User Modelling Approach to Adaptive Route Selection in Intelligent Vehicle Navigation

Kyounga Park,
Email: k.a.park@imperial.ac.uk
Centre for Transport Studies
Imperial College London
SW7 2BU London, UK
Tel: +44 20 75945952
Fax: +44 20 75946102

Michael G.H. Bell,
Email: m.g.h.bell@imperial.ac.uk
Centre for Transport Studies
Imperial College London
SW7 2BU London, UK
Tel: +44 20 75946091
Fax: +44 20 75946102

Ioannis Kaparias
Email: ioannis.kaparias@imperial.ac.uk
Centre for Transport Studies
Imperial College London
SW7 2BU London, UK
Tel: +44 20 75945952
Fax: +44 20 75946102

Klaus Bogenberger
Email: klaus.bogenberger@bmw.de
Corporate Quality – Product Monitoring
BMW Group
AQ-11, 80788 Munich, Germany
Tel: +49 89 38230764
Fax: +49 89 38241136

This paper was submitted to the 11th World Conference on Transport Research

1, May, 2007

Abstract

User modelling is a process used to personalise services or information according to the user's specific requirements or preferences in the Artificial Intelligence (AI) domain. It involves formulating a model from the user's behaviour, but instead of requesting any information directly from him/her, the information needed is retrieved while he/she is making use of the system. This paper describes a way of applying user modelling to route guidance in the light of a user-friendly and intelligent route guidance strategy. An advanced function of the navigation scheme is to propose a single route from a set of feasible ones between an origin and a destination, with respect to the user preferences learnt from past journey records. Regularities in route selection are discovered using a decision tree learning algorithm, a widely used method in user modelling, which has many advantages over other techniques, in terms of the simplicity and comprehensibility of the model structure. In order to analyse the learning ability of the model developed in this study and to measure its performance based on predictive accuracy and error rates, simulation experiments are carried out. Compared with the experimental results using a multinomial logit model, it is found that the decision tree learning algorithm is more advantageous, not only due to its more intelligible structure, but also due to its higher predictive accuracy.

1. Introduction

As information overloads daily life through sources such as the Internet, a process of providing personalised information services to fit the different needs of individuals becomes more and more important. The research interest of this function leads to the application of *user modelling* in information provision services of this kind. User modelling, originally used in the development of education software for students with different learning abilities, is a useful approach to capture user preferences without any information being directly input by the user but being retrieved while the user is making use of the system instead.

User modelling is an attractive approach in the direction of improving user satisfaction with navigation systems. The development of more user friendly navigation systems is one of key issues in introducing advanced functions to navigation systems. A study on users' satisfaction with navigation systems by Schofer et al. (1997) shows that incorporation of user preferences in routing is one of the most demanded functions by the drivers who have even benefited from dynamic route guidance. A reasonable strategy to meet the demands from the user's side may be to develop a personalised route guidance system, capable of suggesting a route by reflecting the driver's preferences as closely as possible.

A key element of personalised route guidance is to collect user feedback along with route characteristics, in order to obtain knowledge on user-specific preferences. As a way to collection of user feedback, it is considered to observe the user's choice among multiple routes between an origin and a destination, as employed in Pang et al. (1999) and Rogers et al. (1999). In these studies, user feedback is collected by observing if the route selected by the route selection rules of the system is the same as the actual choice of the user. In this context, a route selection function integrated with a user modelling method, which can indicate the route that best fits user preferences, can play an important role in personalised route guidance.

One of the most widely used techniques of user modelling is *machine learning* - a synonym of *data mining* in statistical analysis - where *learning* refers to inferring a hypothetical function from training examples (Mitchell, 1997). Machine learning aims to provide increased levels of automation, replacing a great amount of time-consuming human activity with automatic techniques that improve accuracy and efficiency by discovering and exploiting regularities in the observations. Many methods are used in machine learning and can be classified into the following eight general methods: decision tree learning (DTL), artificial neural networks (ANN), instance-based learning, genetic algorithms, analytical learning, Bayesian learning, reinforcement learning and a combined approach of inductive learning and analytical learning (Mitchell, 1997). It is noteworthy that ANN, DTL and reinforcement learning have been applied to modelling travellers' choice behaviour as rule-based approaches (Arentze et al., 2003; Arentze et al., 2004; Arentze et al., 2005; Nakayama et al., 2000; Nakayama et al., 2001; Pang et al., 1999; Rogers et al., 1999; Timmermans et al., 2003)

Among the techniques mentioned, the DTL method is of particular interest since it offers the ability to produce an intelligible model. Fundamentally, DTL is a technique for discovering regularities in observations through exhaustive classification into subsets which are as homogeneous as possible, while at the same time formulating and outputting a hierarchical tree structure. Human readability is of great importance in an adaptive route choice model, as it enables the user to examine whether user preferences are correctly reflected by the models. Highlighting the comprehensibility of the models created by DTL methods, this study makes use of a DTL algorithm for constructing an adaptive route selection model.

