Chapter 7
Data Analytics and Models for Understanding and Predicting Travel Patterns in Urban Scenarios

Jaume Barceló, Xavier Ros-Roca, and Lidia Montero

Abstract The estimation of the network traffic state, its likely short-term evolution and the prediction of the expected travel times in a network are key steps of traffic management and information systems, especially in urban areas and in real-time applications. To perform such functions, most systems have at their core engine specific dynamic traffic models whose main input is a dynamic OD-matrix describing the time dependency of travel patterns in urban scenarios. This chapter provides an overview of the main concepts supporting these dynamic traffic models and their practical implementations in some software platforms, as well as an outline on the main approaches for the estimation of dynamic OD-matrices. Additionally, this chapter provides a basic discussion on one of the main emerging trends: strategies aimed at using the unprecedented amount of new traffic data made available by "new" mobile technologies.

7.1 Dynamic Traffic Assignment Models

Most of the real-time traffic management systems are based on conceptual architectures embedding in their core engines dynamic traffic models, usually a *Dynamic Traffic Assignment* (DTA) or *Dynamic User Equilibrium* (DUE) model. These models are aimed at providing, among others, outputs to predict traffic flows and travel times on road networks, which vary over time because of various factors. One of these factors is particularly relevant: the time variation of the demand. Traffic assignment accounting for these time dependencies are referred to as DTA. When the

J. Barceló (✉) · X. Ros-Roca · L. Montero
Department of Statistics and Operations Research, Polytechnic University of Catalonia, UPC-BarcelonaTech, Catalonia, Spain
e-mail: jaume.barcelo@upc.edu

X. Ros-Roca
e-mail: xavier.ros.roca@upc.edu

L. Montero
e-mail: lidia.montero@upc.edu

M. Martínez-Díaz (ed.), *The Evolution of Travel Time Information Systems*,
Springer Tracts on Transportation and Traffic 19,
https://doi.org/10.1007/978-3-030-89672-0_7

predicted flows are such that no user can unilaterally reduce their travel times, the resulting assignment is said to be a DUE. In any case, their main input is an OD-matrix, that is, the matrix representing the time dependencies of the demand (e.g., Barceló et al. 2004; Allström et al. 2017). Dynamic Traffic Models, either DTA or DUE, are the key tool to estimate traffic states, understanding traffic patterns. And, as already mentioned, to be able to provide a predictive information consistent with the conditions that drivers will experience in the network, thus accounting for traffic evolution. Both important functionalities become more relevant in the case or complex urban networks. This is explained in detail in Ben-Akiva et al. (2010), which describes the approaches on which DynaMIT is based as well as its objectives. Descriptions of other similar systems can be found in Barceló et al. (2007), Heygi et al. (2009), Meschini (2017), and Aimsun (2020). The last two references illustrate these approaches through their implementation in two worldwide used professional systems based on these applications: OPTIMA and Aimsun. The role of DTM becomes even more critical in recent real-time traffic management systems like the *Active Transportation and Demand Management* (ATDM) and the *Dynamic Mobility Applications* (DMA), two programs of the United States Department of Transportation (USDOT) (Mahmassani et al. 2017).

The DTA problem can be considered an extension of the well-known *Static Traffic Assignment* (STA) problem, widely used in transport planning. The dynamic version must be able to determine how link and path flows evolve with time in the traffic network because of a time-dependent demand defined in terms of a time-varying OD-matrix. In other words, the dynamic approach to traffic assignment must describe how traffic flow patterns evolve in time and space on the network (Mahmassani 2001). Subsequently, it must provide the estimations of the link and path travel times and their short-term expected evolution. These are the main inputs to derive the KPIs that lead to specific traffic management policies, namely, those concerning information to travelers, alternative dynamic re-routing, etc.

From this standpoint, the DUE problem can be defined as the dynamic version of Wardrop's Principle (Wardrop 1952; Friesz et al. 1993; Smith 1993; Ran and Boyce 1996): "*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 state*". In other words, the DUE formulation stipulates that the experienced travel cost, including travel time and early/late arrival penalties, is identical for those route and departure time choices selected by travelers between a given OD pair. There are several attempts to translate this formulation into a suitable model.

In a recent paper, Han et al. (2019) review the various formulations of the models and the associated algorithms used to compute DUE, starting from the seminal proposal of Friesz et al. (1993), which formulates it as an open-loop, non-atomic Nask-like game. "Open-loop" means in this context that the selection of routes by the travelers after leaving the origin does not vary in response to changes in the dynamic network conditions. For its part, the term "non-atomic" implies the assumption of techniques based on aggregated traffic flow dynamics instead of techniques based on individual vehicle dynamics. This hypothesis ensures that DUE suitably accounts

for two main aspects of travel behavior: the departure time choice and the route choice. Therefore, the modeling hypothesis implies that travel times are identical for all trips departing at the same time interval using the same route. Following with the contribution of Han et al. (2019), the two main components of DUE modeling approaches are highlighted:

- The mathematical expression of the equilibrium condition.
- The network performance model, which mimics flow propagation through the network. This is usually referred to as *Dynamic Network Loading* (DNL).

DTA/DUE have been the subject of intensive research and developments both from the theoretical point of view and as key components of most software platforms used for the practical implementation of traffic management systems. Consequently, as Han et al. (2019) report, the concept of dynamic equilibrium has been implemented in various ways, as, for example, variational inequalities, nonlinear complementarity problems, differential variational inequalities, etc. In this Chapter, we limit our discussion to the formulation in terms of variational inequalities (Friesz et al. 2013; Smith and Wisten 1995), which is the most frequent in practical implementations. It is based on the mathematical model (Eqs. 7.1. and 7.2) proposed by Wu (1998):

$$\left[tt_{rsp}(t) - \theta_{rs}(t)\right] * x_{rsp}(t) = 0 \forall p \in K_{rs}(t), \forall (r,s) \in, t \in [0,T] \tag{7.1}$$

$$s.t.\ \ tt_{rsp}(t) - \theta_{rs}(t) \geq 0 \forall p \in K_{rs}(t), \forall (r,s) \in, t \in [0,T]$$

$$tt_{rsp}(t), \theta_{rs}(t,), x_{rsp}(t) > 0 \forall p \in K_{rs}(t), \forall (r,s) \in, t \in [0,T]$$

and the flow balancing equations:

$$\sum_{\forall p \in K_{rs}(t)} x_{rsp}(t) = X_{rs}(t)\ \forall (r,s) \in I, t \in [0,T] \tag{7.2}$$

where $x_{rsp}(t)$ is the flow on path p departing from origin r to destination s, $tt_{rsp}(t)$ is the actual path cost from r to s on route p, $\theta_{rs}(t)$ is the cost of the shortest path from r to s, $K_{rs}(t)$ is the set of all available paths from r to s and $X_{rs}(t)$ is the demand (number of trips) from r to s, all of them at time interval t. For their part, I is the set of all origin–destination pairs (r, s) in the network and T the overall time period considered. It can be demonstrated that this is equivalent to solve a finite-dimensional vibrational inequality problem consisting of finding a vector $\boldsymbol{x}*$ of path flows and a vector τ of path travels times, such that

$$\left[x - x^*\right]^T * \tau \geq 0, \forall x \in \aleph \tag{7.3}$$

where $\aleph$ is the set of feasible flows defined by

$$\aleph = \left\{ x_{rsp}(t) \left| \sum_{\forall p \in K_{rs}(t)} x_{rsp}(t) = X_{rs}(t) \forall (r,s) \in I, t \in [0,T], x_{rsp}(t) > 0 \right. \right\} \quad (7.4)$$

Wu et al. (1991,1998a; b) probe that this is equivalent to solve the discretized variational inequality:

$$\sum_{t \in [0,T]} \sum_{p \in \Re} tt_{rsp}(t) * \left[x_{rsp}(t) - x^*_{rsp}(t) \right] \geq 0 \quad (7.5)$$

where $\Re = \bigcup_{(r,s) \in I} * K_{rs}$ is the set of all available paths from origins to destinations.

Reviews of DTA models can be found in Boyce et al. (2001), Peeta and Ziliaskopoulos (2001), Szeto and Lo (2005), Szeto and Wong (2012), Jeihani (2007), and Bliemer et al. (2017).

Algorithms to deal with DTA or DUE problems usually involve solving this variational inequality formulation. A wide variety of algorithms has been proposed: from projection algorithms (Wu et al. 1991,1998a; b; Florian et al. 2001) or methods of alternating directions (Lo and Szeto 2002) to various versions of the *Method of Successive Averages* (MSA) (Tong and Wong 2000; Florian et al. 2002; Mahut et al. 2003a, b; Mahut et al. 2004; Varia and Dhingra 2004).

The computational approaches proposed to solve the DTA problem can be broadly classified into two classes: mathematical formulations, looking for analytical solutions, and traffic simulation-based approaches, looking for approximate heuristic solutions. Both fit the conceptual framework proposed by Florian et al. (2001) and Florian et al. (2002), formalizing the relationships and dependencies between the two main components identified (Fig. 7.1):

- A method to determine the path-dependent flow rates on the paths on the network, usually applying any of the approaches mentioned above (MSA, projection methods, etc.).
- A DNL method, which determines how these path flows give raise to time-dependent arc volumes, arc travel times and path travel times.

Quite frequently, and basically in all practical implementations mentioned above, DNL method is based on a mesoscopic simulation model (Barceló 2010a) emulating the flow propagation through the network in the current conditions. Depending on how the convergence criterion and the iterative process implemented, the resulting assignment is a DTA or a DUE (see Chiu et al. 2011 for more details).

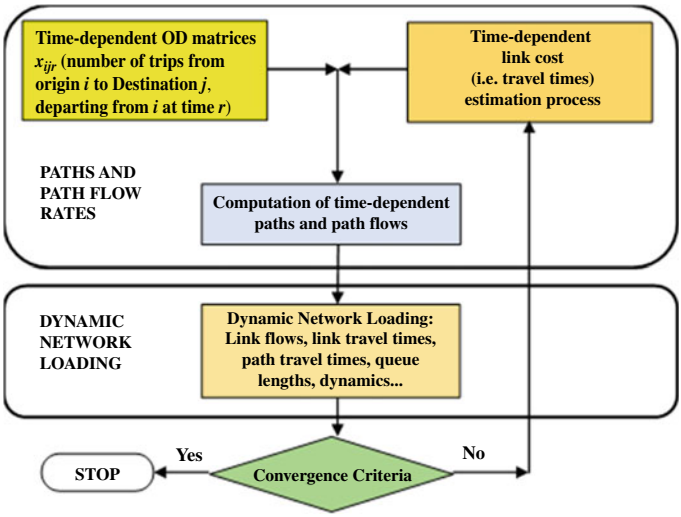

Fig. 7.1 Conceptual algorithmic scheme for DTA

7.1.1 Determining the Path-Dependent Flow Rates by MSA: Convergence Criterion to Equilibrium

If the convergence criteria are not met after one particular iteration of the conceptual algorithmic scheme in Fig. 7.1., a new one is performed. In this new iteration, after computing the new potential paths once the link costs have been updated, the key point is the determination of how the demand will be split among these paths, producing the corresponding path flows. Carey and Ge (2012) or Han et al. (2019) provide a comprehensive panoramic view of the many computational alternatives.

To illustrate these concepts, in this chapter we address the MSA method, one of the most frequently used in practice to estimate the path-dependent flow rates to solve (Eq. 7.5). MSA is a procedure that redistributes the flows among the available paths in an iterative procedure that, at any iteration n, computes a new shortest path, $c_{rs}(t)$, from origin r to destination s at time interval t. Depending on $c_{rs}(t)$ the path flows update process is as follows:

If $c_{rs}(t) \notin K_{rs}^{n}(t)$

$$x_{rsp}^{n+1}(t) = \begin{cases} \alpha_n * x_{rsp}^{n}(t) & if \ \ p \in K_{rs}^{n}(t) \\ (1-\alpha_n) * X_{rs}(t) & if \ \ p = c_{rs}(t) \end{cases} \quad \forall r, s, t \tag{7.6a}$$

$$K_{rs}^{n+1}(t) = K_{rs}^{n}(t) \cup c_{rs}(t) \tag{7.6b}$$

Otherwise if $c_{rs}(t) \in K_{rs}^{n}(t)$

$$x_{rsp}^{n+1}(t) = \begin{cases} \alpha_n * x_{rsp}^n(t) & if\ p \neq c_{rs}(t) \\ \alpha_n * x_{rsp}^n(t) + (1-\alpha_n) * X_{rs}(t) & if\ p = c_{rs}(t) \end{cases} \quad \forall r, s, t \tag{7.7a}$$

$$K_{rs}^{n+1}(t) = K_{rs}^n(t) \tag{7.7b}$$

Depending on the values of the weighting coefficients α_n, different MSA schemes can be implemented (Carey and Ge 2012), probably being the most typical value $\alpha_n = \frac{n}{n+1}$. Many variants have been suggested. For example, Varia and Dhingra (2004) propose a modified MSA algorithm where the weighting coefficient takes into account a variable step length that depends on the current path travel times (Eq. 7.8):

$$\alpha_n = \frac{\lambda_k * \left[exp\left(-tt_{rsp}(t)\right)\right]}{(n+1) * \left[\sum_p * \left[exp\left(-tt_{rsp}(t)\right)\right]\right]} \tag{7.8}$$

One of the potential computational drawbacks of these implementations of MSA is the growing number of paths when dealing with large networks. To avoid this in the case of DTA assignments, an alternative is to specify the maximum number K of paths to keep for each OD pair. Several modified implementations have been suggested to keep control of the number of paths in MSA algorithms (Peeta and Mahmassani 1995; Sbayti et al. 2007). Interesting proposals are those in Mahut et al. (2003a,2004; b). Possibly, one of the most computationally efficient is the one proposed by Florian et al. (2002). This variant of the algorithm initializes the process based on an incremental loading scheme that distributes the demand among the available shortest paths. The process is repeated for a predetermined number of iterations, after which no new paths are added and the corresponding fraction of the demand is redistributed according to the MSA scheme. This modified MSA works as follows:

Let K be the maximum number of iterations to compute new paths.

If $n \leq K$

a new shortest path $c_{rs}(t) \notin K_{rs}^n(t)$ is found. Then,

$$x_{rsp}^{n+1}(t) = \frac{1}{n+1} * X_{rs}(t) \quad \forall p \in K_{rs}^n(t), \forall (r,s) \in I, t \in [0,T] \tag{7.9a}$$

$$K_{rs}^{n+1}(t) = K_{rs}^n(t) \cup c_{rs}(t) \tag{7.9b}$$

If $n > K$

the new shortest path is computed among the existing paths $c_{rs}(t) \in K_{rs}^n(t)$. Then, the set $K_{rs}^n(t)$ does not change, $K_{rs}^{n+1}(t) = K_{rs}^n(t)$, and

$$x_{rsp}^{n+1}(t) = \begin{cases} \frac{1}{n+1} * X_{rs}(t) & if\ p \neq c_{rs}(t) \\ \\ \frac{n}{n+1} * x_{rsp}^{n}(t) + \frac{1}{n+1} * X_{rs}(t) & if\ p = c_{rs}(t) \end{cases} \quad \forall p \in K_{rs}^{n}(t), \forall (r,s) \in I, t \in [0,T] \tag{7.10}$$

However, the possibility of repeating shortest paths from one iteration to the next to keep a maximum K of different shortest paths in a proper implementation of the algorithm implies a requirement: that the number of iterations n is defined for any OD pair and time interval.

All the approaches for DUE based on simulation procedures for the network loading process are, therefore, heuristic in nature. Thus, no formal proof of convergence can be provided. However, a convergence criterion is necessary. In this context, a way to empirically determine if the solution reached can be interpreted in terms of a DUE, in the mentioned sense that "*the actual travel time experienced by travelers departing at the same time are equal and minimal*", can be based on an ad hoc version of the *Relative Gap Function* proposed by Janson (1991):

$$Rgap(n) = \frac{\sum_t \sum_{(r,s)\in I} \sum_{p \in K_{rs}(t)} x_{rsp}^{n}(t) * \left[tt_{rsp}^{n}(t) - \theta_{rs}^{n}(t) \right]}{\sum_t \sum_{(r,s)\in I} X_{rs}(t) * \theta_{rs}^{n}(t)} \tag{7.11}$$

where $x_{rsp}^{n}(t)$ is the flow on path p from r to s departing at time t at iteration n. The difference $tt_{rsp}^{n}(t) - \theta_{rs}^{n}(t)$ measures the excess cost experienced because of using a path of cost $tt_{rsp}^{n}(t)$ instead of the shortest path, with cost $\theta_{rs}^{n}(t)$, at iteration n. The ratio measures the total excess cost with respect to the total minimum cost if all travelers would have used the shortest paths.

7.1.2 Dynamic Network Loading

Once the path flows have been estimated, the next step in the DTA determines how these flows propagate across the network along the assigned paths. Thus, it yields travel times as a function of flows and accounting for their temporal profiles (Xu et al. 1999). The procedures to achieve this goal are precisely the DNL methods, which have been, and still are, a fertile research domain. In fact, a wide variety of DNL have been already proposed. Carey and Ge (2012) or Han et al. (2019) provide comprehensive overviews about them. Some of these methods, for example, those in Friesz et al. (1993), Wu et al. (1998b), or Xu et al. (1999), assume travel time functions of the form $tt_{ij}\left(x_{ij}^{t}\right) = f_{ij}\left(x_{ij}^{t}\right)$, where $f_{ij}\left(x_{ij}^{t}\right)$ is the travel time function for link (ij) that provides the travel time tt_{ij} to traverse the link as a function of x_{ij}^{t}, i.e., the flow in link (ij) at time t.

However, most of the DNL currently used both in research as well as in the professional practice are based on a mesoscopic modeling of traffic flow dynamics.

This is a simplification that, while capturing the essentials of the dynamics, is less data demanding and computationally more efficient than microscopic models, which emulate the dynamics of traffic flows from the detailed dynamics of each vehicle. Mesoscopic approaches sometimes combine microscopic aspects in a simplified way (basically, they can deal with individual vehicles) with macroscopic aspects (e.g., those directly concerning the flow dynamics). There are two main approaches to mesoscopic traffic simulation. First, those in which individual vehicles are not taken into account, and vehicles are grouped into packages or multivehicle platoons that move along the links. This is, for example, the case in CONTRAM (Leonard et al. 1989). Second, those in which flow dynamics are determined by simplified dynamics of individual vehicles. DYNASMART (Jayakrisham et al. 1994), DYNAMIT (Ben-Akiva et al. 1997, 2001,2002,2010), Dynameq (Mahut 2000; Florian et al. 2001, 2002; Mahut et al. 2003a, b, 2004; Mahut and Florian 2010), MEZZO (Burghout 2004; Burghout et al. 2005), or Aimsun (Casas et al. 2010) are well-known examples.

From a methodological point of view, the simulation approach of mesoscopic modeling lays in the way it deals with time. The most common approaches are based on synchronous timing, that is, time-oriented simulations in which time in the model progresses according to an appropriately chosen time unit Δt, also known as the simulation step. This is the case of DYNASMART and DynaMIT. Other approaches are asynchronous or event-based. That is, the state of the model changes when some events occur. Thus, time advances in variable amounts. Dynameq and MEZZO are examples of event-based mesoscopic traffic simulators.

One of the main phenomena determining the time evolution of traffic flows across the network are vehicle queues and their backward propagation (or spillback). As the finite-difference approximations to the fluid flow models in terms of the theory of kinematic waves (LWR, Lighthill and Whitham 1955; Richards 1956), satisfactorily reproduces that dynamics, it has been quite natural to use it to develop DNL models. One of the first was the Cell Transmission Model (CTM) proposed by Daganzo (1994, 1995a), which has been extensively used by other authors (e.g., Lo and Szeto 2002; Szeto and Lo 2004). This model assumes a triangular or trapezoidal flow-density function. Daganzo (1995b) developed a second model similar to the CTM, in this case a Finite-Difference Approximation Method (FDAM), which assumes a general nonlinear flow-density function. This FDAM can be used for network loading in the same way as the Cell Transmission Model for networks in Daganzo (1995a). These basic models exhibit limitations, namely in the case of urban networks, since they only account for flow dynamics in links. This means that they do not explicitly deal with intersections and more in particular with signalized intersections, quite usual in urban networks. In this context, Bellei et al. (2005) propose a DUE approach that is an extension of the CTM. This approach, described theoretically in detail in Gentile et al. (2007), is the basis for the General Link Transmission Model (GLTM), which can deal with any concave fundamental diagram and node topology. The road network is modeled in terms of an oriented graph $G = (N, A)$, where N is the set of nodes, each one representing an intersection and where links A, connecting two intersections, converge or diverge. The forward and backward stars of each node identify the set of links converging or diverging to/from it. The GTLM link model

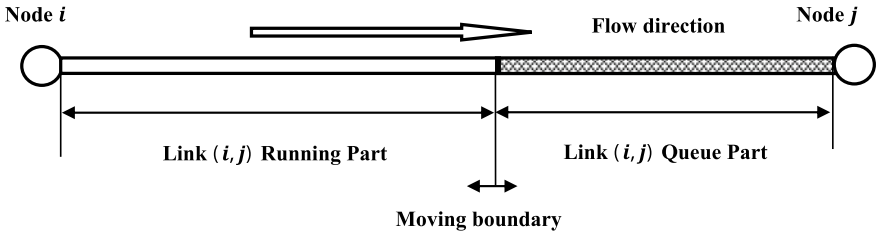

Fig. 7.2 Link model

provides the main input to the node model in terms of the incoming flows. The output from the node model is the outflows that constitute the main input for the link model (Gentile et al. 2010, 2015).

This modeling approach has also been used in many other developments that model the link, explicitly or implicitly, splitting it into two parts: the running part and the queueing part (Fig. 7.2). The running part is that where vehicles are not yet delayed by the queue spillback at the downstream node, where the capacity is limited by stop or give way signs, or traffic lights.

Nodes are modeled according to the interactions between traffic flows at intersections, as node transfer modules, or according to a queue server approach, explicitly accounting for traffic lights and the delays that they cause (Mahmassani et al. 1994). In this case, a simplified car-following model compatible with the macroscopic speed-density relationship on the link approximates the individual vehicle dynamics in the running part. This speed is used to estimate the earliest time at which the target vehicle could exit the link, unless it is affected by the queue spillback when reaching the border between the running part and the queueing part. Vehicle dynamics are then ruled by the queue discharging process. The boundary between the running part and the queueing part is dynamic, according to the queue spillback and queue discharge processes.

Various solutions have been proposed for simulating flow dynamics in the link running part in a simplified way. In essence, they solve the continuity equation of traffic flow:

$$\frac{\partial q}{\partial x} + \frac{\partial k}{\partial t} = g(x, t) \tag{7.12}$$

$$q(x, t) = k(x, t) * u(x, t) \tag{7.13}$$

Link densities are determined by solving the finite differences form of the continuity Eq. (7.12). This can be done using a suitable approach, as, for example, CTM or GTLM, and a functional form (7.13) of the fundamental diagram, where $q(x, t)$ is the flow, $k(x, t)$ the density, $u(x, t)$ the spatial speed and $g(x, t)$ a flow generation term, all of them at time t in x. Jayakrisham et al. (1994) solve these equations in DYNASMART given the densities and the in- and outflows for each section at each

time step and assuming that section speeds are calculated from the densities using the modified Greenshields (1934) speed-density relationship (Eq. 7.14):

$$u_i^t = \left(u_f - u_0\right) * \left(1 - \frac{k_i^t}{k_{jam}}\right)^{\alpha} + u_0 \tag{7.14}$$

where u_i^t and k_i^t are, respectively, the mean speed and density in section i at time step t, u_f and u_0 are the mean free speed and the minimum speed, k_{jam} is the jam density and α is a parameter that captures speed sensitivity to density. DYNAMIT (Ben-Akiva et al. 2001, 2010) includes a speed-density relationship (7.15) that generalizes the one proposed by May and Keller (1967) including a lower bound limiting density, k_{min}, and a second parameter β to capture speed sensitivity to concentration:

$$u = \begin{cases} u_f & if\ k < k_{\min} \\ u_f * \left[1 - \left(\frac{k - k_{min}}{k_{jam}}\right)^{\alpha}\right]^{\beta} & otherwise \end{cases} \tag{7.15}$$

More in particular, the link speed is modeled assuming that it is constant on the upstream section of the link, changes along a deceleration zone covering a downstream section, and varies linearly as a function of the position in this section. According to this assumption, v_u is the speed at the upstream end of the link and the one that is a function of the average density on its running part. That is, v_u is determined by Eq. 7.15. For their part, v_d is the speed at the downstream end of the segment and L_s is the length of the deceleration zone. L_s depends on the geometry of the segment and on traffic conditions. Ben-Akiva et al. (2001) propose a way to determine L_s that is consistent with the empirical evidence that the majority of delays are related to queuing. Finally, assuming that the target link starts at position 0 and ends at position L (i.e., L is the length of the segment), the speed function at an intermediate point x in the segment can be written as follows (Eqs. 7.16 and 7.17):

$$\boldsymbol{v}(x) = \begin{cases} v_d & if\ 0 \le x \le L - L_s \\ \lambda * (x * L) + v_d & if\ L - L_s \le x \le L \end{cases} \tag{7.16}$$

where

$$\lambda = \frac{v_d - v_u}{L_s} \tag{7.17}$$

Other models like MEZZO (Burghout 2004; Burghout et al. 2005) complement this approach according to empirical evidence establishing that there are two limiting densities $k_{\min}$ and $k_{\max}$, which delimit the range in which speed is still a function of the density (del Castillo and Benitez 1995; Eq. 7.18):

$$u = \begin{cases} u_f & \text{if } k < k_{\min} \\ u_0 + (u_f - u_0) * \left[1 - \left(\frac{k - k_{\min}}{k_{\max} - k_{\min}}\right)^{\alpha}\right]^{\beta} & \text{if } k \in [k_{\min}, k_{\max}] \\ u_{\min} & \text{if } k > k_{k\max} \end{cases} \tag{7.18}$$

$u_{\min}$ is the minimum speed in congested conditions. Various queueing models have been proposed to calculate the waiting times in the queuing part of the link. That is, the delays incurred by vehicles because of the output and acceptance capacities of the links. These, respectively, determine the rate at which vehicles can leave the link and how many vehicles can enter it depending on the available space. Obviously, when the acceptance capacity of a link is zero no more vehicles can enter the segment and spillbacks occur. A good example that illustrates this idea is the simplified model in DynaMIT (Ben-Akiva et al. 2001), which considers that the delay of the $i-$ th vehicle in the queue is given by Eq. 7.19:

$$\frac{i}{\rho} \tag{7.19}$$

where ρ is the output capacity of the link. Then, during a time period of length t, $\rho * t$ vehicles will leave the queue. A vehicle in the running part that at time t reaches the end of the queue will find it at $lq(t)$, length of queue at time t, given by

$$lq(t) = lq_0 + l_{\text{eff}} * (\rho * t - m) \tag{7.20}$$

In Eq. 7.20, lq_0 is the position of the end of the queue at time $t = 0$, l_{eff} is the effective length of the queue (i.e., the physical length plus headways), and m is the number of vehicles that reached the queue before the considered vehicle. Obviously, the model is relevant only when $0 < lq(t) < L$.

