Target driven activity planning

Author(s):
Märki, Fabian; Charypar, David; Axhausen, Kay W.

Publication date:
2012-01

Permanent link:
https://doi.org/10.3929/ethz-b-000038469

12-1611: Target Driven Activity Planning

Date of submission: 2011-07-31

Fabian Märki
IVT, ETH Zurich, CH-8093 Zurich
phone: +41-44-633 33 25
fax: +41-44-633 10 57
maerki@ivt.baug.ethz.ch

David Charypar
IVT, ETH Zurich, CH-8093 Zurich
phone: +41-44-633 35 62
fax: +41-44-633 10 57
charypar@ivt.baug.ethz.ch

Kay W. Axhausen
IVT, ETH Zurich, CH-8093 Zurich
phone: +41-44-633 39 43
fax: +41-44-633 10 57
axhausen@ivt.baug.ethz.ch

Words: 5822 words + 5 figures + 1 table = 7322 word equivalents

ABSTRACT

This paper proposes a microscopic travel demand simulation that employs a continuous planning approach with an open time horizon. It uses behavioral targets and the concept of projects to model people's motivation to execute activities. People's behavior originates from a planning heuristic making on the fly decisions about upcoming activities. The planning heuristic bases its decisions on the available activity execution options in the near planning future, its current execution effectiveness and on a discomfort measure derived from deviations between people's performance and their behavioral targets. We illustrate the model through several examples and suggest directions for future research.

INTRODUCTION

Microscopic travel demand simulation softwares simulate virtual people (referred to as agents) individually. For instance, Balmer *(1)* uses agents which choose between different daily schedules. Activities of these schedules are executed and simulation results are handed back to the planning process, allowing agents to improve their schedules based on improved estimates of their generalized costs. This replanning step is repeated until the simulation reaches a stochastic user equilibrium with consistent travel demand and travel cost *(2)*. Simulating agents individually leads to a high computational complexity which often results in computational performance and memory issues. Microscopic models typically introduce restrictive constraints to counter such issues. For instance, Balmer limits the maximum simulation horizon of standard size scenarios to a single day, making it difficult to investigate effects occurring over a period of days or weeks. Another limitation is that agents must commit themselves to a specific day-plan, making it challenging to simulate unexpected events realistically *(3)*. As a consequence, a different simulation approach becomes necessary that is capable to model demand continuously, i.e. agents should be able to make decisions about upcoming activities on the fly and with an open time horizon.

We propose a microscopic travel demand simulation that utilizes behavioral targets to represent agents' decision space. Targets can represent social and cultural norms and are closely related to observed behavior like e.g. execution frequency or time spent for an activity. Since targets use averaged values, they are well suited for modeling recurrent activities. We extend the target based model with the concept of projects to model non-recurrent activities. Projects influence target values during a specific time period, leading to behavioral variability. Agents continuously track their performance and compare it to their behavioral targets using observation windows of different durations. Deviations from the desired behavior cause discomfort which is conveyed to a planning heuristic, making decisions about future activities agents should execute. This enables agents to react spontaneously to unexpected events. At the same time, it also reduces memory consumption because agents do not need to keep track of complete daily schedules.

The remainder of this paper is structured as follows: first, we discuss the model and its behavioral targets. We then introduce the concept of projects and describe how it influences behavioral targets. The next section describes the planning heuristic with its key features and how it utilizes the target based model to decide about future activities to execute. This is followed by a behavior calibration and validation section. We conclude the paper with a perspective on future tasks.

OTHER WORK

Arentze and Timmermans *(4)* introduced need-based theory and proposed a model for activity generation *(5)* that assumes utilities of activities are a dynamic function of needs. Whereas Arentze and Timmermans used needs as people's motivation to execute activities, we see the satisfaction of needs as one possible target in our model. Generally, we assume that people describe their desired performance through measures which are closer to data found e.g. in *(6)*. Axhausen and Schönfelder *(7) (8)* proposed projects as a coordinated set of activities, tied together by a common goal or outcome. Miller *(9)* technically applied projects to organize complex human behavior. We see projects as a mechanism that temporally influences behavioral targets and use it to model non-recurrent activities. We pick up Gliebe and Kim's *(10)* suggestion to use time-dependent utilities and introduce time-dependent effectiveness functions, describing the effectivity of activities and locations towards discomfort reduction. We presented a need-based model in *(11)* that was also designed for a continuous simulation. Our new model drops the need-based approach and introduces measures (we refer to them as *targets*) which are more related to data found e.g. at the Swiss Federal Statistical Office. From this shift, we expect a simplification of model utilization for practitioners. At the same time, it made a complete revision of the concept necessary, resulting in a more comprehensive model and simpler algorithms.

TARGET BASED MODEL

Agents are the central component of our model and represent virtual people. Each agent has a motivation to execute activities and specifies its desired performance through behavioral targets. Deviations to behavioral targets result in discomfort which induces agents to take action against the deviation; higher deviations result in higher discomfort which in turn leads to a higher urge to take action. Agents can reduce discomfort through the execution of activities at different locations. We assume that agents implement activity-location pairs that provide most discomfort reduction. This is similar to Arentze and Timmermans' work *(5)*, where they proposed activity utility as a function of need reduction.

The core assumption of this work is that people have a motivation to execute activities and that they have a perception of their motivation in form of a desired performance. People specify this performance through *behavioral targets* and try to comply with them across *observation windows* of different duration. For instance, a person would like to play 2^{+2}_{-1} hours of tennis about twice per week but not more than six times per month. In this example, the person specifies a *target value* of 2 hours of tennis, a *bandwidth* of $^{+2}_{-1}$ hours, and two observation windows (per week and per month) in which the person tries to comply with the target.

