DYNAMIC TRAFFIC FLOW MODEL – A NEW APPROACH WITH STATIC DATA

Carsten Kemper
Scientific Assistant
Institute of Transport, Road Engineering and Planning (ivh)
University of Hannover - Germany
Appelstr. 9A, 30167 Hannover
Phone: +49-511-7623732
Fax: +49-511-7622520
E-Mail: Kemper@ivh.uni-hannover.de
Website: http://www.ivh.uni-hannover.de/peb

SUMMARY

Information of traffic condition in cities is hardly obtainable by road users and operators. In this paper, a new developed microscopic traffic simulation model is proposed, which can be used for several purposes. This novel traffic simulation model is based on a well known efficient and highly realistic traffic flow model. A new approach of an alternative departure time model using a combination of static OD matrices and multiple time series on each origin for modelling dynamic traffic flows in a transportation network is proposed. It is shown how the exact amount of vehicles of every origin for each time step is determined. Furthermore, three different approaches for route-choice modelling are presented. The paper concludes with the system calibration process and results regarding the travel time and velocity profiles for one OD relation of a real transportation network.

INTRODUCTION

Several approaches have been proposed for modelling vehicular traffic in the past. Most of the approaches are classified into macroscopic and microscopic models by the way the vehicle movements are considered. Macroscopic traffic modelling is based on the assumption that a traffic stream can be considered as a continuum of moving particles. It is common to distinguish two classes of macroscopic models: First-order models such as the LWR-model [19][20][24] describe macroscopic traffic states by continuous functions of the traffic density and the traffic velocity in the spatiotemporal way. Second-order models contain an additional partial differential equation for the average velocity and take into account the finite relaxation time to adapt the velocity to changing traffic conditions [5][21][22].

In microscopic traffic modelling, the motion of each vehicle in a traffic stream is considered. Likewise to the macroscopic models it is common to distinguish two classes of microscopic models: Car-following models focus on the non-linear interaction and dynamics of single vehicles. The driving behaviour of a vehicle depends significantly on the motion of the preceding vehicle. They specify their acceleration mostly as a function of the distance to the preceding vehicle, the own and relative velocity [10][11][23]. Submicroscopic models take into account even details such as perception thresholds, changing of gears, accelerations characteristics of specific car types, reaction to break lights and wipers [8][28]. Because of numerical efficiency, cellular automata describe the dynamics of vehicles in a coarse-grained way by discretizing space and time [6].

Each modelling approach has its own strengths and areas of application. While cellular and macroscopic traffic models are presently in use for identification and forecast of traffic states on highways [7][14][17], microscopic car-following and submicroscopic models are applied to the development of driver and traffic assistant systems as well as to microscopic traffic examination of single intersections [8][27].
Regarding to the available Software for Traffic Assignment Models, both modelling approaches can be further classified in two different classes: Static assignment models and dynamic assignment models. Both approaches have the same goal: Calculation of traffic volume on links. One of the differences between the static and dynamic approach is, that the dynamic assignment models normally require time-dependent OD-matrices to model traffic flow over time, while static assignment models only need at least one static OD matrix of overall trip demand. The acquisition and estimation of the time-dependent OD-matrices can be costly and their consistency can prove difficult to achieve, while static OD-matrices of overall trip demand have been established for various cities for their static assignment models over the last decades. Additionally, most cities and authorities already collect traffic data on several intersections for their statistics. Combining the static OD-matrix of overall trip demand and the time-profiles easily generated from the statistics for several streets or intersections seems to be a practical approach to model traffic flow over time and to generate time-dependent OD matrices. Another big disadvantage of recent microscopic dynamic traffic simulation models is that they are not suitable for the simulation of large-scale networks.

Thus, a new microscopic dynamic traffic model based on a highly realistic cellular automaton implementation is proposed, which uses the combination of traffic counts at intersections and one static OD-matrix, to model dynamic traffic flows in a large transportation network over time.

