A modelling of route choice behaviour in transportation networks: an approach from reinforcement learning

T. Miyagi
Department of Regional Studies, Gifu University, Japan

Abstract

We present a simple, but very powerful traffic equilibrium calculation method. The basic idea of the method is motivated from reinforcement learning with profit sharing. In our model, individual driver is regarded as heterogeneous entity, being assumed to form his or her own value for each route through driving experiences and communications to the environment. Our method realizes a disaggregate user equilibrium on a congested network so that it is useful to analyse the interrelationships between each driver’s characteristic and the resultant network equilibrium. Moreover, this method not only covers from stochastic user equilibrium to deterministic user equilibrium, but it is also applicable to a network with asymmetric cost functions or with discontinuous cost functions.

Keyword: route choice behaviour, disaggregate user equilibrium, Nash equilibrium, reinforcement learning with profit sharing, a fixed-point problem.

1 Introduction

Nowadays, understanding individual travel behaviour becomes increasingly essential for designing Advanced Traveller Information Systems (ATIS). However, the response of road users to information is still an open question. The main purposes of this paper is to propose a model useful for analysing the interaction of individual driver’s route choice behaviour and user equilibrium in congested networks.

For that purpose, we use reinforcement learning approach based on profit sharing, a type of reinforcement learning originally proposed by Grefenstette [1]. Reinforcement learning with profit sharing is machine learning and categorized
into non-bootstrap method in reinforcement learning classes. It simulates behaviours of multi-agents who are assumed to form their own value through their own experiences and communications to the environment [2]. This characteristic seems to be appropriate to build a model for analysing route choice behaviours of heterogeneous individual drivers.

However, reinforcement learning is a simulation-oriented model, there is no theoretical background that relates the model with route choice behaviour of individual in congested networks. In addition, a direct application of reinforcement learning to network equilibrium problems produces undesirable results; it overlooks the existence of unused paths; and it takes thousands of times or more iteration in computation to find user equilibrium flow patterns even in a single origin-destination network connected with two or three paths.

To resolve those issues are the main motivation of this paper. We will present a new model and seek the underlying theoretical background that links the route-choice model to user equilibrium on congested network. For that purpose, we use Brower’s fixed-point theorem [3] and relate the convergence properties of the model with the user equilibrium. This method provides a simple, but very powerful traffic equilibrium calculation procedure. It not only covers from stochastic user equilibrium to deterministic user equilibrium, but also it is applicable to networks with asymmetric cost functions or with discontinuous cost functions. Furthermore, since it is a disaggregate model in its nature, we can get a deep insight into the interaction between individual route choice behaviour and the network equilibrium.

2 Flows on transportation networks

We consider networks described by the 3-triple of a set of nodes, $V$, a set of links, $L$, and a set of ordered pairs of centroids (or origin-destinations), $K$. These links are assumed L in number and indexed by $\ell$, and pairs of OD are assumed K in number and indexed by $k$.

The flow on link $\ell$ is denoted by $f_\ell$, $f = (f_1, \cdots, f_\ell)'$, where (’) represents a transpose of a vector or of a matrix. We assume that OD pair $k$ is connected by $m_k \geq 1$ paths (or routes), whose set is denoted by $P_k$. A path set of the whole OD pairs is denoted by $P = \bigcap_{k \in K} P_k$ with the number of elements, $M = m_1 + \cdots + m_K$. The flow on path $p$ is denoted by $h_p$, $h = (h_1, \cdots, h_M)'$. The demand for travel on OD pair $k$ in a fixed period is assumed a positive number $i_k, k \in K$. If a traveller has his own value of time and makes a decision on path choice under different environment, each trip can be regarded as the result of distinct decision-making. Therefore, a trip and individual driver is interchangeably used. Individual driver is distinguished by index $i$, of which set is denoted by $I$. We represent the total number of individual drivers (trips) by $I = \sum_{k \in K} i_k$. The choice set of paths available to driver $i$ is denoted by $C_i$. 
We assume that the travel cost on link $\ell$ is a nondecreasing function of $f$, and denote it by $c_\ell(f) = c_\ell(f, \cdots, c_\ell(f))$. The link travel costs are assumed a combination of travel times and monetary expenditure; travel times will be transformed to monetary units by the value of time of individual traveller, $w = (w_1, \cdots, w_r)'$. For the time being, the value of time for each traveller is assumed the same. The travel cost on path $p$, $u_p(h)$, will be expressed as a function of the flows of all paths: $u(h) = (u_1(h), \cdots, u_\gamma(h))$.

