Fostering Change of Individual Travel Behavior with Customized Mobility Services

Dominik Galler[a], Christoph Becker[b], Tim Hilgert[c], Martin Kagerbauer[d]

[a] *email: Dominik.Galler@esentri.com, esentri AG, Ettlingen, Germany*
[b] *email:christoph.becker@fzi.de, FZI Research Center for Information Technology, Karlsruhe, Germany*
[c] *email: tim.hilgert@kit.edu, Karlsruhe Institute of Technology, Institute for Transport Studies, Karlsruhe, Germany*
[d] *email: martin.kagerbauer@kit.edu, Karlsruhe Institute of Technology, Institute for Transport Studies, Karlsruhe, Germany*

***Abstract* — The development of new mobility services has progressed rapidly in recent years. Nowadays, users can choose from a variety of different mobility services and constantly new options become available. In this paper, we present the development of a new mobility assistance system that provides decision support for the user in order to facilitate selection of the best suiting means of transportation. Moreover, the impact of new mobility services on the user's individual travel behavior as well as the impact on the transport network in general are elaborated in this paper. The mobility assistance is a distributed system that collects and aggregates data from different providers to offer relevant, context-sensitive information for the user's current situation. Information, such as timetables of public transportation, real time data on delays, availability of car sharing vehicles or traffic congestions are aggregated, refined and then presented to the user. The mobility assistance supports the user regarding route selection, as well as scheduling of activities that are managed within the users' calendar with consideration of different starting times. Moreover, the mobility assistance implements optimization strategies to improve the user's travel needs according to his personal preferences. Activities as well as associated trips during a week can be optimally combined and rescheduled in order to achieve a lower overall travel time or low cost of mobility.**

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

I. INTRODUCTION

Technical developments, such as the Internet of Things (IoT) offer the possibility to develop new innovative services for the user [1]. This also applies for the development of new information services in the field of mobility. New mobility services and mobility management systems are becoming increasingly important. Nowadays, users have the possibility to choose from a variety of mobility options for their daily trips [2]. Due to more flexible working arrangements and changing user preferences regarding modes of transportation, new activity patterns can be observed [3]. The variety of mobility options leads to a greater complexity in decision management. Several mobility options can be used in combination and require the users to compare complex alternatives. Hence, customer-oriented services can help users to find optimal solutions regarding trip and schedule planning in consideration of personal preferences. In this paper, we present the development of customer-oriented mobility services and their impact on the users' travel behavior.

As part of the BMBF-funded research project BiE (Evaluation of integrated Electric Mobility), several project partners are involved in the development of new mobility management services to promote the integration and acceptance of electric mobility. Therefore, electric mobility solutions have to be easily available and integrated in everyday life. In addition, they should support the user in the selection and comparison of mobility services and options. The personal mobility assistance system, which offers the user the best possible support to carry out his daily trips is currently under development. It integrates decision support, especially for complex combination of mobility options and scheduling of activities. The information provided comprises data on travel modes, possible routes and starting times of trips. Furthermore, the mobility assistance system reschedules the user's activities of one week to create trip chains that optimize the user's schedule according to individual preferences, generally to minimize travel times, travel costs or environmental pollution.

In this paper, the architecture and functionality of the mobility assistance system are presented and effects on travel behavior are analyzed. Since the mobility assistance can have an impact on the daily mobility (e.g., changing the sequence of activities due to shorter travel times or the combination of several activities into trip chains), there potentially are more effects on individual travel demand or the transport network. To investigate the reorganization of activity patterns in use of such mobility assistance systems a microscopic multi-agent traffic demand model has been extended with a module for synthetic generation of activities. The traffic demand model thus allows the quantification of the mobility assistance's impact on daily mobility.

The rest of the paper is structured as follows. Section II describes the mobility assistance. Therefore, Section II.A provides an introduction into distributed systems and microservice architectures. Then Section II.B discusses the mobility assistances architecture and subsequently, Section II.C goes into finer details with respect to the selected optimization approach. Section III describes how the mobility assistance could impact on travel behavior. Finally, the conclusion and acknowledgements close the paper.

