Improved estimation of travel demand from traffic counts by a new linearization of the network loading map

Gunnar Flötteröd and Michel Bierlaire
Transport and Mobility Laboratory (TRANSP-OR), EPFL

Abstract

In the context of dynamic traffic assignment (DTA), the network loading map represents a traffic flow model. Assuming that an iterative DTA microsimulator is given, this article discusses several techniques for the linearization of the network loading map. The practical context of this work is the problem of calibrating travel demand from traffic counts: The simulated travel demand is mapped on simulated traffic counts through the network loading map, and hence a linearization of this map provides directional information for the adjustment of the demand such that real-world traffic counts can be reproduced to a reasonable degree. The proposed linearization techniques rely on recursive regressions that are fitted to the traffic flow model during the iterations of the DTA microsimulator. It is demonstrated that this approach performs substantially better than the usually deployed proportional assignment in that it even functions in congested conditions.

1 Introduction

Iterated microsimulations have become popular solution procedures for the dynamic traffic assignment (DTA) problem. Informally, the DTA problem is to obtain consistency between a dynamic model of travel demand and a dynamic model of travel supply. The demand results from the travelers' need for mobility, and the supply is provided by the transportation system that serves these needs. When the demand exceeds the supply, congestion occurs and the performance of the transportation system deteriorates. This effect introduces the complexity in the DTA problem: The travel demand determines where congestion occurs, and the congestion induces changes in the travel demand.

Traffic microsimulations such as AIMSUN (TSS Transport Simulation Systems, 2006), DynaMIT (Ben-Akiva et al., 2001; Wen et al., 2006) or MATSim (Raney & Nagel, 2006) simulate both the demand and the supply at the individual level instead of identifying a traffic equilibrium through mathematical

operations (Peeta & Ziliaskopoulos, 2001). They typically consist of two components, which they alternately execute over many iterations: (i) The demand simulator solves the demand model by simulating the choices of all travelers in fixed network conditions, and (ii) the supply simulator solves the supply model by simulating the interactions of all vehicles in the network, given a fixed demand. That is, microsimulations resolve the mutual dependency of demand and supply through iterations (Nagel et al., 1998).

The microsimulation approach is computationally expensive. Many iterations may be needed to stabilize the demand and supply patterns, and every single iteration involves complex operations. The demand simulator is relatively easy to evaluate because it typically relies on best-response strategies (Gawron, 1998) or discrete choice models (Ben-Akiva & Bierlaire, 2003) that can be evaluated for one simulated traveler at a time. The supply simulator consumes the most computational resources because it solves the supply model jointly for all travelers (Nagel & Marchal, 2007). This fact alone already motivates the development of computationally efficient approximations of the supply simulator.

However, the linear approximations of the supply simulator that are presented in this article are motivated by the more specific problem of calibrating a disaggregate demand simulator from traffic counts (Flötteröd, 2008). Since the (also disaggregate) supply simulator maps a certain demand on a certain link flow pattern, the problem of adjusting the demand such that it becomes consistent with a set of traffic counts requires to identify how a change in the demand affects the simulated flows. The arguably most straightforward solution of this problem is to linearize the supply simulator.

A similar linearization problem is encountered in the field of dynamic origin/destination (OD) matrix estimation, where it is typically solved by a "proportional assignment" that linearly maps OD flows on link flows (Ashok, 1996; Zhou, 2004). The imprecision of a proportional assignment in congested conditions, e.g., (Yang, 1995), has motivated the development of improved linearizations (Bierlaire & Crittin, 2006; Lundgren & Peterson, 2008), which, however, have not yet been applied to the dynamic OD matrix estimation problem. The adjustment of OD matrices differs from the disaggregate demand calibration problem that motivates this work in that OD matrices are aggregate demand representations. However, the results presented here are also applicable to the estimation of OD matrices.

The proposed linearization techniques make no assumptions about the underlying supply model because they infer all required sensitivity information through recursive linear regressions. It is demonstrated that this approach

is superior to a proportional assignment in all but trivially uncongested conditions. A recursive regression model is also used successfully by Bierlaire & Crittin (2006) for the solution of the self-consistent route guidance generation problem (Bottom, 2000). An alternative way to cope with the inadequacy of a proportional assignment in congested conditions is presented by Balakrishna & Koutsopoulos (2008) for the estimation of OD matrices. This approach basically relies on a black-box optimization procedure (Spall, 1992) that minimizes the distance between the available measurements and their simulated counterparts. However, any black-box approach also prevents the exploitation of structural properties of the DTA problem.

The remainder of this article is organized as follows. Section 2 introduces some notation and terminology. Sections 3, 4, and 5 discuss three increasingly precise linearizations of the network loading map and compare them in terms of an academic example. Finally, Section 6 discusses the results and provides an outlook on future work.

2 Notation and terminology

A population of $n = 1 \dots N$ simulated travelers is considered. The activity and traveling intentions of traveler n are represented by its plan $\mathbf{u}_n$. Physically, a plan describes a round trip through the transportation network, which comprises a sequence of trips that connect intermediate stops during which activities are conducted, including all associated timing information. The first and last activity of a plan typically take place at the traveler's home location. This terminology comprises trip-based microsimulations when considering every single trip as an independent plan of an independent individual.

For the purpose of this article, it is sufficient to formally specify a plan $\mathbf{u}_n$ as a (large) vector of indicator variables $u_{i,n}(k)$ that are defined through

$$u_{i,n}(k) = \begin{cases} 1 & \text{if individual } n \text{ plans to enter link } i \text{ in time step } k \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

At a more aggregate level, the link demand $d_i(k)$ denotes the number of travelers that plan to enter link i in time step k:

$$d_i(k) = \sum_{n=1}^{N} u_{i,n}(k). \tag{2}$$

In an iterated microsimulation, the travelers "learn" the expected travel times in the network over many iterations. Consequently, the travel time information that underlies the indicators in (1) can be assumed to deviate only unsystematically from what is actually experienced when the entire population is loaded on the network.

The simultaneous execution of all plans in the network results in simulated traffic counts $q_i(k)$ on all links i in all time steps k. Collecting all link demands in the vector $\mathbf{d} = (d_i(k))$ and all link flows (simulated traffic counts) in the vector $\mathbf{q} = (q_i(k))$, a network loading constitutes a (typically nonlinear and stochastic) function

$$\mathbf{q} = Q(\mathbf{d}), \tag{3}$$

where here and in the following all vectors are column vectors. The following three sections present increasingly precise linearizations of this function that go without any further assumptions about the workings of the underlying supply simulator.

