Marcel Rieser, Kai Nagel, Ulrike Beuck, Michael Balmer, Jens Rümenapp

Agent-Oriented Coupling of Activity-Based Demand Generation with Multiagent Traffic Simulation

WISSEN IM ZENTRUM
UNIVERSITÄTSBIBLIOTHEK

Technische
Universität
Berlin

Agent-Oriented Coupling of Activity-Based Demand Generation with Multiagent Traffic Simulation

Marcel Rieser, Kai Nagel, Ulrike Beuck, Michael Balmer, and Jens Rümenapp

The typical method to couple activity-based demand generation (ABDG) and dynamic traffic assignment (DTA) is time-dependent origin–destination (O-D) matrices. With that coupling method, the individual traveler's information gets lost. Delays at one trip do not affect later trips. However, it is possible to retain the full agent information from the ABDG by writing out all agents' plans, instead of the O-D matrix. A plan is a sequence of activities, connected by trips. Because that information typically is already available inside the ABDG, this is fairly easy to achieve. Multiagent simulation (MATSim) takes such plans as input. It iterates between the traffic flow simulation (sometimes called network loading) and the behavioral modules. The currently implemented behavioral modules are route finding and time adjustment. Activity resequencing or activity dropping are conceptually clear but not yet implemented. Such a system will react to a time-dependent toll by possibly rearranging the complete day; in consequence, it goes far beyond DTA (which just does route adaptation). This paper reports on the status of the current Berlin implementation. The initial plans are taken from an ABDG, originally developed by Kutter; to the authors' knowledge, this is the first time traveler-based information (and not just O-D matrices) is taken from an ABDG and used in a MATSim. The simulation results are compared with real-world traffic counts from about 100 measurement stations.

Arguably, the most advanced state-of-the-practice method for transport forecasting consists of the following three pieces:

• The process starts with an activity-based demand generation (ABDG). Practitioners typically start with a synthetic population (*1*) and then add, to every potential traveler in the synthetic population, status (work, school, other), full activity patterns, activity locations, activity times, and possibly mode choice. Other sequences to add to these elements are possible, and some or all of them may be generated jointly. Examples of practical applications of activity-based models

M. Rieser, K. Nagel, and U. Beuck, Transport Systems Planning and Transport Telematics, Technische Universtät Berlin, Salzufer 17-19, Sekr. SG 12, 10587 Berlin, Germany. M. Balmer, Institute for Transport Planning and Systems, Eidgenössische Technische Hochschule Zürich, 8093 Zurich, Switzerland. J. Rümenapp, Gertz Gutsche Rümenapp, Stadtentwicklung und Mobilität, Büro Berlin, Postfach 120361, 10593 Berlin, Germany. Corresponding author: K. Nagel, nagel@vsp.tu-berlin.de.

can be found in San Francisco, California (*2*); Portland, Oregon (*3*); Florida (*4*); Toronto, Ontario, Canada (*5*); and the Netherlands (*6*, *7*). The typical outputs of an ABDG are time dependent—for example, hourly, origin–destination (O-D) matrices.

• These hourly O-D matrices are then fed into a dynamic traffic assignment (DTA), which assigns routes to time-dependent O-D flows so that the routes, in conjunction with their resulting traffic pattern, fulfill some predefined criterion. For example, the routes may fulfill a Nash equilibrium, meaning that at no time of day can any O-D flow find a faster path than those that are already used. An often-used alternative criterion is a time-dependent stochastic user equilibrium, meaning that each O-D flow is distributed across possible routes following a prespecified distribution function at each point in time. The typical way to solve the DTA problem is to use iterations between a router and a traffic simulation (also called network loading algorithm). Flows on routes that do not fulfill the prespecified criterion are slowly adjusted into the right direction. The iterations stop when no more adjustments are necessary—that is, when the iterations have reached a fixed point (*8*). Examples of DTA projects are Dynasmart (*9*), DynaMIT (*10*), and a dynamic version of VISUM (*11*).

• To close the feedback loop, spatial impedances—often in the form of interzonal travel times—are fed back from the DTA to the ABDG, and the feedback is iterated until a self-consistent solution is found. Although this feedback was postulated some time ago (*12*), it is usually implemented manually by the analyst; that is, the analyst manually goes back and forth between ABDG and DTA until the solution is satisfying.

Unfortunately, this approach of coupling the ABDG and the DTA via O-D matrices and link travel times can have several disadvantages:

• If a traveler is delayed in the morning, this may have temporal repercussions throughout the whole day. Such an effect is not picked up when travelers are converted into O-D streams. In fact, even when feeding back 15-min-averaged link travel times, the resulting travel time map can be highly distorted (*13*).

• The traffic may have temporal patterns that go beyond hourly resolution (e.g., in reaction to a time-dependent toll). Hourly O-D matrices cannot pick up such effects.

• When tolls are charged on certain links, the routing decision may depend on the travelers' attributes (e.g., on income or on time pressure given by activities later during the day). Access to such information gets lost through the O-D matrix.

• Some higher-level decisions, such as mode choice and secondary activity location choice, may depend on relatively small details of

the trip, such as walking distance to the public transit stop. Alternatively, mode choice may depend on one element of a complete tour, such as one location not being reachable by public transit. Such effects are straightforward to include when the DTA still knows about individual agents, but are impossible to pick up once travelers are aggregated into O-D matrices.