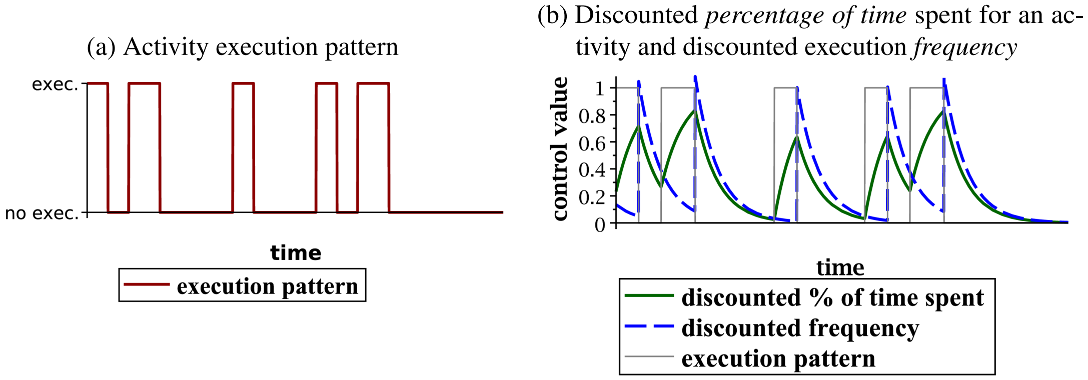
Targets with Observation Windows

This work assumes that people give recent behavior more weight and gradually forget their past performance. The length of the observation window of a target defines the duration of its forgetting process. The proposed model uses *execution frequency* and *cumulative duration* as targets with observation windows. Both targets define a target value (value a person tries to accomplish) and a bandwidth (upper and lower bound of the target value). For instance, a modeler could specify these targets for a sport activity as follows:

- **Frequency:** $2\times^{+2}_{-1}$ **per week.** The desired frequency is twice per week and the agent experiences a limited discomfort in the range of [1..4] times per week. The observation window is one week.
- **Cumulative duration:** $4h^{+2}_{-2}$ **per week.** The desired cumulative execution duration is $4h$ per week and the agent experiences a limited discomfort in the range of [2..6] hours per week. The observation window is one week. Internally, the simulation converts *cumulative duration* targets into *percentage of time* targets which define the percentage of total time an agent should spend for an activity. In our example, the target value is $2 \cdot 4/(7 \cdot 24) = 4.76\%$ with an upper bound of $2 \cdot 6/(7 \cdot 24) = 7.14\%$ and a lower bound of $2 \cdot 2/(7 \cdot 24) = 2.38\%$.

Agents record their performance during simulation and compare these *state values* to target values. State values are exponentially discounted over the observation windows of target values in order to simulate the forgetting process. This is achieved by a convolution with an exponential

FIGURE 1 Illustration of discounted state values



kernel (see Fig. 1). Accordingly, agents weight their recent behavior more and forget their behavioral performance beyond the observation window.

Targets without Observation Windows

The proposed model also enables the specification of targets which do not have a forgetting process. An example is the *execution duration* target. It specifies how long an agent should spend for one activity execution. For instance, a modeler could specify the *execution duration* target for a sport activity as follows:

- **Duration:** $2h^{+2}_{-1}$ **per execution:** The targeted duration is 2 hours per execution and the agent experiences a limited discomfort if it chooses a duration in the range of [1..4] hours.

Effectiveness Function

Similar to Gliebe and Kim's work *(10)*, where they used time-dependent utilities, we introduce a time-dependent effectiveness function. This function informs agents about the effectiveness of activities and locations towards discomfort reduction at a specific time through a value in the range of [0..1]. The effectiveness function is a broad concept and can model different effects. Possible examples are:

- **Shop opening hours for a *daily shopping* activity.** This function takes the value of > 0 when shops are open and zero whenever they are closed. Agents can use this information to either determine if they can shop and for how long or how long it takes until they can shop next time. Since effectiveness functions can be location dependent, it is also possible to model location dependent shop opening hours. Furthermore, this effectiveness function can also contain time dependent information about shop crowdedness. Hereby, we assume that shopping at overcrowded shops is less efficient (smaller value) and therefore takes longer.
- **Daylight intensity for a *sleep* activity.** This function specifies the light intensity. Agents can use this information e.g. as an indication of sleep effectiveness. Hereby, we assume that people sleep at night and have already adapted to their current timezone.
- **Business hours for a *work* activity.** This function can be seen as a cultural norm (cultures may have different business hours) and a social norm (social groups, e.g. professions, may

have different business hours). Agents can use this information e.g. as an indication of work effectiveness. Hereby, we assume that people depend on co-workers to be able to do their work (the degree can differ depending on the profession).

- **Seasonal effects for a *sport* activity.** This function is location dependent and combines different effects like time of the year and weather conditions. As an example, a ski resort can have a hight effectiveness during the winter months after a snowfall whereas the yacht club has a hight effectiveness during the summer months with sunny weather and a good breeze. This enables agents to follow seasonal rhythms because they choose to ski at the ski resort during the winter and to sail at the yacht club during the summer.

PROJECTS

Apart from periodically executed activities (e.g. sleep), people can also have a motivation to execute activities during a certain time period. The motivation and the time period is thereby defined by a special event. An example is the plan to give a party and the necessity to buy extra food before the party starts. In this case, it is the event of having a party that drives people to the shop. We model such events as *projects* which temporally modify values of behavioral targets.

Project Structure

The task is the basic component of a project and has following properties:

- The activity it refers to (e.g. daily shopping),
- The targets it refers to and how much it modifies its values (e.g. *frequency* target: $+1\times^{+0}_{-0}$),
- The time window when the value modifications take place (e.g. 23/07/2011 from 9:00 am to 4:00 pm),
- The location of the activity execution. This parameter is optional since a restriction to a specific location might not be necessary (e.g. location: confectionery).

