Data driven origin–destination matrix estimation on large networks
A joint origin–destination-path-choice formulation

Cao, Yumin; van Lint, Hans; Krishnakumari, Panchamy; Bliemer, Michiel

DOI
10.1016/j.trc.2024.104850

Publication date
2024

Document Version
Final published version

Published in
Transportation Research Part C: Emerging Technologies

Transportation Research Part C 168 (2024) 104850

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Data driven origin–destination matrix estimation on large networks—A joint origin–destination-path-choice formulation ☆

Yumin Cao [a,b,*], Hans van Lint [a], Panchamy Krishnakumari [a], Michiel Bliemer [c]

[a] Delft University of Technology, The Netherlands
[b] Tongji University, China
[c] University of Sydney, Australia

ARTICLE INFO

Keywords:
Dynamic OD matrix estimation
Gravity model
Joint origin–destination-path choice
Principal component analysis

ABSTRACT

This paper presents a novel approach to data-driven time-dependent origin–destination (OD) estimation using a joint origin–destination-path choice formulation, inspired by the well-known equivalence of doubly constraint gravity models and multinomial logit models for joint O–D choice. This new formulation provides a theoretical basis and generalizes an earlier contribution. Although including path choice increases the dimensionality of the problem, it also dramatically improves the quality of the data one can *directly* use to solve it (e.g. measured path travel times versus coarse centroid-to-centroid travel times); and opens up possibilities to combine different assimilation techniques in a single framework: (1) fast shortest path set computation using static (e.g. road type) and dynamic (speed, travel time) link properties; (2) predicting a "prior OD matrix" using the resulting path-shares and (estimated or measured) production and attraction totals; and (3) scaling/constraining this prior using link flows (informative of demand). If the resulting system of equations has insufficient rank, we use principal component analysis to reduce the dimensionality, solve this reduced problem, and transform that solution back to a full OD matrix. Comprehensive tests and sensitivity analysis on 7 networks with different sizes and characteristics give an empirical underpinning of the extended equivalence principle; demonstrate good accuracy and reliability of the OD estimation method overall; and suggest that the method is robust with respect to major assumptions and contributing factors.

1. Introduction

In this paper we discuss a new method for time dependent OD matrix estimation for congested road networks. The estimation of such time dependent origin–destination matrices is important for many applications over the entire transportation domain, from operations, control and management; to planning and policy assessment. The key challenge in estimating OD matrices is that, particularly for large congested networks, the problem is severely underdetermined, a fact that was recognized in the early days of the OD estimation literature (e.g. Van Zuylen and Willumsen (1980), Cascetta (1984), Bell (1991)) and is emphasized in virtually all contemporary OD estimation research still. This underdeterminacy relates to the fact that the number of unknown OD flows, i.e. the size of the OD matrix, grows quadratically with the number of OD zones (a subset of all network nodes); whereas the number of independent equations from which the unknown OD matrix $\mathbf{X}$ can be inferred (using whatever data $\tilde{\mathbf{y}}$ available) typically grows no more than linearly with network size (i.e. the number of links, nodes, zones). For small networks the resulting system of

☆ This article belongs to the Virtual Special Issue on "ISTTT25".

* Corresponding author at: Tongji University, China.
E-mail address: cao97@tongji.edu.cn (Y. Cao).

https://doi.org/10.1016/j.trc.2024.104850
Received 15 January 2024; Received in revised form 28 August 2024; Accepted 10 September 2024
Available online 23 September 2024

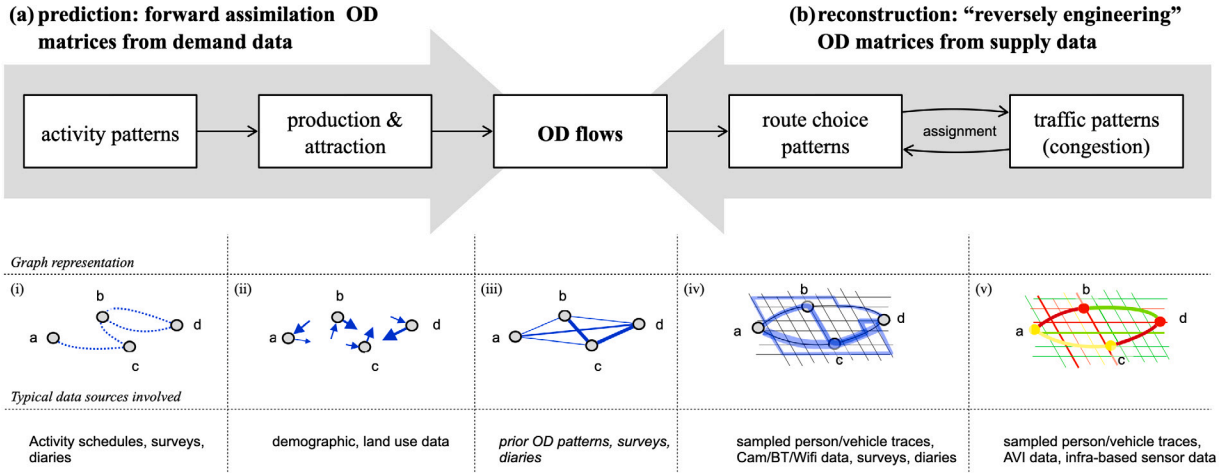


Fig. 1. OD prediction and reconstruction: (a) forward and (b) reverse data assimilation problems.

equations may still be solvable, for larger networks the OD matrix may become unobservable without additional constraints and/or assumptions (Krishnakumari et al., 2020) This is particularly true in congested networks, in which case link flows downstream of saturated bottlenecks are no longer proportional to demand but to queue discharge flow. This exacerbates how fast OD matrices become underdetermined with increasing network size. The more a network is congested, the worse the observability problem becomes.

OD matrix estimation is a data assimilation process, in which data that are directly or indirectly related to OD flows are combined with transportation models — or other models that codify relations between OD flows and observable quantities — to infer the unknown OD matrix, either in the sense of (a) *predicting* an OD matrix in the near or distant future; or (b) *reconstructing* (or *estimating*) a historical or prevailing OD matrix (Kalman, 1960; Hazelton, 2001). Fig. 1 schematically outlines both assimilation problems and sketches the semantics of the variables involved (in the form of a graph representation) and the type of data involved. In the "forward assimilation" or prediction problem, OD matrices are inferred from demand data and models that describe the relationship between activities; land use; and the resulting OD flows. These demand models range from macroscopic trip-production models (Anas, 1983; Scheffer et al., 2017; Cantelmo et al., 2015) to detailed disaggregate activity based (AB) models (e.g. Kitamura et al. (2000), Bhat and Zhao (2002), Arentze and Timmermans (2009) and Arentze et al. (2011)). As illustrated in Fig. 1 (i–iii), activity patterns are walks on a graph of activity locations/zones; production and attraction are properties of these locations/zones; and OD flows are properties of the connections between them. The data used are typically demographics and land use data in combination with e.g. trip length distributions, and survey and diary data. As in any prediction model, how well demand models perform, relates to the number of assumptions and the evidence for them. More complexity (degrees of freedom) means more explanatory power but also more ways to get it wrong.

The second assimilation problem, the OD matrix reconstruction or simply estimation problem,[1] involves "reversely engineering" the most likely OD matrix that has resulted from observed traffic patterns, typically link counts and speeds (on a graph that represents the physical infrastructure). This is also recognized as the network tomography problem, a term coined by Vardi (1996), within the broader context of network studies. A key challenge is that (largely) *unobserved* route choice patterns (Fig. 1-iv), form the causal link between the OD matrix and *observed* traffic patterns (Fig. 1-v). There is a long record of OD estimation methods that fall in this "reverse engineering" category (e.g. Van Zuylen and Willumsen (1980), Cascetta (1984), Cremer and Keller (1987), Bell (1991), Yang et al. (1992), Tebaldi and West (1998), Ashok and Ben-Akiva (2000), Zhou and Mahmassani (2007), Castillo et al. (2012), Cascetta et al. (2013), Cantelmo et al. (2014), Cipriani et al. (2014), Hazelton (2015) and Antoniou et al. (2016) to name a few in chronological order). These OD matrix estimation approaches are typically formulated as optimization problems (Cascetta et al., 2013; Lundgren and Peterson, 2008; Cascetta and Postorino, 2001); or as sequential (recursive) estimation problems (Okutani and Stephanedes, 1984; Van Der Zijpp, 1997; Zhou and Mahmassani, 2007; Djukic et al., 2012a; Carrese et al., 2017), in which an objective function is minimized that typically has two components. The first (f_1) expresses the distance of the estimated matrix to a prior OD matrix $\tilde{\mathbf{X}}$, whereas the second (f_2) expresses the distance between the traffic data observed, and the data predicted by the OD matrix (Cascetta and Marquis, 1993)—which results from assigning the OD matrix to routes over the network:

$$\hat{\mathbf{X}} = \arg\min_{\mathbf{X}} f_1(\mathbf{X}, \tilde{\mathbf{X}}) + f_2(\mathbf{y}(\mathbf{X}), \tilde{\mathbf{y}}), \tag{1}$$

[1] We will use the terms (OD matrix) estimation and reconstruction as synonyms in this paper.

in which

$$\mathbf{y}(\mathbf{X}) = \mathrm{A}(\mathbf{X}, \boldsymbol{\theta}) \tag{2}$$

represents an assignment (simulation) model with $\boldsymbol{\theta}$ depicting all assumptions (parameters, inputs) related to route choice and driving behavior. The distance function $f_1(\mathbf{X}, \tilde{\mathbf{X}})$ acts as a regularization term that penalizes dissimilarity of the estimated OD matrix with this prior OD matrix.

As evident from Fig. 1, these two assimilation problems typically apply to different contexts in the transportation domain. Forward assimilation techniques are used in short- and long term OD demand prediction, e.g. in ex ante analyses of network interventions and/or transport (policy) measures using (strategic) simulation models. Reverse assimilation techniques are applied in ex post analyses, e.g. in estimating prevailing or historical OD patterns from whatever direct and indirect evidence (data, information) available. Clearly, OD matrices inferred using such estimation methods are in turn critically important as input to simulation models in both short- and long term (traffic) prediction tasks. Conversely, as we show in Krishnakumari et al. (2020) and in this paper, prediction models offer substantial added value in OD estimation methods since they assimilate different data (e.g. land use and demographic data, surveys) then typically used in estimation methods (e.g. traffic counts, travel times). Combining different assimilation techniques and multiple data sources allows one to maximize the evidence for the final reconstructed OD matrix.

There are various ways to fuse such different pieces of (assimilated) evidence. The soft constraint (f_1) in Eq. (1) is one approach, which forces the OD matrix estimate $\mathbf{X}$ to stay close to the OD matrix inferred by whatever means from other evidence (e.g. predicted by a gravity model or reconstructed using surveys Bierlaire and Toint, 1995; travel diaries Scheffer et al., 2017; vehicle identification systems Kim et al., 2014; Zhou and Mahmassani, 2006 GSM or GPS traces Ge and Fukuda, 2016; Alexander et al., 2015; Gadzinski, 2018). It can also be approached statistically, with Bayesian inference being a common choice for integrating evidence with a prior distribution to derive a likely (posterior) OD/path estimate (Maher, 1983; Spiess, 1987; Tebaldi and West, 1998; Hazelton, 2008, 2010). Favoring solutions similar to an OD matrix predicted or estimated from alternative evidence is an intuitive, but nonetheless debatable, assumption. First, it is not self-evident how one should weigh the prior OD matrix and what — under different conditions — the effects of this are in terms of estimation accuracy. This clearly depends on how reliable this prior is and how much evidence there is that the prevailing OD matrix indeed "looks like" the prior.

Second, similarity is not a clear-cut criterion and a proper choice of the distance function f_1 is crucial. Standard distance metrics (L2 norm, RMSE) may not necessarily steer the estimation in a direction that favors similarity in spatio-temporal structure, for which other metrics such as the structure similarity index (SSI) (Djukic et al., 2013) or Levenshtein distance (Behara et al., 2020) may be more appropriate. These however, may increase the non-linearity of the solution space spun by Eq. (1) and increase the computational effort needed to find plausible minima in it.

The alternative approach is to encode additional evidence as *hard* constraints to narrow down the solution space of the OD estimation problem, in the same way as non-negativity of flows, and consistency of path- and link flows are imposed as constraints to reduce the solution space. This essentially is the approach taken in Krishnakumari et al. (2020). In that work we put forward a data driven OD estimation method that relaxes the idea that OD path flows should be consistent (should equilibrate) with (perceived) path travel times. Instead, since we use actual observations, travel times along paths between the same OD pairs may be significantly different, and the idea of a Nash equilibrium can be replaced with a heuristic that — in some behaviorally plausible way consistent with observations — distributes the OD flows over these paths. To this end, Krishnakumari et al. (2020) combine forward and reverse assimilation techniques in an attempt to reformulate the problem as a (large) linear system of equations that is (directly) solvable. This approach has three key ingredients: (1) a behavioral heuristic in the form of a (simple) path-size logit model, with path choice inversely proportional to *observed* path travel times; (2) a constant shortest path choice set size N^* (we used 5 for all OD pairs); and (3) a path flow equation in which link counts are used to constrain the path flow totals. In this conservation equation only those link counts are used which are informative of demand, which implies that counts upstream of active bottlenecks are excluded. For small networks these data and assumptions indeed result in a directly solvable system of equations. For large networks, however, the problem (again) becomes under-determined, which in Krishnakumari et al. (2020) is solved by reducing the dimensionality of the solution space through principal component analysis (PCA), which exploits the fact that temporal patterns of production and attraction are typically similar across the network. This PCA approach allows one to reduce the solution-space-dimensionality as much as needed, given the available data. Although the results show promise in recovering the ground truth OD matrix in a toy network and a large network, a theoretical foundation for the assumptions in the method lacks.