In contrast, multiagent simulations (MATSim) of traffic process information about individual travelers at every level. Every modeled agent is assigned at least one plan, a series of activities, and connecting trips. The plan is made complete step by step by behavioral module, inserting, for example, activity patterns, activity locations, a mode choice decision, and precise routes and times. Because these processes occur in the travelers' heads, this is called the "mental layer" of the MATSim. Once all plans are complete, they are submitted to the traffic flow simulation, which attempts to execute them as faithfully as possible, given physical constraints caused by the system or by other travelers (e.g., congestion). Because the traffic flow simulation models physical processes, this is called the "physical layer" of the MATSim. The distinction between the physical and the mental layer is taken from multiagent systems (14).

During the traffic flow simulation, the performance of every agent is recorded, for example, by noting departure and arrival times at activity locations. From that performance information, the score (e.g., utility) of every plan is computed. During repeated iterations between the mental and the physical layer, every agent attempts to modify its plan so that it obtains a larger score. The choice dimensions along which the agents can learn are configurable. In this way, the simulation system can emulate DTA (by allowing only the routes to adapt), or it can, for example, simulate a system in which agents' residences, status, and primary activity locations are fixed, but everything else (secondary activity types and locations, mode, route, time scheduling) is adapted. Similarly, the scoring function is configurable, its only requirement being that it needs to allow plans to be ranked. In this way, not only standard utility functions (15) but also, for example, prospect theory (16) can be included in a conceptually straightforward way.

An early attempt to use true agents that maintain their identity throughout the system (i.e., from the synthetic population generation through the activity-based demand generation and the router to the traffic flow simulation) was TRANSIMS (17). Its main shortcoming for a couple of years was that it was difficult to obtain and to use; now it is open source.

Another attempt to approach the problem is Metropolis (18), which uses a single O-D matrix for the morning peak travel and generates the temporal structure inside the DTA. That is, the time choice dimension is added to the route choice dimension inside the DTA.

MATSim (19) is based on TRANSIMS. It differs from TRANSIMS in several aspects, including using just one hierarchical (XML) file format for the coupling between the behavioral modules (instead of several plain ASCII files), giving performance scores to full 24-h plans based on the execution of those plans in the physical layer, and using those scores for agent learning [instead of assuming that each new solution generated by a behavioral module is better than the solution before—an assumption that was found to be destructively invalid in certain situations (13)]. MATSim also uses a more lightweight, less data hungry, and faster simulation of the physical layer: the queue simulation (20).

To start the learning iterations of any MATSim for traffic, initial conditions are needed. The most reasonable initial condition is a set of agents (synthetic population), in which every agent has at least

one plan in which at least those elements that are not supposed to adapt during the learning iterations are defined.

It makes sense to see if existing methods can provide initial conditions for a MATSim. The first thing to consider here are O-D matrices, because they are so abundant in the community. It is straightforward to extract individual trips from O-D matrices and to generate for each trip a pseudo-agent, whose complete plan consists of that single trip. This means, however, that trips that in reality belong together, because they are executed by a single traveler throughout the course of a day, are now completely decoupled. Therefore, no consistent adaptation between these trips is possible (e.g., time choice, location choice, tours mode choice); the only things that can be sensibly modified within the MATSim are the routes.

Generating complete agents' plans from O-D matrices has also been tried but is a complex and cumbersome process (21).

Fortunately, many implementations for ABDG (3–7, 22) internally use concepts that are much closer to agents and aggregate the results to O-D matrices as their final step. Therefore, it makes sense to investigate whether it is possible to output some other information from the ABDG that is easier to use by the MATSim.

This paper describes the steps taken to transform the individuals' data from an existing ABDG into agents' plans. The ABDG is based on the Kutter model (also known as Berliner Personenverkehrs-Modell) (23, 24) for the region of Berlin, Germany. The plans are then used as input for the multiagent traffic simulation MATSim (25). Then the results of the simulation are compared with real-world data.

CREATING PLANS FROM ACTIVITY CHAINS

The Kutter model is a disaggregated, activity- and behavior-oriented traffic demand generation model. It simulates the traffic demand of person groups with homogeneous behavior with the help of expectancy values (23, 24). This model is currently used to calculate daily O-D matrices for strategic planning and was modified to output the internally used activity chains (26). The remainder of this paper describes how the ABDG data are used in the multiagent simulation.

Each activity chain contains information about the start location (reference to a zone), up to four activities (limitation of the Kutter model), and the frequency of occurrence of the activity chain (see Figure 1). Each of the four activities is described by type, location, and the transportation mode used to reach that location. In addition, the activity chains are grouped, each group corresponding to one of 72 person groups with similar demographic attributes. The activity chains always end at the same place where they started (tours), whereas all activities in between take place somewhere else. The sum of all frequencies corresponds to the total number of tours accomplished by the people in the study area. There is no special provision for people who perform more than one tour per day; this leads to problems, as discussed later.

Much information needed for agents' plans like activities and locations is available, with only the time information missing. In this simulation, time information is generated and optimized over several iterations by a special module (27). Initially, all activities are assigned a random activity duration within a range, where the range depends on the type of activity. These random durations are replaced by more convenient durations during the iterations by the mentioned module. The scoring algorithm ensures that only activities with a configurable minimal duration are considered useful.

