 12892 | 11160 |
| Spiess | $w_2 = 1$ | 12616 | 12088 | 13783 | 10748 | 10758 | 10924 |
| Dynamic Spiess | $w_2 = 0$ | 14474 | 14471 | 14704 | 11603 | 12193 | 11870 |
| on convergence | $w_2 = 1$ | 12897 | 12820 | 14345 | 11159 | 11178 | 11322 |
| Dynamic Spiess with Entropy | $w_2 = 0$ | 12580 | 11729 | 13060 | 10066 | 11677 | 10712 |

Table B.13: Final MSSIM between $\mathbf{X}^*$ and $\mathbf{X}^{GT}$

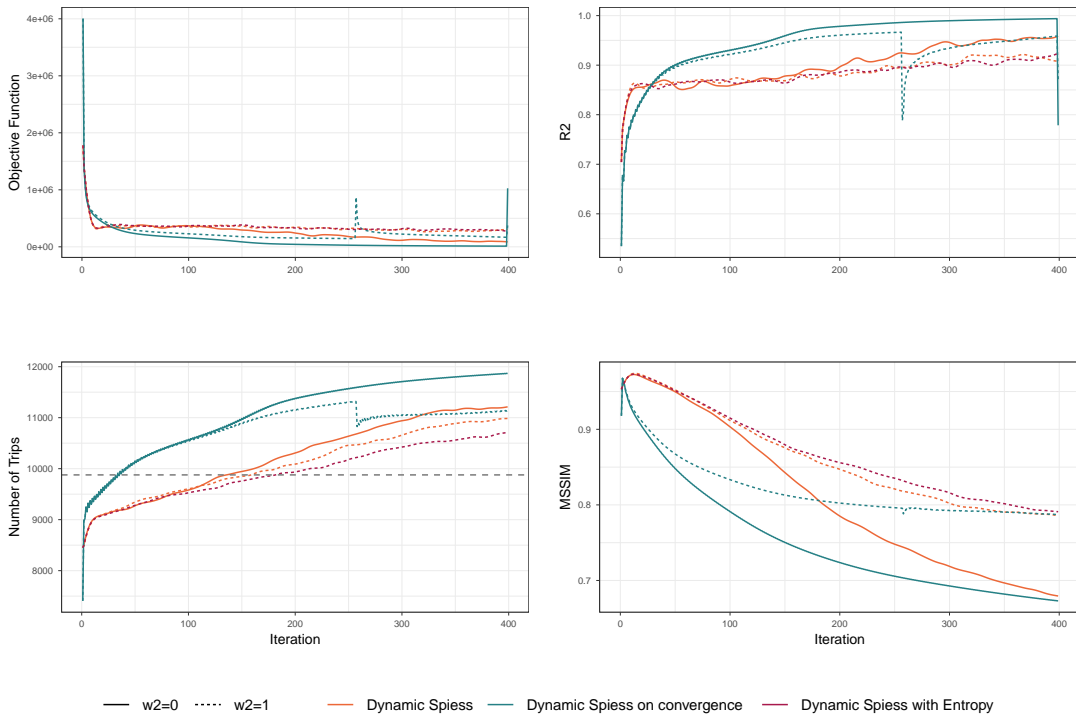| Method | $w_2$ | Chaos | Chaos+Inc$^-$ | Chaos+Inc$^+$ | Inc$^-$ | Inc$^+$ | Multitude |
|---|---|---|---|---|---|---|---|
| Dynamic | $w_2 = 0$ | 0.3730 | 0.3880 | 0.3594 | 0.6852 | 0.6645 | 0.7015 |
| Spiess | $w_2 = 1$ | 0.3795 | 0.3605 | 0.3355 | 0.7945 | 0.9524 | 0.7906 |
| Dynamic Spiess | $w_2 = 0$ | 0.3363 | 0.2836 | 0.3896 | 0.6724 | 0.7911 | 0.6731 |
| on convergence | $w_2 = 1$ | 0.3283 | 0.3008 | 0.3457 | 0.7963 | 0.8901 | 0.7957 |
| Dynamic Spiess with Entropy | $w_2 = 0$ | 0.3509 | 0.3875 | 0.3837 | 0.8524 | 0.8431 | 0.7911 |