A completely different approach is taken in Dynameq (Mahut and Florian 2010). It is based on a simulation model that moves vehicles individually, according to a simplified car-following model. In this model, given two consecutive vehicles, the leader vehicle n and the follower $n + 1$, the position $x_{n+1}(t)$ of the follower at time t relative to the position of the leader at $x_n(t - T)$ is estimated according to Eq. 7.21:

$$x_{n+1}(t) = \text{Min}\left[x_{n+1}(t - \varepsilon) + \varepsilon u_f, x_n(t - T) - l_{\text{eff}}\right] \tag{7.21}$$

where T is the reaction time, u_f the free-flow speed, l_{eff}, as before, the effective vehicle length and ε an arbitrary short time interval. The first term inside the minimizing operator represents the farthest position downstream that can be attained at time t based on the follower's position at time $(t - \varepsilon)$, as constrained by the maximum speed of the vehicle, u_f. The second term inside this operator represents the farthest position downstream that can be attained based on the trajectory of the next vehicle downstream in the same lane, according to a simple collision-avoidance rule (Mahut

1999, 2001; Newell 2002). It is a simplified model that only depends on the free-flow speed and does not account for accelerations. It can be considered a lower-order model, since it only defines the position of each vehicle in time, rather than vehicle speed or acceleration.

The solution of the car-following relationship (Eq. 7.21) for time results in (Eq. 7.22):

$$t_{n+1}(x) = \text{Max}\left[t_{n+1}(x-\delta) + \frac{\delta}{u_f}, t_n(x + l_{\text{eff}}) + T\right] \tag{7.22}$$

This relationship in Eq. 7.22 enables the event-based simulation approach used in Dynameq, because it is possible to derive the following expression in Eq. 7.23. It calculates the link entrance and exit times for each vehicle:

$$t_{n+1}(L_1) = \text{Max}\left[t_{n+1}(0) + \frac{L_1}{u_f^1}, t_n(L_1) + T + \frac{l_{\text{eff}}}{\min\left[u_f^1, u_f^2\right]}, t_{n+L_2/l_{eff}}(L_2) + \frac{L_2}{l_{\text{eff}}} * T\right] \tag{7.23}$$

where L_1 and L_2 are the lengths of two sequential links with speeds u_f^1 and u_f^2, respectively. The vehicle attributes represented by l_{eff} and T are considered identical over the entire traffic stream, and each vehicle adopts the link-specific free-flow speed when traversing a given link. The link lengths are assumed to be integer multiples of the vehicle length, l_{eff}. It can be shown (Mahut 2000) that this model yields the triangular fundamental flow-density diagram (Daganzo 1994). The main events changing the state of the model are the arrivals of vehicles to links, their link departures or transfers from one link to the next, according to the turning movements at intersections.

This one-lane link model can be extended to multilane links, including lane changing decisions and additional terms to (7.23) to account for conflicts at nodes with multiple outgoing links. Details can be found in Florian et al. (2008), Mahut and Florian (2010).

The summary description of the most common DTA and DUE included in this section has shown how they can provide TMS with useful information. On the one hand, with the inputs allowing them to estimate the network traffic state. On the other hand (and what is even more relevant), with the necessary outputs to predict traffic flows and travel times on road networks. Moreover, this prediction accounts for their evolution over time because of various factors, being one of them particularly relevant: the time variation of the demand. The main pending question at this point is how to provide this time variation of the traffic demand that constitutes the main input to DTA or DUE. In other words, how to estimate OD-matrices.

7.2 The Static Formulation of the OD-Estimation Problem

Traffic assignment models aim at estimating traffic flows in the network assigning a trip OD-matrix to it, in terms of a route choice mechanism. Therefore, OD trip matrices become their major data input to describe the patterns of traffic behavior across this network. All formulations of static traffic assignment models (e.g., Florian and Hearn 1995), as well as the dynamic ones (e.g., Ben-Akiva et al. 2001), assume that a reliable estimate of an OD is available. However, OD-matrices are not directly observable yet, especially in the case of the time-dependent OD-matrices that are necessary for DTA models. Consequently, it has been natural to resort to indirect estimation methods. These are the matrix adjustment methods, whose main modeling hypothesis can be stated as follows: "*if traffic flows in the links of a network are the consequence of the assignment of an OD matrix to a network, and if we are capable of measuring link flows, the problem of estimating the OD matrix that generates such flows can be considered as the inverse of the assignment problem*" (Cascetta 2001). In other words, the traffic assignment problem is defined as the direct problem, i.e., "*given the O/D matrix X and the cost conditions for using links on the road network, the user equilibrium assignment problem estimates the user equilibrium flows Y on the links of the road network*" (Eq. 7.24):

$$Y = \text{Assignmt}(X) \tag{7.24}$$

where Y is the set of all link flows, X is the OD-matrix, and *Assignmt* is an equilibrium assignment algorithm assigning the OD-matrix X to the network. The reciprocal problem would be that of estimating, from the observed link flows y_l, the OD-matrix X that originated them. In other words, the reciprocal problem of traffic assignment, as described by Cascetta (2001), consists in "*assuming that the observed flows* y_l *on a subset* $\hat{L} \subseteq L$ *of links in the network (or in all links) constitute an user equilibrium flow pattern as defined by* Wardrop (1952)*, determining the OD matrix X whose assignment would produce the observed flows* y_l*"*. Formally, this implies that (Eq. 7.25)

$$X = \text{Assignmt}^{-1}(Y) \tag{7.25}$$

Since the earlier formulation proposed by van Zuylen and Willumsen (1980), the matrix adjustment problem has been a relevant research and practical problem. Given a road network $G = \{L, N\}$, with a set of links L, a set of nodes N, and a set I of OD pairs, the OD-matrix estimation problem consists in finding a feasible vector (OD-matrix) $X \in \Omega$, where Ω is the set of all feasible OD-matrices. For their part $X = \{X_i\}, i \in I$, are the demands for all OD pairs, being $I = \{\text{setofall}OD\text{pairsinthenetwork}\}$. (r, s), as introduced in Sect. 7.1, stands for the $i - th$ OD pair. The assignment of the OD-matrix explains the observed flows y_l on a subset $\hat{L} \subseteq L$ of links equipped with counting stations. It is usually accepted

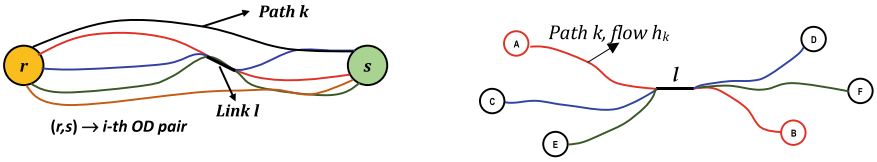

Fig. 7.3 Possible link-path relationships

that the assignment of the OD-matrix to the links of the network is made according to an assignment proportion matrix $P = \{p_{il}\}, \forall i \in I, \forall l \in L$, where each element p_{il} in the matrix is defined as the proportion of the OD demand X_i that uses link l. The notation $P = P(X)$ remarks that, in general, these proportions depend on the demand.

The hypotheses supporting the approach are illustrated in Fig. 7.3, which depicts possible positions of a hypothetical detector at a link l.

Let y_l be the flow measured by one detector and h_k the flow on path k to which this link belongs. If φ_{ik} is the fraction of the demand of the ith OD pair X_i, the flow h_k is given by Eq. 7.26:

$$h_k = \varphi_{ik} * X_i \tag{7.26}$$

δ_{lk} is the link-path assignment matrix, taking the following values (Eq. 7.27):

$$\delta_{lk} = \begin{cases} 1 & \text{if link } l \text{ belongs to path } k : l \in \text{Path } k \\ 0 & \text{otherwise} \end{cases} \quad \forall l \in L, k \in K_i, i \in I \tag{7.27}$$

where K_i = {Set of all paths connecting the ith OD pair}. The relationship between the measured flow y_l on link l and the flows h_k on the paths using link l is given by Eqs. 7.28 and 7.29:

$$y_l = \sum_{i \in I} \sum_{k \in K_i} h_k * \delta_{lk} = \sum_{i \in I} \sum_{k \in K_i} \varphi_{ik} * \delta_{lk} * X_i = \sum_{i \in I} p_{il} * X_i \tag{7.28}$$

$$p_{il} = \sum_{k \in K_i} \varphi_{ik} * \delta_{lk} \tag{7.29}$$

When assigned to the network, the OD-matrix induces a flow $Y = \{y_l\}, \forall l \in L$ in its links. If we assume that observed flows $\widehat{Y} = \{y_l\}$ are available for a subset $\widehat{L}$ of the links, $l \in \widehat{L} \subseteq L$, and that a target matrix $X^H \in \Omega$ is also available, the generic OD-matrix estimation problem can be formulated (Lundgren and Peterson 2008) as (Eq. 7.30):

$$\text{Min}_{XY}(X, Y) = w_1 * F_1\left(X, X^H\right) + w_2 * F_2\left(Y, \widehat{Y}\right)$$

$$s.t. \sum_{i \in I} p_{il}\left(X^H\right) * X_i = \hat{y}_l, \forall l \in \widehat{L} \tag{7.30}$$

$$X \in \Omega$$

The functions $F_1\left(X, X^H\right)$ and $F_2\left(Y, \widehat{Y}\right)$, respectively, represent generalized distance measures. The first one that between the estimated OD-matrix X and the given target matrix X^H, and the second one between the estimated link flows Y and the observed link flows $\widehat{Y}$. The parameters w_1 and w_2 reflect the relative belief (or uncertainty) in the information contained in X^H and $\widehat{Y}$. The problem expressed in Eq. 7.30 can be interpreted as a two-objective optimization problem, being precisely these objectives F_1 and F_2, whereas w_1 and w_2 are the corresponding weighting factors.

The set Ω of feasible OD-matrices normally includes the non-negative OD-matrices. However, it can also be limited to those matrices within a certain deviation from the target values (Eq. 7.31), i.e.,

$$\Omega = \left\{X \geq 0 \middle| (1-\alpha) * X^H \leq X \leq (1+\alpha) * X^H\right\} \tag{7.31}$$

for some suitable parameter $\alpha > 0$ stating the tolerance level. An analogous formulation can be used to state, instead, a maximum deviation from the link flow observations with a tolerance parameter $\beta > 0$ (Eq. 7.32):

$$\Omega = \left\{X \geq 0 | (1-\beta) * \hat{y}_l \leq y_l \leq (1+\beta) * \hat{y}_l\right\} \tag{7.32}$$

Another possibility is to restrict the total travel demand in all OD pairs originating or terminating at a certain node. This is the four-step demand model (Ortúzar and Willunsen 2011), which makes an adjustment of the trip distribution with respect to the trip generation. In any case, all these constraints on Ω are linear or convex and can be easily handled from the optimization point of view.

Obviously, the resulting OD-matrix is dependent on the objective function minimized in (7.30), that is, on the distance measure chosen. One of the distances initially proposed, probably as an analogy with the trip distribution problem, was the maximum entropy function. It was derived from the principle of minimum information (van Zuylen and Willumsen 1980) and is expressed as in Eq. 7.33:

$$F_1\left(X, X^H\right) = \sum_{i \in I} X_i * \left\{\log \frac{X_i}{X_i^H} - 1\right\} \tag{7.33}$$

The function F_2 in (7.30) can be formulated in a similar way.

A type of objective function that is becoming very used in these models is the one based on the least squares formulation. This is equivalent to assume a Euclidean distance function between observed and estimated variables. For example, the function F_2 for the observed volumes would correspond to Eq. 7.34:

$$F_2\left(Y, \widehat{Y}\right) = \sum_{l \in \widehat{L}} \left(y_l - \hat{y}_l\right)^2 \tag{7.34}$$

and could be weighted using the information on the significance of each observation. For instance, when the measurements contained in y are computed as means from a set of observations for each link, the variance σ_l^2 can be used as a measure on how important each link observation is. Equation 7.34 would be then reformulated as Eq. 7.35:

$$F_2\left(Y, \widehat{Y}\right) = \sum_{l \in \widehat{L}} \frac{1}{\sigma_l^2} * \left(y_l - \hat{y}_l\right)^2 \tag{7.35}$$

One disadvantage of the entropy maximizing approaches as formulated in Eq. 7.33 lies in the treatment of link flow observations as error-free constraints (Bell and Iida 1997). An attempt to overcome this disadvantage consists in using a generalized least squares approach to provide a framework accounting for errors from various sources. The method, first proposed by Cascetta (1984), also yields standard errors for the trip table, thereby indicating the relative robustness of the fitted values. The equivalent problem, assuming that the weighting factors w_1 and w_2 have the same value, takes the following form (Eq. 7.36):

$$\begin{aligned} \mathrm{Min}_X F(X) = & \frac{1}{2} * \left[\left(X - X^H\right)^T * \left(X_C^H\right)^{-1} * \left(X - X^H\right)\right] \\ & + \frac{1}{2} * \left[\left(\widehat{Y} - P\left(X^H\right) * X\right)^T * (Y_C)^{-1} * \left(\widehat{Y} - P\left(X^H\right) * X\right)\right] \end{aligned} \tag{7.36}$$

The inputs are prior estimates of OD flows, X^H, link flow measurements, $\widehat{Y}$, variance–covariance matrices of the prior estimates and link flow measurements, respectively, X_C^H and Y_C and the matrix of link choice proportions $P(X^H)$. As the variance–covariance matrices are positive definite and the objective function is convex, the minimum is uniquely given by (Eq. 7.37):

$$\begin{aligned} \nabla F\left(X^*, Y^*\right) = & \left(X_C^H\right)^{-1} * \left(X^* - X^H\right) \\ & - P^T\left(X^H\right) * (Y_C)^{-1} * \left(\widehat{Y} - P\left(X^H\right) * X^*\right) = 0 \end{aligned} \tag{7.37}$$

This yields the following linear estimator (Eq. 7.38):

$$X^* = \left[\left(X_C^H\right)^{-1} + P^T\left(X^H\right) * (Y_C)^{-1} * P\left(X^H\right)\right]^{-1} * \left[\left(X_C^H\right)^{-1} * X^H + P^T\left(X^H\right) * (Y_C)^{-1} * \hat{Y}\right] \tag{7.38}$$

For their part, the sensitivities of this factor are given by Eq. 7.39:

$$\Delta X^* = \left[\left(X_C^H\right)^{-1} + P^T\left(X^H\right) * (Y_C)^{-1} * P\left(X^H\right)\right]^{-1} * \left[\left(X_C^H\right)^{-1} * \Delta X^H + P^T\left(X^H\right) * (Y_C)^{-1} * \Delta\hat{Y}\right] \tag{7.39}$$

Additionally, taking into account that X^H and y are uncorrelated and assuming that $E = \left[\left(X_C^H\right)^{-1} + P^T\left(X^H\right) * (Y_C)^{-1} * P\left(X^H\right)\right]^{-1}$, the variance of X^* is given by Eq. 7.40:

$$\mathrm{Var}\left\{X^*\right\} = E\left(X_C^H\right)^{-1} * E + E * P^T\left(X^H\right) * (Y_C)^{-1} * P\left(X^H\right) * E \tag{7.40}$$

Unlike the maximum entropy model, there is nothing to prevent negative fitted values for the OD flows being produced by the generalized least squares estimator. While negative values would reflect small real values, they are nonetheless counterintuitive. Bell (1991) has also considered the introduction of non-negativity constraints for the fitted OD-matrix.

7.3 Bi-level Optimization Models for OD Adjustment

The estimation of OD-matrices from observed flows as the reciprocal of the assignment problem is a highly undetermined problem. That is, there are in general many OD-matrices, which, when assigned to the network, induce equivalent link flows. The objective function and the set of constraints in the formulation of the problem are aimed at reducing this indetermination. However, this simple formulation can still have some drawbacks. The set of constraints in the generic problem formulation (Eq. 7.30) to determine X is expressed by Eq. 7.41:

$$\sum_{i \in I} p_{il}\left(X^H\right) * X_i = \hat{y}_l, \forall l \in \hat{L} \tag{7.41}$$

$$s.t.\ X \in \Omega$$

It consists of one equation for every link flow observation. Thus, it is an undetermined equation system, as long as the number of OD pairs $|I|$ is greater than the number of link flow observations $\left|\hat{L}\right|$. This fact is especially true for large real-world networks. Additionally, the information transferred through the equation system is delimited by topological dependencies. A basic principle in network flows is that, for consistent flows, the balance equations must hold. In other words, the sum of

incoming and outgoing flows at any intermediate node must be zero. This principle, which can also be interpreted in physical terms using Kirchhoff's law, means that, for each intersection, at least one link flow is linearly dependent from the others. This results in a row-wise dependency for the equation system.

On the other hand, the elements $p_{il}\left(X^H\right)$ are non-zero because they are part of one or more shortest paths for OD pairs $i \in I$. However, since every subpath of a shortest path is a shortest path, every pair of nodes along a certain shortest path is connected through parts of this shortest path. This results in a column-wise dependency for the equation system. Thus, we can conclude that the equation system (Eq. 7.31) is most likely not fully ranked, which further increases the freedom of choice for the OD-estimation problem. Therefore, the way of determining $p_{il}\left(X^H\right)$ is crucial for the quality of the OD-matrix estimation model. This is usually done depending on how the assignment matrix $P(X^H)$ is calculated, and whether it is dependent of X or not. In other words, if the route choices are made depending on the congestion or not. If the assignment of the OD-matrix to the network is independent of the link flows, that is, if we have an uncongested network,$P\left(X^H\right) = P$ is a constant matrix. In that case, the first set of constraints in Eq. 7.41 can be reformulated as in Eq. 7.42:

$$\sum_{i \in I} p_{il} * X_i = \hat{y}_i \ \forall l \in \hat{L} \tag{7.42}$$

$$s.t.\ X \in \Omega$$

In addition, this substitution can be directly performed in the objective, i.e., in the function $F_2\left(Y, \widehat{Y}\right)$, which reduces the OD-matrix estimation to a problem only in the variable X. Assuming that the deviation measures F_1 and F_2 are convex and that the set of feasible OD-matrices Ω is linear or, at least, convex, the OD-estimation problem can be easily solved with some suitable standard algorithms for nonlinear programming. This is the usual approach in most cases (van Zuylen and Willumsen 1980). However:

> The assumption that the assignment, i.e., the route choice, is independent of the load on the links is only realistic in a network with a very low congestion rate or in networks where, in practice, only one route can be used.

If we assume that the network is congested and that the routes are chosen depending on the current travel times, the route proportions are in turn dependent on the existing traffic situation. For its part, this situation depends on the OD-matrix. Thus, the relationship between the route proportions P and the OD-matrix X can only be defined implicitly. In this case, a plausible hypothesis is to assume that the choice proportions can be derived from a traffic assignment model. Then, the set of feasible solutions to the estimation problem (Eq. 7.30) is defined as all points (X, Y) in which Y is the link flow solution satisfying an assignment of the corresponding demand $X \in \Omega$. In this case, the generic OD-matrix estimation problem (Eq. 7.30) can be reformulated as a bi-level optimization problem. Bell and Iida (1997) propose

an approach based on the hypothesis that a traffic assignment model can be represented by a function whose input is the OD-matrix X and whose outputs are the link flows Y (Eq. 7.43)

$$Y = A(X) * X \tag{7.43}$$

That is simply a reformulation of the direct assignment problem as defined in Sect. 7.2, in which, given the OD-matrix X, it is possible to find the link flows $\hat{y}$. The reciprocal problem of finding X given y (Eq. 7.25) is not possible, since the inverse of this function does not exist. However, a way of accounting for this functional relationship in the OD-estimation process could be to reformulate the least squares formulation including it explicitly in the model (Eq. 7.44):

$$\begin{aligned}\text{Minimize}_X F(X) = &\frac{1}{2} * \left(X - X^H\right)^T * \left(X_C^H\right)^{-1} * \left(X - X^H\right) \\ &+ \frac{1}{2} * \left[y - A(X) * X\right]^T * (Y_C)^{-1} * \left[\hat{Y} - A(X) * X\right]\end{aligned} \tag{7.44}$$

If the assignment function (Eq. 7.73) is differentiable, then (Eq. 7.45):

$$\nabla F(X) = \left(X_C^H\right)^{-1} * \left(X - X^H\right) - \nabla A(X)^T * (Y_C)^{-1} * \left[\widehat{Y} - A(X) * X\right] \tag{7.45}$$

And if the Jacobian of the assignment function $\nabla A(X)$ is independent of X^H, then (Eq. 7.46):

$$\nabla^2 F(X) = \left(X_C^H\right)^{-1} + \nabla A(X) * (Y_C)^{-1} * \nabla A(X) \tag{7.46}$$

$\nabla^2 F(X)$ is positive definite, since X_C^H and Y_C are variance–covariance matrices, and there is a unique solution to the equivalent optimization problem. Yang (1995) proposes an efficient heuristic approach to solve this bi-level problem.

However, as Florian and Chen (1995) probe, the assignment function is usually not differentiable. Therefore, analytical approaches are of limited usefulness, since they are constrained to simple uncongested cases. Consequently, other formulations have been proposed. The most common formulation of the bi-level OD-matrix estimation problem for the general case is that Eqs. 7.47 and 7.48, respectively, referred to the upper level and to the lower level problem. Equation 7.47 is as follows:

$$\text{Min}_X F(X, Y) = w_1 * F_1\left(X, X^H\right) + w_2 * F_2\left(Y, \widehat{Y}\right) \tag{7.47}$$

$$\text{s.t. } X \in \Omega$$

We want to find the X that minimizes $F(X, Y)$ subject to $X \in \Omega$ under the hypothesis that the induced link flow $\hat{y}$ satisfies the equilibrium assignment conditions

obtained by solving Eq. 7.48, that is, the lower level problem:

$$
\begin{aligned}
Y(X) &= \operatorname{argmin} \sum_{l \in L} \int_0^{y_l} s_l(x)dx \\
s.t. \sum_{k \in K_i} h_k &= X_i, \forall i \in I \\
h_k &\geq 0 \forall k \in K_i, \forall i \in I \\
y_l &= \sum_{i \in I} \sum_{k \in K_i} \delta_{lk} * h_k, \forall l \in L
\end{aligned}
\tag{7.48}
$$

The algorithm for this OD adjustment method based on a bi-level optimization process can be viewed as the calculation of a sequence of OD-matrices, so that the least squares error between traffic counts coming from detectors and traffic flows obtained by a traffic assignment is increasingly reduced. The estimation of the OD-matrix requires information about the routes used for the trips contained in the OD-matrix, X_{rs}. Particularly, it requires the definition of the route and the trip proportions relative to the total trips X_{rs} originated at zone r and ending at zone s. This information is difficult both to handle and to store in traffic databases, considering that the number of routes connecting all OD pairs on a connected network can grow exponentially with the size of the network. This is the reason to use a mathematical programming approach based on a traffic assignment algorithm, which is solved at each iteration without requiring the explicit route definition. The algorithmic scheme to numerically solve the bi-level formulation of the OD-matrix adjustment problem is illustrated in the conceptual diagram in Fig. 7.4. The solution at the $k-$ th iteration of the upper level nonlinear optimization problem for the current estimates of the link flows, $\widehat{Y}^k$, provides a new estimate X^k of the OD-matrix, which is the input to the lower level equilibrium assignment problem. In turn, the solution to this lower level problem updates the link flow estimates. The iterative process continues until certain convergence criterion is satisfied.