Based on the given description, it appears possible to use all the information from the activity chains and transform them into agents'

```
┌ one activity chain
│ 1.84514    2   1    2   1   99    2   1   0   0   0   0   0   0
└ 3.35083    2   1    5   1   99    2   1   0   0   0   0   0   0
  0.02391    2   5   20   5   99    2   5   0   0   0   0   0   0
  0.01277    2   5   38   5   15   82   5  99   2   5   0   0   0
  0.00098  696  14   66   4   99  696   4   0   0   0   0   0   0
  0.00111  696  14  109   5   99  696   5   0   0   0   0   0   0
```

4 activities, each with activity-type, location, mode of traffic

start location (zone-id)

frequency of occurrence

over 250m different activity chains, divided into 72 groups with similar demographic attributes
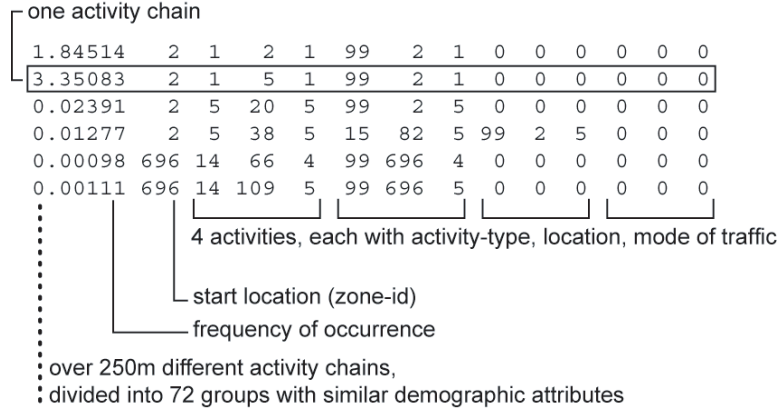
FIGURE 1   Structure of activity chains generated by Kutter model.

plans. This would work flawlessly if the frequency of occurrence of each activity chain were an integer value. However, they are floating-point numbers. This may work in assignments in which frequencies are summed up on each link to get the total volume. However, in agent simulations, the smallest unit is an agent that cannot be split into two or more parts. Thus, a way had to be found to deal with the fractional frequencies of activity chains.

The input data from ABDG contain more than 7 million tours per day in the study area. More than 250 million different activity chains are used to describe the tours. Thus, there are many times more activity chains than there are tours, resulting in an average frequency per tour of much less than 1. This leads to problems when generating agents' plans from the activity chains: not every activity chain can be converted into a plan, but the frequencies of the activity chains must be considered to decide whether to use the data.

It was decided to use the following method: the tour frequencies are summed up one after the other. Every time the sum reaches 1.0 or any higher value, an agent with a plan based on the current activity chain is generated and the sum is reduced by 1.0. If the activity chains are in a random sequence (which is assumed), then the method corresponds to a weighted random draw without replacement. Finally, more than 7 million agents with each assigned one plan were generated, representing the 7 million tours undertaken by the people in the study area. As discussed later, it is important to note that the number of agents created represents not the number of inhabitants in the area but the number of tours performed in 1 day by the inhabitants. This means there are more agents in the simulation than there are in the real world, but the agents have shorter day plans than their real-world counterparts.

MATSim is currently able to simulate only individual car traffic. Therefore, only agents using cars for transportation were considered for the simulation. Every trip starts and ends at a link in MATSim. However, many traditional DTA models use demand at the level of traffic zones, and the corresponding ABDG, including the Kutter model, generate demand only at the level of traffic zones, too. Thus, it was necessary to assign links to activity locations to the location where trips start and end.

The following procedure was used to assign links to activity locations. In a first step, each activity is assigned a coordinate. In a second step, the nearest link to this coordinate is searched for and assigned to the activity. This two-step procedure allows for reusing the plans with different networks as long as the coordinates are maintained. The coordinate is drawn randomly around the centroid of the traffic cell using polar coordinates. By randomly choosing an angle and randomly choosing the distance, the density around the center of the circle is higher than it is on the border of the circle. This corresponds to the different population densities in the center of a village and rural areas. The maximal distance a point can be located away from the centroid is 0.7 times the distance from the centroid to the nearest neighbor. The factor 0.7 has proved to lead to good coverage of the circle areas and keeps overlaps low.

SIMULATION AND SCORING

Once the agents' plans are available, the simulation process can start. MATSim (*19, 25*) iterates between the traffic flow simulation (physical layer; sometimes called network loading) and the behavioral modules (mental layer). The traffic flow simulation moves the agents through the network according to their plans and generates events (e.g., vehicle enters or leaves a link, agent departs or arrives at an activity location) from which travel times, travel speeds, link densities, and other characteristics can be calculated. At the end of an iteration, each plan is evaluated for how successful the agent was in performing the (planned) activities, resulting in a score for the plan (*28*). Scoring a plan is a precondition so that agents learn and react to congestions or tolling. Different plans can be compared and an agent can pick the one with the highest value. A higher score implies that the agent makes better use of its day.