SIMULATION MODEL

Because important quantities like individual travel times, individual routes, velocity profiles and lane-change behaviour are not directly accessible in macroscopic models, a highly realistic microscopic traffic simulation model based on a cellular automaton is used in this approach. Due to their design cellular automata models are very efficient in large-scale network simulations. The first cellular automaton model for traffic flow that was able to reproduce some characteristics of real traffic, was suggested by Nagel and Schreckenberg in 1992 [18]. The model implemented in this simulator uses smaller cells to allow a more realistic acceleration and more speed bins. Presently a cell size of 1.5 m is used in this model. This corresponds to speed bins of 5.4 km/h and an acceleration of 1.5 m/s² (0 – 100 km/h in 19s), which is of the same order as the “comfortable” acceleration of about 1 m/s². A vehicle occupies 2 – 5 successive cells. The inclusion of anticipation, brake lights [14][4] and slow-to-start rules [1] in the modelling leads to a realistic driving. Further, the simulation model uses two different classes of vehicles: Passenger cars and trucks, where the trucks have a lower maximum velocity. With these extensions the simulation model is able to reproduce empirically observed traffic states on highways and in urban networks [4][7][12].

IMPLEMENTATION OF THE TOPOLOGY

The representation of the road network used in this simulator is based on a graph with nodes and links similarly to network models used in macroscopic equilibrium transportation models. Links are attributed with length, number of lanes, speed limits and counted volumes if available. The nodes are attributed with a flag whether they are an origin or destination or an ordinary (signalized) intersection. Due to the fact, that all vehicles are guided on a specific route, there must be information on which lane the vehicles are allowed to turn at the next intersection. Therefore after converting all lanes into a cellular automata, information of allowed turning relations at the next intersection are stored in the last cells of each lane (Figure 1). This has direct influence on the lane changing rules, because two incentives compete each other. All lane-changing rules, no matter if for a cellular automaton or other models follow a similar scheme [25]. In order to change lanes, vehicles need an incentive and the lane change needs to be safe. The two competing incentives can be that the other lane is faster, but the vehicle needs to make a turn on the current lane. In this case reaching the intersection on the correct lane is at higher order than getting faster to the intersection. Thus, a user-defined distance to the next node is introduced into the model that specifies if lane changing under this specific circumstance is allowed or not (Figure 1). Getting into the lane-change controlled area the vehicle only changes its lane if it needs to get on the correct turning lane.
Traffic lights are easily implemented in the model. The easiest cases are fully signalized intersections since the signal is taking care of avoiding crashes. The dynamics resulting from a red light can be generated by placing a virtual car with speed zero and a length of one cell into the last cell on the link, and removing this car once lights turn green (Figure 2).

More difficult are unprotected turns, which mean turns that are not regulated by traffic signals and where vehicles need to merge on their own without accident. Typical examples of this are yield, stop, "right on red", left turns against oncoming traffic and on-ramps to freeways. The mechanism used in this model is "gap acceptance" similar to the safety criterion for lane changes. Referring to Figure 3 the vehicle on the incoming road only moves into the major road if the gap there is big enough. This gap stretches upstream, since the incoming vehicle does not want the vehicle upstream on the major road to crash into itself. The German counterpart to the Highway Capacity Manual (HCM) [26], the "Handbuch fuer die Bemessung von Strassenverkehrsanlagen (HBS)" [9] states that drivers accept gaps that correspond to time headways of approximately 6 seconds, which means that the spatial gap needs to be proportional to the speed of the oncoming vehicle.
Another important point in the design of a simulator is the modelling of the departure time of the vehicles. The precondition for modelling a realistic departure time is, that a proved static traffic matrix (i.e. 24-hour – matrix) is available. In addition to this, some more traffic data like some statistics on the intersections must be available, too. Nearly every city with more than 50000 residents own these types of data. Combining the data sets in the new developed departure time model leads to a realistic departure time of all vehicles. Doing so, in each time step, the amount of vehicles \( n \) to be assigned to the network of every origin is calculated as shown in Figure 4. Every origin in the model is assigned to a junction nearby. The time series can be extracted, by counting the amount of vehicles passing the junction of the relevant stream.

**Figure 3: Illustration of gap acceptance for a left turn against oncoming traffic (according to [15])**

**Figure 4: Extraction of a time series of a junction**
The probability of destination choice for each vehicle and the ratio of vehicles from one source to every destination are determined by combining the original OD-matrix with the extracted time series. The choice of one destination of every vehicle is done using the Monte-Carlo-Method.