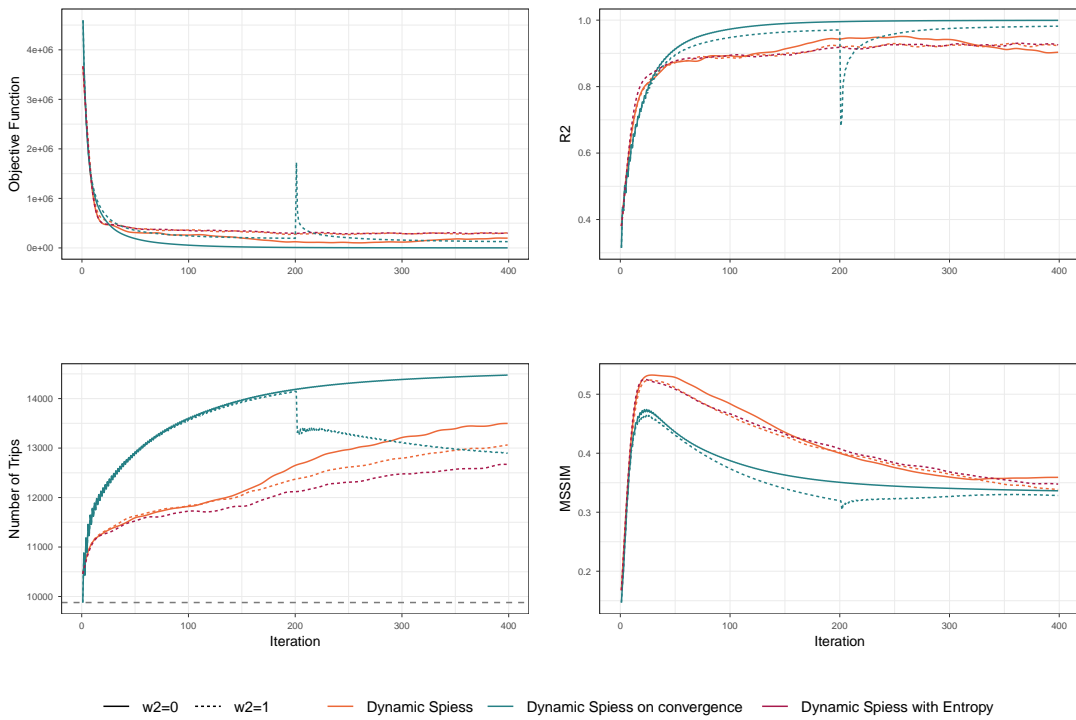Figure B.14: Multitude initial matrix



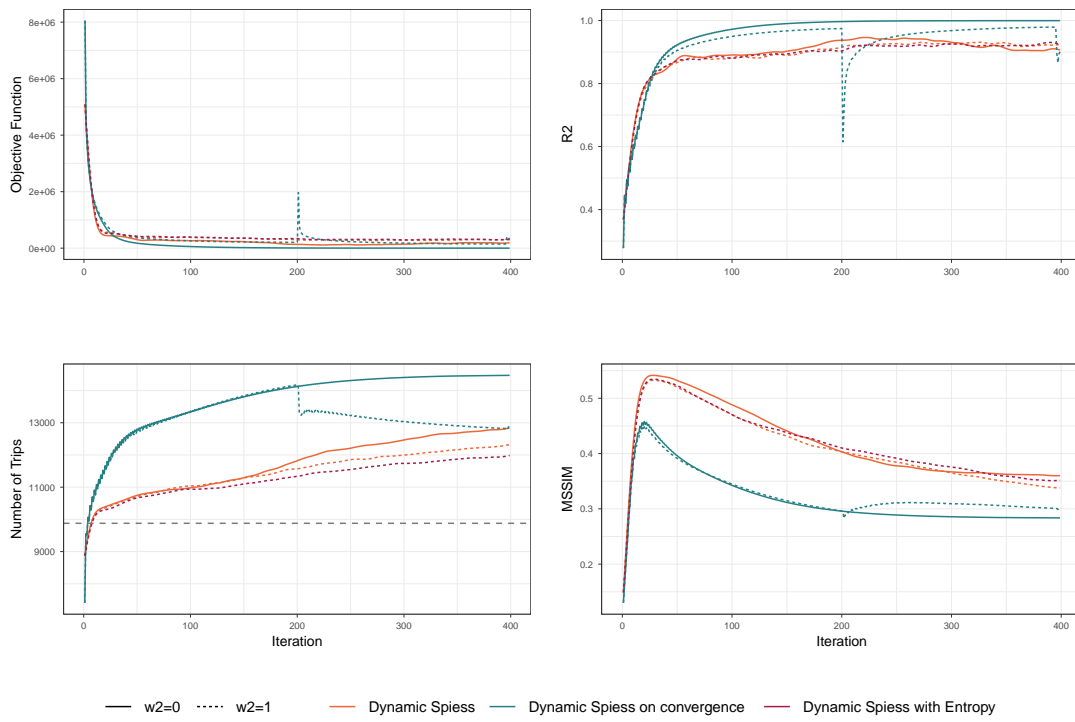Figure B.15: Chaos initial matrix
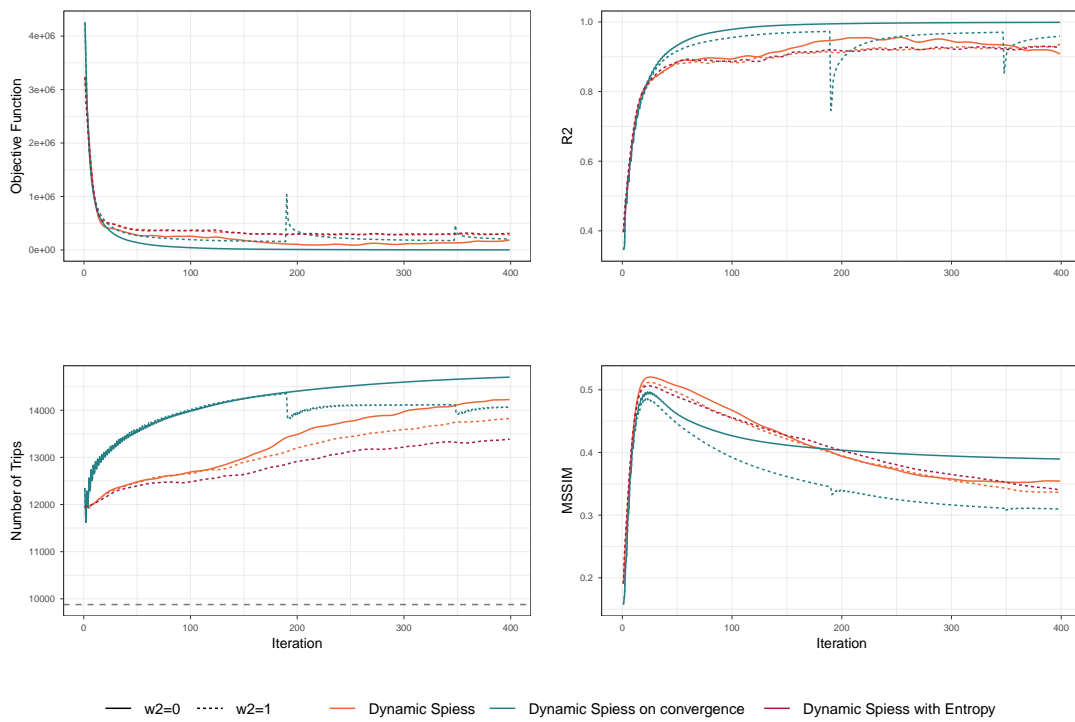
Figure B.16: Chaos+Inc- initial matrix