As a scoring function, the traditional utility function based on the Vickrey bottleneck model is used (*29*) but modified to be consistent with complete day plans. Scoring is based on event information from the physical layer. Performing an activity is rewarded; travel times and late arrival are punished. The overall equation is

$$U_{\text{plan}} = \sum_i U_{\text{act},i} + \sum_i U_{\text{trav},i} + \sum_i U_{\text{late},i} \qquad (1)$$

The utility of performing an activity is assumed to increase logarithmically

$$U_{\text{act},i}(x) = \max\left[0, \alpha \cdot \ln\left(\frac{x}{t_0}\right)\right] \qquad (2)$$

where x is the duration one spends at the activity. Take $\alpha = \beta_{\text{dur}} \cdot t^*$, where β_{dur} is uniformly the same for all activities (6 €/h) (1€ = \$1.38

3

in 2007 U.S. dollars) and only t^* varies between activity types. With this formulation, t^* can be interpreted as a typical duration, with β_{dur} as the marginal utility at that typical duration

$$\left.\frac{\partial U_{act,i}}{\partial x}\right|_{x=t^*} = \beta_{dur} \cdot t^* \cdot \frac{1}{t^*} = \beta_{dur} \qquad (3)$$

t_0 can be seen as a minimum duration of an activity but is better interpreted as a priority: All other things being equal, activities with large t_0 are less likely to be dropped than activities with small t_0 (28).

The utilities of traveling and of being late are both seen as disutilities, which are linear in time

$$U_{trav,i}(x) = \beta_{trav} \cdot x \qquad (4)$$

where x = time spent traveling and

$$U_{late,i}(x) = \beta_{late} \cdot x \qquad (5)$$

where x = time an agent arrives late at an activity. β_{trav} is set to -6 €/h, and β_{late} is set to -18 €/h.

In principle, arriving early or leaving early could also be punished. However, there is no immediate need to punish early arrival, because waiting times are already indirectly punished by foregoing the reward that could be accumulated by doing an activity instead (opportunity cost). In consequence, the effective (dis)utility of waiting is already -6 €/h. Similarly, that opportunity cost has to be added to the time spent traveling, arriving at an effective (dis)utility of traveling of -12 €/h. No opportunity cost needs to be added to late arrivals, because the late arrival time is already spent somewhere else. In consequence, the effective (dis)utility of arriving late remains at -18 €/h. These effective values are the standard values of the Vickrey model.

It would make sense to consider an additional punishment (negative reward) for leaving an activity early. This would describe, for example, the effect when there are, on a specific day, better things to do than to continue to work, but some type of contract (e.g., shop opening hours) forces the agent to remain at work.

A fixed percentage of agents will replan their day plan with one of the behavioral modules. The currently implemented behavioral modules are route finding and time adjustment. Using route finding, agents try to find better routes but do not change their departure times or the duration of activities. To find better routes, they make use of the events to calculate actual travel times and thus recognize jammed links. Using time adjustment, the departure times and activity durations are modified with the goal of optimizing the individuals' plan score (30). Additional behavioral modules are conceptually clear but not yet implemented: activity resequencing would change the order of activities (e.g., shopping after work instead of before work), and activity dropping would remove certain activities in an overloaded plan.

When replanning, an agent keeps its original plan and modifies a copy of it. Thus, agents collect more variants of plans they can perform. Each agent can remember a configurable number of plans. If a behavioral module generates an additional one, the plan with the worst score will be removed to store the new one.

Such a system with several different behavioral modules and adjusted scoring algorithms will react to a time-dependent toll by possibly rearranging the complete day; in consequence, it goes far beyond DTA, which just does route adaptation.

The simulation is stopped when the agent's average score no longer significantly improves.

SCENARIO SETUP

The chosen study area of Berlin and its surroundings covers an area of 150×250 km and has a population of about 6 million inhabitants. The focus is on the urban area of Berlin; therefore, this part of the region is represented with a much higher level of detail and accuracy than Brandenburg, Germany, with regard to network and demand.

The road network was originally developed by the planning department of the city of Berlin (Senatsverwaltung für Stadtentwicklung). It has been used for the city's forecast model representing the year 2015. Manual changes were necessary to exclude modifications of the road network planned until 2015. The final road network representation consists of more than 10,000 nodes and almost 30,000 links.

Nodes are described by their coordinates. Links are described by their from-nodes and to-nodes and possess attributes such as length, free flow speed, number of lanes, and capacity. These network attributes are sufficient for the queue simulation. Unfortunately, the number of lanes is not needed for traditional assignment, and as a consequence the quality of the data is often poor. In addition, capacity is just a calibration factor in static assignment and may be quite unrelated to the "hard" capacity needed by the queue simulation. Both issues are discussed in more detail later.

The network has been used in Berlin with a scope of 24 h. The demand is described by daily O-D matrices, which are based on traffic analysis zones, where the different matrices refer to different types of traffic (e.g., passenger, freight). Such demand is assigned to the network using static assignment according to defined capacity speed functions.

A crucial attribute of a network link is its capacity, which is interpreted very differently by the aggregated model used by the planning department of Berlin and the multiagent simulation. In this simulation, capacity is understood as maximum outflow of a link in a given time period, whereas the assignment model of the planning department does not treat capacity values as hard constraints. As is common, it uses suitable functions to relate capacity and flow to the resulting cost in terms of travel times. Thus, it was necessary to adapt the theoretical capacity values that were the basis for a 24-h static assignment. A factor was derived according to the fact that the daily traffic basically occurs in 12 h of a day. In a second step, the resulting theoretical 1-h values used in static assignment were converted into maximum values of outflow of a link in 1 h. The maximum outflow of a link used in this simulation is double the 1-h values used in traffic assignment. In addition, the storage of a link is constrained. The storage of a link can be calculated as length times the number of lanes divided by the space a vehicle occupies in a jam (7.5 m). Unfortunately, the number of lanes attribute is set to be one on all links of the original network, because the number of lanes is not necessary for static assignment. In this simulation, the number of lanes is set to two to calculate maximum storage, but a better solution has to be found in the near future.

