Available online at www.sciencedirect.com

ScienceDirect

Transportation Research Procedia 19 (2016) 58 – 69

Transportation Research Procedia

www.elsevier.com/locate/procedia

International Scientific Conference on Mobility and Transport Transforming Urban Mobility, mobil.TUM 2016, 6-7 June 2016, Munich, Germany

Optimization of Individual Travel Behavior through Customized Mobility Services and their Effects on Travel Demand and Transportation Systems

Tim Hilgert [a] *, Martin Kagerbauer [a], Thomas Schuster [b], Christoph Becker [c]

[a]*Karlsruhe Institute of Technology, Institute for Transport Studies, Germany*
[b]*esentri AG, Germany*
[c]*FZI Research Center for Information Technology, Germany*

Abstract

In recent years a lot of new mobility services (e.g. multi- or intermodal travel management and information systems) came up. Today, users have to choose between a high variety of different mobility services and options. To reduce the complexity of these services, customized solutions to support the users are needed. Therefore, we present the development of a mobility assistance system. The assistance gathers information from timetables and real time information systems in public transportation, is connected to mobility services like car sharing, knows the users schedule and only presents relevant information for the ongoing situation. It supports the user's travel behavior by providing information on mode, route or alternative starting times of trips. According to the user's preferences, the assistance may adapt and reorganize the user's weekly activity schedule. To evaluate the impact on travel behavior caused by the mobility assistance, we use the travel demand model mobiTopp. We develop a new module in the model to generate activity patterns. That enables us to get more insights into travel behavior and to evaluate the changes occurred by the usage of the mobility assistance on personal level (e.g. how does the number of trips change) as well as on network level (e.g. how does the morning peak shifts when a certain amount of people is using the assistance).

Peer-review under responsibility of the organizing committee of mobil.TUM 2016.

Keywords: mobility patterns; activity generation; mobility assistance; new mobility services

1. Introduction

Extended technical possibilities (Internet of Things, IoT) allow the creation of new services as well as extensions to existing mobility services. Hence, new mobility services, mobility management systems and the supply-side of transportation gain increasing importance. Nowadays users have a lot of additional mobility options for their daily trips. Furthermore, activity patterns are getting more and more flexible due to flexible working conditions and changing preferences. This causes extended complexity in terms of decision management. Different options may possibly be connected for several trips; thus users have to compare complex alternatives to each other. Therefore, customized decision support can help to identify best options for the individual thus to optimize offered services as well as personal travel needs. In this paper we focus on the development of customized mobility services and their effects on travel behavior by integrating these services in a travel demand model.

* *Corresponding author. Tel.: +49-721-60842256; fax: +49-721-60846777.*
E-mail address: tim.hilgert@kit.edu

Peer-review under responsibility of the organizing committee of mobil.TUM 2016.
doi:10.1016/j.trpro.2016.12.068

Within the research project BiE (Evaluation of integrated Electric Mobility) several partners aim to develop support systems to enhance the integration and acceptance of electric mobility solutions. Electric mobility solutions should be easily integrated in daily routines and people should be able to evaluate and compare them to other mobility solutions, also non-electric. One of these systems under development is a personal mobility assistance which supports users to perform their daily travel demand as best as possible. The mobility assistance gives decision support to reduce the complexity arisen from combinatory tasks of several mobility options. Therefore, it provides information on mode, route and starting time of trips. In addition, the assistance is capable to reorganize activities in a person's schedule in the course of one week such as combining activities e.g. due to spontaneous circumstances like messages from IoT (internet of things) and in general for saving travel time, travel cost or environmental pollution.

This paper describes the architecture of the mobility assistance and their functions. Furthermore, the impacts of such mobility assistance systems will be investigated. Since the assistance may influence daily mobility (e.g. different order of activities due to better travel times, cost-minimization due to combination of trips nearby …) this may affect personal mobility demand as well as influence the transportation system itself. Therefore, a microscopic travel demand model is extended with a synthetic individual activity generation module to evaluate the reorganization of activity patterns when using systems like the mobility assistance. The travel demand model allows the quantification of the mobility assistances' influence on personal daily travel behavior and transportation systems as a whole.

2. Mobility Assistance

The mobility assistance is a distributed system which consists of different subsystems connected to each other. It can, for example, be used through a mobile application (smartphone). The assistance gathers information from different services. Those services include, amongst other data, information about time tables and real time information in public transportation, availability of car sharing options or congestions in the road network. In order to reduce complexity, only relevant information about current traffic situation is presented to the user. Furthermore, the assistance is connected to the user's personal schedule and contains individual user preferences. The system monitors the user's schedule for new events. Besides information display, the system subsequently proposes alternative event planning suggestions to optimize required trips according to the user's preferences. This optimization which depends on user criteria (e.g. minimization of travel times, minimization of travel costs) does also consider previously chosen mobility options.

2.1. Distributed System

In order to compute optimized results over distributed information to each participant and in an individual manner, a complex network with different subsystems is necessary. In this case it is important to understand two things about information processing: 1) the optimization can only be calculated by distributed applications 2) during optimization the state of an information object is uncertain. As stated in the CAP theorem 2) holds true, even if synchronization mechanisms are employed: In distributed applications it is not possible to guarantee that even in case of loss of messages (P) a modifiable data object is available (A) and every participant of the distributed system has a consistent (C) view on this object (Fox and Brewer, 1999).

Considering the aforementioned situation, it is obvious that a distributed application which can provide such a mobility assistance has to operate on distributed data and distributed algorithms. So as a first step requirements have to be prioritized in order to identify a software architecture that best fits the demands. The three most important requirements to create a system for mobility assistance are:

a) **Real Time**: Assistance (optimization support) should be provided to each user in (soft) real time.
b) **Suboptimal Results**: It is more important to display an improvement than to compute the optimum.
c) **Scalability**: The system has to operate even in the event of increasing number of users and data (high load).