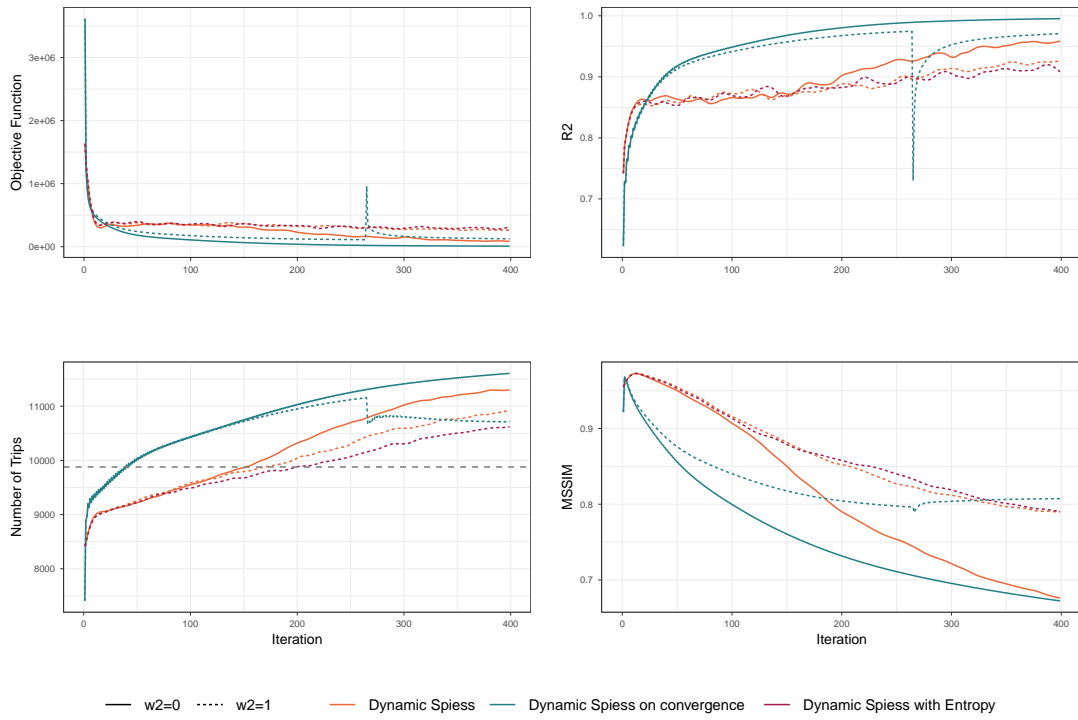Figure B.17: Chaos+Inc+ initial matrix

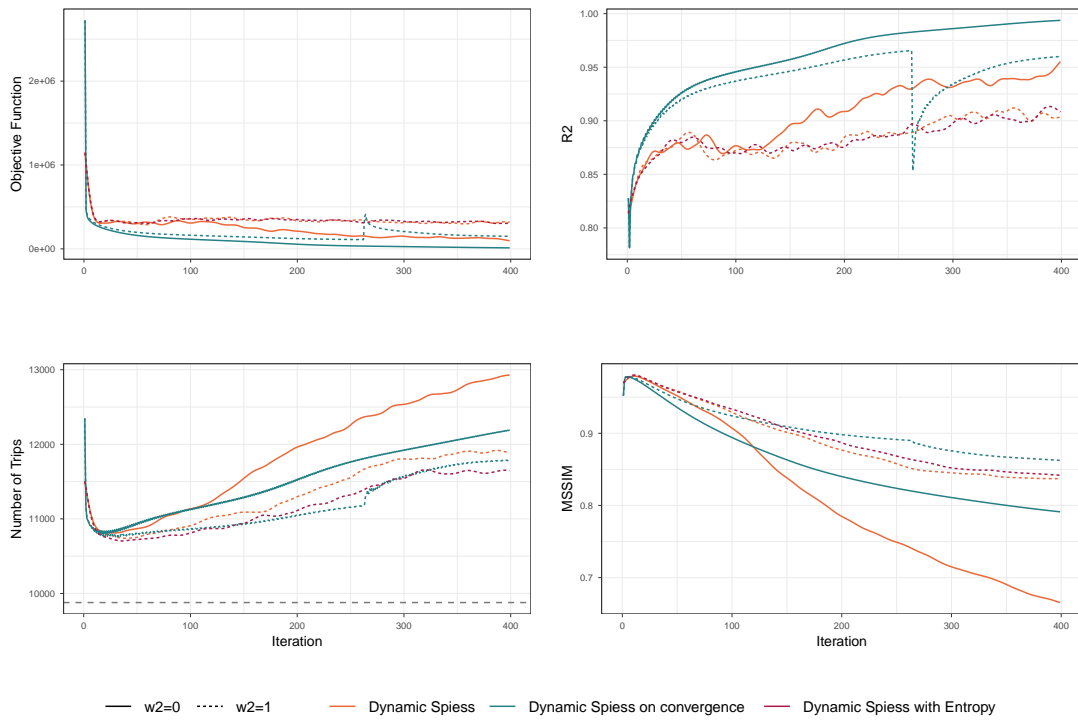Figure B.18: Incremental- initial matrix



Figure B.19: Incremental+ initial matrix

## B.4    Results of Data-Driven Assignment-Free DODME

Table B.14: Initial values of the KPIs

| KPI | Pen Rate | Hist | Obs | Comb1 | Comb2 |
|---|---|---|---|---|---|
| $R_0^2$ | 5% | 0.9248 | 0.9433 | 0.9414 | 0.9414 |
| | 10% | 0.9249 | 0.9243 | 0.9228 | 0.9229 |
| | 15% | 0.9398 | 0.9388 | 0.9379 | 0.9381 |
| $NT_0$ | 5% | 6232.14 | 6200.36 | 6179.25 | 6178.93 |
| | 10% | 6232.14 | 6189.27 | 6177.57 | 6177.58 |
| | 15% | 6232.14 | 6196.09 | 6187.39 | 6187.39 |
| $MSSIM_0$ | 5% | 0.9219 | 0.8025 | 0.8612 | 0.8611 |
| | 10% | 0.9219 | 0.8502 | 0.8857 | 0.8857 |
| | 15% | 0.9219 | 0.8653 | 0.8980 | 0.8979 |

Since the experiments were launched for 100 iterations without a stopping criterion, the final indicators show the values at the iteration when $R^2$ reached the maximum value.

Table B.15: Final $R^2$ between traffic counts

| Penetration Rate | $w_2$ | Hist | Obs | Comb1 | Comb2 |
|---|---|---|---|---|---|
| 5% | $w_2 = 0$ | 0.9821 | 0.9858 | 0.9833 | 0.9855 |
| | $w_2 = 1$ | 0.9718 | 0.9729 | 0.9715 | 0.9725 |
| 10% | $w_2 = 0$ | 0.9640 | 0.9701 | 0.9651 | 0.9700 |
| | $w_2 = 1$ | 0.9531 | 0.9553 | 0.9542 | 0.9566 |
| 15% | $w_2 = 0$ | 0.9775 | 0.9818 | 0.9779 | 0.9820 |
| | $w_2 = 1$ | 0.9690 | 0.9696 | 0.9684 | 0.9693 |

Table B.16: Final total number of trips, NT

| Penetration Rate | $w_2$ | Hist | Obs | Comb1 | Comb2 |
|---|---|---|---|---|---|
| 5% | $w_2 = 0$ | 8600.7125 | 8572.1045 | 8847.1183 | 8686.4964 |
| | $w_2 = 1$ | 8233.1443 | 8144.4301 | 8268.5176 | 8271.5702 |
| 10% | $w_2 = 0$ | 8582.2539 | 8186.3194 | 8636.3183 | 8262.0999 |
| | $w_2 = 1$ | 8208.4121 | 8158.0665 | 8207.2455 | 8226.9533 |
| 15% | $w_2 = 0$ | 8526.7401 | 8376.0333 | 8426.4535 | 8378.8388 |
| | $w_2 = 1$ | 8225.5134 | 8118.8477 | 8183.0890 | 8129.7176 |

