Experiment on Activity-travel Survey System based on Scheduling System

SATOSHI TAKAHIRA¹,²,a)  RYO KANAMORI¹,³,b)  TAKAYUKI ITO¹,³,4,c)

Received: July 7, 2013, Accepted: October 9, 2013

Abstract: To discuss or evaluate certain policies for a smart city (e.g., urban transportation systems), it is effective to develop an agent-based simulation that can reproduce an individual’s travel behavior and social interaction. Here, activity-travel data is needed to develop a behavior model. However, it is difficult to collect such data over a long time period due to a heavy burden on subjects of the survey. This study proposes a web system to collect an individual’s schedule data easily from travel information. Our proposed system has two key characteristics: 1) travel information (e.g., which route is best at a particular time) is recommended automatically based on the concept of a prism when the user enters a new schedule, 2) researchers can utilize users’ schedule information as activity-travel data without conducting a special survey. We tested the system with students as users, who expressed satisfaction with the system’s usability as well as operability.

Keywords: scheduling, travel plan, activity-travel survey

1. Introduction

The government or some policymakers often determines policies by conducting simulations based on various theories. Such simulations are used to reliably evaluate social traffic systems, such as urban transportation. In these efforts, we need travelers’ behavior data sampled randomly to increase the simulation accuracy.

Conducting paper-based surveys on personal trips is the conventional method used to collect relatively accurate data. Another recent method [1], [2], [3], [4] collects travel logs from GPS information. The details of a personal trip survey are given in Section 2. We investigated both approaches over a period of one week; however, neither method is suitable for continuously collecting data over a longer period of time. Therefore, a system must be developed to collect long-term data without disturbing users.

We focused on the online schedule book. People use online schedule system (e.g., iCal, Google Calendar) every day. And we also can find out the route information to transportation. Our system is improved convenience by integrating the route search and scheduling, travel plan, activity-travel survey

1. Introduction

The government or some policymakers often determines policies by conducting simulations based on various theories. Such simulations are used to reliably evaluate social traffic systems, such as urban transportation. In these efforts, we need travelers’ behavior data sampled randomly to increase the simulation accuracy.

Conducting paper-based surveys on personal trips is the conventional method used to collect relatively accurate data. Another recent method [1], [2], [3], [4] collects travel logs from GPS information. The details of a personal trip survey are given in Section 2. We investigated both approaches over a period of one week; however, neither method is suitable for continuously collecting data over a longer period of time. Therefore, a system must be developed to collect long-term data without disturbing users.

We focused on the online schedule book. People use online schedule system (e.g., iCal, Google Calendar) every day. And we also can find out the route information to transportation. Our system is improved convenience by integrating the route search information and scheduling system.

By setting travel constraints based on the “Space-Time Prism” used in traffic engineering, our system calculates the activity time, including travel time, and determines whether a new appointment can be added to a schedule. Our system can also grasp the start of travel times and estimated return times using its schedule app. The system can be used for transportation by car, train and foot.

We added the ability to incorporate behavioral survey results in our schedule management application. Users can report such behavior over a long term by adding a plan to the scheduling system that reflects the schedule times and locations typically used on a daily basis. Researchers can thus get long-term data and make a more detailed simulation to obtain yet more data.

In this study, we implemented our prototype scheduling system and evaluated it through behavioral surveys from users. We achieved Internet operability comparable with Google Calendar by adopting a library called “dhtmlx Scheduler[5]” implemented in JavaScript. All of the participants in our experiments described the proposed system as “effective” in the behavioral surveys, and 88% of the users described its operability as “easy to use.”

2. Background and Challenges

In traffic management, traffic simulations play a very important role in evaluating transportation policies. And it is also important that the travelers’ trajectory and trip purpose data from the real world. Behavioral survey data is entered as parameters when assembling a traffic simulation. Depending on the survey’s scale, behavioral survey data is generally huge, and their analysis is very time-consuming. One problem is that the data is typically handwritten. We described a simulation example in Section 2.1 that was performed using collected data. The current status of the research on life activities is detailed in Section 2.2, and the concept of time and place for a “Space-Time Prism” is explained in Section 2.3.