3 Proportional network loading

3.1 Specification

The notion of a proportional assignment is widely used in the context of OD matrix estimation. It expresses the assumption that any link flow $q_i(k)$ is a weighted linear superposition of all OD flows. Here, more disaggregate demand information than an OD matrix is available because a plan already contains complete route choice information (while an OD matrix is yet to be distributed on a set of routes). Carrying over the original definition of a proportional assignment to the disaggregate setting considered here results in the simple specification of what is subsequently called a proportional network loading:

$$q_i(k) = d_i(k) \qquad \forall i, k. \tag{4}$$

In order to use this model to approximate the effect of a changed demand pattern on the network conditions, a new plan is chosen for every simulated traveler $n = 1 \ldots N$, and these plans are aggregated according to (2). It is generally assumed that new/modified plans are constructed based on the same network travel times that were observed in the previous iterations of the simulation. If only uncongested scenarios are considered, the proportional network loading performs well because changing the demand does not

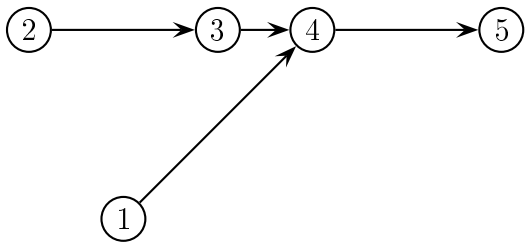

Figure 1: Small test network

change the travel times based on which the new plans are constructed. However, this model may perform very poorly in congested conditions, which is demonstrated by the following example.

3.2 Example

The limitations of the proportional network loading are clarified in terms of the example network shown in Figure 1. The network consists of five nodes and four links. The capacity of all links is 1800 veh/h. Traffic enters the network at nodes 1 and 2, and it leaves the network at node 5. All links are unidirectional. They are labeled according to their upstream and downstream node in that, e.g., link 34 goes from node 3 to node 4. The OD relation (2,5) is connected by a single route called A that consists of the node sequence 2, 3, 4, 5. The OD relation (1,5) is also connected by a single route B that is comprised of the nodes 1, 4, 5.

There are at most 1800 vehicles that could depart in either OD relation during the analysis period of one hour. In order to somewhat realistically simulate an embedding of the test network into a larger scenario, it is assumed that every simulated traveler does not make its trip with a probability of 1/3. From a more global perspective, this might be due to, e.g., the choice of a route or a departure time that results in a plan which does not cover the test network during the analysis period. Consequently, each path flow is a binomial random variable with an expectation of 1200 vehicles and a standard deviation of 20 vehicles.

Link 45 constitutes a bottleneck in that it provides a capacity of 1800 veh/h to a total average demand of 2400 veh/h. Assuming that route B has strict priority over route A, link 45 serves all flow on route B and provides the remaining capacity of 600 veh/h on average to the flow on route A. Consequently, a spillback queue builds up from node 4 upstream along route A. The vehicles within this queue proceed at a rate of 600 veh/h, which equals the

available bottleneck capacity for route A. For simplicity, only a single time interval of one hour with constant demands is considered, and the queue is assumed to dissolve in a later period of the day.

Contrarily to what is assumed in the simulation, a single flow sensor, which is located on link 34 in the real network, reports a measured flow rate of 900 veh/h. A logical explanation of this measurement is that the real demand for route B is not 1200 veh/h but only 900 veh/h, which results in a remaining bottleneck capacity of 900 veh/h for the flow on route A, which consequently crosses the sensor at this rate.

In order to compare the implications of different linearizations of the network loading map, the demands for route A and B are adjusted to the single traffic count using the Cadyts ("Calibration of dynamic traffic simulations") calibration tool (Cadyts, accessed 2009; Flötteröd, 2009). Cadyts is a freely available software package that can be linked to a broad variety of DTA microsimulators. Essentially, Cadyts combines the prior information that is represented by the simulated travel behavior with the sensor data in that it enforces a modified travel behavior in the simulation that is consistent with a Bayesian posterior choice distribution for every individual. It does so by interacting with the simulation during its iterations. This allows to observe the evolution of the calibrated simulation in the same way as one can observe the dynamics of an iterated simulation alone over the iterations. For the purposes of this work, the most relevant aspect of Cadyts is that it adjusts the demand based on directional information, which is is obtained from a linearization of the network loading map.

The measurement reproduction depends only on the simulated flow q_{34} on link 34, and the adjusted quantities are the demands for route A and B, which are denoted by D_A and D_B. (Cadyts calibrates demand at the disaggregate level. The presentation in terms of aggregate OD flows is chosen merely for simplicity.) The sensitivities of q_{34} with respect to D_A and D_B are calculated in the following way:

$$\frac{\partial q_{34}}{\partial D_A} = \frac{\partial q_{34}}{\partial d_{23}} + \frac{\partial q_{34}}{\partial d_{34}} + \frac{\partial q_{34}}{\partial d_{45}} \quad (5)$$

$$\frac{\partial q_{34}}{\partial D_B} = \frac{\partial q_{34}}{\partial d_{14}} + \frac{\partial q_{34}}{\partial d_{45}}, \quad (6)$$

which reflects the fact that a traveler who wants to take route A effectively wants to enter the links 23, 34, 45, and a traveler who wants to take route B effectively wants to enter the links 14 and 45.

An evaluation of the proportional network loading model (4) yields the sensitivity values $\partial q_{34}/\partial D_A = 1$ and $\partial q_{34}/\partial D_B = 0$, which are for completeness

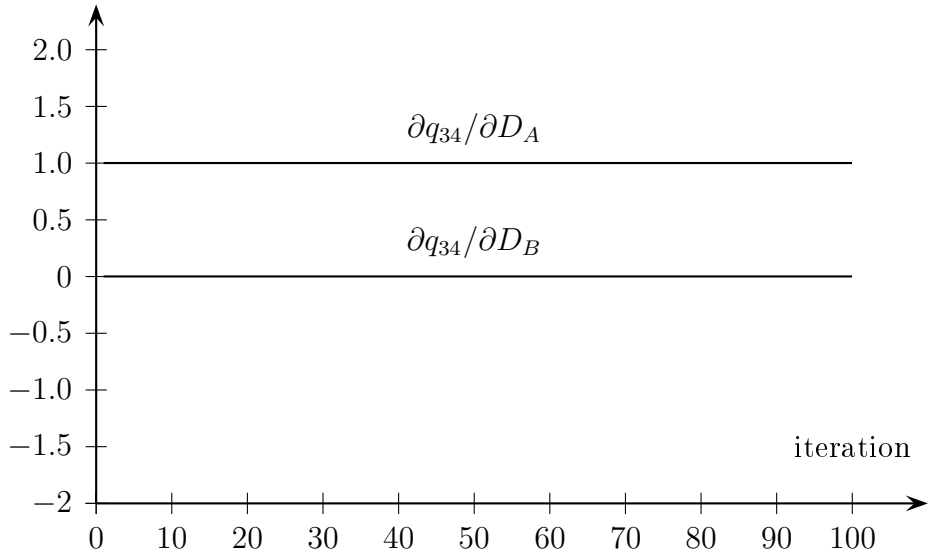

Figure 2: Sensitivities from proportional network loading

drawn over the iterations in Figure 2. These values indicate (wrongly) that the flow on link 34 is proportional to the demand for route A but insensitive to the demand for route B.