The occurrence of links on a path is specified by the link-path incidence matrix, $\Delta$, of which elements are described by means of Kronecker numbers:

$$
\delta_{ip} = \begin{cases} 
1 & \text{if link } \ell \text{ is on the path } p, \\
0 & \text{otherwise.} 
\end{cases}
$$

The flows on links can be then expressed in terms of flows on paths:

$$
f = \Delta h
$$

In addition, the costs on all of the paths are expressed by

$$
u(h) = c(\Delta h) \Delta.
$$

Now we introduce a choice-probability matrix, $X = (x_1, \cdots, x_i, \cdots x_I)$, whose elements, $\{x_{ip}\}$, describe the choice probability of path $p \in C_i$ by a driver $i$, and $X$ must satisfy the conditions,

$$
\sum_{p \in C_i} x_{ip} = 1, i \in I \text{ and } \\
\sum_{i=1} x_{ip} = h_p, p \in P
$$

Let $Ix_{ip} = h_p(i), p \in P$ be the path flow when all drivers are assumed the same class of $i$. A trivial operation, $\sum_{i=1} x_{ip} = \sum_{i=1}^l h_p(i) / I = h_p$, explains a path flow to be the weighted average of path flows generated by all classes of users. We call the fraction of the total trip $\{x_i\}$ the flow element.

Since individual trip $i$ has its specified OD pair, say $k \in K$, the only elements of $x_i$ corresponding to paths included in $P_k \cap C_i$ have positive numbers. To define path choice set $C_i$ in connection with OD pairs, we prepare two matrices. Let $\Lambda$ be the incidence matrix between OD pair $k \in K$ and path $p \in P$, and $\Psi$ be the incidence matrix between a trip $i \in I$ and OD pair $k \in K$. Then we have $\Gamma = \Psi \Lambda$, of which elements are:

$$
\gamma_{ip} = \begin{cases} 
1 & \text{if } p \in C_i \text{ and } p \in P_k \\
0 & \text{otherwise} 
\end{cases}
$$

If we define an element-by-element multiplication of a matrix $X$ and a matrix $\Gamma$ by $X \cdot \Gamma = \{x_{ip} \cdot \gamma_{ip}\}$, $X \cdot \Gamma$ provide the flow elements in the context of path-choice sets of individuals. From the definition on $X$ we have
\[ f = AX'e \]  \hspace{1cm} (4)

where \( e \) represents a column vector with ones. Moreover, path costs are expressed by the flow elements:

\[ u(X) = c(X)A. \]  \hspace{1cm} (5)

3 Models

3.1 Reinforcement learning

Reinforcement learning with profit sharing allows competitive drivers to learn effective behaviours from their experiences in a given environment. It is assumed that the environment consists of a transportation network with nodes and links, a system administrator and the other drivers. The system administrator is assumed to have a role of providing route information to drivers. In this model, a driver is connected to its environment via perception and action. A driver’s action is stated as a dynamic route choice behaviour that may change according to the state space fluctuating day by day when the driver makes a decision. The process of moving from the starting state to the final reward state is known as an episode. Thus, a trip defined in a transportation network forms an episode.

On each step of interaction the driver receives as input, \( s_i \), some indication of the current state, \( s \), of the environment; the driver then chooses a route, \( p \), to generate as output. We assume that the driver views the exact state of the environment through link cost functions and assess them by their value of time. The action changes the state of the environment, and the value of this state transition is communicated to the driver through a reinforcement signal, or reward. The driver’s behaviour should choose a route (or a path) that tends to increase the cumulative sum of values of the reinforcement signal. Other drivers within the environment are also learning independently of each other, without sharing sensory inputs or policies. Thus, the other drivers appear as additional components within the environment, whose behaviour is unpredictable. We assume that there is no communication among drivers.