2.1 Requirements for Activity Data

By collecting activity data, we can utilize them in a variety of simulations. MATSim[6] provides a framework for imple-
menting large-scale agent-based transportation simulations. This framework consists of several modules that can be combined or used separately. Modules can be replaced by other implementations to test particular aspects of activity. Currently, MATSim offers a framework for demand modeling, agent-based mobility simulation (traffic-flow simulation), and replanning, as well as a controller to iteratively run simulations and methods to analyze the output generated by the modules.

2.2 Current Status of Collecting Life Activities

Without determining why, who, and when people use traffic, it is fundamentally impossible to solve some traffic problems in a city. For example, in order to reduce peak traffic volume, various investigations have examined the possibility of staggering office hours in Japan. These policies are called “Transportation or travel demand management (TDM)”.

There are various definitions of TDM. According to the Japanese Ministry of Land, Infrastructure and Transport, TDM is a “method to alleviate the road traffic of a city or a region by prompting changes of traffic behavior and drivers [7].” However, this definition is too narrow. A broader definition, which is becoming more common [8], delves into behavioral factors related to the attitudes and values of the people behind traffic demand. In other words, TDM also includes managing traffic demand by changing the location and the time of individual activities.

In investigations of staggered office hours, problems have surfaced from the aspects of duties, commuting, and living, all of which are affected by changing the time of work. Other opinions favor the introduction of staggered working hours because they offer more effective use of private time, greater morning flexibility, and more family time [9]. If the time difference is set properly in a work system, congestion can be alleviated and personal quality of life can be improved. However, under the present conditions, we cannot grasp the various needs of workers, such as which type of person can effectively utilize private time in a different work system or how frequently they will participate in the system.

In addition to the above example, current methods for studying the activities of actual life situations rely on handwritten input. For example, previous research asked the writer to write down on paper their travel methods (Fig. 1), including dates and times. The answers were collected and analyzed. However, filling out forms places a substantial burden on subjects every day for a certain amount of time, and obtaining information continuously is difficult because researchers, in practice, can only continue their investigations over a limited duration. However, by digitizing recent survey methods using smartphones, we can reduce the burden on users.

In fact, in a study using smartphones, Yingling Fan et al. [1] experimentally monitored the traffic in an application called “UbiActive,” which was developed for the Android smartphone. In their research, “UbiActive” was able to collect real-time physical activity and travel behavior data having the same level of precision as data from a subject who evaluates and reports his own travel action. In their tests, smartphone technology enhanced the awareness of the physical activity of the traffic and showed the behavior of traffic participants. If we want to know the purpose of traveling in this method, we can use “Web-Dialy survey” to complete the information. But it is difficult to investigate continuously.

Caitlin Cottrill Doyle et al. developed a smartphone-based travel survey system, and they asked for complementary information [2]. Users fill in their means of transportation or the purpose of traveling. However, a controlled large scale survey is difficult and costly task for users. And it is also hard to report his/her
correct activity every day.

Tamer Abdulazim et al. [3] and Yijing Lu et al. [4] tried to automatically detect the travel purpose and means of transportation from GPS data. Their system estimates the transportation that the user used from the acceleration and GPS data that is retrieved from the user’s smartphone.

There are more studies, such as using IC cards [11] or the data from the probe trajectory of the cars, however, those approaches have some problems in that clear purposes of their travels can not be grasped. The purposes of moving location (e.g., go to work, go shopping, go to school, etc.) are very important data as the trajectory and means of transportation for simulation and decision-making.

Collected data can be used for action recommendations and policymaking as big data. IBM provide some software [12] for analyzing a huge amount traffic data. From the data analysis results, we can be price optimization or predictive maintenance optimization.

2.3 Space-Time Prism

When we complete a schedule, we must include the actual free time when one is not traveling to a place. First, we visualize a one-day pattern of place movement. “Space-Time Prism” is an expression technique for graphing a daily schedule as place movement.