In the light of these significant requirements, the project partners have not only been looking for software architectural solutions that may cope with high operational demands but also provide improved flexibility regarding implementation. The latter is also important, since the project is driven by different partners with very different expertise. Thinking in services, each partner could implement and provide expertise as independent services. For about the last two decades the concept of service-oriented architecture (SOA) has been developed to support similar scenarios. Nevertheless, within a couple of years these concepts have been developed a lot further. Currently the concept of microservices draws a lot of attention (Lewis and Fowler, 2014; Richardson, 2014). Microservices can be seen as an architectural style for the design of distributed software systems. Briefly, microservices are an approach to implement a system with a larger number of small services. This is comparable to the prime principles of a SOA. With microservices, however, some more strict demands, not associated with SOA, apply. So each service should be carried out independently (own process space), use its own data (database) and offers lightweight communication mechanisms (often REST) to other services. With regard to the size of a service, it can be stated that it is intended to include only functionalities that can be grouped around a single business capability. The services thus have a professional and highly restricted focus, which in turn directly implements the fundamental principles of service-oriented architectures. In particular, loose coupling, strong cohesion and the separation of concerns can be achieved easily. Microservices promote the following principles in particularly:

- Intelligent services and basic communication (Smart Endpoints & Dumb Pipes)
- Evolutionary Design
- Strict encapsulation (Shared Nothing)
- Decentralized Governance
- Decentralized data storage
- Automation of infrastructure (build, test and deployment processes)

In contrast to classical SOA the microservice approach is based on simple communication mechanisms based on the concept of smart endpoints & dumb pipes. Instead of a highly sophisticated Enterprise Service Bus (ESB) microservices tend to use the architectural patterns Pipes & Filters. It follows that the intelligent processing of messages is performed within the services (Smart endpoint) while the communication is established only on simple mechanisms (Dumb Pipe via REST or messaging via a lightweight, asynchronous communication infrastructure). For this reason, microservices can be easily modified or replaced with a new implementation. This follows the principle of evolutionary design. The strict encapsulation is an important prerequisite to enable evolutionary design. These principles allow each partner to develop functionality independent of others. Furthermore, the idea that each partner can bring in a specialized expertise is fostered, since each partner can develop algorithms and functionality independent from others while relying on a distributed system whenever this is necessary.

In consequence multiple partners may contribute their specific knowledge as needed. The microservice approach allows to create a distributed mobility system, which is capable of handling a high load of users concurrently including a real time traffic analysis. Therefore, the BiE mobility assistance is being developed on basis of microservice architecture incorporating asynchronous event handling and other well-known patterns from software engineering. Within the project consortium the decision was made to build the assistance system on reactive microservices implemented as so called verticles (using the Vert.x framework). The architecture of the Vert.x framework (see Figure 1) contributes to a high modularity of the system and facilitates the integration of new mobility data providers.

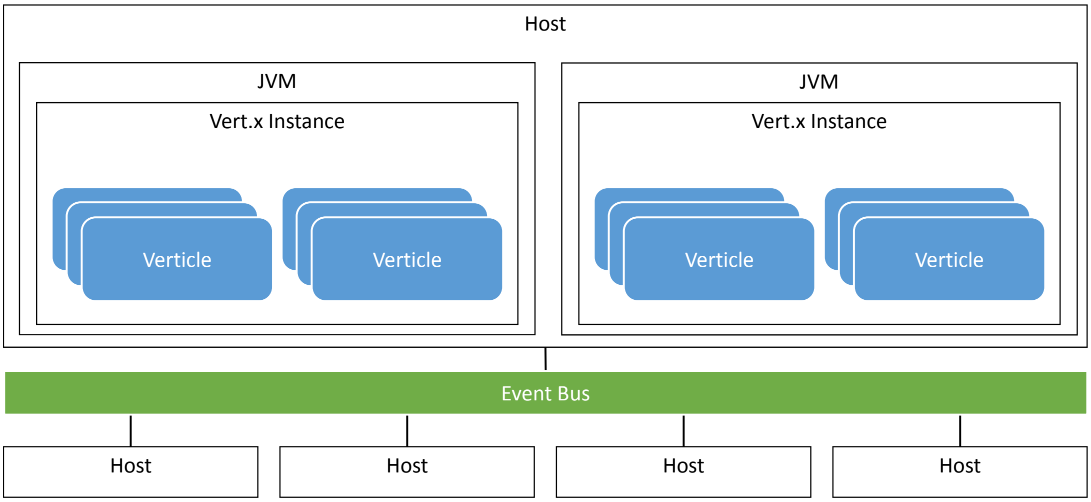


Figure 1: Vert.x - Application Architecture

Vert.x is a lightweight, event based framework that supports the development of distributed applications. Vert.x is polyglot, thus supporting the independence of development teams through the possibility to use various programming languages to implement individual services based on application components (so called verticles). A host can generally provide multiple JVMs, which serve as the execution context of the individual Vert.x instances. Each Vert.x instance again hosts its own set of verticles. Each verticle provides parts of the actual application logic. Usually a verticle will therefore respond to an event, or issue a new event. This is possible in any combination. Communication between the verticles usually takes place via the integrated, distributed event bus. The event bus decouples the verticles. Communication takes place through the typical messaging patterns (publish-subscribe or point-to-point). Also, in addition to the asynchronous communication patterns it is possible to establish synchronous communication patterns.

2.2. System Architecture

The mobility assistance is a complex network of different distributed subsystems that are connected to each other in order to provide the desired mobility assistance functionality for the user. Therefore, the system architecture adapts the principles/concepts of microservice-based architectures as described above (see section 2.1). An overview of the most important components (considering the scope of the paper) is provided in Figure 2. Each component represents either

- a service running of external domains providing data or functionality (white boxes),
- a Vert.X Event Bus (green box),
- a verticle running within the Vert.X instance (blue boxes) or
- the mobile app that provides the user interface for the system (yellow box).