In the current paper we provide this theoretical foundation and generalize the method. We show that, in line with the well-known equivalence of doubly-constrained gravity models and multinomial logit models for origin–destination choice (Anas, 1983), the model proposed in Krishnakumari et al. (2020) is a special case of a combined origin–destination-path choice estimator, with specific parameter choices and a fixed-size choice set for all OD pairs. By relaxing these assumptions we derive a generic and more powerful formulation that allows one to assimilate any data that provide evidence for the OD matrix (and underlying path-flows) we seek to estimate. We illustrate through examples with both synthetic (i.e. a ground-truth OD matrix which we perturb) and validated simulation data the sensitivity of the method to different assumptions, choices of parameters and data availability. Our main contributions are

1. New joint O–D-path choice formulation for dynamic OD estimation in congested networks
2. New solution methodology in which multiple sources of heterogeneous data (trip production, path travel times, link flows, network properties) are fused

3. Extensive sensitivity analysis on multiple networks to validate major assumptions and test various components

The paper is organized as follows. Section 2 revisits and expands upon the equivalence established by Anas (1983) to formulate the problem as a joint origin–destination-path estimator, and a generic solution method is proposed. The test scenarios are described in Section 3 and findings are presented in Section 4. Finally, Section 5 provides concluding remarks and a discussion on potential future research directions.

2. Methodology

In this section we recall the equivalence of doubly constraint gravity models and multi-nominal logit models (MNL) for combined OD choice. We then show that by adding the (dynamic) path-choice dimension, we obtain an OD matrix estimation method, of which the data-driven OD estimation method in Krishnakumari et al. (2020) is a special case. In the final section we elaborate on the solution methodology of the generic method.

2.1. Equivalence combined origin–destination choice & gravity models

Consider a directed graph $G(\mathcal{V},\mathcal{E})$ with nodes (vertices) $v_i \in \mathcal{V}, i = 1,\dots,N_v$ and links (edges) $e_a \in \mathcal{E}, a = 1,\dots,N_a$. The set $\mathcal{X} \subset \mathcal{V}$ describes the N_x origin and destination zones in this network, and an OD matrix with elements x_{ij} describes the OD flows between $v_i, v_j \in \mathcal{X}$. Finally, P_i and A_j depict the production and attraction of origin and destination zones i, and j respectively. Let

$$f_{ij} = \exp(U_{ij}) \tag{3}$$

describe a "deterrence" function between zones i, j, with U_{ij} a utility function with attributes X_b^{ij} (e.g. cost, travel time) and weight parameters α_b, which reads

$$U_{ij} = \sum_b \alpha_b X_b^{ij} \ . \tag{4}$$

Then

$$x_{ij} = a_i b_j P_i A_j \exp(U_{ij}) \ , \tag{5}$$

$$a_i = \frac{1}{\sum_j b_j A_j \exp(U_{ij})} \ , \tag{6}$$

$$b_j = \frac{1}{\sum_i a_i P_i \exp(U_{ij})} \ , \tag{7}$$

describe a doubly constrained gravity model, in which a_i, b_j are adjustment factors consolidating zone production and attraction totals. Anas (1983) rigorously proves the equivalence of this gravity model with an MNL model for combined origin–destination choice (e.g. commuters choosing home and work locations).

To understand this equivalence, consider that the OD flow computed from the MNL model can be expressed as

$$x_{ij} = \beta_{ij} \sum_{i'} P_{i'} \ , \tag{8}$$

or

$$x_{ij} = \beta_{ij} \sum_{j'} A_{j'} \ , \tag{9}$$

in which the fraction of travelers choosing destination j from origin i

$$\beta_{ij} = \frac{\exp(U_{ij})}{\sum_{i'}\sum_{j'} \exp(U_{i'j'})} \tag{10}$$

represents the scaled (dis)utility for simultaneously choosing origin i and destination j. The scaling relates to the underlying *perceived* (dis)utility $U_{ij} = -(\tilde{\alpha} TT_{ij} + \epsilon_{ij})$, in which ϵ_{ij} is an IID Gumbel distributed "error" term (over the population) with mode $\mu = 0$, and variance $\sigma^2 = \lambda^2\pi^2/6$, so that $\alpha = (\pi^2/6\lambda^2)^{1/2}\tilde{\alpha}$. For further details we refer to Anas (1983) and the wealth of literature discussing MNL models before and thereafter. In this paper we just use the resulting "market share" computation using (10), in this case for a particular OD alternative $\{i, j\}$.

It is not difficult to see that Eqs. (5) to (7) and Eq. (8) or (9) describe the same OD prediction model, up to what Anas (1983) calls "aggregation errors". Put simply, both models use (assume) the same utility function but utilize a different computing procedure. This results in the same OD flow pattern, safe for differences that emerge due to the fact that the MNL model uses a sample of microscopic data, whereas the gravity model uses aggregated flows. More precisely, in the gravity formulation (Eqs. (5) to (7)), the parameters (α) are typically a-priori modeling choices, and the balancing factors a_i, b_j are found through optimization (e.g. entropy maximization or information minimization) with *macroscopic* data such as OD travel times TT_{ij} and (cordon) counts y_a. Conversely, in the MNL formulation (Eq. (8) or (9)), the parameters $\{\dots,\alpha_b,\dots\}$ in utility specification (4) result from a utility maximization process using a sample of *microscopic* data (individual choices) from e.g. surveys or diaries. The "balancing" factors β_{ij} now follow directly from this choice model and observed (dis)utility components X_b^{ij} such as average OD travel time observations TT_{ij}.

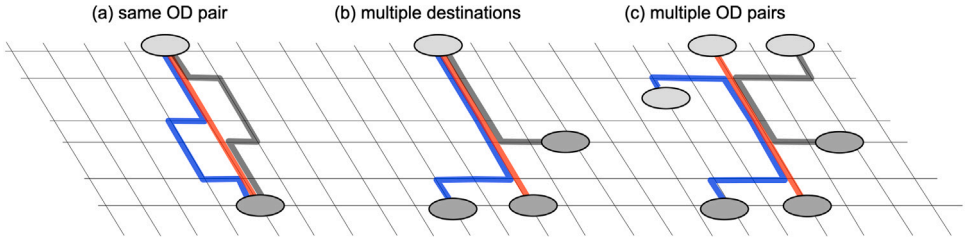


Fig. 2. Examples of path overlap in the joint OD-path estimation problem.

2.2. Extension to (dynamic) path-choice

It is straightforward to extend this model to one that, for a given departure time period $k = 1, 2, \dots, N_k$, describes joint origin–destination-path choice, which analogously to above (in "gravity modelling terms") equates to predicting OD-*path* flows x^k_{ijn} over path-sets $\mathcal{P}^k_{ij} \subset \mathcal{P}^k \subset \mathcal{P}$ containing (shortest) paths $p^k_{ijn} \in \mathcal{P}^k_{ij}, n = 1, \dots, N_{ij}$ between OD pairs $\{i, j\}$, for each departure time period k. We depict the full set of shortest paths (i.e. over all OD pairs) per departure period k with $p^k_r \in \mathcal{P}^k, r = 1, 2, \dots, N_p$ in which $N_p = \sum_i \sum_j N_{ij}$. Note that we will use the shorthand index r to run over all paths between all origins and destinations whenever possible to avoid a large number of subscripts. We first discuss a few prior considerations.

The first is that, although this extension implies estimating path-flows x^k_{ijn}; the primary objective is (still) to find the most likely OD matrix $x^k_{ij} = \sum_n x^k_{ijn}$. Expanding OD estimation with multiple paths per OD pair enhances realism by using actual travel data and topology constraints. Access to path travel times reduces reliance on assumptions, simplifying the estimation process. This approach offers superior evidence for OD matrix estimation compared to traditional methods. By streamlining computation, it improves efficiency while maintaining reliability. Overall, it provides a more accurate and practical model for understanding travel patterns. Put simply, extending the OD estimation problem with paths improves the quality of the evidence dramatically. The price we pay for this improved evidence is an increase of solution space dimensionality from $\mathcal{R}^{N_x}$ to $\mathcal{R}^{N_x \times N^*_{ij}}$ (with N^*_{ij} the average number of paths per OD pair). However, as proposed in Krishnakumari et al. (2020) and discussed below, we can in turn reduce the solution space dimensionality considerably — without significant loss of estimation accuracy — by estimating just those OD flows that explain most of the temporal variance by applying PCA on production and attraction data.

The second remark is that departure time is not considered as a choice dimension. Rather, discrete time k is added as a label because the set of shortest paths $\mathcal{P}^k$ usually changes over time due to changing traffic conditions. The estimator can thus be invoked over consecutive time periods using dynamic data such as time series of zone productions P^k_i and attractions A^k_j, link counts y^k_a, travel times TT^k_{ijn}, etc. We return to this point further below. Third, there is no hierarchy implied in the choice dimensions which are considered, i.e. origin–destination-path. Instead of choosing between OD trips at average (path) costs, travelers now choose from a larger choice set of trips, which encompasses multiple origin, destination and path options per trip.

Finally, for all this to work, it is imperative to take into account path-overlap (e.g. Ben-Akiva et al. (2012)), not just for paths between the same OD pairs, but for all paths, since travelers in this joint OD-path choice problem consider all these options simultaneously. Herein there is no conceptual difference between path overlap for paths between e.g. a single OD pair; paths that share an origin or destination; or paths that share common links only—Fig. 2 gives three examples in case. It may seem far-fetched that travelers consider multiple OD and path choice options simultaneously, particularly in case of commuting, in which the O–D dimensions represent long(er) term decisions and path choice more flexible short term decisions. However, in line with the argument in Anas (1983), we are not interested in which order individuals make their choices, but much rather, in the net combined result of all those choices in a population of travelers.

These considerations lead to a joint Path Size Logit (PSL) origin–destination-path choice model with the following generic (dis)utility specification (Ben-Akiva and Bierlaire, 1999):

$$U_r = -\left[\alpha^r_0 + \sum_b \alpha_b X^r_b\right] + \alpha_{ps} \ln PS_r \ , \tag{11}$$

in which α^r_0 is a path specific utility constant (PSC, we return to it in the next section); $[\dots, X^r_b, \dots], b = 1, \dots, B$ is the vector of path-specific cost components (travel time, etc.); $[\dots, \alpha_b, \dots]$ is a vector with associated weights, and PS_r a (distance-based) path size factor with α_{ps} the penalization weight (Ben-Akiva and Bierlaire, 1999; Ben-Akiva et al., 2012).

$$PS_r = \sum_{a=1}^{N_a} \left(\frac{l_a}{L_r}\right) \frac{1}{\sum_r \delta^a_r} \ , \tag{12}$$

that corrects for inflated utility differences between overlapping paths. In (12) N_a depicts the number of links on path r, l_a depicts the length of link e_a; L_r the length of path p_r and δ^a_r is the link-path incidence variable which equals one if link a is on path r and zero otherwise. Analogously to (8) the resulting path flows can now be expressed as

$$x^k_{ijn} = a_i b_j P^k_i A^k_j \exp(U_{ijn}) \ , \tag{13}$$

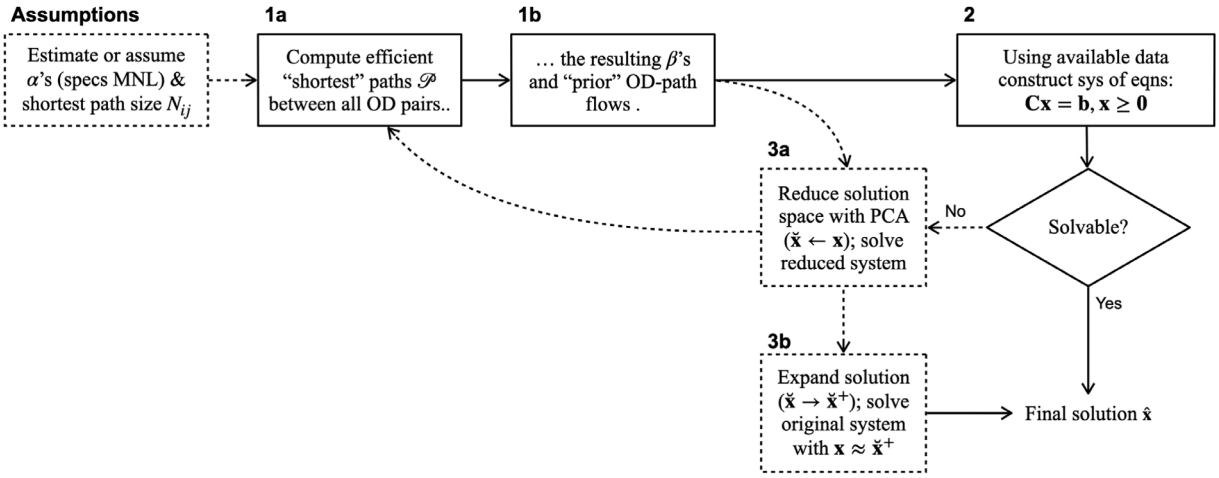


Fig. 3. Schematic overview of the combined OD-path flow estimation solution methodology. For clarity, the time superscript k is omitted.

in which the fraction of travelers simultaneously choosing path r, i.e. the nth path from origin i towards destination j, equals

$$\beta_{ijn} = \beta_r = \frac{\exp(U_r)}{\sum_{r'} \exp(U_{r'})}, \tag{14}$$

Eqs. (11), (12), (13), and (14) describe a doubly constrained path-based gravity model, with a decreasing deterrence function. The data driven OD estimation method proposed in Krishnakumari et al. (2020) is a special case of this model, in which an implicit choice is made for the path utility specification in (11) so that

$$U_r^k = -TT_r^k(1 - PS_r) \ ,$$

in which only path travel time TT_r^k is considered as cost component; an implicit assumption is made about the associated weight (i.e. $\alpha = 1$); and no path-specific constants are considered. Note that also the pathsize factor in Krishnakumari et al. (2020) is formulated slightly differently — multiplicative instead of additive and without logarithmic scaling — compared to the general utility formulation we present in Eq. (15) further below. Additionally, in Krishnakumari et al. (2020) an assumption is made on the path choice set size, which is considered equal for all OD pairs, that is, $N_{ij} = N^*, \forall\ i, j$. Put differently, for all OD pairs the same number of *used* shortest paths are assumed. Finally, in Krishnakumari et al. (2020) the "β's" in Eq. (14) are normalized per OD pair, rather than over the entire choice set. This means the formulation in Krishnakumari et al. (2020) is not strictly equivalent to a corresponding joint MNL model for OD-path choice (but it is in principle). In the next section we elaborate on some specific choices and assumptions related to utility specification (11) and we propose a methodology for solving this generic model.

