A fully disaggregate approach to the calibration of simulation-based DTA

Gunnar Flötteröd[1]
Berlin Institute of Technology, Germany

Abstract

We describe a novel method for the fully disaggregate calibration of a motorist demand simulator from aggregate measurements of flows, densities or velocities that are obtained at a limited set of network locations. The problem is solved in a Bayesian setting where the prior assumption about an individual's choice distribution is combined with the available measurements' likelihood into an estimated posterior choice distribution. The approach is simulation-based in that it (*i*) only requires a simulation system to represent the behavioral prior distribution, and (*ii*) only generates realizations from the behavioral posterior distribution. We focus on the offline-calibration problem in conjunction with an equilibrium-based dynamic traffic assignment system.

Keywords: demand calibration, traffic state estimation, dynamic traffic assignment

1 Introduction

The subject matter of this text is the calibration of a simulation-based dynamic traffic assignment (DTA) system from aggregate measurements of flows, densities or velocities that are obtained at a limited set of network locations. Figure 1 outlines the considered type of simulation system. It consists of a demand simulator and a supply simulator. The demand simulator maps network conditions (such as travel times) on travel behavior (such as route, destination, and departure time choice). The supply simulator models how well a road network serves a traveler's need of driving most conveniently along a route to a destination in a potentially congested traffic situation. Time-dependent Nash equilibria are computed on such models via iteration: Start with some version of time-dependent demand. Have each vehicle execute its pre-computed trips in the supply simulator. Then, re-compute the travel behavior for some fraction of travelers given the most recently observed network conditions. This procedure is iterated until an approximate fixed point is reached.

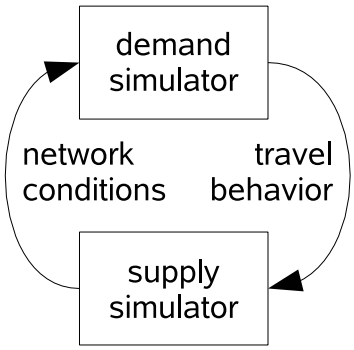

Figure 1: Simulation-based DTA

[1]This project was funded in part by the German research society DFG under the grant "State estimation for traffic simulations as coarse grained systems". Significant amounts of computing time on the computing cluster of TU Berlin's mathematical faculty are gratefully acknowledged.

In this article, we describe a method to calibrate the demand simulator. While our approach can (and should) be complemented by an additional calibration component for the supply simulator, the subsequent presentation assumes the supply simulator to be modeled without error. To serve the purpose of this work, travel behavior in terms of breaking, acceleration, and lane changing is subsumed in the physical representation of traffic flow.

The arguably most frequently adopted approach to demand calibration is origin-destination (OD) matrix estimation. An OD matrix models the demand of a given time interval in terms of number of trips from every origin to every destination of a traffic system. The originally static problem was to estimate such a matrix from observed link volumes, given a linear assignment mapping of demand on link flows. Various methods such as entropy maximization and information minimization [33], Bayesian estimation [22], generalized least squares [2, 10], and maximum likelihood estimation [31] were proposed to solve this task. Non-constant assignment mappings were incorporated by a bilevel-approach that iterates between a nonlinear assignment and a linearized estimation problem [23, 35, 36] until a fixed point of this mutual mapping is reached [13]. The combined estimation of OD matrices at subsequent time slices was demonstrated in [11], and many originally static methods were applied to dynamical problems in this vein, e.g. [1, 21, 29, 37]. The conceptual equivalence of the static and the dynamic OD matrix estimation problem was demonstrated in [9].

Since a time-dependent OD matrix maps (origin, destination, departure time) tuples on demand levels, it directly represents destination and departure time choice. A motorist OD matrix reflects mode choice at least in terms of decisions for or against the vehicular mode. Route choice, however, constitutes no additional degree of freedom but is a function of demand defined by the DTA procedure. The path flow estimators outlined below constitute a notable exception to this, yet only in a (behaviorally) static setting.

The naming "path flow estimator" is usually associated with the approach proposed in [6]. It describes a macroscopic one-step network observer that estimates static path flows from link volume measurements based on a stochastic user equilibrium modeling assumption in a congested network [3]. The estimation problem is transformed into one of smooth optimization which is iteratively solved. The model has been enhanced by multiple user classes and a simple analytical queuing model to represent traffic flow dynamics [5], and has been successfully implemented in various research and development projects [4]. The limitations associated with its original assumption of a logit path choice model ("overlapping path problem", e.g. [7]) have been mitigated by the implementation of a C-logit path choice model [12, 34]. The path flow estimator's non-stochastic user equilibrium counterpart has been proposed in [28, 30] and was further advanced in [24, 25].

The calibration of a fully disaggregate demand simulator from aggregate sensor data appears to be a novel venture. Sophisticated calibration procedures are available for random utility models (RUMs) which capture demand at the individual level [8, 32]. However, we are not aware of any research that calibrates a RUM from aggregate sensor data such as traffic counts.

The remainder of this article is organized as follows. Section 2 states the formal requirements on a DTA system to be calibrated by our methodology. Section 3 describes the

estimation framework and presents two operational estimators. Section 4 demonstrates the method's practical applicability. Finally, Section 5 concludes the article and sketches our continuative research.

2 Modeling requirements

We assume a microsimulation-based demand model and a mixed micro/macro ("mesoscopic") supply simulator to be given. Microsimulation greatly simplifies modeling issues and likewise complicates the calibration task. Consequently, every property of the simulator has to be carefully matched by a formal representation that allows for a mathematical treatment. The formalism set up in this section captures a wide variety of microscopic aspects while ensuring tractability of the mathematical estimation problem.

2.1 Macroscopic traffic flow model

A deterministic and macroscopic representation of traffic flow dynamics is required since these dynamics need to be linearized: The supply simulator maps travel demand on link volumes. Basically, an inverse mapping is needed to deduce the demand from these volumes. Since such an inversion does generally not exist, a linearization of this mapping is used and non-linearities are accounted for in an iterative manner.

For calibration purposes, the traffic flow dynamics are represented in terms of a general state space model

$$\mathbf{x}^{ms}(0) = \mathbf{x}_0^{ms} \tag{1}$$

$$\mathbf{x}^{ms}(k+1) = \mathbf{f}^{ms}[\mathbf{x}^{ms}(k), \boldsymbol{\beta}(k), k]. \tag{2}$$

The vector $\mathbf{x}^{ms}(k)$ denotes this mobility simulation's state at discrete simulation time step k. For a spatially discretized 1st order model (such as the cell-transmission model [14, 15]) this vector contains one element for every cell in the network. Single-commodity flow splits $\boldsymbol{\beta}(k) = (\beta_{ij}(k))$ from every upstream link i to every downstream link j at all intersections are exogenously provided. The vector-valued transition function $\mathbf{f}^{ms}$ defines the system's evolution through time. It fully encapsulates the specifically chosen traffic flow model. It is required that at least approximate Jacobians $\partial \mathbf{f}^{ms}[\ldots, k]/\partial \mathbf{x}^{ms}(k)$ and $\partial \mathbf{f}^{ms}[\ldots, k]/\partial \boldsymbol{\beta}(k)$ can be calculated. The handling of demand sources and sinks is described later in this section.