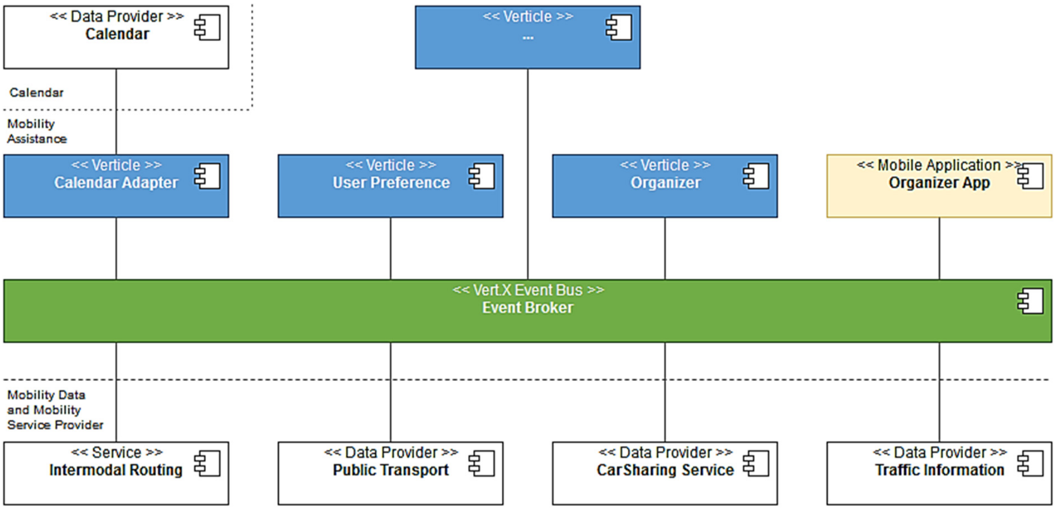


Figure 2: Mobility Assistance Architecture (Overview)

Additionally, each component displayed in Figure 2 can be assigned to a specific domain. All colored components displayed in Figure 2 belong to the mobility assistance domain. These components coordinate monitoring as well as data processing activities and handle the interaction with the customer (users of the Organizer App) and external domains. The remaining components (white boxes) represent external domains that are separated according to the data they provide. Hence, the paper distinguishes between Mobility Data and Mobility Service Provider Domain as well as a Calendar Domain. A detailed description of the components displayed in Figure 2 is given below in Table 1:

Table 1: Component Description of the Mobility Assistance

Component	Type	Functionality
Calendar Service	External Domain/ Data Provider	Calendar service (e.g. Google Calendar or Microsoft Outlook) that provides a set of appointments/activities that should be optimized.
Calendar Adapter	Verticle	Verticle that connects the calendar service (e.g. Google Calendar, …) to the Mobility Assistance. The component monitors the calendar for new events/appointments and pushes a reorganized/optimized order of events back to the calendar service.
Event Broker	Runtime	The event broker is a Vert.X instance that represents the runtime for the Verticle-based components of the mobility assistance.
User Preference	Verticle	The user preference verticle provides the user's individual preferences upon request, thereby enabling other components to take these preferences into account during calculation and optimization processes. The verticle is also responsible for user preference management (Create, Update, Delete).
Organizer	Verticle	The Organizer verticle is the key component of the mobility assistance. It selects and aggregates data provided by other services/verticles and optimizes schedules/activities according to the user's preferences.
Organizer App	Mobile Application	Mobile application represents the user interface to the mobility assistance enabling the user to utilize its functionality (e.g. managing user preferences, optimizing calendars, planning trips…)
Intermodal Routing	Service	Intermodal routing service that allows the calculation of inter- / multimodal routes by taking into consideration the user's preferences.
Public Transportation	External Domain/ Data Provider	External service providing connection detail of Public Transportation Providers (timetables, delays,…).
Car Sharing Service	External Domain/ Data Provider	External service providing the availability of Car Sharing options
Traffic Information	External Domain/ Data Provider	Traffic information service that provides up-to-date traffic reports (regarding accidents, congestion, etc.) to the mobility assistance

2.3. The Organizer Verticle

A specialized Organizer verticle (service) is the central component in this distributed system. While the mobile application (Organizer App) represents the user interface of the mobility assistance, the Organizer verticle comprises the actual optimization and reorganization functionality. The Organizer verticle will regroup the user's activities and appointments when possible in order to optimize the travel demand according to the user's preferences. These preferences comprise information on the user's preferred modes as well as the requested goal of the optimization (costs, travel time, total travel demand …). However, in order to implement the functionality, the Organizer has to interact with other components of the mobility assistance system as well as external domains. This applies especially for the Calendar Adapter verticle, the User Preference (verticle) and the Intermodal Routing service. In the following we will a) provide a short overview on the processing steps performed by the Organizer verticle (see Figure 3) and b) present the chosen optimization protocol (partly shown by Figure 3)..

The calendar optimization is an event-driven process (e.g. regarding a user's travel demand for a certain day) which can be triggered by different types of events. In addition to an optimization request initiated by the user (see Figure 3), it is also possible to retrieve new appointments or messages which trigger appointments automatically. In both cases the optimization process has to be initiated. Before the Organizer starts the actual optimization process, the user's appointments/activities as well as the preferences are loaded using the Calendar Adapter verticle and the User Preference verticle respectively. Afterwards, the Organizer requests routes from the intermodal routing services to connect appointments as well as the user's start position. Therefore, a request with start location/time and the user's travel preferences is transmitted to the Intermodal Routing service. Based on the user preferences the Routing service returns a route in consideration of public and private motorized transportation as well as intermodal transportation. For each route the Organizer applies a utility function to determine a measure for this route. The utility function is based on the user's preferred goals such as time, cost, CO_2 or preferred modes of transportation. Then an optimized trip chain (maximizing utility/reducing the subjective costs) covering all appointments is calculated by the Organizer. Finally, start and end dates of the appointments are updated according to the selected route and pushed back to the Calendar Adapter verticle that updates connected calendars.