Calibration results based on this sensitivity information are given in Figure 3, which shows the estimated route demands D_A and D_B over the iterations of the simulation for 10 independent experiments. The calibration first only observes the simulation for 50 iterations and then takes affect in the subsequent 50 iterations. Right after the 50th iteration, the calibration increases D_A up to almost 1800 veh/h while leaving D_B unchanged. If the sensitivity information from the proportional network loading was right, increasing the demand for route A would indeed increase the flow on link 34, which would be consistent with the sensor data. However, the congested conditions of this scenario invalidate the assumptions of the proportional network loading and hence lead to entirely wrong results.

In the following, two approximations of the network loading map are presented that do not exhibit the unrealistic flow predictions of a proportional network loading in congested conditions.

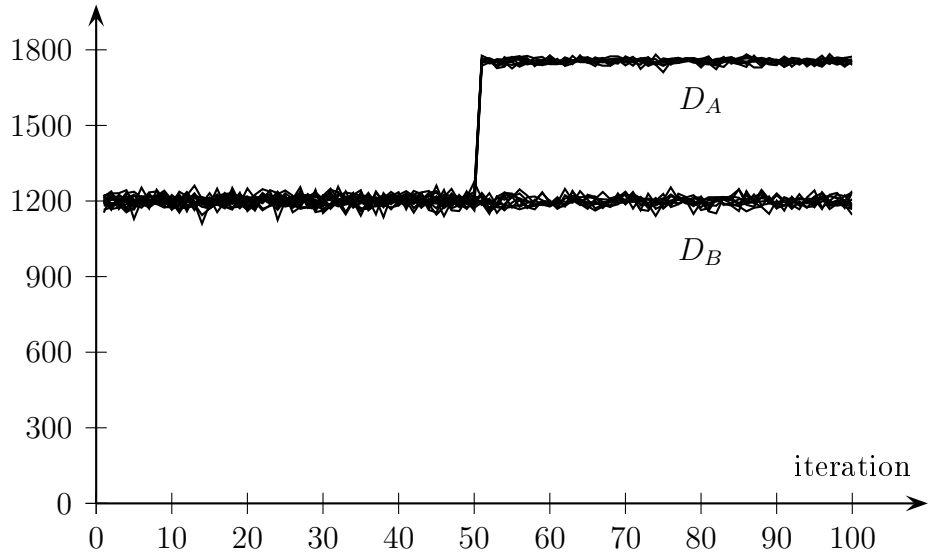

Figure 3: Estimated route demands with proportional network loading

4 Local regression

4.1 Specification

Many link demand vectors $\mathbf{d}$ and network flow vectors $\mathbf{q}$ are simulated during the iterations of a DTA microsimulation. Here, this information is used to estimate a linear regression model of the supply simulator.

The local regression model

$$q_i(k) = \alpha_i(k) + \beta_i(k) d_i(k) \tag{7}$$

specifies the flow on link i in time step k as a function of the demand for this link in this time step only. Reasonable value domains for the model coefficients are $\alpha_i(k) \geq 0$ and $0 \leq \beta_i(k) \leq 1$. This model has the same local scope as a proportional network loading, to which it collapses if one chooses $\alpha_i(k) = 0$ and $\beta_i(k) = 1$. However, choosing β values between zero and one allows to capture the link's insensitivity to the demand in congested conditions.

Given an iterative DTA simulation, the approximation (7) can be fitted by a recursive least squares procedure (Haykin, 2002) that is updated in every iteration based on the most recently simulated link demands $\mathbf{d}$ and link flows $\mathbf{q}$. In order to account for the systematically changing demand and flow patterns during the first iterations of the simulation, the recursive regression

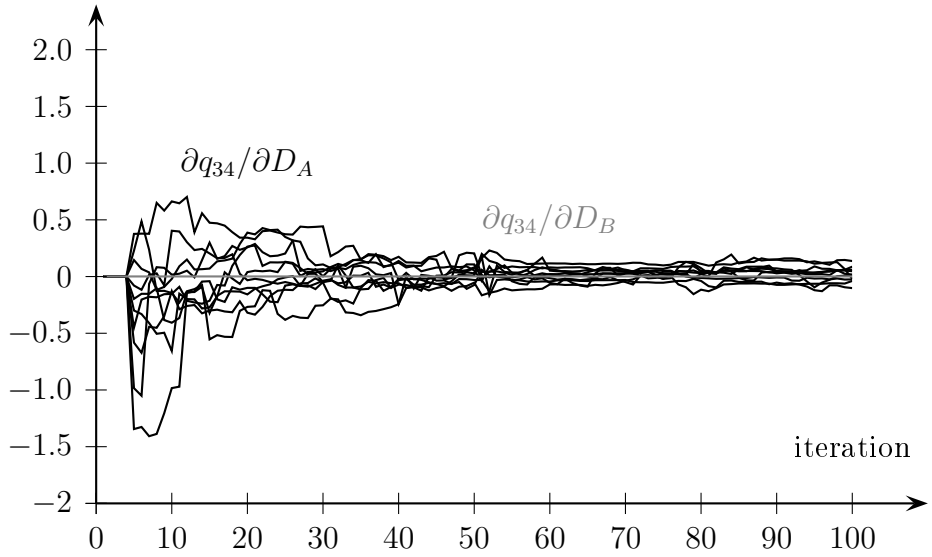

Figure 4: Sensitivities from local regression

should be configured to have tracking abilities, which is typically realized by assigning an exponentially decaying weight to samples from earlier iterations. The parameter that controls this forgetting is called the decay rate λ, which is between zero and one. The recursive regression weights a measurement that has been observed c iterations ago by the factor λ^c.

4.2 Example

Cadyts is applied to the same problem as described in Section 3.2, only that the proportional network loading is replaced by a local regression with a constant decay rate of $\lambda = 0.95$.

Figure 4 shows that the local regression identifies correctly that the demand for route A has no systematic influence on the flow on link 34. The $\partial q_{34}/\partial D_A$ curves fluctuate unsystematically around zero. An interesting effect can be observed from iteration 50 on: As soon as the calibration takes effect, the estimated sensitivities stabilize further. This can be explained by the fact that the precision of the coefficients of the regression model depends on the variability of the input data. If there is nonzero sensitivity information, the calibration reacts by accordingly adjusting the demand, which essentially acts like mechanism that "tests" the correctness of the regression parameters. By definition of the local regression, no sensitivity with respect to the flow on route B is identified: $\partial q_{34}/\partial D_B$ is exactly zero.