2.3. Generic solution for OD matrix estimation

Solving the combined OD-path gravity model, and by implication the OD matrix estimation problem, implies formulating sufficient constraints for the unknown OD-path-flows to construct a solvable (i.e. full-rank) system of equations, or, if this is not possible, to reduce the dimensionality of the solution space so that a solvable system *can* be constructed. In doing so, we aim to reconcile the available evidence from different data sources, i.e., path travel times TT_{ijn}^k; link counts y_a^k; time series of zone production P_i^k and attraction A_j^k totals, and, importantly, the dynamic graph $G(\mathcal{V}, \mathcal{E}, k)$ describing the (infrastructure) network, including dynamic edge weights (e.g. speeds) and static characteristics such as road type, length, etc. The main steps in the methodology are summarized below and schematically outlined in Fig. 3.

Assumptions The starting point of the methodology are assumptions, e.g., a choice for the (initial) path set sizes N_{ij}, and specification (calibration) of the utility function (11).

Step 1 Compute path sets $\mathcal{P}_{ij}$ with N_{ij} shortest paths — shortest in the sense of maximum utility (11) — for all OD pairs i, j. From these, compute path shares β_{ijn}^k (14) and the corresponding "prior" path flows $\check{x}_{ijn}^k$ (13). In Section 2.3.2 we will describe two alternative shortest path algorithms (LP and ESX) used in this step and discuss the computational costs associated with them.

Step 2 Construct a system of equations $\mathbf{CX} = \mathbf{b}, \mathbf{X} \geq 0$ (see below) and solve for OD flows x_{ij}^k. If the system has sufficient rank solve the equations to obtain the posterior OD flows, and the corresponding posterior path flows $\hat{x}_{ijn}^k$, which follow from re-normalizing the β's per OD pair. If the system is not solvable *go to step 3*.

Step 3 Apply PCA to the prior OD flows $\tilde{x}_{ij}^k$ — which in matrix notation reads $\tilde{\mathbf{X}}$ — to reduce the dimensionality. Using PCA the unknown matrix is approximated as a linear combination of the largest eigenvectors $\tilde{\mathbf{X}} = \mathbf{Z}\mathbf{V}^T + \mu_{\tilde{\mathbf{X}}}$. Then convert and solve the system equations $\mathbf{C}'\mathbf{Z} = \mathbf{b}'$. Expand solution using principal components to recover the full OD matrix.

2.3.1. Assumptions and considerations

The starting point for computing the N_{ij} "shortest" (i.e. minimum disutility) paths between all OD pairs $\{i, j\}$ is to choose N_{ij}. For example, in Krishnakumari et al. (2020) the best results are reported with $N_{ij} = 5, \forall i, j$, but any reasonable — possibly OD-specific — assumption is possible. The optimal "cut-off" number of shortest paths will depend on network topology and management/control policies. Second, a (dis)utility specification (i.e., Eq. (11)) is required, which will result in the most likely distribution of trips over these paths given the available data. This utility model encodes assumptions about behavioral preferences and physical or regulatory constraints that govern how attractive or costly a specific path alternative between an OD pair is. In very specific cases, e.g. large-scale events with clearly identifiable and mandatory paths between specific OD pairs, large path-specific constants (PSC) could be used to encode mandatory choices for these paths, regardless of high travel time or other costs along it. Without loss of generality, however, in this paper we do not consider PSC's, i.e. $\alpha_0^r = 0$, but an OD-specific constant α_0^{ij} is considered instead which functions as a balancing factor to address disparities in order of magnitude. This adjustment is a consequence of the combination of origin–destination-path choice, as opposed to treating each OD pair independently in conventional approaches. Consequently, the costs attributed to different OD pairs inherently vary in orders of magnitude, and the OD-specific constant aids in equalizing these discrepancies.

The key requirement for the utility specification is that it supports (efficient) shortest path algorithms in 3D (network × time) graphs $G(\mathcal{V}, \mathcal{E}, k)$, that is, graphs with dynamic link speeds u_a^k, flows q_a^k, or other properties w_a^k through which paths between a given OD pair $\{i, j\}$ starting in period k can be constructed (for brevity, here we use index a as a shorthand of link e_a to discuss on the utility function specifications). Most importantly, this requires that the (generalized) costs along a path are additive, that is, equal to the sum of the link costs along the path. To this end, we propose two main utility components, w_r^k and τ_r^k. The former encodes functional properties of the path p_r^k, which may be static (e.g. road type) or dynamic (e.g. tidal lanes, capacity restrictions); the latter represents (generalized) travel time, respectively. This yields the following general (dis)utility specification:

$$U_r^k = -\left(\alpha_0^{ij} + \alpha_w w_r^k + \alpha_\tau \tau_r^k\right) + \alpha_{ps} \ln PS_{ijn} \,, \tag{15}$$

Since we consider a dynamic network, in the limit of infinitely small time periods ($\Delta t \downarrow 0$, i.e. continuous time), both utility components represent path-integrals over p_r^k, that is, $w_r^k = \int_{p_r^k} w(s)ds$; and $\tau_r^k = \int_{p_r^k} \tau(s)ds$, respectively. In the discrete case, with constant link properties w_a^k and τ_a^k during each discrete period k of typically one or a few minutes, the computations become simple summations along path p_r^k, that is,

$$w_r^k = \sum_{a=1}^{N_a} \delta_{p_r^k}^a w_a^k \tag{16}$$

$$\tau_r^k = \sum_{a=1}^{N_a} \delta_{p_r^k}^a \tau_a^k \,. \tag{17}$$

Generalized route travel time (τ_r^k) in turn is computed using link length (l_a), dynamic link speed (u_a^k), link costs c_a^k (e.g. tolls), and value of time (VoT) η_τ, that is,

$$\tau_a^k = \frac{l_a}{u_a^k} + \frac{c_a^k}{\eta_\tau} \,. \tag{18}$$

We emphasize that, since link speeds and costs are given (observed), both functional link properties (16) and generalized link travel times (17) can be pre-computed through linear combination (i.e. $\sum_a [\alpha_w w_a^k + \alpha_\tau \tau_a^k]$) to construct the 3D graph $G(\mathcal{V}, \mathcal{E}, k)$ and thus to incrementally compute shortest paths in the sense of maximum utility (i.e., minimum disutility) according to (15). Note that in the simplified case of travel time cost *only* (Krishnakumari et al., 2020), constructing path travel time in a 3D network boils down to applying a piece-wise constant speed-based trajectory method (Van Lint, 2010).

Two final assumptions are required to solve the OD estimation problem using this path-flow formulation. The first is that to map path-flows to link-flows, we discretize observed travel times using the observation period duration Δt—more on this further below. The second is that FIFO (first-in-first-out) applies to all links. Link e_a satisfies the FIFO property if for each pair $\{t, t'\}$ of times with $t < t'$, $TT_a^t + t \leq TT_a' + t'$. The FIFO property further implies that no utility can be gained from waiting at a node before traversing the link of interest. This implies that the key requirement for a suitable shortest path algorithm in our case is similar to that of any shortest path problem on a graph with given link weights: *computational efficiency*.

2.3.2. Step 1: shortest paths computation

A major assumption in Krishnakumari et al. (2020) is the "cut-off" number of shortest path N_{ij}^* for all OD pairs. A constant value $N_{ij}^* = N^*, \forall i, j$ disregards variations in network structure and management policies across different OD pairs. To address this limitation, we employ two alternative shortest path algorithms that *endogenously* determine the appropriate number of shortest paths for each OD pair. Additionally, both algorithms incorporate mechanisms that limit path overlap—this is crucial because in the joint OD-path formulation this path overlap problem increases dramatically. We provide detailed explanations of both algorithms below.

The first algorithm, referred to as the link penalty (LP) algorithm, adopts a penalization mechanism for searching shortest paths (Cheng et al., 2019). In this approach, given a specified number of shortest paths K, the algorithm iteratively applies the Dijkstra algorithm, while artificially increasing costs on reused links. In the first iteration, the Dijkstra algorithm finds the actual shortest paths based on the original cost. In subsequent iterations, for each link already utilized, the link cost is multiplied by a weight λ to increase it. Increasingly, new paths computed on this "penalized" network may already exist in the set of previously computed paths, enabling the algorithm to determine an appropriate number of feasible (in the sense of costs) shortest paths.

The second algorithm, known as ESX, is a heuristic shortest path algorithm that addresses path overlap in a different way (Chondrogiannis et al., 2020). Instead of applying penalties to used links, the ESX algorithm progressively *removes* links altogether. Similar to the LP algorithm, the ESX algorithm first identifies the actual shortest path using the Dijkstra algorithm. It then removes links with the highest link cost and conducts the Dijkstra search on the pruned network. Due to the link removal mechanism, in networks with lower connectivity, the number of feasible shortest paths can be lower than the expected K, thus endogenously determining the appropriate number of shortest paths.

Compared to typical K-shortest path algorithms (e.g., Yen's algorithm) that compute exactly K paths, both the LP and ESX algorithms can determine OD-specific numbers of shortest paths, but use different considerations to arrive at these. The LP algorithm penalizes previously used links by increasing their costs with a fixed coefficient, allowing penalized paths to still be labeled as "shortest". Consequently, a new path is only discovered when previously used paths become excessively costly. Conversely, the ESX algorithm adopts a more direct approach by progressively removing links from the network to search for new paths. Our major interests in terms of the differences between both algorithms pertain to two questions: (1) do they generate a reasonable set of (indeed) shortest paths; and (2), can this path-set be computed fast enough to make it a feasible option for our OD estimation method. We evaluate the two algorithms in Section 4 on both aspects. For more in-depth technical information about the algorithms and their implementation, we refer to Cheng et al. (2019) and Chondrogiannis et al. (2020).

The shortest paths $\mathcal{P}$ computed from either algorithm above will result in the (time-dependent) OD-path market shares (the β's), i.e.

$$\beta_{ijn}^{k} = \frac{\exp(U_{ijn}^{k})}{\sum_{i'}\sum_{j'}\sum_{n'}\exp(U_{i'j'n'}^{k})}\ , \tag{19}$$

in which U_{ijn}^{k} is computed according to utility specification in Eq. (15). These in turn allow us to compute a corresponding set of (a-priori) path flows $\breve{x}_{ijn}^{k}$, which read

$$\breve{x}_{ij}^{k} = \sum_{n}\left(\beta_{ijn}^{k}\sum_{i'}P_{i'}^{k}\right) = \sum_{n}\left(\beta_{ijn}^{k}\sum_{j'}A_{j'}^{k}\right) \tag{20}$$

An a-posteriori estimate $\hat{x}_{ij}^{k}$ of the OD matrix can now be constructed by adding constraints using link counts (and possibly other data), which scales and restructures the prior towards observed link flow totals.

2.3.3. Step 2: full system of equations

We now construct a system of equations to estimate the full dynamic OD matrix $x_{ij}^{k}, \forall\, i,j,k$. First consider link counts y_{a}^{m}, which are fully informative of demand, i.e. of the set of path flows $\mathcal{P}_{a}^{k\leq m}$ that go through link e_a in period $m \geq k$. This is the case if (and — for the first two points — only if)

1. link e_a is not congested in period m;
2. none of the links on the paths $p_{r}^{k} \in \mathcal{P}_{a}^{k\leq m}$ upstream of e_a were congested during period $[k,m]$; and
3. the travel time $TT_{r|a}^{k}$ on route r up to link a starting in period k is (approximately) equal to the time difference between the link count and the departure of the path flow, that is, $(TT_{r|a}^{k}/\Delta t) - (m-k) \approxeq 0$

The first two requirements are needed because otherwise y_{a}^{m} is — at least partially — composed of queue discharge flows and thus not informative of demand. The third requirement follows from the FIFO assumption and observed travel times. As a result we can now write

$$y_{a}^{m} = \sum_{k=m-\lfloor TT^{max}/\Delta t\rfloor}^{m}\ \sum_{r\in\mathcal{P}_{a}^{k}}\delta_{r|a}^{mk}x_{r}^{k}\ . \tag{21}$$

in which TT^{max} is the maximum travel time from any of the origin nodes i towards e_a (a pragmatic choice would be a sufficiently high travel time suitable for all links), and

$$\delta_{r|a}^{mk} = \begin{cases} 0, & \left|(TT_{r|a}^{k}/\Delta t) - (m-k)\right| \geq \epsilon_{TT}, \\ 1, & \text{else}\ , \end{cases} \tag{22}$$

the dynamic path/link flow indicator variable, in which ϵ_{TT} is a small number to account for travel time round-off errors. By switching LHS (left-hand-side) and RHS (right-hand-side) of (21), and reformulating as a sum over paths per OD pair $\{i,j\}$, we have:

$$\sum_{i}\sum_{j}\ \sum_{k=m-\lfloor TT^{max}/\Delta t\rfloor}^{m}\ \sum_{n=1}^{N_{ij}}\delta_{r|a}^{mk}\beta_{ijn}^{k}x_{ij}^{k} = y_{a}^{m}\ . \tag{23}$$

Eq. (23) represents the connection between the link count y_a^m and the various time-dependent path flows x_r^k that potentially traverse link e_a during time $m \geq k$. Next, we expand (13) into:

$$\sum_j \sum_{n=1}^{N_{ij}} \beta_{ijn}^k x_{ij}^k = P_i^k \tag{24}$$

$$\sum_i \sum_{n=1}^{N_{ij}} \beta_{ijn}^k x_{ij}^k = A_j^k \tag{25}$$

The set of Eqs. (23), (24), and (25) form a large system of equations when expanded for all origins $i = 1, 2, \ldots, N_x$, destinations $j = 1, 2, \ldots, N_x$, and time periods $k = 1, 2, \ldots, N_k$. The unknown OD matrix x_{ij}^k can be solved by transforming this system into a matrix equality

$$\mathbf{CX} = \mathbf{b}, \quad \mathbf{X} \geq \mathbf{0}; \tag{26}$$

where $\mathbf{X}$ represents the OD matrix x_{ij}^k, while $\mathbf{C}$ and $\mathbf{b}$ denote the matrix and vector containing the market shares, link-flow proportions, and the RHS elements of Eqs. (23)–(25), respectively.