In detail, for each interval e.g. 1 hour, the amount of vehicles \( n \) to be assigned to the network in each time step (one second) of every origin is calculated. For example, there is an origin with an amount of 10000 vehicles for the whole simulated time period. The vehicles of this origin are distributed to three destinations. 1000 vehicles arrive in destination one, 3000 vehicles arrive in destination two and 6000 vehicles arrive in destination three. Regarding to the flow profile over time, 14.4% of the actual simulated hour depart from the first origin. The amount of vehicles leaving this origin in every time step can be easily calculated by:

\[
\frac{1440}{3600} = 0.4 \text{veh/s}
\]

Referring to the time step of the simulator, which is one second, \( n \) is going to be increased in every time step, until the condition \( n \geq 1 \) is reached. After assigning one vehicle to the network, \( n \) is decreased by 1 (see Table 1)

\[
n(t + \Delta t) = n(t) + n(t + \Delta t) - 1 \quad \text{if} \quad n(t) + n(t + \Delta t) \geq 1
\]

\[
n(t) + n(t + \Delta t) \quad \text{else}
\]

Table 1: Calculation of the amount of vehicles \( n \) to be put into the network

<table>
<thead>
<tr>
<th>t</th>
<th>( n(t) )</th>
<th>( n(t + \Delta t) )</th>
<th>( n(t) + n(t + \Delta t) )</th>
<th>put vehicle</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0.4</td>
<td>0.4</td>
<td>false</td>
</tr>
<tr>
<td>2</td>
<td>0.4</td>
<td>0.8</td>
<td></td>
<td>false</td>
</tr>
<tr>
<td>3</td>
<td>0.8</td>
<td>1.2</td>
<td></td>
<td>true</td>
</tr>
<tr>
<td>4</td>
<td>0.2</td>
<td>0.6</td>
<td></td>
<td>false</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>

To prevent that for one origin \( i = 1 \) and its destinations \( j = 2,3,...,n \), first all vehicles from \( i \) to two, then from \( i \) to three etc. are assigned to the network, the Monte-Carlo-Method is used to determine the probability (Table 2b) and the distribution function (Figure 3) for the destination of the vehicles. In this case the probability for destination 1 is \( p=0.1 \), for destination 2 \( p=0.3 \) and the probability for destination 3 is \( p=0.6 \).
Table 2: OD-Matrix

(a) $f_{ij}$ [veh/4h]

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-</td>
<td>1000</td>
<td>3000</td>
<td>6000</td>
</tr>
<tr>
<td>2</td>
<td>2000</td>
<td>-</td>
<td>4000</td>
<td>4000</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

(b) $p_{ij}$ [-]

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-</td>
<td>0.1</td>
<td>0.3</td>
<td>0.6</td>
</tr>
<tr>
<td>2</td>
<td>0.2</td>
<td>-</td>
<td>0.4</td>
<td>0.4</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

As one can see $f_{ij}$ is the probability $p_{ij}$, with which a vehicle chooses a route between $i$ and $j$.

$p_{ij} = f_{ij}$ with $\sum_{j=1}^{n} p_{ij} = 1$

Now, regarding to Table 2b, intervals for each origin are assigned to their destinations, which reflect the probability of their choice (Table 3).

Table 3: Separation of the interval for origin $i = 1$

<table>
<thead>
<tr>
<th>destination</th>
<th>interval</th>
<th>$p_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>j = 2</td>
<td>(0.0;0.1)</td>
<td>0.1</td>
</tr>
<tr>
<td>3</td>
<td>(0.1;0.4)</td>
<td>0.3</td>
</tr>
<tr>
<td>4</td>
<td>(0.4;1.0)</td>
<td>0.6</td>
</tr>
</tbody>
</table>

Figure 3: Sample distribution function

Referring to this, the choice of a destination $j$ for one vehicle, which shall be assigned to the network at the origin $i$ is done as follows:

(i) generation of a random number of the interval (0.0;1.0)
(ii) comparison of the random number with the intervals of the destination
(iii) assignment to the specific destination

If a vehicle cannot be assigned to the network, because the outgoing links of the origin are busy, the vehicle is put into a queue and will be assigned to the outgoing link as soon as possible during the next time steps.
DYNAMIC ROUTE-CHOICE

The dynamic flow of traffic concerning the route choice, is characterised that road users of a certain OD relation do not select inevitably the same route. The consequence for traffic modelling is that different routes for the same OD relation must be made available. Furthermore it is proven that the routes are variable concerning their costs over the time. Thus if alternative routes remain constant over time, this procedure is not really dynamic. For this reason the costs of a route are changed according to the choice of the road users, who also use different routes in the reality for different times. So, the costs of routes are variable over time.