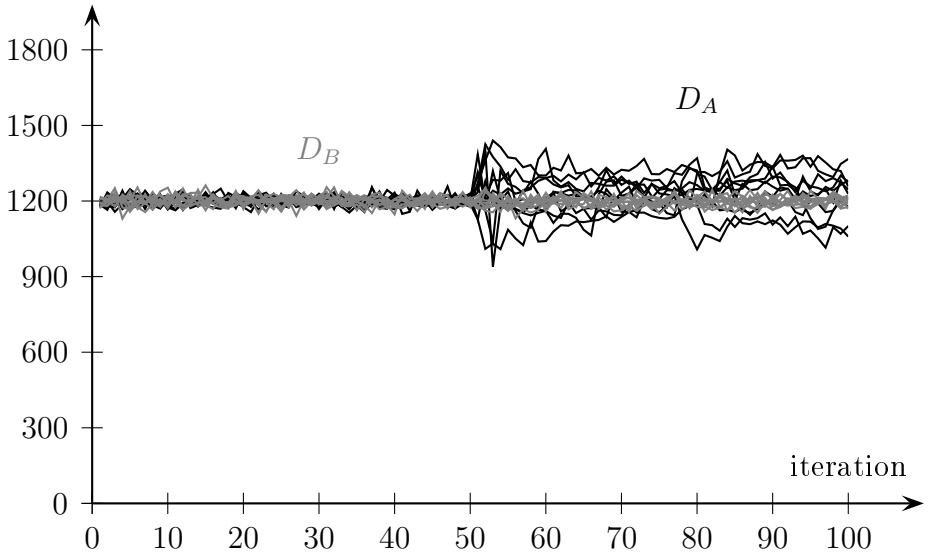

Figure 5: Estimated route demands with local regression

The resulting route demands are shown in Figure 5. Clearly, the calibration increases the variability of the demand for route A, which is a consequence of the continuous variability of the estimated sensitivities. No systematic change in the demand can be observed when the calibration is turned on. Because of its locality, the local regression is unable to reveal the true cause of the measured flow. However, at least no additional systematic error is introduced. Basically, one can state that the local regression never produces results that are worse than the simulated prior information, but it might be unable to account for the full measurement information.

5 Global regression

5.1 Specification

A straightforward yet computationally cumbersome generalization of (7) is

$$q_i(k) = \alpha_i(k) + \sum_j \beta_{ij}(k) d_j(k), \tag{8}$$

where the demand for many links j is used to explain the flow on link i. A more general specification could even account for link demands at different time steps, which, however, is omitted here for simplicity. The model (8) has two drawbacks: (i) The dimension of the model is large, and hence the

recursive regression needs many iterations to provide meaningful information. (ii) There is likely to be a lot of correlation in the link demands. For example, the demand for link 23 and link 34 in the network of Figure 1 is perfectly correlated, which implies that the respective β coefficients cannot be identified.

In order to reduce the dimension of model (8) while capturing a maximum of structural knowledge about the demand, a demand transformation based on a principal component (PC) analysis (PCA, e.g., (Shlens, 2009)) is proposed. Essentially, the simulated demand patterns of a number of iterations are observed, the PCs of the demand vector $\mathbf{d}$ are identified, and these PCs are fed into the global regression model (8) instead of the link demands. This resolves both aforementioned drawbacks in that (i) the number of PCs turns out to be relatively low, and (ii) PCs are orthogonal, which resolves the identifiability issue.

The following analysis relies on the assumption that all travelers draw their plans independently from fixed choice distributions (that may have been adjusted over many iterations such that they are consistent with an equilibrium assumption). This implies that the only correlation between two link demands d_i and d_j results from plans that contain both links. For notational simplicity, the time index is subsequently omitted.

In order to capture the covariance structure of the link demand $\mathbf{d} = (d_i)$, the number

$$d_{ij} = \sum_{n=1}^{N} u_{i,n} u_{j,n} \tag{9}$$

of simulated travelers having both link i and link j in their current plan is evaluated, where the link entry indicators u are defined in (1). The expectation of d_{ij} is

$$\mathrm{E}\{d_{ij}\} = \mu_{ij}, \tag{10}$$

and it is assumed that the variance of d_{ij} is proportional to its expectation,

$$\mathrm{VAR}\{d_{ij}\} \propto \mu_{ij}, \tag{11}$$

which is a reasonable assumption that holds exactly if the d_{ij} are Poisson distributed.

The covariance of two link demands d_i and d_j results from (i) decomposing d_i into a sum of d_{ij} and a remainder that is independent of d_j, (ii) decomposing d_j into a sum of d_{ij} and a reminder that is independent of d_i, and (iii) evaluating the covariance of these sums, which turns out to be

$$\mathrm{COV}\{d_i, d_j\} \propto \mu_{ij}. \tag{12}$$

Algorithm 1 Orthogonal iteration

1. choose orthonormal starting values $\mathbf{b}_m^0$ for $m = 1 \dots M$
2. for $k = 1, 2, \dots$
   (a) $Z^k = C \cdot [\mathbf{b}_1^{k-1}\, \mathbf{b}_2^{k-1} \cdots \mathbf{b}_M^{k-1}]$
   (b) $[\mathbf{b}_1^k\, \mathbf{b}_2^k \cdots \mathbf{b}_M^k] \cdot R^k = Z^k$ (QR factorization)

That is, the covariance matrix

$$C = \mathrm{COV}\{\mathbf{d}\} \tag{13}$$

can be identified based on a simple enumerative analysis of the plans only. However, since this matrix can in general be large, it is inconvenient to evaluate and store it explicitly. This is avoided by the procedure outlined in Algorithm 1, which finds the M largest eigenvectors (PCs) $\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_M$ of the matrix C by orthogonal iteration (Golub & van Loan, 1996).

The matrix-vector multiplications $C\mathbf{b}_m^{k-1}$ in step 2a can be conducted without an explicit representation of C based on (9) and (13): The ith element of $C\mathbf{b}_m^{k-1}$ can be written as

$$\sum_j c_{ij} b_{jm}^{k-1} = \sum_{n=1}^{N} u_{i,n} \sum_j u_{j,n} b_{jm}^{k-1} \tag{14}$$

where $C = (c_{ij})$, $\mathbf{b}_m^{k-1} = (b_{jm}^{k-1})$, and d_{ij} is used as a substitute for its expectation μ_{ij}. This is a reasonable approximation for large populations that is further improved by evaluating many plans per traveler, which are obtained from several iterations of the DTA simulation. That is, the evaluation of step 2a only requires to iterate once over the population, where for every traveler the relevant b_{jm}^{k-1} components are summed up and added to the according Z^k components. The QR factorization in step 2b is unproblematic given that only a limited number of eigenvectors is required.