A Space-Time Prism graph is represented in either 2D (Fig. 3) or 3D (Fig. 2). One axis is the time axes, and the remaining axis are places. The difference is reflected by having one- or two-dimensional space; however, the underlying concept is the same.

A “Space-Time Prism” has two important concepts:

- Specific individuals’ Space-Time is available
- Specific travel behavior of individuals is involved in the collection of choices

Space-Time accessibility determines whether an individual’s Space-Time is available. Space-Time accessibility $A_k$ is expressed in Eq. (1):

$$A_k = f(d_k, a_k, T_k)$$

- $d_k$: Travel distance (travel time) through activity opportunities $k$
- $a_k$: Attractive activity opportunities $k$
- $T_k$: Maximum time available for activity opportunities $k$

Space-Time Accessibility $A_k$, from which the activity time’s “goodness” is determined from $d_k$, $a_k$, and $T_k$, has three indicators: attractive location, travel time, and a schedule for an individual that is always appropriate.

When a user moves from his/her home to another place, finishes work, and returns home, the range is fixed from the departure time to the time when he/she must return. A limit on place movement by time is called a prism limitation.

For example, a prism constraint is shown as a diamond in Fig. 3. If a user drives the car from home to place $j$, the range includes when she moves at the speed of the car from the beginning to the end. If a user leaves home, finishes work at place $j$, and then returns home, she wants to spend all of her free time based on an idealized schedule (Fig. 3).

However, as shown in Fig. 4, if a user moves from home to a train station and waits for a regularly scheduled train to move to a true objective even after arriving at place $j$, an actual schedule can be started.

As described above, a Space-Time Prism is used not only to visualize the movement of input and schedule constraints but also to understand people’s actual movements.

3. Proposed System

To overcome the handwriting input burden described in Section 2, we focus on daily diaries in this study. Although sched-
uled times and destination or venues are determined in advance, many obligations occur suddenly due to uncertainties. Another common situation is to add new appointments to a schedule that has already been decided. Features of schedule management systems are currently available, and they often manage to describe the start and end times of activities.

If a user adds a new obligation to his schedule, he must verify with a line search whether he has time to make the new appointment based on the relationship of its place in the previous schedule. In this study, we support the reduction of user effort in scheduling appointments by automating the traveling route at the location search on the same schedule management application.

We consider the following concepts for achieving automated scheduling of travel plans:

(1) Transportation support  
(2) Scheduling movement plans  
(3) Platform support

Transportation support corresponds to movement by foot, by train, or by car in this study. It is difficult to predict moving time if one takes the train without knowing in advance the transfer time and the distance to the station. However, route searches are very time-consuming.

Based on the Space-Time Prism, we implement our scheduling concept described in Section 2.

As a platform, we chose an Internet application that can be accessed from a terminal that frequently connects to the Internet. Users register their schedules with PCs, which can easily perform complicated operations, and browse their schedules using terminals such as smart phones.

Consequently, our implemented system can grasp scheduled start times, return-home movement time, and new appointments added to the schedule by setting a movement limitation condition based on the Space-Time Prism.

3.1 System Overview

The library dhtmlxScheduler[5], which is implemented in JavaScript, can set, change, and save schedules on the Internet in the manner of Google Calendar. The dhtmlxScheduler library can freely add functions, and since the source code is open to the public, we were able to implement the desired functions. PHP is used to save schedules in dhtmlxScheduler, and since such features as “dynamic schedule display” are generally implemented in JavaScript, we developed our system in it. Because it was developed in JavaScript, we can view and edit it without a problem in many browsers, even on mobile devices.

Since we performed Internet access for route searches from the server side, users can comfortably perform a route search even if they are using a narrow-band connection. We give the details in Section 3.4. For searching schedule routes during registration, compared with the start of a schedule based on the previous schedule, the system calculates the planning time and displays the estimated arrival and departure times. If users can travel the route within the time constraints, the time and the means of transportation are automatically registered in the schedule.

Automatically registered traveling schedules and events added to a schedule are displayed in different colors. In the prism constraints that occur during scheduling, our schedule management application moves along the time that has elapsed on the side of the prism constraints. By this operation, free time can be maximized at the locations of previously scheduled events.