The rest of this paper is organised as follows; in the next section, methodological approaches for constructing an adaptive route selection model are discussed, starting with a brief description of the system architecture of personalised route guidance and accounting for the adaptation process and the identification of route attributes. The C4.5 algorithm, a DTL algorithm used in this study, is also described in this section. Following that, the third section presents the experiments carried out in order to investigate the learning ability of the adaptive route selection model developed here. The results of the experiments are compared with a multinomial logit model, which is widely used in travel choice behaviour studies. In the last section, the conclusions of the study are summarised, and areas of future work are identified.

2. Adaptive route selection models

Fundamentally, an adaptive route choice model is to be applied to personalised route guidance. Thus, a brief description of the system architecture of personalised navigation would provide a useful background to understand the framework of adaptive route choice models. Figure 1 shows the system architecture of a personalised navigation scheme, named *adaptive autonomous navigation* (Park et al., 2006).

The scheme is composed of five components on the vehicle's side: transport network data, route generation, route selection, the user interface and the learning model. Here, relevant information, including real-time traffic information and other information such as weather, is provided through the Traffic Message Channel (TMC), which is part of the Radio Data System (RDS), which broadcasts digital data on FM channels. Once an origin and a destination are input through the user interface, a set of multiple routes is generated between them. The reason for computing multiple routes is to enable the collection of user feedback by observing the driver's route choice, while at the same time minimising efforts from the user's side as much as possible. In route selection, the best route fitting the driver's preferences is selected and knowledge about the user preferences is gained from past journey records through the learning model. The best route and its alternatives are displayed through the user interface, whose purpose is not only to provide information, but also to collect user feedback. User feedback is defined as positive if the driver accepts the suggestion and negative otherwise.

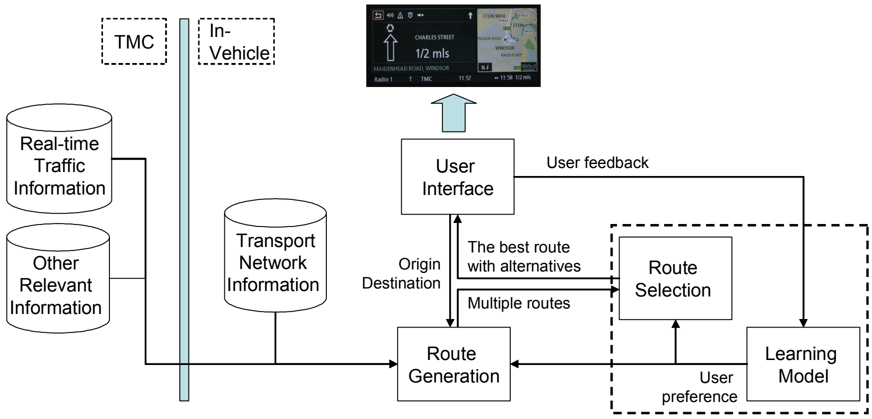


Figure 1 System architecture of personalised route guidance.

An adaptive route choice model corresponds to a combination of a route selection process and a learning model in accordance with the system architecture. Taking the system architecture into account, the elements for constructing an adaptive route choice model are discussed next.

2.1. The adaptive process

With respect to developing an adaptive system, there are three important elements: task (objective), experience and performance measures (Mitchell, 1997). The task of an adaptive route choice model is to select, among route candidates, the route that best fits user preferences, using the C4.5 algorithm. On the other hand, experience is the main means for the C4.5 algorithm to gain knowledge on the user's preferences on route choice and should therefore be continuously updated and stored, while observing route choices made by the user. The experience consists of the "training examples", which comprise the attributes of multiple paths between an origin-destination (OD) pair and the revealed preferences (i.e. choices) of the driver. As a performance measure for adaptive route choice, the ability to predict routes corresponding to actual choices of the user may be a sensible indicator. The performance measure is necessary in order to validate the predictive ability of the model devised. Taking into account the three elements of a route choice model, the adaptive process is shown in Figure 2.