A combination of several tasks build a project (see Fig. 2). For instance, the tasks *buy dessert* and *buy extra food* build the project *organize party food*. Projects can also have a recursive structure and contain other projects. The project *organize party food* could be re-used, e.g. for the project *organize home party* which also includes tasks like *prepare food* and *clean apartment*. The project *organize home party* could then become a subproject of *organize wedding* together with other tasks (e.g. *pick up guests*) and other projects (e.g. *organize ceremony*). This concept provides a mechanism where tasks and projects can be re-used to build bigger and more complex projects.

Projects and Targets

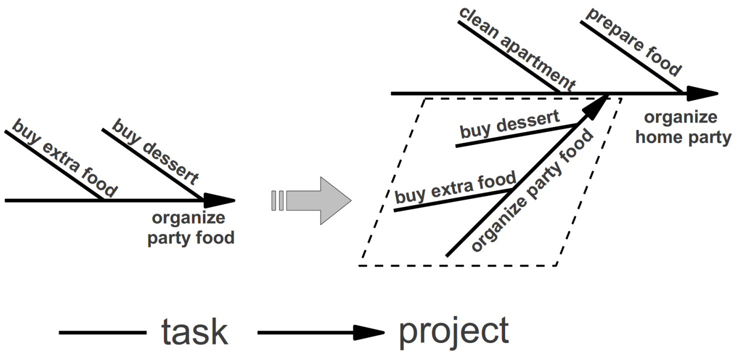
In the party example above, the agent has to spend additional time for the daily shopping activity in order to comply with additional shopping needs. It can achieve this in three different ways:

1. The agent can combine the additional shopping effort with other daily shopping needs,
2. The agent can do an extra trip to a specific shop (e.g. to buy an extra delicious cake at a special confectionery),
3. The agent has the freedom to implement either of the above options.

All three options temporally increase the *cumulative duration* target value and its upper bound during the time window when the agent should buy additional food. This temporally increases

FIGURE 2 Recursive structure of projects enables composition of bigger and more complex projects

(a) The combination of subproject *organize party food* and tasks *clean apartment* and *prepare food* builds project *organize home party*



the discomfort and forces the agent to spend some additional time for that activity. The first option additionally increases the *execution duration* target value and its upper bound. This avoids the induced discomfort of the extra long activity duration. The second option increases the *frequency* target value and its upper bound as well as the *execution duration* target value and its lower bound. This avoids the induced discomfort of the extra trip and the short activity duration. The third option applies all changes to avoid potential discomfort of both possibilities. While the above example temporally increases target values, it is also possible to construct examples which lead to a decrease. An example is a short vacation which temporally decreases the target values of the work activity.

Target Value Modification

Projects modify target values during a specific time period. These changes (difference to actual target values) are also exponentially discounted as it is done for state values (see section Target Based Model). This is necessary because abrupt changes would cause a sudden deviation between target and state value. Consequently, discomfort would also instantaneously increase, leaving agents with no time to react.

PLANNING HEURISTIC

Other approaches to agent-based microsimulations revealed disadvantages like poor performance for large scenarios *(12)*, high computational costs *(1)* or inflexibilities when agents should spontaneously react to unexpected events *(13)*. We consider a planning heuristic as a feasible approach to overcome such limitations. Since a heuristic aims to approximate a good solution, it is possible to use incompletely knowledge about the state of mind and plans of other agents. This is helpful since we plan to simulate our agents in a distributed computation environment *(14)* where global knowledge induces extremely high computational costs. A heuristic also enables agents to react to unexpected events because they can make their decisions spontaneously. One could argue that people seek optimal day plans. However, other authors (e.g. *(15)* and *(16)*) doubt that behavior can be explained as a utility maximization function. One goal of this ongoing work is to demonstrate how far a decision procedure, that approximates a good solution with limited information, can produce real world behavior.

The next section introduces mathematical formulations the planning heuristic uses during its decision procedure. The section thereafter demonstrates the actual decision steps and the application of the mathematical formulations.

Mathematical Formulations

Discomfort

Discomfort levels identify the urgency an agent experiences to take action against them. The discomfort an agent receives from an activity at time t is defined as

$$D(t) = \sum_{k=1}^{n} (f^k_{target}(t) - f^k_{state}(t))^2 \cdot \begin{cases} w^k_1 & \text{if } f_{state}(t)_k \leq f_{target}(t)_k \\ w^k_2 & \text{otherwise} \end{cases} \tag{1}$$

$$w^k_1 = \frac{1}{(f^k_{state}(t) - f^k_{lower-bandwidth}(t))^2} \tag{2}$$

$$w^k_2 = \frac{1}{(f^k_{state}(t) - f^k_{upper-bandwidth}(t))^2} \tag{3}$$

the sum of the squared difference of the target value $f_{target}(t)$ and state values $f_{state}(t)$ of all targets n normalized by the squared difference of the state value and the lower bandwidth w^k_1 if $f_{state}(t)_k \leq f_{target}(t)_k$ or by the squared difference of the state value and the upper bandwidth w^k_2 otherwise.

Discomfort Reduction

The discomfort reduction an agent receives for executing an activity at a specific location is defined as

$$DR(t_{es}, t_{ee}) = D(t_{es}) - D(t_{ee}) \tag{4}$$

the difference of the discomfort $D(t_{es})$ at execution start t_{es} and the expected discomfort $D(t_{ee})$ at execution end t_{ee}. This work assumes that agents prefer activity-location pairs that yield higher discomfort reduction.

