Calibrating a New Reinforcement Learning Mechanism for Modeling Dynamic Activity-Travel Behavior and Key Events

Marlies Vanhulsel
Davy Janssens
Geert Wets[1]

Hasselt University - Campus Diepenbeek
Transportation Research Institute
Wetenschapspark 5, bus 6
BE - 3590 Diepenbeek
Belgium
Tel: +32(0)11 26 {9133; 9128; 9158}
Fax: +32(0)11 26 91 99
E-mail: {marlies.vanhulsel;davy.janssens; geert.wets}@uhasselt.be

Submission date 25/07/2006

Word count	
Abstract:	230
Text:	6.451
Figures:	0
Tables:	3*250
Total number of words:	7.431

[1] Corresponding author

ABSTRACT

Recent travel demand modeling mainly focuses on activity-based modeling. However the majority of such models are still quite static. Therefore, the current research aims at incorporating dynamic components, such as short-term adaptation and long-term learning, into these activity-based models. In particular, this paper attempts at simulating the learning process underlying the development of activity-travel patterns. Furthermore, this study explores the impact of key events on generation of daily schedules.

The learning algorithm implemented in this paper uses a reinforcement learning technique, for which the foundations were provided in previous research. The goal of the present study is to release the predefined activity-travel sequence assumption of this previous research and to allow the algorithm to determine the activity-travel sequence autonomously. To this end, the decision concerning transport mode needs to be revised as well, as this aspect was previously also set within the fixed schedule. In order to generate feasible activity-travel patterns, another alteration consists of incorporating time constraints, for example opening hours of shops. In addition, a key event, in this case "obtaining a driving license", is introduced into the learning methodology by changing the available set of transport modes.

The resulting patterns reveal more variation in the selected activities and respect the imposed time constraints. Moreover, the observed dissimilarities between activity-travel schedules before and after the key event prove to be significant based on a sequence alignment distance measure.

1 INTRODUCTION

Traditionally travel demand modeling highly concentrated on trip-based modeling. However, more recently activity-based modeling is gaining importance. This type of modeling assumes that travel patterns are the result of the activity schemes that individuals and groups of individuals execute in their attempt to achieve certain goals. In this context an individual implicitly needs to decide on the following items: which activity to perform at which location, when to start this activity and for how long, which transport mode to use in order to get to the desired location, who will accompany the individual during the course of the activity, etc. Moreover, in this choice process the individual has to take into consideration a whole range of environmental constraints, for example coupling constrains, opening hours of shops and public authorities, possibility of congestion, availability of transport mode and available time windows in which the activity can take place. *(1)(2)(3)(4)(5)*

Several models have been developed to model this choice process and to predict activity-travel patterns (see for instance *(6)(7)(8)(9)*). However, as activity-based models include a set of decision rules or utility approximations which are founded on a snapshot of mobility data, the outcome of such models is still quite static and consequently in terms of short terms dynamics in activity-travel patterns, these activity-based models have -at their current state of development- much less to offer. In addition to short term dynamics, issues such as uncertainty, learning and non-stationary environments are most of the time also not considered. *(2)(10)(11)*

One of the exceptions to this general trend is the work by Timmermans and Arentze *(9)(12)(13)* and the Aurora model *(14)(15)*, which focuses on the formulation of a comprehensive theory and model of activity rescheduling and re-programming decisions and complements this with uncertainty and various types of learning. In order to contribute to this line of research, this paper evaluates the use of a reinforcement learning paradigm that allows one to simulate activity-travel scheduling decisions, within day re-scheduling and long-term learning processes in high resolution of space and time. In this definition, short term adaptation, can be considered as within-day rescheduling due to the occurrence of unexpected events during the execution of the planned activity program of an individual. Some examples of such unexpected events are: an (unexpected) change in the duration of preceding activities, congestion or changes in the availability of transport modes. These unexpected events can cause changes in the different aspects of one or more following activities: the duration of an activity can be shortened or extended, the starting time of the activity can be altered, the activity can be performed at another location than originally planned, the activity can be omitted, a different transport mode can be chosen, etc. In addition, changes in these aspects can occur simultaneously. *(2)(3)(10)(16)*

Long term learning on the other hand accounts for the impact of experiences of previous actions on activity-travel patterns. While conducting activities, individuals build up expectations and beliefs about the outcomes of their behavior (i.e. learn). New experiences will be looked at against this mental map of the environment and they will cause these expectations and beliefs to be updated. This study assumes that the occurrence of key events impacts the composition of these mental maps and thus causes changes in the activity-travel patterns. Key events contain rare events in one’s life cycle, such as a change in residential, work or study location, but also a change in household composition, car availability or household income. *(2)(10)(17)(18)*

In this paper, the development of a new learning scheme that is based on principles of reinforcement learning was developed and tested by means of a case study in a limited empirical setting. From a conceptual point of view, the approach that has been undertaken in this paper is innovative due to the fact that long term learning (through the incorporation of key events) is integrated in the proposed technique. Additionally, an initial framework of the learning process underlying the composition of the mental map of the environment is drawn and the decision process of an individual related to activity-travel decisions is simulated. Finally, the updating of these mental maps after the occurrence of a key event is also taken into account.