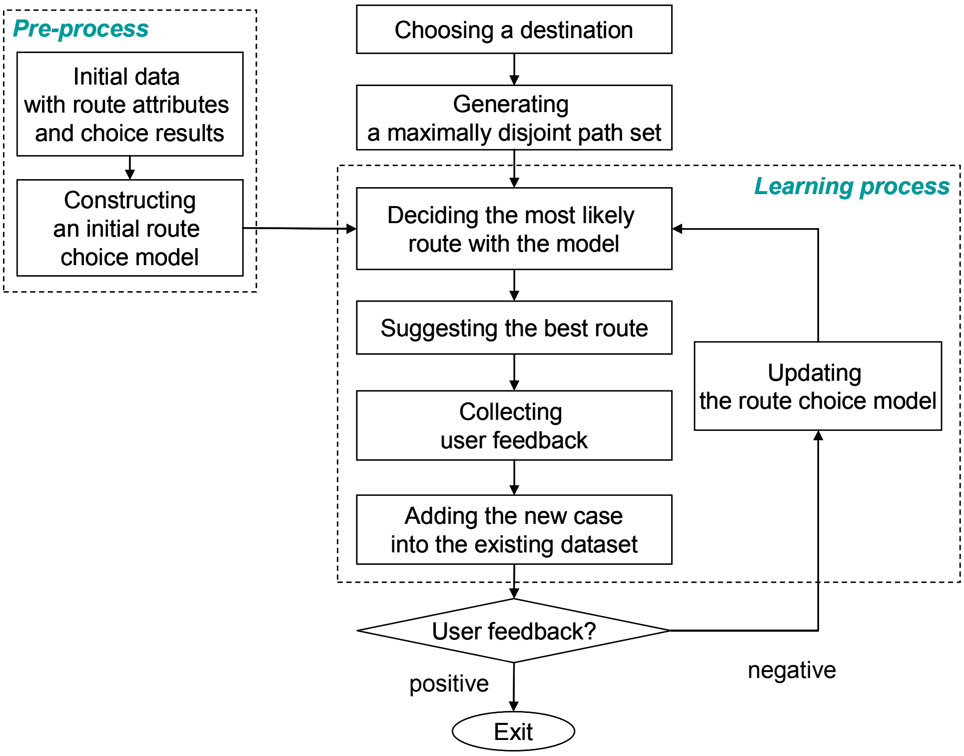


Figure 2 Adaptation process

An initial model is required to classify a route set at the start and several methods for its construction can be formulated. It may be possible to initialise a decision tree with certain values, or construct an initial decision tree with the route choice data observed in pilot surveys. Alternatively, the process of suggesting the best route only starts when sufficient information has been collected. Among these methods, the latter is chosen for this research study, as it may be more sensible since the initial decision tree is being personalised to the driver's preferences. The minimum number of training examples required for the initial decision tree depends on the number of attributes, as decision tree algorithms cannot be used until the number of training examples is larger than the number of attributes.

Once the initial model is built, the best route is selected and displayed to the user. Whenever user feedback is collected from an actual choice, data on the route set, including not only the best route, but also other candidate routes, are stored in the database. Then, if the user chooses a route different from the best route as suggested by the model, an update process of the route choice model is activated, using the data collected at the time.

2.2. Route attributes

During the past decades, considerable effort has been made to determine the important attributes relating to route choice, enabling attribute selection using information collected from literature. Thus, the attributes affecting route choice behaviour are identified, based on results from previous studies

To identify route attributes, three rules are considered; firstly, in order to avoid missing out on potentially important factors, the comprehensive list of route choice attributes suggested by Bovy et al. (1990), as well as additional attributes suggested by other studies regarding route choice behaviour under traffic information provision, are included. On the other hand, availability of data sources should be taken into account. For instance, attributes that require the user to provide information at any time he/she uses the system, may be unsuitable for an automation process. Lastly, the relative importance of attributes is also an important consideration. A literature review on attributes primarily used in route choice behaviour studies may be a sensible approach. It is noteworthy that the relative importance of attributes differs according to the definition of the attributes and situations. Consequently, it seems to be more suitable to categorise attributes as primary and secondary, as in Pang et al. (1995). As a result, the primary attributes of the route choice model developed in this study are shown in Table 1.

Table 1 Attributes of the adaptive route guidance

Attributes	Data sources
Travel distance, Directness, No. of turns, Type of road, Familiarity, No. of stops, Road surface	Transport networks
Travel time, Travel time reliability, Congestion, Traffic density in traffic flow, Waiting time, Safety & probability of accident	Traffic information
Travel cost, Aesthetics, Driving comfort, Commercial/Business area along route	Geographic information
Time of day , Time pressure*	Systems

* Time of pressure can be either provided by the user or derived from destinations, time of day, etc.

It should be noted that the attributes listed in Table 1 are identified, assuming an ideal situation in terms of data sources. Thus, some attributes may not be available in some cases. For example, the simulation experiment in this study deals with seven attributes that are available in a digital map format of ICNavS, which is a software tool for implementing a reliable route guidance algorithm (Kaparias et al., 2006; Kaparias et al., 2007). The details of the route attributes derived from the attributes in the digital map of ICNavS will be described in the discussion of the simulation experiments.

2.3. The C4.5 algorithm for modelling route selection

As mentioned above, a DTL algorithm is utilised for discovering regularities in the route selection mechanism. Algorithms widely used for producing a decision tree are the C4.5 algorithm (Quinlan, 1993), the CHAID algorithm (Kass, 1980) and the CART algorithm

(Breiman et al., 1984). The C4.5 algorithm constructs a decision tree with respect to information entropy, which refers to the amount of information required for describing a data set, while the CHAID algorithm, where the acronym CHAID stands for *Chi*-squared Automatic Interaction Detector, uses the *Chi*-square test to determine the best attributes in order to split a data set. While these two algorithms deal with discrete values of the target classes, the CART, an acronym for Classification and Regression Tree, can deal with continuous values of the target classes, presenting the means and standard deviations of the target classes with respect to the subsets.

Among those algorithms, the C4.5 algorithm is used in this study due to its straightforward mechanism of constructing a decision tree and classifying data and its fairly simple implementation in an adaptive process. Initially, decision trees constructed by the C4.5 algorithm classify data by sorting down from the root to the leaf nodes, so that data in the same segments becomes as homogenous as possible. A path from the root to a leaf node corresponds to an "if-then" rule and each node in the tree specifies a homogeneity test, which is the main objective of the method.