3.2 System Structure

Figure 5 shows the rough flow of our system.

The system verifies that a movement path exists that satisfies the starting time and the next event’s starting time when a user enters an appointment in her schedule. If the schedule can be carried out within the permitted time, the system removes the unneeded path and registers the event.

Our SQL database schema is shown in Table 1.

The traveling route and the schedule are both managed as an event in the same database and allocated a unique id. When type 0 represents the schedule, 1 represents the data of the movement path. The start and end times are represented by event_start and event_end, the event name is event_text, places are event_location, etc.

![Diagram](image-url)
and an event’s latitude and longitude are represented as \( \text{lat, lng} \). Previous events and the next event are managed by \( \text{post\_id} \) and \( \text{pre\_id} \), and all events are given in list structure. Users are managed by \( \text{user\_id} \) and thus distinguished from each other when a schedule is loaded.

### 3.3 Schedule Input

Users can intuitively add events by a double-click and drag operation. They can enter a new item in their schedules by dragging from the item’s start time to its end time.

When they change times or register appointments to schedules, the system automatically searches for a route. When searching for routes, users must enter the location and the start and end times of their schedules.

### 3.4 Route Search

In this study, we use the results of a Google route search because it offers a line search as an API. If the search path involves traveling by train, we register the search from the nearest point of departure to the nearest station of the destination point and then transfer the pathway to a route within walking distance from the station. In the future, we must address real-time traffic information, traffic congestion, and probe cars which equipped with a GPS can collect and send vehicle information.

### 3.5 System Operation

When there is no appointment entry on the calendar, the user drags a new event from top to bottom to specify its start and end times and then enters its location and name. If the location cannot be entered, an error is displayed. If a new entry is made that overlaps existing plans, the schedule edit screen will not appear. Our system plans the movement of a single person, and thus overlapping is prohibited because one user cannot be simultaneously scheduled for more than one event. Furthermore, it is impossible to generate a travel plan if the times of events overlap.

When a schedule has multiple pieces, the system automatically searches for a travel course and shows whether the movement to the place is feasible in terms of time based on the schedule (Fig. 6). If the user can accomplish the movement within the available free time, the event is registered in the database as a traveling plan. For an auto-registered route, the maximum time available for activities at the location of the previous event will be computed by the constraints.

Moving plans can be seen from the scheduler after being registered in the database. An event’s start and end times in the movement plan show the start of moving and estimated arrival times. Moving plans are automatically created and displayed in a different color, and only the user can delete them.

Here, a user tries to add a new event to his schedule on Grove Street in San Francisco. An example of the Space-Time Prism is shown in Fig. 7. If a user adds an event to his schedule at 18:30, an error is displayed (the user will not be on time for dinner at 20:00) to encourage reconsideration of the schedule. If he changes the time to 18:00, his schedule is set for that time, and registration can be achieved. The Space-Time Prism at this point is shown in Fig. 8. When this schedule is registered, the movement path between Spear Street and West Oakland, which was originally registered, will be automatically removed.

### 4. Usability Assessment

In Section 3, we described the background of our application’s development and its operation. Next, we present the results of experiments in the laboratory evaluations of our application and introduce user requests and problems. About 20 students as subjects attempted to use the web-based scheduling system on a PC.

First, for the question on whether the presented mobile plan, from the automatic search, was satisfactory, most users answered “Very good” or “Good” (Fig. 9). Since the movement plans reflect a Google route search, we could search correctly. Those answering only “Acceptable” said the plan routes they wanted to exploit were not displayed. Next, for the question about system operation, 77% said it was “Easy” or “Very easy” to understand.
With an intuitive interface that can be manipulated, users were able to operate the system without confusion (Fig. 10). Furthermore, 77% said that the time it took to search for a movement plan “was appropriate.” By communicating at the server side to GoogleAPI, our system performed the route search with stable communication that prevented long waiting. Finally, all users believed that our system was useful for scheduling. Our method can explore routes automatically during an ongoing schedule, which is also a very important feature.