 ISBN: 978-1-61208-517-3

II. Mobility Assistance

The mobility assistance is implemented as distributed system. The system comprises many components that act independently, but co-ordinate their actions in order to provide the required functionality. The coordination is technically based on the exchange of messages over the Internet. For the individual user of the mobility assistance, a mobile application (smartphone app) is provided as a user interface to optimize his personal travel behavior. The optimization is calculated by mobility assistance based on information received from various services (message exchange). These services include real-time information regarding different transportation alternatives, such as the availability of car-sharing vehicles, public transportation, possible delays or traffic congestions. In order to control the system's complexity, the user is only presented with information that is relevant for the current situation and scheduling of future activities. Moreover, the user will receive notifications regarding possible optimizations of his schedule according to his personal preferences. For this purpose, the system monitors the user's calendar and reacts to new events. In addition to the presentation of context related information, the system proposes alternative routes and modes of transportation. This optimization is based on individual user preferences (e.g., minimizing travel time, minimizing travel costs, selection or deselection of certain modes of transportation).

A. Distributed Systems and Microservice Architectures

In order to calculate optimal results for each individual user based on distributed heterogeneous data, a complex network of different subsystems is essential. In this connection, two principles of information processing are important: first, the optimization can only be provided by distributed systems and secondly, the state of an information object is subject to a certain degree of uncertainty during the optimization process. The latter is due to CAP theorem known in IT: In distributed systems, where loss of single messages (partition tolerance) can always occur due to network failures, it is not possible to ensure that a changeable data object is available (availability) and that each participant of the distributed system has a consistent (consistency) view on this respective object [4].

Therefore, a distributed system (running the mobility assistance) that relies on distributed data has to provide algorithms that can cope with this degree of uncertainty. For the design of an appropriate software architecture, the most important requirements are extracted: a) real time: the mobility assistance should calculate and present the results (schedule optimization) in (almost) real-time for each user. b) sub-optimal results are acceptable: it is more important to display an improvement than to calculate the absolute optimum. c) scalability: the system must remain functional even with an increasing number of users and data (high load).

Given these requirements, BiE project partners have selected a software architecture that is suitable for a high load and provides a good flexibility regarding future development at the same time. The latter is particularly important because the partners involved in the project focus on different aspects of the mobility assistance. The architecture should support a partner-independent implementation of services. This idea is rooted in the paradigm of service-oriented architecture (SOA). However, SOA is currently subject to an ongoing discussions regarding future development and application, where the concept of microservices attracts a lot of attention [5]. Microservices can be seen as an architectural pattern for the design of distributed software systems. Briefly: microservices are an approach to implement a system based on a large number of small services. This is similar to the primary principle of SOA. However, some more stringent requirements are generally associated with microservices. Within the concept of microservices each service should be carried out independently from other services (own process space), use its own data (database) and offer lightweight communication mechanisms (often REST) to other services. With regard to the size of a service, it is intended to bundle only functionalities within the service, which serve a single business capability. Hence, the scope of a single microservice is very limited, thereby reinforcing basic principles of service-oriented architectures. In particular, loose coupling and separation of concerns can be easily achieved this way. Additionally, microservices strengthen the following principles: intelligent services and basic communication (smart endpoints & dumb pipes), evolutionary design, strict encapsulation (shared nothing), decentralized governance, distributed data storage and automation of infrastructure (build, test and deployment processes).

Unlike traditional SOA, the microservice approach is based on simple communication mechanisms. Instead of a sophisticated Enterprise Service Bus (ESB) microservices rely on the architectural pattern pipes and filters. The intelligent processing of messages takes place within the services (smart endpoint) while the communication is implemented using simple mechanisms like REST or asynchronous messaging. Hence, microservices can be easily replaced by new implementations, thereby following the principle of evolutionary design. Strict encapsulation in turn is an important prerequisite to enable evolutionary design.