The C4.5 algorithm, developed in the early nineties following the ID3 algorithm, its predecessor, is among the most widely used DTL algorithms. The following description is distilled from Quinlan (1993) and Mitchell (1997). The C4.5 algorithm constructs a decision tree through a recursive process where computing split criteria and selecting an attribute and its threshold values are carried out until the data are perfectly classified or all attributes are tested. Table 2 extended from the table in Mitchell (1997:56) shows the recursive process of building a decision tree

Table 2 The process of building a decision tree in the C4.5 algorithm

BuildTree(*Examples, Classes, Attributes*)
Where, Examples : the training examples.
Classes : class membership. Here, it is assumed to be Boolean (i.e. + or –).
Attributes : a set of attributes that consist of the training examples

- Create a *Root* node for the tree
- If all *Examples* are positive, *Root* ← a single node with label +
- If all *Examples* are negative, *Root* ← a single node with label –
- Otherwise, begin
    - With all attributes from *Attributes and their values*
        - Compute the split criterion, *Gain_ratio*
        - *A* ← the attribute which has the largest value of *Gain_ratio* with the threshold value, v_i
    - The decision attribute for Root ← A
    - For each subset of Examples with v_i of A
        - Add new tree branches below *Root*, corresponding to the condition $A \geqq v_i$, or $A < v_i$.
        - Let *Examples*_v_i be the subset of *Examples* classified with the threshold value, v_i for A
        - If *Examples*_v_i is empty,
            - Below this new branch add a leaf node with label = most common value of *Classes* in *Examples*
        - Else, add the sub-tree below this new branch
            - Create *Childnode* ← BuildTree (*Examples, Classes, Attributes*) for the sub-tree
            - Add *Childnode* to *Root*
- End
- Return *Root*

The split criterion, *Gain ratio* is the outcome of other statistical measures; *entropy, information gain, and split information*. Entropy, the most fundamental measure among them, is devised by referring to information theory that provides a method to measure the amount of information required for describing the purity of a data set (Mitchell, 1997). The formulae of the measures specified for modelling route choice are as follows:

$$Entropy(S) = \sum_{i=1}^{c} - p_i \log_2 p_i \qquad (1)$$

Where, p_i : the proportion of *S* belonging to class *i*

c : the number of target classes

This measure can be defined as "the (im)purity of the example" (Mitchell, 1997), which means how much disorder the (sub-)set of data possesses in terms of class membership. The higher the entropy of the examples, the more information is required in order to completely describe them. Thus the aim of the algorithm is to determine attributes by which a dataset is partitioned into subsets which have the least entropy and are therefore as pure as possible. The crucial measure to determine an attribute that causes the entropy to decrease most effectively is its information gain, which is the reduction in the entropy caused by partitioning the dataset according to the attribute.

$$Information\ gain(S,A) = Entropy(S) - \sum_{v \in Value(A)} \frac{|S_v|}{|S|} Entropy(S_v) \qquad (2)$$

Where, S : a set of past journey records composed of route attributes and choice results

$|S|$: the size of the data set

$Value(A)$: the set of all possible values of route attributes A

S_v : the subset of S for which route attribute A has value v

There is, however, a natural bias in the information gain that favours attributes with a large number of categories, each containing just a few cases. Those attributes tend to offer very high information gains and are likely to be selected as the decision attributes. However, subsets formed by those attributes may be useless in terms of prediction. To avoid this bias, the *gain ratio*, a normalised information gain measure, is devised, by dividing the *gain* by the *split information* that represents the potential information generated by dividing S into c subsets.

$$Split\ Infomation(S,A) = -\sum_{i=1}^{c} \frac{|S_i|}{|S|} \log_2 \frac{|S_i|}{|S|} \qquad (3)$$

$$Gain\ ratio(S,A) = \frac{Gain(S,A)}{Split\ Information(S,A)} \qquad (4)$$

The recursive partitioning method constructs decision trees until each subset contains a single class or until there is no data left and the result is often a very complex tree that overfits the data, which implies that the tree has unnecessary branches. In response to the overfitting problem, the C4.5 algorithm removes those parts of the tree that do not contribute to classification accuracy, by using a pruning process called "error-based pruning".

Error-based pruning calculates the probability of error rates on any leaf with a given confidence level. If the probability of error rates on leaf A is lower than the probability of error rates on leaf B, which is below leaf A, then the branches from leaf B to the end leaves are pruned. In the C4.5 algorithm, the error rate is predicted using the confidence limits for the binomial distribution with a confidence level of 25%. The error rate of a leaf with a confidence level can be calculated as

$$Error\ rate = \sum_{i=1}^{N} K_i \times U_{CF}(E, K_i) \tag{5}$$

where, K_i : the number of training cases covered by a leaf in class i

E : the number of associated errors on the training set

$U_{CF}(\bullet)$: the upper confidence limit for the binomial distribution

3. Simulation experiments

3.1. Experiment procedure

Aimed to investigate the learning ability of the model developed in this study, experiments with data generated by fictitious route selection rules are carried out. The base data for the experiments consists of sets of paths, one for each OD pair, which are *maximally disjoint*, i.e. sharing as few links as possible with each other, in order to reduce the risk of one link failing affecting all the paths containing it (Kaparias et al., 2006) and to avoid suggesting routes that are virtually similar to each other. The sets of maximally disjoint paths are generated by ICNavS, whose multiple-routing strategy involves penalising unreliable links and links already included in a path by increasing their weights, so as to exclude them from the route search. The paths computed must satisfy the following four constraints: maximum path duration, maximum path reliability, maximum path overlapping ratio and maximum number of paths (Kaparias et al., 2006; Kaparias et al., 2007) .

