The Optimization of Activity-Travel Sequences by means of Reinforcement Learning

Davy Janssens[1] Yu Lan[2] Geert Wets[1] Guoqing Chen[2]

[1]Universiteit Hasselt, Campus Diepenbeek, Agoralaan - Gebouw D, B-3590 Diepenbeek
Email: {davy.janssens; geert.wets}@uhasselt.be
[2]School of Economics and Management, Tsinghua University, Beijing 100084, China
Email: {yul1; chengq}@em.tsinghua.edu.cn

Abstract

Given a sequence of activities and transport modes, this paper evaluates the use of a Reinforcement Machine Learning technique. The technique simulates time and location allocation for a given set of sequences and enables the prediction of a more complete and consistent activity pattern. A computer code has been established to automate the process and has been validated on empirical data.

Keywords: Reinforcement learning, Time allocation, location allocation, activity-based modeling.

1. Introduction

In the transportation research area, activity and travel modes are critically important information, based on which the transportation demand/status are simulated or predicted. Once the sequential activity-travel combination is known, such as for instance Sleep-Eat-*car*-Work-Eat-Work-*car*-Eat-*bike*-Shop-*bike*-Leisure-*bike*-Sleep, it is meaningful to observe how a learning agent can allocate time and location information for given activity-travel pattern combinations in a reasonable way. Given a constrained environment, we simulate and look into a learning agent's behavior under the framework of Reinforcement Learning, which is in fact a synonym for learning by interaction [1]. More specifically, the vector <activity, starting time, duration, location> denotes the agent's current state, where duration indicates how long the agent has spent on current activity. There are two actions available for each state: *Stay* (continue current activity for another time slot at the same location) or *Move* (travel to a possible location where the agent starts to perform the next activity in the pattern). At each state, the agent will receive a reward from the environment when any possible action is chosen. By accumulating this reward information that it obtained from its trial and error search in the state space, the agent finally gets the optimal/satisfactory time and location arrangement. Previous research work in this area generally deals with only one of these two allocation problems: they either focus on the time planning of the activity patterns [2], or search the shortest path in a dynamic programming way [3]. In reality, however, a rational person will consider the time and location arrangements simultaneously in order to achieve a total maximal reward. To the best of our knowledge, it is the first time that both problems are integrated and solved using Reinforcement Learning.

2. Reinforcement Learning

Under a constrained environment, the learning agent can perceive a set S of distinct states, which are normally characterized by a number of dimensions, and has a set A of actions to perform at each state. Reinforcement learning tasks are generally treated in discrete time steps. At each time step t, the agent observes the current state s_t and chooses a possible action to perform, which leads to its succeeding state $s_{t+1}=\delta(s_t, a_t)$. The environment responds by giving the agent a reward $r(s_t, a_t)$. These rewards can be positive, zero or negative. It is probable that these preferable rewards come with a delay. In other words, some actions and their consequential state transitions may bring low rewards in short term, while it will lead to state-action pairs later with a much higher reward. On the contrary, an action in a given state may receive an immediate high reward, whereas it makes the agent enter into a path where a series of actions followed, have very low or even negative rewards.

When the agent has observed the reward and state transition functions responded by its environment at any state, it is able to calculate the optimal action a at any state s. Q-learning [4] is a technique which is often used to select these actions, even when the agent has no full knowledge about the reward and state transition functions.

The Q-learning process does not specify however how actions are chosen by the learning agent. Reinforcement learning is a Markov decision process

(MDP), where the functions $\delta(s, a)$ and $r(s, a)$ depend only on current state and action-pairs. It is important that the agent selects actions in such a way that it visits all possible state-action pairs infinitely often. In each state the agent basically can choose from two kinds of behavior: either it can explore the state space or it can exploit the information already present in the *Q*-values. By choosing to exploit, the agent usually gets to states that are close to the best solution so far. Because of this, it can refine its knowledge about that solution and collect relatively high rewards. On the other hand, by choosing to explore states that are further apart from the current best solution, it is possible that it discovers a solution that yields higher rewards than the one already known. The strategy above is similar to the local and global search in most known optimization algorithms.