This state space model is supplemented with an output equation

$$\mathbf{y}(k) = \mathbf{g}[\mathbf{x}^{ms}(k), \boldsymbol{\epsilon}(k)] \tag{3}$$

that maps $\mathbf{x}^{ms}(k)$ by a linearizable function $\mathbf{g}$ on the vector $\mathbf{y}(k)$ of macroscopic observables. These may include flows, velocities, and densities generated by sensors such as inductive loops, floating cars, and traffic surveillance cameras. The influence of various sources of error on these observations is accounted for by the random disturbance vector $\boldsymbol{\epsilon}(k)$ that turns $\mathbf{y}(k)$ into a random variable itself. The resulting probability density

function (p.d.f.) of $\mathbf{y}(k)$ is

$$p(\mathbf{y}(k)|\mathbf{x}(k)) = \int \delta(\mathbf{y}(k) - \mathbf{g}[\mathbf{x}(k), \boldsymbol{\epsilon}])p(\boldsymbol{\epsilon})d\boldsymbol{\epsilon} \tag{4}$$

where δ is the Dirac function and $p(\boldsymbol{\epsilon})$ is the known p.d.f. of $\boldsymbol{\epsilon}$. Here and in the following, a lower-case p denotes a p.d.f. while an upper-case P represents a discrete probability.

An exemplary traffic flow model that meets all requirements of this subsection is described in [16, 17].

2.2 Mesoscopic supply simulation

Consider a set of **particles** $n = 1 \dots N$ (a population of travelers, agents or vehicles) moving through the network. Particles have no "mass" insofar as they do not contribute to the macroscopic occupancy of a link. At the time of a particle's entrance into the network an appropriate amount of macroscopic flow is also dismissed into the system, resulting in a mass balance between particles and total macroscopic occupancy.

The macroscopic traffic flow model is required to specify a local velocity $v_i(k)$ on every link i at every time step k. At any such time step of duration T, each particle n advances according to the local velocity of its current link. Particle locations within a link are continuous variables and particle movement is regarded as continuous in time as well: When n crosses a link boundary during a single move of duration T, it freely chooses its next link (if there is more than one downstream link) and continues with the velocity encountered there until its available move time ends. When a particle has reached its destination, it is removed from the system and an appropriate amount of macroscopic flow is also filtered out of the traffic stream passing the exit location.

The route choice of particle n is expressed by a vector $\mathbf{u}_n(k) = (u_{ij,n}(k))$ of **turning move indicators** where

$$u_{ij,n}(k) = \begin{cases} 1 & \text{if } n \text{ proceeds from link } i \text{ to } j \text{ at time step } k \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

An additional state vector $\mathbf{x}^{cnt}(k) = (x_{ij}(k))$ is introduced. Each element $x_{ij}(k)$ represents the accumulated count of particles having moved from link i to j until time step k. The dynamics of this **turning counter** $\mathbf{x}^{cnt}(k)$ are defined by

$$\mathbf{x}^{cnt}(0) = \mathbf{0} \tag{6}$$

$$\mathbf{x}^{cnt}(k+1) = \mathbf{x}^{cnt}(k) + \textstyle\sum_{n=1}^{N} \mathbf{u}_n(k). \tag{7}$$

The macroscopic flow splits $\boldsymbol{\beta}(k) = (\beta_{ij}(k))$ of the state space model (2) are now specified through

$$\beta_{ij}(\mathbf{x}^{cnt}(k)) = x_{ij}(k) / \textstyle\sum_l x_{il}(k). \tag{8}$$

This is a maximum likelihood estimator of the turning probabilities if the particle turning moves follows a multinomial distribution [19]. While the update equation (7) assumes

time-independent turning probabilities, a straightforward approach to introduce time dependency is to define an additional forgetting parameter $w \in (0, 1)$ in a modified turning counter state equation

$$\mathbf{x}^{cnt}(k+1) = w\mathbf{x}^{cnt}(k) + (1-w)\sum_{n=1}^{N} \mathbf{u}_n(k). \tag{9}$$

In the absence of newly observed turning moves, this scheme causes an exponential forgetting of previously learned counts. A useful property of this filter is its infinite memory: Even if no particles arrive at an intersection for a while, turning counts remain strictly positive and thus ensure well-defined flow splits (8). In order to avoid undefined 0/0 divisions at the beginning of a simulation, turning counters should be initialized with small positive values instead of all zeros.

A state space representation of the combined system (2) and (9) can now be given. Defining

$$\mathbf{x}(k) = \begin{bmatrix} \mathbf{x}^{ms}(k) \\ \mathbf{x}^{cnt}(k) \end{bmatrix} \tag{10}$$

and

$$\mathbf{f}[\mathbf{x}(k), \mathbf{u}_1(k) \dots \mathbf{u}_N(k), k] = \begin{bmatrix} \mathbf{f}^{ms}[\mathbf{x}^{ms}(k), \boldsymbol{\beta}(\mathbf{x}^{cnt}(k)), k] \\ w\mathbf{x}^{cnt}(k) + (1-w)\sum_{n=1}^{N} \mathbf{u}_n(k) \end{bmatrix}, \tag{11}$$

one obtains

$$\mathbf{x}(k+1) = \mathbf{f}[\mathbf{x}(k), \mathbf{u}_1(k) \dots \mathbf{u}_N(k), k]. \tag{12}$$

Given a linearizable model (2) of traffic flow dynamics, the combined state transition function $\mathbf{f}$ is likewise linearizable with respect to the macroscopic states $\mathbf{x}$ and all $\mathbf{u}_n$.

A more elaborate description of this combined micro/macro simulation logic that includes a number of exerimental results can be found in [16, 18].

2.3 Microscopic demand simulation

The decision making process of a traveler is structured according to the framework given in [8]:

1. definition of the choice problem,
2. generation of alternatives,
3. evaluation of attributes of alternatives,
4. choice,
5. implementation.

These steps are made precise in the remainder of this section. The discussion omits specific modeling assumptions and algorithmic details that would be necessary for the implementation of an applicable behavioral model. This is justified by the intention to provide a calibration procedure that is compatible with a broad range of demand simulators.

2.3.1 Definition of choice problem

The activity and traveling intentions of a motorist are denoted as her **plan**. For simplicity, only plans for a single day are considered. Physically, a plan describes a round trip through the transportation network. This round trip comprises a sequence of routes that connect intermediate stops during which activities are conducted. The first and last activity of a plan typically take place at the individual's home location.

Formally, a **(simple) route** $\mathcal{U}$ can be specified as a (physically feasible) sequence of turning moves

$$\mathcal{U} = \ldots \mathbf{u}(k-1), \mathbf{u}(k), \mathbf{u}(k+1) \ldots = \{\mathbf{u}(k)\}_k \tag{13}$$

with $\mathbf{u}(k)$ defined in (5). This notation can be extended to express all mobility related aspects of a complete plan if additional turning moves for all possible network entrances and exits are specified: The concatenation of all (network entry, route, network exit) turning move sequences for all trips in a particular plan is defined as the **(generalized) path** $\mathcal{U}$ of that plan.