Let \( t \) be the time period of an episode or a trip. Let \( S' \) and \( C_i^t (i \in I) \) be a set of environment states and a choice set of driver \( i \). Initially, the driver \( i \) observes \( s^t \subseteq S^t \), the partially available state of its environment at time \( t \). A choice is then selected from the choice set \( C_i^t \), which contains all the available paths at time \( t \). The probabilities of choosing a path \( p \in C_i^t \) of individual \( i \in I \) is defined as:

\[
x_i(s', p) = \frac{Q_i(s', p)}{\sum_{p \in C_i^t} Q_i(s', p)} \hspace{1cm} (6)
\]

where \( Q_i(s', p), \hspace{1cm} p \in C_i, i \in I \) defined for the state-choice pair, \((s,p)\), will be called the choice evaluation function. The sum of choice evaluation function over available paths define the value function of a driver \( i \) for his choice:
\[ V(s', t) = \sum_{p \in \mathcal{C}_i} Q_i(s', p) \]

After the choice is made, the driver determines if a reward has been generated. If there is some reward after choice \( p \in \mathcal{C}_i \), the driver stores the state-choice pair, \( \{s, p\} \), in its episodic memory, and repeats this cycle until no reward is generated. The term, “state-choice pair”, gives a “rule” on how drivers increase their choice intensity on specific route. Once the driver receives the reward \( r'_{ip} \), it reinforces the choice-rules stored in its episodic memory by modifying the state-choice values using the following updating formula:

\[ Q_i(s^{t+1}, p) = Q_i(s', p) + r'_{ip} \]

This formula is rewritten as:

\[ Q_i(s^{t+1}, p) = Q_i(s^0, p) + r'_{ip} + r'_{ip} + \cdots + r'_{ip} \]

Thus, the evaluation value of specific path is determined based on the history of trip memory of that path. This updating formula may be called the knowledge-base rule. The next formula is also available that is based on the acquainted information:

\[ Q_i(s^{t+1}, p) = Q_i(s', p) + \alpha \{r'_{ip} - Q_i(s', p)\} \]

where \( \alpha(0 < \alpha \leq 1) \) is a stepsize parameter or the rate of learning.

Since we assume that drivers perceive their environment via path cost functions, the choice evaluation functions may be specified by

\[ Q^{'t+1}_{ip} = Q^{'t}_{ip} + x^{'t}_{ip} \{\hat{u}_i(X') - \tilde{u}_ip(X')\} \quad (7) \]

where the choice evaluation function \( Q_i(s^t, p) \) is simplified to \( Q^{'t}_{ip} \) and the reward \( r'_{ip} \) is replaced by \( x^{'t}_{ip} \{\hat{u}_i(X') - \tilde{u}_ip(X')\} \). The newly defined variables included in the reward function are follows:

\[ x^{'t}_{ip} = \frac{Q^{'t}_{ip}}{\sum_{p \in \mathcal{C}_i} Q^{'t}_{ip}} \quad \text{and} \quad (8) \]

\[ r_p(X') = x^{'t}_{ip} \{\hat{u}_i(X') - \tilde{u}_ip(X')\}, \text{ where } \hat{u}_i(X') = \max_{p \in \mathcal{C}_i, x_{ip} > 0} \tilde{u}_ip(X') \quad (9) \]

You should note that path cost functions are no longer vector valued.

Suppose that path costs are evaluated by travel times along these paths. Because of the value of time distributed among drivers and additional travel costs such as toll imposed on some paths, perceived path costs are driver-specific and defined by

\[ \tilde{u}_{ip}(X) = z_p(X) + w_iu_p(X) \text{ or } \tilde{u}_i(X) = z(X) + w_iu(X) \quad (10) \]

where \( z(X) = (z_1, \ldots, z_p, \ldots, z_M)^t \) is a vector of tolls imposed on paths.