The matrix equality represented by Eq. (26) can be solved as a bound-constrained minimization problem (Branch et al., 1999) with a lower bound constraint set at 0 to ensure the non-negativity of the solution. By incorporating this constraint, the estimated origin–destination (OD) matrix is guaranteed to contain only non-negative values.

2.3.4. Step 3: reduced system of equations

The number of OD flows grows quadratically with the number of production and attraction zones. However, the increase in the number of rows in matrix Eq. (26) is linear with respect to the number of zones in Eqs. (24) and (25), and the number of link flow constraints in Eq. (23). Consequently, in large networks with limited link flow constraints, the linear system represented by Eq. (26) becomes severely underdetermined. To solve the OD matrix estimation problem for such large networks, we use insights from prior research (Djukic et al., 2012b; Zhou and Mahmassani, 2007) which suggest that a substantial portion of demand flow variance can be ascribed to dominant temporal patterns. These patterns primarily pertain to daily and weekly seasonal fluctuations, whereas deviations from these patterns and random fluctuations constitute minor components (Djukic et al., 2012b).

Krishnakumari et al. (2020) assume a similar phenomenon holds for the production and attraction flow totals P_i^k and A_j^k, and use PCA to reduce the dimensionality of these time series. In doing so, the dominant production and attraction zones are identified and a reduced OD set is constructed. Solving the system of equations of this reduced OD flow set *only* provides an upper bound for the same (dominant) OD flows in the original system of equations. In this way, the solution space is constrained sufficiently to find a reliable solution. However, as will be presented in the results, this upper bound may not always be valid — e.g. the actual values may be larger than the upper bound — and therefore may introduce errors in the final solution.

So, instead of providing an upper bound when solving the original equations, we propose to use PCA to directly on the (prior) OD matrix as in Djukic et al. (2012b), which also reduces the dimensionality of the system described by (26). The rationale in this paper is thus similar to Krishnakumari et al. (2020), but with a few key adjustments: (a) we compute the principal components on the prior OD matrix (computed with the β's obtained from the shortest path algorithm in Eq. (20)) instead of applying PCA on the productions and attractions time series; (b) we do not solve the original system of equations but transform Eq. (26) to a reduced version $\mathbf{C'Y} = \mathbf{b'}$, where $dim(\mathbf{Y}) \ll dim(\mathbf{X})$; and (c) we then "inversely transform" the reduced solution $\hat{\mathbf{Y}}$ to a full (and final) solution $\hat{\mathbf{X}}$ by linear combination of the principal components.

We emphasize that PCA is a linear procedure that re-structures the solution space in terms of orthogonal directions of decreasing (co)variance. Cutting off the transformed solution-space beyond some arbitrary amount of explained variance may inadvertently remove relevant non-linear correlations—we discuss this and other limitations in Section 4.4.

Below we outline the main procedure; for further details on the PCA method, refer to e.g. Jolliffe (2002).

Consider the prior OD flow computed with Eq. (20) in the form of matrix

$$\tilde{\mathbf{X}} = [\cdots, \tilde{\mathbf{x}}_k^T, \ldots] = \begin{bmatrix} \tilde{x}_{11}^0 & \cdots & \tilde{x}_{ij}^0 & \cdots \\ \vdots & \ddots & \vdots & \ddots \\ \tilde{x}_{11}^k & \cdots & \tilde{x}_{ij}^k & \cdots \\ \vdots & \ddots & \vdots & \ddots \end{bmatrix} \tag{27}$$

where each column vector $\tilde{\mathbf{x}}_k$ is the set of OD flows at time period k. Let $\boldsymbol{\mu}_{\tilde{\mathbf{X}}}$ depict the mean matrix of $\tilde{\mathbf{X}}$. By applying PCA with N_p components (or n_p% variance explained), we have

$$\tilde{\mathbf{X}} = \mathbf{ZV}^T + \boldsymbol{\mu}_{\tilde{\mathbf{X}}} \tag{28}$$

where the column vectors of $\mathbf{Z}$ are a set of orthogonal uncorrelated variables, i.e., principal components. In Eq. (28) the original matrix is approximated using a linear combination of N_p principal components. If N_p is large, the approximation is near-perfect; if it is small, we disregard some of the temporal variations in the multi-variate "OD signal" (and accept some error-variance in the reconstruction later on). Elementwise, Eq. (28) can be written as

$$\tilde{x}_{ij}^k = \sum_{p=1}^{N_p} z_p^k v_{ij}^p + \mu_{ij} \tag{29}$$

Finally, we replace the original OD flow variables in Eqs. (23) to (25) with the PCA approximation in Eq. (29), so that the original problem with unknowns x_{ij}^k is transformed into a reduced problem with unknowns z_p^k, which we can write as

$$\sum_i \sum_j \sum_{k=m-\lfloor TT^{max}/\Delta t \rfloor}^{m} \sum_{n=1}^{N_{ij}} \sum_{p=1}^{N_p} \delta_{r|a}^{mk} \beta_{ijn}^k v_{ij}^p z_p^k = y_a^m - N_k \sum_i \sum_j N_{ij} \mu_{ij} \ . \tag{30}$$

$$\sum_j \sum_{n=1}^{N_{ij}} \sum_{p=1}^{N_p} \beta_{ijn}^k v_{ij}^p z_p^k = P_i^k - \sum_j N_{ij} \mu_{ij} \tag{31}$$

$$\sum_i \sum_{n=1}^{N_{ij}} \sum_{p=1}^{N_p} \beta_{ijn}^k v_{ij}^p z_p^k = A_j^k - \sum_i N_{ij} \mu_{ij} \tag{32}$$

For brevity, the reduced problem can be written in matrix form as $\mathbf{C}'\mathbf{Z} = \mathbf{b}'$. By this transformation, the dimension of the original problem is reduced from $N_k \times N_x \times N_x$ to $N_k \times N_p$, which is now potentially an *overdetermined* system as $N_p \leq N_k$. By applying ordinary least squares, we can find a unique solution $\hat{\mathbf{z}}$ to the reduced system of equations. The full solution can then be obtained via Eq. (29). Furthermore, to ensure non-negativity in the final OD estimates, we employ a heuristic approach to set any negative values to zero.

2.3.5. Summary of method and relation to literature

Our method has some similarities with existing data driven path-flow estimators (e.g. Ma and Qian (2018), Rao et al. (2018), Wei and Asakura (2013) and Wu et al. (2018)) but it differs in terms of the combination of data sources it uses and in the methods and underlying assumptions it applies to fuse these data. Specifically, we use four data sources, that is, (1) link speeds from which we derive path travel times τ_r and functional path properties w_r (i.e. fraction of the path over motorways); (2) estimated utility weights (the α's in utility function (15)) to trade off these choice dimensions and compute the OD-path-market shares (the β's) which in turn are combined with (3) zone production and attraction totals P_i, and A_j to construct a prior OD matrix; and finally, (4) link counts, which are used to scale this prior to a full posterior estimate of the OD matrix.

It is important to emphasize that we use a path-flow estimation method because this allows us to use superior evidence for this posterior OD matrix, i.e. observed path travel times versus average zone to zone travel times. The consequence of using *observed* travel times, is that we do not (have to) consider (stochastic) equilibrium assignment principles as in Abareshi et al. (2017) and Wei and Asakura (2013). Rather, we formulate the problem as a large system of equations constraint by the data as in Ashok and Ben-Akiva (2002), Nie et al. (2005), and apply a (plausible link-additive) utility function as a heuristic to derive a plausible distribution of trips over all OD-path alternatives. A further difference of our method with respect to other methods is that we utilize (two alternative) shortest path algorithms to endogenously compute *OD-specific* path choice sets, thereby generalizing (Krishnakumari et al., 2020). In computing the OD-path-flow "market shares", we furthermore assume that the equivalence of the doubly constraint gravity model and the MNL formulation (Anas, 1983) also holds for joint OD-path choice, not just theoretically but also empirically—this is an assumption we will test further below. Finally, we explicitly exclude link counts which are not informative of demand and use PCA to reduce the dimensionality of the prior OD matrix in case the problem becomes underdetermined. We then solve the reduced problem, and transform that solution back to a full posterior OD matrix.

Also, one may find that our prediction–correction mechanism appears similar to Bayesian inference methods (prior-posterior). Although our approach is not explicitly Bayesian, it does share certain conceptual parallels. In our method for dynamic OD estimation in congested networks, we employ a combination of heuristics and models carefully chosen to transform the estimation problem into a solvable linear system. For instance, the MNL model facilitates the computation of plausible OD-shortest path sets from travel times and other additive cost components, allowing us to construct a prior OD matrix. This matrix, combined with trip productions, serves as a foundation for our prediction–correction mechanism. Through successive steps of refinement, such as incorporating link flow constraints and applying PCA to filter out noise, our method iteratively updates this prior to what could be informally considered as a posterior, although not in a strictly Bayesian sense. The use of PCA, in particular, highlights a level of "degeneracy" in our prior that varies depending on the complexity of demand dynamics, demonstrating how our method adapts to different scenarios. While our approach may not fit neatly into traditional Bayesian frameworks, its conceptual alignment with aspects of Bayesian inference underscores the sophistication and flexibility of our methodology.

In summary, we use a mix of data sources and assimilation methods (both forward and reverse as illustrated in Fig. 1) and make just those assumptions needed to fuse these data. If the resulting system of equations is nonetheless below rank, we reduce its dimensionality until it is solvable.

3. Case setup

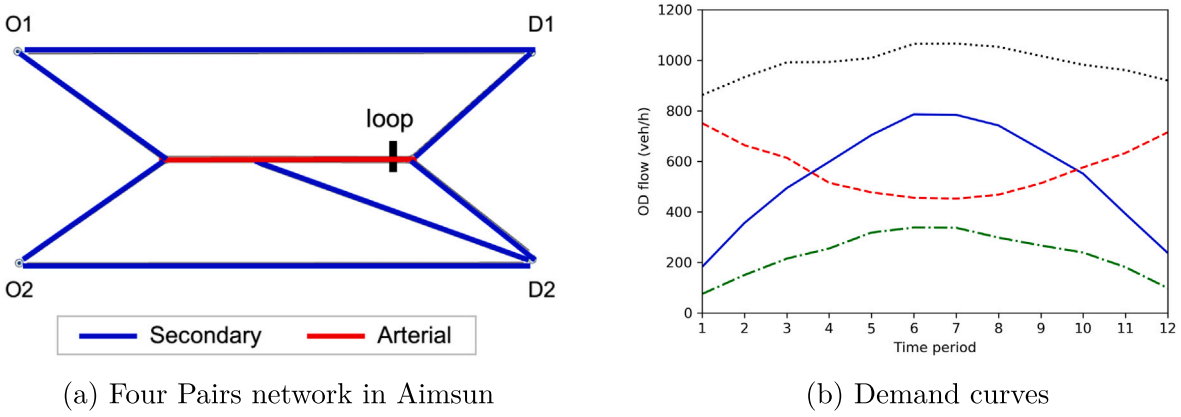
This section describes the case setup, in which we present the configuration of (7) different networks and describe the experiments conducted on each of them. The overall setup is summarized in Table 1.

As shown in Table 1, the first 5 networks are well-known transportation networks widely used for studying static traffic assignment problems. The data for these are publicly available,[2] and contain network topology, static OD demand, and assigned

[2] For example at https://github.com/bstabler/TransportationNetworks.

Table 1
Case study setup.

Networks	Nodes	Links	OD pairs	Validation of …	Varying with …
Sioux falls	24	76	576	–	Path set generation algorithms
Anaheim	416	914	1406	–	Path set generation algorithms
Winnipeg	1052	2836	4345	–	Path set generation algorithms
Barcelona	1020	2522	7922	–	Path set generation algorithms
Chicago-Sketch	933	2950	142,890	–	Path set generation algorithms
Four-pairs	7	9	4	Extended equivalence The proposed method	PSL utility functions Levels of demand
Santander	1630	4205	13,689	Extended equivalence The proposed method	Path set generation algorithms Number of shortest paths Degrees of noise in PSL Usage of PCA Number of principal components in PCA



(a) Four Pairs network in Aimsun (b) Demand curves

Fig. 4. Four pairs network and demand.

static link flow. In order to provide a comprehensive analysis, the two algorithms (LP and ESX) that generate a set of K shortest paths with limited overlap are qualitatively and quantitatively compared on these "classic" networks, which vary in scale. It is important to note that the validation of the overall OD estimation cannot be performed on these networks due to the absence of dynamic OD matrices and observations in the available data. The comparative analysis primarily focuses on three key aspects: computation time, the path set cost, and the spatial distribution of the paths.

To validate the complete method proposed in this paper we use two simulation networks, Four-pairs and Santander, and we use Aimsun Next software (Aimsun, 2017) as a ground-truth platform for our validations and tests.