One of the first operational approaches of the bi-level algorithm was the one proposed by Spiess (1990), whose bi-level optimization adjustment procedure solves the following bi-level nonlinear optimization problem (Eqs. 7.49 and 7.50):

$$
\text{Min } F[Y(X)] = \frac{1}{2} * \left\{ \sum_{l \in \widehat{L}} \left[y_l(X) - \hat{y}_l \right]^2 \right\} \tag{7.49}
$$

$$
Y(X) = \operatorname{argmin} \sum_{l \in L} \int_0^{y_l} s_l(x)dx \tag{7.50}
$$

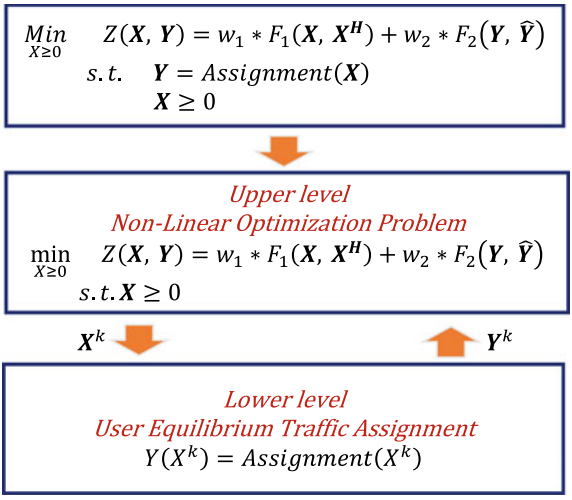

Fig. 7.4 Algorithmic scheme for the bi-level approach to the OD adjustment problem

$$s.t. \sum_{k \in K_i} h_k = X_i, \forall i \in I$$

$$h_k \geq 0 \forall k \in K_i, \forall i \in I$$

$$y_l = \sum_{i \in I} \sum_{k \in K_i} \delta_{lk} * h_k = \sum_{i \in I} X_i \sum_{k \in K_i} \delta_{lk} * p_k, p_k = \frac{h_k}{X_i} \forall l \in L$$

where $y_l(X)$ is the flow on link l estimated by the lower level traffic assignment problem with the adjusted trip matrix X, h_k is the flow on the $k-$th path for the $i-$th O-D pair and $\hat{y}_l$ is the measured flow on link l. I is the set of all OD pairs in the network, and K_i is the set of paths connecting the $i-$th O-D pair. $s_l(y_l)$ is the volume-delay function for link $l \in L$. The algorithm used to solve the problem is heuristic in nature, of steepest descent type, and does not guarantee that a global optimum of the problem will be found. The iterative process for a generic iteration k is as follows:

- Given a solution X_i^k, an equilibrium assignment is solved, yielding link flows y_l^k and proportions $\{p_{il}^k\}$ satisfying the relationship in Eq. 7.51:

$$y_l^k = \sum_{i \in I} p_{il}^k * X_i^k \quad \forall l \in L \tag{7.51}$$

  The target matrix is used in the first iteration (i.e., $X_i^1 = X_i^H, \forall i \in I$).
- The estimate of the OD-matrix at iteration $k+1$ is calculated in terms of the gradient of the objective function $F[Y(X)]$ with Eq. 7.52:

$$X_i^{k+1} = \begin{cases} X_i & \text{for } k = 0 \\ X_i^k * \left[1 - \lambda^k * \left(\frac{\partial F[Y(X)]}{\partial X_i}\right)_{X_i^k}\right] & \text{for } k = 1, 2, 3 \ldots \end{cases} \tag{7.52}$$

That is, a change in the demand is proportional to the demand in the initial matrix and zeroes are preserved in the process.

- The gradient is approximated as in Eq. 7.53:

$$\frac{\partial F[Y(X)]}{\partial X_i} = \sum_{k \in K_i} p_k \sum_{l \in \widehat{L}} \delta_{lk} * \left(\hat{y}_l - y_l\right) \; \forall i \in I \tag{7.53}$$

where $\widehat{L} \subset$ is the subset of links with flow counts and $p_k = \frac{h_k}{X_i}$.

- The step length is approximated as in Eq. 7.54 and 7.55:

$$\lambda^* = \frac{\sum_{l \in \widehat{L}} y_l^{'} * \left(\hat{y}_l - y_l\right)}{\sum_{l \in \widehat{L}} \left(y_l^{'}\right)^2} \tag{7.54}$$

where

$$y_l^{'} = -\sum_{i \in I} X_i * \left(\sum_{k \in K_i} p_k \sum_{l \in \widehat{L}} \delta_{lk} * \left(\hat{y}_l - y_l\right)\right) * \left(\sum_{k \in K_i} \delta_{lk} * p_k\right) \tag{7.55}$$

To ensure the convergence the step length must satisfy the condition in Eq. 7.56:

$$\lambda^* \frac{\partial F[Y(X)]}{\partial X_i} < 1 \; \forall i \in I \tag{7.56}$$

If the condition is violated for some I, the step length must be bounded accordingly (Eq. 7.57):

$$\lambda^* = \frac{1}{\max\limits_i \left\{\frac{\partial F[Y(X)]}{\partial X_i}\right\}} + \varepsilon \tag{7.57}$$

where ε is added to avoid numerical errors.

Further details on the algorithmic properties of this approach are available in Florian and Chen (1995). Alternative approaches improving the simplified gradient approach can be found in Codina and Barceló (2004) and Lundgren and Peterson (2008), among others.

In summary, the most common practices consist in using an initial OD estimate, the OD seed X^H as input, and adjusting it. This adjustment is done based on the

available link counts y provided by an existing layout of traffic counting stations and on other additional information, whenever it is available. Adjustments can be considered as indirect estimation methods based on optimization approaches. All of them share two fundamental modeling hypotheses:

- A mapping scheme of OD flows-link flow counts is available
- If L is the set of links in the network, flow detectors are only located in a subset $\hat{L} \subset L$, from which link flow measurements $\hat{y}_l$, $l \in \hat{L}$ are available.

Assuming these hypotheses, a bi-level optimization model can be proposed, which is usually solved by computational schemes like the one conceptually depicted in Fig. 7.4. That is, iterating between an upper and a lower level. Again, the upper level solves a nonlinear optimization problem that minimizes the distance between available empirical evidence (i.e., a target OD-matrix X^H and observed flows $\widehat{Y}$ in a subset of links) and the estimations provided by the algorithm, while the lower level solves a *User Equilibrium Traffic Assignment* (UETA). The solution to the upper level nonlinear optimization problem provides new estimates of the OD-matrix, which constitute the input to the lower level assignment problem. In turn, the solution to this latter problem provides new estimates of link flows. This computational scheme is in fact a computational framework from which multiple algorithmic variants to solve the problem, both at the upper and at the lower level, can be derived.

The second modeling hypothesis strongly depends on the detection layout available in the network. Unfortunately, they are usually designed and implemented with the primary purpose of providing the data required by traffic control applications. Therefore, current detection layouts in traffic networks are not appropriate for the reconstruction of OD-matrices, as they do not take into account the OD pattern structure explicitly. This could represent a serious drawback regarding the quality of the OD reconstruction, since it has been observed in practice that the adjustment procedure can act implicitly as a metaregression model. That is, it would fit quite well those parts of the network with a relatively rich detection infrastructure (in fact overfit them is most cases), while completely distorting other parts of the network where detection is sparse. This would generate an unbalancing process moving trips between parts of the network, depending on the numerical requirements of the process, but completely unrelated to the underlying transportation phenomena modeled by the OD pattern. In this context, the objective of identifying a detection layout that optimizes the coverage of origin–destination demand on the road network while minimizing the uncertainties of the estimated OD is a subsidiary prior requirement. Since the seminal work of Yang and Zhou (1998), the problem has received substantial attention in recent years, being Ehlert et al. (2006), Fei et al. (2007) just example references. Castillo et al. (2008), who formulate the problem from the perspective of the observability of systems being a *sine qua non* condition for their state estimation and forecasting, must be highlighted. Larsson et al. (2010) provide an overview of the pros and cons of various approaches, and Barceló et al. (2012) complement the detection layout models with a sensitivity analysis, enabling the analyst to establish a relationship between the quality of the layout and the quality of the OD pattern reconstruction.

In consequence, for the practice of the matrix adjustment, it is not only relevant the mathematical modeling approach to be used but it is also highly recommendable to pay attention to the detection layout whose measurements are going to be used for the adjustment of an OD matrix.

7.4 Analytical Formulations for the Dynamic OD Matrix Estimation (DODME) Problem

The static bi-level optimization OD adjustment problem can be reformulated as (Eqs. 7.58 and 7.59):

$$\text{Min } Z(X, Y) = w_1 * F_1\left(X, X^H\right) + w_2 * F_2\left(Y, \widehat{Y}\right) \tag{7.58}$$

$$s.t.\ Y = \text{Assignmt}(X) \tag{7.59}$$

$$X \geq 0$$

where F_1 and F_2, as before, are suitable distance functions between estimated and observed values, while w_1 and w_2 are weighting factors reflecting the uncertainty of the information contained in X^H and $\widehat{Y}$, respectively. The underlying hypothesis is that $Y(X)$ are the link flows predicted by assigning the demand matrix X to 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 $A(X)$, the proportion of OD flow that contributes to a certain link traffic count is (Eq. 7.60):

$$Y = A(X) * X \tag{7.60}$$

This is a bi-level optimization problem that solves (at the upper level) the nonlinear optimization problem by substituting the estimated flows Y in the objective function (Eq. 7.59) using the relationship in Eq. 7.60. Thus, it results in (Eq. 7.61):

$$\text{Min } Z(X, Y) = w_1 * F_1\left(X, X^H\right) + w_2 * F_2\left(A(X) * X, \widehat{Y}\right) \tag{7.61}$$

$$s.t.\ X \geq 0$$

To estimate a new assignment matrix X while at the lower level, a Static User Equilibrium Assignment is used to solve the assignment problem $Y = \text{Assignmt}(X)$, i.e., to estimate the assignment matrix $A(X)$ induced by the new X. Spiess (1990) is a good example of a seminal model based on this approach. Static models have made wide use of the analytical approaches that include flow counts as complementary information to reduce indeterminacy when solving the minimization problem

(Eq. 7.61), as in Codina and Montero (2006), Lundgren and Peterson (2008), and Spiess (1990). The various algorithmic approaches to numerically solve the problem look for algorithmic efficiency, convergence properties, and stability. However, since they are static, they are supported by static assignment models.

In this context, some researchers as Frederix et al. (2011), Lundgren and Peterson (2008), Toledo and Kolechkina (2013), or 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, they proposed either analytical or empirical approaches for improving it. The analytical approaches assume a functional dependency that allows for a Taylor expansion around the current solution. While some authors like Lundgren and Peterson (2008) still derive the expansion from a static traffic assignment, others like Frederix et al. (2013) or Toledo and Kolechkina (2013) propose a dynamic traffic assignment to account for time dependencies. The approaches based on the hypothesis of linear relationships may be invalid when congestions build up in the network, resulting in non-linearities. The dynamic assignment would be more appropriate for working with congestion building processes that would be captured by the analytical expansion of the dynamic assignment matrix. Frederix et al. (2013) offer a relevant theoretical contribution, while Toledo and Kolechkina (2013) provide more insights to apply it to large networks.

A simpler approach is the modification of the Spiess procedure performed by Ros-Roca et al. (2020). They used, on the one hand, a first-order approach to the assignment matrix that is provided by replacing the static assignment at the lower level by a dynamic traffic assignment. On the other hand, 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 following notation is used for the dynamic analytical extension from this point until the end of the chapter:

- I is the set of OD pairs.
- $\mathcal{T} = \{1, \ldots, T\}$ is the set of time intervals.
- L is the set of links in the network. $\widehat{L} \subseteq L$ is the subset of links that have sensors.
- $\hat{y}_{l,t}$ are the measured flow counts at link l during time period t. $y_{l,t}$ are the corresponding simulated flow counts, $\forall l \in \widehat{L} \subseteq L$ and $\forall t \in \mathcal{T}$. $Y = \left(y_{l,t}\right)$ and $\widehat{Y} = \left(\hat{y}_{l,t}\right)$ are the link flow counts in vector form.
- $x_{n,r}$ are the OD flows for $n-$ th OD pairs departing during time period r, $\forall n \in I$ and $\forall r \in \mathcal{T}$.$X = \left(x_{n,r}\right)$ are the OD flows in vector form.
- $a_{n,r}^{l,t}$ is the flow proportion of the $n-$ th OD pair, $n \in I$, departing at time period $r \in \mathcal{T}$ and captured by link $l \in \widehat{L}$ at time period $t \in \mathcal{T}$. $A = \left[a_{n,r}^{l,t}\right]$ is the assignment matrix.

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) $X^* \in \mathrm{G} \subseteq \mathbb{R}_+^{I \times \mathcal{T}}$, where $X^* = \left(x_{n,r}^*\right)$, $n \in I, r \in \mathcal{T}$ consists of the demands for all OD pairs. It can be assumed that the assignment of the time-sliced

OD-matrices to the links of the network should be done according to an assignment proportion matrix $A = \left[a_{n,r}^{l,t}\right], \forall l \in L, \forall n \in I, \forall r, t \in \mathcal{T}$, where each element in the matrix is defined as the proportion of the OD demand $x_{n,r}$ that uses link l at time period t. The notation $A = A(X)$ is used to indicate that, in general, these proportions depend on the demand. The linear relationship between the flow count on a link and the given OD pair has a matrix form, which thus sets the vector of detected flows as $Y = (Y_1, \ldots, Y_T) = (y_{1,1}, \ldots, y_{L,1}, \ldots, y_{1,T}, \ldots y_{L,T})$ and the vector of OD flows as $X = (X_1, \ldots, X_T) = (x_{1,1}, \ldots, x_{N,1}, \ldots, x_{1,T}, \ldots, x_{N,T})$. Expressing this relationship as the matrix product (Eq. 7.42), $A(X)$ is now (Eq. 7.62):

$$A(X) = \begin{pmatrix} A^{1,1} & 0 & \cdots & 0 \\ A^{1,2} & A^{2,2} & 0 & \vdots \\ \vdots & \ddots & \ddots & 0 \\ A^{1,\mathrm{T}} & \cdots & A^{T-1,T} & A^{T,T} \end{pmatrix} \text{where} A^{r,t} = \begin{pmatrix} a_{1,r}^{1,t} & \cdots & a_{N,r}^{1,t} \\ \vdots & \ddots & \vdots \\ a_{1,r}^{L,t} & \cdots & a_{N,r}^{L,t} \end{pmatrix} \tag{7.62}$$

$a_{n,r}^{l,t}$ represents the proportion of OD flow departing at time $r, x_{n,r}$, passing through link l at time t, $y_{l,t}$. $A^{r,t}$ represents the assignment matrix for the departing flows at time window r detected at time window t. Therefore, A is a lower-diagonal matrix, because OD flows departing at time r cannot pass through link l at time $t < r$.

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 (Frederix et al. 2011, 2013; Toledo and Kolechkina 2013). Let X' be in the neighborhood of X. Then, the Taylor expansion is (Eq. 7.63):

$$\begin{aligned} y_{l,t} &= \sum_{n\in I}\sum_{r=1}^{t} a_{n,r}^{l,t}(X') * x'_{n,r} + \sum_{n\in I}\sum_{r=1}^{t} \frac{\partial y_{l,t}(X')}{\partial x_{n,r}} * \left(x_{n,r} - x'_{n,r}\right) = \\ &= \sum_{n\in I}\sum_{r=1}^{t} a_{n,r}^{l,t}(X') * x'_{n,r} \\ &\quad + \sum_{n\in I}\sum_{r=1}^{t} \left.\frac{\partial\left[\sum_{n\in I}\sum_{r=1}^{t} a_{n,r}^{l,t}(X') * x_{n,r}\right]}{\partial x_{n,r}}\right|_{X'} * \left(x_{n,r} - x'_{n,r}\right) = \\ &= \sum_{n\in I}\sum_{r=1}^{t} a_{n,r}^{l,t}(X') * x'_{n,r} \\ &\quad + \sum_{n\in I}\sum_{r=1}^{t} \left(x_{n,r} - x'_{n,r}\right) * \left[\sum_{n'\in I}\sum_{r'=1}^{t} \left.\frac{\partial a_{n',r'}^{l,t}(X')}{\partial x_{n,r}}\right|_{X'} * x'_{n',r'}\right] \end{aligned} \tag{7.63}$$

This enables redefining Spiess' approach to the dynamic case by simply using the first term in the above Taylor expansion. It does not account for the propagation effects, but it explicitly considers time dependencies. The traffic assignment problem at the lower level must now be a dynamic traffic assignment (DTA). Then, the time

periods for the entire formulation must be considered as follows (Eqs. 7.64 and 7.65):

$$\text{Min } Z(X) = \frac{1}{2} * \sum_{t\in\mathcal{T}} \sum_{l\in\hat{L}} \left(\left(\sum_{n\in I} \sum_{r=1}^{t} a_{n,r}^{l,t} * x_{n,r} \right) - \hat{y}_{l,t} \right)^2 \tag{7.64}$$

$$s.t.\ a_{n,r}^{l,t} = \text{Assignment}(X) \tag{7.65}$$

$$x_{n,r} \geq 0$$

where $a_{n,r}^{l,t}$ is the assignment matrix described before. Therefore, the linear combination inside the brackets is the simulated flow $y_{l,t}$, applying (Eq. 7.66):

$$\frac{\partial y_{l,t}}{\partial x_{n,r}} = a_{n,r}^{l,t} \tag{7.66}$$

As in Spiess (1990), the chain rule can be used to obtain the gradient of the objective function (Eq. 7.67):

$$\frac{\partial Z}{\partial x_{n,r}} = \sum_{t\in\mathcal{T}} \sum_{l\in\hat{L}} \frac{\partial y_{l,t}}{\partial x_{n,r}} * \left(y_{l,t} - \hat{y}_{l,t} \right) = \sum_{t\in\mathcal{T}} \sum_{l\in\hat{L}} a_{n,r}^{l,t} * \left(y_{l,t} - \hat{y}_{l,t} \right) \tag{7.67}$$

We obtain similar equations finding the optimal step size by using the same procedure (Eq. 7.68):

$$y_{l,t}^{'} = \frac{dy_{l,t}}{d\lambda} = \sum_{r=1}^{t} \sum_{n\in I} \frac{dx_{n,r}}{d\lambda} * \frac{\partial y_{l,t}}{\partial x_{n,r}} = \sum_{r=1}^{t} \sum_{n\in I} -x_{n,r} * \frac{\partial Z}{\partial x_{n,r}} * \frac{\partial y_{l,t}}{\partial x_{n,r}} \tag{7.68}$$

The optimal step length λ can be calculated solving the 1-dimensional optimization problem in Eq. 7.69 and whose solution is given by Eq. 7.70:

$$Z^{'}(\lambda) = \sum_{t\in\mathcal{T}} \sum_{l\in\hat{L}} y_{l,t}^{'} * \left(\tilde{y}_{l,t} - \hat{y}_{l,t} + \lambda * y_{l,t}^{'} \right) = 0 \tag{7.69}$$

$$\lambda^{*} = \frac{-\sum_{t\in\mathcal{T}} \sum_{l\in\hat{L}} y_{l,t}^{'} * \left(y_{l,t} - \hat{y}_{l,t} \right)}{\sum_{t\in\mathcal{T}} \sum_{l\in\hat{L}} {y_{l,t}^{'}}^{2}} \tag{7.70}$$

Then, the iterative procedure described by Spiess (1990) can be used in DTA using these new equations, which are expanded with the time windows. In addition, this procedure can be improved by adding a second term in the objective function to compare it with a historical OD-matrix. If the quadratic function is used, and replacing w_1 and w_2 by $w = w_2 / w_1$ for simplification, Eq. 7.71 arises

$$\text{Min } Z = \frac{1}{2} * \sum_{t\in\mathcal{T}} \sum_{l\in\hat{L}} \left(\left(\sum_{n\in I} \sum_{r=1}^{t} a_{n,r}^{l,t} * x_{n,r} \right) - \hat{y}_{l,t} \right)^2 + \frac{w}{2} * \sum_{r\in\mathcal{T}} \sum_{n\in I} \left(x_{n,r} - x_{n,r}^{H} \right)^2 \tag{7.71}$$

In this case, Eq. 7.47 is updated, resulting in Eq. 7.72:

$$\begin{aligned} \frac{\partial Z}{\partial x_{n,r}} &= \sum_{t\in\mathcal{T}} \sum_{l\in\hat{L}} \frac{\partial y_{l,t}}{\partial x_{n,r}} * \left(y_{l,t} - \hat{y}_{l,t} \right) + \frac{w}{2} * x_{n,r} \\ &= \sum_{t\in\mathcal{T}} \sum_{l\in\hat{L}} a_{n,r}^{l,t} * \left(y_{l,t} - \hat{y}_{l,t} \right) + \frac{w}{2} * x_{n,r} \end{aligned} \tag{7.72}$$

Therefore, the *Iterative Dynamic Spiess Procedure* would be (Eq. 7.73):

$$X_i^{(k+1)} = \begin{cases} X_i^H & \text{for } k = 0 \\ X_i^{(k)} * \left(1 - \lambda^{(k)} * \left[\frac{\partial Z(X)}{\partial X_i} \right]_{X_i^{(k)}} \right) & \text{for } k > 0 \end{cases} \tag{7.73}$$

The use of Euclidean distances to measure the distance between the estimated OD, X, and the historical X^H has been discussed critically in Frederix et al. (2013). For example, Djukic (2014) 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. Other distance measures have been suggested, for example, in Ros-Roca et al. (2020). Although additional measurements are expected to improve the outcome of the OD-estimation in terms of structural similarity, the analytic approaches do not seem capable of adding measurements different from link counts.

The resort to the classical entropy function, as in the original analytical formulations, is an appealing option because of its structural meaning. With this approach, Eqs. 7.71 and 7.72, respectively, become Eqs. 7.74 and 7.75:

$$\text{Min } Z = \frac{1}{2} * \sum_{t\in\mathcal{T}} \sum_{l\in\hat{L}} \left(\left(\sum_{n\in I} \sum_{r=1}^{t} a_{n,r}^{l,t} * x_{n,r} \right) - \hat{y}_{l,t} \right)^2 + \frac{w}{2} * \sum_{r\in\mathcal{T}} \sum_{n\in I} x_{n,r} * \log\left(\frac{x_{n,r}}{x_{n,r}^H} \right) \tag{7.74}$$

$$\frac{\partial Z}{\partial x_{n,r}} = \sum_{t\in\mathcal{T}} \sum_{l\in\hat{L}} a_{n,r}^{l,t} * \left(y_{l,t} - \hat{y}_{l,t} \right) + \frac{w}{2} * \left(\log\left(\frac{x_{n,r}}{x_{n,r}^H} \right) + 1 \right) \tag{7.75}$$

7.5 Practical Applications for Traffic Management

Because DTA is a core component of most Dynamic Traffic Management Systems and the Dynamic Origin–Destination Matrices are the main input to DTA, algorithms to numerically implement DODME approaches become a basic procedure in all of them. The main approaches are:

- The strict analytical dynamic approaches based on State-Space Modeling (Ashok and Ben-Akiva 1993, 2002), which are the basis of DynaMIT (Ben-Akiva et al. 2020).
- The numerical approximations of analytical optimization approaches, as the ones proposed by Frederix et al. (2011), Frederix et al. (2013), Toledo and Kolechkina (2013), or Ros-Roca et al. (2020). Other variants are those studied by Djukic et al. (2017,2018,2019), currently implemented in Aimsun Live, Aimsun (2020), or OPTIMA.
- Simulation-based approaches: Stochastic Perturbation Stochastic Approximation (SPSA).

7.5.1 Analytical Approaches Based on State-Space Modeling

The approach taken in DynaMIT to estimate dynamic OD-matrices, aimed at providing support to real-time management decisions, is different from the bi-level optimization considered so far. DynaMIT formulates the real-time dynamic OD-estimation based on the Kalman Filtering framework proposed by Ashok and Ben-Akiva (1993). The basic information, as in all other approaches, is that contained in the historical OD-matrix, which is combined with traffic count data from the counting stations along the network. Other differential aspects of the estimation proposed in DynaMIT are the use it makes of each day's estimate to update the original historical OD estimate in a learning process. These updated historical OD-matrices contain rich information about the latent factors that affect travel demand and its daily variations, which the approach tries to capture. To achieve this goal, this approach uses as state variables the deviations of the OD flows from the historical OD estimates, instead of the actual flows themselves.

The underlying hypothesis states that (Antoniou et al. 2007) modern surveillance systems generate data and historical information that can be used for the estimation and prediction of the time evolving demand patterns represented by OD-matrices. The wealth of information contained in these off-line values, which affects trip making and traffic dynamics, as well as their temporal and spatial evolution, can be incorporated into the DODME process as a priori estimates.

The approach based on Kalman filtering assumes an autoregressive procedure that provides a prediction tool consistent with the estimation process. That autoregressive procedure models the temporal relationships among deviations in OD flows, also accounting for unobserved factors that are correlated over time, as, for instance,

weather effects. A proper approach that incorporates this information and its associated errors in the estimation process considers transport systems as dynamic systems and resorts to the state-space modeling approach. The formulation of the DODME problem discussed so far shows that the most critical issue is the calculation of the assignment matrix, a_{ijr}^{lt}, mapping the observed link flows, y_{lt}, and the unobserved OD flows, x_{ijr}. This matrix must be estimated at each step of the iterative processes by solving numerically the corresponding mathematical model (Eq. 7.76):

$$y_{lt} = \sum\nolimits_{(i,j)\in I} \sum\nolimits_{r=1}^{t} a_{ijr}^{lt} * x_{ijr} \ \forall l \in \hat{L}, t \in T \tag{7.76}$$

The dynamic problem formulation assumes that the assignment matrix depends on link and path travel times and on traveler route choice factors, being all of them time-varying. Precisely, time variations are captured by the time indices in Eq. 7.76. The mapping can be interpreted as the contribution, i.e., the fraction, of the OD flow of pair (i, j) departing origin i with destination j, at time interval r, that flows across detectors located at link l, during time interval t.