Execution Effectiveness

The execution effectiveness an agent receives for an activity at a location is defined as

$$eff(t_{es}, t_{ee}) = \frac{\int_{t_{es}}^{t_{ee}} f_{effect}(t)\,\mathrm{d}t}{t_{ee} - t_{es}} \tag{5}$$

the integral of the effectiveness function $f_{effect}(t)$ between execution start t_{es} and execution end t_{ee} normalized by the activity duration $t_{ee} - t_{es}$. This parameter introduces a preference for efficient time windows and e.g. helps to prevent agents from executing activities during time windows where the agent cannot or can only partially execute the activity (e.g. because the shop closes).

Look-Ahead Measure

Atkinson *(17)* and Ioannou *et al. (18)* highlight the importance of information about future execution options for scheduling problems with time window constraints. Effectiveness functions provide information about future execution options. For instance, shop opening hours inform agents about either if they can shop and for how long or how long it takes until they can shop next time. Agents can use such information to plan ahead and e.g. postpone execution of activities because time windows of other activities are going to close soon.

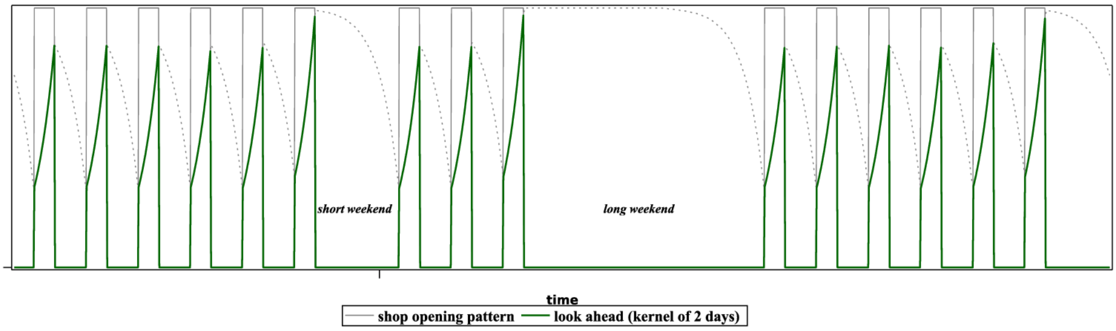
The aim of the proposed look-ahead measure is to provide agents with an awareness of decreasing execution options of activities. A simple approach for the shop opening example would be to measure the percentage of total time available for a potential activity execution in the near future (e.g. within the next four days). This approach has the disadvantage that it does not distinguish between execution options which open soon and options which open further in the future. Applying higher weights to execution options that open sooner avoids this problem. We do this through a convolution of the effectiveness function with an exponential kernel (see Fig. 3), similar to the convolution of state values (see Fig. 1). The look-ahead measure an agent receives for executing an activity at time t is defined as

$$LA(t) = \begin{cases} 1 + w_1 \cdot (1 - \int_0^h (f_{effect}(t+x) \cdot kernel(x))\,\mathrm{d}x) & \text{if } f_{effect}(t) > u \\ 1 & \text{otherwise} \end{cases} \tag{6}$$

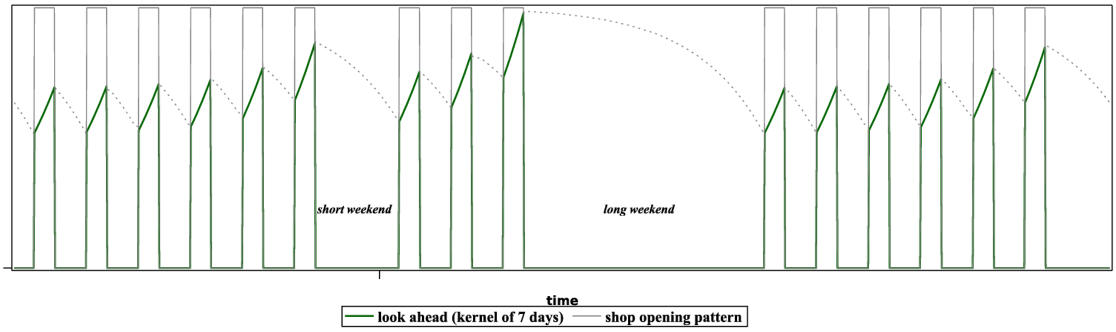
1 plus the multiplication of w_1 with the difference of 1 minus the integral of the effectiveness function $f_{effect}(t)$ multiplied by $kernel(x)$ between 0 and the look ahead horizon h if the effectiveness function $f_{effect}(t)$ yields a higher value than a predefined threshold u (e.g. 0 for closed shops) or 1 otherwise. Since we use $LA(t)$ as a factor in the final heuristic, we designed it in such a way that it yields a value in the range of $[1..w_1]$ (1 if execution is not possible and a value approaching w_1 for decreasing execution options).

FIGURE 3 Illustration of look-ahead measure for shop opening hours with short and long weekends.

(a) Look-ahead measure with a kernel of 2 days. The higher the measure the closer the end of the current shop opening window. The measure is higher before weekends indicating less shopping options in the near future.



(b) Look-ahead measure with a kernel of 7 days. This kernel can differentiate between short and long weekends (measure is higher before long weekend). Choosing the right kernel length is important and we propose a duration of approximately 2 to 3 times the average interval between two activity executions (e.g. $3 \cdot 2$ days = 6 days for daily shopping).