Any traffic flow model of structure (12) can be steered by the turning move indicators of generalized paths instead of simple routes without formal modification. Macroscopically, a particle entry or exit merely corresponds to a local density modification the effect of which can be globally extrapolated by the linearizable state space model. Consequently, the model is also linearizable with respect to the newly introduced turning moves that represent such entries and exits. That is, the mobility simulation can linearly predict the effect of a single traveler's plan choice on the global network conditions.

2.3.2 Generation and evaluation of alternatives, choice

The **choice set** of behavioral alternatives available to decision maker (agent) n is denoted by C_n. The elements of this set are plans, formally represented by (generalized) paths $\mathcal{U}$. It is required that a non-empty choice set C_n is available to every decision maker n. Since the goal of this work is to treat the behavioral model as much as a black box as possible, it is only required that there exists a nonempty set C_n of alternatives that contains all possible choices of n in a given situation. However, an enumeration of this set is not required.

The **systematic (deterministic) utility** of an alternative, represented by a real-valued number, is a model of the benefits the decision maker expects from choosing this alternative. It reflects the decision maker's preferences. The perception of utility can vary among decision makers, and clearly utility can differ among alternatives. Formally, a systematic (deterministic) utility $V_n(\mathcal{U})$ is associated with every plan $\mathcal{U}$ in the choice set C_n of traveler n. An evaluation of this function only has to be available on request and on a per-plan basis. It is not required that the choice set is enumerated for evaluation before a choice is made. Furthermore, if the decision protocol sequentially composes a choice, e.g. by incrementally building a plan as a sequence of activities and legs, the utility function may be limited to an evaluation of the according plan components.

The choice of a plan is modeled non-deterministically. The probability that decision maker n chooses plan $\mathcal{U} \in C_n$ is denoted as $P_n(\mathcal{U})$. This **choice distribution** may be parameterized in an agent-specific way but otherwise is required to depend only on the systematic utilities of the elements in C_n. (In its general form, the calibration procedure does not even require a utility function. This is detailed in Section 3.) A probabilistic choice logic may represent randomness in human behavior or account for modeling imprecisions [8]. The specific modeling assumptions that underly a particular choice distribution are not relevant for the subsequently developed calibration approach. Furthermore, no explicit (e.g. closed-form) representation of the choice distribution is required. Only realizations of choices need to be generated by the demand simulator.

2.3.3 Implementation

The implementation of a choice requires its realization in the mobility simulation. However, a traveler with imperfect knowledge of the actual traffic conditions may observe an inconsistency between what she wants to do and what is physically possible. In particular, the generalized path representation of a plan comprises a sequence of turning move indicators that prespecify the timing of every turning move and every entry/exit move in the network. It is unlikely that (congested) traffic conditions admit precisely this timing. A generalized path does, however, implicate a logically feasible sequence of activities and routes. It therefore is assumed that the mobility simulation extracts the physically relevant information from a generalized path whenever it is stated that "$\mathcal{U}_1 \ldots \mathcal{U}_N$ are loaded on the network" or "$\mathcal{U}_1 \ldots \mathcal{U}_N$ are fed into the mobility simulation".

The specific properties of the simulation components described in this section are now exploited in the formulation and solution of a DTA calibration problem.

3 Calibration methodology

The considered problem is to use spatially and temporally incomplete sensor information to reconstruct spatially and temporally complete system state information. Macroscopically, the system state to be reconstructed is represented by the state vector sequence

$$\mathcal{X} = \{\mathbf{x}(k)\}_k \tag{14}$$

of traffic flow model (12). Since this model unfolds deterministically given an initial state and a driver population's plans $\mathcal{U}_1 \ldots \mathcal{U}_N$, the calibration problem becomes to identify control sequences $\mathcal{U}_1 \ldots \mathcal{U}_N$ that steer $\mathcal{X} = \mathcal{X}(\mathcal{U}_1 \ldots \mathcal{U}_N)$ towards most likely values given the available measurements and the behavioral a priori knowledge represented by the demand simulator.

3.1 General formulation of estimator

Aggregate measurements alone do not provide sufficient information for unique plan estimates since usually there are many behavioral combinations that generate the same

observations. Here, this problem is resolved by the incorporation of additional behavioral information in a Bayesian setting.

Consider a single iteration of a simulation-based DTA procedure. An (arbitrary) demand simulator draws choices $\mathcal{U} \in C_n$ according to an individual choice distribution $P_n(\mathcal{U})$ for every agent $n = 1 \ldots N$. Only realizations from these distributions are available. Given mutually independent traveler decisions, the **behavioral prior** for the whole population is defined as

$$P(\mathcal{U}_1 \ldots \mathcal{U}_N) = \prod_{n=1}^{N} P_n(\mathcal{U}_n). \tag{15}$$

According to (4), the measurements

$$\mathcal{Y} = \{\mathbf{y}(k)\}_k \tag{16}$$

result from a joint distribution

$$p(\mathcal{Y}|\mathcal{X}) = \prod_k p(\mathbf{y}(k)|\mathbf{x}(k)) \tag{17}$$

where stochastic independence between outputs on different time steps is assumed. This is, so far, the not unexpected result that all spatiotemporal measurements $\mathcal{Y}$ can be probabilistically described if all spatiotemporal system states $\mathcal{X}$ are known – no behavioral information is needed directly. However, since $\mathcal{X} = \mathcal{X}(\mathcal{U}_1 \ldots \mathcal{U}_N)$, the **likelihood** of a particular plan choice combination $\mathcal{U}_1 \ldots \mathcal{U}_N$ is

$$p(\mathcal{Y}|\mathcal{U}_1 \ldots \mathcal{U}_N) = p(\mathcal{Y}|\mathcal{X}(\mathcal{U}_1 \ldots \mathcal{U}_N)). \tag{18}$$

Bayes' theorem allows to combine the behavioral prior and the likelihood into a **behavioral posterior**

$$P(\mathcal{U}_1 \ldots \mathcal{U}_N|\mathcal{Y}) = \text{const} \cdot p(\mathcal{Y}|\mathcal{U}_1 \ldots \mathcal{U}_N) P(\mathcal{U}_1 \ldots \mathcal{U}_N). \tag{19}$$

The estimation objective is to make the population choose its plans according to the posterior (19) instead of the prior (15). This can be enforced if draws are taken from the prior but are rejected with a certain probability that depends on the measurements. Denote by $\phi(\mathcal{U}_1 \ldots \mathcal{U}_N)$ the probability to accept a draw $\mathcal{U}_1 \ldots \mathcal{U}_N$ from the prior. If this probability is specified as

$$\phi(\mathcal{U}_1 \ldots \mathcal{U}_N) = p(\mathcal{Y}|\mathcal{U}_1 \ldots \mathcal{U}_N)/D, \tag{20}$$

$$D \geq \max_{\mathcal{V}_1 \in C_1 \ldots \mathcal{V}_N \in C_N} p(\mathcal{Y}|\mathcal{U}_1 \ldots \mathcal{U}_N), \tag{21}$$

then the following accept/reject procedure draws from the posterior, as can be shown by straightforward manipulations.