The approach used in this model, is a combination of the length and the travel time of the link.

\[ w_i = \beta \cdot l_i + (1 - \beta) \cdot t_i \]

Thus the evaluation of a link contains both, a time-independent component, the length \( l_i \) of the link, and a time-dependent component, the travel time \( t_i \) of the link. At the beginning of a simulation each link gets a computed travel time as initial value assigned.

\[ t_i = \frac{l_i}{v_i} \]

\( v_i \) corresponds thereby the permissible speed on the link. In user definable intervals the travel time of each vehicle \( k \) on each link \( i \) \((i \in n)\) is measured. The arithmetic mean of the travel times of all vehicles on the link forms the travel time determining for the evaluation.

\[ t_i = \frac{\sum_{k=1}^{n} t_i^k}{n} \]

\( t_i^k \) represents thereby the travel time of the vehicle \( k \) on the link \( i \), while \( n \) is the number of all vehicles driven over this link in the regarded interval. Thus the impedance of each link for the dynamic case is given to:

\[ w_i = \beta l_i + (1 - \beta) \frac{\sum_{k=1}^{n} t_i^k}{n} \]

The Utility of each route is calculated by:

\[ U = \frac{1}{\hat{w}_i} \]

\( \hat{w}_i \) is the normalized impedance of the route.

The implementation of the route-choice is done using three different approaches. The first method is that every vehicle gets its route before it starts. For each OD relation a quantity of \( n \) alternative routes, based on the approach shown above, are calculated and stored in an array. With the Multinomial Logit [2] or C-Logit Model [3], the probabilities for the choice of a certain route are computed by the vehicles. By picking one of the alternative routes, the vehicles follow their route to their destination.

In the second approach, every vehicle can choose another route during its journey from every node of the network if "Trip-Update" is true for the vehicle. However, this is only possible, if the vehicle exceeds the time limit to its destination. Therefore, every vehicle gets the estimated driving time to the destination when it leaves its origin, based upon the actual network loading. If driving time and the introduced individual "satisfaction-factor" is exceeded, the vehicle can choose one of the alternative routes in the array to its destination, or if no route meet its requirements it can start a new shortest path search (over all nodes), depending on the actual network loading. The updated routes are stored in the array, so all other vehicles can access these new routes.

Alternatively to the described approaches of route-choice, the algorithm introduced by HILLIGES [12] for his macroscopic model was transferred to this microscopic model. This approach makes the storage
of an assortment of alternative routes needlessly. In this case, the vehicle only knows its origin and its destination but not the route. Starting from the origin, the vehicle decides at every node, on which outgoing link it wants to leave the current node. Every outgoing link gets a utility, which is calculated dependence on the destination of the vehicle. Every length of the outgoing link from the current node is summed to the length of the shortest path of the successors node outgoing link. Then the utility of every path from the current node to the destination is calculated and the probability of using one of the outgoing links for the current OD-Relation is determined by using the Monte-Carlo-Method. The vehicles themselves find their route through the transportation network. The advantage of this method is that the vehicles can use all available routes, and not only the available assortment of alternative routes. For sure, the route a vehicle chooses is not always the best, but the transportation network loading is more balanced than using only an assortment of alternative routes. Nevertheless, the shortest path trees need to be updated in user specified intervals, because the network loading and the impedance (and the utility) of all links changes over time.

**SYSTEM CALIBRATION PROCESS AND RESULTS**

All simulation results were made using a real transportation network. The test field is a district of Hannover - Germany. The network is based on a graph with 280 nodes and 660 links. 140 links have counted traffic volumes and 48 nodes are marked as origins and destinations. Speed limits range from 30 to 60 km/h (grey and black) and every street is attributed with the original number of lanes. The area is controlled by 55 signalized intersections (Figure 6).