Heuristic Function

The final heuristic value is defined as

$$HF(t_{ts}, t_{es}, t_{ee}) = DR(t_{es}, t_{ee}) \cdot LA(t_{ee}) \cdot eff(t_{es}, t_{ee}) \cdot \frac{t_{ee} - t_{es}}{t_{ee} - t_{ts}} \tag{7}$$

the multiplication of the discomfort reduction $DR(t_{es}, t_{ee})$ between execution start t_{es} and execution end t_{ee} with the look-ahead measure $LA(t_{ee})$ at execution end multiplied by the execution effectiveness $eff(t_{es}, t_{ee})$ and the execution time quota (the time between execution start and execution end $t_{ee} - t_{es}$ normalized by the time between travel start and execution end $t_{ee} - t_{ts}$). Including $DR(t_{es}, t_{ee})$ ensures that agents prefer activities with higher discomfort reduction. Using the look-ahead measure $LA(t_{es})$ guarantees a preference for activities with fewer execution options in the future. Including execution effectiveness $eff(t_{es}, t_{ee})$ ensures that activities and locations which have a higher effectiveness get a higher heuristic value and are therefore preferred by agents. In combination with the execution time quota, this also provides for a simplistic location choice procedure. The execution time quota also prevents agents from changing locations too frequently, leading to a pattern where agents tend to execute more than one activity at a location (e.g. at home).

Decision Procedure

The planning heuristic uses a two-step decision procedure to determine the activity-location pair an agent should execute next. In a first step, it identifies promising activity-location pair candidates. Here, the planning heuristic makes *best guesses* for values which are expensive to compute. In a second step, the planning heuristic computes optimal values of promising candidates and decides to implement the most promising activity-location pair.

First Step

In the first step, the planning heuristic makes the following assumptions to determine promising activity-location pair candidates:

- It uses the free speed travel time to compute travel durations between locations. Computing the exact travel time is expensive because it depends on the current time and could include different computer nodes (we plan to run the simulation on a distributed computation environment *(14)*). We reduce deviations between exact travel time and free speed travel time through a multiplication with a factor. The current version of the simulation uses a constant factor (e.g. 1.2) but we consider a learning process for later versions where agents adapt the factor based on their past experience (e.g. by time-of-day, type-of-location etc.).
- It uses the *execution duration* target value (see section Targets without Observation Windows) as the planned duration. Determining the optimal duration requires a numerical optimization which is computationally expensive.

The planning heuristic computes the discomfort reduction density $HF(t_{ts}, t_{es}, t_{ee})$ for all activity-location pairs using the above mentioned assumptions.

Second Step

In the second step, the planning heuristic uses real travel durations and computes optimal activity durations (using the discomfort reduction density function $HF(t_{ts}, t_{es}, t_{ee})$) for the 30% most promising activity-location pair candidates of the first step. The optimization of the activity duration is done numerically using Brent's method from Press *et al. (19)*.

Since the optimization is computationally expensive, the heuristic tries to reduce the range of valid activity durations before it starts the optimization. The lower and upper bound of the *execution duration* target provide the initial range. Effectiveness functions (see section Effectiveness Function) can further narrow valid upper duration bounds.

$$duration - bound_{upper} = \begin{cases} f_{effect}(y)^{-1} & \text{if } f_{effect}(y)^{-1} < duration - bound_{upper} \\ duration - bound_{upper} & \text{otherwise} \end{cases} \tag{8}$$

The upper duration bound is updated with the time $f_{effect}(y)^{-1}$ when the effectiveness function drops below a predefined value y (e.g. to 0 because shops close), if this time is earlier than the current upper duration bound.

Finally, the planning heuristic searches the optimal activity duration within the lower and upper duration bound and implements the activity-location pair that yields the highest heuristic value.

BEHAVIORAL VALIDATION

This section illustrates the possibilities of adjusting agents' behavior to provide an overview of the calibration mechanism and to show how one can create an agent with a desired behavior pattern.

The person we want to simulate works full-time and lives alone. We assign 12 unconstrained activities to the agent and run the simulation over a period of 10 consecutive weeks with 100 identical copies of the agent. Table 1 provides an overview of these activities, their targets and the deviation of an initial simulation run from the target values. We attune agents' target values and configure almost a full capacity utilization, leaving time for an activity we will add later. The table illustrates that the *frequency* target and the *execution duration* target have the highest deviations. Since we left some time for an additional activity, agents tend to spend more time for their activities (reflected by the *percentage of time* target). Target values of activities with extra long and extra short execution durations experience higher deviations (e.g. sleep and eat). When we disable the execution time quota in the heuristic (see section Heuristic Function), agents can almost meet their target values. This could be a hint that deviations are a result of inflexibilities, preventing agents from freely choosing their activities.

In the reminder of this section, we are going to incorporate *grocery shopping* into the list of activities and adapt its configuration to validate the behavior produced by the decision heuristic.

- In a first simulation, we restrict the execution window (opening hour) from Monday to Saturday, 9:00 am to 6:30 pm. We also define that the agent shops every second day, spends 1.0 hours per shopping trip and and specify an observation window of one week. This results in an *frequency* target value of 3.5 times per week, a *cumulative duration* target value of 3.5 hours per week and an *execution duration* target of 1.0 hour. Fig. 4(a) illustrates that agents perform the activity during the predefined execution windows. Furthermore, they realize that shops will be closed during the night through the look-ahead capability of the planning heuristic. Accordingly, the shopping pattern has a peak in the evenings. In average, about 46% of the agents (thin blue curve and right y-axis) shop per day. The difference to the configured 50% (since agents should shop every second day) results from the restriction of the opening hour during the night and on Sunday. More than the average number of agents refill their food stock on Saturday, also a result of the look-ahead capability that makes agents aware of closed shops during Sunday. Because the food stock lasts about two days, many agents need to refill their stock on Monday.
- In a second simulation, we reuse the configuration from the first simulation and use a project to introduce additional shopping needs on Wednesday. Thereby, we leave it open to the agent to do an additional shopping trip or to combine the additional shopping needs with other shopping duties. This results in an increase of the *cumulative duration* target value and an extension of the *frequency* and *execution duration* target bandwidth for Wednesday (also see section Projects and Targets). The increase of the *cumulative duration* target reaches its peak at lunchtime. Fig. 4(b) shows a prominent pattern change on Wednesday with further effects on Thursday and Friday. Wednesday's peak is slightly after lunchtime, showing that agents need some time to react to the additional need.
- In a third simulation, we reuse the configuration from the first simulation and introduce a second grocery shop. This shop is identical to the first shop with the exception of Saturday afternoon when we define it to be less effective (e.g. to simulate that it is overcrowded on

TABLE 1 Activities, target values and deviations of an initial simulation run. All targets have an observation window of one week.