1. Draw candidate choices $\mathcal{U}_1 \ldots \mathcal{U}_N$ from the prior (15).
2. With probability $1 - \phi(\mathcal{U}_1 \ldots \mathcal{U}_N)$, discard the candidates and goto 1.
3. The first accepted plans $\mathcal{U}_1 \ldots \mathcal{U}_N$ constitute a draw from the posterior (19).

The behavioral posterior can thus be generated by suppressing certain draws from the prior. Somewhat coarsely expressed: (*i*) The simulation is run many times with different random seeds, (*ii*) a large portion of these runs is "thrown away", based on the above

rejection criterion, and (*iii*) the remaining runs are draws from an accurate Bayesian combination of the behavioral prior and the measurements. Although appealing because of its simplicity, this approach is in this form computationally intractable in all but trivial cases. There are two major problems.

1. It is computationally infeasible to evaluate all possible $p(\mathcal{Y}|\mathcal{U}_1 \ldots \mathcal{U}_N)$ values beforehand since every such evaluation requires a full network loading in order to map $\mathcal{U}_1 \ldots \mathcal{U}_N$ on a macroscopic state sequence $\mathcal{X}$ that enters the likelihood via (18). However, these evaluations are required in order to guarantee a feasible denominator (21) for the acceptance probabilities. Furthermore, the need for a choice set enumeration implies that the estimation logic is aware of this set, which constitutes an unwanted dependency of the estimator on modeling details.
2. Even if the acceptance probabilities' denominator was replaced by an estimate in order to mitigate problem 1, a single draw from the posterior might still require a substantial number of mobility simulation runs since every draw from the prior needs to be loaded on the network at least once, and since it cannot be guaranteed that an "accept" occurs after a fixed number of draws from the prior.

In light of these difficulties, simplifying assumptions that speed up the simulation of the posterior are highly desirable even at the cost of some loss in accuracy. Two such simplifications are proposed in the following.

3.2 Operational accept/reject estimator

The estimation problem is considerably simplified if the full likelihood is replaced by an approximation. Appendix A derives the following linearization of the log-likelihood $\ln p(\mathcal{Y}|\mathcal{U}_1 \ldots \mathcal{U}_N)$ with respect to the plans $\mathcal{U}_1 \ldots \mathcal{U}_N$:

$$\ln p(\mathcal{Y}|\mathcal{U}_1 \ldots \mathcal{U}_N) \approx \sum_{n=1}^{N} \langle \Lambda, \mathcal{U}_n \rangle + \text{const} \tag{22}$$

where the "inner product" $\langle \Lambda, \mathcal{U}_n \rangle$ is defined as

$$\langle \Lambda, \mathcal{U}_n \rangle = \sum_k \sum_{ij} \lambda_{ij}(k) u_{n,ij}(k). \tag{23}$$

The time-dependent λ coefficients represent the sensitivities of the log-likelihood with respect to the corresponding turning move indicators. These coefficients are identical for all agents. Subsequently, Λ will be used as a collective term for all λ coefficients. The linearized log-likelihood implies the following likelihood approximation:

$$p(\mathcal{Y}|\mathcal{U}_1 \ldots \mathcal{U}_N) \approx \text{const} \cdot \prod_{n=1}^{N} e^{\langle \Lambda, \mathcal{U}_n \rangle}. \tag{24}$$

Substitution of this in the behavioral posterior (19) yields

$$P(\mathcal{U}_1 \ldots \mathcal{U}_N|\mathcal{Y}) \approx \text{const} \cdot \prod_{n=1}^{N} e^{\langle \Lambda, \mathcal{U}_n \rangle} P_n(\mathcal{U}_n). \tag{25}$$

The benefits of the linearization are twofold. First, the population's joint posterior (25) is decomposed into a product of individual posteriors that can be evaluated agent by agent. These individual-level posteriors are subsequently denoted by

$$P_n(\mathcal{U}|\mathcal{Y}) = \text{const} \cdot e^{\langle \Lambda, \mathcal{U} \rangle} P_n(\mathcal{U}). \tag{26}$$

Algorithm 1 Accept/reject estimator

1. Initialization.
   (a) Set iteration counter $m = 0$.
   (b) Fill $\bar{\Lambda}^{(m)}$ (estimate of Λ fixed point) with all zeros.
2. One iteration of simulation-based DTA (plus calibration procedure).
   (a) For all $n = 1 \ldots N$:
      i. Draw candidate choice $\mathcal{U}_n^{(m)}$ from n's behavioral prior.
      ii. Evaluate acceptance probability $\phi_n(\mathcal{U}_n^{(m)})$ based on $\bar{\Lambda}^{(m)}$ as defined in (27). With probability $1 - \phi_n(\mathcal{U}_n^{(m)})$, discard the candidate and goto 2(a)i.
      iii. Retain the first accepted choice $\mathcal{U}_n^{(m)}$.
   (b) Load $\mathcal{U}_1^{(m)} \ldots \mathcal{U}_N^{(m)}$ on the network and obtain $\mathcal{X}^{(m)}$.
   (c) Linearize $\ln p(\mathcal{Y}|\mathcal{U}_1 \ldots \mathcal{U}_N)$ and obtain $\Lambda^{(m)}$.
   (d) Update $\bar{\Lambda}^{(m+1)} = (m\bar{\Lambda}^{(m)} + \Lambda^{(m)})/(m+1)$.
3. If another iteration is desired:
   (a) Increase m by one.
   (b) Goto step 2.

Second, a single run of the mobility simulation (plus one calculation of the Λ coefficients) is sufficient to parameterize these posteriors for all agents in the population.

The accept/reject procedure can now be applied to every decision maker individually. The acceptance probability for plan $\mathcal{U}$ from n's choice set is defined as

$$\phi_n(\mathcal{U}) = e^{\langle \Lambda, \mathcal{U} \rangle} / D_n \tag{27}$$

$$D_n \geq \max_{\mathcal{V} \in C_n} e^{\langle \Lambda, \mathcal{V} \rangle} \tag{28}$$

but otherwise the method remains unchanged. This approach is subsequently denoted as the **accept/reject (AR) estimator**. The only simplifying assumption made here is that the log-likelihood can be linearized with sufficient precision. Since this linearization is likely to be different given the network conditions that result either from the behavioral prior or the posterior, an iterative approach is appropriate: Starting from the behavioral prior, successively improved linearizations are generated from iteration to iteration until a stable state is reached where the estimator draws from the behavioral posterior based on a linearization that in turn is most appropriate given this very posterior. That is, a fixed point of the Λ coefficients is sought after. Here, the existence of such a fixed point is merely assumed and an elementary stochastic approximation method is employed for its identification.