![Figure 6: Test field Hannover (List) – Germany with the examined OD relation (Red Route 1)](image)

The control sample used for this study was about 20 driven journeys for the given OD relation (red route in Figure 6) between 6 am and 10 am. All journeys were logged with a GPS-system with a time step of 1 second. The distance between the origin and the destination was about 3.9 kilometres. 16 signalized intersections are on the distance within this OD relation.

Calibration of the simulation system is the process by which the parameters of various components of the simulation system are set, so that it will accurately replicate observed traffic conditions. Calibration of traffic simulation tools is not a trivial task. The source of the difficulty is that calibration data are aggregate measurements of traffic characteristics (e.g. flows, speeds, travel times etc.), which are the emergent results of the interactions between various behaviours of individual vehicles. Driving behaviour in this case encompasses a set of rules at the operational and tactical level, such as
acceleration, lane changing and gap acceptance. The travel behaviour is composed of strategic-level algorithms, such as route-choice.

In addition to the traffic measurements, the information available for calibration include a-priori values of parameters, such as historical OD flows extracted from a planning model and default values for the behaviour parameters extracted from similar studies and from the literature.

**OD-ESTIMATION**

However, assuming plausible behaviour parameters the historical OD matrix was used as the seed OD-matrix. Although this matrix was not up to date, it still contained valuable information regarding the structural relation among OD pairs. The real (up to date) matrix was estimated iteratively from the simulation model itself. Simulating vehicles of the historic matrix using the described departure time model and processing the correction algorithms proposed of VAN ZUYLEN/WILLUMSEN [28] after each simulation process, an OD matrix that fits the observed traffic counts was calculated after approximately 4 iterations of simulation and correction. The algorithms of VAN ZUYLEN/WILLUMSEN are based on minimum information and entropy maximizing principles.

**TRAVEL TIMES**

Assuming a correct OD matrix and plausible behaviour parameters finding the correct output travel times that are consistent with the experienced travel times was by itself a fixed-point problem. The problem was solved by re-calibrating some of the behaviour parameters. Especially re-calibrating the velocity-dependent temporal range of interaction with the brake light of the car ahead, leads to better consistency with the experienced travel times.

Because the model uses a velocity dependent randomization, the probability function dependent on the velocity of the vehicle and of the brake light of the next vehicle in front was slightly modified. Changing these parameters the simulation model produces rather good results referring to travel times and the velocity profile for the given pair of OD relation (Figure 7 and Figure 8).

![Comparison of Travel Times - Route 1](image)

*Figure 7: Comparison of Travel Times of route 1 (red route in Figure 6)*
ARRIVAL DISTRIBUTION AT INTERSECTIONS

Assuming a correct OD matrix, good behaviour parameters and fitting time profiles for every origin the arrival distribution at several intersections were examined. Due to the design of the departure time model, vehicle volumes at an intersection arm directly connected with an origin must fit the observed traffic volumes within one hour (Figure 9).

As one can see, in this case the simulated traffic volume matches the counted traffic volume. More difficult is the case if the intersection is not directly connected with an origin, so vehicles of more than one origin and with different routes arrive at the examined intersection arm. In this case, the routes of all vehicles concerning the intersection arm have to be examined and the time profile of the concerning origins have to be adjusted. This is not a trivial task, because multiple routes exist from an origin to a destination and the route-choice has to be explicitly modelled, which complicates the problem.
However, this problem could not be solved for the whole examined network, because empirical route-choices were not available, but for several intersections, the model calculated fitting traffic volumes (Example in Figure 10).

![Traffic Volume per hour at an intersection arm](image)

**Figure 10:** Traffic volume per hour at an intersection arm *not* directly connected to an origin

**CONCLUSIONS**

This paper presents a microscopic traffic model based on a state-of-the-art cellular automaton implementation with a new developed departure time model. Combining static and already proven OD-matrices with different time series for each area or origin, seems to be a good approach to use already proven static OD matrices in dynamic assignment models. Furthermore, three different approaches for route-choice modelling are presented. Assuming correct input-parameters and model-parameters, the presented microscopic traffic simulator produces reliable results referring to travel times and velocity profiles. Once the model is calibrated for a given situation, it is able to produce time dependent OD matrices for other dynamic traffic simulation models. Due to the design of the implemented traffic flow model, the traffic simulator is able to simulate large networks in an adequate time. The presented network was simulated in 9-times real-time on a standard Desktop PC (P4, 2.4GHz). At present the simulator is for offline use only, but it will be enhanced to an online simulator for city traffic in the future.