Activity	Frequency	% of Time (time of a week spent)	Execution Duration
sleep	7 × per week Sim. deviation: +18%	32.5% of a week Sim. deviation: +2%	7.8h per execution Sim. deviation: −4%
work morning	5 × per week Sim. deviation: +6%	14.0% of a week Sim. deviation: +1%	4.7h per execution Sim. deviation: −4%
work afternoon	5 × per week Sim. deviation: +6%	14.0% of a week Sim. deviation: +1%	4.7h per execution Sim. deviation: −4%
eat	21 × per week Sim. deviation: −6%	6.6% of a week Sim. deviation: −2%	0.5h per execution Sim. deviation: +4%
slack time	7 × per week Sim. deviation: +6%	4.6% of a week Sim. deviation: +1%	1.6h per execution Sim. deviation: −6%
personal care	7 × per week Sim. deviation: +4%	4.4% of a week Sim. deviation: +1%	1.1h per execution Sim. deviation: −4%
daily housekeeping	7 × per week Sim. deviation: +4%	4.4% of a week Sim. deviation: +1%	1.1h per execution Sim. deviation: −4%
social contact	2 × per week Sim. deviation: +8%	2.4% of a week Sim. deviation: +1%	2.0h per execution Sim. deviation: −8%
physical exercise	2 × per week Sim. deviation: +8%	2.4% of a week Sim. deviation: +1%	2.0h per execution Sim. deviation: −8%
leisure green	1 × per week Sim. deviation: +12%	2.4% of a week Sim. deviation: 0%	4.0h per execution Sim. deviation: −12%
leisure active	1 × per week Sim. deviation: +12%	1.8% of a week Sim. deviation: 0%	3.0h per execution Sim. deviation: −12%
gardening	1 × per week Sim. deviation: +12%	1.2% of a week Sim. deviation: 0%	2.0h per execution Sim. deviation: −10%

Saturday afternoon). Fig. 4(c) and Fig. 4(d) illustrate that agents react and prefer the more efficient shop on Saturday afternoon (with the exception of the evening rush, when some agents also choose the less efficient shop).

Performance Analysis

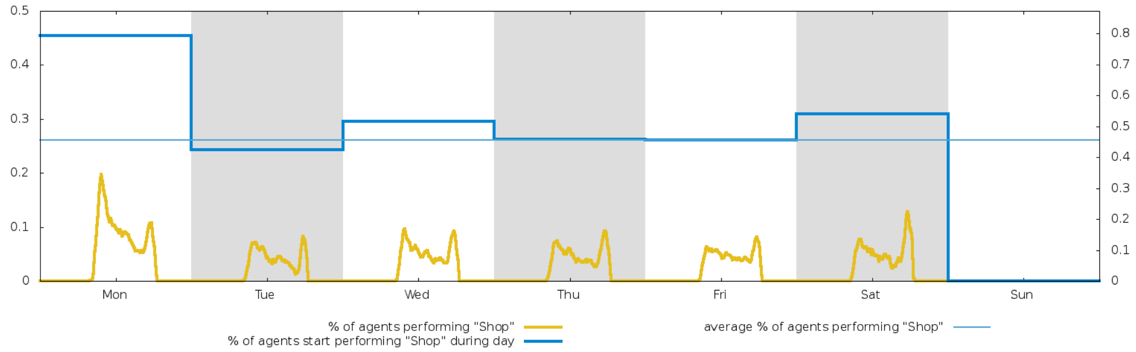
The performance of the proposed simulation is important because we plan to use it for the simulation of very large scenarios. This is the reason why we design our algorithms for a distributed environment. At the same time, it is also important to use algorithms that scale well on the computation nodes. We use the configuration of the first simulation to do an empirical performance analysis of our code. This analysis shows that the code scales in $O(n)$ where n is the number of simulated agents (see Fig. 5). This is a satisfying result because it shows that the simulation time grows linear by the number of agents simulated by a computation node.

OUTLOOK

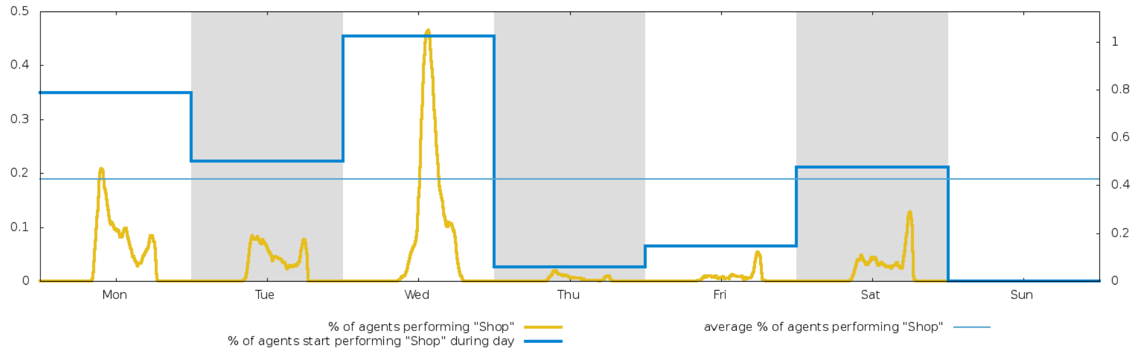
The next step is to validate the proposed model on real data. We plan to use two existing six-week continuous travel diaries (*(20)*, *(21)* and *(22)*) to configure our model. We will focus

FIGURE 4 Illustration of the influence of calibration mechanisms to the behavior of agents. Please note that y-axes have different scales.