The AR estimator is summarized in Algorithm 1. The behavioral prior implemented by the demand simulator is arbitrary. Since a choice set enumeration is only required to provide a lower bound for the acceptance probabilities' denominator defined in (28), it can be avoided if this denominator is treated as a tuning parameter. Choosing a large value is likely to comply with the (unknown) lower bound but also to result in low acceptance probabilities and increased computational cost. Vice versa, a smaller denominator yields faster but also increasingly imprecise estimates. A computationally more efficient yet not as broadly applicable estimator is presented in the next section.

3.3 Operational utility-modification estimator

This estimator assumes a particular prior choice distribution

$$P_n(\mathcal{U}) \propto s_n(\mathcal{U}) e^{\mu V_n(\mathcal{U})} \tag{29}$$

which is a multinomial logit model with positive scale parameter μ and a likewise positive plan-specific probability scaling function $s_n(\mathcal{U})$. Substituting this prior into the behavioral posterior (26) for a single decision maker yields

$$P_n(\mathcal{U}|\mathcal{Y}) \propto s_n(\mathcal{U}) e^{\mu(V_n(\mathcal{U}) + \langle \Lambda, \mathcal{U} \rangle / \mu)}. \tag{30}$$

This posterior is structurally identical to its prior. Only the addition of $\langle \Lambda, \mathcal{U} \rangle / \mu$ to $V_n(\mathcal{U})$ is different. This allows to force a demand simulator that implements (29) to immediately draw from the posterior only by adding a correction term $\langle \Lambda, \mathcal{U} \rangle / \mu$ to every alternative $\mathcal{U}$'s systematic utility. For this, the $s_n(\cdot)$ coefficients need not be known by the estimator.

This approach is called the **utility-modification (UM) estimator**. Its requirements are more restrictive than those of the AR estimator since the demand simulator is required to implement (29). However, if this prior is given, the UM estimator and the AR estimator yield equivalent results since both rely on the same linearization-based approximation (26) of the posterior. In this case, the UM estimator is to be preferred over the AR estimator since it is computationally more efficient in that it rejects no draws from the prior but immediately draws from the posterior. The UM estimator follows the same logic as outlined in Algorithm 1, only that steps 2(a)i to 2(a)iii need to be replaced by an appropriate utility-modification logic.

Technically, the UM estimator can be applied in conjunction with an arbitrary utility-driven demand simulator. The following analysis identifies the conceptual limitations of such an approach. Assume that decision maker n disposes of a choice set C_n and that prespecified utilities $V_n^0(\mathcal{U})$ for every $\mathcal{U} \in C_n$ are given. Based on these utilities, the arbitrary demand simulator draws from well-defined but to the estimator unknown choice probabilities $P_n^0(\mathcal{U})$. These choice probabilities can be perfectly reproduced by the model (29) if the $s_n(\cdot)$ coefficients are re-defined as

$$s_n(\mathcal{U}) = P_n^0(\mathcal{U}) / e^{\mu V_n^0(\mathcal{U})}. \tag{31}$$

The resulting choice probabilities are

$$P_n(\mathcal{U}) \propto P_n^0(\mathcal{U}) e^{\mu(V_n(\mathcal{U}) - V_n^0(\mathcal{U}))} \tag{32}$$

such that $V_n(\mathcal{U}) = V_n^0(\mathcal{U})$ results in $P_n(\mathcal{U}) = P_n^0(\mathcal{U})$ for all $\mathcal{U} \in C_n$. Loosely speaking, any behavioral prior can be approximated up to 0th order in this way. The adequacy of this approximation for others than the prespecified utilities only depends on the approximated prior's elasticities, i.e. the way relative utility changes induce relative changes in the choice probabilities.

Recall that the UM estimator functions without explicit knowledge of the $s_n(\cdot)$ coefficients. This implies that an application of the UM estimator can be justified by the

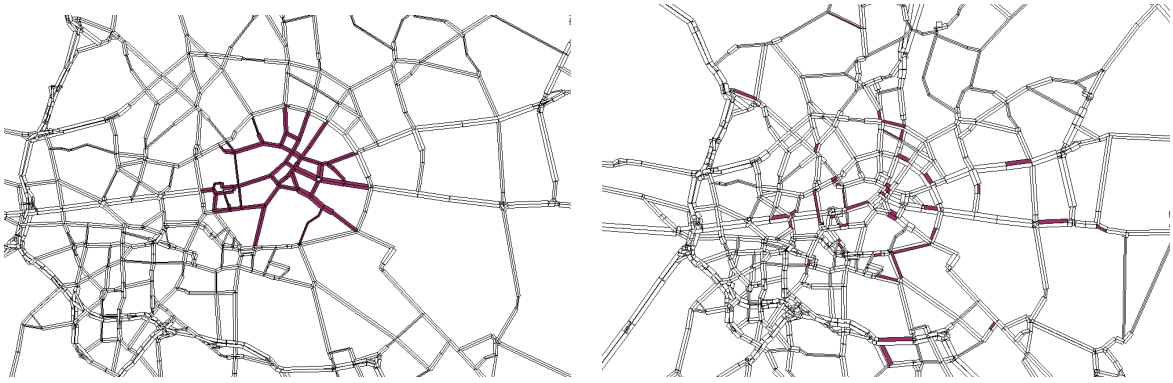

Figure 2: Berlin network. Left: toll zone, right: sensor locations.

approximation (32) even if the P_n^0 and V_n^0 values that define the $s_n(\cdot)$ coefficients in (31) are unknown. However, it is required that the prior choice distribution's elasticities are sufficiently similar to those of (29). Otherwise, only a heuristic application of the UM estimator is possible.

Summarizing, this section derives two operational procedures for drawing from a Bayesian posterior choice distribution that results from the combination of aggregate sensor data and a general demand simulator. The computational efficiency of these procedures results from a linearization of the log-likelihood function that allows to decompose the population's joint posterior choice distribution at the level of individual decision makers.

4 Test case

This section outlines a number of experimental results. A detailed description can be found in [16]. The considered test case comprises the metropolitan region of Greater Berlin. The underlying network has 2459 links and 1083 nodes. A synthetic population of 206'353 travelers with complete activity plans is available for this scenario [27]. All experiments are constrained to the time span from 6 to 9 am. This interval exhibits the most variable traffic conditions because of the morning rush hour.

4.1 Simulation procedure

The only behavioral degree of freedom considered here is route choice. That is, all behavioral aspects apart from route choice are retained unchanged in the original plans. Since route choice can be generalized to plan choice by minor modifications to the original network as shown in Section 2.3, an effective route choice estimator is likely to be applicable in a more general setting as well.

A time-independent toll of 0.24 EUR/km is charged on all link in the city center shown in Figure 2 (left), and no toll is charged outside of this area. The unitless utility of a route $\mathcal{U}$ is

$$V_n(\mathcal{U}) = (-\mathrm{tt}(\mathcal{U}) - \mathrm{toll}(\mathcal{U})/\mathrm{VOT}_n)/1\mathrm{s} \tag{33}$$

where $\mathrm{tt}(\mathcal{U})$ is the travel time on route $\mathcal{U}$, $\mathrm{toll}(\mathcal{U})$ is the toll accumulated along route $\mathcal{U}$, and VOT_n is individual n's value of time (in monetary units per time unit). For simplicity, it is assumed that all drivers have an identical value of time, i.e. $\mathrm{VOT}_n = \mathrm{VOT}$, $n = 1 \ldots N$.