From a methodological point of view, the use of reinforcement learning in this paper relies on work which has been proposed by Janssens *et al. (19)* and which has also been adopted previously by

Charypar *et al. (3)*, and by Arentze and Timmermans *(16)*. These authors have claimed that the use of reinforcement learning is mainly advantageous to simulate a flexible response to unforeseen events in a very short period of time *(16)*. However, also at this methodological level, the presented approach is innovative. While Charypar, *et al. (3)* and Janssens, *et al. (19)* mainly focus on time and location allocation, assuming a fixed sequence in which the activities occur, agents (humans) do not execute activities in a fixed, pre-scheduled order. This paper describes a framework which allows drawing up a flexible activity-travel schedule including temporal constraints. Secondly, when simulating the daily activity pattern of an agent, this study does not adopt reinforcement learning to predict one single choice facet, but it aims at integrating the different choice aspects simultaneously into the decision process. Thirdly, the present study attempts at simulating the effect of a key event on the generation of an activity-travel schedule.

The remainder of this paper has been organized as follows. Section 2 discusses the theoretical framework underlying the technique used in this research. Section 3 redefines the goals and contributions of the current research to the current state-of-the-art. In the empirical section, the generation of results and the validation of these outcomes have been introduced. The paper concludes by defining future research topics and recapitulating research findings.

2 REINFORCEMENT LEARNING

2.1 Learning by Interaction

When learning a certain (new) behavior, the reinforcement technique generally assumes that an individual starts by exploring all possible actions. Subsequently, the individual is supposed to perceive the response of the environment while trying out one of these actions. Next, the individual will adapt future behavior according to this feedback. Reinforcement learning refers to this common way of learning. Kaelbing, *et al. (20)* define reinforcement learning as follows: “Reinforcement learning is the problem faced by an agent that must learn behavior through trial and error interactions in a dynamic environment.” *(20)(21)(22)*

Furthermore, the consequences of actions change over time, may not be fully predictable, and depend on the current and future state of the environment. Reinforcement learning has the potential deal with this uncertainty through continuous observation of the environment continuously and through consideration of indirect and delayed effects of actions. Moreover, reinforcement learning also takes into account the fact that an individual uses his previously gathered experience to improve the decision making process. *(16)(21)*

2.2 Basics

In this section some basic concepts concerning reinforcement learning are introduced. For a more profound theoretical foundation one can turn to Janssens *(22). (21)(24)*

2.2.1 Agent

When employing the term *agent*, one denotes the decision making unit under consideration, often being an individual or household within the context of our research.

2.2.2 State

A system is composed of a finite *set of states S*, which are compounded of a number of dimensions. *(22)* Traditionally in this context of activity-travel behavior, each *state s* can be defined by following dimensions: the activity which is being performed, the location at which the activity occurs, the starting time and duration of the activity, the time frame and position in the activity schedule in which the activity is executed, the accompanying people during this activity, etc.

2.232 Action

The *action set A(s)* for a certain state *s* refers to all possible decisions which can be made starting from that state. The action set may differ for each state and has to be finite and defined in advance. Moreover, for a given state *s*, the action *a* determines the next state *s'*. The action set may for instance be stay (i.e. perform the activity for one more time unit) or move (i.e. perform the next activity of the predefined, fixed sequence of activities). *(3)(22)*

2.2.4 Rewards

Finally, while making decisions an agent receives feedback, which can be either immediate or delayed, and direct or indirect. This feedback is called the *reward R*, which can be compared to the concept of utility in conventional choice models. A *reward R* is expressed as a function of the state *s* and the action *a* and can contain multiple dimensions. Furthermore, it should be emphasized that rewards are subject to alterations over time due to shifts in personal preferences and changing environments. *(3)(16)(22)(23)*

2.2.5 Discounting factor

As stated earlier, the reinforcement learning agent does not only take into account immediate rewards, it also incorporates the effect of delayed rewards. To serve this purpose, a discounting factor $\beta<1$ has been introduced. This measure assesses the weight of a future reward. For example, having a discounting factor β close to zero means that the agent tries to maximize immediate rewards only. However, a high β-value denotes that rewards in the future will have a large impact on decisions. *(3)(22)*

2.2.6 Q-value

The Q-value of an action *a*, given a state *s*, denotes the expected utility of an agent taking action *a* in state *s*. This value is calculated as follows:

$$Q(s,a) = R(s,a) + \beta.\max_{a'} Q(s',a')$$

where

s is the current state,
a is the performed action,
s' is the state defined by performing action a in state s,
a' are all possible actions of the action set A(s') in state s',
Q(s,a) is the Q-value of action a in state s,
Q(s',a') is the Q-value of an action a' in state s',
R(s,a) is the reward of performing action a in state s,
β is the discounting factor.