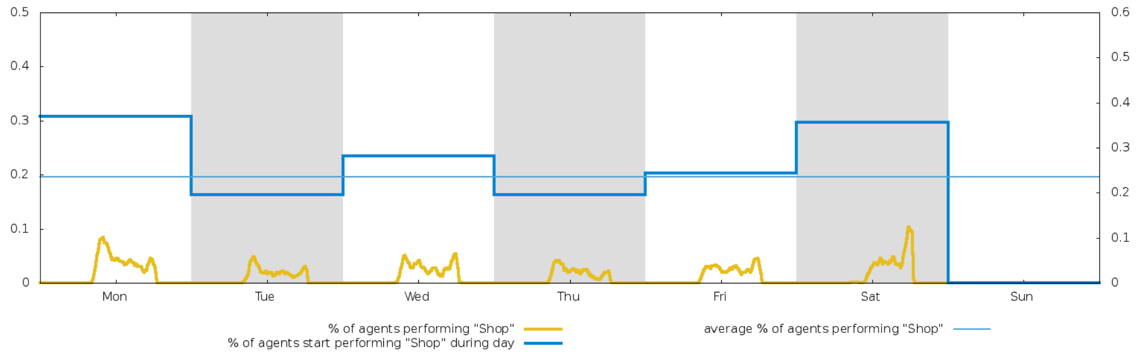
(a) Activity execution is restricted to Monday to Saturday, 9:00 am to 6:30 pm.



(b) A project introduces additional shopping needs on Wednesday.



(c) Add a second shop that is less efficient on Saturday afternoon (pattern of the more efficient shop)



(d) Add a second shop that is less efficient on Saturday afternoon (pattern of the less efficient shop)

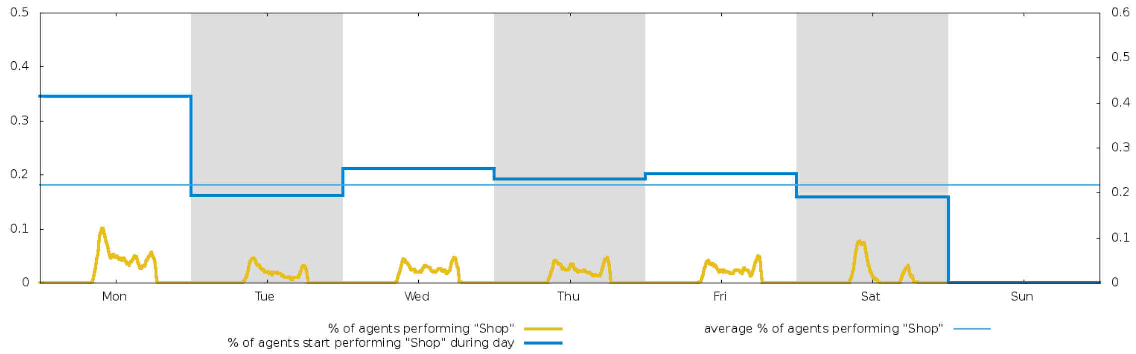
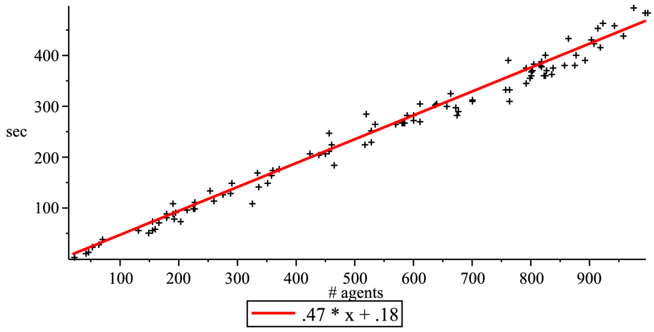


FIGURE 5 Performance analysis of the first simulation scenario (see above)

(a) Mean computing time on a 2.53GHz Notebook. Agents handle 12 activities with a simulation horizon of one week.)



on specific agent types (e.g. working male, housewife, student) and extract time dependent target values from their diaries. Time dependent in the sense that e.g. the *execution duration* target for a home activity has different target values for weekdays and Sundays. This can model situations when people spend more time at home during Sundays compared to weekdays. The *frequency* and *cumulative duration* target extraction will use different observation windows. A long term window (e.g. three weeks for a work activity) extracts long term target values. During the simulation, such targets ensure that agents reproduce long term goals (e.g. about 42 hours of work per month). A short term window (e.g. three days for a work activity) extracts short term target values. During the simulation, such targets ensure that agents reproduce short term fluctuations (e.g. that people have different work habits on Mondays and Fridays and do not work on Saturdays and Sundays). The simulation results will be validated using intra-day behavior measures (numbers, duration and start time of activity types per weekday), inter-day behavior measures (average time intervals and transition probabilities between activity types) and Joh's *(23)* multidimensional affinity-measurement function.

We also need to validate the location choice procedure of our model. The current approach which combines location effectiveness and travel time might not be able to reproduce real location choice distributions. Horni's *(24)* findings provide helpful insights for this task.

The introduction of shared targets will provide the possibility to model household activities. Since we plan to simulate our agents in a distributed computation environment *(14)*, it is computationally expensive for agents to communicate and negotiate about the responsibility for the discomfort reduction of shared targets. Therefore, we will make sure that agents meet in cyclic intervals (at the same location and therefore also at the same computation node) and negotiate a reallocation of their shared targets based on a workload measure (e.g. the time needed to reduce an agent's discomfort multiplied by an agent specific time value). Targets that are assigned to an agent becomes its responsibility. This is a simple and powerful solution because it neither requires a special treatment of shared targets during the decision process nor communication between involved agents.