Ashok and Ben-Akiva (2002), in an extension to their previous seminal work in Ashok and Ben-Akiva (1993), make the observation that "*all quantities are imperfectly observed, thereby they introduce errors into the OD estimation process, erroneous travel times and/or route choice fractions resulting in an imperfect assignment matrix*". Therefore, they propose reformulating Eq. 7.76 as Eq. 7.77:

$$y_{lt} = \sum\nolimits_{(i,j)\in I} \sum\nolimits_{r=1}^{t} a_{ijr}^{lt} * x_{ijr} + v_{lt} \ \forall l \in \hat{L}, t \in T \tag{7.77}$$

where v_{lt} is the measurement error. The reformulation of the DODME as a state-space model involves two types of equations:

- Transition equations that capture the evolution of the state vector over time.
- Measurement equations that, according to Antoniou et al. (2007), "*capture a mapping of the state vector on the measurements:* a priori*values of the model parameters provide direct measurements of the unknown parameters*".

Let X_k be the vector of state variables whose values define the state of the system at time interval k. A Kalman filter iterates between an *updating* (prediction) of the system's state at time k, obtained from the system's state at time $k-1$, and a *correction* based on an update of the measurements of the system. This corresponds to a process model that models the transformation of the system's state in terms of a linear stochastic difference equation (Eq. 7.78):

$$X_k = \Phi X_{k-1} + w_{k-1} \tag{7.78}$$

where Φ is the *transition matrix* from system's state at time $k-1$ to system's state at time k, and w_{k-1} is the process error term. Additionally, a measurements model describes the relationship between the process changing the system's state and the system measurements (Eq. 7.79):

$$Y_k = A * X_k + v_k \tag{7.79}$$

Assuming initial estimates of the state vector $\widehat{X}_{k-1}$ and of the error covariance P_{k-1} at time interval $k-1$, the *prediction phase* consists of two steps: (i) a state projection step (Eq. 7.80) and (ii) a covariance projection step (Eq. 7.81), respectively, projecting forward the state estimate or the covariance from time step $k-1$ to step k:

$$\widehat{X}_k^{k-1} = \phi * \widehat{X}_{k-1}^{k-1} + w_{k-1} \tag{7.80}$$

$$P_k^{k-1} = \Phi * P_{k-1}^{k-1} * \Phi^T + Q \tag{7.81}$$

The correction regarding the measurements update consists of three steps: (i) the computation of the Kalman Gain (Eq. 7.82), (ii) the update of the error covariance (Eq. 7.83) and (iii) the update of the state estimates with the measurements Z_k (Eq. 7.84):

$$K_k = P_k^{k-1} * A^T * \left(A * P_k^{k-1} * A^T + R\right)^{-1} \tag{7.82}$$

$$P_k^k = (I - K_k * A) * P_k^{k-1} \tag{7.83}$$

$$\widehat{X}_k^k = \widehat{X}_k^{k-1} + K_k * \left(Y_k - A * \widehat{X}_k^{k-1}\right) \tag{7.84}$$

where w_k and v_k, the process and measurement errors, are independent, white noise, and normally distributed (Eqs. 7.85 and 7.86):

$$p(w) \sim N(0, Q) \tag{7.85}$$

$$p(v) \sim N(0, R) \tag{7.86}$$

Q and R are, respectively, the covariance matrices of the process and the measurement errors.

When applying Kalman filtering to DODME, the state vector is the vector X of unknown OD flows, and the transition equation represents an autoregressive process. However, Ashok and Ben-Akiva (1993) state that "*an autoregressive process can only capture interdependencies among OD flows. It does not include structural information about trip patterns, which are a function of spatial and temporal distribution of activities, as well as of the characteristics of the transportation system*". Therefore, it is desirable to modify the model in such a way that it also incorporates structural information. This information could be, for example, that contained in a prior estimate. For instance, a historical OD-matrix X^H provided by a reliable surveillance system. It can be accommodated in the model by reformulating the state vector in terms of the deviations from that historical OD flows. The transition equation would then be as follows (Eq. 7.87):

$$X_{ij(t+1)} - X^H_{ij(t+1)} = \sum_{r=t-s}^{t} \sum_{(p,q)\in I} f_{ijt}^{pqr} * \left(X_{pqr} - X^H_{pqr}\right) + w_{ijt} \tag{7.87}$$

where f_{ijt}^{pqt} describes the effect of the deviation $\left(X_{pqr} - X^H_{pqr}\right)$ on the deviation $\left(X_{ij(t+1)} - X^H_{ij(t+1)}\right)$. The first one is the deviation of the OD flow from origin p to destination q and departing at time r. Equivalently, the second one is the deviation of the OD flow from origin i to destination j and departing at time $t+1$. In this second deviation, w_{ijt} is a random term error for OD pair (i,j) at time t and s is the order of the autoregressive process, that is, the number of lagged OD flow deviations assumed to affect the OD deviation in interval $t+1$. Equation 7.87 "*models the temporal relationship among deviations in OD flows, capturing the correlation over time among deviations which arise from unobserved factors that correlated over time. It assumes dependency of deviations corresponding to one OD pair on deviations corresponding to other OD pairs in prior periods*" (Ashok and Ben-Akiva 1993). It can be rewritten in matrix form (Eq. 7.88):

$$\Delta X_{t+1} = X_{t+1} - X^H_{t+1} = \sum_{r=t-s}^{t} \Phi_t^r * \left(X_{pqr} - X^H_{pqr}\right) + w_t \tag{7.88}$$

In the general case, the computation of the transition matrix Φ_t^r involves estimating linear regression models for each OD pair and for each time interval. However, depending on the network topology, some of these correspondences may

be ignored and thus the matrix is simplified. There are also some other hypotheses enabling further simplifications, as, for example, the assumption that the autoregressive process remains constant with respect to t. This implies that it depends only on the difference $(t-s)$ and not on the individual values of t and s. Equation 7.77 can be rewritten accordingly to get the measurements equation in terms of deviations with respect to historical values y_{lt}^{H}, as in Eq. 7.89:

$$y_{lt} - y_{lt}^{H} = \sum\nolimits_{(i,j)\in I} \sum\nolimits_{r=\mathrm{t-s}}^{t} a_{ijr}^{lt} * \left(x_{ijr} - x_{ijr}^{H}\right) + v_{lt} \ \forall l \in \widehat{L}, t \in T \tag{7.89}$$

It can also be expressed in matrix form (Eq. 7.90):

$$\Delta Y_t = Y_t - Y_t^{H} = \sum\nolimits_{r=\mathrm{t-s}}^{t} A_r^t * \left(X_r - X_r^{H}\right) + v_t \tag{7.90}$$

where v_t is the measurements random error vector at time$t$. Error terms w_t and v_t are uncorrelated, which means that$E[w_t] = E[v_t] = 0$. The variance–covariance matrices are Q_t and R_t, respectively.

There is an additional advantage in reformulating the Kalman filtering in terms of deviations as state variables and measurements, since the traffic flow variables have skewed distributions (Antoniou et al. 2007). However, the deviations from these variables from available estimates have symmetric distributions and, hence, are more amendable to approximations to normal distributions. This is a useful property in terms of Kalman filtering (Kalman 1960; Gelb 1974). Then, assuming an initial state of the system with ΔX_0, with mean $\Delta \overline{X}_0$, and variance–covariance P_0, the Kalman filtering algorithm for DODME, for a time horizon T divided into N intervals of equal length, is

Initialization

$$\Delta X_0^0 = \Delta X_0$$
$$P_0^0 = P_0$$

For k=1 to N do

Time update (Transition)

$$\Delta \hat{X}_k^{k-1} = \phi * \Delta \hat{X}_{k-1}^{k-1} + w_{k-1}$$

$$P_k^{k-1} = \Phi * P_{k-1}^{k-1} * \Phi^T + Q$$

Measurement update

$$K_k = P_k^{k-1} * A^T * \left(A * P_k^{k-1} * A^T + R\right)^{-1}$$

$$\Delta \hat{X}_k^k = \Delta \hat{X}_k^{k-1} + K_k * \left(\Delta Y_k - A * \Delta \hat{X}_k^{k-1}\right)$$

$$P_k^k = (I - K_k * A) * P_k^{k-1}$$

End

Many alternative versions of these basic algorithms resorting to variants of Kalman filtering have been proposed, as those in Ashok and Ben-Akiva (2002), Hu et al. (2001), Antoniu et al. (2007), Lin and Chang (2007). In essence, many of the most appealing ones deal with the calculation of matrices Φ and A. That is, with the characteristics of the autoregressive model, the mapping OD paths and the links flows, being these latter the most critical. Antoniu et al. (2007) propose nonlinear relationships for the measurement equations, generically defined as (Eq. 7.91):

$$\Delta M_t = M_t - M_t^H = \mathcal{S}(\Delta X_t) - M_t^H + v_t \tag{7.91}$$

where M_t is the vector of measurements at time t, $\mathcal{S}(\Delta X_t)$ is a simulation model and $M_t^H = \mathcal{S}(\Delta X_t^H)$. When traffic flow models are used to simulate the time progression of traffic flows through the network, they can be approximated by continuous functions $h(x)$(Antoniou 2004). These functions can be linearized to approximate the measurement equation as in Eq. 7.92:

$$H_t = \left.\frac{\partial h(x^*)}{\partial x^*}\right|_{x^*=X_t^{t-1}} \tag{7.92}$$

An example based on this linearization included in Antoniou (2004) and Antoniou et al. (2007) is the following Extended Kalman Filter (EKF):

Initialization

$$\Delta X_0^0 = \Delta X_0$$
$$P_0^0 = P_0$$

For k=1 to N do

Time update (Transition)

$$\Delta \hat{X}_k^{k-1} = \phi * \Delta \hat{X}_{k-1}^{k-1} + w_{k-1}$$

$$P_k^{k-1} = \Phi * P_{k-1}^{k-1} * \Phi^T + Q$$

Linearization step

$$H_k = \left.\frac{\partial h(x^*)}{\partial x^*}\right|_{x^*=X_k^{k-1}}$$

Measurement update

$$K_k = P_k^{k-1} * H_k^T * \left(H_k * P_k^{k-1} * H_k^T + R\right)^{-1}$$

$$\Delta \hat{X}_k^k = \Delta \hat{X}_k^{k-1} + K_k * \left(\Delta Y_k - H_k * \Delta \hat{X}_k^{k-1}\right)$$

$$P_k^k = (I - K_k * H_k) * P_k^{k-1}$$

End

Equation 7.91 also opens the door to the consideration of additional measurements in Kalman filters other than traffic variables like the link flow counts from fixed counting stations (e.g., inductive loop detectors, magnetometers…). For example, the travel times between pairs of points in the network, as measured by ICT applications (e.g., Bluetooth, GPS…).

7.5.2 Aimsun Live

A professional software platform for traffic management with a DTA as core engine and that haves as main input dynamic OD-matrices is Aimsun Live (Aimsun 2020). The DODME process implemented in Aimsun Live is a variant of the numerical approximations of analytical optimization approaches discussed Sect. 7.4. Djukic et al. (2017,2018,2019) reformulate the bi-level approach (Eq. 7.61) as in Eq. 7.93:

$$\text{Min } Z(X) = \alpha * \|X - X^H\|^2 + (1-\alpha) * \|A(X) * X - \widehat{Y}\|^2 \tag{7.93}$$

$$s.t.\ X \geq 0$$

Assuming that the flow estimates are provided by the DTA at the lower level, i.e., at the algorithmic framework in Fig. 7.4. implemented in Aimsun (2020), then $Y = DTA(X)$. This allows a Taylor expansion as in Eq. 7.63. Then, Djukic et al. (2018) propose a modified bi-level approach that, at iteration k, replaces at the upper level the objective function in Eq. 7.95 by the approximation in Eq. 7.94:

$$Z_k(X) = \alpha * \|X - X^H\|^2 + \|\widehat{Y} - Y_k - A_k * (X - X_k)\|^2 \tag{7.94}$$

where at iteration k, X_k is the estimated OD demand vector, A_k the assignment matrix estimated from Aimsun's DTA using Eq. 7.63 and Y_k the vector of estimated link flow counts in the subset of links with counting stations. Aimsun's DTA estimates A_k by stopping the Taylor expansion at either the first or the second term, depending on the desired degree of accuracy or on the affordable computing cost. Djukic et al. (2018) propose to solve the approximated upper level optimization problem (Eq. 7.94) with non-negative variable constraints, using a gradient descent method. Particularly, one using as descent direction the one defined by the following gradient (Eq. 7.95):

$$d_k = -\nabla Z_k(X) \tag{7.95}$$

This gradient can be calculated from (7.94) as

$$\nabla Z_k(X) = 2 * \alpha * \left(X - X^H\right) + 2 * \left(A_k^T * A_k * X - A_k^T * \hat{Y} + A_k^T * Y_k - A_k^T * A_k * X_k\right) \quad (7.96)$$

Then, the new OD-matrix for the lower level iteration $k+1$ is given by Eq. 7.97:

$$X_{k+1} = X_k + \lambda_k * d_k \quad (7.97)$$

where λ_k is the optimal step length in the gradient movement along the descent direction. The gradient procedure to optimize Eq. 7.94 is also iterative. It recalculates the step size at each iteration until either a convergence criterion is met or a maximum number M of iterations is reached, whatever occurs first. At gradient iteration m, the estimated demand is X_k^m, the search direction at this iteration is given by $\nabla Z\left(X_k^m\right)$ (calculated from Eq. 7.96) and the step size calculation can be calculated solving Eq. 7.98, using any of the available line search procedures (Bazaraa et al. 1993):

$$\lambda_k^m = Min_\lambda Z\left[X_k^m - \lambda * \nabla Z\left(X_k^m\right)\right] \quad (7.98)$$

However, since $Z(X)$ is quadratic, the optimal step can be computed analytically using Eq. 7.99:

$$\lambda_k^m = \frac{\left\|\nabla Z\left(X_k^m\right)\right\|^2}{\left\|\nabla Z\left(X_k^m\right)\right\|^2 + \left\|A_k * \nabla Z\left(X_k^m\right)\right\|^2} \quad (7.99)$$

The proposed algorithm iteratively updates the demand at iteration $k+1$ from the demand at the previous iteration k, until some convergence criteria are satisfied. The algorithm is modified with respect to the usual approaches to better fit the requirements for congested large-scale networks. The proposed modification relaxes the assumption on link flow proportions provided by the DTA assignment matrix by computing the marginal effects of the demand deviations on link flows given by Eq. 7.63. Therefore, it reduces the number of OD variables in this Eq. 7.63 by including only those OD pairs whose change in demand values causes significant deviations in the link flows. The modified algorithm is, according to Djukic et al. (2018), as follows:

1) Initialization

Initiate prior OD demand matrix. Set $k = 0$, $I' = X_0$ (seed matrix).

2) Assignment

Assign the demand to the network to obtain the assignment matrix, A_k and to estimate the link traffic counts with traffic observations with Equation 7.63 (1st or 1st & 2nd terms of Taylor's expansion, depending on the choice).

3) Convergence test

Check the value convergence of the objective function. If it has converged, stop and accept the current demand. Otherwise, proceed to 4).

4) Objective Function performance test

Check the performance of the objective function value. If the objective function decreases, proceed to 5). Otherwise, proceed to 6) with $k = k - 1$.

5) Update the OD demand

Estimate the OD demand with the link flows obtained from DTA, as given by Equation 7.63. Otherwise, proceed to 2) with $k = k + 1$.

6) Select OD pairs

Determine the OD pairs whose variation has a considerable impact on the link flows variation in the previous iteration, and insert them in *I'*.

7) Update assignment

For the selected OD pairs in *I',* update the link-flow proportions in the assignment matrix $A_{k-1}$with values obtained from the chosen version of Equation 7.63.

8) Update the OD demand

Estimate the OD demand with the link flows obtained from Equation 7.63. Proceed to 2) with $k = k + 1$.

End

The computational testing of this proposed modified bi-level optimization framework, which solves the high-dimensionality of nonlinear OD-estimation problems by computing the marginal effects only for the most significant OD pairs with respect to traffic observations, allows the modeler to control the trade-off between the simplicity of the model and the level of realism. It is thus very efficient for practical purposes.

7.5.3 Simulation-Based Approaches: Stochastic Perturbation Stochastic Approximation (SPSA)

The optimization problem in Eqs. 7.58 and 7.59, as already mentioned, is highly underdetermined because there are many more variables than equations in the system. In other words, $X \in \mathbb{R}^{|I| \times T}$, $Y \in \mathbb{R}^{\left|\widehat{L}\right| \times T}$ and $|I| \gg |\widehat{L}|$. Therefore, the problem is very sensitive to the quantity of data and the detection layout in the real network. As the availability of new measurements like those provided by smartphone and GPS localization allows calculating travel times between arbitrary pairs of points, the use of these data seems to be a promising approach for reducing the aforementioned underdetermination. An apparently straightforward extension of the bi-level formulation in Eqs. 7.58 and 7.59 accounting for measured, $\widehat{tt}$, and estimated travel times, tt, would be the expansion of the objective function adding a third term, $F_3\left(tt, \widehat{tt}\right)$. This term would be aimed at minimizing the distance between measured and estimated travel times between arbitrary pairs of points in the network, assuming that trips are most likely made via the shortest paths. The hypothetical formulation (Ros-Roca et al., 2021a) would be (Eqs. 7.100–7.102):

$$\text{Min } Z(X) = w_1 * F_1\left(X, X^H\right) + w_2 * F_2\left(Y, \widehat{Y}\right) + w_3 * F_3\left(tt, \widehat{tt}\right) \tag{7.100}$$

$$s.t.\ Y(X) = \text{Assignmt}(X) \tag{7.101}$$

$$tt(X) = \mathcal{F}(X) \tag{7.102}$$

$$X \in \Omega$$

Assuming that $Y(X) = \text{Assignmt}(X) = A(X) * X$, that is, the relationship between the estimated link flows and the estimated OD-matrix defined by the assignment, the problem can be reformulated as follows (Eqs. 7.103 and 7.104):

$$\text{Min } Z(X) = w_1 * F_1\left(X, X^H\right) + w_2 * F_2\left(A(X)X, \widehat{Y}\right) + w_3 * F_3\left(tt, \widehat{tt}\right) \tag{7.103}$$

$$s.t.\ tt(X) = \mathcal{F}(X) \tag{7.104}$$

$$X \in \Omega$$

The analytical relationship in Eq. 7.104 either does not exist or is unclear. However, in practice, travel times can be estimated from it if the assignment is a DTA. Therefore, it can be accepted that some kind of relationship exists and the relationship $tt(X) \sim \text{Assignmt}(X)$ is assumed. The problem to be solved is again reformulated

as (Eq. 7.105):

$$\text{Min } Z(X) = w_1 * F_1\left(X, X^H\right) + w_2 * F_2\left(Y, \widehat{Y}\right) + w_3 * F_3\left(tt, \widehat{tt}\right) \tag{7.105}$$

$$s.t.\ (Y, tt) = \text{Assignmt}(X)$$

$$X \in \Omega$$

As mentioned before, it is unclear how these new measurements can be included in the analytical formulations. Nevertheless, it seems rather easy to deal with them by using approaches based on derivative-free optimization methods that approximate the descent direction based on simulation. Among them, simulation optimization techniques are especially suited to deal with optimization problems that cannot be solved with the usual analytical algorithms. Some reasons are:

- The objective function cannot be analytically expressed as a function of parameters because its evaluation requires a simulation. Therefore, it is not differentiable in terms of the parameters.
- The time cost of evaluating the objective function is expensive, as it requires having simulated data for each evaluation of the function.

Simulation-based optimization techniques can be generically formulated assuming that there is a mathematical model $\mathcal{M}$ with a set of parameters $P = \{p_1, p_2, \ldots, p_N\}$ and an objective function $\mathcal{F}(\mathcal{R}, S)$ defined as the sum of error functions between real observations $\mathcal{R}$ and the corresponding simulated data S. The purpose of $\mathcal{M}$ is then to provide (Eq. 7.106):

$$\text{Min } \mathcal{F}(\mathcal{R}, S) \tag{7.106}$$

$$s.t.\ P \in \Omega \subseteq \mathbb{R}^N$$

When $\mathcal{F}(\mathcal{R}, S)$ (i) is on-convex, nonlinear, (ii) cannot be represented analytically as a function of the set of parameters P and (iii) has to be evaluated by simulation.

There is a wide range of different simulation optimization techniques to solve Eq. 7.106. For example, Nelder-Mead, SNOBFIT, and SPSA are optimization techniques, either derivative free or approximating the gradient, that evaluate it using simulation. Osorio and Linsen (2015) make an approximation of the upper level function by building a metamodel that can be solved analytically. Its conceptual diagram is depicted in Fig. 7.5.

Simultaneous Perturbation Stochastic Approximation (SPSA) (Spall 1992) is commonly used in OD-matrix estimation (Cipriani et al. 2011; Cantelmo et al. 2014; Antoniou et al. 2015; Lu et al. 2015; Ros-Roca et al. 2020) and it can easily account for additional measurements (Bullejos et al. 2014; Antoniou et al. 2016; Carrese

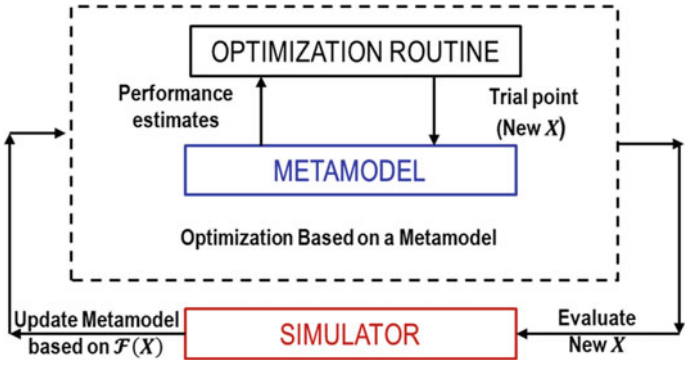

Fig. 7.5 Conceptual diagram of the simulation-based optimization approach of Osorio and Linsen (2015)

et al. 2017; Nigro et al. 2018). SPSA preserves the original upper level formulation and is easy to implement for simulation optimization problems.

SPSA is a simulation-based optimization algorithm, and it only requires two evaluations of the objective function to approximate the gradient instead of N, as in the case of a finite-difference gradient approach. Like in many iterative procedures, it begins with an initial OD-matrix (usually a historical OD-matrix). The next matrix at iteration $k+1$ is computed from the matrix at iteration k, moving a distance a_k along the descent direction provided by the following gradient (Eq. 7.107):

$$X_{k+1} = X_k - a_k * \hat{g}_k(X_k) \tag{7.107}$$

Two particularities distinguish this method from the conventional gradient descent method:

- The estimated gradient $\hat{g}_k(X_k)$, is calculated according to Eq. 7.108:

$$\hat{g}_k(X_k) = \frac{Z(X_k + c_k * \Delta_k) - Z(X_k)}{c_k} * \begin{pmatrix} \Delta_{k,1}^{-1} \\ \vdots \\ \Delta_{k,N}^{-1} \end{pmatrix} = \begin{pmatrix} \frac{Z(X_k + c_k * \Delta_k) - Z(X_k)}{c_k * \Delta_{k,1}} \\ \vdots \\ \frac{Z(X_k + c_k * \Delta_k) - Z(X_k)}{c_k * \Delta_{k,N}} \end{pmatrix} \tag{7.108}$$

  where Δ_k is a random perturbation N-dimensional vector with $\Delta_i, \forall i$ independent identically distributed random variables that satisfy $\mathbb{E}(\Delta_i) = 0$ and $\left|\mathbb{E}\left(\left(\Delta_i^{-1}\right)^n\right)\right| < \infty, \forall n$. One commonly used perturbation is $\Delta_i \sim Be(1/2, \pm 1)$, which is a Bernoulli distribution with a probability of ½ for each ± 1. This is the asymmetric design, although a symmetric design using $Z(X_k + c_k * \Delta_k)$ and $Z(X_k - c_k * \Delta_k)$ can also be considered.

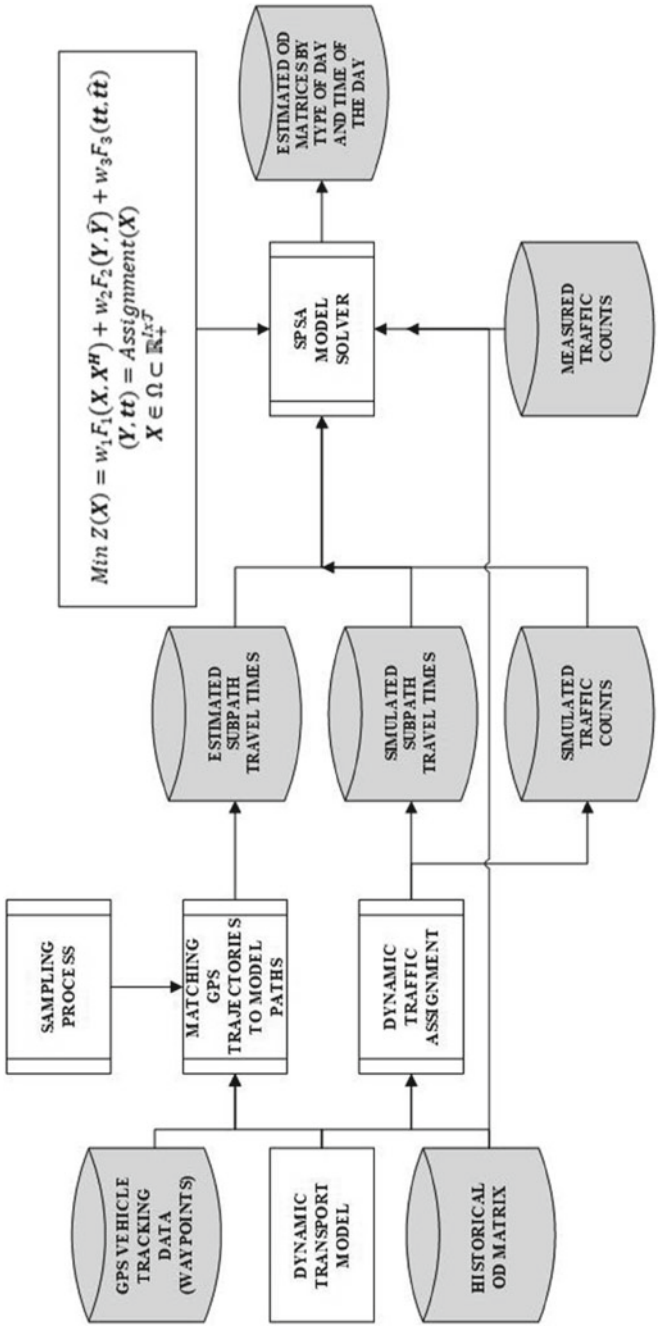

Fig. 7.6 Conceptual diagram of a SPSA approach adding travel times

- The spacing coefficient c_k and the step size a_k are decreasing sequences of positive real values, and they satisfy some regularity conditions in order to ensure the convergence of the method, as detailed in Spall (1992). Typically, the sequences used are (Eqs. 7.109 and 7.110):

$$a_k = \frac{a}{(A + k + 1)^{\alpha}} \tag{7.109}$$

$$c_k = \frac{c}{(k + 1)^{\gamma}} \tag{7.110}$$

  where a, A and c are chosen depending on the problem, while $\alpha = 0.602$ and $\gamma = 0.101$.