3. Location Allocation for Activity and Travel Combinations

In this section, a hypothetical example has been presented to improve the understanding of location allocation by means of *Q*-learning. A similar application has been presented in Charypar and Nagel [4] for a time allocation problem. The integration of time and location allocation in a more realistic environment is treated in the next section.

For location allocation, it is assumed that people try to maximize/minimize the reward/cost of its travel in total. Travel distance may not be an optimal measure for determining the burden of travel because it is plausible in a realistic situation that the distance between location A and location B is shorter than the distance between location A and C, while the travel time may be longer (for instance because of a better road network). Furthermore, it is possible that there is a difference in the transport mode that is used.

Translated into a context of *Q*-learning, the agent learns to find a travel policy that achieves maximal reward/minimal cost. It is assumed that the immediate reward of traveling between two locations depends upon the travel mode, and has a negative correlation with travel time.

Consider a simple example with the following simplifying assumptions to better understand the behaviour of the decision agent:

One activity-travel sequence: Home – *public transport* – work – *walk* – leisure – *walk* – shop – *public transport* – Home.

A state is characterized by the activity and current location, and is denoted as (a, l).

For a state, an action is to choose the location where the agent can perform the next activity in sequence. Activities can be carried out in a limited number of locations: Home at Location A, Work at Location B, Leisure at Location C or D and Shop at Location E or F.

Only the rewards that come from travel are learned to be maximized.

Parameter setting: Learning rate $\alpha = 1$; Discounting factor $\gamma = 0.9$.

In addition to these assumptions, reward tables are artificial and extremely simple, as shown in Table 1.

Table 1: An example of a simple reward table for travel

	Public transport						Walk					
	A	B	C	D	E	F	A	B	C	D	E	F
A	/	-12	/	/	-14	-16	/	/	/	/	/	/
B	-12	/	/	/	/	/	/	/	-8	-5	/	/
C	/	/	-	/	/	/	/	-8	/	/	-10	-4
D	/	/	/	-	/	/	/	-5	/	/	-6	-6
E	-14	/	/	/	-	/	/	/	-10	-6	/	/
F	-16	/	/	/	/	-	/	/	-4	-6	/	/

Taking these simplifying assumptions into account, Home and Work can only be carried out at location A and B. It is obvious that the agent only has to decide about the location of Leisure and Shop activities, and each of them has two possible choices. The remainder of this section illustrates the learning procedure of the agent.

After all state-action pairs are initialized as zeros, a random state *s* will be chosen. It should be recalled that the state is defined by an activity and an origin location. Assume that the agent first visits state (Work, B). In the third step of the learning procedure, the agent chooses a random action in order to explore the state space in an attempt to find a better solution than the one already know. Let us assume that action (destination) C has been chosen to perform the next activity Leisure. The travel mode lies on the sequence and here is walk. The updated *Q* (Work, B; C) thereby equals -8 + 0.9 * max (*Q* (Leisure, C; E), *Q* (Leisure, C; F) = -8. Assume that the agent selects to walk to E for Shop when it is at the new state (Leisure, C), *Q* (Leisure, C; E) turns to be -10 + 0.9 * max (*Q* (Shop, E; A)) = -10. As shown in Table 2, the agent visited

Table 2. Visited states per loop

Origin/ Activity	Home	Work	Leisure	Shop
A	4,8,12			
B		1,5,9,13		
C			2,10	
D			6,14	
E				3,15…
F				7,11

these states sequentially. These actions at each state and their corresponding updated *Q*-values are demonstrated in Table 2. The Q-values and the corresponding state-action pairs were shown in Table 3.