The employed simulation logic is very simple. It recalculates the routes for 10% of all travelers in every iteration. For each replanning agent, a "proposal route" is generated by calculation of a time-dependent best path based on a randomly chosen VOT and the link travel times of the previous iteration. The proposal route is adopted by the agent if and only if has a higher utility than the hitherto applied route according to the agents actual VOT. Otherwise, the agent maintains its previous route.

4.2 Calibration procedure

A synthetic reality is generated by a simulation where all travelers react to the toll according to a 12 EUR/h VOT. Time-dependent flow rates are collected as synthetic sensor data at the 50 sensor locations indicated in Figure 2 (right). Since the sensors locations are fairly scattered, the measurements are assumed to follow independent normal distributions with identical variance σ^2.

The UM estimator is deployed. It does not interfere with the proposal route generation itself but only affects the subsequent choice between the proposal route and the hitherto applied route. Since the underlying route choice model cannot be shown to be of structure (29), this constitutes a heuristic application of the UM estimator.

The UM estimator replaces the original utility $V_n(\mathcal{U})$ of a route by $V_n(\mathcal{U}) + \langle \Lambda, \mathcal{U} \rangle / \mu$, cf (30), where the second addend results from a linearization of the sensor data's log-likelihood. For univariate normal measurements, this log-likelihood becomes a sum of squared deviations between estimated and measured flows, which are weighted by σ^{-2}. That is, the overall utility modification is effectively divided by $\mu\sigma^2$. This product defines the sole tuning parameter of the UM estimator, which is

$$w_{prior} = \sqrt{\mu\sigma^2}. \tag{34}$$

This parameter defines the weight of the behavioral prior information (represented by $V_n(\mathcal{U})$) when compared to the sensor data (basically represented by $\langle \Lambda, \mathcal{U} \rangle$).

4.3 Results

A prior scenario with an infinite VOT is assumed. That is, the demand simulator itself effectively ignores the toll. The question is investigated to what degree the UM estimator is able to reconstruct the network-wide traffic conditions of the synthetic reality (which results from a 12 EUR/h VOT) given only a limited set of flow measurements.

Figure 3 shows the resulting root mean square error measures over different w_{prior} values. The flow measurement reproduction error at the 50 sensor locations is given on the left, and the network state reproduction error between the estimated occupancies of all links in the network and those in the synthetic reality is shown on the right. For comparison,

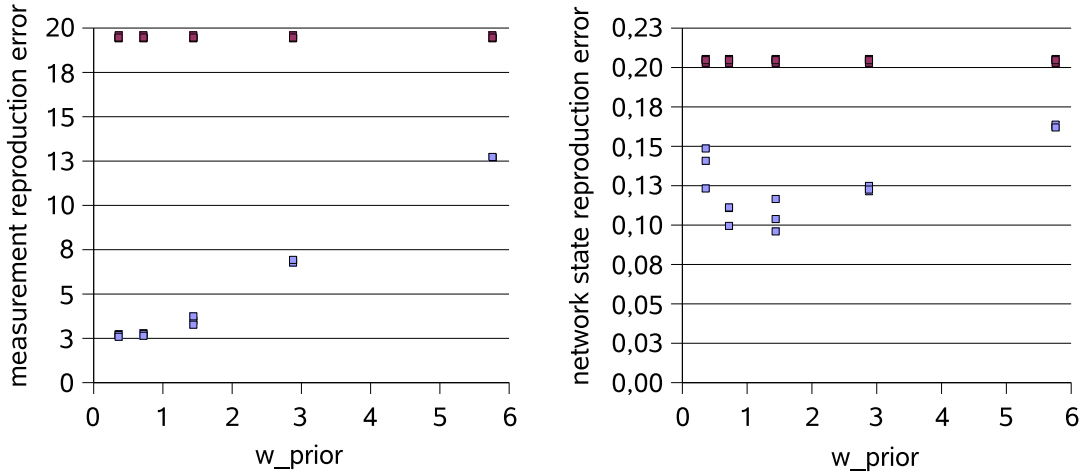

Figure 3: Result overview. Blue dots: estimation, red dots: plain simulation.

the error measures of four plain simulations of the prior scenario are also given in each diagram. They are equivalent to running the estimator without sensor input. For ease of comparison, they are re-drawn over every w_{prior} value in red color. The three estimation results per w_{prior} value are drawn in blue.

All results are fairly stable in that there is limited variability among repeated runs. Often enough, the dots lie on top of each other and cannot be distinguished. Reproducible convergence is a desirable and not at all self-evident feature for a nonlinear estimator. In these experiments, it can be observed with good precision.

The left diagram of Figure 3 shows that the measurement reproduction error decreases monotonously with w_{prior}. This is plausible: the smaller the belief in the behavioral model, the more weight is put on measurement reproduction. The greatest estimation improvement over a plain simulation of the prior is 86%.

The right diagram of Figure 3 shows a non-monotonous relation between w_{prior} and the network state reproduction error. As w_{prior} grows, the measurement influence vanishes and the estimation quality gracefully deteriorates towards that of a plain simulation. However, as w_{prior} decreases, a minimum error value is invariably encountered after which a further decrease of w_{prior} results in an increased network state reproduction error. This is an over-fitting effect. The attained minimum error value reflects the estimator's ability to spatiotemporally extrapolate the available flow measurements. A 48% improvement in network-wide traffic conditions is achieved – based on traffic counts from only 50 measurement locations out of altogether 2459 links.

Overall, these quality measures must be considered in light of the idealized setting in which they were obtained. However, it can be concluded that the estimator performs structurally correct, and that the calibration results in a specific application will mainly depend on the available data and modeling quality.

5 Summary and outlook

We presented a novel approach for the Bayesian calibration of plan choice distributions from aggregate sensor data. The proposed method is independent of the underlying demand simulator. The main precondition for its application is the availability of a supply simulator with linearizable traffic flow dynamics. The weakening of this requirement is an important aspect of our future research.

Another relevant issue is the utilization of the obtained estimation results. The identification of plan choice distributions clearly has predictive power within a considered day (this motivates an application in a telematics context), but it does not immediately allow to predict plan choice in changed conditions. Therefore, our method is currently extended to the joint calibration of plan choice distributions and the parameters of the underlying demand simulator. An example of such parameters are the utility coefficients of a random utility model. The following paragraph outlines this approach.