The first network, referred to as the Four-pairs network, is a toy network consisting of two origins and two destinations. Fig. 4(a) provides a visual representation of the network. It is important to note that only one link is equipped with a loop detector to measure the link flow. Despite the network's limited size, the system remains underdetermined under this configuration, as the number of observations (one) is smaller than the number of unknowns (four). The OD demand for the four OD pairs is generated using a combination of sine functions with random noise, as depicted in Fig. 4(b). The demand generation process has a time granularity of 10 min, and the total duration of the test period is two hours. During the OD estimation process, the demand is aggregated into 5-minute time intervals, resulting in a total of 24 time periods. Given the simplicity of route choice on this network, no assumptions were tested regarding the shortest path algorithms (as path enumeration on this network is straightforward). We first examine the validity of the extended equivalence by comparing the path flow computed from the doubly-constrained gravity model with that obtained from the PSL model. Subsequently, we establish a baseline case to validate the OD estimation method, and further evaluate the sensitivity of OD estimation performance to different forms of the PSL utility function and varying levels of demand. The utility functions used in this case read

$$U_{ijn}^{k} = -(\alpha_{\tau}\tau_{ijn}^{k}) + \alpha_{ps}\ln PS_{ijn} \qquad (ttonly) \tag{33}$$

$$U_{ijn}^{k} = -(\sum_{w=1}^{2}\alpha_{w}w_{ijn}^{k} + \alpha_{\tau}\tau_{ijn}^{k}) + \alpha_{ps}\ln PS_{ijn} \qquad (multi) \tag{34}$$

$$U_{ijn}^{k} = -(\alpha_{ijn}^{0} + \sum_{w=1}^{2}\alpha_{w}w_{ijn}^{k} + \alpha_{\tau}\tau_{ijn}^{k}) + \alpha_{ps}\ln PS_{ijn} \qquad (psc), \tag{35}$$

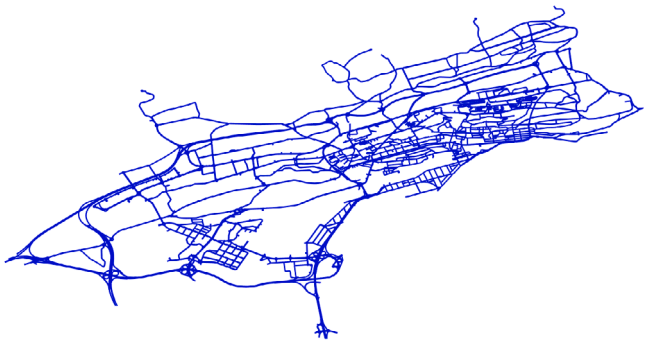


Fig. 5. Santander network.

Table 2
Metrics of the generated path set ($K = 10$).

Metric	Algorithm	Sioux falls	Anaheim	Winnipeg	Barcelona	Chicago sketch
Computation time	Yen's	1.91	59.19	798.60	–	–
	ESX	3.18	73.08	570.26	805.54	11 771.55
	LP	0.47	14.15	120.73	217.48	3929.15
Mean cost	ESX	25.54	19.53	27.94	12.90	75.89
	LP	12.74	14.76	17.86	10.13	68.77
Detour ratio	ESX	3.22	1.75	2.21	1.66	1.39
	LP	1.24	1.10	1.14	1.13	1.07

wherein w_{ijn} depicts the mileage fraction of two different road classes (arterial and secondary) on the path, τ^k_{ijn} depicts the dynamic path travel time, and α^0_{ijn} depicts the path-specific constants (PSC). For brevity, we refer to these three different forms of the utility function as *ttonly, multi* and *psc*.

The second network is a large and validated (with actual data) network of Santander, Spain. This network, depicted in Fig. 5, comprises 4205 links belonging to 4 different road classes. Among these links, 295 are equipped with loop detectors to measure link counts. The test period spans 4 h during peak periods, divided into 48 5-minute time periods. On the Santander network, we also begin by examining the extended equivalence as described earlier. Following this, we proceed with several tests using the calibrated demand. These tests include investigating different path set generation algorithms, the impact of the number of shortest paths (parameter K) specified in the algorithms, the two different usages of PCA mentioned in Section 2.3.4, the number of principal components in PCA, and the effect of noise in the PSL utility parameters. For this case study, the utility function employed corresponds to the *psc* form used in the Four-pairs network setting. The specific form of the utility function is as follows:

$$U^k_{ijn} = -(\alpha^0_{ijn} + \sum_{w=1}^{4} \alpha_w w^k_{ijn} + \alpha_5 \tau^k_{ijn}) + \alpha_{ps} \ln PS_{ijn} \tag{36}$$

Based on this utility function, we calibrate the weight parameters and compute 20 shortest paths for each OD pair, thereby constructing the path set. These paths are then used to compute the dynamic path shares.

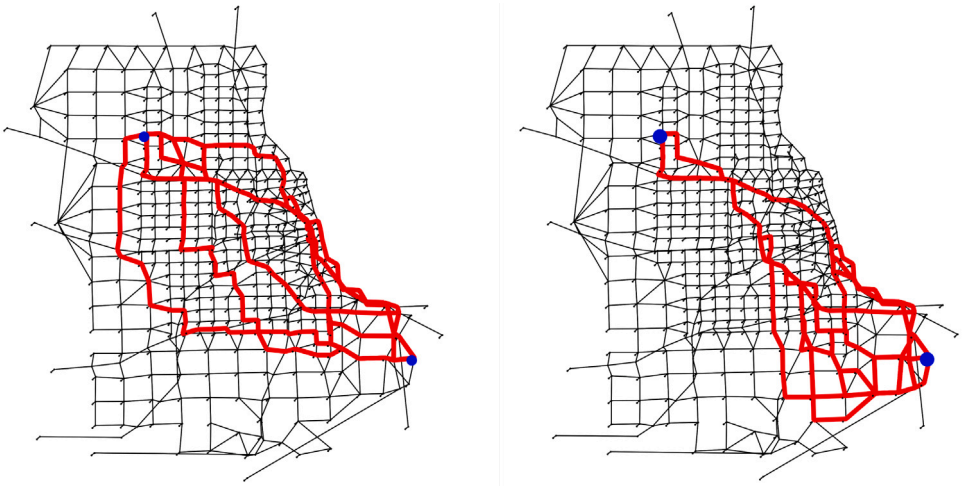
4. Case study results

4.1. Comparison of path set generation algorithms

Our hypothesis in Section 2.2 is that extending the OD estimation problem with path choice improves the quality of the evidence we can use. However, it does complicate the problem considerably. First it requires a plausible set of chosen paths per OD pair. In this section we assess the efficacy of two path set generation algorithms (ESX and LP). We examine the computational characteristics of these algorithms and the quality of the resulting path sets. Further below we discuss the impact on the overall OD estimation performance.

We first compare the computation time of the ESX and LP algorithms on the 5 aforementioned "classic" networks, using Yen's K shortest path algorithm as a benchmark. All tests are performed on a personal computer equipped with a 12-core Intel Core i7-9750H CPU and 32 GB of RAM. The computation time, measured in total time spent (in seconds), is presented in Table 2. As indicated in the table, both the LP and ESX algorithms exhibit superior performance and scalability compared to Yen's algorithm, with the LP algorithm performing fastest.

Second, we assess the mean path cost of the generated path sets, which is computed by averaging the path costs of all the paths within the set. Assuming both algorithms find the same actual shortest paths (both use Dijkstra), higher average path cost indicate



(a) LP algorithm (b) ESX algorithm

Fig. 6. Example of detour on the network of Chicago Sketch.

more diversity amongst the set. The results are presented in Table 2. As evidenced by the table, the mean path cost computed by the ESX algorithm is significantly higher than that of LP (with a cost reduction ranging from 20% to 50% for the first four networks and a reduction of 9.4% for the largest network). This disparity is expected since the ESX algorithm employs a "harder" mechanism in which links are removed from the network.

Lastly, we examine the spatial distribution of the path sets based on the detour ratio. The detour ratio is computed by dividing the path mileage by the shortest mileage between each OD pair and then taking average across the entire path set. The results are presented in Table 2. As can be observed in the table, the detour ratio computed from ESX is also higher than that of LP, partially accounting for the higher cost reported in Table 2. Additionally, the LP algorithm demonstrates a relatively stable detour ratio across different networks, whereas the ESX algorithm yields a fluctuating detour ratio (with a maximum of 3.22 and a minimum of 1.39). Fig. 6 provides an illustrative example on the Chicago Sketch network, showcasing how the ESX algorithm generates a path set with a high detour ratio comparing with the LP algorithm.

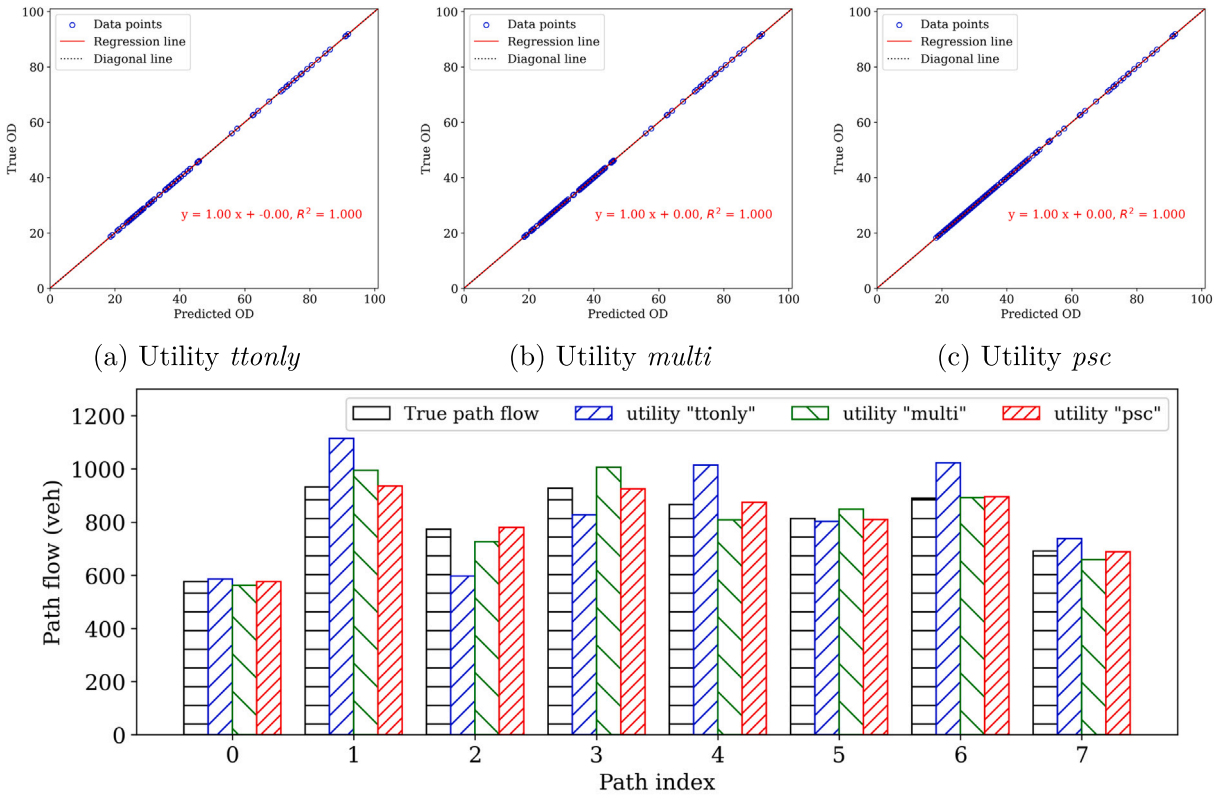
These results suggest that the LP algorithm outperforms the ESX algorithm on all counts, i.e., in terms of computational cost, path cost, and detour ratio.

4.2. Validation of the extended equivalence

As described in Section 1, MNL models for joint OD choice and doubly-constrained gravity models are mathematically equivalent (Anas, 1983) safe for aggregation errors. This equivalence serves as the foundational premise for all subsequent computations within our proposed methodology. Therefore, in this subsection, we empirically show this equivalency indeed holds in the case of joint OD-path choice. We validate the equivalency on both the Four-pairs network and Santander network.

On the Four-pairs network, we use all three different forms of the utility function (*ttonly* (33), *multi* (34) and *psc* (35)) and examine their equivalence to the resulting path flows from gravity model, and also the impact on the estimation of path flows. The equivalence under different utilities is presented in Fig. 7. As evident, a perfect fit between the path flow computed from the PSL and the one derived from the gravity model is consistently observed, regardless of the utility function employed. The results are further presented in Fig. 7, which provides a visual representation of the results, showing the comparison between the estimated path flows and the true path flows. It is evident from the plot that the *psc* function produces path flows that closely align with the true values—i.e. the values available to us in the test network data. Nonetheless, the path flows generated by the other two utility function forms also deviate no more than slightly from the true values on most paths. Based on these findings, we can tentatively conclude that the extended equivalence principle still holds given the utility function captures drivers' route choice behavior sufficiently well. The close resemblance between the path flows obtained using the *psc* utility function and the true values is likely due to the extra degree of freedom this utility function offers to accommodate drivers' preferences and behavior in this case.

On the Santander network, we also examine the extended equivalence by comparing flows generated from the gravity model and the route choice model. Based on the calibrated utility function, we iteratively adjust the a_i, b_j to balance the production and attraction totals. After four iterations, the relative error is below 0.01%, and Fig. 8 presents the regression results of the estimated and ground truth OD flows in the Santander case. The virtually perfect fit shows that, indeed, the gravity model produces an excellent approximation of the ground truth OD flows, *given the correct utility (PSL) specification is known*. This of course will not be the case in real-life.



(a) Utility *ttonly* (b) Utility *multi* (c) Utility *psc*

(d) Comparison of different utilities

Fig. 7. Extended equivalence on Four pairs network.