These principles allow each project partner to develop functionality independently of other partners. Within the project this enables partners to yield their specialized know-how in the best possible way and allow for algorithms and functions that can be independently developed and deployed. The microservice approach thus enables the implementation of the mobility assistance as distributed system that is capable of performing real-time traffic analysis even under heavy load (number of users). The system is based on so-called reactive microservices adopting the Vert.x framework. The individual services are implemented as verticles in Vert.x. The architecture of Vert.x contributes to a high modularity of the system and facilitates the integration of new services.

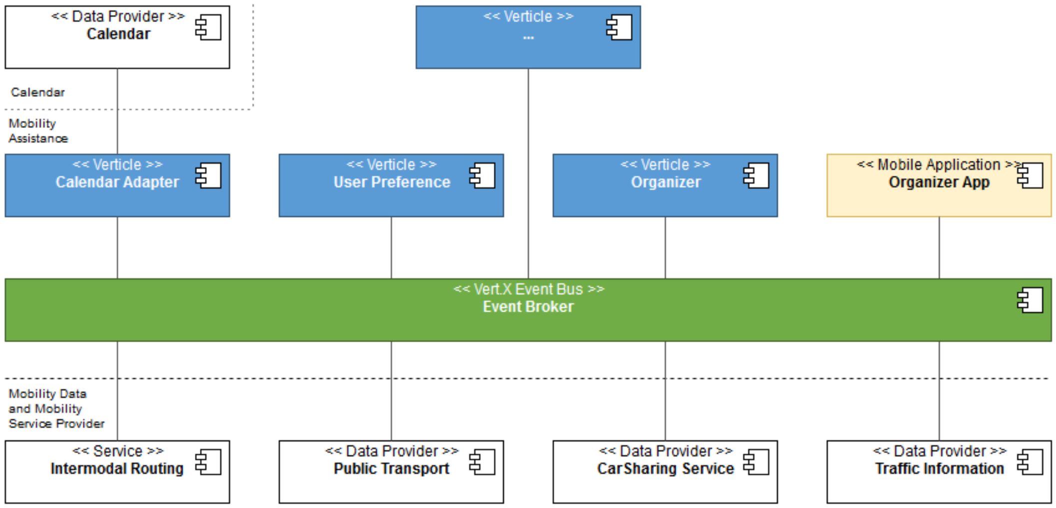

Figure 1. Mobility Assistance Architecture (Overview)

Vert.x itself is a lightweight, event based framework that supports the development of distributed systems. Different programming languages can be used to implement the services as verticle, thereby strengthening the independence of the development team. Each verticle comprises some aspects of the actual application logic of the mobility assistance system. In general, a verticle will respond to an event or create a new event. Communication between the verticles is established via an integrated and distributed bus. Communication takes place through the typical messaging patterns (publish-subscribe or point-to-point).

B. *Architecture of new Mobility Assistance*

The mobility assistance is a complex network of distributed systems and leverages a microservice-based architecture as discussed in Section II.A. In order to provide the required functionality, the mobility assistance makes use different interconnected components/subsystems that have been implemented as microservice (in this context also known as verticle). The components can be classified as follows (see Fig. 1): external data or service provider (white), Vert.x Event Bus (green), services / verticles (blue) and mobile application (yellow).

Moreover, each component illustrated in Fig. 1 can be assigned to specific domain, i.e., a) the mobility assistance domain, b) the mobility data and mobility provider domain and c) the calendar domain. Components within the mobility assistance domain implement the core functionality as well as the communication between different components and the mobile application. The remaining (non-colored) components comprise external domains that provide supporting calendar as well as mobility-related data and functions that are used by the mobility assistance.

C. *Optimization within the Organizer Verticle*

Within the organizer verticle, we developed a service called *calendar optimization service*. Its purpose is to optimize the whole week's schedule of a given user. Before the optimization process itself is presented, the different sources that induce travel demands are discussed.

Travel demands are not solely generated when a user searches actively for a route from point A to point B. More likely, the need for mobility is generated when the user is invited to an appointment and accepts the invitation in the first place. Therefore, the optimization of the user's travel demand doesn't start with the optimization of the routes between his appointments, but rather with optimizing his appointments themselves. Furthermore, there are not only appointments which generate travel demand. Thus, for the sake of simplicity, we will call all schedule elements of the user's week-schedule – regardless of whether they generate travel demand or not – activities.