Once the M largest eigenvectors $\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_M$ of $\mathrm{COV}\{\mathbf{d}\}$ are found, the global regression model

$$q_i(k) = \alpha_i(k) + \sum_{m=1}^{M} \beta_{im}(k) \langle \mathbf{d}(k) - \boldsymbol{\mu}(k), \mathbf{b}_m(k) \rangle \tag{15}$$

can be estimated by essentially the same technique as the local regression model. For simplicity, the model is still static in that the vector $\mathbf{d}(k)$ only

contains the link demands of time step k. This is reasonable because the link demands that have the greatest effect on $q_i(k)$ are likely to be near link i. $\boldsymbol{\mu}(k)$ is a vector of (estimated) average link demands at time step k, which can be tracked together with the model parameters over the iterations, and $\langle \cdot, \cdot \rangle$ denotes the inner product. The sensitivities of the link flow with respect to any link demand underline the global scope of this model:

$$\frac{\partial q_i(k)}{\partial d_j(k)} = \sum_{m=1}^{M} \beta_{im}(m) b_{jm}(k). \tag{16}$$

That is, every link that is covered by a principal demand component is put in relation with the considered flow.

5.2 Example

The test network consists of the links 14, 23, 34, 45, and the average (prior) demand for either route A and route B is 1200 veh/h. The resulting link demand correlation matrix is

$$C = \begin{bmatrix} \mu_{14,14} & \mu_{14,23} & \mu_{14,34} & \mu_{14,45} \\ \mu_{23,14} & \mu_{23,23} & \mu_{23,34} & \mu_{23,45} \\ \mu_{34,14} & \mu_{34,23} & \mu_{34,34} & \mu_{34,45} \\ \mu_{45,14} & \mu_{45,23} & \mu_{45,34} & \mu_{45,45} \end{bmatrix} = \begin{bmatrix} 1200 & 0 & 0 & 1200 \\ 0 & 1200 & 1200 & 1200 \\ 0 & 1200 & 1200 & 1200 \\ 1200 & 1200 & 1200 & 2400 \end{bmatrix}. \tag{17}$$

This matrix has only two non-zero eigenvalues, which are $\lambda_1 = 4341.64$ and $\lambda_2 = 1658.36$. The according eigenvectors are

$$\mathbf{b}_1 = [\ b_{14,1} \quad b_{23,1} \quad b_{34,1} \quad b_{45,1}\ ]^T = [\ 0.28 \quad 0.45 \quad 0.45 \quad 0.72\ ]^T \tag{18}$$

and

$$\mathbf{b}_2 = [\ b_{14,2} \quad b_{23,2} \quad b_{34,2} \quad b_{45,2}\ ]^T = [\ 0.72 \quad -0.45 \quad -0.45 \quad 0.28\ ]^T \tag{19}$$

where the superscript T denotes the transpose. These vectors are automatically identified by the calibration in the first iterations of an experiment by tracking the actual choices of every traveler and running Algorithm 1 when no new demand patterns are encountered for a while. The global regression model built for this scenario is three-dimensional because there are two principal demand components and one offset. If every link was separately considered in the regression, the model would be five-dimensional (four links and one offset). Apart from the higher dimension, the strong correlation

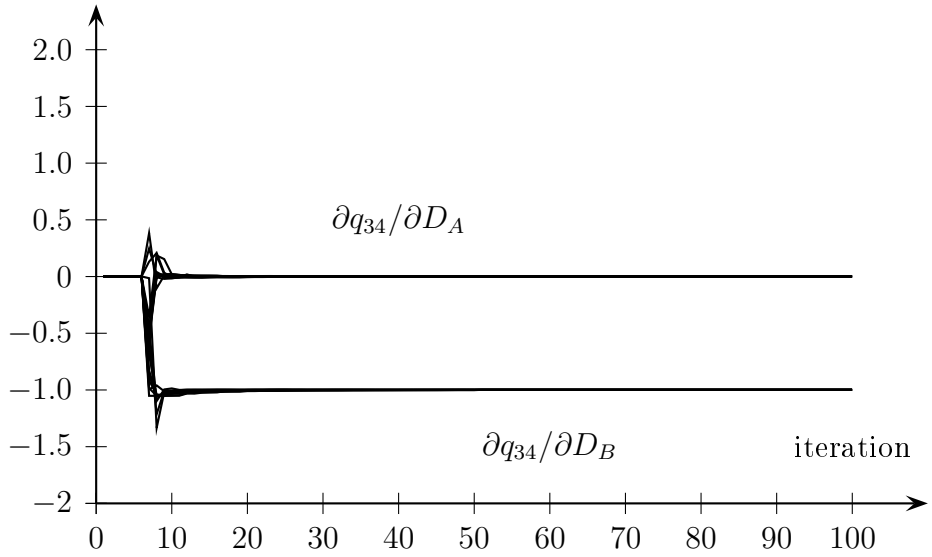

Figure 6: Sensitivities from global regression

in the link demands would render a regression-based analysis without this pre-processing a numerically ill-posed problem.

Figure 6 shows the estimated sensitivities using the global regression. After 5 iterations, the calibration decides that it has observed enough plans to build the principal demand components. After this, the global regression begins. The calibration almost immediately identifies the correct values of $\partial q_{34}/\partial D_A = 0$ and $\partial q_{34}/\partial D_B = -1$. The negative sensitivity accounts for the fact that route B has priority when entering the merge, such that an increase in route flow B proportionally decreases route flow A and hence link flow q_{34}. The very quick convergence of the sensitivities is due to the fact that the regression model operates without any noise in the signal because all variability in the link flow q_{34} is explained by the two route demands.

The estimated route demands in Figure 7 are also consistent with the real workings of the test case. When the calibration takes effect after iteration 50, it first overshoots somewhat but quickly adjusts the demand for route B from 1200 veh/h towards 900 veh/h. This makes a capacity of 900 veh/h in link 34 available for route A, and hence it generates the measured flow of 900 veh/h on link 34.