While a driver can know his travel cost at the current day through his experience, he cannot recognize the most expensive route of the day. We assume that information on routes available are given by the system administrator. However, some drivers may not have information on routes available to them.
This situation is described by replacing \( \hat{u}_i(X') = \max_{p \in C_i} \tilde{u}_i(X') \) to \( \hat{u}_i \) where \( \hat{u}_i \) is the cost of most expensive route that the driver estimates. In spite of with or without route information, the most expensive cost is a signal for a driver to select a route, by which a driver knows whether he can get a positive reward or not. Throughout this paper, we assume that every driver has complete information on routes available to him so that the system is expected to be in a user equilibrium in Wardropian sense.

### 3.2 Disaggregate user equilibrium: Nash equilibrium

We will examine that the iterative calculation method provided by eqns (7), (8) and (9) achieves a user equilibrium state. We only show the sketch of the proof for the convergence properties of the procedure.

Let \( n_i \) be the number of paths contained in \( C_i \). Flow elements will be nonnegative vector of dimension \( n_i \) summing to one for each driver. For each diver, the flow elements form a simplex of dimension \( n_i - 1 \):

\[
S^{n_i-1} = \{ x_i \in \mathbb{R}_{+}^{n_i} | \sum_{p=1}^{n_i} x_{ip} = 1 \}.
\]

Therefore, the whole space spanned by all drivers is \( S^{n_1-1} \times \cdots \times S^{n_I-1} \), of which dimension is \( n_1 + \cdots + n_I - I \). Let \( N = n_1 + \cdots + n_I - I + 1 \) be the number of vertices of the simplex. Then, the flow elements are defined on the simplex:

\[
X \in S^{N-1} = \{ X \in \mathbb{R}^N_+ | x_1 \geq 0, \cdots, x_I \geq 0, x_ie = 1, \cdots, x_i e = 1 \}
\]

From eqns (7) and (8), we understand that to find the flow elements on congested networks becomes the problem of finding a fixed point defined on the simplex:

\[
X = F(X)
\]

First, we assume that the link cost function is continuous, convex. Then, the perceived path cost is also continuous, convex and \(-\tilde{u}_i(X)\) is continuous concave. Therefore, the reward function in eqn (9) is continuous. Since the simplex is compact and convex and \( F : S^{N-1} \rightarrow S^{N-1} \) is continuous, by Brower’s fixed-point theorem [3] there exists \( X^o \in S^{N-1} \) for which \( X^o = F(X^o) \).

Next, we will show that if eqn (11) has a fixed point, \( X^o \), and then it is a user equilibrium flow pattern. It follows from eqn (7) that at the point \( X^o \) it should be \( Q_{ip}^{o+1} - Q_{ip}^f = 0 \), and then

\[
x_{ip}^o \{ \hat{u}_i(X^o) - \tilde{u}_i(X^o) \} = 0
\]

This implies that if \( x_{ip}^o > 0 \), then \( \hat{u}_i(X^o) = \tilde{u}_i(X^o) \), constant, and that if \( \hat{u}_i(X^o) \geq \tilde{u}_i(X^o) \), then \( x_{ip}^o = 0 \). However, inequality does not hold; if so, driver \( i \) can get a positive reward from path \( p \) such that his choice probability for the path is increasing. This contradicts \( X^o \) be a fixed point.
For paths with almost zero flows, if rewards from those paths are also almost zero, then they are eliminated from the choice set of driver \( i \). This operation seems to be inconsistent with the definition of the choice probability, however, as far as the link costs do not change dramatically at \( \varepsilon \)-neighbourhood of zero flows, the operation does not affect the equilibrium point.

Summarising the above results, we conclude: *equilibrium will be reached when no traveller can improve his perceived cost by unilaterally changing paths.* This is *Nash equilibrium*. Since we assume heterogenous drivers with different values of time, the equal travel cost principle holds only for individual driver. Nash equilibrium coincides with *Wardrop equilibrium* only if all drivers have the same values of time.