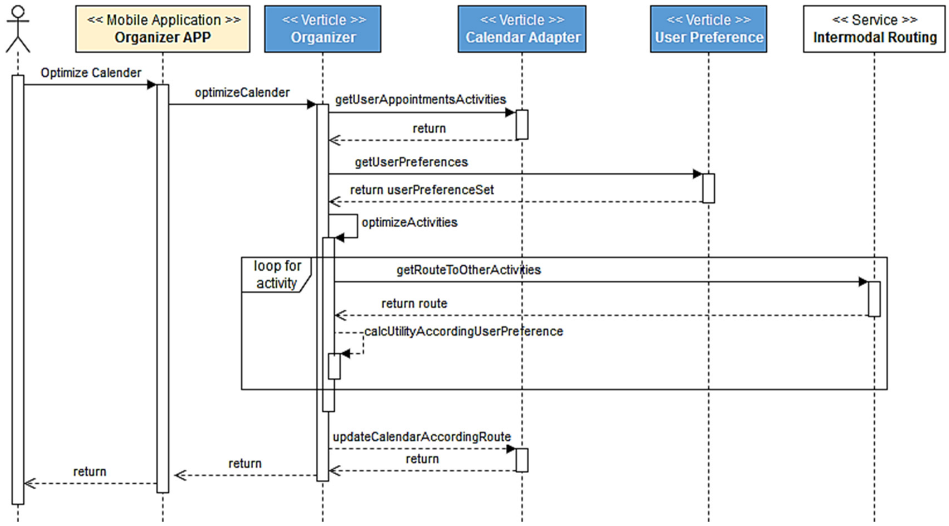


Figure 3: Calendar Optimization

The actual calendar optimization problem can be modelled as travelling salesmen problem (TSP); hence it is non-deterministic polynomial-time hard (NP-hard) and we may transform it to a graph as follows:

$$G = (L, R, c) \tag{1}$$

The set $L = LS \cup A$ comprises the nodes within the graph. LS represent the user's start location whereas A is a set of appointments returned from the Calendar Service. Set $R = L \times L$ are the edges within the full graph representing the routes which connect all appointments as well as the start position with each other. The cost function c assigns the user's individual costs to a route $r \in R$. It is based on a utility function that evaluates the route returned from the intermodal routing service according the user's preferences. The lower the value of c is the better it fits the user's preferences. The objective function uses the variable x_{ij} to indicate whether or not the route connecting L_i and L_j is part of the solution.

$$x_{ij} = \begin{cases} 1 & if\ node\ L_j\ is\ visited\ after\ node\ L_i \\ 0 & else \end{cases}$$

The formalized representation of the corresponding problem is formulated as:

$$\min \sum_{i}^{n} \sum_{j}^{n} c_{ij}\ x_{ij} \tag{2}$$

subject to

$$\sum_{j=1}^{n} x_{ij} = 1 \qquad \forall\, i \in L \tag{3}$$

$$\sum_{i=1}^{n} x_{ji} = 1 \qquad \forall\, j \in L \tag{4}$$

$$\sum_{i \in SR} \sum_{j \in SR} x_{ij} \leq |SR| - 1 \qquad \forall\, SR \subset L \tag{5}$$

$$x_{i,j}\ \in \{0,1\} \tag{6}$$

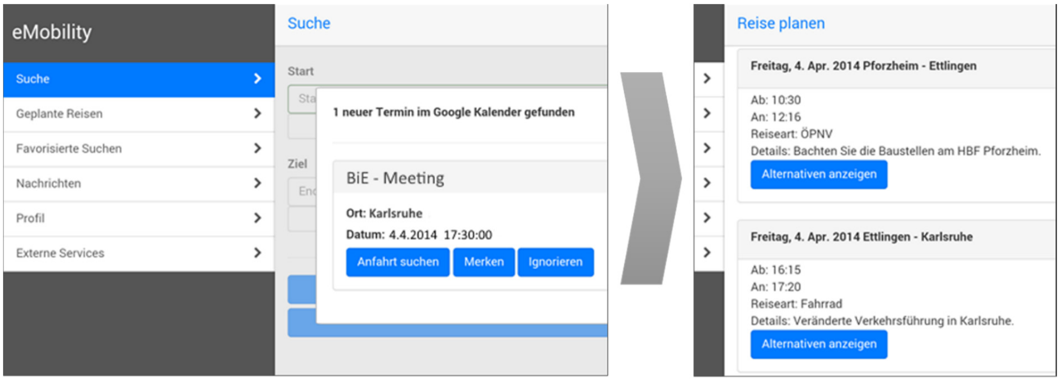
$$i, j = 1, \dots, n \tag{7}$$

Formula (3) and (4) ensure that each node is only visited once whereas (5) defines the sub-tour elimination constraint. In order to solve the rescheduling problem, i.e. find the shortest Hamiltonian cycle through all nodes, our prototype implementation of the Organizer verticle utilizes heuristics. At the beginning the chosen approach calculates a first valid solution using the nearest neighbor heuristics (Rosenkrantz et al., 1977). The given algorithm is a greedy algorithm that takes a directed, weighted graph as input parameter and returns a Hamiltonian cycle. This intermediary result, which in general does not provide an optimal solution, is improved using the 2-opt heuristics, a local search algorithm (Croes, 1958). As mentioned at the beginning the system is still under construction. Thus the optimization protocol and the algorithms within the optimizer component and will be further evaluated and might be subject to change in due course of the project. In the following development stages we plan to extend the current procedure with adaptive algorithms in order to directly incorporate analysis results into our calculations. Since both protocol and algorithms have to deal with events one option is also to employ so called online optimization algorithms (Borodin and El-Yaniv, 2005).