The robustness of the global regression against unexplained noise in the signal is evaluated by adding a random disturbance in every iteration to the simulated flow on link 34 before feeding it into the regression. This may reflect some variability in the merging process at node 4, which carries over to the flow on link 34 because of the spillback effect. Figures 9 to 13 show the

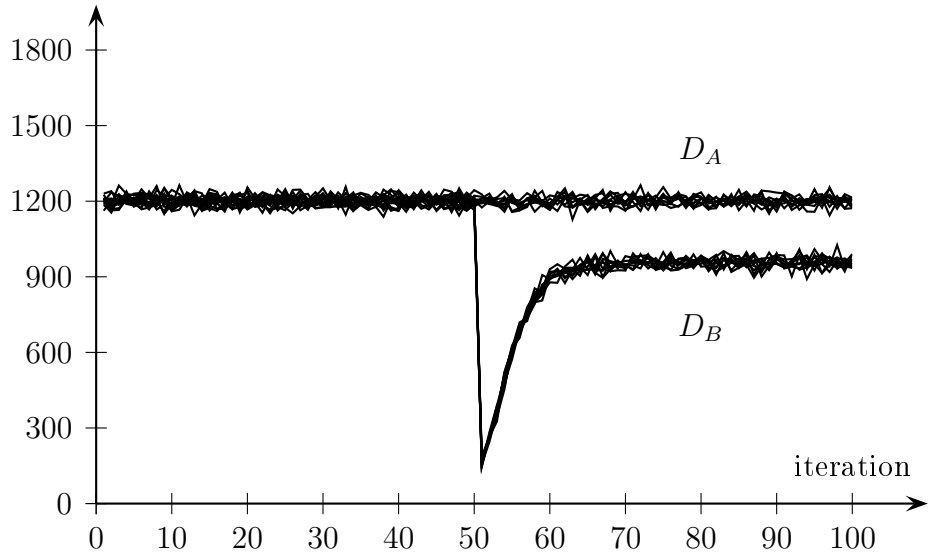

Figure 7: Estimated route demands with global regression

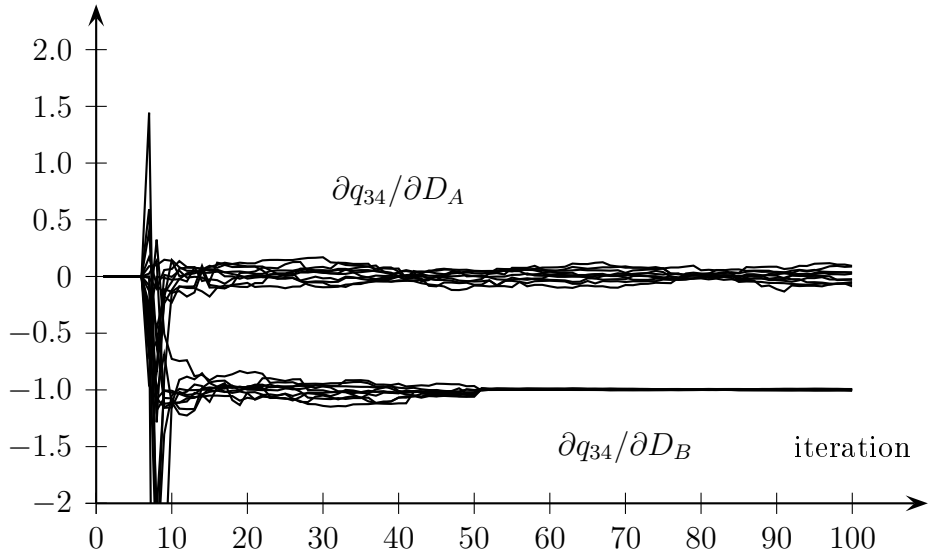

Figure 8: Sensitivities from global regression for additive noise with $\sigma = 5\,\mathrm{veh/h}$

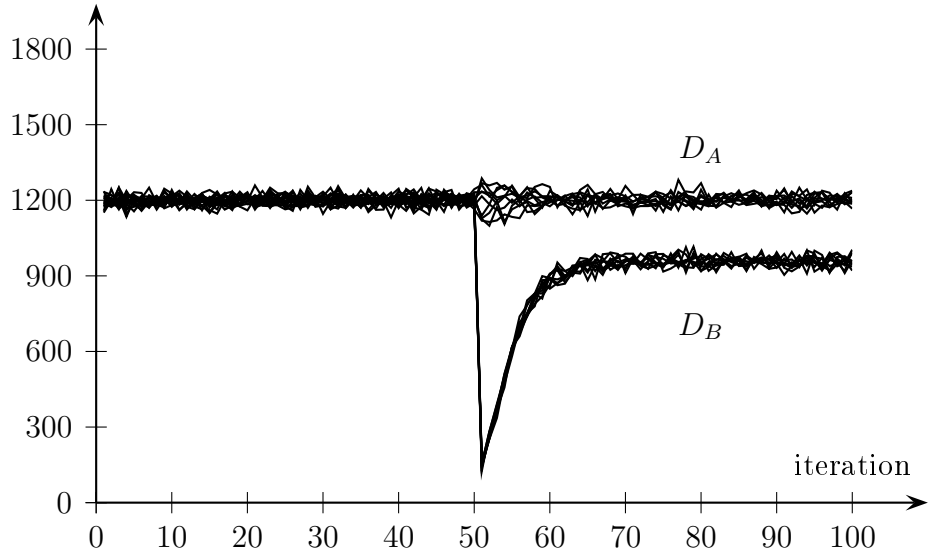

Figure 9: Estimated route demands with global regression for additive noise with $\sigma = 5\,\mathrm{veh/h}$

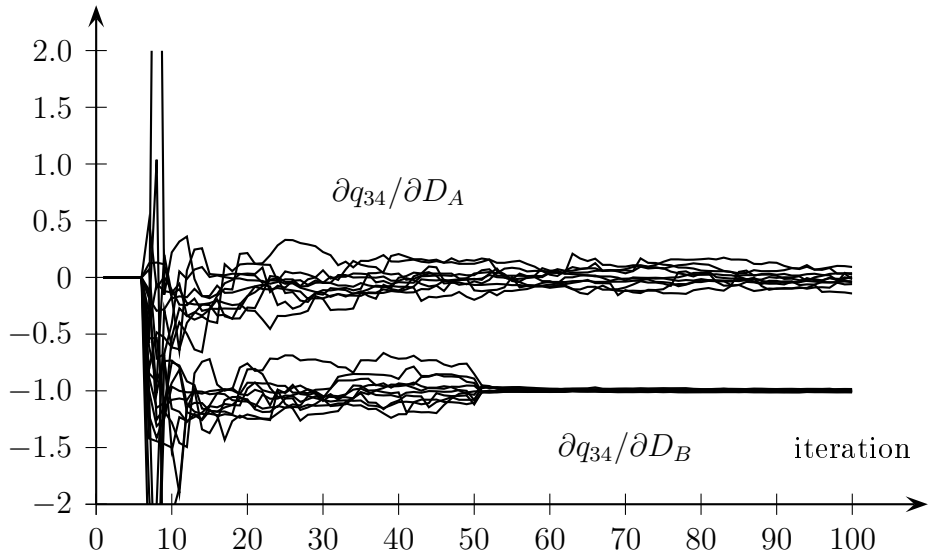

Figure 10: Sensitivities from global regression for additive noise with $\sigma = 10\,\mathrm{veh/h}$