Averaging many independent estimates of the gradient of Eq. 7.108 contributes to a more stable and quicker convergence of the SPSA method (Spall 1992). Therefore, the gradient estimation is finally calculated as (Eq. 7.111):

$$\hat{g}(X_k) = \frac{1}{n_g} * \sum_{j=1}^{n_g} \hat{g}_k^j(X_k) \tag{7.111}$$

where $\hat{g}_k^j(X_k)$ is precisely calculated as in Eq. 7.108. The asymmetric design for the gradient saves a large number of assignments, since all $\hat{g}_k^j(X_k)$, $\forall j$ share the mid-point X_k evaluation.

The versatility of simulation optimization techniques, especially when using SPSA, allows including additional information in a new form, such as the constraints in the OD-estimation problem. Ros-Roca et al. (2017) tried adding constraints to simulation optimization problems when dealing with the calibration of microsimulation models.

A potential improvement with respect to the original formulation (Bullejos et al. 2014; Cantelmo et al. 2014) replaces the gradient by the Conjugate Gradient (CG) (Luenberger and Ye 2008), a descent method for the optimization algorithm of the OD-estimation problem. This modifies the descent direction in the iterative procedure by using the previous iteration gradient. It can be incorporated into SPSA by replacing Eq. 7.107 with Eqs. 7.112–7.114:

$$X_k = X_{k-1} + a_k * d_k \tag{7.112}$$

$$d_k = -\hat{g}(X_k) + \beta_k * \hat{g}(X_{k-1}) \tag{7.113}$$

$$\beta_k = \frac{\hat{g}(X_k)^T * d_{k-1}}{\|d_{k-1}\|^2} \tag{7.114}$$

SPSA's main drawback for the OD-estimation problem is that all different OD flows receive the same perturbation magnitude (Eq. 7.108). As OD flows usually have very different magnitudes, this implies very different changes to each flow, which can lead to several problems of convergence. Tympakianaki et al. (2015) approached this phenomenon by clustering the variables according to their magnitude. A different alternative can be normalizing to the interval [0, 1] all variables using some particular reasonable bounds $[a_i, b_i]$. For example, Ros-Roca et al. (2018) performed a classical linear transformation from $[a_i, b_i]$ to [0, 1], where a_i and b_i were based on additional information from the network, particularly socioeconomic or past reliable OD-matrices. The normalization was performed using the following linear application (Eq. 7.115):

$$\begin{array}{ccc} \varphi_i : [a_i, b_i] & \rightarrow & [0, 1] \\ X_i & \mapsto & \frac{X_i - a_i}{b_i - a_i} \end{array} \tag{7.115}$$

Using the normalized variables in SPSA procedure, each variable will be perturbed according to its magnitude.

Experience with similar problems shows that the selection of SPSA gain sequences a_k and c_k is crucial for the convergence and performance of the algorithm. The sequences in the form of Eq. 7.109 and 7.110 are widely used, as they satisfy the conditions of convergence that were proved in Spall (1992). This reduces the problem of selecting appropriate values for a, A, α, c and γ. Kostic et al. (2017b) showed the sensitivity of SPSA with respect to these parameters. Based on the guidelines in Spall (2003), an automated selection of the parameters a, A and c, can be based on the objective function's variability that results from the simulation, and on the desired perturbation steps in the early iterations. The selection would be done according to the following schema:

- First, those values stated as optimal for convergence in Spall (1998) are fixed. That is, $\alpha = 0.602$, $\gamma = 0.101$.
- Several evaluations of $Z(X^H)$ to capture the variability of the objective function are computed. Since the variables have been normalized, it seems natural to use the coefficient of variation ($CoV(Z) = \sigma_Z / \mu_Z$) for this purpose. The parameter c is set at $c = CoV$.
- A is set as 10% of the maximum number of iterations ($A = 0.1 \cdot \text{iter}_{\max}$).
- n_g experiments are simulated using the SPSA logic $X_i = X^H + c\Delta_N$. This allows finding the respective gradients g_k as in the SPSA procedure.
- The desired iterative modification of the first iteration must be determined with Eq. 7.116:

$$X_{k+1} = X_k - a_k * g_k \rightarrow X_{k+1} - X_k = |a_k * g_k| \tag{7.116}$$

- The corresponding a for the desired change in the initial iteration must be computed using Eq. 7.117:

$$\left|a_k * g_k\right| = \frac{a}{(1+A+k)^{\alpha}} * |g_k| \rightarrow a = \frac{\left|a_k * g_k\right| * (1+A+k)^{\alpha}}{|g_k|} \tag{7.117}$$

- The minimum of the n_g performed experiments must be finally chosen. That is (Eq. 7.118):

$$a = \min\left\{a_{\{i=1\}}, \ldots, a_{\{i=N_g\}}\right\} \tag{7.118}$$

As already mentioned, the underdetermination of the OD-estimation problem can lead to different adjusted OD-matrices that show the same traffic counts at the sensor locations even though they are different. Furthermore, the adjusted OD-matrix can also be non-consistent with the socioeconomic factors of the area under study. In traffic analyses, practitioners usually have access to historical data in the form of an OD-matrix X^H which, with a certain degree of uncertainty, provides prior information about the mobility patterns of the target area. Therefore, including constraints in the SPSA formulation that accounts for this information can lead to more realistic results. A possible approach is to add bounding values to the OD values, which is not easy to do in analytical formulations (Codina and Montero 2006) but is relatively easy to manage in SPSA. In Cipriani et al. (2011), a single generation constraint is added to the minimization problem (Eq. 7.119):

$$\sum_{i=1}^{n_h} G_o^i \leq G_o^* \quad \forall o \in \{\text{origins}\} \tag{7.119}$$

with G_o^* being the a priori generation value for the origin zone 0 and n_h the number of time periods. Other approaches, that of Ros-Roca et al. (2020), specify upper and lower bounds for each OD flow, defined in terms of a percentage β of this flow's historical value, according to its degree of uncertainty. With the constraints, the minimization problem is updated as follows (Eqs. 7.120 and 7.121):

$$\text{Min } Z(X,Y) = w_1 * F_1\left(X, X^H\right) + w_2 * F_2\left(Y, \widehat{Y}\right) \tag{7.120}$$

$$s.t.\ Y = \text{Assignmt}(X) \tag{7.121}$$

$$X \in G = \left\{(1-\beta) * x_{n,r}^H \leq x_{n,r} \leq (1+\beta) * x_{n,r}^H, \forall x_{n,r} \in X\right\} \subset \mathbb{R}_+^{I \times T}$$

$$X \geq 0$$

This single constraint in Eq. 7.119 results from summing for each origin all the upper bounds in the former minimization problem. The addition of all constraints makes the feasible region bigger. Greater values are therefore allowed for some

variables, but this is compensated by others having low values. On the contrary, the proposal for constrained SPSA in Ros-Roca et al. (2020) defines a smaller feasible region that accounts for further information of each OD pair.

These constraints added to the problem also have an effect on the originally presented SPSA algorithm. Sadegh and Spall (1998) proposed to add a projection to the set G during the iterative procedure shown in Eq. 7.107. The projection would be applied only to the iterative procedure as $X_{k+1} = \pi_G\left(X_k - a_k * \hat{g}_k(X_k)\right)$, while $Z(X_k + c_k * \Delta_k)$ could be computed subject to non-negative OD values. This method, in which some strict constraints are added to the procedure, is called Constrained SPSA.

Inspired in Wang and Spall (1999), other formulations equivalent to Eqs. 7.120 and 7.121 add penalty functions to the objective function (Eqs. 7.122 and 7.123):

$$\text{Min } Z(X, Y) = w_1 * F_1\left(X, X^H\right) + w_2 * F_2\left(Y, \widehat{Y}\right) + r_k * P\left(X, X^H\right) \tag{7.122}$$

$$s.t.\ Y = \text{Assignmt}(X) \tag{7.123}$$

$$X \geq 0$$

where r_k is an increasing sequence of the form $r_k = r*(1+k)^{\rho}$ and $P\left(X, X^H\right)$ is a set of penalization functions for the set of constraints that delimit the constraints of set G. Formally (Eq. 7.124):

$$G \triangleq \left\{q_{n,r}\left(X, X^H\right) \leq 0, \forall n \in I, r \in T\right\} =$$

$$= \left\{x_{n,r} - (1+\beta) * x_{n,r}^H \leq 0, (1+\beta) * x_{n,r}^H - x_{n,r} \leq 0\ \forall n \in I, r \in T\right\} \tag{7.124}$$

The penalty function $P\left(X, X^H\right)$ must be differentiable, non-negative, and an increasing function. Wang and Spall (1999) propose a sum for each constraint of penalizing functions that satisfy $p(x) = 0$ if and only if $x \geq 0$. That is (Eq. 7.125):

$$\begin{aligned} P\left(X, X^H\right) &= \sum_{n \in I} \sum_{r=1}^{T} w_{n,r} * p\left(q_{n,r}\left(X, X^H\right)\right) \\ &= \sum_{n \in I} \sum_{r=1}^{T} w_{n,r} * \max\left\{0, q_{n,r}\left(X, X^H\right)\right\}^2 \end{aligned} \tag{7.125}$$

As in the previous variant, the iterative procedure is also modified to incorporate the gradient of the penalization function (Eq. 7.126):

$$X_{k+1} = X_k - a_k * \hat{g}_k(X_k) - a_k * r_k + \nabla P\left(X_k, X^H\right) \tag{7.126}$$

When additional information from ICT measurements is available, it can be included in the SPSA formulation (Eq. 7.107) as long as it can be estimated from the current OD-matrix X by means of a DTA. This is, for example, the case of subpaths travel times $\hat{tt}$ measured either by Bluetooth (Bullejos et al. 2014; Antoniou et al. 2016) or by GPS tracking (Ros-Roca et al. 2021a). The logical diagram of this process is described in Fig. 7.6. The calculation of these observed subpaths travel times $\hat{tt}$ requires the identification of the most used paths from the available measurements and their map matching to the transport model supporting the DTA. This allows computing the corresponding estimated travel times tt from the current OD, which will be added in the additional term to the objective function in Eq. 7.107. The processing of the GPS data to calculate $\hat{tt}$ is described in Sect. 7.6.

In Kostic et al. (2017a), the additional term of the objective function in Eq. 7.107 is formulated as a function of the measured speeds at detection stations equipped with conventional technologies (i.e., inductive loops), and the DTA used is TRE (Gentile et al. 2007; Gentile 2010), supporting OPTIMA.

7.6 Data-Driven Approaches

The availability of new traffic data supplied by ICT applications, i.e., mobile phones, image processing techniques for license plate recognition, Bluetooth devices, FCD from onboard tracking mobile devices vehicles like GPS, etc., prompted the research interest in finding which could be the advantages of including these data explicitly in the OD-estimation methods. In this context, probe (or equipped) vehicles can be grouped into two generic classes (Nanthawichit et al. 2003; Eiseman and List 2004), according to the explanations in Chap. 1. First, those vehicles equipped with devices that can only be detected at specific locations (i.e., where the detection technology is located), as, for example, those equipped with a tag-reader or with a Bluetooth or Wi-Fi device. Known as "space-based" probe vehicles, their true origin and destination are not known, and their approximate estimates can only be inferred, being this inference strongly dependent on the layout of the detection devices (e.g., tag-readers, Bluetooth antennas). Second, those vehicles equipped with wireless communication mobile devices that are fully visible in the areas covered by the corresponding telecommunications system. Therefore, these systems can provide seamless data about their location, speed, travel direction, etc., depending on the device. These are known as "time-based" probe vehicles.

Methodologies related to space-based probe vehicles that have received significant attention are those based on the identification and reidentification of the license plate of all vehicles passing the area covered by a TV camera with a LPR technology (Mo et al. 2020). Also, those based on the identification of Bluetooth devices between coupled pairs of Bluetooth antennas (Barceló et al. 2013; Behara et al. 2021). However, as already mentioned, results of these methodologies have a strong dependency on the layout of TV Cameras or Bluetooth antennas in the network, this layout becomes a critical aspect for the observability of the system (Castillo et al. 2008) and thus determines the capability of the methods to estimate and predict its state.

As regards time-based probe vehicles, the pervasive penetration of mobile phones has allowed a better understanding of human mobility patterns from their traces, that is, by means of their digital footprints. As mobility patterns include information about where people are and how they got there, mobile phones were soon identified as an important data source for urban modeling. They attracted the interests of researchers and practitioners, as they were seen as a powerful data source that would allow overcoming the well-known drawbacks and limitations of conventional methods in transportation analysis (i.e., household survey). Analyses are usually conducted using datasets, the so-called Call Detail Records (CDR), previously recorded by a mobile provider for communication and billing purposes, after an anonymization process. A seminal example of this process can be found in González et al. (2008), where each individual calling activity is characterized to allow monitoring the user's movement over time. Calabrese et al. (2013) provide an example of techniques aimed at extracting useful mobility information from mobile phone traces of millions of users from which to infer individual mobility patterns in large urban areas, especially OD-matrices (Zhang et al. 2010; Calabrese et al. 2011). Since CDR are time tagged and locations can be identified after suitable processing, added value information for a variety of mobility analyses can be extracted from the (Çolak et al. 2015). Additionally, OD-matrices can be differentiated by purpose and time of the day (Alexander et al. 2015). However, this requires resorting to very specific Data Analytics techniques, given the huge amount of data frequently recorded from millions of users. Gundlegård et al. (2015) or Jianga et al. (2016) are good examples of this data processing to extract the OD-matrices.

However, the type of OD-matrices that dynamic traffic models used in traffic management systems require as input is rather different from the matrices directly extracted from DCR. Indeed, the mobility patterns modeled by these latter OD-matrices are global, that is, they include all types of trips without distinguishing the transportation mode used. Conversely, the OD-matrices of interest for traffic management purposes are usually those modeling the passenger cars patterns. Additional work is necessary to estimate these specific OD-matrices. For example, DCR OD-matrices can be combined with simulation models like MITSIM (Iqbal et al. 2014) or they can be fused with other data sources (Montero et al. 2019). Bassolas et al. (2019) propose also a fusion variant to generate inputs to activity-based travel demand models using MATSIM.

Among the time-based probe vehicles, the better suited to generate OD-matrices that can be exploited by dynamic traffic models seem to be those allowing the tracking of the equipped individual vehicles and the reconstruction of their trajectories. Assuming that the collected data from the tracking technologies include geolocation and time stamps, i.e., waypoints in the terminology of commercial GPS providers, map matching and path inference procedures could provide comprehensive information about origins, destinations, taken paths, and path travel times. This was essentially the assumption in an early paper of van Aerde et al. (1993), accepting that probe vehicles were fully visible. The mentioned seminal papers of Nanthawichit et al. (2003), and Eisenman and List (2004) later accepted this hypothesis. Therefore, assuming that these sampled trajectory data are available, the question is whether

and how they can be used to find sound estimates of dynamic OD-matrices, that is, OD-matrices discretized in time, exploiting for that purpose the time tag recorded data.

Research on the potential use of these mobile data for transport analysis has also prompted a key question with relevant practical applications. Most of the DODME approaches discussed in the previous sections usually assume that one of the inputs is provided by an available historical matrix. The reliability and quality of such historical OD-matrices has been questioned in practical transport planning practice, as it could be largely outdated or even not exist. However, this is not the case in most practical traffic management applications due to the amount and quality of data supplied by modern surveillance systems. Therefore, the improvement of the seed matrices used in DODME by means of sample data from probe vehicles is a relevant contribution. However, the previous discussion on DODME approaches makes evident that all of them rely on the estimate of a dynamic assignment matrix. The fact that this assignment matrix must be estimated by a DTA or a DUE and that the approach implies an iterative process, this could represent a heavy computational burden not affordable in real-time applications. Therefore, the key question is: can the dynamic traffic assignment matrix be empirically estimated from probe vehicle data? And, if possible, how can it be used to improve DODME approaches? A positive answer to the first question opens the door to build models some of whose components are directly derived from an empirical procedure, which would be based on the observed data, instead of from an analytical procedure. In other words, this approach paves the way to build data-driven models.

7.6.1 A Conceptual Proposal on Data-Driven Modeling

From this latter perspective, an interesting proposal is that made by Yang et al. (2017). They wanted to determine whether the availability of such trajectory data could be used to develop an approach to DODME independent of the reliability of an historical OD. That is, to make a good empirical estimation of the assignment matrix, making it unnecessary to resort to DTA. According to the above-mentioned statements, it is assumed that each probe vehicle reports its position in the form of GPS coordinates after a preprocessing procedure performed with map-matching techniques. In summary the approach is as follows:

- It is assumed that vehicle trajectories from origins to destinations are traceable for each probe vehicle, and that the supplied GPS locations have been suitable preprocessed by data cleansing and map-matching procedures. Therefore, GPS locations in the approach are assumed to be exact.
- If $\widehat{L} \subseteq L$ is the subset of links with counting stations, two link flow measurements are available for each time period r. There are $\hat{y}_{lr}, l \in \widehat{L}, r \in T$ flows from the counting stations at links $l \in \widehat{L}$, and $\widehat{h}_{lr}, l \in \widehat{L}, r \in T$ flows of probe vehicles crossing that link l at time interval r.

- The OD probe ratios, that is, the average number of vehicles observed across the entire network during a time interval are given by Eq. 7.127:

$$\gamma_r = \frac{\sum_{l \in \hat{L}} \hat{h}_{lr}}{\sum_{l \in \hat{L}} \hat{y}_{lr}} \forall r \in T \tag{7.127}$$

- Thus, the seed OD-matrix $\hat{x}_{nr}$ can be estimated with Eq. 7.128:

$$\hat{x}_{nr} = \frac{\hat{z}_{nr}}{\gamma_r} \forall n \in I, \forall r \in T \tag{7.128}$$

  where I is the set of all OD pairs, and $\hat{z}_{nr}$ is the number of identified probe vehicles traveling from the origin to the destination of the $n-th$ OD pair.
- The assumption of the identification of locations of probe vehicles allows, in a similar way, directly estimating the assignment matrix (Eq. 7.129):

$$a_{ln}^t = \frac{1}{|T|} * \sum_{r \in T} \left(\frac{\hat{z}_{ln}^{r.r+t}}{\hat{z}_{nr}} \right) t \in T, n \in I, l \in \hat{L} \tag{7.129}$$

Assuming this data-driven approach, a variant of the model in Eq. 7.71 is proposed (Eqs. 7.130):

$$\underset{x_{nr}}{Min} \left[\sum_{r \in T} \sum_{n \in I} \frac{\left(x_{nr} - \hat{x}_{nr}\right)^2}{w_{nr}^2} + \sum_{r \in T} \sum_{l \in \hat{L}} \frac{\left(y_{lr} - \hat{y}_{lr}\right)^2}{q_{rl}^2} \right] \tag{7.130}$$

s.t.

$$y_{lr} = \sum_{t \in T} \sum_{n \in I} a_{ln}^t * x_{n,r-t} \forall l \in \hat{L}, \forall r \in T$$

$$-\beta x_{nr} \leq x_{n,r+1} - x_{nr} \leq \beta x_{nr} \forall n \in I, \forall r \in T$$

$$x_{nr} \geq 0 \forall n \in I, \forall r \in T$$

where the first constraint expresses, as in the analytical models, the relationships set up by the empirical assignment matrix a_{ln}^t between y_{lr}, the estimated flows at the links l with traffic detection stations $l \in \hat{L}$ for each time interval r, and $x_{n,r-t}$, the OD flows leaving the origin at time $r-t$, observed in link l at time r. The objective function in this case is formulated in terms of a quadratic distance function. Its metrics are defined, as in Eq. 7.35, respectively, by the matrix of variances, w_{nr}^2 and q_{rl}^2, of the empirical OD-matrix, $\hat{x}_{nr,}$ and the link flow measurements, $\hat{y}_{lr}$. The coefficient β in

the bounding constraints of Eq. 7.130 is the maximum change percentage of OD flows between two consecutive intervals.

Yang et al. (2017) also propose a more general variant of the model assuming that there is a correlation between the OD probe ratios, γ_{nr}, and the link probe ratios, $\theta_{lr.}$. In other words, that there exists a function $\theta_{lr} = P(\gamma_{nr})$, for which they postulate the following form (Eq. 7.131):

$$\theta_{lr} = \sum_{t\in T}\sum_{i\in I}\rho_{ln}^{t} * \gamma_{nr} = \sum_{t\in T}\sum_{i\in I}\rho_{ln}^{t} * \left(\frac{\hat{z}_{n,r-t}}{x_{n,r-t}}\right) \forall l \in \hat{L}, \forall r \in T \tag{7.131}$$

where ρ_{ln}^{t} define the assignment matrix of probe ratios, which is assumed to be computed empirically from the collected data according to the main hypothesis of the method. These link probe ratios depend on the estimated OD-matrix, x_{nr} as expressed in Eq. 7.131, being therefore new variables of the model. Assuming that the available GPS data allow estimating the empirical values $\hat{\theta}_{lr,}$ they can be added to the objective function (Eq. 7.130) yielding the enhanced model in Eq. 7.132:

$$Min_{x_{nr}}\left[\sum_{r\in T}\sum_{n\in I}\frac{\left(x_{nr}-\hat{x}_{nr}\right)^2}{w_{nr}^2} + \sum_{r\in T}\sum_{l\in\hat{L}}\frac{\left(y_{lr}-\hat{y}_{lr}\right)^2}{q_{rl}^2} + \sum_{r\in T}\sum_{l\in\hat{L}}\frac{\left(\theta_{lr}-\hat{\theta}_{lr}\right)^2}{v_{rl}^2}\right] \tag{7.132}$$

s.t.

$$y_{lr} = \sum_{t\in T}\sum_{n\in I} a_{ln}^{t} * x_{n,r-t} \forall l \in \hat{L}, \forall r \in T$$

$$\theta_{lr} = \sum_{t\in T}\sum_{i\in I}\rho_{ln}^{t} * \left(\frac{\hat{z}_{n,r-t}}{x_{n,r-t}}\right) \forall l \in \hat{L}, \forall r \in T$$

$$-\beta x_{nr} \le x_{n,r+1} - x_{nr} \le \beta x_{nr} \forall n \in I, \forall r \in T$$

$$x_{nr} \ge 0 \forall n \in I, \forall r \in T$$

where, as before, v_{rl}^2 is the variance of the observed ratios. Since the optimization model is quadratic, the gradient can be easily calculated and a gradient algorithm is proposed to numerically solve the problem.

7.6.2 Accounting for Mobility Learning from ICT Data Collection

Cascetta et al. (2013) formulate the hypothesis that "*an OD estimator can be based on the assumption of constant distribution shares across larger time horizons with respect to the within-day variation of the production profiles, leading to an estimator that dramatically improves the unknowns/equations ratio*". Krishnakumari et al. (2019) propose to go a step further. They assume that all realized travel times are available over all (shortest) paths. Also, that it is only necessary to specify how many of the shortest paths are actually used for each OD pair and the proportions of each OD flows over these used shortest paths. These proportions are a behavioral assumption at the macroscopic scale (a path flow proportion), and not in the form of a detailed route choice model with (elaborated) trade-offs.

Nevertheless, the assumptions in Krishnakumari et al. (2019) about the distribution of traffic over the network are not sufficient to estimate the underlying OD-matrix, They must be complemented with additional information that, for instance, can be provided by measured link flow counts $\hat{y}_{lt}$ from counting stations, measured at links $l \in \hat{L}$ at time $t \in T$. Link flow counts y_{lt} that, as shown in Eq. 7.76, can be estimated in terms of the relationships between flows and OD flows x_{ijr} departing from origin i with destination j at time interval r, arriving at link l at time t, and the assignment matrix a_{ijr}^{lt}.

However, to be valid, these relationships must be set up considering that counts in or downstream congestion are not informative of demand, but of (discharge) capacity, as shown in Frederix et al. (2011). Information on demand is only provided if y_{lt} are estimated in uncongested conditions, and no path flows for OD pair $(i, j) \in I$ using a path k to which l belongs, experience a bottleneck upstream before crossing link l. Therefore, in order to overcome these limitations, the computation of the assignment matrix, or that of any related terms, must be done in a way that explicitly accounts for congestion effects.