Keeping this in mind, we divided the users' appointments into two different classes. The first class is the class of activities which cannot be moved in any way, e.g. appointments which involve more than one people or conference calls. The other class is called free activities and contains all other schedule elements that are more or less freely moveable within the week. There are two different subclasses within the free activities. First, there are free activities which have to take place within a certain time frame on a certain day, e.g., from 8am to 3pm on Monday. The second subclass contains free activities which have larger time frames, i.e., they last more than one day. Generally speaking, optimization is achieved when the free activities are arranged optimally around the fixed activities, which – by definition – cannot be moved. During the optimization process it must be guaranteed, that every activity can be reached within the defined time frame.

Within the time frames, activities can be attended by using various means of transport. The means of transportation may differ in departure time, which implies that they can only be used according to their departure time. Aside from this effect, traffic congestions in the morning or evening rush-hours may increase travel times as well as costs. To depict this behavior, we modelled the activities according to the asymmetric time-dependent travelling salesman problem with time windows (TDATSP-TW) as proposed by Albiach, Sanchis and Soler [6] and designed the transformation in accordance with a generalized ATSP (GATSP) presented also in this paper.

Other than Albiach et al. we had no access to CPLEX or other fast solvers for NP-hard problems. Instead of transforming the GATSP into an ATSP like Albiach et al. did, we developed an algorithm to solve the resulting GATSP while leveraging some of the GATSP's characteristics to solve the problem instances.

In order to design the algorithm, the Nearest-Neighbor-Heuristic has been adopted. If a path is found by the heuristic in the reduced GATSP within the reduction presented by Albiach et al., we initialize our algorithm with this path and try to optimize it. Since there might occur many dead ends while cutting out the clusters, the heuristic stops if it leads to a dead end to avoid a brute force solution when there is, e.g., only one feasible path within the graph. After the opening procedure, we try to optimize the initial tour in our algorithm in case an initial tour has been found. In case no initial tour has been found, we compute all feasible solutions otherwise. Therefore, we start with all possible paths from the depot to all of the reachable nodes. For any reachable node, we compute recursively all nodes, which are reachable from this node, and build the paths piece by piece. The termination criterion for the recursion is either reached if the computed path is feasible (i.e., a Hamiltonian Cycle in the GATSP-Graph) and cheaper as the cheapest path found so far, or if the path is infeasible and more expensive than the cheapest feasible path found so far. Within this process, there may occur problems without feasible solutions in the event that the combination of activities consumes more time within one day than the user can afford.

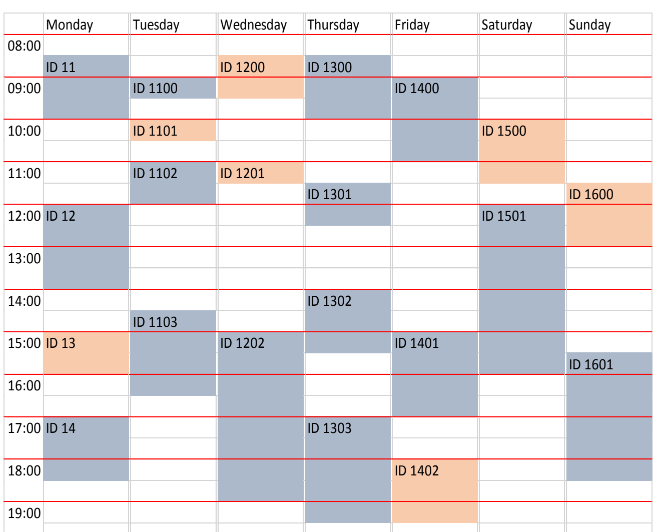

Figure 2. Initial week-schedule

To improve the runtime of the algorithm, we split the problem into various sub-problems, which can be solved on different distributed nodes. Therefore, we build the TDATSP-TW instances by computing all possible combinations of fixed and free activities for a given day. Our algorithm is then applied to each combination and returns a feasible and optimal solution – provided a solution exists – for the combination. Subsequently, the best option within all of the distributed computed solutions we've collected, is selected. The best option contains all activities, violates none of the user's preferences and has, according to the objective function, the lowest value. This solution solves the mentioned optimization problem best and thus is presented to the user.