Table B.17: Final MSSIM between $\mathbf{X}^*$ and $\mathbf{X}^{\mathrm{GT}}$

| Penetration Rate | $w_2$ | Hist | Obs | Comb1 | Comb2 |
|---|---|---|---|---|---|
| 5% | $w_2 = 0$ | 0.7048 | 0.4358 | 0.5940 | 0.5588 |
| | $w_2 = 1$ | 0.8437 | 0.6475 | 0.7501 | 0.7225 |
| 10% | $w_2 = 0$ | 0.6806 | 0.4932 | 0.6046 | 0.5500 |
| | $w_2 = 1$ | 0.8099 | 0.6664 | 0.7312 | 0.6684 |
| 15% | $w_2 = 0$ | 0.7163 | 0.5292 | 0.6955 | 0.5954 |
| | $w_2 = 1$ | 0.8000 | 0.7073 | 0.7863 | 0.7566 |
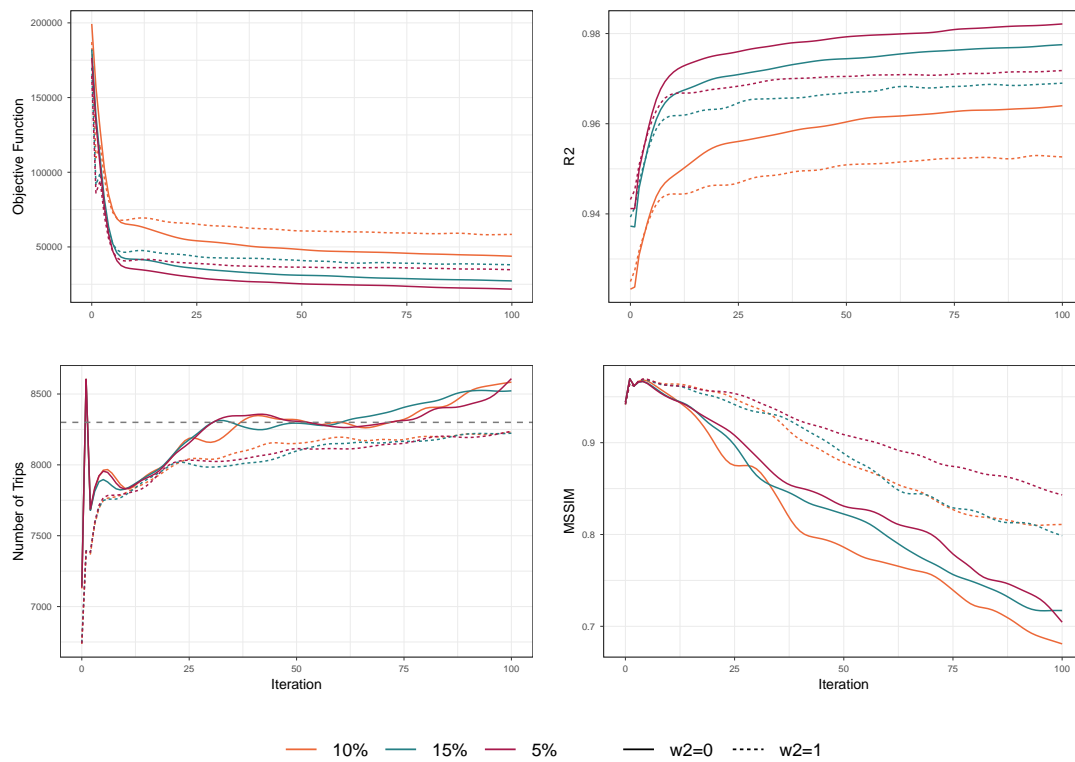


Figure B.20: Hist initial matrix
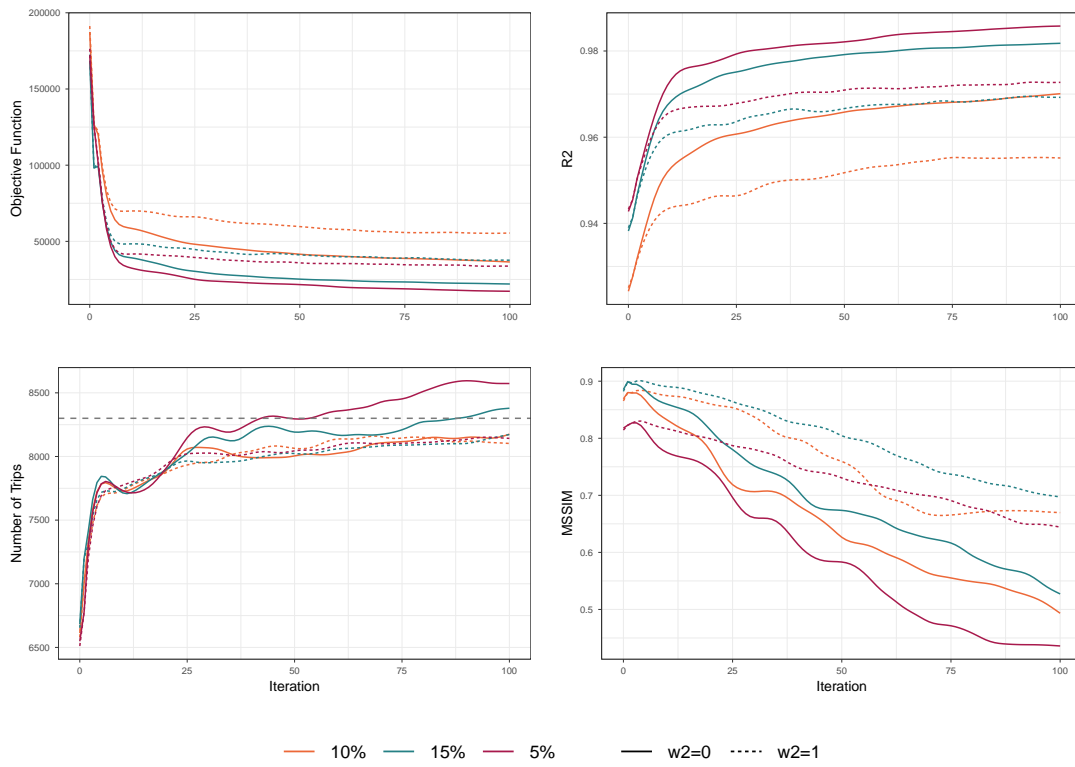
Figure B.21: Obs initial matrix
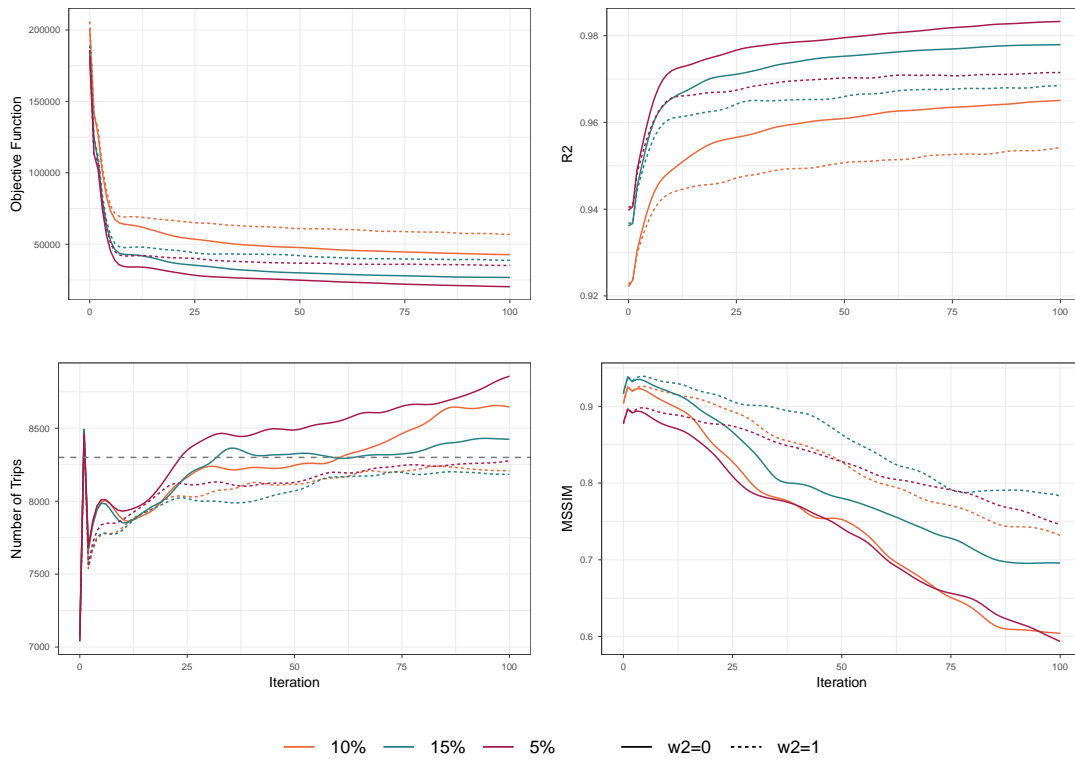