The approach proposed by Krishnakumari et al. (2019) assumes that, in addition to the availability of OD travel times, also the productions P_{ir}, i.e., the total outgoing flows from each origin i, during period r, as well as the attractions A_{jr}, i.e., the total incoming flows to each zone j during period r, are observable and, therefore, available.

The availability of these inputs from the observed data is based on a methodology proposed by López et al. (2017b) that is based on specific Data Analytics techniques suited to build consensual 3D speed maps by clustering techniques from link speeds. These speeds are estimated from field data by a heuristic procedure defined in López et al. (2017a). According to the authors, this procedure can exploit classical data (e.g., from inductive loops or cameras) as well as those from more modern data sources (e.g., mobile phone records, GPS tracking, etc.). The methodology is illustrated in the referenced papers for a case in which link speeds were estimated from individual travel times recorded by TV cameras equipped with LPR technology. However, it is extensible to other technologies as long as the requirements are met.

In any case, field data must be appropriately cleansed out and the outliers removed. Krishnakumari et al. (2019) discuss several procedures for this purpose, among them a moving average process where, if τ_n is the $n-$ th realized travel time for a given OD pair, Eq. 7.133 gives the moving average:

$$\overline{\tau}_n = \frac{1}{k}\sum_{i=0}^{k-1}\tau_{n-i} \tag{7.133}$$

Outliers are defined by $\overline{\tau}_n + \Delta\tau$, where $\Delta\tau$ is a time window empirically determined, for instance, as the standard deviation times recorded during the peak demand. The mean of the observed travel times for a given OD pair (i, j) at a given period is considered the travel time from i to j at time t,$tt_{ij}(t)$. Additionally, the considered k-shortest paths as the most likely used between each OD pair. For a particular one(i, j), a path L_p is characterized by a sequence of links $L_p = \left(l_{p1}, l_{p2}, \ldots, l_{pn}\right)$. Then, the path speed is (Eq. 7.134):

$$s_p = \frac{\text{dist}(L_i)}{tt_{ij}(t)} \tag{7.134}$$

Krishnakumari et al. (2019) also consider various approaches to impute link speeds when no data are available.

The main assumption behind the approach proposed by Lopez et al. (2017b) is that the availability of the data provided by these more modern data sources allows finding empirically driving insights of human mobility, namely, those concerning their dynamic aspects, and thus enables their use in mathematical models aimed at predicting that dynamic mobility. This means to investigate the regularity of macroscopic mobility patterns, how they vary within days and from day to day. For that purpose, Lopez et al. (2017b) propose a methodology based on what they call 3D maps, in essence spatial–temporal speed cluster maps, which are a joined partition of space (i.e. the road network) and time into homogeneous clusters characterized by constant mean speeds. The proposed approach considers that link speed data can be reconstructed from trip travel time observations with Eq. 7.134, as in Lopez and al. (2017b), and that the network is coded in Open Street Map Geographical Information System (OSM GIS) Database, also used to compute all shortest paths. The cluster building process is based on the following partitioning criteria:

- All clusters should contain a single connected component. In other words, all links in the cluster are reachable within the cluster.
- An intra-cluster homogeneity criterion, formulated in terms of the minimization of the internal speed variance for all clusters. If n is the number of clusters, the total within cluster variance TV_n is given by (Eq. 7.135):

$$TV_n = \frac{1}{\sum_{i=1}^{n} n_i} * \left(\frac{\sum_{i=1}^{n} n_i * s_i^2}{s^2}\right) \tag{7.135}$$

where n_i is the number of links in cluster i, s_i the standard deviation of links speeds in cluster i, and s the standard deviation for the whole network. It is assumed that link speeds have been estimated from Eq. 7.134.

- An inter-cluster dissimilarity criterion that maximizes the difference in speed between neighbor clusters, where the inter-cluster dissimilarity is given by Eqs. 7.136 and 7.137:

$$CCD_n = \frac{\sum_{i=1}^{n} \sum_{k=i+1}^{n} \delta_{ik} * \sqrt{n_i * n_k} * |\overline{v}_i - \overline{v}_k|}{\sum_{i=1}^{n} \sum_{k=i+1}^{n} \delta_{ik} * \sqrt{n_i * n_k}} \tag{7.136}$$

$$\delta_{ik} = \begin{cases} 1 & \text{if clusters } i \text{ and } k \text{ have a common border} \\ 0 & \text{otherwise} \end{cases} \tag{7.137}$$

where $\overline{v}_i$ is the mean speed in cluster i.

Lopez et al. (2017b) test three different clustering approaches, *k-means*, DBSCAN, and S-cut and conclude that, at least in the case study reported in the paper, *k-means* is the most economical in terms of computational cost to obtain the envisaged 3D speed maps. Furthermore, assuming that the observational data cover a period of M days, they add a new process to find commonalities in these days' congestion patterns, the so-called "consensual" patterns, by means of *Consensus Learning Techniques* (Filkov and Skiena 2004).

The approach proposed by Krishnakumari et al. (2019) uses these results for different purposes:

- To estimate or predict the production and attraction patterns using the identified 3D speed and flow patterns (possibly augmented with other data) using machine learning techniques (especially Neural Network techniques, although other techniques could also be used).
- To compute N weighted (by travel time) shortest paths, where N is an assumption on how many alternative routes are used on average for each OD flow on these paths.
- To estimate path flows on the used paths assuming that are inversely proportional to the realized travel times on these paths, considering path overlap, and under the additional constraint that the path flow solution space is determined by all admissible link flow counts.

Let's assume that x_{ijr}^k is the path flow from origin $i \in O$ (where O = {set of all origins}) to destination $j \in D$ (where D = {set of all destinations}), departing from origin i at time period $r \in T$ (where T is the time horizon) on path $k \in N_{ij}^k$ (where N_{ij}^k is the set of all paths between origin i and destination j at time period k}; x_{ijr} is the OD flow from origin $i \in O$ to destination $j \in D$, departing from origin i at time period $r \in T$; P_{ir} is the production of origin i during period r, A_{jr} is the attraction of destination j during period r; TT_{ijr}^k is the travel time for vehicles traversing path k from origin i to destination j departing from i in time period r; P_{ijr}^k is

the proportion of vehicles traveling on path k from origin i to destination j departing from i in time period r and $\hat{y}_{lr}$ is the measured flow count in link l at time period r. P_{ir} is the sum of all outgoing flows from i at this time period along all paths $k \in N_{ij}^{r}$ from i to all destinations $j \in D$ (Eq. 7.138):

$$P_{ir} = \sum_{j \in D} \sum_{k \in N_{ij}^{r}} x_{ijr}^{k} \tag{7.138}$$

In a similar way, the attraction A_{jr} of destination j during period r is the sum of all incoming flows to destination j from all origins $i \in O$ along all paths $k \in N_{ij}^{k}$ (Eq. 7.139):

$$A_{jr} = \sum_{i \in O} \sum_{k \in N_{ij}^{r}} x_{ijr}^{k} \tag{7.139}$$

Since links speeds are available, path travel times TT_{ijr}^{k} can be calculated. From them, a behavioral assumption can be made on the proportion of trips using each available path in terms of each utility, which is defined by the path travel time. Krishnakumari et al. (2019) estimate this path proportion with the modified logit-based model proposed by Ben-Akiva and Bierlaire (1999) (Eq. 7.140):

$$P_{ijr}^{k} = \frac{e^{TT_{ijr}^{k} * (1 - PS^{k})}}{\sum_{p \in N_{ij}^{r}} e^{TT_{ijr}^{p} * (1 - PS^{p})}} \tag{7.140}$$

In this Eq. 7.140 a correction term PS^{k} is added to the deterministic component of the discrete-choice mode. It is the path size factor defined by Eqs. 7.141 and 7.142:

$$PS^{k} = \sum_{a \in \text{Path } k} \left(\frac{l_a}{L^{k}} \right) * \frac{1}{\sum_{p \in N_{ij}^{r}} \delta_{ap}} \tag{7.141}$$

$$\delta_{ap} = \begin{cases} 1 & \text{if link } a \text{ belongs to path } p \\ 0 & \text{otherwise} \end{cases} \tag{7.142}$$

where l_a is the length of link a, L^{k} is the length of paths k, and δ_{ap} is the link-path incidence matrix. The path size factor tries to capture the correlations between alternative paths measuring the dependencies in terms of a certain degree of similarity among the shared links. The calculation of the path proportions allows setting up the relationships between the OD flows, $x_{ijr,}$ and the path flows, x_{ijr}^{k} (Eq. 7.143):

$$x_{ijr}^{k} = P_{ijr}^{k} * x_{ijr} \forall i \in O_i, \forall j \in D_j, \forall r \in T, \text{and } k \in N_{ij}^{r} \tag{7.143}$$

The number of paths N_{ij}^{r} can be exponentially large but, in practice, as not all of them are significantly used, this number can be reduced to a smaller set $N_{ij}^{r*} \leq N_{ij}^{r}$. This smaller set can be identified as part of the data analytics procedures to estimate the values of the model components. This leads to the approximation of the estimated OD-matrix as in Eq. 7.144:

$$x_{ijr} = \sum_{k \in N_{ij}^{r*}} x_{ijr}^{k} \tag{7.144}$$

This approximation is sufficiently good if N_{ij}^{r*} has been properly defined.

The relationship between link flows and path flows can be reformulated explicitly considering the effects of congestion in order to account for the conditions discussed above. That is, that flows y_{lr} measured in link l at time r are informative of path flows crossing the link only if they are not congested at that time and if none of the links upstream of it experiences a bottleneck. The approach chosen by Krishnakumari et al. (2019) considering the subset of paths to which link l belongs and satisfying these conditions can be formulated as follows. If $\wp_r^l$ is the set of paths to which link l belongs at time r, the subset of paths satisfying the conditions is given by (Eq. 7.145):

$$\wp_{ijt}^{k} \in \wp_r^l, t \leq r - \overline{TT}_{ijr}^{k \setminus l} \ \forall i, j, k \quad k \text{ all paths traversing } l \text{ during period } r \tag{7.145}$$

where $\overline{TT}_{ijr}^{k \setminus l}$ estimates the partial arrival travel times to link l along the paths in $\wp_{ijt}^{k}$. This implies that (Eq. 7.146):

$$\sum_{\wp_{ijt}^{k} \in \wp_r^l} P_{ijt}^{k} * x_{ijr} = \begin{cases} 0 & \text{if link supstream of } l \in \wp_{ijt}^{k} \text{ are congested} \\ P_{ijt}^{k} * x_{ijr} & \text{otherwise} \end{cases} \tag{7.146}$$

Thus, if $\hat{y}_{lr}$ are the link flows measured at links $l \in \hat{L} \subseteq L$ equipped with detection stations, their relationships with the OD flows x_{ijr} can be stated with Eq. 7.147:

$$\hat{y}_{lr} = \sum_{(i,j) \in I} P_{ijr}^{k} * x_{ijr} \ \forall l \in \hat{L}, \forall r \in T \tag{7.147}$$

Together with the corresponding reformulations of Eqs. 7.134 and 7.135 and in terms of Eq. 7.146, a system of equations (Eqs. 7.148 and 7.149) is defined:

$$P_{ir} = \sum_{j \in D} x_{ijr} \ \forall i \in O, \forall r \in T \tag{7.148}$$

$$A_{jr} = \sum_{i \in O} x_{ijr} \ \forall j \in D, \forall r \in T \tag{7.149}$$

As highlighted in Krishnakumari et al. (2019) "*this system is underdetermined or overdetermined (or rare cases full rank) depending on the available link counts and the choice and number of link paths for each OD pair*". To solve the system, the authors propose to use the constrained least squares algorithm of Altman and Gondzio (1999), either with lower bounds set to 0 to ensure non-negative solutions, or without bounds when no solution exist and ignoring the negative values in computing the solution error.

A potential limitation of the proposed approach arises for large networks. That is, when the number of origins and destinations grows and, then, the number of OD flows grows quadratically. However, the number of Eqs. 7.148 and 7.149 in the system only grows linearly, as link flow equations do (Eq. 7.147), assuming also an increase in the number of detection stations. The authors propose to use in this case the dimensionality reduction techniques studied in Djukic et al. (2012), which are based on the application of the *Principal Components Analysis* (Jolliffe 2002).

To end this section, it should be noticed that this data-driven approach is the planned forthcoming OD-estimation method in future versions of the corresponding modules of Aimsun Next and Live software platforms for traffic analysis and management.

7.6.3 Estimating Assignment Matrices from FCD Data

As mentioned, the computational burden associated with the DTA required in the analytical approaches to the DODME problem, which is necessary to estimate the assignment matrix, and the existing doubts on how to integrate the additional information that can be available, have fostered research on these issues not only among researchers, but also among practitioners and developers of professional software platforms. An example of this motivation can be found in a recent work of the team supporting the OPTIMA traffic management platform (Mitra et al. 2020). This platform is aimed at estimating base OD demand matrices for large-scale networks using the information that can be extracted from large amounts of FCD data and link flow counts. The main assumption, similar than that of previous approaches, is that a detailed analysis of FCD trajectories, if properly and accurately done, enables the estimation of the two main required inputs: (i) a revealed OD-matrix X^0 extracted from the FCD trajectories, playing the role of seed matrix and (ii) information to build from FCD data a reliable assignment matrix that can replace the one provided by DTA in analytical approaches.

A critical point is that of the quality of the FCD data, since they can be poor, not homogeneous, or biased. However, Mitra et al. (2020) claim that, even in these cases, it is possible to take advantage of these data. Their suitable cleansing and filtering and their clustering according to similar average behaviors are useful techniques to apply. Also, the use of specialized map-matching algorithms matching each individual raw GPS trajectory on the transportation graph in order to reconstruct the most likely paths in this graph (Hart et al. 1968; Marchal et al. 2004; Quddus et al. 2007; Kubicka et al.

2018; Millard-Ball et al. 2019). The map-matched trajectories can be associated with origin and destination zones, departing from origin zones at specific times of the day.

Let's assume that X is the estimated OD vector of size $|I|*|T|$ (where I is the set of OD pairs and T the set of time intervals), that $\widehat{Y}$ is the vector of traffic counts (of size $\left|\widehat{L}\right| * |T|$, being $\widehat{L}$ the set of links with counting stations), and that A is the estimated assignment matrix from FCD data. Then, mapping the estimated OD flows to the estimated link flow counts Y (with $Y = A * X$) and simplifying the formulation for a simple fixed time interval (no interdependencies between time intervals are assumed in this approach. See Mitra et al. (2020) for additional details), the DODME problem can be formulated as in Eq. 7.150:

$$\text{Min } \varphi(X) = \frac{1}{2} * \|A * X - \widehat{Y}\|^2 + \frac{\lambda}{2} * \|X - \widehat{X}\|^2 \tag{7.150}$$

where λ is the relative weight of the demand term, and $\widehat{X}$ is the reference demand vector, whose $ij-$ th element is given by Eq. 7.151:

$$\widehat{X}_{ij} = \gamma * \alpha_i * \beta_j * X_{ij}^0 \ \forall i \in O, \forall j \in D \tag{7.151}$$

being O the set of origins, D the set of destinations and X^0 the observed seed OD-matrix from FCD trajectories. γ is a constant factor that homogenously scales all OD pairs, and $\alpha_i, \forall i \in O$ and $\beta_j, \forall j \in D$, respectively, are the generation and attraction factors for each origin and each destination.

The solution to Eq. 7.150 is found by an iterative process that generates a sequence of feasible solutions $\{X^k\}$. This is done in such way that a new solution is found at iteration $k+1$ from the solution at iteration k by moving a step of length $\theta^k \in (0,\ 1]$ along a feasible descent direction ΔX^k (Eq. 7.152):

$$X^{k+1} = X^k + \theta^k * \Delta X^k \tag{7.152}$$

Since $\varphi(X)$ is a quadratic problem, the descent direction can be found by a Newton method solving with Eq. 7.153:

$$\Delta X^k = \left[\nabla^2 \varphi\left(X^k\right)\right]^{-1} * \nabla \varphi\left(X^k\right) \tag{7.153}$$

where $\nabla \varphi\left(X^k\right)$ is the gradient of $\varphi(X)$, and $\nabla^2 \varphi\left(X^k\right)$ the Hessian at X^k. In practice, Eq. 7.153 can be solved efficiently without inverting the Hessian and, since $\varphi(X)$ is quadratic, the solution can be exactly found in one step if the Hessian is definite positive.

Several alternatives have been proposed (Mitra et al. 2020) to estimate the values of factors γ, α_O and β_D, where α_O and β_D are the vectors of attraction and generating factors. An example procedure that simultaneously optimizes α_O and β_D could consist

in (i) calculating the optimal value of γ the common global factor by solving the quadratic problem in Eq. 7.154 and (ii) calculating the optimal values of α_O and β_D by solving Eq. 7.155:

$$\begin{aligned} &\text{Min } \varphi(X) \\ &\gamma \geq 0 \end{aligned} \tag{7.154}$$

s.t. $\alpha_O = \beta_D$=1

$$\begin{aligned} &\text{Min } \varphi(X) \\ &\alpha_O, \beta_D \geq 0 \end{aligned} \tag{7.155}$$

Mitra et al. (2020) present promising results of this approach applied to the large-scale network of Turin, with 438 zones, 96,420 links, 6,352 nodes, 1203 counting locations and GPS data for 1 year.

The potential problems of dealing with GPS data reported when discussing previous approaches fostered the search for other practical solutions. Most of these problems concern the unbiased reconstruction of vehicle trajectories and the estimation of the observed seed OD-matrix X^0 and are usually linked to the fact that most of the available commercial GPS data are obtained from non-homogeneous vehicle fleets (e.g., indiscriminate mix of commercial vehicles and passenger cars). Another source of issues is trajectories being split by random identity changes due to privacy reasons. However, once these data properly cleansed and filtered out, the waypoints or POIs (Points of Interest) supplied by GPS data can be considered reliable. These are usually given as an ordered sequence of waypoints containing the information (*IDk, date, ts(kl), latkl, longkl*), as illustrated in Table 7.1. *IDk* is the identity of each trip k, the date stands for the recording date, *ts*(k,l) is the time tag for the $l-$ th observation of trip k and *latkl* and *longkl,* respectively, are its latitude and longitude.

However, these geographically referenced data do not usually correspond to the analyzed road network. Therefore, as already mentioned, they must be properly map-matched to transform these sequences of waypoints in points corresponding to

Table 7.1 Example of GPS recorded waypoints

ID	Date	Time stamp	Latitude	Longitude
4,261,353	2019–11-30	22:43:58	45.445988	9.1244048
4,261,353	2019–11-30	22:44:27	45.445496	9.1241952
……	……	……	……	……
4,261,353	2019–11-30	22:50:57	45.444767	9.1192517
4,261,355	2019–11-30	22:43:58	45.445980	9.1247048
4,261,355	2019–11-30	22:44:27	45.445574	9.1192821
……	……	……	……	……
4,261,355	2019–11-30	22:50:57	45.444767	9.1197541
……	……	……	……	……

paths on that network. The most used procedures (Marchal et al. 2004; Schuessler and Axhausen 2009; Pereira et al. 2009; Rahmani and Koutsopoulos 2013; Kubicka et al. 2018) assign each waypoint to a point in the nearest link of the network. There are available tools provided by software platforms to perform this operation, as, for instance, OpenLR (OpenLR 2020), or GPX (PTV Visum 2020). An example on how this works is depicted in Fig. 7.7, in which the red stars are the waypoints and the red numbers near the links are the relative position of the waypoint projection over the target link. Timestamps for waypoints are depicted in green.

Link travel times can be heuristically estimated from waypoint timestamps according to their sequence (Ros-Roca et al. 2021b). In this example, for all links in the sequence, the interpolated travel time for a link is the sum of the timestamp differences of two consecutive waypoints mapped in 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 (lk is the length of link k in Fig. 7.7). For instance, the travel time for link l_3 can be estimated taking into account that the travel time for the trip between the 3rd and 4th waypoints is 20 s, and that it 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(Eq. 7.156, with the result in s):

$$tt_3 = \frac{l_3}{0.2{*}l_2 + l_3 + 0.7{*}l_4} * 20 \tag{7.156}$$

The estimated travel time in link l_4 is obtained by adding two parts, the first part is the travel time proportion between the 3rd and 4th 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 4 and 5th timestamps (a fraction of 7 s, which is the travel time between waypoints, calculated as 0.3 of the l_4 distance over the total distance between the 4th and 5th waypoints, that is 0.3 l_4+0.2 l_5). Overall, the travel time in link l_4 is given by Eq. 7.157 (in s):

$$tt_4 = \frac{0.7 * l_4}{0.2 * l_2 + l_3 + 0.7 * l_4} * 20 + \frac{0.3{*}l_4}{0.3{*}l_4 + 0.2 * l_5} * 7 \tag{7.157}$$

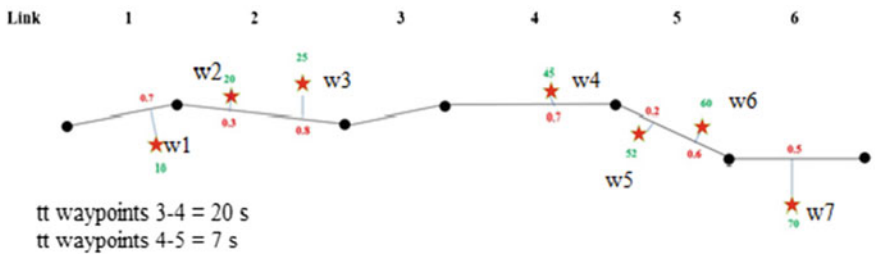

Fig. 7.7 Schematic overview of the map-matching process

Finally, once all the waypoint sequences are converted to several paths with full details at the link level, the link travel times are averaged. The outcome of this process is the set of observed link travel times at each time period t, $\widehat{tt}_{lt} \forall l \in L, \forall t \in T$, for all links in the network that are monitored by GPS tracking. That is, the dataset of estimated link travel times. Despite possibly being huge the quantity of trajectories available for the target network, which will depend on the penetration rate of devices with GPS among the population, the final sample may uncover links. It is also possible that some of them are not fully covered by time information, as, for instance, the first and last links in each sequence (e.g., links 1 and 6 in the example in Fig. 7.7). Moreover, the procedure that infers link travel times can produce non-feasible values when they are below the free-flow link travel time. In these situations, scaled travel times are used (Eqs. 7.158 and 7.159):

$$\hat{tt}_{l't} = R * tt_{0l'} \tag{7.158}$$

$$R = \text{mean}_{l \in GPS} \left(\frac{\hat{tt}_{lt}}{tt_{0l'}} \right) \tag{7.159}$$

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. That is, R is the arithmetic mean of the expanding factors found for each link and can be understood as a global expanding factor linked to the congestion effect. The methodological process for generating the observed link travel times dataset is summarized in Fig. 7.8.

The estimated average link travel times $\widehat{tt}_{lt}$ for each link $l \in L$, for each time interval $t \in T$ can be used to generate a plausible *Route Choice Set* $\mathcal{K} = \left\{K_{ijr}, \forall i \in O, \forall j \in D, \forall r \in \mathcal{T}\right\}$ of the most likely used paths between each origin and each destination at each departure time. This can be done by applying variants of Dijkstra-based algorithms explicitly accounting for commonalities between paths in terms of shared links, as in Krishnakumari et al. (2019). However, as we are in this case considering link travel times, other alternatives like those proposed by Chabini (1998), dealing directly with time-dependent shortest paths, can be more appropriate. Nassir et al. (2014), Janmyr and Wadell (2018), use the penalization of overlapping in

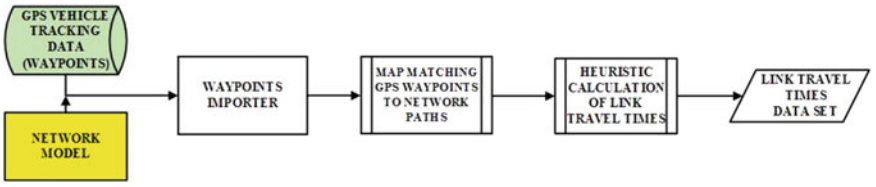

Fig. 7.8 Conceptual methodological approach to the process of importing waypoints into a network model and their use to estimate link travel times

terms of "commonality factors" proposed by Cascetta et al. (1996), Cascetta (2001) as a measure of similarity between alternatives. This allows overcoming the problems derived from the basic hypothesis of irrelevant alternatives with discrete-choice models reducing the systematic utility of paths, being this utility measured in terms of travel time, in proportion to its level of overlapping with other alternative paths. Such procedures can be additionally strengthened by applying the modification of the variant of Bovy et al. (2008) proposed by Janmyr and Wadell (2018). According to this modification, paths in K_{ijr} are denoted here as $k(i,j,r) \in K_{ijr}$ in order to explicitly show the dependence on (i,j,r). Let's assume that the sequence of links that compound a certain path $k(i,j,r)$ is $\Gamma_{k(i,j,r)} = \{e_1, \ldots, e_m\}$. Then, the proportion of paths choice for each path in the set K_{ijr} is calculated in terms of the following modified discrete logit-based choice model that uses the commonality factor (CF) for each OD pair and time period (Eqs. 7.160 and 7.161):

$$CF_{k(i.j.r)} = \frac{1}{\mu_{CF_k}} * \sum_{a \in \Gamma_{k(,i,j,r)}} \left(\frac{l_a}{L_{k(i,j,r)}} * \log \left(\sum_{h \in K_{ijr}} (\delta_{ahr} + 1) \right) \right) \tag{7.160}$$

$$P_{k(i,j,r)} = \frac{\exp[\mu_{P_k}(-\hat{tt}_{k(i,j,r)} - CF_{k(i,j,r)})]}{\sum_{h \in K_{ijr}} \exp[\mu_{P_k}(-\hat{tt}_{h(i,j,r)} - CF_{h(i,j,r)})]} \tag{7.161}$$

where $\delta_{ahr} = 1$ if path $h \in K_{ijr}$ uses link a at time r and 0 otherwise, l_a is the length of link a and $L_{k(i,j,r)}$ is the total length of path $k \in K_{ijr}$. In order to adapt magnitudes for the discrete-choice summation, μ_{P_k} and μ_{CF_k} are parameters fixed as in Eq. 7.162:

$$\mu_{P_k} = \mu_{CF_k} = \frac{1}{\text{mean}_{k \in K_{ijr}} \left(\hat{tt}_{k(i,j,r)} \right)} \tag{7.162}$$

These calculations provide the flow distribution for each path on the basis of observed path travel times, which are the summation of the observed time-dependent link travel times. That is, they consider the arrival time, $\hat{tt}_{at(k)}$, at each link a belonging to the path $k(i,j,r)$ (Eq. 7.163):

$$\hat{tt}_{k(i,j,r)} = \sum_{a \in \Gamma_{k(i,j,r)}} \hat{tt}_{at(k)} \tag{7.163}$$

Once $P_k = \left\{ P_{k(i,j,r)} \right\}$ is determined from the k shortest paths obtained from the estimated travel times, the estimated time-dependent assignment matrix $\overline{A} = \left[\overline{a}_{ijr}^{lt} \right]$ can be calculated with Eq. 7.164 and 7.165:

$$\overline{a}_{ijr}^{lt} = \sum_{k \in K_{ijr}} \delta_{k(i,j,r)}^{lt} * P_{k(i,j,r)} \quad \forall i,j,r,l,t \tag{7.164}$$

$$\delta_{k(i,j,r)}^{lt} = \begin{cases} 1 \text{ if path } k(i,j,r) \text{ uses link } l \text{ at time } t \\ 0 \text{ otherwise} \end{cases} \tag{7.165}$$

where $\delta_{k(i,j,r)}^{lt}$ is the estimated incidence indicator.