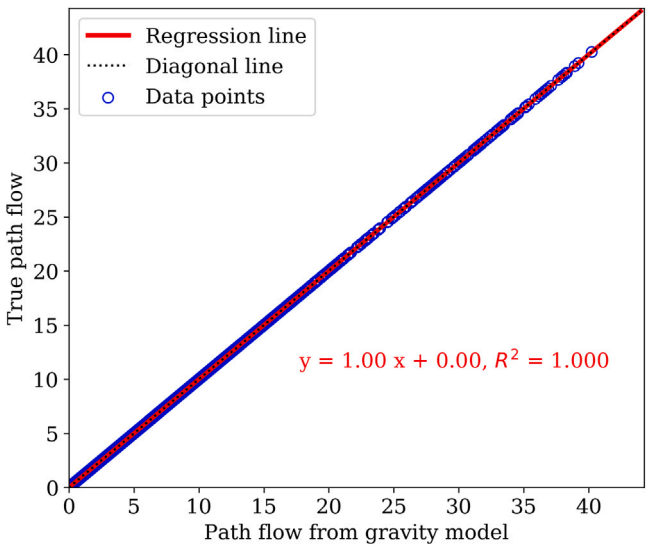


Fig. 8. Extended equivalence on Santander network.

4.3. Validation and benchmark of the overall OD estimation method

In this subsection, we examine the base case performance of the proposed method on the two simulation networks and benchmark it against a similar methods reported in the literature (Krishnakumari et al., 2020; Ma and Qian, 2018). In the section hereafter we analyze the sensitivity of our method to variations of the utility function, different demand levels and other aspects that may affect performance. We denote the base case demand level as 1.0 (100%), and utilize the PSL utility function in the "psc" form. To add some realism, we add (around 20%) noise to the sampled data with which we calibrate the PSL parameters to mimic a sampling bias in doing travel surveys.

We first discuss the base case performance of our method presented here. On the Four-pairs network, the test reveals a Root Mean Square Error (RMSE) of ≈12 vehicles per 5 min and a Mean Absolute Percentage Error (MAPE) of ≈22%, indicating a reasonably accurate estimation performance—this MAPE is in the same order as the measurement noise. To provide a visual representation of the results, we present the regression plot in Fig. 9(a), which illustrates the relationship between the estimated OD flows and the true OD flows obtained from the simulator. The plot demonstrates a satisfactory level of agreement between the estimated and true OD flows, with the majority of the points closely aligning along the diagonal line. In the four-pairs network we observe a slight overestimation of OD flows. Overall, however, the estimated OD flows closely match the true OD flows, validating the effectiveness of the estimation method in capturing the underlying traffic patterns.

The base case for the Santander network is similar to the one used in Krishnakumari et al. (2020), in which the choice set size is $N^* = 1$, implying the two shortest paths algorithms compute the same path set. In the dimensionality reduction step, we use only the first principal component, which in this simulated case explains almost 100% variance in the prior OD flow. This no coincidence, the OD matrix is generated by multiplying a static OD matrix with a time series model, so that the dynamics of all OD flows in this network are similar. This degree of predictability is an idealization. It implies that the reduced problem has only 48 decision variables, whereas the number of equations equals $48 \times (334 + 295) = 25{,}392$: a now severely over-determined problem. The consolidated solution of the reduced problem is then expanded as the final estimate.

The results are presented in Fig. 9(b). The corresponding RMSE and MAPE of the estimates are 0.14 veh/5 min and 23.5%, indicating a generally satisfactory performance for estimating dynamic OD flow in the granularity of 5 min. These correspond with the results in Krishnakumari et al. (2020). In contrast to the Four-pairs network, in the Santander case, particularly the larger OD flows appear to be slightly underestimated by our method.

We now compare our proposed method to a second data driven time-dependent origin–destination (OD) estimation method proposed in Ma and Qian (2018). Also this method uses high-granular (5 min) traffic data (link counts and speeds) and uses machine learning techniques to enhance the estimation. In Ma's method, K-shortest paths is used to generate path sets, and a Logit-based route choice model is used to map the OD flows to paths. A data-driven method for estimating the dynamic assignment ratio (DAR) is then proposed. The method solves the dynamic OD estimation problem by solving an ordinary least square (OLS) problem that minimizes the deviation between flow counts and mapped OD flows, in which the only constraint is flow non-negativity. Like our method, this method combines a wide variety of methods that formulates and solves the problem as a linear system, it requires observations only without the need for a prior OD matrix or the notion of an equilibrium assignment.

To rigorously compare methods, we conduct experiments on both the Four-Pairs network and the Santander network. Within the implementation of the benchmark method, Yen's K shortest path algorithm is applied to generate 5 shortest paths per OD pair, and the conventional Logit route choice model with path travel time as the only attribute is used to compute the path shares. Finally an iterative method is applied to solve the OLS. According to the results, the method from Ma and Qian (2018) produced estimates with RMSE of 16.26 and MAPE of 27.27% on the Four-Pairs network, which is close to but slightly worse than our proposed method. The advantage of our method seems to come from the usage of path sets with less overlap and a slightly better "assignment" (a utility function that better matches the one that generated the data). On the Santander network, the differences are significant: without the usage of PCA, the OLS method now has to solve a severely underdetermined linear system: there are $295 + 334$ independent equations for no less than 13,689 unknowns. In this case, this leads to overestimation of the OD flows, whereas our method produces a reasonable estimation. The comparison is illustrated in Fig. 9.

Not unexpectedly, these results suggest that particularly a dimension reduction technique is critically important to reliably estimate OD matrices using a data driven method. Our approach in this sense is robust: it generates a reliable prior OD flow based on the extended equivalence, and the use of PCA subsequently enables us to solve even a severely underdetermined problem on large-scale networks.

4.4. Sensitivity analysis

Based on the performance presented in base case experiments, in this section, we analyze the OD estimation performance by varying factors such as demand levels, and critical variables within each methodological component, including shortest path algorithm parameter, utility configuration and calibration noise in PSL, PCA usage and number of components to be used.

First, on the Four-pairs network, we assess the impact of different PSL utility function forms and demand levels on the estimation accuracy. This analysis examines the variations in estimation error while jointly considering different utility function forms with varying numbers of attributes and three different demand levels. The results of the sensitivity analysis are presented in Fig. 10, which consists of three sub-figures, each representing one demand level. The demand level in base case is level 2, and the level 1 and 3 case is formed by multiplying the demand curve in Fig. 4(b) by 0.75 and 1.25, respectively. From Fig. 10 we observe that the choice of utility function form has a minor impact on the estimation error, with the "psc" utility function form performing

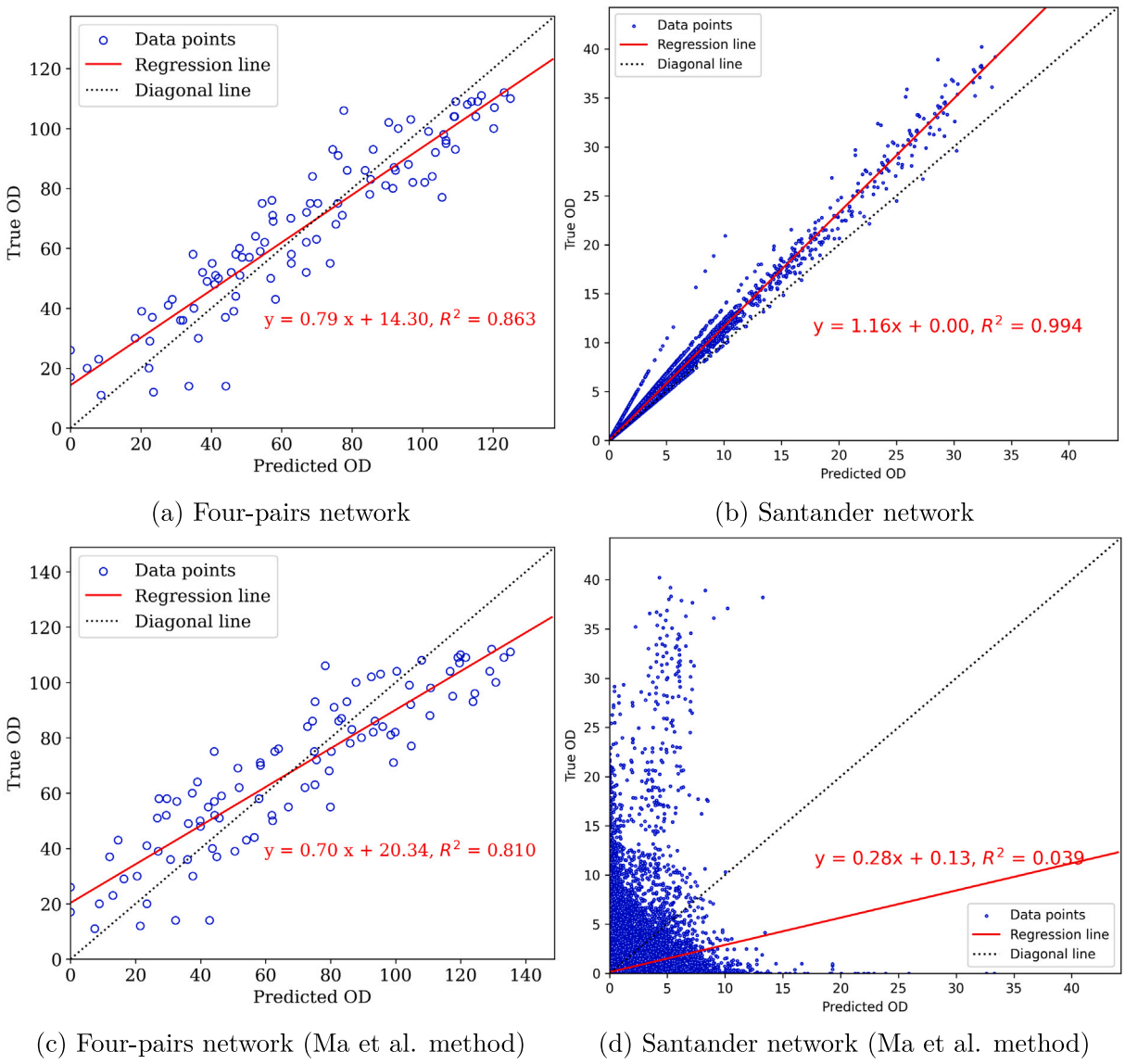


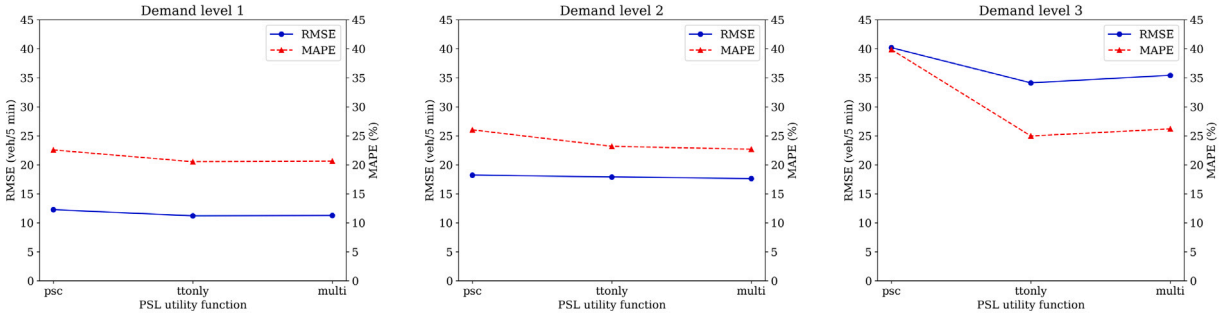
Fig. 9. Base case regression plots on the simulated networks.

slightly better than the other two forms, "ttonly" and "multi". However, the overall differences in estimation error among the utility function forms are relatively small. As the demand level increases, the estimation error exhibits a mild and generally linear increase, indicating a positive correlation between demand level and estimation error.

The differences in estimation error between the utility function forms become more pronounced as the demand level rises. Under the highest demand level of 3 (factor 1.25), where a bottleneck link in the (Four-pairs) network causes congestion and queue build-up, the estimation performance remains stable and satisfactory. The estimation error, as measured by MAPE, ranges from 25% to 40% for different utility functions.

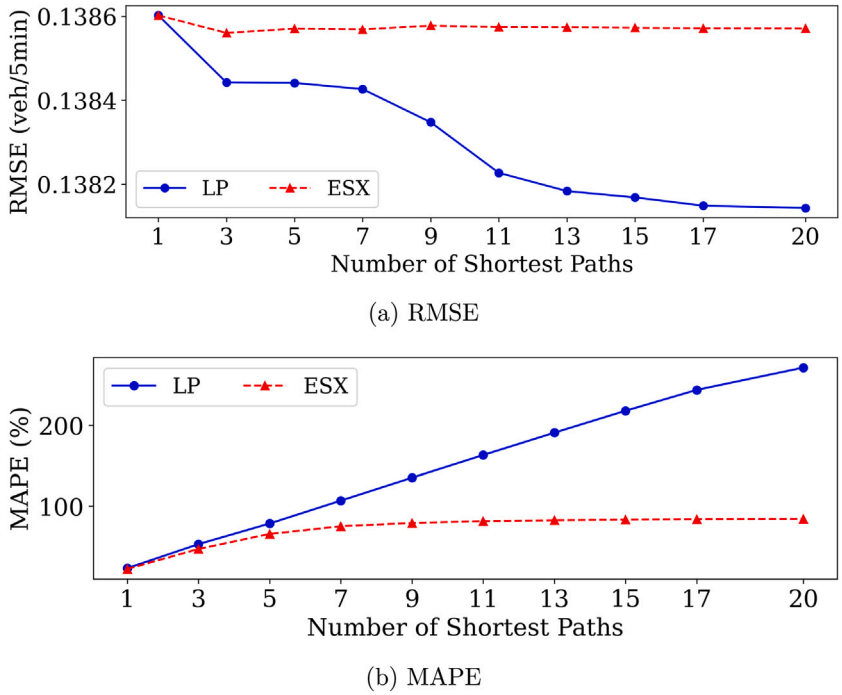
Considering the marginal differences in estimation error between the different utility function forms, we omit this test on the Santander network and focus on varying with other variables.