Figure B.22: Comb1 initial matrix
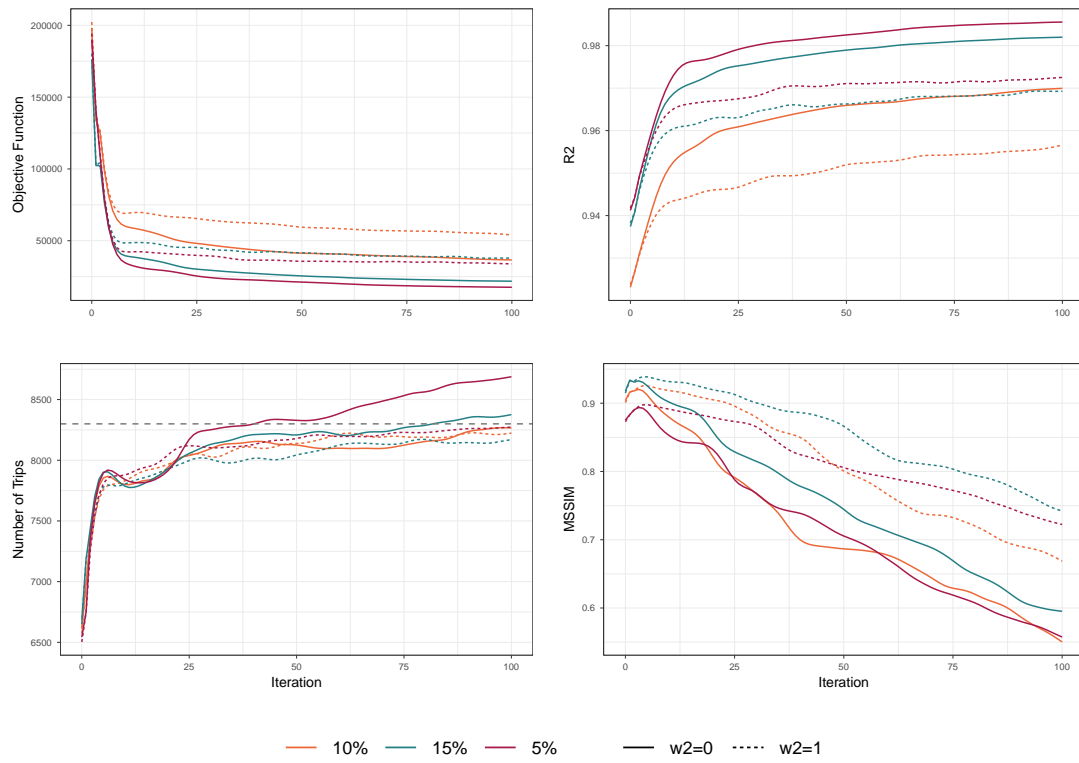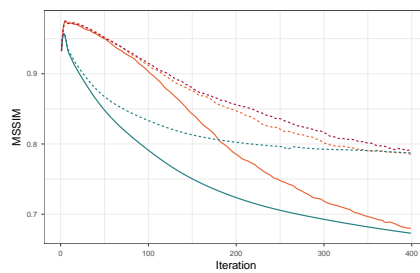
Figure B.23: Comb2 initial matrix

# C
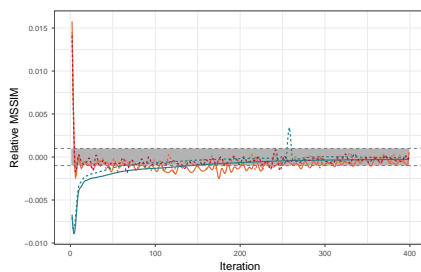
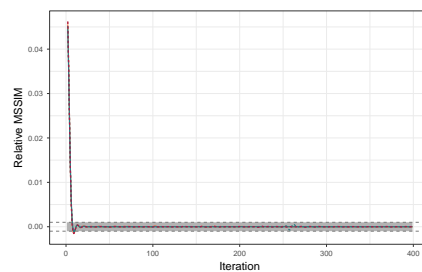# Full results of the stopping criteria

## C.1   Results of Dynamic Spiess



(a) MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{GT}\right)$

(b) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{H}\right)$

(c) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{(k-1)}\right)$

—— w2=0   ····· w2=1   —— Dynamic Spiess   —— Dynamic Spiess on convergence   —— Dynamic Spiess with Entropy

Figure C.1: Multitude initial matrix, with threshold $\varepsilon = 10^{-3}$
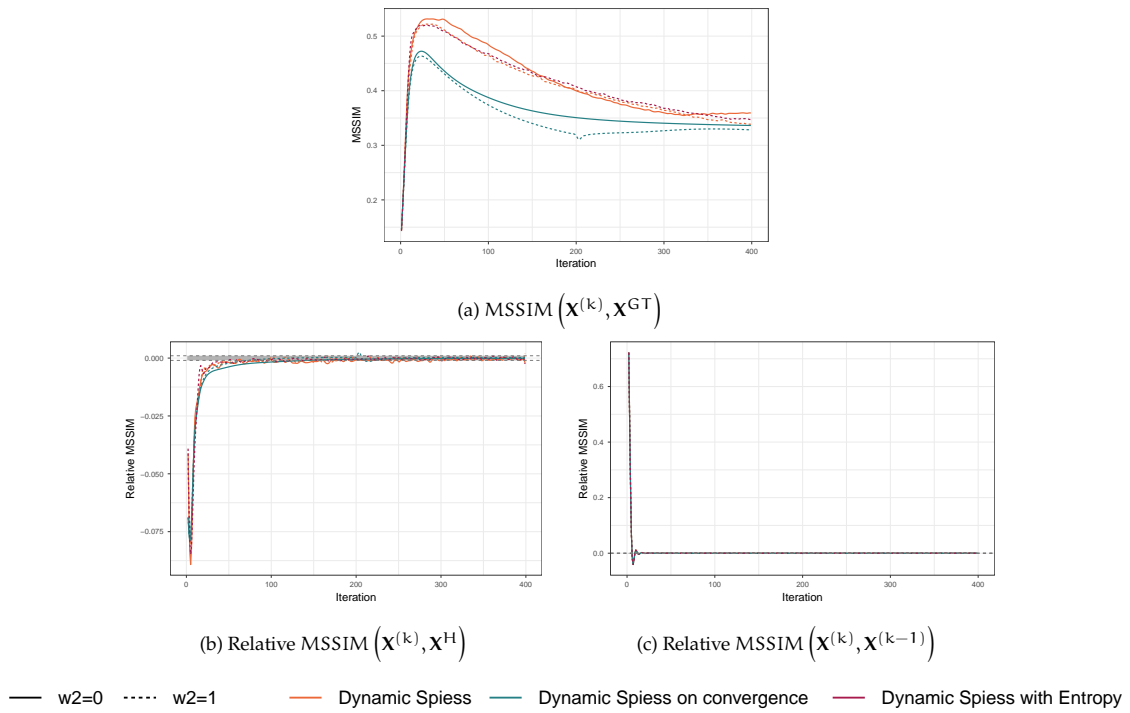
(a) MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{GT}\right)$