(3)(16)(22)

2.3 Q-Learning Algorithm

The above basics have only described the outcome of the algorithm after learning (so-called steady-state). In order to reach this steady-state, the agent has to run through the actual learning process. As this algorithm is founded on the Q-value, which has been defined earlier, it is also called the Q-learning algorithm. The process can be written as follows:

1. Initialize the Q-values.
2. Select a random state *s* which has at least one possible action to select from.
3. Select one of the possible actions. This action will get the agent to the next state *s'*.
4. Update the Q-value of the state-action pair (*s*,*a*) according to following update rule:

$$Q_{t+1}(s,a) = (1-\alpha).Q_t(s,a) + \alpha\left[R(s,a) + \beta.\max_{a'} Q_t(s',a')\right]$$

where

$Q_t(s,a)$ is the Q-value calculated and memorized during a former visit of this state-action pair (*s,a*),

$Q_{t+1}(s,a)$ is the Q-value to be updated in the current time step,

$R(s,a)+\beta.\max_{a'} Q_t(s',a')$ is the Q-value linked to the current state-action pair (*s,a*),

α denotes the learning rate, which is a parameter of the algorithm. This learning rate will decrease as time increases. This signifies that, at the start of the learning algorithm, the updated Q-value will solely reflect the Q-value directly related to the current time step, whereas after a while, the updated Q-value will also incorporate previous experiences due to the larger weight of the former Q-value in the update rule.

5. Let *s = s'* and continue with step 3 if the new state has at least one possible action. If it has none, go to step 2.

(3)(22)

2.4 Explore vs. Exploit

In the context of Q-learning two concepts tend to pop up frequently: explore and exploit. *Exploit* signifies choosing the action that is known to yield the highest reward. By doing so, the agent aims at reaching the state that is next to the currently best solution. This approach is also called, the greedy. *Explore* denotes the random selection of a possible action. The goal of exploring is arriving at a state that might not be visited otherwise. Besides, the consequences of such random action may produce a higher reward than that of the most optimal action so far. *(3)(16)(21)*

As such, in each state agents have to trade off exploring choice opportunities and exploiting current knowledge. This behavior is represented in a parameter called the exploration rate, which denotes the probability of choosing a random action. Frequently, the exploration rate will be set rather high at the beginning of the learning process, and will decrease in the course of the learning process: while the system will initially favor exploration, it will stress exploitation afterwards. However, as the agent operates in a dynamic, ever changing environment, it is necessary to explore once in a while in order to discover new opportunities. Furthermore, it should be noted that the value of the exploration rate depends on to which extent the individual is risk averse and thus on the individual's tendency to explore. *(2)(22)*

3 CONTRIBUTIONS TO THE CURRENT STATE-OF-THE-ART

3.1 Goals

As mentioned in the introduction the purpose of this study is to further elaborate existing Q-learning algorithms in the area activity-travel behavior. The basis of the learning algorithm was given by Janssens *(22)* and takes into account fifteen activity classes: in home work or study, out of home work or study, bring or get persons or goods, daily shopping, non-daily or window shopping, service activities, medical visits, eating, sleeping, out of home leisure, out of home non-leisure, in home leisure and in home non-leisure activities, and other activities. These activity classes are incorporated in the reward tables underlying the model. The data making up these reward tables are derived frequency data based on activity-travel diaries.

The main goal of this study is to abandon the predefined activity-travel sequence and to generate a flexible activity-travel schedule, taking temporal constraints into consideration. Furthermore as opposed to previous researches *(3)(16)(19)(22)*, which aimed at simulating the decision process of one single choice facet, and in order to reproduce the development of an activity-travel pattern more realistically, this study concentrates on predicting simultaneously several dimensions regarding activity-travel schedules.

After implementing these improvements, the present study aspires at capturing the consequences of a key event on the activity-travel pattern. More specifically, the intended key event compromises the fact of an individual obtaining his driving license. This key event is simulated, given the knowledge that solely the availability of transport modes will alter and assuming all other input variables equal. The details of these modifications and the results of the simulations are discussed subsequently.

3.2 Implementation

3.2.1 Basic Algorithm

This section will describe the development of the learning model which is taken into consideration in this paper. As stated above, the initial program is based on the algorithm developed by Janssens *(22)* and contains only two possible actions: stay (the current activity will be executed for one more time unit) or move to the next state (for which all dimensions are predefined in the fixed activity sequence).