The AR estimator draws plans from an individual-level posterior choice distribution $P_n(\mathcal{U}|\mathcal{Y}) \propto e^{\langle\Lambda,\mathcal{U}\rangle} P_n(\mathcal{U})$, cf. (26). If the choice $\mathcal{U}_n$ of individual n depend on a parameter vector $\boldsymbol{\theta}_n$ which is distributed according to $p(\boldsymbol{\theta})$, the AR estimator can immediately be applied to a demand simulator that draws not only from n's parameterized choice distribution $P_n(\mathcal{U}|\boldsymbol{\theta}_n)$ but also from $p(\boldsymbol{\theta})$ itself. The modified AR procedure for a single decision maker n is as follows:

1. Draw $\boldsymbol{\theta}_n$ from $p(\boldsymbol{\theta})$.
2. Draw $\mathcal{U}_n$ from $P_n(\mathcal{U}|\boldsymbol{\theta}_n)$.
3. Accept $(\boldsymbol{\theta}_n, \mathcal{U}_n)$ with probability $\propto e^{\langle\Lambda,\mathcal{U}_n\rangle}$, otherwise goto 1.

The first accepted draw is from $p_n(\mathcal{U}, \boldsymbol{\theta}|\mathcal{Y})$ which is individual n's joint posterior plan and parameter distribution given the sensor data. That is, the plan choice distributions are now calibrated together with their underlying parameters. Our current work concentrates on an experimental investigation of this parameter calibration procedure.

A Linearization of the log-likelihood function

A linearization of the log-likelihood function with respect to plan choice must account for the coupling between $\mathcal{U}$ and $\mathcal{X}$ through the dynamical system constraint (12) that represents the demand simulator. This difficulty can be dealt with by well-known methods from control theory [20, 26]. A self-contained exposition is given in the following.

For the sake of generality, a functional $\Phi(\mathcal{X})$ of the macroscopic system states is linearized with respect to the population plan choice $\mathcal{U}_1 \ldots \mathcal{U}_N$. A time-additive structure of Φ is assumed. This corresponds to the structure of the log-likelihood since the likelihood itself is a product over subsequent time steps, cf. (17) and (18). Denote

$$\Phi(k) = \sum_{\kappa=k}^{K} \varphi[\mathbf{x}(\kappa), \kappa] \tag{35}$$

for $k = 1 \dots K$. This is the remaining contribution to $\Phi(\mathcal{X})$ from time step k on until the final time step K. It can be recursively written as

$$\Phi(k) = \begin{cases} \varphi[\mathbf{x}(k), k] + \Phi(k+1) & k = 1 \dots K-1 \\ \varphi[\mathbf{x}(K), K] & k = K. \end{cases} \tag{36}$$

As a first step, sensitivities with respect to states are computed by

$$\frac{d\Phi(k)}{d\mathbf{x}(k)} = \begin{cases} \dfrac{\partial\varphi[\mathbf{x}(k), k]}{\partial\mathbf{x}(k)} + \dfrac{d\Phi(k+1)}{d\mathbf{x}(k)} & k = 1 \dots K-1 \\ \dfrac{\partial\varphi[\mathbf{x}(K), K]}{\partial\mathbf{x}(K)} & k = K. \end{cases} \tag{37}$$

Since the interplay between variables at different time steps is fully defined by the state equation (12),

$$\frac{d\Phi(k+1)}{d\mathbf{x}(k)} = \frac{\partial\mathbf{f}[\mathbf{x}(k), \mathbf{u}_1(k) \dots \mathbf{u}_N(k), k]^T}{\partial\mathbf{x}(k)} \frac{d\Phi(k+1)}{d\mathbf{x}(k+1)} \tag{38}$$

holds for $k < K$, where $\mathbf{x}(k+1) = \mathbf{f}[\dots]$ was used and the superscript ^{T} denotes the transpose.

Now, sensitivities with respect to control variables $\mathbf{u}_1(k) \dots \mathbf{u}_N(k)$ result from

$$\frac{d\Phi(\mathcal{X})}{d\mathbf{u}_n(k)} = \frac{\partial\mathbf{f}[\mathbf{x}(k), \mathbf{u}_1(k) \dots \mathbf{u}_N(k), k]}{\partial\mathbf{u}(k)}^T \frac{d\Phi(k+1)}{d\mathbf{x}(k+1)}. \tag{39}$$

Here, $\partial\varphi[\mathbf{x}(k), k]/\partial\mathbf{u}_n(k)$ disappears since $\mathbf{u}_n(k)$ influences no state earlier than $\mathbf{x}(k+1)$. $\partial\mathbf{f}[\dots]/\partial\mathbf{u}(k)$ denotes the partial derivative of $\mathbf{f}[\dots]$ with respect to any $\mathbf{u}_n(k)$, which is independent of n. This independence allows to entirely omit the n subscript in Φ's sensitivities and to subsequently write $d\Phi(\mathcal{X})/d\mathbf{u}(k)$ instead of $d\Phi(\mathcal{X})/d\mathbf{u}_n(k)$, and it allows to compute all sensitivities for all agents simultaneously.

In summary, $d\Phi(\mathcal{X})/d\mathbf{u}(k)$ is obtained in a two-pass-procedure.

1. Using (38), solve (37) recursively for $k = K \dots 1$. Moving backwards through time introduces a "far sightedness" into the calculation that is necessary to predict the influence of present state variations on future system states.
2. Determine the influence of controls by (39) for $k = 0 \dots K-1$. Since this expression is identical for all agents, it needs to be evaluated only once for the entire population.

One obtains the following linearization of $\Phi(\mathcal{X})$ with respect to $\mathcal{U}_1 \dots \mathcal{U}_N$:

$$\Phi(\mathcal{X}(\mathcal{U}_1 \dots \mathcal{U}_N)) \approx \Phi(\mathcal{X}^0) + \sum_{k=0}^{K-1} d\Phi(\mathcal{X}^0)/d\mathbf{u}(k)^T \sum_{n=1}^{N} (\mathbf{u}_n(k) - \mathbf{u}_n^0(k)) \tag{40}$$

where $\mathbf{u}_n^0(k)$ is the control vector of traveler n at step k around which linearization took place and $\mathcal{X}^0$ is the resulting macroscopic state sequence. The computational logic behind this approximation can be seen most clearly if it is fully expanded:

$$\Phi(\mathcal{X}(\mathcal{U}_1 \dots \mathcal{U}_N)) \approx \sum_{n=1}^{N} \left(\sum_k \sum_{ij} \lambda_{ij}(k) u_{ij,n}(k) \right) + \text{const} \tag{41}$$

with

$$\lambda_{ij}(k) = d\Phi(\mathcal{X}^0)/du_{ij}(k). \tag{42}$$

Only such $\lambda_{ij}(k)$ coefficients are summed up in (41) that correspond to a nonzero turning move $u_{ij,n}(k)$ that is actually contained in $\mathcal{U}_n$. Using the same λ coefficients for all agents reflects that the sensitivities of Φ to a turning move (sequence) are independent of which agent is actually moving.

References

[1] K. Ashok. *Estimation and Prediction of Time-Dependent Origin-Destination Flows*. PhD thesis, Massachusetts Institute of Technology, 1996.

[2] M.G.H. Bell. The estimation of origin-destination matrices by constrained generalised least squares. *Transportation Research B*, 25B:13–22, 1991.

[3] M.G.H. Bell. Stochastic user equilibrium assignment in networks with queues. *Transportation Research B*, 29(2):125–137, 1995.