Specifically, on the Santander network, we compare the model performance using different shortest path algorithms (i.e., LP and ESX) as well as different sizes of the path set (i.e., different K parameter). This test is not conducted on the Four-pairs network as the path set generation is too simple there. Fig. 11 presents the errors with respect to different number of shortest paths for each algorithm. Fig. 11(a) shows that the RMSE of LP slightly decreases with more shortest paths included, whereas that of ESX remains stable. However, the vertical axis range is small, and the performance decrease in terms of RMSE for a 2000% increase in N^* (from 1 to 20) is less than 0,2%, which demonstrates its scalability. In terms of relative OD flow errors the picture is opposite: Fig. 11(b) shows that the MAPE of LP increase almost an order (10 times) over the same (20-fold) N^* increase, whereas ESX remains stable in



(a) Demand level 1 (b) Demand level 2 (c) Demand level 3

Fig. 10. Sensitivity on PSL utility function and demand level on the Four-pairs network.



(a) RMSE

(b) MAPE

Fig. 11. Comparison of different shortest path algorithms.

terms of MAPE for $N^* > 5$. This is because the ESX algorithm uses a link removal mechanism, which increasingly reduces network connectivity and naturally results in a smaller number of viable shortest paths than LP. When specifying K equals to 20 for each OD pair, the average number of paths generated per OD by ESX is 5.5, whereas that of LP is 17.1. In general, we can conclude that the two algorithm do not show significant differences in terms of estimation accuracy, which is partly due to the overdetermination. In practice, choosing either algorithm is workable from a computational perspective, so we recommend also weighing in which algorithm entails more realistic path choice behavior for a given network.

Next, we test the effect of noise in PSL parameters also on the Santander network. Fig. 12 shows the combined results in terms of two error metrics, RMSE and MAPE. Fig. 12(a) shows that the RMSE increases with the noise level (horizontal axis), whereas it fluctuates with different number of shortest paths (vertical axis). There is a clear mutual effect in that both mean and variation in RMSE over different choice set sizes are proportional to noise level. Fig. 12(b) shows that the relative error (MAPE) produces a more monotonic error surface over the two factors (noise vs number of shortest paths).

Combining the above results, we can see that the increased number of shortest paths worsens the estimation accuracy on ODs with small flows, which causes the extremely high MAPE values for combinations where RMSE values remain small. These results suggest that the method is less sensitive to the number of shortest paths than to errors in the PSL specification. This is reasonable since different numbers of shortest paths changes the coefficient matrix of an overdetermined system like the one here. In such an

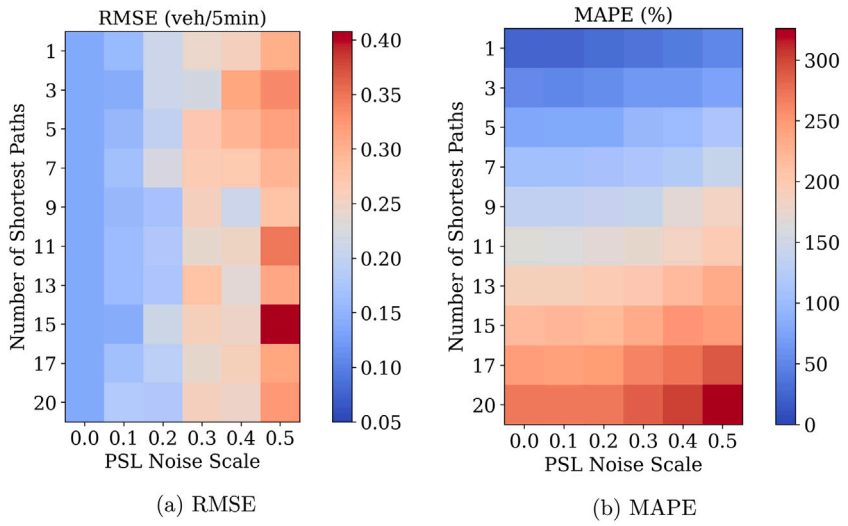


Fig. 12. Sensitivity on number of shortest path and PSL noise.

overdetermined system, the effect (whether positive or negative) of errors in path set sizes will be averaged. This leads to a fairly robust response in OD estimation quality. In contrast, the noise due in the PSL specification (i.e. the errors in our assumptions on how travelers trade off paths) is not averaged but simply added when expanding the reduced solution to a full solution.

Subsequently, we test the usage and parameters in the PCA algorithm on the Santander network. In the base case, we apply PCA on the prior OD flow (Eq. (20)) to transform the under determined problem into an over determined problem. The error induced by this direct PCA procedure can be attributed to disregarding principal components that explain only minor portions of the variance in this multivariate prior OD "signal". The fewer components we choose to retain, the more variance we potentially disregard (this depends of course on the complexity of the dynamics!).

As discussed in Section 2.3.4, in Krishnakumari et al. (2020) PCA is applied on the production and attraction time series (rather than on the prior OD), and we will refer to this procedure here as "bounding PCA". This approach differs from "direct PCA" as it does not directly reduce the rank of the equation matrix but rather provides an upper bound to constrain the solution space. The error associated with bounding PCA (i.e. an incorrect upper bound) differs from that of direct PCA. To support this claim, Fig. 13 illustrates the evidence through a cumulative plot. The x-axis represents the difference between the provided upper bound and the true value, while the y-axis represents the cumulative percentage of all dynamic OD flows. The plot clearly demonstrates that approximately 72% of the dynamic OD matrix elements are incorrectly bounded, indicated by instances where the upper bound is lower than the true value (i.e., upper bound minus true value is less than zero). Although nearly 50% of the inaccurately bounded elements exhibit errors of less than 1 veh/5 min, the cumulative effect can be significant. Under conservation constraints, errors in the upper bound will transfer to other elements and lead to a more biased estimation. Therefore, we conclude that the bounding PCA procedure computes bounds that are in many cases too tight to recover the true OD flows.

Finally, the impact of different number of principal components (1∼5) when performing PCA is also examined. As shown in Fig. 14, our two error metrics both increase when using more principal components. This is a network specific result. In the Santander case, the temporal dynamics are highly predictable, since the OD matrix is "dynamised" by imposing common seasonal patterns for all OD pairs. This implies that the added principal components add little explained variance, whereas the number of decision variables double for each additional principal component. Put differently, more principal components reduce the over determinancy of the reduced system. This naturally decreases estimation accuracy since there is more "wiggle room" to consolidate the (unnecessary additional) constraints.

So counter-intuitively, less information yields more accurate results in this specific case.

Sensitivity analyses on more realistic large(r) scale networks are needed to scrutinize the conditions under which this is valid. Such analysis must also include sensitivity to the dimensionality reduction technique itself (PCA) and its assumptions. First, as mentioned in Section 2, PCA is a linear dimension reduction technique, which may inadvertently lead to loss of relevant information. In real networks, many possible sources of non-linearities (disregarded by PCA) exist, including traffic management and control, heterogeneous demand dynamics (e.g. due to mixing of commuting, business and leasure trips, short and long trips), mode-captivity due to specific network topology, land use and demographics, and there may exist strong limitations in the observability of all these (and other) factors.

Second, PCA — due to its close relation to LS — is highly sensitive to outliers, which may exacerbate such biases. Moreover, PCA may is not a favorable dimension reduction technique for multi-variate time series with large differences in magnitude, as is the case with OD flows (few large ones, many small ones). There are, however, many possibilities to "robustify" PCA, e.g. Mateos and Giannakis (2012), and there is a wide variety of other — more sophisticated — dimension reduction techniques that could be applied (Velliangiri et al., 2019; Krishnakumari et al., 2020).

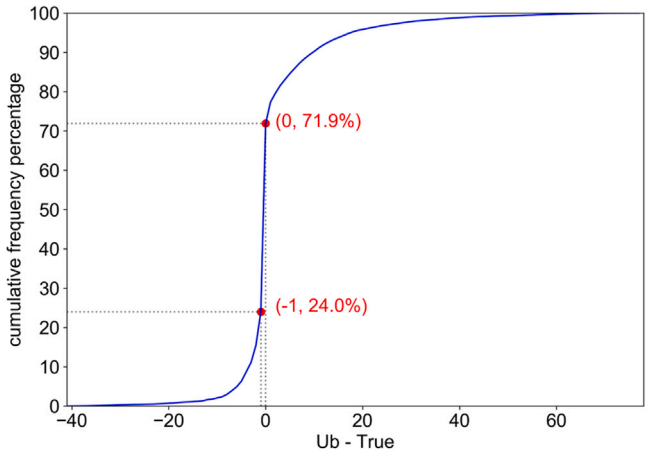


Fig. 13. Cumulative density of the difference between computed upper bound using the "bounding PCA" method, and the actual upper bound for all OD flows.

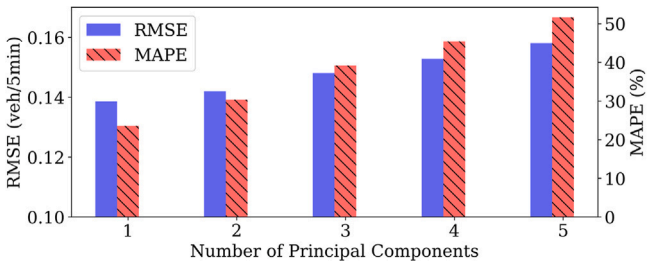


Fig. 14. Sensitivity on number of principal components.

5. Conclusions

This paper presents a novel approach to data-driven time-dependent origin–destination (OD) estimation by introducing a joint origin–destination-path choice formulation, which provides a theoretical foundation for and generalizes an earlier data driven OD estimation method. We demonstrate that the equivalency between the multinomial logit model for combined O–D choice and the doubly constrained gravity model, also extends to this combined O–D-path choice model. The advantage of this extended equivalence principle is that it allows us to combine different assimilation techniques in a single framework: (1) shortest path set computation using static and dynamic link and network properties (2) predicting a "prior OD matrix" using path-shares production and attraction totals, and (3) scaling/constraining this prior using link flows.

The paper also introduces a generic and powerful solution for OD estimation that assimilates and reconciles different data sources, allowing for the construction of sufficient constraints to estimate the target OD matrix. The solution is applicable to large networks and utilizes dimensionality reduction techniques when necessary. The performance and sensitivity of the proposed methodology are comprehensively evaluated on various networks:

- We test two K shortest path algorithms, LP and ESX, on five classical and publicly available networks of varying scales. The results indicate that LP generally scales better and outperforms ESX.
- We empirically validate the extended equivalency on two simulation networks — a toy network (Four-pairs), and a large validated network of the Spanish city Santander — thus confirming the methodology's theoretical foundation.
- On the same two networks we demonstrate that the overall OD estimation framework shows good accuracy and reliability.
- Sensitivity analysis reveals that the framework is not overly sensitive to the methodology's major assumptions, that is, the complexity of the utility function, and the number of shortest paths. This is amongst other things due to the inherent ability of the applied algorithms (ESX and LP) to determine the number of paths on an OD-specific basis.
- We finally compare two ways of applying principal component analysis — one proposed in this paper and one proposed in a previous contribution — to reduce the complexity and solve the OD estimation problem for large networks. The results suggest the new method is superior.

We see several interesting directions for further research. First, we need to validate and scrutinize all assumptions under a wider array of network sizes, topologies and characteristics, degrees of congestion, demand scenarios, and data availability scenarios, to name just a few dimensions. Second, since the path sets produced by LP and ESX shortest path algorithms produce such different

Table 3
List of Notations and Variables.

Variables	Meaning
$\mathbf{X}$	(Dynamic) OD matrix
$\tilde{\mathbf{y}}$	Observational data for inferring OD matrix
$\mathbf{y}(\mathbf{X})$	Mapping function of OD matrix to network observations
$f_1(\cdot,\cdot)$	Distance function expressing the distance of the estimated matrix to a prior OD matrix
$f_2(\cdot,\cdot)$	Distance between the traffic data observed and the data predicted by the OD matrix
A	Abstract assignment (simulation) model
θ	Parameter of assignment model depicting assumptions
$\mathcal{E}$	Set of links (edges) in the network $e_a \in \mathcal{E}$
$\mathcal{V}$	Set of nodes (vertices) in the network $v_i \in \mathcal{V}$
$\mathcal{X}$	Set of origin and destination nodes $\mathcal{X} \in \mathcal{V}$
$G(\mathcal{V},\mathcal{E})$	Directed graph with nodes (vertices) $\mathcal{V}$ and links (edges) $\mathcal{E}$
P_i	Production of origin zone i
A_j	Attraction of destination zone j
f_{ij}	Deterrence function between zone i,j
U_{ij}	Utility function between zone i,j
X_b^{ij}	Utility attribute b between zone i,j
α_b	Weight for attribute b in utility function
x_{ij}	OD flow between zone i,j
x_{ij}^k	OD flow between zone i,j during interval k
x_{ijn}^k	Flow of path n between zone i,j during interval k
a_i, b_j	Adjustment factors in doubly-constrained gravity model
β_{ij}	Fraction of travelers choosing destination j from origin i
TT_{ij}	Travel time between zone i,j
ϵ_{ij}	Gumbel distributed error in Logit model
k	Index for time periods
N_a	Number of links on path r
N_i	Number of origin zones
N_j	Number of destination zones
N_j	Number of paths between zone i,j
N_k	Number of time periods
N_p	Number of principal components used in PCA
N_x	Number of OD zones
$\mathcal{P}_{ij}^k$	Path set between zone i,j in interval k
α_0^r	Path-specific utility constant (PSC)
α_{ps}	Penalization weight for path size factor
α_w	Weight of link property attribute in utility function
α_τ	Weight of travel time attribute in utility function
$PS_{ijn}(PS_r)$	Path size factor with α_{ps} the penalization weight
l_a	Length of link e_a
L_r	Length of path r
δ_r^a	Link-path incidence variable
$\mathbf{C}, \mathbf{C}'$	Coefficient matrices of (reduced) system of equations
$\mathbf{b}, \mathbf{b}'$	Right-hand side constants of (reduced) system of equations
$\mathbf{Z}$	Principal component matrix
$\mathbf{V}$	Principal component loading matrix
$\mu_{\mathbf{X}}$	Mean vector of the original data
w_a^k, w_r^k	Link, route properties during interval k
τ_a^k, τ_r^k	Link, route travel time during interval k
u_a^k	Link speed during interval k
η^τ	Value-of-time
λ	Weight used in the ESX algorithm for link removal

detour ratio's and average costs they arguably correspond to different underlying behavioral mechanisms. For example, a possible behavioral interpretation of the ESX algorithm (which progressively prunes the network of already considered links) is that travelers consider alternative *partial* paths — or conversely, consider avoiding specific links —, rather than consider a set of full paths, as arguably is the case for the LP algorithm. To the best of our knowledge there is no evidence for either hypothesis, so this might be an interesting question to explore. Although LP is faster, both algorithms are computationally feasible for large networks.