This is the estimated assignment matrix that can replace the calculated assignment matrix from DTA in an alternative formulation of DODME. Therefore, the relationship in Eq. 7.76 that the assignment matrix establishes between estimated link flows y_{lt} and estimated OD flows x_{ijr} can now be rewritten as Eq. 7.166:

$$y_{lt} = \sum_{i \in O} \sum_{j \in D} \sum_{r=1}^{t} \bar{a}_{ijr}^{lt} * x_{ijr} \tag{7.166}$$

If data collected from a sample of GPS-tracked vehicles is available and if it is possible to create a discrete time estimate of a seed OD-matrix from it, that is, the observed OD-matrix $X^0 = \left[x_{ijr}^0\right]$, this last matrix could be expanded to estimate the OD-matrix in terms of the scaling factors per origins, $\alpha_i, \forall i \in O$, and per destinations $\beta_j, \forall j \in D$, such that (Eq. 7.167):

$$x_{ijr} = \alpha_i * \beta_j * x_{ijr}^0 \tag{7.167}$$

It can be assumed, as in all previous formulations, that a reliable historical OD-matrix X^H is available. As already mentioned, this assumption would be questionable in long-term planning applications, as this matrix could be either largely outdated or simply not exist. However, its existence is a reasonable hypothesis in traffic management applications, where a surveillance system is already in operation and provides rich structural information (Ashok and Ben-Akiva 1993; Ben-Akiva et al. 2001; Djukic et al. 2018; Aimsun 2020). Once the existence of a historical OD-matrix accepted, the DODME problem can be reformulated in terms of the estimation of the scaling factors α_i, and β_j, in the following way (Eq. 7.168):

$$\begin{aligned} \underset{\alpha_i, \beta_j}{\text{Min}} \Bigg[& w \left(\sum_{i \in O} \sum_{j \in D} \sum_{r=1}^{t} \left(x_{ijr}^H - \alpha_i * \beta_j * x_{ijr}^0 \right)^2 \right) \\ & + \sum_{l \in \hat{L}} \sum_{t \in \mathcal{T}} \left(\hat{y}_{lt} - \sum_{i \in O} \sum_{j \in D} \sum_{r=1}^{t} \alpha_i * \beta_j * \bar{a}_{ijr}^{lt} * x_{ijr}^0 \right)^2 \Bigg] \\ & s.t.\ \alpha_i, \beta_j \geq LB \quad \forall i \in O, \forall j \in D \end{aligned} \tag{7.168}$$

The problem variables are multiplicative scaling factors for each origin α_i and each destination β_j, which significantly reduces the number of variables from $|I| * |J| * |T|$ to $|I| + |J|$. Moreover, the fact of using the scaling factors as variables aims at

preserving the structure of the seed OD-matrix, as gravity models do. Since the model is no longer quadratic and is bounded from below, other optimization procedures could be advisable. Ros-Roca et al. (2021a, b) report good results using the L-BFGS-B method (Morales and Nocedal 2011). It is a quasi-Newton method suitable for constrained nonlinear problems with a high number of variables, and it efficiently reduces the memory requirements and the computational burden.

Theoretically, the lower bound (LB) should be a non-negativity constraint for all the scaling factors α_i, β_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 to certain destination would become null. Therefore, considering that the seed OD-matrix in Eq. 7.167 comes from reliable information on mobility, the scaling factors cannot be null and the lower bound should therefore be larger than zero.

If the quality of the observed seed matrix X^0 is questionable due to the conditions in which GPS data have been collected, (this could be the case for some commercial GPS data, as mentioned) but the historical matrix X^H is very reliable, both matrices could be fused to generate an improved seed matrix (Ros-Roca et al. 2021b).

7.7 Measuring the Quality of the OD Estimates

A critical question when estimating an OD is how the quality of the resulting estimated matrix can be assessed. This quality has been usually assessed in terms of the convergence of the objective function and the R^2 fit between measured and simulated traffic counts at links with counting stations. From the optimization point of view, these measures are a good selection because they can show explicitly that the used method works specifically for the purpose of minimizing the objective function designed as an OD-matrix estimation problem. Furthermore, it verifies that the estimated OD acceptably replicates the observed flows. However, despite R^2 being a good indicator of how the optimization problem is performing, it can produce misleading results. For example, it is possible that a high regression is achieved but the resulting estimated OD-matrix does not match the reality of the demand pattern and the internal mobility of the study area. Therefore, some other indicators that evaluate the mobility patterns in the OD-matrices are needed.

These indicators do not pay any attention to the quality of the results from a structural point of view. In other words, they do not distinguish whether the traffic OD patterns resulting from the adjustment approach exhibit an acceptable degree of structural similarity to the historical OD-matrix (when a reliable one is available), or whether the used approach provides a perturbed matrix that, even fitting the observed link flows, is structurally different. If this last is the case, it could be doubtful that such a structural change could be physically interpretable in terms of the underlying transportation system. Particularly when considering increases or decreases in the total number of trips between transportation zones that cannot be consistent with the socioeconomic attributes of the zone generating or attracting them. Looking at the link-path relationships visualized on the right-hand side of Fig. 7.3, it may happen

that the optimization process used to solve the DODME problem locally behaves as a retrogression model. This model could pull forth and back the OD flows in paths crossing the link with the counting station in order to fit measured and simulated flows as well as possible. This would just be a consequence of a numerical procedure, ignoring the underlying structure of the modeled reality.

A widely used proposal has been to resort to other goodness of fit indicators, like the Mean Square Error (MSE) and other similar ones (Hollander and Liu 2008). Other approaches consider alternative formulations of the objective function in terms of the distance function between the historical and the estimated OD-matrices. Classical distances between vectors can be applied to matrices by considering these matrices $X^H, X \in \mathcal{M}_n(\mathbb{R})$ as vectors of $X^H, X \in \mathbb{R}^{n \times n}$. Euclidean, Manhattan, and other vector distances can be used in the objective function of the OD-estimation problem aimed at minimizing the distance between matrices. However, these metrics, although comparing the OD-matrices cell by cell, do not have the ability to capture the differences and similarities of many aspects, such as their structure. Therefore, the spatio-temporal similarities of OD-matrices are not captured by these measures (Djukic 2014) and it seems clear that alternatives to these vector measures must be used. Djukic (2014) or Behara (2019) present a reference matrix M_R, which could be considered as a hypothetical ground truth matrix, X^{GT}, and two additional matrices M_1 and M_2 generated by perturbations of that reference matrices, such that they clearly have different structures but are indistinguishable in terms of measures like MSE or similar. The example in Fig. 7.9. illustrates this situation. Let us consider the three matrices,M_R, M_1 and M_2, the reference and perturbed matrices, respectively, generated following the guidelines of Djukic (2014).

Comparing M_R, M_1 and M_2 in terms of MSE, the results are that MSE (M_R, M_1) = MSE (M_R, M_2) = 16. Therefore, MSE does not help to discriminate which of the two matrices M_1 and M_2 is closer to M_R.

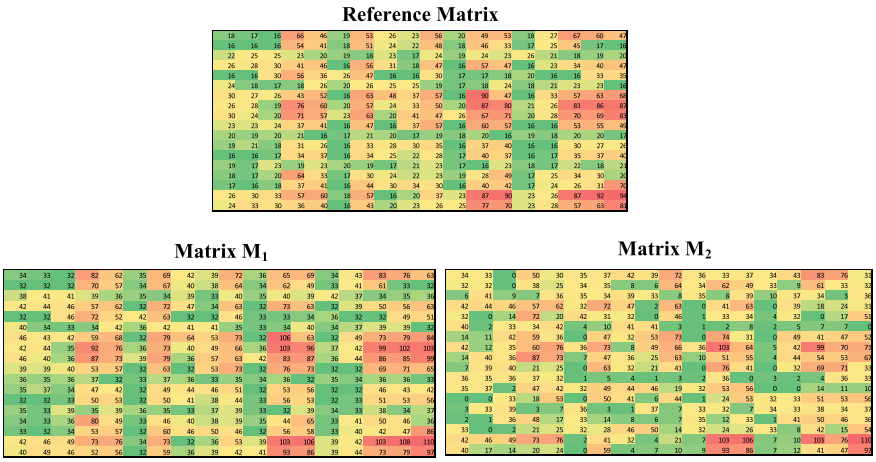

Fig. 7.9 Comparing matrices with the same MSE and different structures

Djukic (2014) proposes a measure of structural similarity based on the *Image Quality Assessment* process for comparing two different images (Wang et al. 2004). This measure is the Structural SIMilarity index (SSIM) for a matrix of pixels, that is, the product of three different comparison components: luminance, contrast, and structure. Luminance corresponds to the intensity of illumination, which is indeed the mean of the different pixels in a sub-matrix. Contrast is the squared average between pixels once the luminance is removed, thus making it the standard deviation. Finally, the structure is compared by using the covariance between the two matrices. These three factors are firstly transformed with the aim of adjusting them to the interval [0, 1], where 1 means perfect match and 0 means no match. SSIM is therefore a similarity measure that is independent of the magnitude of the values in the matrix. Equation 7.169 gives the formula summarizing this explanation:

$$SSIM\,(x, y) = l(x, y)^{\alpha} * c(x, y)^{\beta} * s(x, y)^{\gamma} \tag{7.169}$$

where luminance, contrast, and structure are, respectively, defined by Eqs. 7.170–7.172:

$$l(x, y) = \frac{2{*}\mu_x * \mu_y + C_1}{\mu_x^2 + u_y^2 + C_1} \tag{7.170}$$

$$c(x, y) = \frac{2{*}\sigma_x * \sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \tag{7.171}$$

$$s(x, y) = \frac{\sigma_{xy} + C_3}{\sigma_x \sigma_y + C_3} \tag{7.172}$$

and μ_x, σ_x, μ_y, σ_y, σ_{xy} are the mean, standard deviation, and covariance of the vectors x and y, respectively. C_1, C_2, C_3 are stability constants aimed at avoiding numerical problems and are typically set to $C_1 = C_2 = 2 * C_3 = 1$. For their part, α, β, γ are weighting coefficients typically set to 1 (Wang et al. 2004). In image comparison, because pixel proximity is crucial in image pattern recognition, Wang et al. (2004) propose to first generate sliding submatrices of dimension N entirely covering the image, then compute the SSIM index for each of them and, finally, calculate the MSSIM as the mean of the SSIM of all submatrices of dimension N. Djukic (2014) assimilates the OD-matrix to an image whose pixels would be the OD cells and explores various alternatives for generating these sliding windows in terms of proximities. Behara (2019) and Behara et al. (2020) propose a procedure to generate them based on the geographical structure of the area spanned by the transport system object of study. Ros-Roca et al. (2020) propose to use rectangular sliding windows as submatrices corresponding to either rows or columns in the OD-matrix. In any case, SSIM will capture the similarity between these distributions by considering the mean, the variance, and the structure of departing and arriving distributions, all

of which correspond to the structural property of the trip patterns described by the OD-matrix.

Furthermore, let us assume that the number of generated submatrices is N_s that a and b are, respectively, the corresponding windows of the matrices A and B to compare and that $SSIM(a,b)$ is their similarity value. Then, if MSSIM is $SSIM\,(a, b)$ averaged over N_s sliding windows, a key question arises. Particularly, whether all windows have the same weight or whether 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 or destinations are equivalent in a transport network. Therefore, a weighted MSSIM as in Wang and Simoncelli (2008) prioritizes those origins and destinations with more impact on the network. This proposed weighting average is defined as in Eq. 7.173:

$$MSSIM\,(A, B) = \frac{\sum_{i=1}^{N_s} W\,(a_i, b_i) * SSIM\,(a_i, b_i)}{\sum_{i=1}^{N_s} W\,(a_i, b_i)} \tag{7.173}$$

where a_i, b_i are, respectively, the $i - th$ windows of A and B, while the weight $w(a_i, b_i)$ is given by Eq. 7.174:

$$w(a_i, b_i) = \log\left[\left(1 + \frac{\sigma_{a_i}^2}{C_2}\right) * \left(1 + \frac{\sigma_{b_i}^2}{C_2}\right)\right] \tag{7.174}$$

The weighting factors for the sliding windows, in the case of OD-matrices, account for variances of the selected windows that, given how they are defined, represent the variance of trips from an origin to all destinations or from all origins to one destination. The use of MSSIM in addition to the conventional performance indicators has demonstrated that the usual R^2 goodness of fit between observed and simulated links flows must be carefully complemented (e.g., Djukic 2014; Behara et al. 2020; Ros-Roca et al. 2020, 2021b). Particularly, it must be complemented with a MSSIM analysis in order to check the structural quality of the estimated OD-matrix X when an acceptable historical X^H that conveys reliable structural information on the OD patterns is available.

Comparing again M_R, M_1, and M_2 in terms of MSSIM, the results are, MSSIM $(M_R, M_1) = 0.914882$ and MSSIM $(M_R, M_2) = 0.510276$, which clearly shows that M_2 is structurally different from M_R.

The relevance of this structural similarity measure (Behara et al. 2020; Behara et al. 2021) led to explicitly include it in the objective function of the mathematical model for DODME, reformulating it as follows (Eqs. 7.175–7.179):

$$\begin{aligned}\text{Min } Z(X) = \frac{1}{2} * &\left[\left(c_1 + \left(Y - \hat{Y}\right)^T * \left(Y - \hat{Y}\right)\right)\right] \\ &* \left[\left(c_2 + f\left(s, \hat{s}\right)\right)^T * \left(c_2 + f\left(s, \hat{s}\right)\right)\right]\end{aligned} \tag{7.175}$$

$$Y = A * X \tag{7.176}$$

$$s = Q * X \tag{7.177}$$

$$f\left(s, \hat{s}\right) = \frac{1 - \rho\left(s, \hat{s}\right)}{2} \tag{7.178}$$

$$\rho\left(s, \hat{s}\right) = \frac{\left(\hat{s} - \mu_{\hat{s}}\right)^T * (s - \mu_s)}{\sqrt{\left(\hat{s} - \mu_{\hat{s}}\right)^T * \left(\hat{s} - \mu_{\hat{s}}\right)} * \sqrt{(s - \mu_s)^T * (s - \mu_s)}} \tag{7.179}$$

where A is the assignment matrix, Y and $\widehat{Y}$ are, respectively, the estimated and the observed link flows at links with counting stations and s and $\hat{s}$ denote the observed and simulated flows at subpaths detected by Bluetooth (or Wi-Fi) antennas. For their part, Q is the corresponding subpath assignment matrix, while c_1 and c_2 are stabilizing constants. The algorithmic approach assumes that t A and Q are locally constant.

7.8 Concluding Remarks

The main objective of this Chapter has been to highlight the role of two key components of the engine of most traffic management and information systems. First, a *Dynamic Traffic Model*, usually a DTA or DUE, which is quite frequently supported by a *Network Loading* process based on a mesoscopic traffic simulation approach. Second, a *Dynamic Origin–Destination Matrix Estimator* (DODME) that suitably models the time-dependent mobility patterns. The main goals of these components are the estimation of the traffic state in the managed road network and its short-term prediction, accounting for impacts of external events like traffic incidents that would change the operational conditions in the network. Travel times are one of the main outputs describing these states for both managers and travelers in the network. Figure 7.10 conceptually summarizes a generic architecture of a traffic management and information system highlighting the role of these two key components and their interactions since, as it has been discussed in the chapter, the main input to a DTA or DUE is a Dynamic OD, and DOME procedures usually rely on information generated by a DTA.

This chapter has also provided an overview of the main approaches to both models, DODME and DTA/DUE, and their relationships. The role of one critical component, the dynamic assignment matrix, has been extensively discussed. This matrix describes the structure of the dynamic of the use of the links of the network by the traffic flows in the paths from origin to destinations. The possibility of exploiting the huge amount of traffic data supplied by ICT applications, which allows empirically reproducing the assignment matrix from data instead of from models in the direction

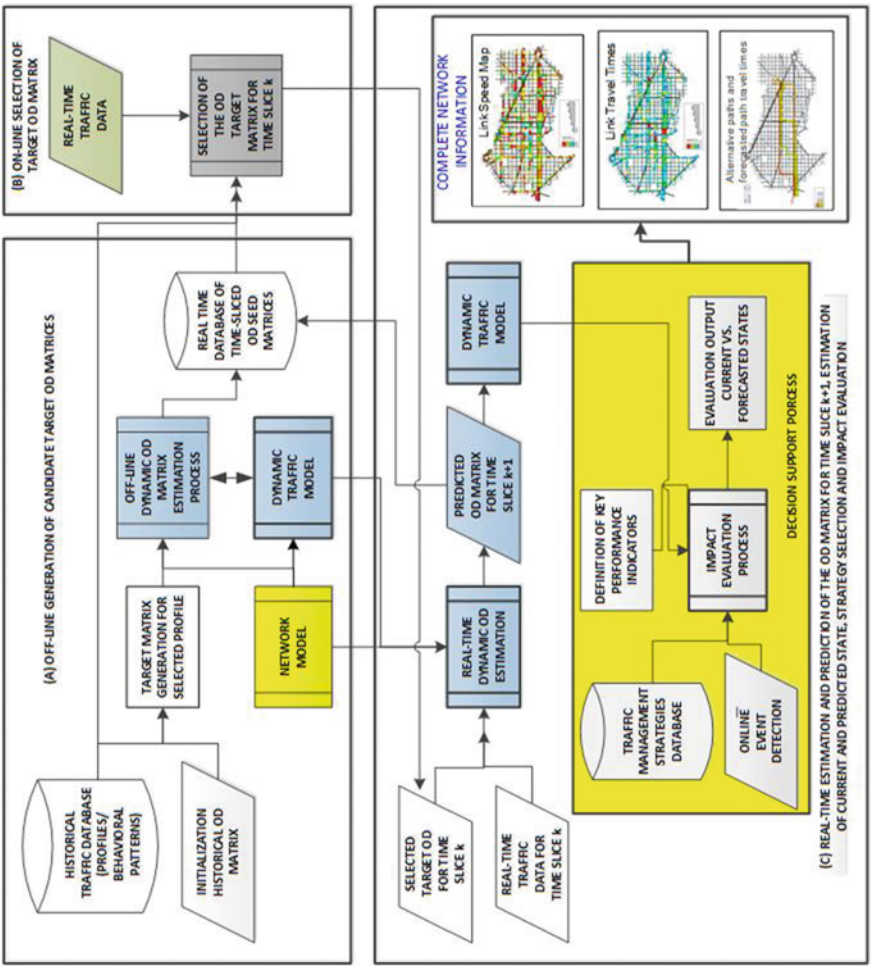

Fig. 7.10 Generic architecture of a traffic management and information system highlighting the role of the two main engine components: a DOMME and a DTA/DUE

of the data-driven modeling, has also been addressed. This trend is intellectually very appealing and, in fact, it is currently leaving the Academia realm to enter the domain of real-life applications, as it can be deduced from the last versions of some professional software platforms.

Acknowledgements The authors are very grateful to Professor Guido Gentile and Mr. Lorenzo Meschini, respectively, of SISTeMA S.R.L. and PTV Group, for supplying information OPTIMA. We also express our gratitude to Mr. Josep M. Aymamí and Dr. Emmanuel Bert (Aimsun SLU), for the information regarding Aimsun Next and Aimsun Live.

References

Aimsun SLU (2020) *Aimsun Next*. https://www.aimsun.com/es/aimsun-next/. Accessed 5 May 2021

Alexander L, Jiang S, Murga M, González MC (2015) Origin–destination trips by purpose and time of day inferred from mobile phone data. Transp Res Part C: Emerg Technol 58(B):240–250

Allström A, Barceló J, Ekström J, Grumert E, Gundlegård D, Rydergren C (2017) Traffic management for smart cities. In: Angelakis V, Tragos E, Pöhls HC, Kapovits A, Bassi A (ed) Designing, developing and facilitating smart cities. Springer, Switzerland. ISBN 978-3-319-44922-7

Altman A, Gondzio J (1999) Regularized symmetric indefinite systems in interior point methods for linear and quadratic optimization. Optim Meth Softw 11(1–4) (interior Point Methods)

Antoniou C (2004) On-line calibration for dynamic traffic assignment. PhD dissertation, Mass. Inst. Technol, Cambridge, MA

Antoniou C, Ben-Akiva ME, Koutsopoulos HN (2007). Nonlinear Kalman filtering algorithms for on-line calibration of dynamic traffic assignment models. IEEE Trans Intell Transp Syst 8(4):661–670

Antoniou C, Azevedo CL, Lu L, Pereira F, Ben-Akiva M (2015) W-SPSA in practice: approximation of weight matrices and calibration of traffic simulation models. Transp Res Part C: Emerg Technol 59:129–146

Antoniou C, Barceló J, Breen M, Bullejos M, Casas J, Cipriani E, Ciuffo B, Djukic T, Hoogendoorn S, 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. Transp Res Part C: Emerg Technol 66:79–98

Ashok K, Ben-Akiva M (1993) Dynamic origin-destination matrix estimation and prediction for real-time traffic management systems. In: Daganzo C (ed) Transportation and traffic theory. Elsevier Science Publishing Company. Inc. Proceedings of the 12th ISTTT.

Ashok K, Ben-Akiva ME (2002) Estimation and prediction of time-dependent origin-destination flows with a stochatic mapping to path flows and link flows. Transp Sci 36(2):184–198

Barceló J, Codina E, Casas J, Ferrer JL, García D (2004) Microscopic traffic simulation: a tool for the design, analysis and evaluation of intelligent transport systems. J Intell Rob Syst 41:173–203

Barceló J, Delgado M, Funes G, García D, Torday A (2007) An on-line approach based on microscopic traffic simulation to assist real time traffic management. In: 14th World congress on intelligent transport systems, 2007. Beijing

Barceló J (2010a) Models, traffic models, simulation and traffic simulation. In: Barceló J (ed) Fundamentals of traffic simulation. Springer, Switzerland. ISBN 978-1-4419-6142-6

Barceló J, Gilliéron F, Linares MP, Serch O, Montero L (2012) Exploring link covering and node covering formulations of detection layout problem. Transp Res Records: J Transp Res Board 2308:17–26

Barceló J, Montero L, Bullejos M, Serch O, Carmona C (2013) A Kalman filter approach for the estimation of time dependent OD matrices exploiting bluetooth traffic data collection. JITS J Intell Transp Syst: Technol, Plan Oper 17(2):1–19

Bassolas A, Ramasco JJ, Herranz R, Cantú-Ros OG (2019) Mobile phone records to feed activity-based travel demand models: MATSim for studying a cordon toll policy in Barcelona. Transp Res Part A: Policy Pract 121:56–74

Bazaraa MS, Sherali HD, Shetty CM (1993) Nonlinear programming. theory and algorithms. Wiley, USA

Behara K (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 KN, Bhaskar A, Chung E (2020) Geographical window based structural similarity index for origin-destination matrices comparison. J Intell Transp Syst 1–22

Behara KNS, Bhaskar A, Chung E (2021) A novel methodology to assimilate sub-path flows in bi-level OD matrix estimation process. IEEE Trans Intell Transp Syst (in press)

Bell MGH (1991) The estimation of origin-destination matrices by constrained generalized least squares. Transp Res B: Methodol 25B:115–125

Bell MGH, Iida Y (1997) Transportation network analysis. Wiley, USA

Bellei G, Gentile G, Papola N (2005) A within-day dynamic traffic assignment model for urban road networks. Transp Res Part B: Methodol 39:1–29

Ben-Akiva M, Bierlaire M, Bottom J, Koutsopoulos HN, Mishalani RG (1997) Development of a route guidance generation system for real-time application. In: Proceedings of the 8th IFAC symposium on transportation systems, 1997. Chania, Crete

Ben-Akiva M, Bierlaire M (1999) Discrete choice models and their application to short term travel decisions. In: Hall RW (ed) Handbook of transportation science. Springer, Switzerland. ISBN: 0-7923-8587-X

Ben-Akiva M, Bierlaire M, Burton D, Koutsopoulos HN, Mishalani R (2001) Network state estimation and prediction for real-time traffic management. Netw Spatial Econ 1:293–318

Ben-Akiva M, Bierlaire M, Koutsopoulos HN, Mishalani R (2002) Real-time simulation of traffic demand-supply interactions within DynaMIT. In: Gendreau M, Marcotte P (ed) Transportation and network analysis: current trends. Miscellanea in honour of Michael Florian. Kluwer Academic Publishers, Boston/Dordrecht/London

Ben-Akiva M, Koutsopoulos HN, Antoniou C, Balakrishna R (2010) Traffic simulation with DynaMIT. In: Barceló J (ed) Fundamentals of traffic simulation. Springer, Switzerland. ISBN 978-1-4419-6142-6

Bliemer MCJ, Raadsen MPH, Brederode LJN, Bell MGH, Wismans LJJ, Smith MJ (2017) Genetics of traffic assignment models for strategic transport planning. Transp Rev 37(1):56–78

Bovy P, Bekhor S, Prato C (2008) The factor of revisited path size. Transp Res Board 2076:132–140

Boyce D, Lee DH, Ran B (2001) Analytical models of the dynamic traffic assignment problem. Netw Spatial Econ 1:377–390

Bullejos M, Barceló J, Montero L (2014) A DUE based bi-level optimization approach for the estimation of time sliced OD matrices. International symposium of transport simulation, 2014. France, pp 1–19

Burghout W (2004) Hybrid microscopic-mesoscopic traffic simulation. Doctoral Thesis, Royal Institute of Technology, Stockholm, Sweden

Burghout W, Koutsopoulos H, Andréasson I (2005) Hybrid mesoscopic-microscopic traffic simulation. In: Proceedings of the 83rd TRB annual meeting, 2005. Washington, DC.