The main loop of this program, which constitutes the core of the learning process, looks as follows:

```
For i = 0 to I do
    o  Determine a random state $s$ (i.e. decide on activity, starting time, duration, location)
    o  For j = 0 to J do
         ▪  Determine whether to explore or to exploit
         ▪  If explore:
              •  Determine a random action $a$ (i.e. stay or move)
              •  In case of "MOVE": determine a random location at which to perform the next activity
         ▪  If exploit:
              •  Determine the best possible action $a$ based on current "knowledge" of the system (i.e. stay or move)
              •  In the case of "MOVE": determine the best location at which to perform the next activity
         ▪  Determine the next state $s'$ (i.e. calculate activity, starting time, duration, location)
         ▪  Calculate the reward $R(s,a)$ of this action
         ▪  Calculate the Q-value $Q(s,a)$ of this action
         ▪  Update the table containing the Q-values
```

where
I = the number of iterations within the leaning process
J = the number of successive actions within each iteration

As one can notice, this learning process contains most components discussed in the second section of this paper. The most notable element covers the incorporation of the trade-off between exploration and exploitation: each learning iteration starts by determining whether to explore the possible action space or to exploit current knowledge. In this algorithm, exploring is equivalent to deciding randomly either to stay or to move. Exploiting, on the other hand, signifies selecting both the action (stay or move) and corresponding location (in case of "move") that yields the currently best Q-value, given the current state *s*.

3.2.2 Releasing the Predefined Sequence

The main goal of this study consists of eliminating the fixed sequence of activities which is specified in advance. It is thus necessary to enable the algorithm to determine the activity schedule entirely autonomously. To reach this aim it is necessary to extend the action space *A(s)* in every state *s*. Opposed to previous models, the action set *A(s)* will no longer consist of the actions stay or move, but each action will correspond to one of the fifteen possible activities. For instance, choosing action 0 stands for performing activity 0 or "in home work or study", or selecting action 1 signifies executing activity 1 or "bring or get persons or goods".

Moreover, because travel was incorporated in the fixed activity pattern of previous models, the travel mode used to travel from one activity location to the next was also fixed beforehand. In order to get rid of the predefined structure, the travel component thus needs to be detached as well. In addition, the transport mode is supposed to be interrelated with activity location. For example, to reach a location within 1 km from home people will often prefer to use a slow transport mode, such as walking or going by bike. To get to a location at 100 km from home, people will presumably choose to go by car or public transportation. Bearing this in mind and given the fact that various authors seem to disagree about whether location choice or choice of transport mode needs to be selected first, these two choice facets are combined into one single decision. Consequently an agent will decide on these two aspect simultaneously. *(1)(25)(26)(27)*

3.2.3 Incorporating Constraints

Another major change in the algorithm compromises the implementation of temporal constraints, such as the incorporation of opening hours of shops and leisure locations, and the definition of time windows for working, social activities, home activities and the like. These constraints are built into the application in order to limit the available choice set *A(s)* in each state *s*. Indeed, a problem which is now likely to occur because the sequence is no longer predefined in advance is that the algorithm may generate unrealistic activity-patterns. For instance, the activity "Daily shopping" is unlikely to occur since regular shops are probably shut at 3 am. Therefore, the action set may for instance only consists of the activity "Sleeping" for this time window. Other examples are: for 9 am the action set may only consists of the activities "in home work or study" and "out of home work or study" given that the agent has to execute the compulsory activity "work or study" between 9 am and noon and between 1 pm and 5 pm. However, after 5 pm the agent is allowed to perform all activities except for the activity "Sleeping". For the time being, the action set is supposed to depend solely on the time of the day. Other dimensions have not yet been taken into account while determining the limited action space.

Having introduced these modifications, the main iteration of the learning process can be written as follows:

```
For i = 0 to I do
    o   Determine a random state s (i.e. decide on activity, starting time, duration, location)
    o   For j = 0 to J do
            ▪   Determine the possible action set A(s)
            ▪   Determine whether to explore or to exploit
            ▪   If explore:
                    •   Determine a random action a
                    •   Determine the activity belonging to action a
                    •   Determine a random location at which to perform this activity and
                        determine which transport mode to reach this location (i.c. only one
                        decision)
            ▪   If exploit:
                    •   Determine the best possible action a (based on current "knowledge" of
                        the system
                    •   Determine the activity belonging to action a
                    •   Determine the best location at which to perform this activity and
                        determine which transport mode to reach this location (i.c. only one
                        decision)
            ▪   Determine the next state s' (i.e. calculate activity, starting time, duration,
                location)
```

- Calculate the reward *R(s,a)* of this action
- Calculate the Q-value *Q(s,a)* of this action
- Update the table containing the Q-values

where
I = the number of iterations within the leaning process
J = the number of successive actions within each iteration

The careful reader will notice the following differences. Firstly, the agent needs to determine the possible action set *A(s)*. After all, when this action set consist of only one activity, the agent is not able to choose between the different actions and is thus obliged to perform this activity. In the next step and similar to the original model, the agent decides on whether to explore or exploit. In both cases, the agent has to select an action (activity) from the action space *A(s)* determined beforehand. Subsequently, the agent makes the dual choice of at which location to perform the selected activity and which transport mode to use to travel to this activity location. It should be mentioned that - at present - each location can only be reached by only one transport mode. However, these choice combinations will certainly be extended in the near future. The final calculations of the original learning process remain the same.