Additionally, we could explore the idea of *dynamic* path sets (in which different choice set sizes may apply over time). Along the same lines, we could explore the effects of more diverse utility formulations. We could even explore multi-modal extensions of the method. A final methodological avenue of research lies in the dimension reduction (feature selection) techniques we use to reduce the OD solution space. In this paper we compare two alternative ways to use principal component analysis for this purpose, one on the prior OD matrix and the other on the production and attraction data time series. There are many other more sophisticated dimension reduction techniques than PCA to explore and test under different scenario's of demand dynamics and data availability. While this paper introduces and validates the core ideas and methodology using simulation data, we are committed to further examining how variations in data quality, including empirical data, will affect our method in subsequent studies.

CRediT authorship contribution statement

Yumin Cao: Writing – original draft, Visualization, Validation, Software, Methodology, Formal analysis, Data curation, Conceptualization. **Hans van Lint:** Writing – review & editing, Supervision, Funding acquisition, Conceptualization. **Panchamy Krishnakumari:** Writing – review & editing, Methodology, Conceptualization. **Michiel Bliemer:** Writing – review & editing, Conceptualization.

Declaration of competing interest

The authors declare that they have no competing interests.

Data availability

The authors do not have permission to share data.

Acknowledgments

This research is sponsored by the NWO/TTW project MiRRORS under grant agreement 16270. We thank the anonymous reviewers for their invaluable comments, which have improved this work tremendously. The debate back and forth reminded us that OD matrix estimation is a highly cross-disciplinary problem, with many ways to frame sub problems and solutions.

Appendix. Variables and notations

Throughout the paper, we use normal lower case to represent scalars and variables, and upper case for certain special variables (e.g., TT for travel time, P, A for production and attraction). Bold lower case and upper case denote vectors and matrices, respectively. Calligraphic upper case letters are used to represent sets (see Table 3).

References

Abareshi, M., Zaferanieh, M., Keramati, B., 2017. Path flow estimator in an entropy model using a nonlinear L-shaped algorithm. Netw. Spat. Econ. 17 (1), 293–315.
Aimsun, 2017. Aimsun Next 8.2 User's Manual. Aimsun SL, Barcelona, Spain.
Alexander, L., Jiang, S., Murga, M., Gonzalez, M.C., 2015. Origin-destination trips by purpose and time of day inferred from mobile phone data. Transp. Res. C 58 (Part B), 240–250.
Anas, A., 1983. Discrete choice theory, information theory and the multinomial logit and gravity models. Transp. Res. B 17 (1), 13–23.
Antoniou, C., Barcelo, 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. C 66, 79–98.
Arentze, T.A., Ettema, D., Timmermans, H.J.P., 2011. Estimating a model of dynamic activity generation based on one-day observations: Method and results. Transp. Res. B 45 (2), 447–460.
Arentze, T., Timmermans, H., 2009. A need-based model of multi-day, multi-person activity generation. Transp. Res. B 43 (2), 251–265.
Ashok, K., Ben-Akiva, M.E., 2000. Alternative approaches for real-time estimation and prediction of time-dependent origin-destination flows. Transp. Sci. 34 (1), 21–36.
Ashok, K., Ben-Akiva, M.E., 2002. Estimation and prediction of time-dependent origin-destination flows with a stochastic mapping to path flows and link flows. Transp. Sci. 36 (2), 184–198.
Behara, K.N.S., Bhaskar, A., Chung, E., 2020. A novel approach for the structural comparison of origin-destination matrices: Levenshtein distance. Transp. Res. C 111, 513–530.
Bell, M.G.H., 1991. The real time estimation of origin-destination flows in the presence of platoon dispersion. Transp. Res. B 25 (2–3), 115–125.
Ben-Akiva, M., Bierlaire, M., 1999. Discrete choice models and their applications to short term travel decisions. In: Hall, W. (Ed.), Handbook of Transportation Science. Kluwer, Dordrecht, The Netherlands, pp. 5–24.
Ben-Akiva, M.E., Gao, S., Wei, Z., Wen, Y., 2012. A dynamic traffic assignment model for highly congested urban networks. Transp. Res. C 24, 62–82.
Bhat, C., Zhao, H., 2002. The spatial analysis of activity stop generation. Transp. Res. B 36 (6), 557–575.
Bierlaire, M., Toint, P.L., 1995. Meuse: An origin-destination matrix estimator that exploits structure. Transp. Res. B 29 (1), 47–60–.
Branch, M.A., Coleman, T.F., Li, Y., 1999. A subspace, interior, and conjugate gradient method for large-scale bound-constrained minimization problems. SIAM J. Sci. Comput. 21 (1), 1–23.
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.
Cantelmo, G., Viti, F., Cipriani, E., Marialisa, N., 2015. A two-steps dynamic demand estimation approach sequentially adjusting generations and distributions. In: IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC. Vol. 2015-October, pp. 1477–1482.
Carrese, S., Cipriani, E., Mannini, L., Nigro, M., 2017. Dynamic demand estimation and prediction for traffic urban networks adopting new data sources. Transp. Res. C 81, 83–98.
Cascetta, E., 1984. Estimation of trip matrices from traffic counts and survey data: A generalized least squares estimator. Transp. Res. B 18 (4–5), 289–299.
Cascetta, E., Marquis, G., 1993. Dynamic estimators of origin-destination matrices using traffic counts. Transp. Sci. 27 (4), 363–373.
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. B 55 (Supplement C), 171–187.
Cascetta, E., Postorino, M.N., 2001. Fixed point approaches to the estimation of o/d matrices using traffic counts on congested networks. Transp. Sci. 35 (2), 134–147–.
Castillo, E., Rivas, A., Jimenez, P., Menendez, J.M., 2012. Observability in traffic networks. Plate scanning added by counting information. Transportation 39 (6), 1301–1333.

Cheng, D., Gkountouna, O., Züfle, A., Pfoser, D., Wenk, C., 2019. Shortest-path diversification through network penalization: A washington DC area case study. In: Proceedings of the 12th ACM SIGSPATIAL International Workshop on Computational Transportation Science. IWCTS '19, Association for Computing Machinery, New York, NY, USA.

Chondrogiannis, T., Bouros, P., Gamper, J., Leser, U., Blumenthal, D.B., 2020. Finding k-shortest paths with limited overlap. VLDB J. 29 (5), 1023–1047.

Cipriani, E., Nigro, M., Fusco, G., Colombaroni, C., 2014. Effectiveness of link and path information on simultaneous adjustment of dynamic O-D demand matrix. Eur. Transp. Res. Rev. 6 (2), 139–148.

Cremer, M., Keller, H., 1987. A new class of dynamic methods for the identification of Origin-Destination Flows. Transp. Res. B 21 (2), 117–132.

Djukic, T., Flotterod, G., Van Lint, H., Hoogendoorn, S., 2012a. Efficient real time OD matrix estimation based on principal component analysis. In: IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC. pp. 115–121.

Djukic, T., van Lint, J.W.C., Hoogendoorn, S.P., 2012b. Application of principal component analysis to predict dynamic origin-destination matrices. Transp. Res. Rec. (2283), 81–89.

Djukic, T., Lint, J.v., Hoogendoorn, S., 2013. Reliability assessment of dynamic OD estimation methods based on structural similarity index. In: Transportation Research Board Annual Meeting. National Academies, Washington D.C., p. 13.

Gadzinski, J., 2018. Perspectives of the use of smartphones in travel behaviour studies: Findings from a literature review and a pilot study. Transp. Res. C 88, 74–86.

Ge, Q., Fukuda, D., 2016. Updating origin-destination matrices with aggregated data of GPS traces. Transp. Res. C 69, 291–312.

Hazelton, M.L., 2001. Inference for origin-destination matrices: Estimation, prediction and reconstruction. Transp. Res. B 35 (7), 667–676.

Hazelton, M.L., 2008. Statistical inference for time varying origin-destination matrices. Transp. Res. B 42 (6), 542–552–.

Hazelton, M.L., 2010. Bayesian inference for network-based models with a linear inverse structure. Transp. Res. B 44 (5), 674–685.

Hazelton, M.L., 2015. Network tomography for integer-valued traffic. Ann. Appl. Stat. 9 (1), 474–506.

Jolliffe, I.T., 2002. Principal Component Analysis. Springer.

Kalman, R.E., 1960. A new approach to linear filtering and prediction problems. In: Transactions of the ASME Journal of Basic Engineering. Transactions of the ASME?Journal of Basic Engineering (82 (Series D)), 35–45.

Kim, J., Kurauchi, F., Uno, N., Hagihara, T., Daito, T., 2014. Using electronic toll collection data to understand traffic demand. J. Intell. Transp. Syst.: Technol. Plan. Oper. 18 (2), 190–203.

Kitamura, R., Chen, C., Pendyala, R., Narayanan, R., 2000. Micro-simulation of daily activity-travel patterns for travel demand forecasting. Transportation 27 (1), 25–51.

Krishnakumari, P., van Lint, H., Djukic, T., Cats, O., 2020. A data driven method for OD matrix estimation. Transp. Res. C 113, 38–56.

Lundgren, J.T., Peterson, A., 2008. A heuristic for the bilevel origin-destination-matrix estimation problem. Transp. Res. B 42 (4), 339–354.

Ma, W., Qian, Z.S., 2018. Statistical inference of probabilistic origin-destination demand using day-to-day traffic data. Transp. Res. C 88, 227–256.

Maher, M., 1983. Inferences on trip matrices from observations on link volumes: A Bayesian statistical approach. Transp. Res. B 17 (6), 435–447.

Mateos, G., Giannakis, G.B., 2012. Robust PCA as bilinear decomposition with outlier-sparsity regularization. IEEE Trans. Signal Process. 60 (10), 5176–5190, Export Date: 24 May 2024; Cited By: 72.

Nie, Y., Zhang, H.M., Recker, W.W., 2005. Inferring origin-destination trip matrices with a decoupled GLS path flow estimator. Transp. Res. B 39 (6), 497–518.

Okutani, I., Stephanedes, Y.J., 1984. Dynamic prediction of traffic volume through Kalman filtering theory. Transp. Res. B 18 (1), 1–11.

Rao, W., Wu, Y.J., Xia, J., Ou, J., Kluger, R., 2018. Origin-destination pattern estimation based on trajectory reconstruction using automatic license plate recognition data. Transp. Res. C 95, 29–46.

Scheffer, A., Cantelmo, G., Viti, F., 2017. Generating macroscopic, purpose-dependent trips through Monte Carlo sampling techniques. In: Transportation Research Procedia. Vol. 27, pp. 585–592.

Spiess, H., 1987. A maximum likelihood model for estimating origin-destination matrices. Transp. Res. B 21 (5), 395–412–.

Tebaldi, C., West, M., 1998. Bayesian inference on network traffic using link count data. J. Amer. Statist. Assoc. 93 (442), 557–573.

Van Der Zijpp, N., 1997. Dynamic origin-destination matrix estimation from traffic counts and automated vehicle identification data. Transp. Res. Rec.: J. Transp. Res. Board 1607 (-1), 87–94.

Van Lint, J.W.C., 2010. Empirical evaluation of new robust travel time estimation algorithms. Transp. Res. Rec. (2160), 50–59.

Van Zuylen, H.J., Willumsen, L.G., 1980. The most likely trip matrix estimated from traffic counts. Transp. Res. B 14 (3), 281–293–.

Vardi, Y., 1996. Network tomography: Estimating source-destination traffic intensities from link data. J. Amer. Statist. Assoc. 91 (433), 365–377.

Velliangiri, S., Alagumuthukrishnan, S., Thankumar joseph, S.I., 2019. A review of dimensionality reduction techniques for efficient computation. Procedia Comput. Sci. 165, 104–111.

Wei, C., Asakura, Y., 2013. A Bayesian approach to traffic estimation in stochastic user equilibrium networks. Transp. Res. C 36, 446–459.

Wu, X., Guo, J., Xian, K., Zhou, X., 2018. Hierarchical travel demand estimation using multiple data sources: A forward and backward propagation algorithmic framework on a layered computational graph. Transp. Res. C 96, 321–346.

Yang, H., Sasaki, T., Iida, Y., Asakura, Y., 1992. Estimation of origin-destination matrices from link traffic counts on congested networks. Transp. Res. B 26 (6), 417–434.

Zhou, X., Mahmassani, H.S., 2006. Dynamic origin-destination demand estimation using automatic vehicle identification data. IEEE Trans. Intell. Transp. Syst. 7 (1), 105–114.

Zhou, X., Mahmassani, H.S., 2007. A structural state space model for real-time traffic origin-destination demand estimation and prediction in a day-to-day learning framework. Transp. Res. B 41 (8), 823–840.