5. Conclusion and Future Work

In this study, we developed a schedule management application system to reduce the effort of users by automatically searching for travel plans. By setting constraints with a Space-Time Prism, which is used in traffic engineering, we implemented a system that can calculate the available time for activities, including travel time. It can also grasp the time needed to return home with the schedule management app.

From the questionnaire results of our evaluation experiment, 100% of the users answered that our application is useful, and 88% thought it was easy to use. Travel purpose and arrival and departure time have been included in the data obtained, and it is also a very important feature.

Some users want to use superexpress or express trains. Students often use local trains, but working commuters generally use express trains, especially during rush hour. By providing priority settings on user data at login, the route search can be improved to best suit particular users. If users could search for other routes after the system recommendation, this would provide greater convenience.

Convenience would also be improved by incorporating the schedule information of aircraft and ship transportation. We could thus identify the relative movements of many more people. If the system knew the overall schedule of commuters, that is, when many users are scheduled to travel by car in the same place at the same time, the system could predict the traffic situation and relieve a congestion before it occurs by recommending a different route. By increasing the transportation channels the system can handle, truly digitized use could be achieved based on the analog (handwritten) Life Survey described in Section 2.

For future work, we plan to evaluate our system on a larger scale, i.e., over 100 people. The planned study is intended to expose more users to the system and to solve remaining problems like server durability. We will also compare the verification accuracy of the data collected by existing investigation methods, such as person trip surveys, with that of the data collected by this system.

Acknowledgments

We thank Mr. Yousuke Ishiguro of DENSO, whose comments and suggestions were very valuable. We also thank Mr. Akira Tsukamoto of DENSO, whose comments contributed enormously to this work. This work was partially supported by the Funding Program for Next Generation World-Leading Researchers (NEXT Program) of the Japan Cabinet Office.

References

Satoshi Takahira is a master course student of Nagoya Institute of Technology. His department is Computer Science Graduate School of Engineering.

Ryo Kanamori received his Doctor of Engineering (major in civil engineering) from Nagoya University in 2007. He is currently a research associate professor at Nagoya Institute of Technology. His research interests include evaluation of transport policies with travel demand forecasting models and travel behavior analysis.

Takayuki Ito is an associate professor of Nagoya Institute of Technology. He received his B.E., M.E., and Doctor of Engineering from Nagoya Institute of Technology in 1995, 1997, and 2000, respectively. From 1999 to 2001, he was a research fellow of the Japan Society for the Promotion of Science (JSPS). From 2000 to 2001, he was a visiting researcher at USC/ISI (University of Southern California/Information Sciences Institute). From April 2001 to March 2003, he was an associate professor of Japan Advanced Institute of Science and Technology (JAIST). From 2005 to 2006, he is a visiting researcher at Division of Engineering and Applied Science, Harvard University and a visiting researcher at the Center for Coordination Science, MIT Sloan School of Management. From 2008 to 2010, he was a visiting researcher at the Center for Collective Intelligence, MIT Sloan School of Management. He is a board member of IFAAMAS, the PC-chair of AAMAS2013, PRIMA2009, General-Chair of PRIMA2014, and was a SPC/PC member in many top-level conferences (IJCAI, AAMAS, ECAI, AAAI, etc.). He received the JSPS Prize, 2014, the Prize for Science and Technology (Research Category), The Commendation for Science and Technology by the Minister of Education, Culture, Sports, Science, and Technology, 2013, the Young Scientists’ Prize, The Commendation for Science and Technology by the Minister of Education, Culture, Sports, Science, and Technology, 2007, the Nagao Special Research Award of the Information Processing Society of Japan, 2007, the Best Paper Award of AAMAS2006, the 2005 Best Paper Award from Japan Society for Software Science and Technology, the Best Paper Award in the 66th annual conference of 66th Information Processing Society of Japan, and the Super Creator Award of 2004 IPA Exploratory Software Creation Projects. He is Principle Investigator of the Japan Cabinet Funding Program for Next Generation World-Leading Researchers (NEXT Program). His main research interests include multi-agent systems, intelligent agents, group decision support systems, agent-mediated electronic commerce, and software engineering on offshoring.