3.2.4 Introduction of Key Events

The final advancement of the current study includes the possibility of simulating the occurrence of a key event. More specifically, the present program allows generating different activity-travel patterns before and after a certain point in time. The key event introduced here grasps the occasion in which an individual obtains his driving license. In this study, this key event is believed to change the set of available transport modes, causing locations, which could previously only be reached by car (as passenger) or by public transportation, to be paired to the transport mode “car (as driver)”. For example, suppose a certain location is situated at 100 km away from the home location and, in addition, this location cannot be reached within an acceptable time window using public transportation. Given the fact that the individual does not possess a driving license, the only way to get to this location is asking someone else to take him (car as passenger). After obtaining his driving license, the individual will be able to make it to this location independently by car (car as driver).

Within this study, the above effect is presumed to happen under ceteris paribus assumption: the effect is therefore assumed to be the only consequence of this key event, no other alterations concerning activity or travel dimensions are supposed. The goal is thus to check whether the algorithm responds to such changes through the generation of a different activity-travel schedule. The actual impact of a key event on the daily activity-travel pattern may be recorded in the near future for instance by means of a revealed preference experiment.

4 EMPIRICAL SECTION

4.1. Results

After having implemented and tested above described alterations, a first run of this program produces the following activity-travel scheme.

TABLE 1 Activity Pattern of Run 1 Before the Occurrence of the Key Event

Activity	Starttime	Finishing time	LocationID	Transport mode to next location
Sleeping	3:00	7:00	82	Bike
Bring or get persons or goods	7:00	7:15	93	Bike
Daily shopping	8:00	8:30	36	Bike
In home work or study	9:15	11:45	82	Public transportation
Out of home work or study	12:45	14:30	29	Bike
In home work or study	15:30	17:00	82	Bike
Eating or drinking	17:00	17:30	33	
Non-daily or window shopping	17:30	18:00	33	
Eating or drinking	18:00	18:30	33	Bike
In home leisure	19:00	19:45	82	
In home non-leisure	19:45	21:00	82	
In home leisure	21:00	22:30	82	
In home non-leisure	22:30	23:00	82	
In home leisure	23:00	23:15	82	Car as passenger
Other	1:00	1:15	22	Bike
Sleeping	3:00	7:00	82	

Some explanation is required while interpreting table 1. The major difference compared to the founding model, is the increased amount of variation which seems to appear in above activity-travel pattern. This variation is the main result of allowing the algorithm to select the activity sequence autonomously. Future research will compare such activity-travel schedules to real-data activity-travel patterns in order to determine whether this algorithm produces more realistic schemes than the founding model.

Apart from that, one can see that the activities “Sleeping”, “In home work or study”, “In home leisure”, “In home non-leisure” all take place at the home location of this agent: location 82. The home location is linked to the transport mode “Bike”, regardless of the transport modes employed to reach previous locations. This observation states the most unfavorable consequence of coupling of the decisions concerning activity location and transport mode: the algorithm is not able to incorporate efficient use of transport modes, including trip chaining, into the activity-travel pattern. To solve this issue, future researches will separate the choice aspects location and transport mode.

Furthermore, in order to determine the starting time of the next activity, the agent calculates travel times between two locations while drawing up the optimal daily schedule. Apart from that, one can see that the agent complies with the postulated constraints. For example, the compulsory activity “work or study” mainly happens between 9 am and noon and 1 pm and 5 pm, and “sleeping” occurs before 7 am. Moreover looking at the activities “Daily shopping” and “Non-daily or window shopping”, it can be confirmed that the agent also respects the opening hours of shops (between 7 am and 6 pm for “Daily shopping” and 9 am and 6 pm for “Non-daily or window shopping”).

The second run of the program, incorporates the changes due to the occurrence of the key event “obtaining a driving license”, results in the following activity pattern.

TABLE 2 Activity Pattern of Run 2 After the Occurrence of the Key Event

Activity	Starttime	Finishing time	LocationID	Transport mode to next location
Sleeping	3:00	7:00	82	Car as driver
Daily shopping	7:00	7:30	2	
Bring or get persons of goods	7:30	8:00	2	Car as driver
Out of home work or study	8:15	12:00	29	Car as driver
Bring or get persons of goods	12:15	12:45	5	Car as driver
Medical visits	13:00	13:15	0	Car as driver
Out of home work or study	13:30	16:00	29	Car as driver
In home work or study	16:15	17:15	82	Bike
Eating or drinking	17:30	20:45	41	Bike
Out of home non-leisure	21:00	22:00	33	Car as driver
Other	22:15	22:45	58	Car as driver
In home leisure	23:00	0:00	82	Car as driver
Sleeping	0:00	7:00	82	

Comparing these two patterns, one will notice that some differences exist. For example, the location of the activity “Daily shopping” has changed from location 36 to location 2, which is further away from home compared to location 36, but which can be reached faster by car (as driver) than location 36 by bike. In addition the duration of some activities, such as “Bring or get persons or goods”, has altered, or new activities turn out to be profitable, e.g. “Medical visits”. Another change which attracts attention is the sequence in which the activities occur: the activities “Daily shopping” and “Bring or get persons or goods” executed in the morning switched places in the sequence.