2.4. The Organizer App

Based on the general overview presented above a concrete example will be discussed in the following. Therefore, we will consider a user with the following user preferences, i.e.: a) minimize travel demand, b) public transportation as preferred means of transportation, as well as c) bike sharing as alternative means of travel for short distances. All preferences have been set using Organizer App and are considered during the optimization process. Within this example the user is currently located in Pforzheim and is supposed to attend a meeting in Ettlingen. The Organizer has already calculated a trip using public transportation.

When a new event is added to the user's calendar the system will extract date, time, duration and location of the new event and compare it to other existing events that could be reorganized to one trip. Then the user receives a notification regarding the new appointment and is requested to decide whether or not he wants to optimize his schedule. Within the given example the user receives a notification about a BiE project meeting taking place in Karlsruhe (see Figure 4 a).



a) b)

Figure 4: Notification (a) and trip planning (b) of the Organizer App

In case the user accepts the appointment the Organizer starts the optimization and attempts to regroup the new event around events with the same location or local proximity in order to suit the user's preferences. In this case, the Organizer will extent the current trip and append the new appointment in Karlsruhe right after the appointment in Ettlingen. Additionally, the Organizer proposes a route as well as means of transportation for each trip, while keeping track of relevant traffic information like construction sites or altered traffic routing (see Figure 4 b). The trip and detailed view on the associated routes proposed by the Organizer can be reviewed and altered by the user (Figure 5). Finally, the user has to confirm the new event dates as well as the trip with the corresponding routes. Due to the fact that the Organizer optimizes the calendar according to the user's preferences generally there is no need for change. However, the user can always edit the trip and associated routes proposed by the Organizer, i.e. changing appointment dates, routes or means of transportation. In this case the appointment was appended to an existing trip, thereby minimizing the travel demand and optimizing the route according to preferred means of transportation (public transportation and bike).

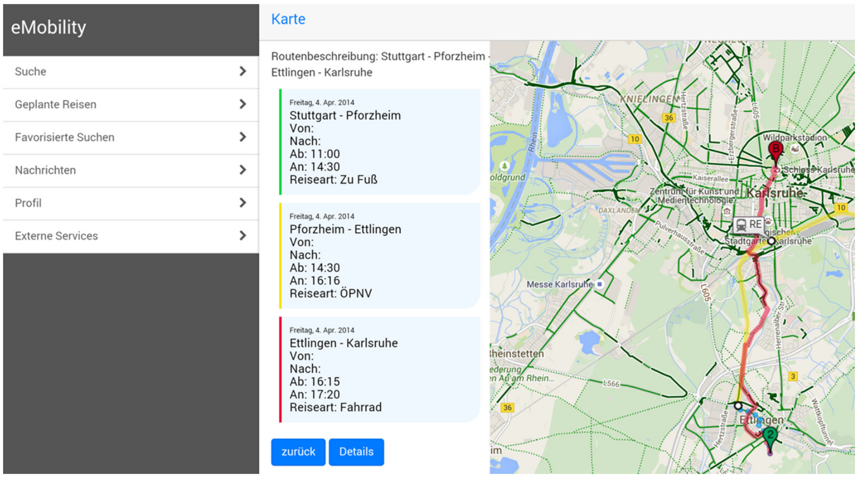


Figure 5: Route Visualization

3. Impacts on Travel Behavior

As shown in the previous section, the mobility assistance will influence travel behavior of people. It will modify their travel behavior in terms of changing activity patterns or destination and mode choices. Since these modifications are user-specific and may influence the user's activities in different ways depending on the optimization strategy, e.g. cost or travel time optimization, we use a travel demand model to simulate and evaluate the behavior changes occurred by the assistance.

In recent years, the agent-based, microscopic travel demand model mobiTopp (Mallig et al., 2013) has been developed at the Institute for Transport Studies at the Karlsruhe Institute of Technology. mobiTopp simulates the travel demand (all trips) including the choices of activities, destinations and modes. The model is used to simulate the travel behavior of all inhabitants of the planning area (in this case the Greater Stuttgart Region in Germany with 2.7 Mio inhabitants) in the course of one week. Figure 6 shows an overview of the mobiTopp framework.