Now, assume that some of the link-cost functions are discontinuous at \( X^* \in S^{N-1} [4] \):

\[
c_\ell(x^*) = \lim_{x \rightarrow x^*} c_\ell(x) = \lim_{\varepsilon \rightarrow 0} \inf \{ c_\ell(x) \mid x - x^* \leq \varepsilon \} \quad (13)
\]

Then, the reward functions will be upper semicontinuous. According to Asmuth’s theorem [5], it can be shown that the variational inequality problem (or the stationary problem),

\[
\sum_{p \in \mathcal{P}} (x_{ip}^* - x_{ip}) r_{ip}(X^*) \geq 0, X, X^* \in S^{N-1}
\]

has a solution. Therefore, equilibrium exists such that \( x_{ip}^* (\hat{u}_l(X^*) - \hat{u}_p(X^*)) \geq 0 \).

This equilibrium conditions state that even though a equilibrium exists, the costs of paths used by individual driver are not necessary equal. This result may characterize Nash equilibrium with discontinuous cost functions.

**Figure 1:** A network with a single OD pair.

### 4 Numerical experiments

A network for numerical experiments that is quoted from Braess [6] is depicted in fig. 1. It is assumed that eight trips between origin-destination pair have three
alternative paths (as indicated in dashed lines in the fig. 1). We assume that all drivers have the same values of time except for the fourth experiment.

4.1 Route choice behaviour and user equilibrium

Assume that the link travel cost functions are:

\[ c_1(f_1) = f_1 + 50, \quad c_2(f_2) = f_2 + 50 \]
\[ c_3(f_3) = 4f_1, \quad c_4(f_4) = 4f_4, \quad c_5(f_5) = f_5 + 10 \]

Different from the original Braess’s problem, the user equilibrium principle generates the flow pattern \( h_1 = (0.0,0.0,8.0)' \) where only the third path has a positive flow, resulting in the path cost: \( u_1 = (82.0,82.0,82.0)' \). However, a toll pricing policy of imposing 28 units on link 5 generates completely different flow pattern from the first case: \( h_2 = (4.0,4.0,0.0)' \) and \( u_2 = (70.0,70.0,70.0)' \). Alternative pricing policy, say the policy of imposing 21 units on link 5, yields the flow pattern \( h_3 = (3.0,3.0,2.0)' \) with \( u_3 = (73.0,73.0,73.0)' \).

The choice probability matrix \( X_3 \) corresponding to the third experiment is given in Table 1. If each driver chooses the path with the highest probability, then the resultant flow pattern will be \( \bar{h} = (2,4,2) \). This implies that if you extract a time span from long range of period and observe the traffic conditions of that moment, you may not be able to find any equilibrium state in the Wardropian sense. The observed choice probability distribution as shown in Table 1 can be regarded as a sample from the most probable state of the system.

<table>
<thead>
<tr>
<th>( i )</th>
<th>( x_{i1} )</th>
<th>( x_{i2} )</th>
<th>( x_{i3} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.3913</td>
<td>0.5147</td>
<td>0.0940</td>
</tr>
<tr>
<td>2</td>
<td>0.3424</td>
<td>0.4607</td>
<td>0.1969</td>
</tr>
<tr>
<td>3</td>
<td>0.1030</td>
<td>0.7161</td>
<td>0.1808</td>
</tr>
<tr>
<td>4</td>
<td>0.6756</td>
<td>0.0587</td>
<td>0.2657</td>
</tr>
<tr>
<td>5</td>
<td>0.7054</td>
<td>0.0262</td>
<td>0.2684</td>
</tr>
<tr>
<td>6</td>
<td>0.3697</td>
<td>0.5597</td>
<td>0.0706</td>
</tr>
<tr>
<td>7</td>
<td>0.1711</td>
<td>0.3560</td>
<td>0.4729</td>
</tr>
<tr>
<td>8</td>
<td>0.2415</td>
<td>0.3079</td>
<td>0.4506</td>
</tr>
<tr>
<td>( \sum_i x_{ip} )</td>
<td>3.0000</td>
<td>3.0000</td>
<td>2.0000</td>
</tr>
<tr>
<td>Perceived cost</td>
<td>73.0000</td>
<td>73.0000</td>
<td>73.0000</td>
</tr>
</tbody>
</table>

Suppose that driver 1 and 2 have relatively high time values, say 20% up. We assume the same setting used in the second experiment. In this case, those drivers choose path 3 with relatively high choice probabilities. The resultant flow pattern is \( h_4 = (3.3333,3.3333,1.3333)' \). Drivers 1 and 2 sometime use path 3 with addition payment because of the fastest route, however, others never use...
that path. That causes a variety of the average perceived costs among three paths, as is shown in Table 2. If we increase the values of time of the two drivers to 50% up, then the choice probabilities of two drivers for path 3 become ones. The flow pattern is then given by \( h_5 = (3.0, 3.0, 2.0)' \).