A total of 2,042 paths from 675 route sets are generated, using the road network of the South Kensington area in West London, which consists of 384 nodes and 956 links and for which on average a route set has three alternative paths. In the simulation with the adaptive route choice models, 250 route sets are randomly selected and sequentially used in the adaptive process. The process of predicting route choice according to route attributes and the route choice model built so far is repeated 250 times and the model is updated when the actual choice differs from the predicted one.

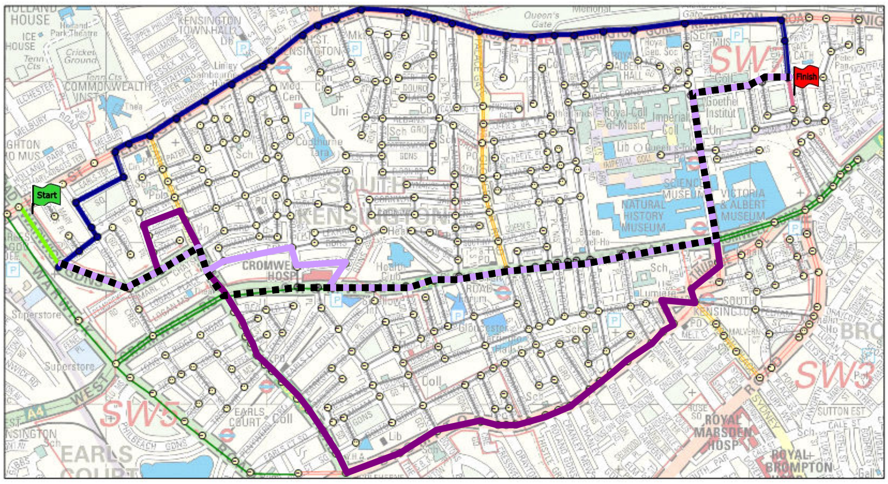


Figure 3 The study network and an example of maximally disjoint paths.

The main objective of the experiments is to investigate how the structure of a decision tree evolves and how long it takes for the route choice model to adapt itself to the user's preferences while collection information on route choice decisions. The route choice preferences for these experiments are predefined as a single objective which changes twice, after the 50th route set and after the 150th route set in the interest of inspecting the adaptive process. In this experiment, the following route selection rules under three different scenarios are used:

- Test 1 : the shortest path → the least averse path → the most reliable path
- Test 2 : the fastest path → the least complex path → the shortest path
- Test 3 : the most reliable path → the most direct path → the fastest path

Although this approach is effective to investigate adaptability of the model developed here, it leads to the problem of using information irrelevant to current route choice decision. For example, if the training examples used in the second phase of Test 1 contain a great deal of past journey records of the first phase where the user is in favour of the shortest path, this may cause bias in modelling route choice behaviour of the second phase and consequently result in low predictive accuracy of the model. A sensible method to deal with this problem is to select a subset of the training examples called a *window* with respect to the sequence of the data, discarding the training examples previously used, but not relevant any longer. In this study, the maximum sizes of the training examples is are 30 sets of routes with the DTL model and 100 sets pf routes with the MNL model.

3.2. Normalisation of route attributes

The digital map of ICNavS provides five attributes for each link: length, road type, speed, turning angles between connected links and travel time reliability. Based on the link attributes, seven attributes of each route are computed: travel distance, travel time, directness, number of turns, travel time reliability, types of road and familiarity. The identification of each route attribute is explained in details in Park et al.(2006)

The values of each attribute should be normalised by converted into relative units that range from a value of 0 to a value of 1 with respect to the best route of a path set, in order to enable the user to recognise differences between alternative paths efficiently. Regarding the normalisation of attribute values, it should be noted that each attribute has different concepts on the “best” route. For example, the best route from the point of view of travel distance or travel time is a route with the minimum value of each attribute, while the route with the maximum value of the reliability attribute is the best route with respect to reliability. Thus, the value close to a value of 1 is the best value for the reliability measure, while the value close to a value of 0 is the best vale in terms of travel time. To avoid confusing the user with the normalised value of relative units, it is necessary to set a consistent range of each measurement. A value of 1 is set as the best value, while smaller values correspond to worse routes. More details of this process are described in Park et al.(2007)

3.3. Multinomial logit model as a comparable method

In order to analyse the advantages and disadvantages of the route choice model using the C4.5 algorithm more precisely, its performance is compared with the results from a multinomial logit (MNL) model. The estimated probability that individual n chooses route i among a set of multiple routes, C_n, is expressed as

$$P_n(i) = \frac{e^{\boldsymbol{\beta}'_n \mathbf{x}_{in}}}{\sum_{j=1}^{c_n} e^{\boldsymbol{\beta}'_n \mathbf{x}_{jn}}} \qquad (6)$$

Here, the utility function is expressed in a linear-in-parameters form, comprising a set of coefficient estimates, $\boldsymbol{\beta}'_n$, which is personalised for individual n and for a set of route attributes $\mathbf{x}_n$. The coefficients of the MNL model are estimated by a maximum likelihood estimation procedure using the Newton-Raphson algorithm. It should be noted that for the purpose keeping the analysis simple, this study assumes that each choice observation is independent, such that repeated choice data can be used in the same way as purely cross-sectional data.