To speed up the scenario, the demand and the network capacities were scaled down to 10% of the actual values.

RESULTS

The average score of all agents' plans usually gives a good overview of the iterations' progress. In the first iterations, the average score is very low as the system is far from being relaxed. With ongoing iterations, the agents learn how to avoid traffic jams by choosing different routes or by starting their trips at different times of day. Figure 2
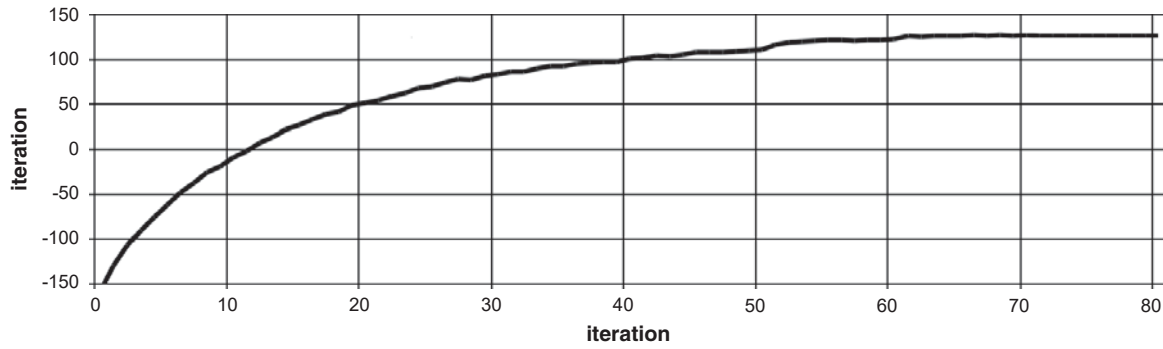
FIGURE 2   Agents' average score of first 80 iterations.

presents the average score of the first 80 iterations. The improvement of the average score is enormous in the beginning, but slows down as more agents find better times and routes for their plans. After Iteration 50, the behavioral module for time adjustment is deactivated, and the behavioral module for route finding is deactivated after Iteration 60. As indicated in Figure 2, the score improves slightly both times when a behavioral module is switched off. This can be explained by the reduction in the number of replanning agents. In the first few iterations, a large number of replanning agents is desirable to move quickly to a better solution than the initial one. But after some iterations, a too large number of replanning agents can lead to instabilities. When every replanning agent searches for the fastest or shortest route from one activity to another, new traffic jams can be initiated as many of the agents select the same link for similar route sections. By switching off behavioral modules

and thus reducing the number of replanning agents, the probability of such traffic jams is reduced, leading to shorter travel times for the agents, and thus to a higher average score. After Iteration 60, each agent selects in each iteration one of its remembered plans for simulation. Assuming that a relaxed state was reached before Iteration 60, this usually leads to a simulation of traffic in a relaxed network with small fluctuations, as can also be observed in the real world.

Adjusting trip departure times and activity durations are the most efficient ways to get a relaxed system. Initially, all agents are assigned a random start time for the first activity and random activity durations, each within a certain range of time. The range depends on the type of activity. Each time an agent replans, it tries to optimize its possible score by reallocating activity durations (27). This leads to a differentiated distribution of trip departure times. Figure 3 presents the number of trip departures over the course of a day. Figure 3a
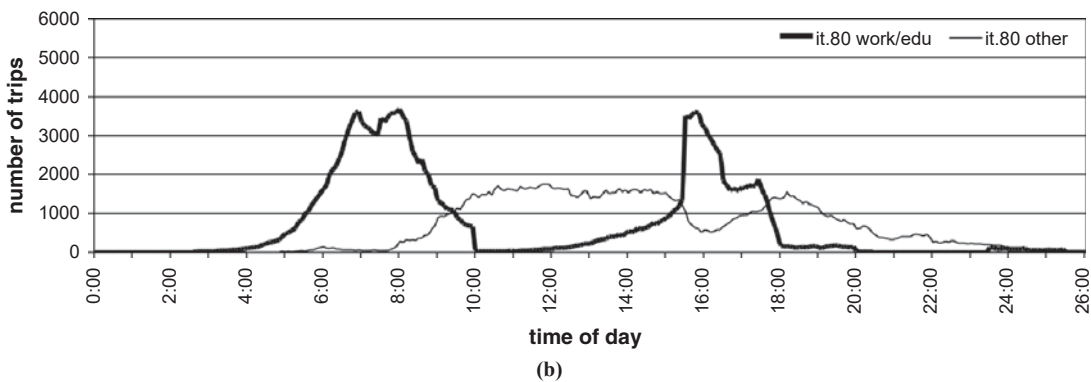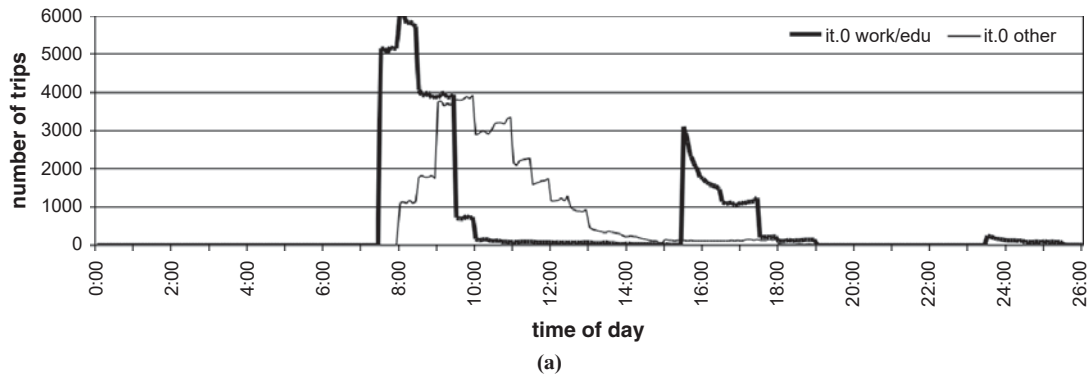