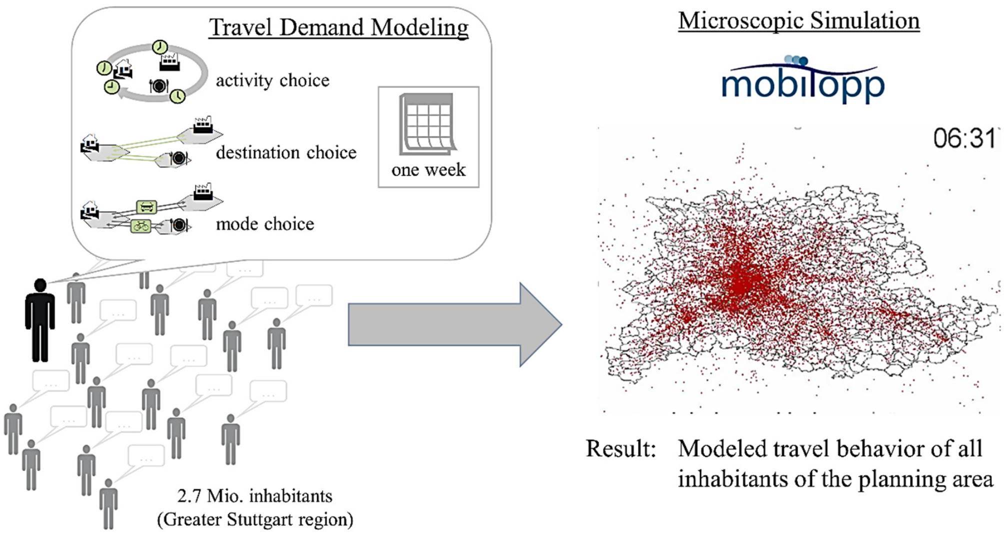


Figure 6: Overview of the mobiTopp framework

3.1. actiTopp – A new activity generation module for mobiTopp

Answering the question of how the assistance affects and shifts mobility patterns, the activity choice part of the actual model implementation needs to be further developed. Until now, the agents' activity schedules in the model are derived from a travel survey in the Greater Stuttgart Region. During the population synthesis part of the model, activity schedules from the survey are selected according to the households of the population. While we do not "generate" the schedules as a function of sociodemographic or other personal circumstances, it is difficult to evaluate changes that occur as a result of the mobility assistance (see section 2.3). Actually the model is not sensitive to different measures that influence the travel behavior in terms of changing activity patterns. We have no mechanisms to model activity choice and therefore we are not able to evaluate measures. Hence, we develop a new activity generation module to get more possibilities to influence activities, to better understand and evaluate the impacts of different parts of the activity generation process. That means we will synthetically model activity schedules for the households/persons in mobiTopp. The activity schedules will be sound concerning the week and household context.

When investigating the current state of the art concerning modeling travel demand, especially activity modeling, a lot of models can be mentioned. Travel demand models usually have different emphases. Some focus on the individual's activity patterns and try to model explicitly the decision process of human beings concerning travel demand. They try to predict activity choices using different approaches. Usually, utility-based and rule-based modeling techniques are distinguished. Utility-based generally follow the paradigm of utility-maximization of people. Some examples of utility-based models are the Boston/Portland Model (Bowman, 1998), CT-RAMP (Davidson et al., 2010), TASHA (Roorda et al., 2008) or CEMDAP (Bhat et al., 2004). Rule-based models try to overcome the fact that people always want to optimize utility and use rule-based approaches like decision trees for the generation of activities instead. One model of this type is ALBATROSS (Arentze et al., 2000). Despite the focus on individual activity patterns, in recent years, general frameworks came up. They focus on both, demand and supply side. They try not to model individual decisions only but also the supply side in terms of the network people are using as well as traffic assignment. Some example of these frameworks are MATSim (Balmer et al., 2008; Raney and Nagel) or SimMobility (Adnan et al., 2016).

While there are a lot of activity based travel demand models used all over the world, there are only a few models that cover more than one day and, to the knowledge of the authors, there is no other comprehensive model that by default covers a complete week. Concerning the modeling of a week, there were some experimentations with MATSim (Horni and Axhausen, 2012; Ordonez M. et al., 2012) and also the SimMobility model offers a learning day-by-day module (Adnan et al., 2016) but currently no published use case for that is known to the authors.

The mobiTopp framework also uses a utility-based approach like other models do. It has been designed as a system of integrated demand and traffic flow simulation (Schnittger and Wittowsky, 2002; Schnittger and Zumkeller, 2004) and has been further developed in recent years (Kagerbauer et al., 2016; Mallig et al., 2013).

Since mobiTopp simulates a whole week, approaches to model activity schedules within one-day travel demand models are not sufficient. They cannot simulate the stability and variability of personal activity schedules within a period longer than one day. Therefore, we expand a known approach by several components to reflect the travel behavior over a week. Modeling a full week results in a higher complexity concerning the activity generation module. Taking 7 one-day activity schedules in a row is not sufficient. Since there are elements of the activity schedule that are similar day by day (e.g. leaving in the morning for work mostly at the same time) but also others that do not occur every day (e.g. shopping or sport activities) some additional complexity needs to be implemented into the model to reflect the personal behavior over a week.

One approach used in one-day models in practice for a longer time (e.g. in the Boston/Portland model, SimMobility, CT-RAMP) is "The Day Activity Schedule Approach to Travel Demand Analysis" (Bowman, 1998). To model the travel demand, the approach of Bowman uses an activity schedule for one day that reflects the daily routines of a person. Since there is an infinite number of combinations to determine an activity schedule for one day, Bowman suggests splitting the selection of a specific activity schedule into a series of simpler decisions. This approach results in a stepwise modeling of a complex activity schedule for one day. For the modeling decisions, logit and nested logit models are used. The concept of Bowman starts with the decision on the primary activity of the day followed by different steps that model the primary tour structure and other tours during the day.

For our new module, called actiTopp, we use the basic idea of the stepwise modeling and enhance it on relevant parts concerning the modeling of activity schedules for a whole week. Table 2 gives an overview of the different steps actiTopp is performing to model activity patterns.

Table 2: Component Overview of actiTopp

step	modeling level and decision			
	persons (week)	**day**	**tour**	**activity**
1	# working days # education days			
2		type of main tour		
3		# tours before # tours after		
4			tour types	
5			# activities before # activities after	
6				activity types
7	time budgets for different types			
8				activity durations
9	main start time			
10			start time	

The structure basically follows the approach of Bowman. We used binomial as well as multinomial logit approaches to model the decisions of different steps. Nevertheless, a lot of decisions cannot be modeled using a logit model only, e.g. activity durations (step 8). Since a logit model needs to be configured for different, discrete choices, a high time resolution is difficult to achieve. To model the duration with high accuracy (by minute), we used in these cases a combination of a multinomial logit model to determine the duration class (e.g. 30-60 minutes) and a random draw from the exact distribution of durations in this duration class. Using this method, we are able to model activity duration as well as start times with the accuracy of one minute.

To extend the approach of Bowman, we added decisions on personal level that reflect decisions for the whole week. Step 1 models the number of working and educations days per person. Since this attribute determines a high share of daily routines, the results of this step are used in the following to determine the type of the main tour. Another example is given by Step 9 of actiTopp. Step 9 models the main start time for working or educations main tours. The result of this step is used in the following step when starting times are determined.