4.2 Validation

For the sake of quantifying the difference of the two runs, the distance between the resulting activity schemes can be calculated using the DANA-tool (Dissimilarity ANalysis of Activity-travel patterns) developed by C.H. Joh, T.A. Arentze and H.J.P. Timmermans *(28)*. The distance between two activity patterns reflects the similarity of these activity patterns: the lower the distance measure, the more the sequences are alike. *(29)* The distance measure, which will be applied in the current study, is based on a method called SAM (Sequence Alignment Methods) determined in Joh, *et al. (28)*. The distance measure produced by SAM indicates how much effort was needed to transform one sequence into the other one. The higher the SAM-score, the more maneuvers (inserting, deleting or reordering of activities) were performed in order to equalize the patterns and thus the less similar the sequences are.

Furthermore, to confirm whether the distance between the two initial runs cannot only be assigned to random variation, these runs are repeated 100 times. Subsequently all of the resulting 200 activity-travel schedules are mutually compared using the SAM-distance measures.

The outcomes of these calculations are assigned to one of two classes. Class 0 consists of the cases for which the comparison is made between activity patterns of the same group: every activity scheme of the first batch (before the key event) is compared to every other activity scheme of this batch. As such, every activity pattern of the second batch (after the occurrence of the key event) is put aside every activity pattern of the second batch. Class 0 contains 9.900 observations (2*(100*99/2)). Class 1 compromises all cases reflecting the comparisons between activity patterns of different groups: for every activity scheme of the first batch the distance to every activity scheme of the second batch is calculated. This class contains 10.000 cases (100*100).

An overview of the basic statistics is included on page 13. The average SAM-distance between two activity patterns of the same batch equals 66, while the average SAM-distance between two activity patterns of different batches runs up to 100.

Based on these numbers, one could now deduce that the activity-travel schedules generated by the two runs are divergent, as the average distance between schemes of the two different runs exceeds the average distance between patterns within one run. However, these basic statistics do not reveal the suggested conclusion just like that. After all, the key question is: do these activity patterns *significantly* differ? Or reformulating this problem the question remains: do the distances in class 0 *significantly* differ from the distances in class 1? In this case, a t-test is applied to verify whether the differences between populations (i.c. class 0 and class 1) are meaningful or whether these dissimilarities can be attributed to randomness of the data. Employing a t-test forces formulating a null hypothesis and alternative hypothesis of the problem:

$$H_0 : \mu_0 = \mu_1$$
$$H_0 : \mu_0 \neq \mu_1$$

where

μ_0 is the mean of distance in class 0,

μ_1 is the mean of distance in class 1.

However, before answering the key question, some additional checks need to be performed. Firstly, the assumption of normality of the data is met, based on the central limit theorem as both groups in the sample contain more than 50 cases. Secondly Levene's test for equal variances has to be executed. As shown below it is not possible to assume equal variances. At last a t-test for equality of means is carried out, which unveils that the null hypothesis of equal means between the two classes can be rejected. The initial impression that the activity pattern before the key event differs from the activity scheme after the key event, is hereby confirmed.

TABLE 3 Some Basic Statistics

T-Test

Group Statistics

	Class_2	N	Mean	Std. Deviation	Std. Error Mean
Distance_DPSAM	0	9900	65,9248	18,61836	,18712
	1	10000	100,1830	16,05278	,16053

Independent Samples Test

		Levene's Test for Equality of Variances		t-test for Equality of Means						
									95% Confidence Interval of the Difference	
		F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	Lower	Upper
Distance_DPSAM	Equal variances assumed	247,923	,000	-139,057	19898	,000	-34,25815	,24636	-34,74104	-33,77526
	Equal variances not assumed			-138,954	19418,562	,000	-34,25815	,24654	-34,74140	-33,77491

:

4 FUTURE RESEARCH

The reward table of above described algorithm solely takes into account the current state *s* of the system and the action *a* taken. These reward tables can be substituted by reward functions or utility functions which include more parameters when determining the utility of an action. As such, apart from the starting time and the duration of the activity, the activity location, the position of the activity within the activity schedule and the activity history are also incorporated in these utility functions. *(30)* Moreover, such utility functions allow generating activity patterns based on socio-demographical data.

In addition, it can be useful to improve the learning algorithm through the substitution of the deterministic action set by a probabilistic one, signifying that - given a certain state - certain actions will be selected more likely than other (feasible) actions within the (limited) action set. This improvement will partly replace the constraints currently implemented in the algorithm.