(a)

(b)

FIGURE 3   Number of trip departures over course of day in (a) Iteration 0 and (b) Iteration 80, differentiated by type of primary activity of the corresponding plan.

5

indicates the number of trips for the initially assigned times in Iteration 0, and Figure 3b indicates the number of trips for Iteration 80, where the times were optimized during the iterations nearly to reach a relaxed system. The numbers of trip departures are furthermore differentiated between trips of plans having work or education as a primary activity and plans having other primary activities like shopping or leisure. The agents try to avoid traffic jams in the morning by leaving home earlier than initially assigned. In addition, agents that do not have to work or go to school and thus are more flexible try to avoid the evening peak period by performing activities before or after the peak hour.

Data are available from about 100 measurement stations in Berlin where the traffic passing by was measured. The counts are available in hourly slices, but not all measurement stations have values for every hour of a day.

These counts for each measurement station can be compared with the number of vehicles that travel across the corresponding links in the simulation during the time period of interest. Alternatively, one can calculate an average volume capacity ratio of all measurement stations based on the links' capacities. For this, the capacities of all links with counts are summed for a specified hour. Next, the counts for those links are summed. With these two sums, an average volume-to-capacity ratio can be calculated for the specified hour. The same can be done for the number of vehicles on those links in the simulation. Figure 4a shows the two volume capacity ratios compared over the course of a day. The system can trace the evolution of that number as a function of the time of day.

The average volume-to-capacity ratio is generally lower in the simulation than it is in the real world, except during the morning and evening hours. The missing traffic between morning and evening can be explained by the lack of commercial traffic in the Kutter model. The overestimation of traffic during the peak hours may be explained with the input data containing only tours but not complete day plans. Therefore, there is no temporal relationship between two tours of one person on the same day. It is possible and likely that the times the two activities are performed will overlap. Because of the missing information that one activity can take place only after another one, there is currently no need to perform activities late in the day. Instead, the agents try to accomplish the activities during regular work hours, starting in the morning.

For each measurement station, a relative error can be calculated for every hour for which data are available. The relative error is defined as the absolute difference between real-world and simulated counts, divided by the real-world counts. An average of the relative error over all measurement stations can be plotted. As indicated in Figure 4b, the average relative error changes over the course of the day. It is relatively high at night and improves during the day. The large error at night can be explained by the fact that, during the night, the divisor (real-world traffic counts) is relatively small.

DISCUSSION AND FUTURE WORK

This work shows that it is possible to couple ABDG with multiagent traffic simulations, but the results are not yet satisfying. Although it is possible to reuse internal data from ABDG, the data have, at least
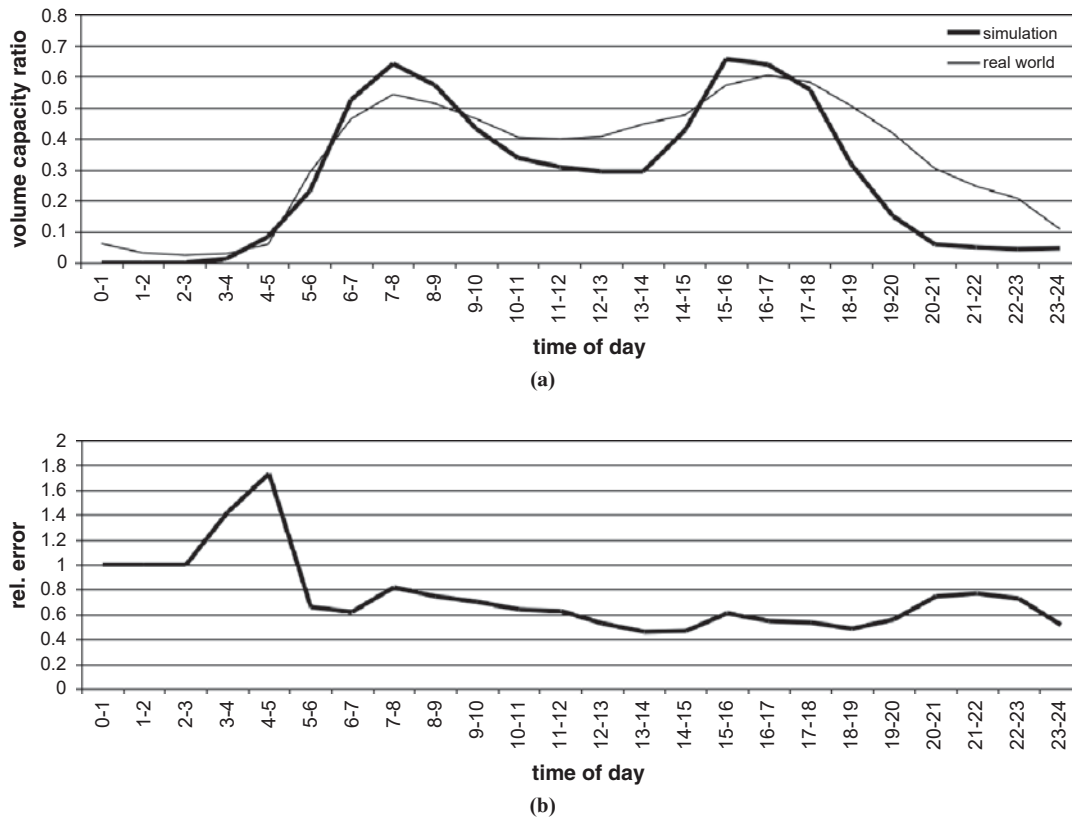