To demonstrate the power of our solution, we provide the following case study: Since the routing service is needed to compute the edges of our GATSP and the fixed activities are always the same for each combination, we added a cache to this instance. This way, we can save many expensive requests over the network. Furthermore, network accesses, e.g., over HTTP may be slow. Thus, our algorithm was applied two times to the provided example instance. Firstly, with a cold, and then with a hot cache, to show the effect of the travel planning instance in computation time. As travel planning instance, we used Google's Distance-Matrix API [7]. The algorithm was executed on a desktop computer with an Intel i7 4770K CPU and 32GB DDR3 memory.

TABLE I: ACTIVITY SCHEDULING AFTER OPTIMIZATION

	Activity ID	Departure	Arrival	Distance (km)	Costs
Monday	11	08:10	08:20	1,74	2,914
	12	10:00	10:10	2,45	3,582
	13	14:00	14:10	2,32	3,269
	14	15:10	15:30	3,23	4,712
	0	18:30	18:50	4,53	5,944
Tuesday	1100	08:10	08:20	1,75	2,85
	1102	09:30	09:40	0	0,092
	1103	12:00	12:10	0	0,081
	0	16:30	16:40	1,17	2,111
Wednesday	1200	09:40	09:50	1,91	2,671
	1202	10:50	11:10	6,33	6,102
	1101	19:00	19:20	4,26	5,039
	0	19:30	19:40	0,08	0,159
Thursday	1300	08:00	08:20	6,28	7,444
	1301	10:00	10:20	4,55	6,146
	1201	12:30	12:40	2,72	4,033
	1302	12:50	12:50	0	0
	1303	15:30	15:40	2,34	3,653
	0	19:30	19:40	1,17	2,065
Friday	1400	08:40	09:00	6,3	6,166
	1401	11:00	11:20	8	7,394
	1402	17:00	17:20	4,78	5,927
	0	18:50	19:00	0,46	1,004
Saturday	1500	08:00	20:00	26,16	15,673
	1501	10:00	10:30	16,88	13,305
	1600	16:00	16:30	24,74	17,387
	0	18:00	18:10	1,72	2,794
Sunday	1601	08:20	08:30	1,77	2,99
	0	18:30	18:40	1,77	2,5

Fig. 2 shows the initial calendar of the week-schedule, the way any person would probably have planned it. This schedule was optimized using the calendar optimization service. We then optimized this schedule. The solution for the initial calendar was determined by using Google Maps. To compute the total solution, we chose the fastest route between the activities and the highest time value for the time variations presented by Google Maps. The described behavior could also be the one of a real person. Based on the results, we computed the total costs by applying the objective function. This lead to total costs of 164.583. Our algorithm was applied and took 355 seconds with a cold and 6 seconds with a hot cache to perform the optimization. The result was a schedule with total costs of 138.007 thereby leading to cost savings of 26.57 euro. The detailed optimization result is presented in TABLE I. Our algorithm works without penalty costs for arriving too early at the location – which is particularly permitted. However, waiting after the activity has been finished, is strictly forbidden. As a result of our evaluation, one must add penalty costs for waiting in order to avoid arriving way too early. Furthermore, it may be useful to consider waiting after the activity. This way, it's possible to wait, e.g., for an express train or plane, which could lead to a travel route, which is cheaper than all connections directly available.

III. IMPACTS ON TRAVEL BEHAVIOR

As shown above, the mobility assistance is capable – through their functionality – to influence people's travel behavior. This results in adaptions concerning activity chains, as well as influences on destination or mode choice. To quantify the impacts of the assistance on individual travel behavior but also on the transport network we use the agent-based travel demand modeling framework mobiTopp. It has been developed at the Institute for Transport Studies at Karlsruhe Institute of Technology (KIT). The framework simulates the travel demand (all trips within one week) for people – including activity, destination and mode choice – and explicitly takes into account behavioral stability and variability of peoples' travel behavior [8][9]. Within the last years it has been further developed to model also the impacts of new aspects like car sharing or electric vehicles [10][11]. For this work, we are using the mobiTopp framework to simulate the travel demand for the Greater Stuttgart Region in Germany with 2.7 million inhabitants.