(b) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{H}\right)$    (c) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{(k-1)}\right)$

—— w2=0    ····· w2=1    —— Dynamic Spiess    —— Dynamic Spiess on convergence    —— Dynamic Spiess with Entropy

Figure C.2: Chaos initial matrix, with threshold $\varepsilon = 10^{-3}$



(a) MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{GT}\right)$



(b) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{H}\right)$    (c) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{(k-1)}\right)$

—— w2=0    ····· w2=1    —— Dynamic Spiess    —— Dynamic Spiess on convergence    —— Dynamic Spiess with Entropy
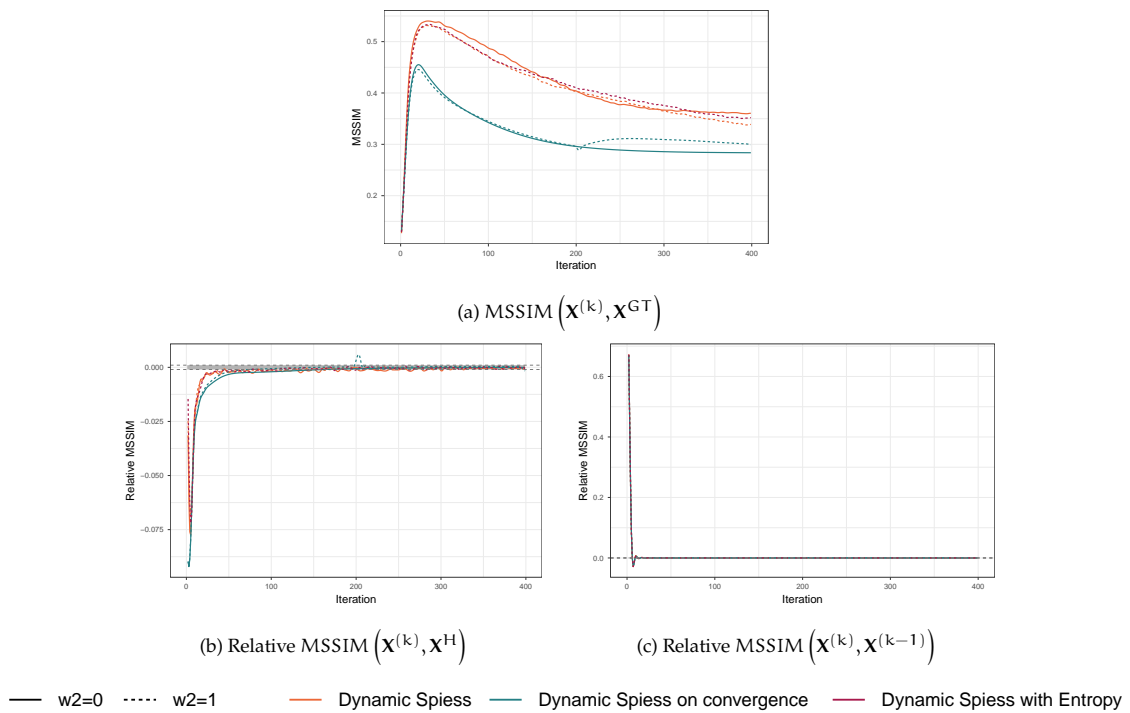
Figure C.3: Chaos+Inc- initial matrix, with threshold $\varepsilon = 10^{-3}$

(a) MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{GT}\right)$

(b) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{H}\right)$

(c) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{(k-1)}\right)$

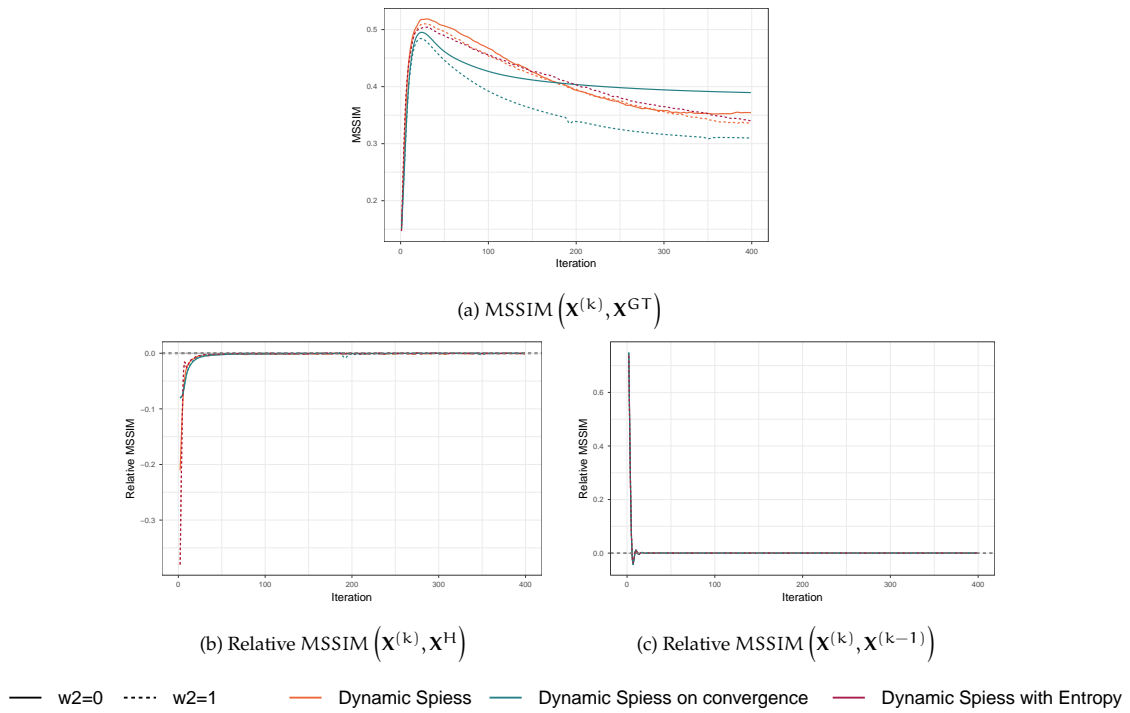w2=0 ······ w2=1 —— Dynamic Spiess —— Dynamic Spiess on convergence —— Dynamic Spiess with Entropy