As illustrated in this research, the impact of a key event on the activity-travel pattern can be reproduced. The effect of such events can be extended by introducing simultaneously changes to the reward table, the available action set in each state, the available transport modes, the available locations for each activity or the travel time tables for each transport mode and each location pair. This way also the effect of modifications in the environment, for example altering opening hours of shops of public authorities, or a change in availability or travel times of public transportation, can be simulated. Additionally, to serve the purpose of short-term, within-day rescheduling, the effect of unexpected events occurring during the execution of the daily activity schedule can be simulated as well.

Finally, the result of such micro-simulations, differentiated according to socio-demographical profiles, will be compared to real-data activity patterns extracted from activity-travel diaries, using the distance measure applied in this recent study. When this can be established, methodologies like the one advanced in this paper have the potential to be used as the founding learning algorithm for micro-simulation in dynamic activity-based transportation models.

5 CONCLUSIONS

This paper proposes an extended framework of learning activity-travel schedules based on reinforcement learning. The basis of the program was already developed by Janssens *(22)*. The most important modification to this program consists of enabling the algorithm to determine the activity-travel sequence autonomously, for the basis program was founded on a fixed, predefined order of activities. In addition to this, the transport mode used to travel from one location to the next, needed to be detached from this fixed sequence as well. Therefore this component is paired to the choice aspect location. Consequently, in order to generate feasible activity-travel patterns the algorithm applied in this research includes time constraints. Examples of such constraints are: opening hours of shops and limiting the set of possible activities for a certain point in time. As the results show, the produced activity-travel sequences contain more variation.

In addition to these changes, the algorithm allows simulating the impact of a key event on the development of an activity-travel schedule. The key event taken into consideration in this research is “obtaining one’s driving license”. This event is assumed to alter only the availability of transport modes. When comparing the generated activity-travel sequences before and after the occurrence of this key event, one can detect differences between these two sequences. Aiming at estimating these dissimilarities numerically, the distance between these sequences is calculated using a SAM-measure. For reasons of validation, the two runs are repeated 100 times. Subsequently, the distances between each sequence of every run are determined. Approaching these distances statistically, one can conclude that the algorithm is able to simulate the effect of the key event by generating different activity-travel patterns based on the alterations in the input (i.c. available transport modes).

REFERENCES

(1) Arentze, T.A. and H.J.P. Timmermans. A Learning-Based Transportation Oriented Simulation System. *Transportation Research Part B: Methodological*, Vol. 38, No. 7, 2004, pp.613-633.

(2) Arentze, T. and H. Timmermans. Modeling Learning and Adaptation in Transportation Contexts. *Transportmetrica*, Vol. 1, No. 1 - Special issue: Some Recent Advances in Transportation Studies, 2005, pp.13-22.

(3) Charypar, D., P. Graf and K. Nagel. Q-Learning for Flexible Learning of Daily Activity Plans. Proceedings of the 4th Swiss Transport Research Conference (STRC), Monte Verità, Ascona, Czechoslovakia, 2004.

(4) Charypar, D. and K. Nagel. Generating Complete All-Day Activity Plans with Genetic Algorithms. *Transportation*, Vol. 32, No. 4, 2005, pp. 269-397.

(5) Ettema, D. and H. Timmermans. *Activity-Based Approaches to Travel Analysis*. Elsevier Science Ltd, Oxford, UK, 1997, 1st edition.

(6) Vovsha, P., M. Bradley and J.L. Bowman. Activity-Based Travel Forecasting in the United States: Progress since 1995 and Prospects for the Future. Proceedings of the Conference on Progress in Activity-Based Models, Maastricht, The Netherlands, 2004.

(7) Bhat, C.R., J.Y. Guo, S. Srinivasan and A. Sivakumar. A Comprehensive Micro-Simulator for Daily Activity-Travel Patterns. Proceedings of the Conference on Progress in Activity-Based Models, Maastricht, The Netherlands, 2004.

(8) Pendyala, R.M., R. Kitamura, A. Kikuchi, T. Yamamoto and S. Fujji. FAMOS: Florida Activity Mobility Simulator. Proceedings of the 84th Annual Meeting of the Transportation Research Board, Washington D.C., 2005.

(9) Arentze, T.A. and H.J.P . Timmermans. Albatross 2: A Learning-Based Transportation Oriented Simulation System. European Institute of Retailing and Services Studies. Eindhoven, The Netherlands, 2005.

(10) Timmermans, H., T. Arentze, M. Dijst, E. Dugundji C.-H., Joh, L. Kapoen, S. Krijgsman, K. Maat and J. Veldhuisen. Amadeus: A Framework for Developing a Dynamic Multi-Agent, Multi-Period Activity-Based Micro-Simulation Model of Travel Demand. Paper presented at the 81st Annual Meeting of the Transportation Research Board, Washington DC, 2002.

(11) Arentze, T.A., H.J.P. Timmermans, D. Janssens and G. Wets. Modeling Short-Term Dynamics in Activity-Travel Patterns: from Aurora to Feathers. Paper presented at the Innovations in Travel Modeling conference, Austin, Texas, 2006.