To depict the impacts of the mobility assistance, the activity generation part of the model has been enhanced. Using this new module, it is possible to generate week activity schedules synthetically (which person makes what kind of activity how long and at which time). The advantage of the synthetic generation of these schedules is the modelling flexibility. Impacts of the assistance can be mapped directly into the simulation. Impacts of the assistance can also be mapped only to certain user groups and hence to investigate effects on different scenarios.

Fig. 3 shows the interaction between the mobility assistance (right side) and mobiTopp (left side) within the BiE-project. Week activity schedules are generated (see step 1) and then handed over as input to the mobility assistance. These schedules are optimized by the assistance (see step 3 in Fig. 3 or Section II in this paper) given different criteria. Following, adapted schedules feed back to mobiTopp in which they are analyzed concerning the impacts on travel behavior.

Usually the assistance suggests different optimizations of the original activity schedules. Therefore, we analyze different scenarios: First, a scenario where only the cost-minimized schedules are accepted by the agents; second a scenario where only the travel time-minimized plans are accepted and third a scenario where the agents randomly choose between the original schedules and the time- and cost-optimized schedule. Furthermore, we analyzed how the assistance impacts different user groups, e.g., commuters. The analysis of adapted schedules can evaluate changes in travel behavior: What kind of tours are adapted by the assistance? How are these tours changing? For every change in a schedule, the differences to the original schedules can be evaluated. This answers the following questions: How is the number of trips, the trip length and the travel time changing on individual level? How is the infrastructure load changing when a certain market penetration of the assistance is given? How is modal split changing within the system? Does the mobility assistance support sustainability? Does the assistance influence the infrastructure load in peak hours? How do changing trip lengths and modal splits influence the environment? After the implementation phase that is actually ongoing we will investigate these questions.

Further research will be a deeper investigation of the user's acceptance for the suggestions. Do people follow the suggestions of the assistance? Within the BiE-project we investigate only potentials of impacts on travel behavior and optimizations concerning travel times or costs. The travel demand model allows for a good scaling to simulate the usage of the assistance for different users and hence to investigate impacts for some given market penetration rates.

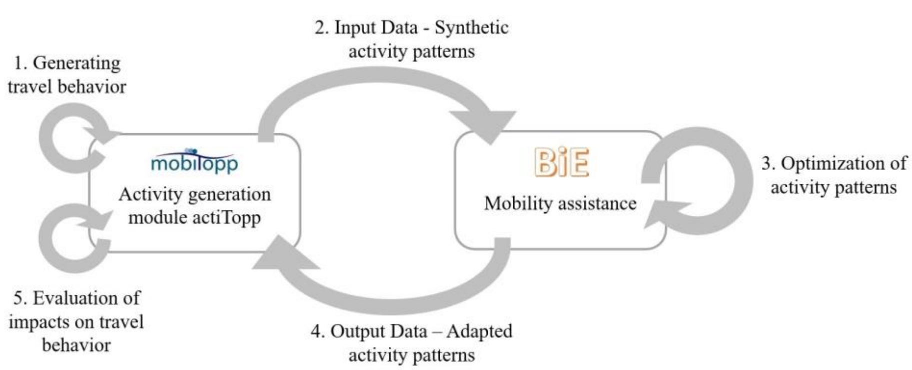

Figure 3. Interaction between mobiTopp and the mobility assistance

IV. CONCLUSION AND OUTLOOK

In recent years, a variety of new mobility services (e.g., multi- or inter-modal traffic management and information systems) have been created enabling users to choose from a wide range of different mobility services and options. To reduce the complexity of current mobility services the need for a customer-oriented assistance system is constantly getting more important. For this purpose, the presented mobility assistance system has been developed.

The mobility assistance gathers and aggregates information from timetables and real time information systems in public transportation, accesses mobility services, such as car sharing as well as the user's calendar and only presents selected information that is relevant in the current situation.