Figure C.4: Chaos+Inc+ initial matrix, with threshold $\varepsilon = 10^{-3}$



(a) MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{GT}\right)$

(b) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{H}\right)$

(c) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{(k-1)}\right)$

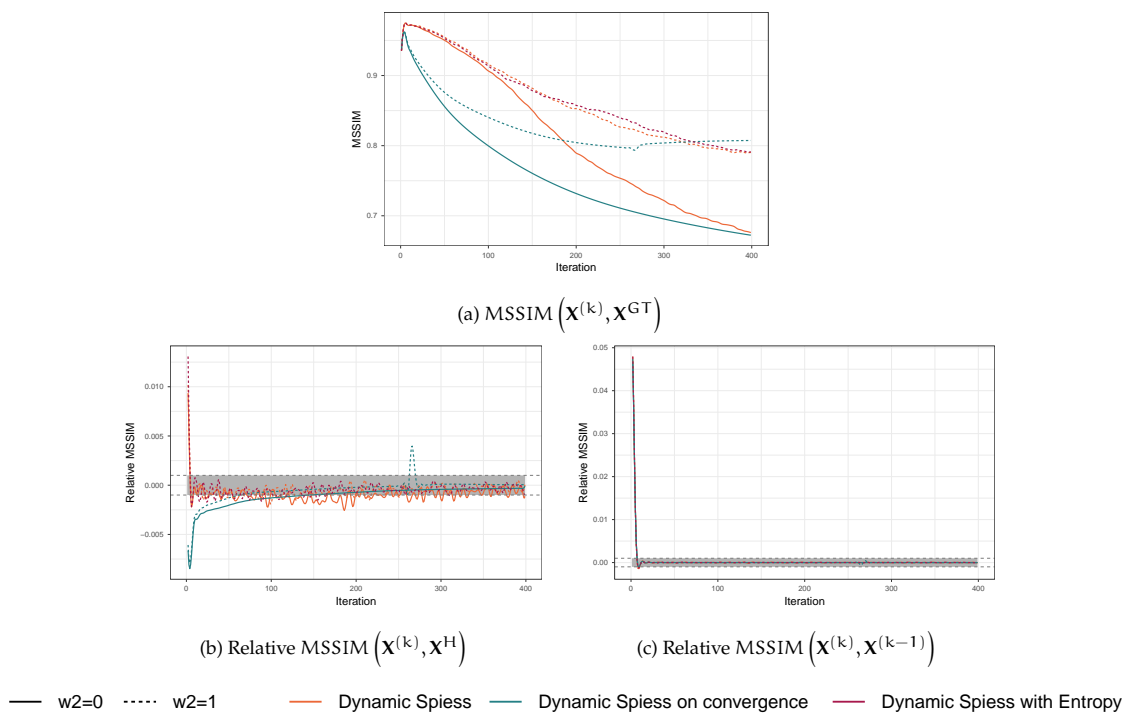w2=0 ······ w2=1 —— Dynamic Spiess —— Dynamic Spiess on convergence —— Dynamic Spiess with Entropy

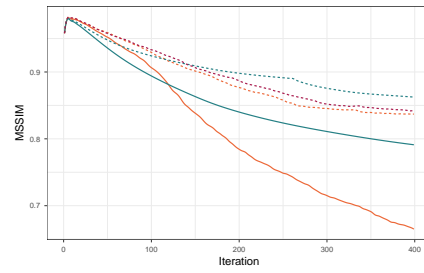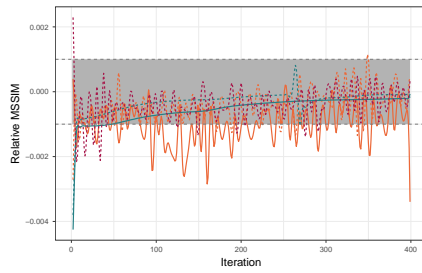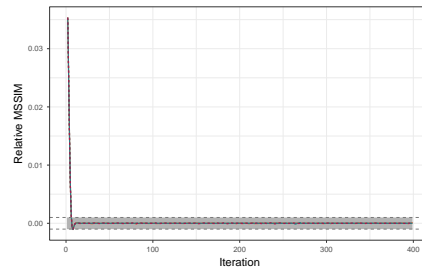Figure C.5: Incremental- initial matrix, with threshold $\varepsilon = 10^{-3}$

(a) MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{GT}\right)$



(b) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{H}\right)$

(c) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{(k-1)}\right)$

——— w2=0    ······ w2=1    ——— Dynamic Spiess    ——— Dynamic Spiess on convergence    ——— Dynamic Spiess with Entropy

Figure C.6: Incremental+ initial matrix, with threshold $\varepsilon = 10^{-3}$
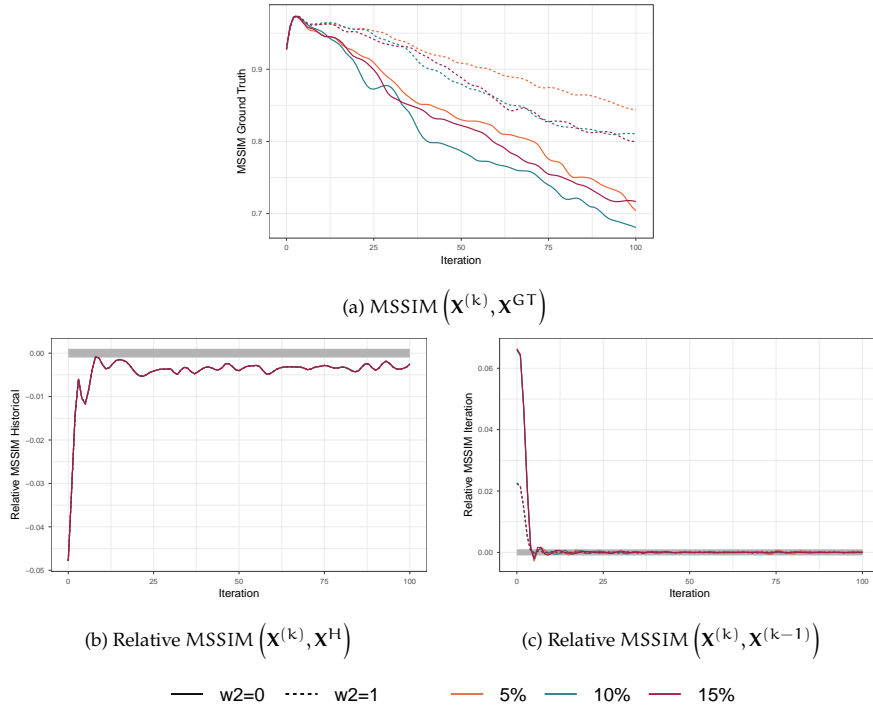
## C.2   Results of Data-Driven Assignment-Free DODME



(a) MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{GT}\right)$



(b) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{H}\right)$          (c) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{(k-1)}\right)$

—— w2=0   ····· w2=1   —— 5%   —— 10%   —— 15%

Figure C.7: Hist initial matrix, with threshold $\varepsilon = 10^{-3}$



(a) MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{GT}\right)$



(b) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{H}\right)$          (c) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{(k-1)}\right)$