4. Analysis of comparison results

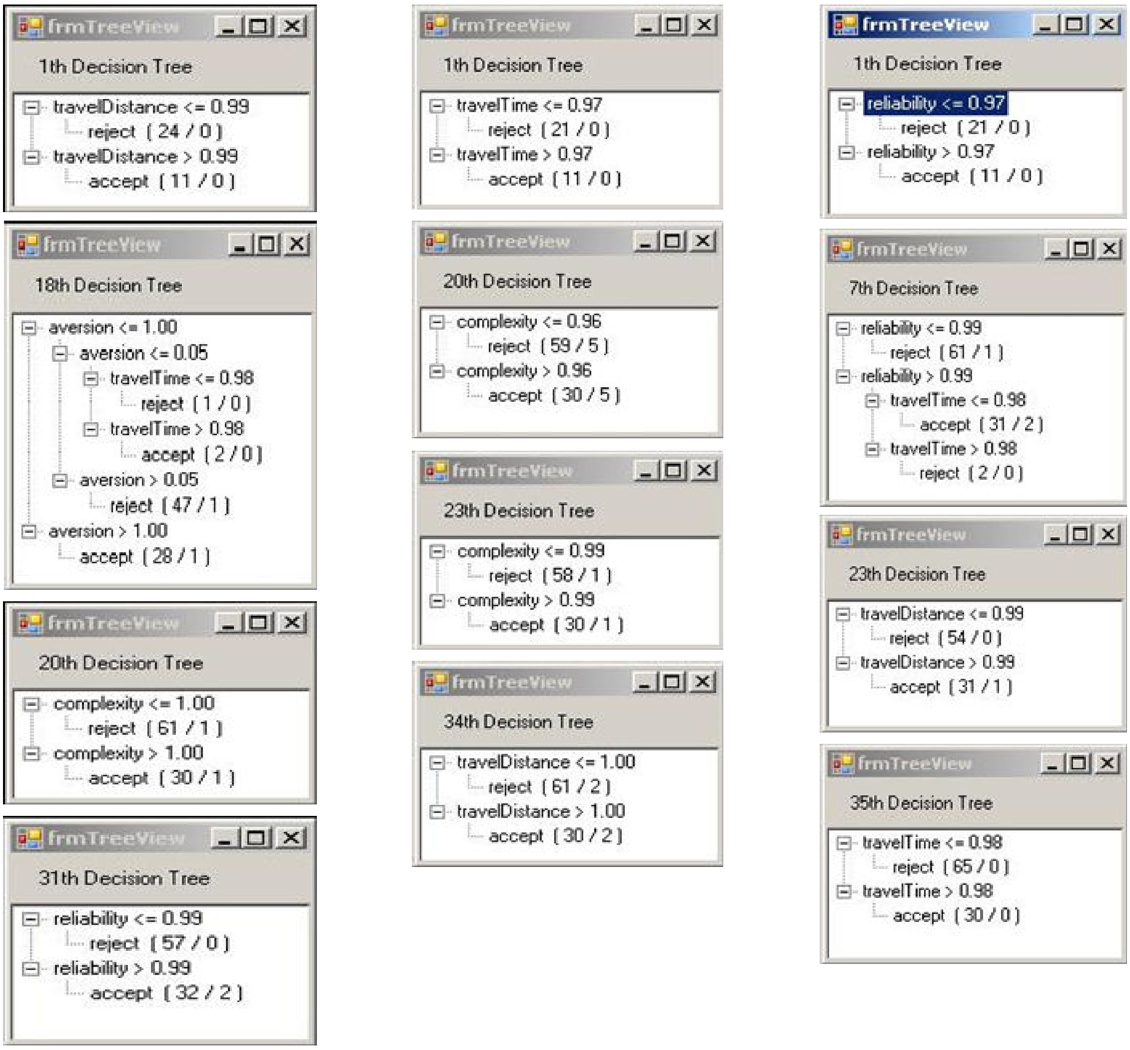
The results of the simulation experiments carried out using the two route choice models are summarised in Table 3. While simulating the adaptive process 250 times, the DT model was updated 31 times and the MNL model was updated 63 times. The less updates during the same period means that the model is capable of predicting route choice correctly so that it is unnecessary to update the model. In that sense, it appears that the DT model outperforms the MNL model in terms of predictability; the predictive accuracy of each update in all tests will be discussed, along with the results of the percentage of correct predictions of each test.