Calabrese F, Di Lorenzo G, Liu L, Ratti C (2011) Estimating origin-destination flows using mobile phone location data. IEEE Pervasive Comput 10

Calabrese F, Diao M, Di Lorenzo G, Ferreira J Jr, Ratti C (2013) Understanding individual mobility patterns from urban sensing data: a mobile phone trace example. Transp Res Part C: Emerg Technol 26:301–313

Cantelmo G, Cipriani E, Gemma A, Nigro M (2014) An adaptive bi-level gradient procedure for the estimation of dynamic traffic demand. IEEE Trans Intell Transp Syst 15(3):1348–1361
Carrese S, Cipriani E, Mannini L, Nigro M (2017) Dynamic demand estimation and prediction for traffic urban networks adopting new data sources. Transp Res Part C: Emerg Technol 81:83–98
Carey M, Ge YE (2012) Comparison of methods for path flow reassignment for dynamic user equilibrium. Netw Spatial Econ 12:337–376
Casas J, Ferrer J, García D, Perarnau J, Torday A (2010) Traffic simulation with Aimsun. In: Barceló J (ed) Fundamentals of traffic simulation. Springer, Switzerland. ISBN 978-1-4419-6142-6
Cascetta E (1984) Estimation of origin-destination matrices from traffic counts and survey data: a generalised least squares estimator. Transp Res Part B: Methodol 18(B):289–299
Cascetta E, Nuzzolo A, Russo F, Vitetta A (1996) A modified logit route choice model overcoming path overlapping problems. In: Proceedings of the 13th international symposium on the theory of road traffic flow, 1996. France
Cascetta E (2001) Transportation systems engineering theory and methods. Springer, Switzerland
Cascetta E, Papola A, Marzano V, Simonelli F, Vitiello I (2013) Quasi-dynamic estimation of OD flows from traffic counts: formulation, statistical validation and performance analysis on real data. Transp Res Part B: Methodol 55:171–187
Castillo E, Conejo AJ, Menéndez JM, Jiménez P (2008) The observability problem in traffic network models. Comput-Aided Civil Infrastruct Eng 23:208–222
Chabini I (1998) Discrete dynamic shortest path problems in transportation applications: complexity and algorithms with optimal run time. Transp Res Records 1997
Chiu YC, Bottom J, Mahut M, Paz A, Balakrishna R, Waller T, Hicks J (2011) Dynamic traffic assignment: a primer. Transp Res E-Circular (E-C153)
Cipriani E, Florian M, Mahut M, Nigro M (2011) A gradient approximation approach for adjusting temporal origin-destination matrices. Transp Res Part C: Emerg Technol 19(2):270–282
Codina E, Barceló J (2004) Adjustment of O-D matrices from observed volumes: an algorithmic approach based on conjugate gradients. Eur J Oper Res 155:535–557
Codina E, Montero L (2006) Approximation of the steepest descent direction for the O-D matrix adjustment problem. Ann Oper Res 114:329–362
Çolak S, Lima A, González MC (2015) Understanding congested travel in urban areas. Nat Commun 7:10793
Daganzo CF (1994) The cell-transmission model: a simple dynamic representation of highway traffic. Transp Res Part B: Methodol 28(4):269–287
Daganzo CF (1995) The cell transmission model part II: network traffic. Transp Res Part b: Methodol 29:79–93
Daganzo CF (1995) A finite difference approximation of the kinematic wave model of traffic flow. Transp Res Part B: Methodol 29(4):261–276
Del Castillo JM, Benitez FG (1995) On the functional form of the speed-density relationship I: general theory. Transp Res Part B: Methodol 29(5):373–389
Djukic T, van Lint JWC, Hoogendoorn SP (2012) Application of principal component analysis to predict dynamic origin-destination matrices. Transp Res Record: J Transp Res Board 2283(1):81–89
Djukic T (2014) Dynamic OD demand estimation and prediction for dynamic traffic management. PhD Thesis, TU Delft
Djukic T, Breen M, Masip D, Perarnau J, Budin J, Casas J (2017) Marginal effects evaluation with respect to changes in OD demand for dynamic OD demand estimation. In: Proceedings of the international conference on intelligent transport systems in theory and practice. TUM'17, 2017, Munich
Djukic T, Masip D, Breen M, Perarnau J, Casas J (2018) Heuristic-based framework for dynamic OD demand estimation in the congested networks. Transportation research board 97th annual meeting transportation research board, 18, 03283

Djukic T, Masip D, Breen M, Casas J (2019) Efficient metamodel framework for nonlinear OD matrix estimation problem. Transportation research board 98th annual meeting transportation research board, 19, 05188

Ehlert A, Bell MGH, Grosso S (2006) The optimisation of traffic count locations in road networks. Transp Res Part B: Methodol 40:460–479

Eisenman SM, List GF (2004). Using probe data to estimate OD matrices. In: Proceedings of the 7th international IEEE conference on intelligent transportation systems (ITSC '04), October 2004. Washington, DC, USA, pp 291–296

Fei X, Eisenman SM, Mahmassani H (2007) Sensor coverage and location for real-time traffic prediction in large-scale networks. In: 86th annual meeting of the transportation research board, January 2007. Washington, DC, USA

Filkov V, Skiena S (2004) Integrating microarray data by consensus clustering. Int J Artif Intell Tools 13:863–880

Florian M, Chen Y (1995) A coordinate descent method for the bi-level OD matrix adjustment problem. Int Trans Oper Res 2(2):165–175

Florian M, Hearn D (1995) Network equilibrium models and algorithms. In: Ball MO et al (ed) Handbooks in operations research and management science, 8. Elsevier Science B.V., The Netherlands

Florian M, Mahut M, Tremblay N (2001) A hybrid optimization-mesoscopic simulation dynamic traffic assignment model. In: Proceedings of the 2001 IEEE intelligent transport systems conference, 2001. Oakland, pp 118–123

Florian M, Mahut M, Tremblay N (2002) Application of a simulation-based dynamic traffic assignment model. In: Kitamura R, Kuwahara M (eds) International symposium on transport simulation, 2002, Yokohama (also in: Simulation approaches in transportation analysis, 2005. Kluwer, US

Florian M, Mahut M, Tremblay N (2008) Application of a simulation-based dynamic traffic assignment model. Eur J Oper Res 189(3):1381–1392

Frederix R, Viti F, Corthout R, Tampère C (2011) New gradient approximation method for dynamic origin-destination matrix estimation on congested networks. Transp Res Record: J Transp Res Board 2263(1):19–25

Frederix R, Viti F, Tampère C (2013) Dynamic origin-destination estimation in congested networks: theoretical findings and implications in practice. Transportmetrica a: Transport Science 9(6):494–513

Friesz TL, Bernstein D, Smith TE, Tobin RL, Wie BW (1993) A variational inequality formulation of the dynamic network user equilibrium problem. Oper Res 41(1):179–191

Gelb A (1974) Applied optimal estimation. MIT Press, Cambridge, MA

Gentile G, Meschini L, Papola N (2007) Spillback congestion in dynamic traffic assignment: a macroscopic flow model with time-varying bottlenecks. Transp Res Part B: Methodol 41:1114–1138

Gentile G (2010) The general link transmission model for dynamic network loading and a comparison with the DUE algorithm. In: Immers LGH, Tampere CMJ, Viti F (eds) New developments in transport planning: advances in dynamic traffic assignment. Transport Economics, Management and Policy Series, Edward Elgar Publishing, MA, USA

Gentile G (2015) Using the general link transmission model in a dynamic traffic assignment to simulate congestion on urban networks. Transp Res Procedia 5:66–81

González MC, Hidalgo A, Barabasi A-L (2008) Understanding human mobility patterns. Nature 453(7196):779–782

Greenshields BD (1934) A study of traffic capacity. In: Proceedings of the fourteenth annual meeting of the highway research board, held at Washington, D.C. December 6–7, 1934, Part I, 14, 448–477

Gundlegård D, Rydergren C, Barcelo J, Dokoohaki N, Görnerup O, Hess A (2015) Travel demand analysis with differentially private releases. D4D challenge Senegal 2014, Netmob 2015, November 2015, MIT, Boston

Han K, Eve G, Friesz TL (2019) Computing dynamic user equilibria on large-scale networks with software implementation. Netw Spatial Econ 19:869–902

Hart PE, Nilsson NJ, Raphael B (1968) A formal basis for the heuristic determination of minimum cost paths. IEEE Trans Syst Sci Cybern 4:100–107

Hegyi A, Bellemans T, De Schutter B (2009) Freeway traffic management and control. In: Meyers RA (ed) Encyclopedia of complexity and systems science. Springer, New York. ISBN 978-0-38730440-3

Hollander Y, Liu R (2008) The principles of calibrating traffic microsimulation models. Transportation 35:347–362

Hu S, Madanat SM, Krogmeier J, Peeta S (2001) Estimation of dynamic assignment matrices and OD demands using adaptative Kalman filtering. Intell Transp Syst J 6:281–300

Iqbal MS, Choudhury CF, Wang P, González MC (2014) Development of origin–destination matrices using mobile phone call data. Transp Res Part C: Emerg Technol 40:63–74

Janmyr J, Wadell D (2018) Analysis of vehicle route choice during incidents. MSc Thesis, University of Linkoping, Department of Science and Technology

Janson BN (1991) Dynamic traffic assignment for urban road networks. Transp Res Part B: Methodol 25(2):143–161

Jayakrisham R, Mahmassani HS, Yu TY (1994) An evaluation tool for advanced traffic information and management systems in urban networks. Transp Res Part C: Emerg Technol 2C(3):129–147

Jianga S, Yanga Y, Gupta S, Veneziano D, Athavale S, González MC (2016) The TimeGeo modeling framework for urban mobility without travel surveys. Proc Natl Acad Sci USA 113:37

Jeihani M (2007) A review of dynamic traffic assignment computer packages. J Transp Res Forum 46:35–46

Jolliffe IT (2002) Principal component analysis. Springer, Switzerland

Kalman RE (1960) A new approach to linear filtering and prediction problems. J Basic Eng (ASME) 82D:35–45

Kostic B, Gentile G, Antoniou C (2017a) Techniques for improving the effectiveness of the SPSA algorithm in dynamic demand calibration. In: 5th IEEE international conference on models and technologies for intelligent transportation systems, MT-ITS 2017. Napoli, Italy

Kostic B, Annunziata A, Gentile G, Meschini L (2017b) A sequential approach to time-dependent demand calibration: application, validation and practical implications for large-scale networks. In: 5th IEEE international conference on models and technologies for intelligent transportation systems, MT-ITS 2017. Napoli, Italy

Krishnakumari P, van Lint H, Djukic T, Cats O (2019) A data driven method for OD matrix estimation. Transp Res Part C: Emerg Technol 113:38–56

Kubicka M, Mounier H, Niculescu SI, Cela A (2018) Comparative study and application-oriented classification of vehicular map-matching methods. IEEE Intell Transp Syst Mag 10(2):150–166

Larsson T, Lundgren JT, Peterson A (2010) Allocation of link flow detectors for origin-destination matrix estimation: a comparative study. Comput-Aided Civil Infrastruct Eng 25:116–131

Leonard DP, Gower P, Taylor N (1989) CONTRAM. Structure of the model, transport and road research laboratory. Research Report 178, Department of Transport, Crowthorne

Lighthill M, Whitham G (1955) On kinematic waves. II. A theory of traffic flow on long crowded roads. In: Proceedings of the royal society of London. Series A, mathematical and physical sciences, vol 229, no 1178, pp 317–345

Lin P, Chang G (2007) A generalized model and solution algorithm for estimation of the dynamic freeway origin-destination matrix. Transp Res Part b: Methodol 41:554–572

Lo HK, Szeto WY (2002) A cell-based variational inequality formulation of the dynamic user optimal assignment problem. Transp Res Part b: Methodol 36:421–443

López C, Krishnakumari P, Leclercq L, Chiabaut N, van Lint H (2017) Spatio-temporal partitioning of the transportation network using travel time data. Transp Res Record: J Transp Res Board 2623(1):98–107

López 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. Sci Rep 7:14029

Lu L, Xu Y, Antoniou C, Ben-Akiva M (2015) An enhanced SPSA algorithm for the calibration of dynamic traffic assignment models. Transp Res Part C: Emerg Technol 51:149–166

Luenberger DG, Ye Y (2008) Linear and nonlinear programming. Springer, Switzerland
Lundgren JT, Peterson A (2008) A heuristic for the bilevel origin–destination-matrix estimation problem. Transp Res Part b: Methodol 42:339–354
Mahmassani HS, Hu TY, Peeta S, Ziliaskopoulos A (1994) Development and testing of dynamic traffic assignment and simulation procedures for ATIS/ATMS applications. Technical Report DTFH61–90-R00074-FG, Center for Transportation research, The University of Texas at Austin
Mahmassani H (2001) Dynamic network traffic assignment and simulation methodology for advanced system management applications. Netw Spatial Econ 1:267–292
Mahmassani H, Hong Z, Xu X, Mittal A, Yelchuru B, Kamalanathsharma R (2017) Analysis, modeling, and simulation (AMS) testbed development and evaluation to support dynamic mobility applications (DMA) and active transportation and demand management (ATDM) programs evaluation report for the Chicago testbed. Final Report—April 2017, FHWA-JPO-16–387
Mahut M (1999) Behavioural car following models. Report CRT-99–31. Centre for Research on Transportation. University of Montreal. Montreal, Canada
Mahut M (2001) Discrete flow model for dynamic network loading. PhD Thesis, Département d'informatique et de recherhe opérationelle, Université de Montréal. Published by the Center for Research on Transportation (CRT), University of Montreal
Mahut M, Florian M, Tremblay N (2003a) Space-time queues and dynamic traffic assignment: a model, algorithm and applications. Transportation research board, 82nd annual meeting, 2002. Washington DC, USA
Mahut M, Florian M, Tremblay N (2003b) Traffic simulation and dynamic assignment for off-line applications. In: 10th world congress on intelligent transportation systems, 2003. Madrid, Spain
Mahut M, Florian M, Tremblay N, Campbell M, Patman D, McDaniel ZK (2004) Calibration and application of a simulation based dynamic traffic assignment model. Transp Res Record: J Transp Res Board 1876:101–111
Mahut M, Florian M (2010) Traffic simulation with dynameq. In: Barceló J (ed) Fundamentals of traffic simulation. Springer, Switzerland. ISBN 978-1-4419-6142-6
Marchal F, Hackney JK, Axhausen KW (2004) Efficient map-matching of large GPS data sets—tests on a speed monitoring experiment in Zurich. Arbeitsbericht Verkehrs und Raumplanung. Technical report, UNAM, p 244
May AD, Keller HEM (1967) Non-integer car-following models. Highway Res Rec 199:19–32
Meschini L (2017) Modern traffic control centres and traffic management systems. In: Fusco G (ed) Intelligent transport systems (ITS): past, present and future directions. NOVA Science Publishers. ISBN 978-1-53611-815-5
Millard-Ball A, Hampshire RC, Weinberger RR (2019) Map-matching poor-quality GPS data in urban environments: the pgMapMatch package. Transp Plan Technol 42(6):539–553
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. In: IET intelligent transport systems special issue: the scientific seminar of the Italian association of transport academicians 2019 (SIDT 2019), vol 14, pp 1704–1711
Mo B, Li R, Dai J (2020) Estimating dynamic origin–destination demand: a hybrid framework using license plate recognition data. Comput Aided Civil Infrastruct Eng 35(7):1–19
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. Transp Res Procedia 37:417–424
Morales JL, Nocedal J (2011) Remark on algorithm 778: L-BFGS-B: fortran subroutines for large-scale bound constrained optimization. ACM Trans Math Softw 38(1):7
Nanthawichit C, Nakatsuji T, Suzuki H (2003) Application of probe vehicle data for real-time traffic state estimation and short-term travel time prediction in a freeway. Transp Res Record: J Transp Res Board 1855(1):49–59
Nassir N, Ziebarth J, Sall E, Zorn L (2014) Choice set generation algorithm suitable for measuring route choice accessibility. Transp Res Record 2430(1):170–171
Newell GF (2002) A simplified car-following theory: a lower order model. Transp Res Part B: Methodol 36B(3):195–205

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. J Adv Transp 2018:1–18

OpenLR (2020. OpenLR White Paper. Version 1.5, revision 2. https://ww.openlr-association.com/fileadmin/user_upload/openlr-whitepaper_v1.5.pdf

Ortúzar JD, Willumsen LG (2011) Modelling transport. Wiley, USA

Osorio C, Linsen C (2015) A computationally efficient simulation-based optimization algorithm for large-scale urban transportation problems. Transp Sci 49(3):623–636

Peeta S, Mahmassani HS (1995) System optimal and user equilibrium time-dependent traffic assignment in congested networks. Ann Oper Res 60:81–113

Peeta S, Ziliaskopoulos AK (2001) Foundations of dynamic traffic assignment: the past, the present and the future. Netw Spatial Econ 1:233–265

Pereira FC, Costa H, Pereira NM (2009) An off-line map-matching algorithm for incomplete map databases. Eur Transp Res Rev 1:107–124

PTV AG Visum (2020) PTV Visum 2020—user's manual. PTV Group, Karlsruhe, Germany

Quddus MA, Ochieng WY, Noland RB (2007) Current map-matching algorithms for transport applications: state-of-the art and future research directions. Transp Res Part C: Emerg Technol 15:312–328

Rahmani M, Koutsopoulos HN (2013) Path inference from sparse floating car data for urban networks. Transp Res Part C: Emerg Technol 30:41–54

Ran B, Boyce D (1996) Modeling dynamic transportation networks. Springer, Switzerland

Richards PI (1956) Shockwaves on the highway. Oper Res 4(1):42–51

Ros-Roca X, Montero L, Barceló J (2017) Notes on using simulation-optimization techniques in traffic simulation. Transp Res Procedia 27:881–888

Ros-Roca X, Montero L, Schneck A, Barceló J (2018) Investigating the performance of SPSA in simulation-optimization approaches to transportation problems. Transp Res Procedia 34:83–90

Ros-Roca X, Montero L, Barceló J (2020) Investigating the quality of Spiess-like and SPSA approaches for dynamic OD matrix estimation. Transportmetrica 17(3):235–257

Ros-Roca X, Montero L, Barceló J, Nökel K (2021a) Dynamic origin-destination matrix estimation with ICT traffic measurements using SPSA. Accepted for presentation at *MTITS2021*, to appear in Scopus-indexed IEEE Xplore Digital Library conference proceedings (conference number 49943)

Ros-Roca X, Montero L, Barceló J, Nökel K, Gentil G (2021b) A practical approach to assignment-free dynamic origin-destination matrix estimation problem. Accepted for publication in Transportation Research C: Emerging Technologies

Sadegh P, Spall JC (1998) Optimal random perturbations for stochastic approximation using a simultaneous perturbation gradient approximation. IEEE Trans Autom Control 43(10):1480–1484

Sbayti H, Lu C, Mahmassani HS (2007) Efficient implementations of the method of successive averages in simulation-based DTA models for large-scale network applications. TRB 2007 Annual Meeting, 2007. Washington DC, USA

Schuessler N, Axhausen KW (2009) Map-matching of GPS traces on high-resolution navigation networks using the multiple hypothesis technique (MHT). Working paper 568 Institute for Transport Planning and Systems, Swiss Federal Institute of Technology Zürich

Smith MJ (1993) A new dynamic traffic model and the existence and calculation of dynamic user equilibria on congested capacity-constrained road networks. Transp Res Part B: Methodol 27:49–63

Smith MJ, Wisten MB (1995) A continuous day-to-day traffic assignment model and the existence of a continuous dynamic user equilibrium. Ann Oper Res 60(1):59–79

Spall JC (1992) Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. IEEE Trans Autom Control 37(3):332–341

Spall JC (1998) An overview of the simultaneous perturbation method for efficient optimization. Johns Hopkins APL Tech Digest (applied Physics Laboratory) 19(4):482–492

Spall JC (2003) Introduction to stochastic search and optimization: estimation, simulation, and control. Wiley-Interscience
Spiess H (1990) A gradient approach for the OD matrix adjustment problem. Publication No. 693, Centre de Recherche sur les Transports, Université de Montréal
Szeto WY, Wong SC (2012) Dynamic traffic assignment: model classifications and recent advances in travel choice principles. Open Eng 2(1):1–18
Szeto WY, Lo HK (2005) Dynamic traffic assignment: review and future. Inf Technol 5:85–100
Szeto WY, Lo HK (2004) A cell-based simultaneous route and departure time choice model with elastic demand. Transp Res Part B: Methodol 38:593–612
Tympakianaki A, Koutsopoulos HN, Jenelius E (2015) C-SPSA: cluster-wise simultaneous perturbation stochastic approximation algorithm and its application to dynamic origin-destination matrix estimation. Transp Res Part C: Emerg Technol 55:231–245
Toledo T, Kolechkina T (2013) Estimation of dynamic origin-destination matrices using linear assignment matrix approximations. IEEE Trans Intell Transp Syst 14(2):618–626
Tong CO, Wong SC (2000) A predictive dynamic traffic assignment model in congested capacity-constrained road networks. Transp Res Part b: Methodol 34:625–644
van Aerde M, Hellinga B, Yu L, Rakha H (1993) Vehicle probes as real-time ATMS sources of dynamic OD and travel time data. Queen's University, Department of Civil Engineering
van Zuylen HJ, Willumsen LG (1980) The most likely trip matrix estimated from traffic counts. Transp Res Part B: Methodol 14:281–293
Varia HR, Dhingra SL (2004) Dynamic user equilibrium traffic assignment on congested multidestination network. J Transp Eng 130(2):211–221
Wang IJ, Spall JC (1999) A constrained simultaneous perturbation stochastic approximation algorithm based on penalty functions. In: IEEE Proceedings of the 1999 American control conference (Cat.No.99CH36251), 1999. USA
Wang Z, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: from error visibility to structural similarity. IEEE Trans Image Process 13(4):600–612
Wang Z, Simoncelli EP (2008) Maximum differentiation (MAD) competition: a methodology for comparing computational model of perceptual quantities. J vis 8(12):1–13
Wardrop JG (1952) Some theoretical aspects of road traffic research. Proc Inst Civil Engineers II:325–378
Wu JH (1991) A study of monotone variational inequalities and their application to network equilibrium problems. Ph. D. Thesis, Centre de Recherche sur les Transports, Université de Montréal, Publication #801
Wu JH, Chen Y, Florian M (1998a) The continuous dynamic network loading problem: a mathematical formulation and solution method. Transp Res Part B: Methodol 32(3):173–187
Wu JH, Florian M, Xu YW, Rubio-Ardanaz JM (1998b) A projection algorithm for the dynamic network equilibrium problem. In: Yang Z, Wang KCP, Baohua M (eds) Traffic and transportation studies, ASCE proceedings of the ICTTS'98
Xu YW, Wu JH, Florian M, Marcotte P, Zhu DL (1999) Advances in the continuous dynamic network loading problem. Transp Sci 33(4):341–353
Yang H (1995) Heuristic algorithms for the bi-level origin-destination matrix estimation problem. Transp Res Part B: Methodol 29:231–242
Yang H, Zhou J (1998) Optimal traffic counting locations for origin-destination matrix estimation. Transp Res Part B: Methodol 32B(2):109–126
Yang X, Lu Y, Hao W (2017) Origin-destination estimation using probe vehicle trajectory and link counts. J Adv Transp 2017:4341532
Zhang Y, Qin X, Dong S, Ran B (2010) Daily O-D matrix estimation using cellular probe data, Paper 10–2472. In: 89th TRB annual meeting, 2010. Washington DC