(12) Arentze, T.A. and H.J.P. Timmermans. A Theoretical Framework for Modeling Activity-Travel Scheduling Decisions in Non-Stationary Environments under Conditions of Uncertainty and Learning. In: Proceedings International Conference on Activity-Based Analysis, Maastricht, The Netherlands, 2004.

(13) Arentze, T.A. and H.J.P. Timmermans. A Cognitive Agent-Based Simulation Framework for Dynamic Activity-Travel Scheduling Decisions. In: Proceedings Knowledge, Planning and Integrated Spatial Analysis, LISTA, 2005.

(14) Joh, C.-H., T.A. Arentze and H.J.P. Timmermans. Understanding Activity Scheduling and Rescheduling Behavior: Theory and Numerical Simulation. In B. Boots, et al. (eds.), *Modeling Geographical Systems*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2003, pp. 73-95.

(15) Joh, C.-H., T.A. Arentze and H.J.P. Timmermans. Activity-Travel Rescheduling Decisions: Empirical Estimation of the Aurora Model. Transportation Research Record, Vol. 1898, 2004, pp. 10-18.

(16) Arentze, T.A. and H.J.P. Timmermans. Modeling Learning and Adaptation Processes in Activity-Travel Choice. *Transportation*, Vol. 30, No. 1, 2003, pp.37-62.

(17) van der Waerden, P. and H. Timmermans. Key Events and Critical Incidents Influencing Transport Mode Choice Switching Behavior: An Exploratory Study. Proceedings of the 82nd Annual Research Board Meeting, Washington DC, 2003.

(18) van der Waerden, P., H. Timmermans and A. Borgers. The Influence of Key Events and Critical Incidents on Transport Mode Choice Switching Behavior: a Descriptive Analysis. Proceedings of the 10th International Conference on Travel Behavior Research, Lucerne, Swiss, 2003.

(19) Janssens, D., Y. Lan, G. Wets and G. Chen. Optimizing activity-travel sequences by means of reinforcement learning. BIVEC-GIBET Transport Research Day, Diepenbeek, Belgium, 2005, pp. 259-277. Also in: Allocating Time and Location Information to Activity-Travel Patterns through Reinforcement Learning, Proceedings of the International Association for Travel Behaviour Research, Kyoto, Japan, 2006, to appear.

(20) Kaelbing, L.P., M.L. Littman and A.W. Moore. Reinforcement Learning: A Survey. *Journal of Artificial Intelligence Research*, Vol. 4, 1996, pp. 237-285.

(21) Sutton, R.S. and A.G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, HTML-version:http://www.cs.ualberta.ca/~sutton/book/ebook/the-book.html,Cambridge,Massachusetts, USA/London, England, 1998.

(22) Janssens, D. Calibrating Unsupervised Machine Learning Algorithms for the Prediciton of Activity-Travel Patterns. Doctoral dissertation, Hasselt University, Faculty of Applied Economics, Belgium, 2005.

(23) Watkins, C.J.C.H. Learning from delayed rewards. Ph.D. Thesis, King's College, Cambridge, 1989.

(24) Watkins, C. and P. Dayan. Technical note: Q-learning. Machine Learning, 8, 1992, pp. 279-292.

(25) Bowman, J.L. and M.E. Ben-Akiva. Activity-Based Disaggregate Travel Demand Model System with Activity Schedules. *Transportation Research Part A: Policy and Practice*, Vol. 35, No. 1, 2000, pp. 1-28

(26) Dong, X., M.E. Ben-Akiva, J.L. Bowman and J.L. Walker. Moving from Trip-Based to Activity-Based Measures of Accessibility. *Transportation Research Part A: Policy and Practice*, Vol. 40, No. 2 , 2006, pp. 163-180.

(27) Algers, S., J. Eliasson and L.-G. Mattsson. Activity-Based Model Development to Support Transport Planning in the Stockholm Region. Presented at the 5th Workshop of the TLE Network, Nynäshamn, 2001.

(28) Joh, C.-H. T.A. Arentze & H.J.P. Timmermans. Pattern Recognition in Complex Activity-Travel Patterns: A Comparison of Euclidean Distance, Signal Processing Theoretical, and Multidimensional Sequence Alignment Methods", Presented at the 80th Annual Meeting of the Transportation Research Board, Washington D.C., USA, 2001.

(29) Hay, B., G. Wets and K. Vanhoof. Clustering Navigation Patterns on a Website Using a Sequence Alignment Method. Intelligent Techniques for Web Personalization: 17th International Joint Conference on Artificial Intelligence, Seattle, WA, USA, 2001, pp. 1-6.

(30) van Bladel, K., T. Bellemans, G. Wets, T. Arentze and H. Timmermans. Fitting S-Shaped Activity Utility Functions Based on Stated Preference Data. Proceedings of the 11th International Conference on Travel Behavior Research, Kyoto, Japan, 2006.