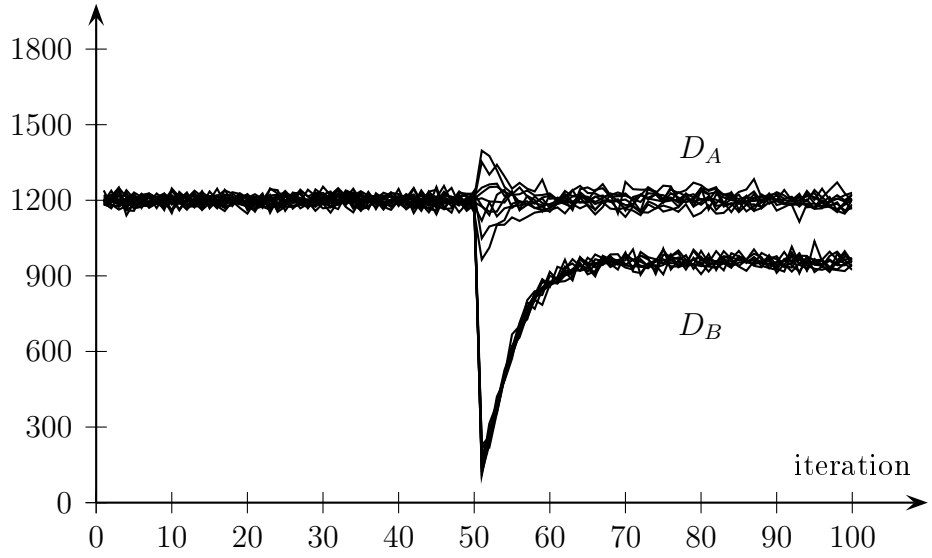

Figure 11: Estimated route demands with global regression for additive noise with $\sigma = 10\,\mathrm{veh/h}$

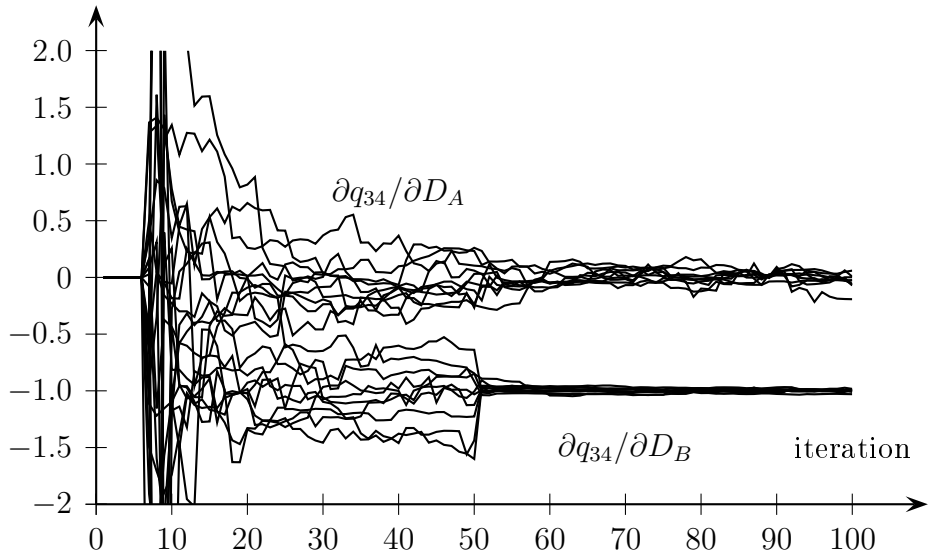

Figure 12: Sensitivities from global regression for additive noise with $\sigma = 20\,\mathrm{veh/h}$

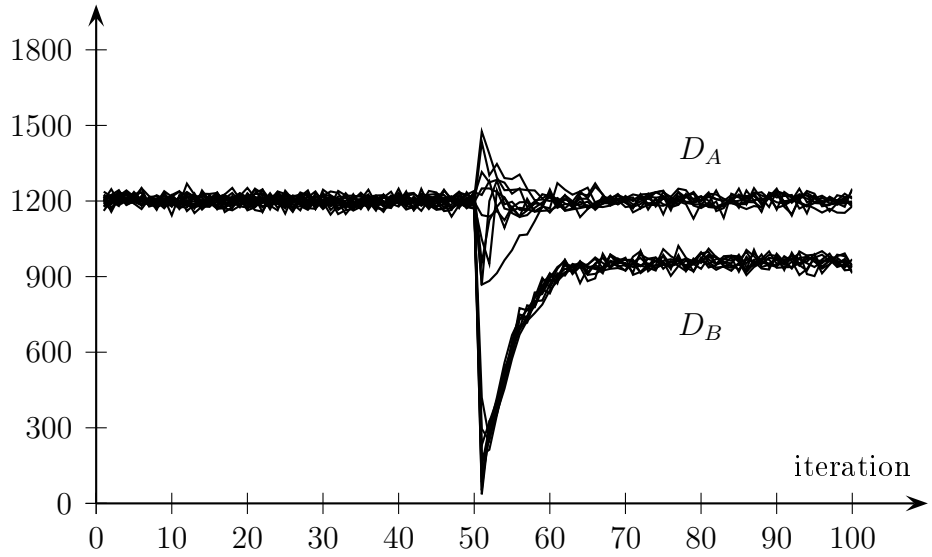

Figure 13: Estimated route demands with global regression for additive noise with $\sigma = 20\,\text{veh/h}$

estimated sensitivities and demand levels for disturbances with a standard deviation of $\sigma = 5\,\text{veh/h}$, $\sigma = 10\,\text{veh/h}$, and $\sigma = 20\,\text{veh/h}$. Clearly, the variability in the estimation increases as the noise becomes larger. However, the variability decreases even more drastically when the calibration is turned on. This confirms the hypothesis that the calibration's interaction with the simulation effectively tests the sensitivities based on which it is based such that the regression quality improves.

6 Discussion and outlook

This article presents a new, regression-based linearization of the network loading map, and it demonstrates the superiority of this approach over the usual proportional assignment, which fails in congested network conditions. Since the proportional assignment is a widely used technique in the field of OD matrix estimation, the question arises why it has been deployed for many decades with apparently substantial success. Two possible causes can be identified. First, a complete network typically contains both congested and uncongested links, and the correct adjustment of the flows on the uncongested links may compensate for the erroneous modifications of the flows on the congested links. Second, in dynamic conditions, the simulation typically starts with an empty network such that the proportional assignment