Example: Step 9 results in a starting time class between 8:00 a.m. to 9:00 a.m. for working main tours. In Step 10, five working main tours are modeled since the person goes to work on five days within the week. Step 10 now first decides whether the start times of these tours are located within the main start time (result of step 9) or not. Using this technique enables us to improve the stability of start times over the different days of the week and to represent better daily routines of people that are typical over the period of one week.

Currently, we are in the implementation phase of this model into the mobiTopp framework. We estimated the model using the data of the German Mobility Panel (MOP) (Weiss et al., 2016; Zumkeller and Chlond, 2009). The MOP-database of 2004-2013 that we used for the estimation of the different model steps shown above consists of over 17,800 week activity schedules. Apart from the reported schedules we could use enhanced socio demographic information and household context information to estimate the model steps. After the implementation we will apply the model to the population of the Greater Stuttgart area in mobiTopp.

3.2. Evaluating the impacts of the mobility assistance on travel behavior

Using the new activity generation module actiTopp, we are able to synthetically generate activity schedules for a whole week. In the context of the project, the schedules of the mobiTopp-agents are used by the mobility assistance. The schedules feed as synthetic input into the assistance. The assistance then works on these schedules, meaning that the schedules will be optimized by the assistance (see section 2.3) and suggestions will be given where to adapt activities in different manners due to different circumstances like congestion, availability of different options or personal needs. This results in different adapted activity patters (see point 4 in Figure 7) per person, e.g. optimized by travel times needed or travel costs minimized.

Subsequently, we will integrate these modified patterns again into mobiTopp to evaluate the suggestions of the assistance. Since we usually have different alternative schedules returned by the assistance, we will evaluate different scenarios within the project: A scenario where the agents only accept the cost-minimized patterns, a scenario where they choose randomly between the original pattern (no change), the cost and travel time optimized pattern, … Future research needs to be done on the real acceptance of people to the suggestions by a user survey. In this project we first evaluate potential effects given some assumptions. However, the usage of the travel demand model allows us to easily scale the market share of such an assistance and to evaluate possible effects within the network. Analyzing the modified patterns after optimization by the assistance, we can evaluate changes that impact the different steps of the activity generation module. What kind of tours are changing? How do they change? For each difference between the original and the modified pattern, we can determine the impacts and evaluate how the change results in different travel times (e.g. per day, over the week …) or in different loads within the network.

Figure 7 shows the interaction between the travel demand model mobiTopp and the mobility assistance developed within the project.

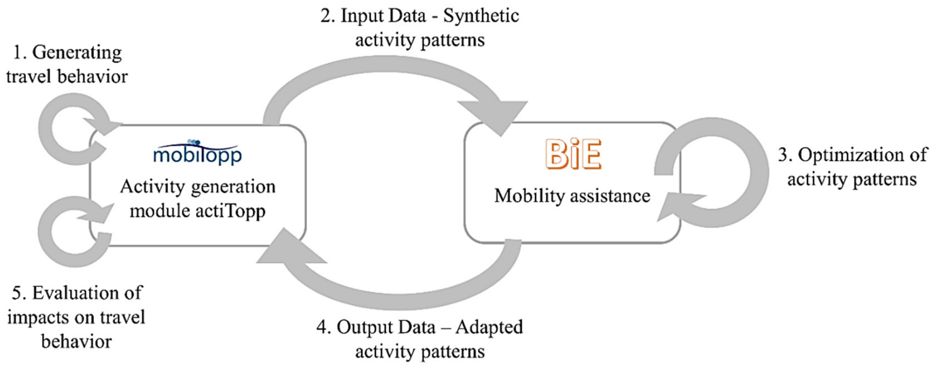


Figure 7: Interaction between mobiTopp and the mobility assistance

4. Conclusions and Outlook

In result we expect that people will have enhanced and more explicit opportunities to adjust their mobility behavior. The mobility assistance is a central part to support this. Regarding mobility assistance, the mobile application is the key component to improve user acceptance and user experience. This mobile application represents the user interface which will provide information to users and decision support functionalities. Of course decision support can only be achieved by a distributed system which is able to: a) manage information on global traffic conditions, b) store information about individual user preferences and c) calculate optimizations regarding planning and travel optimization. All these functionalities are currently under development within the joint research project BiE.

Given the enhanced possibilities to manage individual travel behavior, we believe that the mobility assistance is capable to influence travel demands. The average amount of optimization potential (e.g. travel time saved) is evaluated by the travel demand modeling software mobiTopp. We are currently implementing the mobility assistance as well as the activity generation module. Since we evaluate the suggested optimized mobility patterns by the assistance after model implementation and compare them to the original patterns (see section 3.2), we will be able to answer the following questions:

- How does the number of trips, trip distances, trips times and modal split change on individual level?
- How does traffic volumes shift on infrastructural level when a certain market penetration of the system is given? How does the system's modal split change? Will public transportation increase? Will the shifts in modal split rely to more sustainability?
- How do traffic volumes change over time? Will there be a more even distribution over the day? What about morning and evening peaks?
- How do changing trip lengths and modal split changes affect the environmental balance?

In this project we develop a complex mobility assistance system. After the currently ongoing implementation phase of the project, we are able to investigate possible effects of the mobility assistance regarding travel behavior and transportation systems.