Table 2: Perceived travel costs: mixed-users example.

<table>
<thead>
<tr>
<th>( i )</th>
<th>( \tilde{u}_{i1} )</th>
<th>( \tilde{u}_{i2} )</th>
<th>( \tilde{u}_{i3} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>86.4000</td>
<td>86.4000</td>
<td>86.3999</td>
</tr>
<tr>
<td>2</td>
<td>86.4000</td>
<td>86.4000</td>
<td>86.3999</td>
</tr>
<tr>
<td>3</td>
<td>72.0000</td>
<td>72.0000</td>
<td>-</td>
</tr>
<tr>
<td>4</td>
<td>72.0000</td>
<td>72.0000</td>
<td>-</td>
</tr>
<tr>
<td>5</td>
<td>72.0000</td>
<td>72.0000</td>
<td>-</td>
</tr>
<tr>
<td>6</td>
<td>72.0000</td>
<td>72.0000</td>
<td>-</td>
</tr>
<tr>
<td>7</td>
<td>72.0000</td>
<td>72.0000</td>
<td>-</td>
</tr>
<tr>
<td>8</td>
<td>72.0000</td>
<td>72.0000</td>
<td>-</td>
</tr>
<tr>
<td>Perceived cost</td>
<td>72.8409</td>
<td>74.0392</td>
<td>86.3999</td>
</tr>
<tr>
<td>Toll</td>
<td>0.0</td>
<td>0.0</td>
<td>28.0</td>
</tr>
<tr>
<td>Travel time</td>
<td>72.0000</td>
<td>72.0000</td>
<td>48.6666</td>
</tr>
</tbody>
</table>

### 4.2 Applications to networks with ill-defined cost functions

#### 4.2.1 Asymmetric cost function

Let us change the link cost functions such that the link travel costs of the entrance links (link 1 and 5) to link 4 are influenced by the flow on the forward link and additional delays are observed.

\[
c_1(f_1) = f_1 + 3f_4 + 50, \quad c_2(f_2) = f_2 + 50 \\
c_3(f_3) = 4f_1, \quad c_4(f_4) = 4f_4, \quad c_5(f_5) = 3f_4 + f_5 + 10
\]

We see that the flow pattern is given by \( h_6 = (2.7170, 4.6981, 0.5849) \) together with \( u_6 = (75.8302, 75.8303, 75.8304) \).

#### 4.2.2 Discontinuous cost function

Assume that the link cost function of link 5 is the following step function.

\[
c_5(f_5) = 17 \text{ if } f_5 \leq 2, \quad c_5(f_5) = 24 \text{ if } f_5 \leq 3 \\
c_5(f_5) = 31 \text{ if } f_5 \leq 4, \quad c_5(f_5) = 38 \text{ if } f_5 \leq 8
\]

The flow pattern asymptotically converges to \( h_7 = (2.5000, 2.4999, 3.0001) \) and \( u_7 = (74.5010, 74.5004, 75.0018) \). If the flow on path 3 falls in 3.0, the travelling cost of link 5 gets down to 24, reducing the travelling cost of path 3 to 68. However, This causes the increasing flow on path 3 again. The user equilibrium flow pattern in Wardropian sense is never realized in this case, however, Nash equilibrium exists. This can be examined by reassigning a small amount of flow, \( \rho \), on path 3 to path 1 and 2 by \( \rho / 2 \).
5 Conclusion

The method presented in this paper can deal with comprehensive and general network problems such as a mixed traffic assignment including mass transit, price policy assessments and AITS. It should be in order here to touch on that the method is also applicable to networks with multi-OD pairs and that further improvement on computational efficiency is one of the issues to be resolved.

References