The mobility assistant supports the user in his daily mobility by providing routing information as well as information on alternative modes of transportation and starting times for trips. Depending on the user's individual preference, the mobility assistance may plan and reschedule activities as well as associated trips in the course of a week. To investigate the impact of mobility assistance systems on individual travel behavior, the travel demand model mobiTopp has been used. This allows for an estimation of effects at an individual level as well as network level (e.g., the shifts in the morning peak, when a certain number of people is using the mobility assistance).

As a result, it is expected that users will get better and clearer proposals and adjust their travel behavior accordingly. The proposals regarding schedule optimization will be presented on a mobile device that acts as user interface for the mobility assistance. The effectiveness of such a decision support system can only be achieved on the basis of a decentralized system, which is able to take account of a) information on general traffic situation, b) storage individual user preferences and c) calculation of optimizations regarding scheduling and trip planning. All these functionalities are currently being developed in the research project BiE.

Further research may relate to the development of assistance systems as well as to the analysis of simulated traffic behavior. In further steps for example, the acceptance of the calculated proposals from the mobility assistance system could be examined based on a user survey. In future development steps, the identification of further influencing factors (such as penalty costs for waiting times before and after activities) as well as the incorporation of these factors into the optimization routines could be addressed. Moreover, the current routines could be enhanced with adaptive algorithms (the procedure is adapted based on real-time information). Analysis results (see Section III) could be injected directly into the calculations of the organizer (see Section II.C) and thus improve the decision support offered by the mobility assistance.

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

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," Computer Networks, vol. 54, no. 15, pp. 2787 – 2805, 2010.

[2] G. Ambrosino, J. D. Nelson, M. Boero, and I. Pettinelli, "Enabling intermodal urban transport through complementary services: From Flexible Mobility Services to the Shared Use Mobility Agency: Workshop 4. Developing inter-modal transport systems," Research in Transportation Economics, 2016.

[3] L. Adams, F. McAndrew, L. Fellows, and S. ALI, "The scope for flexible working in future," 2014. [Online]. Available: https://trid.trb.org/view.aspx?id=1316106. [Accessed: 07-Nov-2016].

[4] A. Fox and E. A. Brewer, "Harvest, yield, and scalable tolerant systems," in *Hot Topics in Operating Systems, 1999. Proceedings of the Seventh Workshop on*, pp. 174–178, 1999.

[5] C. Richardson, "Microservices: Decomposing Applications for Deployability and Scalability," *InfoQ*. [Online]. Available: http://www.infoq.com/articles/microservices-intro. [Accessed: 07-Nov-2016].

[6] J. Albiach, J. M. Sanchis, and D. Soler, "An asymmetric TSP with time windows and with time-dependent travel times and costs: An exact solution through a graph transformation," *Eur. J. Oper. Res.*, vol. 189, no. 3, pp. 789 – 802, 2008.

[7] Google, "Google Distance-Matrix-API," 2014. [Online]. Available: https://developers.google.com/maps/documentation/distance-matrix/?hl=de. [Accessed: 07-Nov-2016].

[8] M. Kagerbauer, N. Mallig, P. Vortisch, and M. Pfeiffer, "Modeling Variability and Stability of Travel Behavior in a Longitudinal View Using the Agent Based Model mobiTopp," in *Transportation Research Board 95th Annual Meeting*, 18p, 2016.

[9] N. Mallig, M. Kagerbauer, and P. Vortisch, "mobiTopp – A Modular Agent-based Travel Demand Modelling Framework," *Procedia Comput. Sci.*, vol. 19, pp. 854 – 859, 2013.

[10] N. Mallig, M. Heilig, C. Weiss, B. Chlond, and P. Vortisch, "Modelling the weekly electricity demand caused by electric cars," *Future Gener. Comput. Syst.*, vol. 64, pp. 140 – 150, 2016.

[11] M. Heilig, N. Mallig, O. Schröder, M. Kagerbauer, and P. Vortisch, "Multiple-day Agent-based Modeling Approach of Station-based and Free-floating Carsharing," in *Transportation Research Board 94th Annual Meeting*, p. 18, 2015.