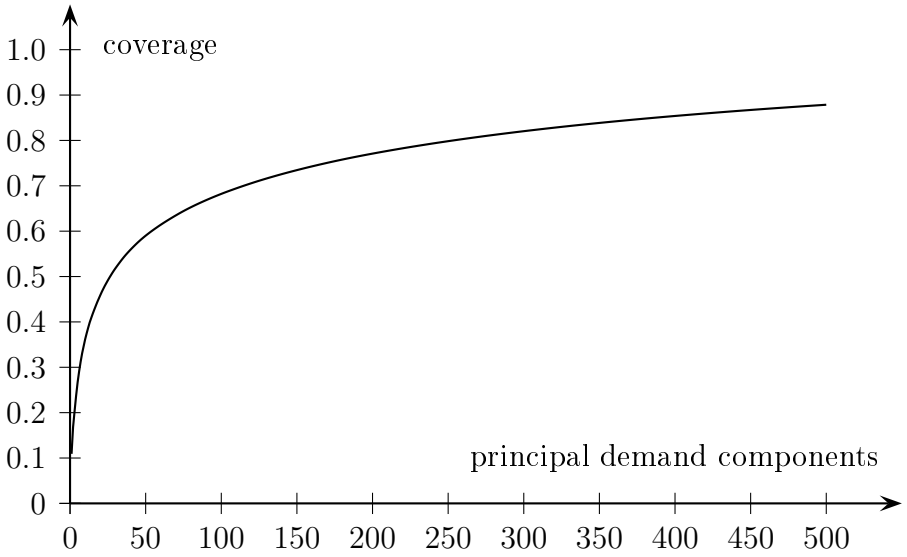

Figure 14: Coverage of demand variability during simulated Zurich morning peak

predicts well the early flows from which the congestion is eventually built up. Interestingly, this implies that a proportional assignment may perform better in dynamic conditions than in static conditions.

The presented experiments are of academic nature, and more results in more realistic settings are needed to ascertain that the method performs well in practice. Preliminary results have been obtained with a large real-world test case for the city of Zurich (Flötteröd et al., 2009, accepted for presentation), which indicate that the local regression clearly outperforms the proportional assignment. However, since the deployed microsimulation implements mechanisms that limit queue spillback in order to avoid gridlocks (Rieser & Nagel, 2008), it is yet unclear if a calibration of this particular microsimulation also benefits from the global regression approach.

Finally, an interesting further application of the principal component analysis for demand aggregation purposes is outlined. Figure 14 shows the coverage of the variability in the simulated link demands over the number of deployed principal demand components during the morning rush hour in the aforementioned Zurich test case. The figure reveals that a relatively low number of principal demand components captures a substantial portion in the link demand variability. This suggests that the use of principal demand components as aggregate demand representations may complement or even replace the usual time/dependent OD matrices in certain applications.

7 Acknowledgments

Yu Chen continuously supported the computational experiments for this article.

References

K. Ashok (1996). *Estimation and Prediction of Time-Dependent Origin-Destination Flows*. Ph.D. thesis, Massachusetts Institute of Technology.

R. Balakrishna & H. Koutsopoulos (2008). 'Incorporating within-day transitions in simultaneous offline estimation of dynamic origin-destination flows without assignment matrices'. *Transportation Research Record* **2085**:31–38.

M. Ben-Akiva & M. Bierlaire (2003). 'Discrete choice models with applications to departure time and route choice'. In R. Hall (ed.), *Handbook of Transportation Science, 2nd edition*, Operations Research and Management Science, pp. 7–38. Kluwer. ISBN:1-4020-7246-5.

M. Ben-Akiva, et al. (2001). 'Network state estimation and prediction for real–time transportation management applications'. *Networks and Spatial Economics* **1**:293–318.

M. Bierlaire & F. Crittin (2006). 'Solving noisy large scale fixed point problems and systems of nonlinear equations'. *Transportation Science* **40**(1):44–63.

J. Bottom (2000). *Consistent Anticipatory Route Guidance*. Ph.D. thesis, Massachusetts Institute of Technology.

Cadyts (accessed 2009). 'Cadyts web site'. http://transp-or2.epfl.ch/cadyts.

G. Flötteröd (2008). *Traffic State Estimation with Multi-Agent Simulations*. Ph.D. thesis, Berlin Institute of Technology, Berlin, Germany.

G. Flötteröd (2009). 'Cadyts – a free calibration tool for dynamic traffic simulations'. In *Proceedings of the 9th Swiss Transport Research Conference*, Monte Verita/Ascona.

G. Flötteröd, et al. (2009, accepted for presentation). 'Behavioral calibration of a large-scale travel behavior microsimulation'. In *Proceedings of 12th International Conference on Travel Behaviour Research*, Jaipur, India.

C. Gawron (1998). 'An iterative algorithm to determine the dynamic user equilibrium in a traffic simulation model'. *International Journal of Modern Physics C* **9**(3):393–407.

G. Golub & C. van Loan (1996). *Matrix computations*. The John Hopkins University Press. 3rd edition.

S. Haykin (2002). *Adaptive filter theory*. Prentice-Hall. 4th edition.

J. Lundgren & A. Peterson (2008). 'A heuristic for the bilevel origin-destination-matrix estimation problem'. *Transportation Research Part B* **42**(4):339–354.

K. Nagel & F. Marchal (2007). 'Computational methods for multi-agent simulations of travel behaviour'. In K. Axhausen (ed.), *Moving Through Nets: The Physical and Social Dimensions of Travel. Selected Papers from the 10th International Conference on Travel Behaviour Research*, pp. 131–188. Elsevier.

K. Nagel, et al. (1998). 'The dynamics of iterated transportation simulations'. In *Proceedings of the 3rd Triennial Symposium on Transportation Analysis*, San Juan, Puerto Rico.

S. Peeta & A. Ziliaskopoulos (2001). 'Foundations of dynamic traffic assignment: the past, the present and the future'. *Networks and Spatial Economics* **1**(3/4):233–265.

B. Raney & K. Nagel (2006). 'An improved framework for large-scale multi-agent simulations of travel behavior'. In P. Rietveld, B. Jourquin, & K. Westin (eds.), *Towards better performing European Transportation Systems*, pp. 305–347. Routledge.

M. Rieser & K. Nagel (2008). 'Network breakdown "at the edge of chaos" in multi-agent traffic simulations'. *The European Physical Journal B – Condensed Matter and Complex Systems* **63**(3):321–327.

J. Shlens (2009). 'A tutorial on principal component analysis'. Tech. rep., Center for Neural Science, New York University and Systems Neurobiology Laboratory, Salk Institute for Biological Studies.

J. Spall (1992). 'Multivariate stochastic approximation using a simultaneous pertubation gradient approximation'. *IEEE Transactions on Automatic Control* **37**(3):332–341.

TSS Transport Simulation Systems (2006). *AIMSUN 5.1 Microsimulator User's Manual Version 5.1.4*.

Y. Wen, et al. (2006). 'Online deployment of dynamic traffic assignment: architecture and run-time management'. *IEE Proceedings Intelligent Transport Systems* **153**(1):76–84.

H. Yang (1995). 'Heuristic algorithms for the bilevel origin/destination matrix estimation problem'. *Transportation Research Part B* **29**(4):231–242.

X. Zhou (2004). *Dynamic Origin-Destination Demand Estimation and Prediction for Off–Line and On–Line Dynamic Traffic Assignment Operation*. Ph.D. thesis, University of Maryland, College Park.