Future research activities may trigger effects on both, the assistance and the simulated behavior analysis. In a subsequent step we will evaluate the acceptance of suggestions made by the mobility assistance, e.g. in a user survey. As shown in this paper the mobility assistance does not necessarily compute the optimum but constant improvements. In future development stages we plan to extend the current procedure with adaptive algorithms (i.e. the algorithm's behavior is changed based on information available at run-time). This allows for a direct incorporation of analysis results (see section 3) into our calculations within the Organizer (see section 2.3), thereby improving our decision support. This will also lead to higher precision of simulation results. Furthermore, in respect to these findings, we consider additional projects to examine how the acceptance of suggested improvements can be increased. Currently we assume that acceptance is influenced by the type of suggestions (e.g. work versus free time activities) and that time and way of presentation (e.g. in form of gamification). We plan to examine this deeply in next projects. Associated to this are cultural and social aspects, like: What kind of suggestions is accepted by whom? Concerning the evaluation, thoughts about the implementation of a feedback mechanism may valuable in addition to the assistance. Regarding gamification, it might be worth to display system wide effects based on individual behavior (people might want to know what kind of effects a certain suggestion may have).

Acknowledgements

The project of this publication has been funded by the German Federal Ministry of Education and Research, funding code EM1CLU039. The responsibility for the content is up to the authors.

References

Adnan, M., Pereira, F.P., Azevedo, C.M.L., Basak, K., Lovric, M., Raveau, S., Zhu, Y., Ferreira, J., Zegras, C., Ben-Akiva, M., 2016. SimMobility: A Multiscale Integrated Agent-Based Simulation Platform, in: TRB 95th Annual Meeting Compendium of Papers, Washington, D.C.

Arentze, T., Hofman, F., van Mourik, H., Timmermans, H., 2000. ALBATROSS: MULTIAGENT, RULE-BASED MODEL OF ACTIVITY PATTERN DECISIONS. Transportation Research Record (1706), 136–144.

Balmer, M., Meister, K., Rieser, M., Nagel, K., Axhausen, K.W., 2008. Agent-based simulation of travel demand: Structure and computational performance of MATSim-T, in: 2nd TRB Conference on Innovations in Travel Modeling, Portland.

Bhat, C.R., Guo, J.Y., Srinivasan, S., Sivakumar, A., 2004. Comprehensive Econometric Microsimulator for Daily Activity-Travel Patterns. Transportation Research Record (1894), 57–66.

Borodin, A., El-Yaniv, R., 2005. Online computation and competitive analysis, 1st ed. Cambridge Univ. Press, Cambridge, 414 pp.

Bowman, J.L., 1998. The Day Activity Schedule Approach to Travel Demand Analysis. Dissertation, USA, 185 pp.

Croes, G.A., 1958. A Method for Solving Traveling-Salesman Problems. Operations Research 6 (6), 791–812. 10.1287/opre.6.6.791.

Davidson, W., Vovsha, P., Freedman, J., Donnelly, R., 2010. CT-RAMP family of activity-based models, in: Proceedings of the 33rd Australasian Transport Research Forum (ATRF).

Fox, A., Brewer, E.A., 1999. Harvest, Yield, and Scalable Tolerant Systems, in: Proceedings of the The Seventh Workshop on Hot Topics in Operating Systems. IEEE Computer Society, Washington, DC, USA, pp. 174–178.

Horni, A., Axhausen, K.W., 2012. MATSim Agent Heterogeneity and a One-Week Scenario. ETH, Eidgenössische Technische Hochschule Zürich, IVT, Institut für Verkehrsplanung und Transportsysteme.

Kagerbauer, M., Mallig, N., Vortisch, P., Pfeiffer, M., 2016. Modeling Variability and Stability of Travel Behavior in a Longitudinal View Using the Agent Based Model mobiTopp, in: TRB 95th Annual Meeting Compendium of Papers, Washington, D.C.

Lewis, J., Fowler, M., 2014. Microservices. http://martinfowler.com/articles/microservices.html. Accessed 12.02.16.

Mallig, N., Kagerbauer, M., Vortisch, P., 2013. mobiTopp – A Modular Agent-based Travel Demand Modelling Framework. Procedia Computer Science 19, 854–859. 10.1016/j.procs.2013.06.114.

Ordonez M., S.A., Erath, A., Axhausen, K.W., 2012. Simulating urban transport for a week time horizon. Working paper Future Cities Laboratory. ETH Zurich, Zurich, 18 pp.

Raney, B., Nagel, K. An Improved Framework for Large-Scale Multi-Agent Simulations of Travel Behavior, in: Conference papers STRC 2004. Swiss Transport Research Conference, Monte Verità. 2004.

Richardson, C., 2014. Microservices: Decomposing Applications for Deployability and Scalability. http://www.infoq.com/articles/microservices-intro.

Roorda, M.J., Miller, E.J., Habib, K.M., 2008. Validation of TASHA: A 24-h activity scheduling microsimulation model. Transportation Research Part A: Policy and Practice 42 (2), 360–375. 10.1016/j.tra.2007.10.004.

Rosenkrantz, D.J., Stearns, R.E., Lewis, P.M., 1977. An analysis of several heuristics for the traveling salesman problem. SIAM Journal of Computing 6 (3), 563–581. 10.1007/978-1-4020-9688-4_3.

Schnittger, S., Wittowsky, D., 2002. mobiTopp - ein verhaltensorientiertes Verkehrsplanungsinstrumentarium, in: HEUREKA '02. Optimierung in Verkehr und Transport. HEUREKA, Stuttgart. FGSV, Köln.

Schnittger, S., Zumkeller, D., 2004. Longitudinal microsimulation as a tool to merge transport planning and traffic engineering models - the mobiTopp model, 2004, Straßburg.

Weiss, C., Chlond, B., Hilgert, T., Vortisch, P., 2016. Deutsches Mobilitätspanel (MOP) - Wissenschaftliche Begleitung und Auswertungen Bericht 2014/2015: Alltagsmobilität und Fahrleistung, Karlsruhe.

Zumkeller, D., Chlond, B., 2009. Dynamics of Change: Fifteen-Year German Mobility Panel, in: TRB 88th Annual Meeting Compendium of Papers, Washington, D.C.