FIGURE 4 Time-of-day-dependent analysis of simulation outcome: (a) comparison of average volume-to-capacity ratios over the course of a day and (b) average relative error when comparing real-world counts with simulated volumes.

in this case, some severe shortcomings. If data should be used from ABDG for multiagent simulations, these shortcomings must first be recognized and overcome. This can be done by modifying the ABDG or by a more thorough postprocessing of the internal data to make them suitable for multiagent simulations.

A major shortcoming is that each agent in the simulation corresponds to a route but not to a "real" person. This can be seen in the number of agents (7 million agents compared with 6 million inhabitants) as well as in the fact that every agent is at home only at the start and end of a plan, but never in between two activities (e.g., having lunch at home). This leads to missing temporal relationships between trips and too little traffic in the evening, as indicated in the preceding section. Combining two or more activity chains into one agent would reduce the number of agents and increase the average complexity of a plan. The combination of several activity chains into one plan must be done carefully and needs some research, as not every combination of activity chains has the same probability. However, the fewer but longer plans might help to relax the pressure in the morning hours and might lead to increased traffic later in the day.

If ABDG is modified to include complete day plans, a huge step is made toward agent-based demand generation. In addition, traditional DTA could make use of the improved ABDG when the traffic assignment is done for a limited time frame only and not for the whole day.

Some traffic segments, such as long-distance traffic, tourists and business travelers, and commercial traffic, are missing. In principle, those segments could be handled by agent-based models similar to the one described in this paper. Alternatively, and arguably more pragmatically, one could account for those trips by adding single trips based on conventional O-D matrices for those travel segments only. This is what the authors intend to do for Berlin.

The results have to be interpreted with regard to Berlin's special history. Its partition into two parts by The Wall in 1961 and the reunification in 1989 led to a city with more than one city center. In addition, the behavior of some parts of the population (mainly of older generations) still differs based on their origin and historical background. This requires special modifications of the behavioral modules and the algorithm used for scoring. The number of modifications remains to be figured out.

Finally, the data for the Berlin scenario must be improved. Some attributes of the network provided by the planning department of the city of Berlin cannot be reconstructed or fully understood, whereas other attributes (like the number of lanes per link) are missing. Heuristics are currently to overcome these shortcomings, but having the correct values from the source of the data would clearly help to improve the results of the simulation.