[4] M.G.H. Bell and S. Grosso. Estimating path flows from traffic counts. In W. Brilon, F. Huber, M. Schreckenberg, and H. Wallentowitz, editors, *Traffic and mobility: simulation–economics–environment*, pages 85–102. Springer, 1999.

[5] M.G.H. Bell, W.H.K. Lam, and Y. Iida. A time-dependent multi–class path flow estimator. In J.-B. Lesort, editor, *Proceedings of the 13th International Symposium on Transportation and Traffic Theory*, Lyon, France, July 1996. Pergamon.

[6] M.G.H. Bell, C.M. Shield, F. Busch, and G. Kruse. A stochastic user equilibrium path flow estimator. *Transportation Research C*, 5(3/4):197–210, 1997.

[7] M. Ben-Akiva and M. Bierlaire. Discrete choice methods and their applications to to short-term travel decisions. In R. Hall, editor, *Handbook of transportation science*, pages 5–34. Kluwer, 1999.

[8] M.E. Ben-Akiva and S.R. Lerman. *Discrete Choice Analysis*. MIT Press series in transportation studies. The MIT Press, Cambridge, Massachusetts and London, England, 1985.

[9] M. Bierlaire. The total demand scale: a new measure of quality for static and dynamic origin-destination trip tables. *Transportation Research B*, 36:837–850, 2002.

[10] E. Cascetta. Estimation of trip matrices from traffic counts and survey data: a generalised least squares estimator. *Transportation Research B*, 18B:289–299, 1984.

[11] E. Cascetta, D. Inaudi, and G. Marquis. Dynamic estimators of origin-destination matrices using traffic counts. *Transportation Science*, 27:363–373, 1993.

[12] E. Cascetta, A. Nuzzolo, F. Russo, and A. Vitetta. A modified logit route choice model overcoming path overlapping problems. Specification and some calibration results for interurban networks. In J. B. Lesort, editor, *Proceedings of the 13th International Symposium on Transportation and Traffic Theory*, Lyon, France, July 1996.

[13] E. Cascetta and N.N. Posterino. Fixed point approaches to the estimation of o/d matrices using traffic counts on congested networks. *Transportation Science*, 35(2):134–147, 2001.

[14] C.F. Daganzo. The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research B*, 28B(4):269–287, 1994.

[15] C.F. Daganzo. The cell transmission model, part II: Network traffic. *Transportation Research B*, 29B(2):79–93, 1995.

[16] G. Flötteröd. *Traffic State Estimation with Multi-Agent Simulations.* PhD thesis, Berlin Institute of Technology, Berlin, Germany, 2008.

[17] G. Flötteröd and K. Nagel. Some practical extensions to the cell transmission model. In *Proceedings of the 8th IEEE Intelligent Transportation Systems Conference*, pages 510– 515, Vienna, Austria, September 2005. IEEE.

[18] G. Flötteröd and K. Nagel. High-speed combined micro/macro simulation of traffic flow. In *Proceedings of the 10th IEEE Intelligent Transportation Systems Conference*, pages 1–6, Seattle, USA, September/October 2007. IEEE.

[19] M.C. Jones and S.K. Vines. Choosing the smoothing parameter for unordered multinomial data. *Sociedad de Estadistica e Investigacion Operativa Test*, 7(2):411–424, 1998.

[20] A. Kotsialos, M. Papageorgiou, C. Diakaki, Y. Pavlis, and F. Middelham. Traffic flow modeling of large-scale motorway networks using the macroscopic modeling tool METANET. *IEEE Transactions on Intelligent Transportation Systems*, 3(4), 2002.

[21] F. Logi, M. Ullrich, and H. Keller. Traffic estimation in munich: Practical problems and pragmatical solutions. In *Proceedings of the 4th IEEE Intelligent Transportation Systems Conference*, Oakland (CA),USA, August 2001. IEEE.

[22] M. Maher. Inferences on trip matrices from observations on link volumes: a bayesian statistical approach. *Transportation Research B*, 17:435–447, 1983.

[23] M.J. Maher, X. Zhang, and D. Van Vliet. A bi-level programming approach for trip matrix estimation and traffic control problems with stochastic user equilibrium link flows. *Transportation Research B*, 35:23–40, 2001.

[24] Y. Nie and D.-H. Lee. An uncoupled method for the equilibrium-based linear path flow estimator for origin-destination trip matrices. *Transportation Research Record*, (1783):72–79, 2002.

[25] Y. Nie, H.M. Zhang, and W.W. Recker. Inferring origin-destination trip matrices with a decoupled GLS path flow estimator. *Transportation Research B*, 39:497–518, 2005.

[26] J.B. Pearson and R. Sridhar. A discrete optimal control problem. *IEEE Transactions on automatic control*, 11(2):171–174, April 1966.

[27] M. Rieser, K. Nagel, U. Beuck, M. Balmer, and J. Rümenapp. Truly agent-oriented coupling of an activity-based demand generation with a multi-agent traffic simulation. In *Proceedings of the 86. Annual Meeting of the Transportation Research Board*, Washington DC, USA, 2007.

[28] H.D. Sherali, A.Narayan, and R. Sivanandan. Estimation of origin-destination trip-tables based on a partial set of traffic link volumes. *Transportation Research B*, 37:815–836, 2003.

[29] H.D. Sherali and T. Park. Estimation of dynamic origin-destination trip tables for a general network. *Transportation Research B*, 35:217–235, 2001.

[30] H.D. Sherali, R. Sivanandan, and A.G. Hobeika. A linear programming approach for synthesizing origin-destination trip tables from link traffic volumes. *Transportation Research B*, 28(3):213–233, 1994.

[31] H. Spiess. A maximum likelihood model for estimating origin-destination models. *Transportation Research B*, 21:395–412, 1987.

[32] K.E. Train. *Discrete Choice Methods with Simulation*. Cambridge University Press, Cambridge, United Kingdom, 2003.

[33] H. van Zuylen and L. G. Willumsen. The most likely trip matrix estimated from traffic counts. *Transportation Research B*, 14B:281–293, 1980.

[34] M. Vrtic, K.W. Axhausen, M.G. H. Bell, S. Grosso, and W. Matthews. Methoden zu Erstellung und Aktualisierung von Wunschlinienmatrizen im motorisierten Individualverkehr – methods for estimating and updating of origin-destination matrices from traffic counts. Forschungsauftrag auf Antrag der Vereinigung Schweizerischer Verkehrsingenieure 2000/379, Eidgenössisches Departement für Umwelt, Verkehr, Energie und Kommunikation / Bundesamt für Strassen, 2004. partially in German.

[35] H. Yang. Heuristic algorithms for the bilevel origin/destination matrix estimation problem. *Transportation Research B*, 29:231–242, 1995.

[36] H. Yang, T. Sasaki, and Y. Iida. Estimation of origin-destination matrices from link traffic counts on congested networks. *Transportation Research B*, 26(6):417–434, 1992.

[37] X. Zhou. *Dynamic origin-destination demand estimation and prediction for off–line and on–line dynamic traffic assignment operation*. PhD thesis, University of Maryland, College Park, 2004.