Table 3. Q-values and State-action pairs

Loop	Action	Q-value	Loop	Action	Q-value
1	C	Q(Work,B; C) = -8	9	C	Q(Work, B; C) = -8
2	E	Q(Leisure,C; E) = -10	10	F	Q(Leis. C; F) = -28.12
3	A	Q(Shop,E; A) = -14	11	A	Q(Shop, F; A) = -30.85
4	B	Q(Home,A; B) = -12	12	B	Q(Home, A; B) = -16.5
5	D	Q(Work,B; D) = -5	13	D	Q(Work, B; D) = -5
6	F	Q(Leisure,D; F) = -6	14	E	Q(Leis. D; E) = -18.6
7	A	Q(Shop,F; A) = -26.8	15	A	Q(Shop, E; A) = -28.85
8	B	Q(Home,A; B) = -16.5	…	…	…

The *Q*-values tend to converge when each state-action pair has been visited for a sufficient large number of times. Then at each state, the agent chooses the optimal action that achieves maximal *Q*-value, thus constructing a policy (chart), as shown in Table 4.

Table 4. Policy chart for iterations going to infinity

Origin/activity	Home	Work	Leisure	Shop
A	B			
B		D		
C			F	
D			E	
E				A
F				A

The optimal location allocation for this sample sequence is thus equal to:
Home (A) – *public transport* – Work (B) – *walk* – Leisure (D) – *walk* – Shop (E) – *public transport* – Home (A)…

4. Empirical Results

The previous section has independently considered location allocation in an artificial environment. In reality, however, the reward function will be more complex, there may exist a more refined time granular; an abundant number of locations may be available for a certain activity, and the distribution of these locations may be more disarrayed. In addition to this, people will simultaneously take the time and location arrangements into account in order to get a maximal reward in total. It is recalled that the reward of daily activities depends upon the duration as well as start time, people will not simply endeavor to obtain an optimal route for travel, since such a route design may not be perfectly suitable for the time arrangement of daily activities. On the other hand, when people allocate time for activities, they have to consider the flexible travel times since a number of locations are available for the next activity. The time and location arrangements are therefore interacted. We will integrate the two problems under the framework of *Q*-learning in a more realistic environment, which can be described as follows:

The elements of sequences are limited to four kinds of activities (i.e. Home, Work, Shop and Leisure) and four kinds of travel modes (i.e. walk, bike, car and public transport).

Time of the day is discretized with a refined time slot of 15 minutes, and the maximal duration of each activity is 12 hours.

A state *s* is characterized by activity, starting time of activity, time already spent at activity (duration) and the origin location where the activity is performed.

For a state *s*, an action *a* may be to Stay: keep performing the activity at current location for another time slot, or to Move: move to a possible location where it starts to perform the next activity. The travel mode the agent uses to reach these locations is determined by the sequence.

The reward functions of these four activities are artificial.

With respect to location allocation, 100 locations were collected in a city and we recorded the distances among them. Of these 100 locations, 20 locations are available for Shopping, and 15 for Leisure. For each person, there is only one location available, both for Home and for Work.

For each travel mode, the travel time among these locations are logged. It is assumed that the reward/cost function in term of travel time is as follows:

$$\text{Reward}\ (t) = -c*(b*t)^{a}$$

,where *c* is identical for all travel modes and is applied to easily control the relative importance of travel compared with daily activities. The parameters *b* and *a* are specifically set for each travel mode in order to respectively dominate the range of reward and its evolution trend. The setting of these parameters are shown in Table 5 by example.

In such a complicated environment, it is required for the learning agent to look far into the future in order to find a good daily plan of time and locations. The discounting factor γ is set at 0.99, which is close

Table 5. Parameter setting for reward functions of each travel mode

	Walk	Bike	Car	Public transport
A	1.6	1.1	0.6	0.8
B	1/12	1/10	1/6	1/8
C	5			

to one and makes the learning procedure harder to converge.

Due to the use of discrete time intervals, the starting time of activity is calculated as the ending time of previous activity plus, instead of real travel time, the minimal number of time slots that contains the travel time. It is expected that this adaptation causes trivial influence because of the small time granular.

Furthermore, the discount per time slot should be the same during the learning procedure. As a result, the discount factor is equal to γ^m if it takes the agent *m* time slots to travel to the next location.