ACKNOWLEDGMENTS

REFERENCES

1. Beckman, R. J., K. A. Baggerly, and M. D. McKay. Creating Synthetic Base-Line Populations. *Transportation Research A,* Vol. 30, No. 6, 1996, pp. 415–429.
2. Jonnalagadda, N., J. Freedman, W. A. Davidson, and J. D. Hunt. Development of Microsimulation Activity-Based Model for San Francisco: Destination and Mode Choice Models. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1777,* TRB, National Research Council, Washington, D.C., 2001, pp. 25–35.
3. Bowman, J. L., M. Bradley, Y. Shiftan, T. K. Lawton, and M. Ben-Akiva. Demonstration of an Activity-Based Model for Portland. In *World Transport Research: Selected Proc., 8th World Conference on Transport Research 1998,* Vol. 3, Elsevier, Oxford, United Kingdom, 1999, pp. 171–184.
4. Pendyala, R. M., R. Kitamura, A. Kikuchi, T. Yamamoto, and S. Fujii. Florida Activity Mobility Simulator. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1921,* Transportation Research Board of the National Academies, Washington, D.C., 2005, pp. 123–130.
5. Miller, E. J., and M. J. Roorda. Prototype Model of Household Activity-Travel Scheduling. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1831,* Transportation Research Board of the National Academies, Washington, D.C., 2003, pp. 114–121.
6. Arentze, T. A., and H. J. P. Timmermans. *Albatross: A Learning-Based Transportation Oriented Simulation System.* European Institute of Retailing and Services Studies, Eindhoven, Netherlands, 2000.
7. Arentze, T. A., and H. J. P. Timmermans. *ALBATROSS, Version 2.0. A Learning Based Transportation Oriented Simulation System.* European Institute of Retailing and Services Studies, Eindhoven, Netherlands, 2005.
8. Watling, D. Asymmetric Problems and Stochastic Process Models of Traffic Assignment. *Transportation Research B,* Vol. 30, No. 5, 1996, pp. 339–357.
9. Dynasmart home page. www.dynasmart.com. Accessed July 2006.
10. DynaMIT home page. mit.edu/its. Accessed July 2006.
11. Friedrich, M., I. Hofsäß, K. Nökel, and P. Vortisch. A Dynamic Traffic Assignment Method for Planning and Telematic Applications. In *Proc., Seminar K, Vol. P445 of European Transport Conference,* PTRC, Cambridge, United Kingdom, 2000, pp. 29–40.
12. Loudon, W. R., J. Parameswaran, and B. Gardner. Incorporating Feedback in Travel Forecasting. In *Transportation Research Record 1607,* TRB, National Research Council, Washington, D.C., 1997, pp. 185–195.
13. Raney, B., and K. Nagel. Iterative Route Planning for Large-Scale Modular Transportation Simulations. *Future Generation Computer Systems,* Vol. 20, No. 7, 2004, pp. 1101–1118.
14. Ferber, J. *Multi-agent Systems. An Introduction to Distributed Artificial Intelligence.* Addison-Wesley, Boston, Mass., 1999.
15. Jara-Diaz, S. R., and R. Guerra. Modelling Activity Duration and Travel Choice from a Common Microeconomic Framework. *Proc., Meeting of the International Association for Travel Behavior Research,* Lucerne, Switzerland, 2003. www.ivt.baum.ethz.ch.
16. Avineri, E., and J. N. Prashker. Sensitivity to Uncertainty: Need for Paradigm Shift. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1854,* Transportation Research Board of the National Academies, Washington, D.C., 2003, pp. 90–98.
17. TRANSIMS home page. www.transims.net. Accessed July 2006.
18. de Palma, A., and F. Marchal. Real Case Applications of the Fully Dynamic Metropolis Tool-Box: An Advocacy for Large-Scale Mesoscopic Transportation Systems. *Networks and Spatial Economics,* Vol. 2, No. 4, 2002, pp. 347–369.
19. MATSim home page. www.matsim.org. Accessed July 2006.
20. Cetin, N., A. Burri, and K. Nagel. A Large-Scale Agent-Based Traffic Microsimulation Based on Queue Model. *Proc., Swiss Transport Research Conference,* Monte Verita, Switzerland, 2003. www.strc.ch.
21. Balmer, M., M. Rieser, A. Vogel, K. W. Axhausen, and K. Nagel. Generating Day Plans Using Hourly Origin-Destination Matrices. In *Jahrbuch 2004/05 Schweizerische Verkehrswirtschaft* (T. Bieger, C. Laesser, and R. Maggi, eds.), St. Gallen, Switzerland, 2005, pp. 5–36.

22. Bhat, C. R., J. Y. Guo, S. Srinivasan, and A. Sivakumar. Comprehensive Econometric Microsimulator for Daily Activity-Travel Patterns. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1894,* Transportation Research Board of the National Academies, Washington, D.C., 2004, pp. 57–66.

23. Kutter, E. *Integrierte Berechnung städtischen Personenverkehrs–Dokumentation der Entwicklung eines Verkehrsberechnungsmodells für die Verkehrsentwicklungsplanung Berlin (West).* Berlin, 1984.

24. Kutter, E., H-J. Mikota, J. Rümenapp, and I. Steinmeyer. *Untersuchung auf der Basis der Haushaltsbefragung 1998 (Berlin und Umland) zur Aktualisierung des Modells "Pers Verk Berlin / RPlan," sowie speziell der Entwicklung der Verhaltensparameter '86–'98 im Westteil Berlins, der Validierung bisheriger Hypothesen zum Verhalten im Ostteil, der Bestimmung von Verhaltensparametern für das Umland.* Entwurf des Schlussberichts im Auftrag der Senatsverwaltung für Stadtentwicklung Berlin. Berlin, 2002.

25. Raney, B., and K. Nagel. Improved Framework for Large-Scale Multi-agent Simulations of Travel Behavior. Presented at 84th Annual Meeting of the Transportation Research Board, Washington, D.C., 2005.

26. Rümenapp, J., and I. Steinmeyer: *Activity-Based Demand Generation: Anwendung des Berliner Personenverkehrsmodells zur Erzeugung von Aktivitätenketten als Input für Multi-Agenten-Simulationen.* Arbeitsberichte Verkehrssystemplanung und Verkehrstelematik Nr. 06–09, Berlin, 2006.

27. Meister, K., M. Balmer, K. W. Axhausen, and K. Nagel. Planomat: A Comprehensive Scheduler for a Large-Scale Multi-agent Transportation Simulation. *Proc., Swiss Transport Research Conference,* Monte Verita, Switzerland, 2006. www.strc.ch.

28. Charypar, D., and K. Nagel. Generating Complete All-Day Activity Plans with Genetic Algorithms. *Transportation,* Vol. 21, No. 4, 2005, pp. 369–397.

29. Arnott, R., A. de Palma, and R. Lindsey. A Structural Model of Peak-Period Congestion: A Traffic Bottleneck with Elastic Demand. *American Economic Review,* Vol. 83, No. 1, 1993, p. 161.

30. Balmer, M., B. Raney, and K. Nagel. Adjustment of Activity Timing and Duration in an Agent-Based Traffic Flow Simulation. In *Progress in Activity-Based Analysis* (H. J. P. Timmermans, ed.), Elsevier, Oxford, United Kingdom, 2005, pp. 91–114.