CONCLUSION

This paper proposes a microscopic travel demand simulation that can continuously simulate agent's behavior. The continuous nature of the simulation will enable an investigation of traffic effects that occur between days and between weeks. Behavioral targets are central for the proposed model. These targets are closely related to statistical data provided by various sources (e.g. Swiss Federal Statistical Office *(6)*), simplifying model utilization for practitioners. We illustrate different targets and their parameters. Some targets have observation windows, enabling performance tracking over different time horizons. Time-dependent effectiveness functions model various effects like shop opening hours or social and cultural norms. We propose using projects to model non-recurrent activities. Projects are limited to a specific time period during which they influence target values. Agents keep track of their performance and compare it to behavioral targets. Deviations cause discomfort which is conveyed to a planning heuristic, making on the fly decisions about upcoming activities agents should execute. This enables agents to react spontaneously on unexpected events. We conclude by validating agents' behavior using different case examples.

REFERENCES

1. Balmer, M. (2007) Travel demand modeling for multi-agent traffic simulations: Algorithms and systems, Ph.D. Thesis, ETH Zurich, Zurich, May 2007.

2. Nagel, K. and G. Flötteröd (2009) Agent-based traffic assignment: Going from trips to behavioral travelers, paper presented at the *12th International Conference on Travel Behaviour Research (IATBR)*, Jaipur, December 2009.

3. Charypar, D., A. Horni and K. W. Axhausen (2009) Need-based activity planning in an agent-based environment, paper presented at the *12th International Conference on Travel Behaviour Research (IATBR)*, Jaipur, December 2009.

4. Arentze, T. A. and H. J. P. Timmermans (2006) A new theory of dynamic activity generation, paper presented at the *85th Annual Meeting of the Transportation Research Board*, Washington, D.C., January 2006.

5. Arentze, T. A. and H. J. P. Timmermans (2009) A need-based model of multi-day, multi-person activity generation, *Transportation Research Part B: Methodological*, **43** (2) 251–265.

6. Swiss Federal Statistical Office (2006) *Ergebnisse des Mikrozensus 2005 zum Verkehrsverhalten*, Swiss Federal Statistical Office, Neuchatel.

7. Axhausen, K. W. (1998) Can we ever obtain the data we would like to have?, in T. Gärling, T. Laitila and K. Westin (eds.) *Theoretical Foundations of Travel Choice Modeling*, 305–323, Pergamon, Oxford.

8. Schönfelder, S. and K. W. Axhausen (2009) Travel as a function of (life) projects, paper presented at the *European Transport Conference*, Leeuwenhorst, October 2009.

9. Miller, E. J. (2005) An integrated framework for modelling short- and long-run household decision-making, in H. J. P. Timmermans (ed.) *Progress in Activity-Based Analysis*, 175–201, Elsevier, Oxford.

10. Gliebe, J. P. and K. Kim (2010) Time-dependent utility in activity and travel choice behavior, *Transportation Research Record*, **2156**, 9–16.

11. Märki, F., D. Charypar and K. W. Axhausen (forthcoming) Continuous activity planning for a continuous traffic simulation, *Transportation Research Record*.

12. Charypar, D. and K. Nagel (2006) Q-learning for flexible learning of daily activity plans, *Transportation Research Record*, **1935**, 163–169.

13. Kuhnimhof, T. and C. Gringmuth (2009) Multiday multiagent model of travel behavior with activity scheduling, *Transportation Research Record*, **2134**, 178–185.

14. Charypar, D., A. Horni and K. W. Axhausen (2010) Pushing the limits: A concept of a parallel microsimulation framework, *Working Paper*, **640**, IVT, ETH Zurich, Zurich.

15. Simon, H. (1955) A behavioral model of rational choice, *Quarterly Journal of Economics*, **69**, 99–118.

16. Schlich, R. (2004) Verhaltenshomogene Gruppen in Längsschnitterhebungen, Ph.D. Thesis, ETH Zurich, Zurich, April 2004.

17. Atkinson, J. B. (1994) A greedy look-ahead heuristic for combinatorial optimization: An application to vehicle scheduling with time windows, *Journal of the Operational Research Society*, **45** (6) 673–684.

18. Ioannou, G., M. Kritikos and G. Prastacos (2001) A greedy look-ahead heuristic for the vehicle routing problem with time windows, *Journal of the Operational Research Society*, **52** (5) 523–537.

19. Press, W. H., S. A. Teukolsky, W. T. Vetterling and B. P. Flannery (2007) *Numerical Recipes: The Art of Scientific Computing*, 3. edn., Cambridge University Press, Cambridge.

20. Axhausen, K. W., A. Zimmermann, S. Schönfelder, G. Rindsfüser and T. Haupt (2002) Observing the rhythms of daily life: A six-week travel diary, *Transportation*, **29** (2) 95–124.

21. Löchl, M., S. Schönfelder, R. Schlich, T. Buhl, P. Widmer and K. W. Axhausen (2005) *Untersuchung der Stabilität des Verkehrsverhaltens*, 1120, Eidgenössisches Departement für Umwelt, Verkehr, Energie und Kommunikation, Berne.

22. Schönfelder, S. (2006) Urban rhythms: Modelling the rhythms of individual travel behaviour, Ph.D. Thesis, ETH Zurich, Zurich.

23. Joh, C.-H. (2004) Measuring and predicting adaptation in multidimensional activity-travel patterns, Ph.D. Thesis, Technical University Eindhoven, Eindhoven.

24. Horni, A., K. Nagel and K. W. Axhausen (2011) High-resolution destination choice in agent-based demand models, *Working Paper*, **682**, IVT, ETH Zurich, Zurich.