Three sequences were dealt with in this paper by means of example:

1. Home – *car* – Work – *car* – Shop – *car* – Leisure – *car* – Home
2. Home – *public transport* – Work – *public transport* – Home – *bike* – Leisure – *bike* – Shop – *bike* – Home
3. Home – *public transport* – Work – *walk* – Leisure – *walk* – Shop – *public transport* - Home

The optimal behaviour of three persons are presented for each pattern by means of example. The home and location pair for each person can be listed as follows:
Person A: Home – location 7; Work – location 82
Person B: Home – location 29; Work – location 9
Person C: Home – location 30; Work – location 54

The outputs are displayed below. For example, when person A chooses sequence 2 for everyday life, he/she would like to stay at home from 22:30 P.M. to 6:45 A.M., and then moves by public transport to location 82. At 17:30 PM, he/she stops working and returns home. Person A does not spend in home time (which means that the home activity that was assumed to exist in the given sequence, is skipped) and directly rides bicycle to location 0 for Leisure. After two hours leisure, he heads to location 6 for one hour's shopping. Finally, he starts to move by bike back home at 22:00 P.M.

Person A:

H(23:15 --07:15, 7), W(07:45 --18:00, 82), S(18:15 --19:45, 87), L(20:00 --23:00, 0)
H(22:30 --06:45, 7), W(07:45 --17:30, 82), H(18:30 --18:30, 7), L(18:45 --20:45, 0), S(21:00 --22:00, 6)
H(23:15 --06:45, 7), W(07:45 --17:30, 82), L(18:45 --20:30, 33), S(21:00 --22:00, 36)

Person B:

H(23:30 --07:15, 29), W(07:45 --18:00, 9), S(18:15 --19:45, 11), L(20:00 --23:00, 10)
H(22:30 --06:15, 29), W(07:45 --17:15, 9), H(18:45 --18:45, 29), L(19:15 --20:45, 0), S(21:00 --22:00, 3)
H(23:15 --06:30, 29), W(08:00 --17:45, 9), L(18:30 --20:15, 10), S(21:00 --22:00, 11)

Person C:

H(23:30 --07:30, 30), W(08:00 --18:00, 54), S(18:15 --19:45, 55), L(20:00 --23:00, 33)
H(22:30 --06:30, 30), W(07:45 --17:30, 54), H(18:45 --18:45, 30), L(19:00 --20:30, 27), S(21:00 --22:00, 25)
H(23:15 --06:45, 30), W(08:00 --18:00, 54), L(18:45 --20:30, 39), S(21:00 --22:00, 38)

As mentioned above, the equation Reward (*t*)= $-c*(b*t)^a$ is applied to calculate the travel reward (cost). We also run our optimization program when *c* is set as infinite large. Suppose that sequence 1 is adopted, person A prefers location 83 over location 87 for Shop, and person C prefers location 39 over location 33 for Leisure. The output is the result of the fact that in sequence 1, traveling by car suffers from low cost and the route arrangement is often subject to the activity arrangement in order to achieve highest reward in total, while traveling by public transport, bike or walk is costly and the route should also be carefully designed to alleviate the travel cost as much as possible.

5. Conclusion

The main contributions of the paper to the current state-of-the art are the allocation of location information in the simulation of activity-travel patterns, the non-restriction to a given number of activities and the incorporation of realistic travel times. Furthermore, the time and location allocation problem were treated and integrated simultaneously, which means that the respondents' reward is not only maximized in terms of minimum travel duration, but also simultaneously in terms of optimal time allocation.

6. References

[1] L.P. Kaelbling, M.L., Littman and A. Moore "Reinforcement learning: a survey," *Journal of Artific. Intell. Research* ***4***, pp. 237-285, 1996.

[2] D. Charypar, P., Graf, and K. Nagel. "Q-learning for flexible learning of daily activity plans, " *Proc. of the Swiss Transport Research Conference (STRC)*. 2004.

[3] E. Dijkstra,"A note on two problems in connection with graphs," *Numerical Mathematics* ***1***, pp. 269-271, 1959.

[4] C. Watkins and P. Dayan, "Technical note: Q-learning," *Machine Learning* ***8***,pp.279-292, 1992.