—— w2=0   ····· w2=1   —— 5%   —— 10%   —— 15%
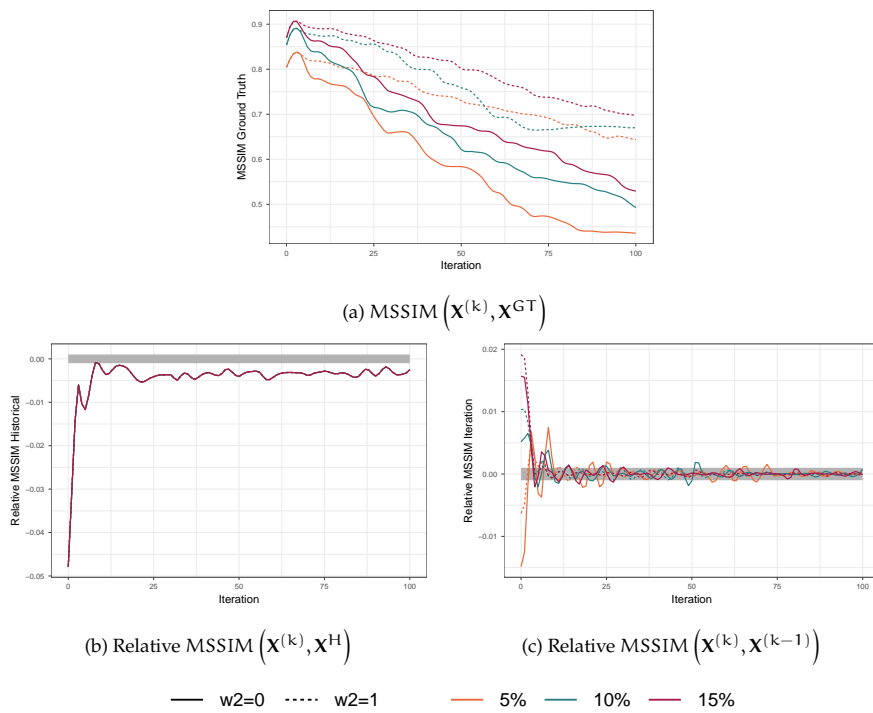
Figure C.8: Obs initial matrix, with threshold $\varepsilon = 10^{-3}$

(a) MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{GT}\right)$



(b) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{H}\right)$

(c) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{(k-1)}\right)$

— w2=0    ····· w2=1    — 5%    — 10%    — 15%
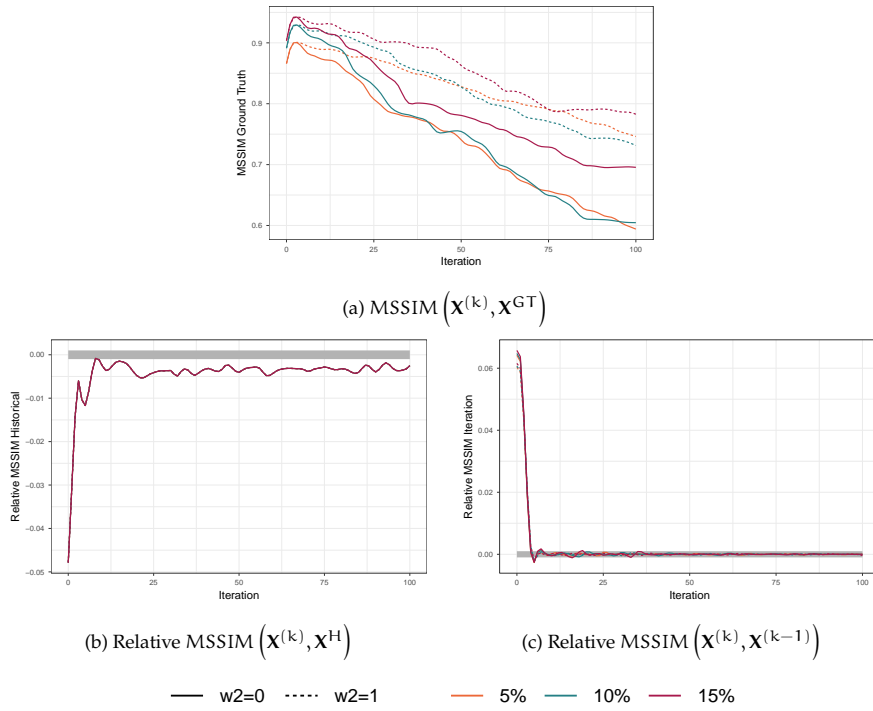
Figure C.9: Comb1 initial matrix, with threshold $\varepsilon = 10^{-3}$



(a) MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{GT}\right)$



(b) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{H}\right)$

(c) Relative MSSIM $\left(\mathbf{X}^{(k)}, \mathbf{X}^{(k-1)}\right)$

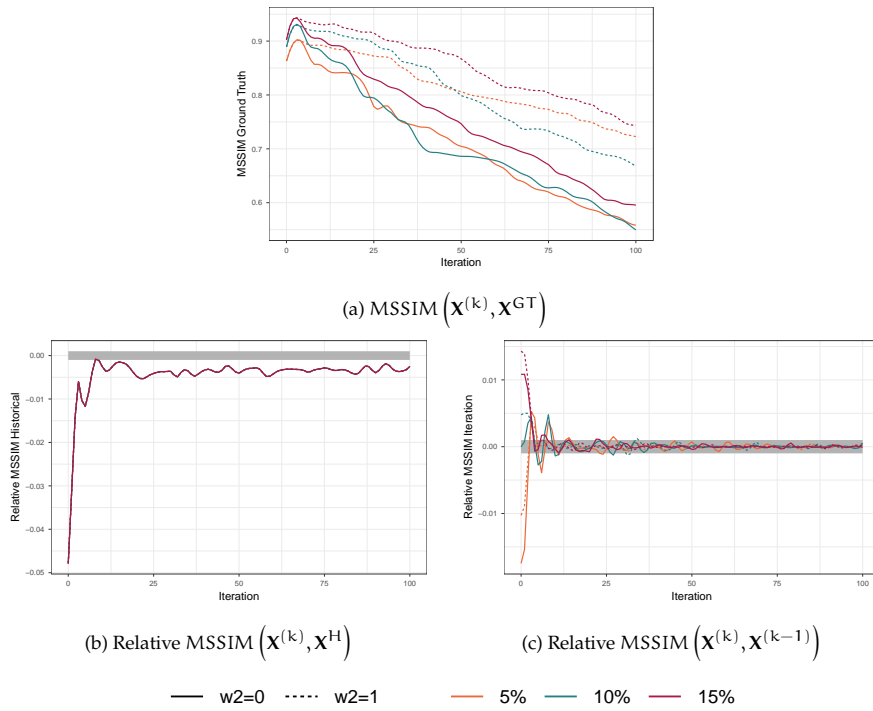— w2=0    ····· w2=1    — 5%    — 10%    — 15%

Figure C.10: Comb2 initial matrix, with threshold $\varepsilon = 10^{-3}$