To observe how well the models fit the sets of observations, error rates over the updates of the models are used as an informal goodness-of-fit measure. An error rate is calculated by dividing the number of incorrect predictions by the total number of cases and it is found that the DT model has a lower error rate than the MNL model on average, their values being 5.3% and 17.7% respectively. The cause of the higher error rate of the MNL model may lie in the assumption of compensatory relationship between attributes from the linear-in-parameter utility functions or in the lack of ability to accommodate nonlinearity between attributes.

Table 3 A summary of the adaptive route choice modelling

	Test 1	Test 2	Test 3	Average
Average number of updates				
Decision tree	29	36	31	31
MNL	56	81	52	63
Average no. of sets per update				
Decision tree	9	7	8	8
MNL	5	3	5	4
Average error rate				
Decision tree	4.1%	7.1%	4.6%	5.3%
MNL	16.3%	19.5%	17.2%	17.7%

The following figures show examples of the evolutionary changes of the decision trees and the coefficient estimates of the MNL models over the updates. As can be seen in the figure, the adaptive process using the DT model can be recognised fairly easily, enabling one to inspect if the model successfully accommodates the changes of the predefined preferences. On the other hand, understanding the significance of changes of the coefficient estimates of the MNL model is more complex.



(a) Test 1 (b) Test 2 (c) Test 3

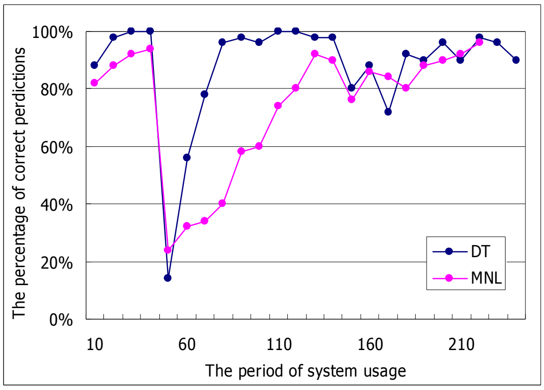
Figure 4 The decision trees of each test.

Parameters of MNL model

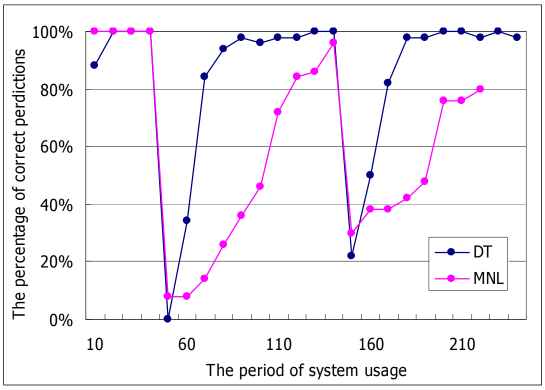
No.	Trave...	t-va...	Reliab...	t-v...	Direct...	t-va...	Famili...	t-val...	Aversi...	t-val...	Compl...	t-value	Null-LL	Final-LL	Rho-sq...
27	1.30	0.68	5.10	1.59	4.58	0.99	-0.99	-0.96	0.80	0.66	1.19	0.65	-1020.08	-859.15	0.16
28	1.17	0.62	5.26	1.65	5.24	1.13	-0.91	-0.88	0.80	0.66	0.97	0.53	-1010.79	-911.88	0.10
29	1.22	0.65	4.47	1.43	5.31	1.14	-1.00	-0.96	0.70	0.58	1.32	0.72	-1065.72	-890.09	0.16
30	0.10	0.06	3.69	1.21	7.68	1.70	-0.77	-0.75	0.23	0.20	1.43	0.79	-1046.71	-957.35	0.09
31	0.10	0.06	3.28	1.11	8.03	1.76	-0.70	-0.70	0.34	0.30	1.34	0.75	-1116.74	-971.86	0.13
32	0.34	0.19	3.63	1.23	8.23	1.80	-0.72	-0.72	0.29	0.26	1.44	0.81	-1135.64	-1050.29	0.08
33	-0.13	-0.08	3.21	1.12	9.50	2.12	-0.60	-0.60	0.04	0.04	1.64	0.96	-1216.85	-1109.94	0.09
34	-0.27	-0.16	2.63	0.93	9.98	2.23	-0.56	-0.56	0.01	0.01	1.70	0.98	-1279.80	-1129.71	0.12
35	0.15	0.09	3.11	1.15	9.58	2.15	-0.45	-0.46	-0.05	-0.05	1.83	1.08	-1308.23	-1254.94	0.04
36	0.09	0.06	2.87	1.07	10.32	2.30	-0.36	-0.38	-0.02	-0.01	1.78	1.05	-1438.31	-1341.12	0.07
37	0.10	0.06	3.00	1.14	10.76	2.40	-0.25	-0.26	0.21	0.21	1.27	0.78	-1528.89	-1404.70	0.08
38	0.17	0.11	2.84	1.10	10.89	2.43	-0.29	-0.30	0.19	0.19	1.32	0.82	-1596.86	-1438.80	0.10
39	0.20	0.12	2.60	1.02	11.02	2.46	-0.26	-0.27	0.23	0.23	1.34	0.83	-1633.74	-1453.32	0.11
40	0.25	0.16	2.56	1.01	11.27	2.52	-0.26	-0.27	0.25	0.24	1.30	0.82	-1652.06	-733.54	0.56
41	0.60	0.39	2.95	1.24	10.93	2.48	-0.11	-0.12	0.27	0.29	1.25	0.83	-881.42	-754.07	0.14
42	0.47	0.30	2.65	1.12	11.40	2.58	-0.06	-0.07	0.23	0.25	1.29	0.86	-903.05	-760.50	0.16
43	0.57	0.37	2.04	0.88	10.94	2.50	-0.05	-0.06	0.35	0.37	1.32	0.87	-910.58	-726.60	0.20
44	0.57	0.37	1.97	0.85	11.00	2.51	0.00	0.00	0.41	0.44	1.21	0.80	-877.78	-727.99	0.17
45	1.01	0.67	3.01	1.35	10.08	2.36	-0.10	-0.11	0.81	0.90	0.38	0.28	-881.65	-725.61	0.18
46	0.85	0.57	2.60	1.18	10.54	2.49	-0.05	-0.05	0.76	0.85	0.40	0.29	-881.35	-726.12	0.18
47	0.83	0.56	2.42	1.10	10.65	2.52	-0.06	-0.07	0.77	0.86	0.39	0.28	-882.96	-724.00	0.18
48	0.69	0.46	2.50	1.14	11.03	2.60	0.09	0.10	0.70	0.79	0.35	0.25	-882.23	-741.46	0.16
49	0.31	0.23	2.92	1.42	12.74	3.18	0.13	0.16	0.69	0.84	-0.09	-0.07	-906.45	-807.04	0.11
50	0.31	0.24	2.26	1.11	12.49	3.18	0.08	0.09	0.65	0.79	0.07	0.06	-974.51	-774.94	0.20
51	-0.24	-0.19	1.77	0.92	14.88	3.82	0.19	0.22	0.45	0.57	0.01	0.01	-957.04	-876.60	0.08
52	-0.53	-0.45	1.25	0.67	17.49	4.59	0.28	0.34	0.27	0.35	0.11	0.09	-1092.57	-1179.31	-0.08
53	-0.51	-0.43	1.07	0.58	17.31	4.60	0.27	0.33	0.23	0.29	0.20	0.16	-1396.66	-1163.73	0.17