**REFERENCES**

[1] Barlovic, R.; Santen, L.; Schadschneider, A.; Schreckenberg, M. 
"Metastable states in cellular automata for traffic flow."

"Discrete choice analysis - Theory and Application to Travel Demand."

A Modified Logit Route Choice Model Overcoming Path Overlapping Problems.

"OLSIM: Future Traffic Information"

[5] Cremer, M.  
"Der Verkehrsfluss auf Schnellstraßen"  
Springer, Berlin, 1979

"A fast simulation model for traffic flow on the basis of Boolean operations"  

[7] Esser, J.; Schreckenberg, M.  
"Microscopic simulation of urban traffic based on cellular automata"  
Int. J. Mod. Phys. C8, 1025-1036, 1997

[8] Fellendorf, M.  
"VISSIM for traffic signal optimisation"  
Traffic Technology International ’96, 190-192, UK & International, Dorking, 1996

[9] Forschungsgesellschaft für Straßen und Verkehrswesen  
"HBS – Handbuch für die Bemessung von Straßenverkehrsanlagen"  
Köln, 2001

[10] Gazis, D. C.; Herman, R.; Potts, R. B.  
"Car-following theory of steady-state traffic flow"  
Opns. Res. 7, 499-505, 1959

"Nonlinear follow the leader models of traffic flow"  
Opns. Res. 8, 545-567, 1961

[12] Hilliges, M.  
"Ein phänomenologisches Modell des dynamischen Verkehrsflusses in Schnellstraßennetzen."  
Dissertation, Universität Stuttgart, Shaker-Verlag, Aachen, 1995

"Dynamische Verkehrsumlegung mit statischen Matrizen in einem mikroskopischen Simulationsmodell"  
Proceedings of the HEUREKA ’05 Conference, 405-422, 2005

[14] Knospe, W.; Santen, L.; Schadschneider, A; Schreckenberg, M.  
"Towards a realistic microscopic description of highway traffic"  

[15] Nagel, K.  
"Multi-agent transportation simulation"  
not published yet, state February 2004

[16] Nagel, K.  
"Individual adaptation in a path-based simulation of the freeway traffic of Northrhine-Westfalia"  

[17] Nagel, K.  
"Experiences with iterated traffic microsimulations in Dallas"  

[18] Nagel, K.; Schreckenberg, M.  
"A cellular automaton model for freeway traffic."  
J. Physique I2, 2221-2229, 1992

"On kinematic waves: I. Flood movement in long rivers."  
Proceedings of the Royal Society A 299, 281 – 317
"On kinematic waves: II. A theory of traffic on long crowded roads."  
Proceedings of the Royal Society A 299, 317-345

[21] Payne, H.J.  
"Models of freeway traffic and control"  

[22] Payne, H.J.  
"FREFLO: A macroscopic simulation model of freeway traffic"  

[23] Reuschel, A.  
"Fahrzeugbewegungen in der Kolonne"  
Österr. Ingen.-Archiv 4, 193-215, 1950


[25] Sparmann, U.  
"Spurwechselvorgänge auf zweispurigen BAB-Richtungsfahrbahnen"  
Forschung Straßenbau und Straßenverkehrstechnik 263, Bundesminister für Verkehr, Bonn, Germany, 1978

[26] Transportation Research Board  
"Highway Capacity Manual"  

[27] Treiber, M.; Helbing, D.  
"Microsimulations of freeway traffic including control strategies"  
Automatisierungstechnik 49, 475-484, 2001

[28] van Zuylen, H.J.  
"Some improvements in the estimation of an OD-Matrix from traffic counts"  
The 8th International Symposium on Transportation and Traffic Theory, Toronto, 1981

[29] Wiedemann, R.  
"Simulation des Straßenverkehrsflußes"  
Institut für Verkehrswesen, Universität Karlsruhe, 1974