Figure 5 The coefficient estimates of each update in Test 3. (a part of estimation)

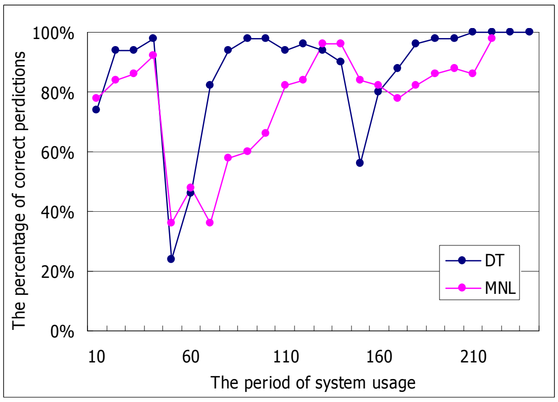
The adaptive processes can be effectively depicted by plotting the percentage of correct precision and the error rates against the updates of each model. In all tests, each model successfully adapts itself to the preferences from the beginning. Once the preferences change, the DT model is capable of discovering the new preferences within the relatively shorter periods of around 30 ~ 40 times, while the MNL model appears to take longer to accommodate the preferences around the 110^{th} time later, showing lower predictive accuracy overall.



(a) Test 1



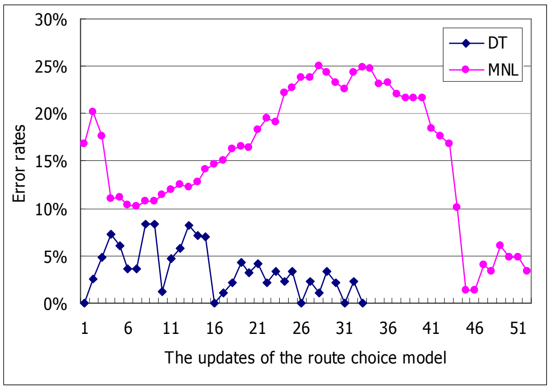
(b) Test 2



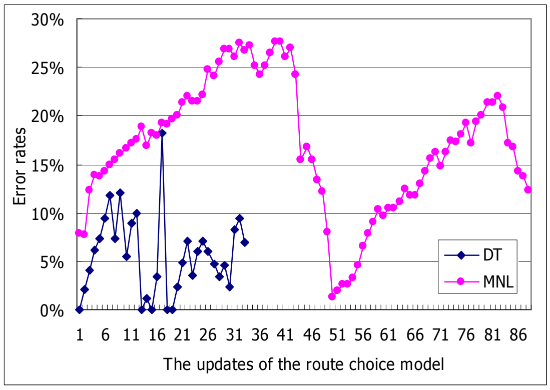
(c) Test 3

Figure 6 The percentage of correct predictions

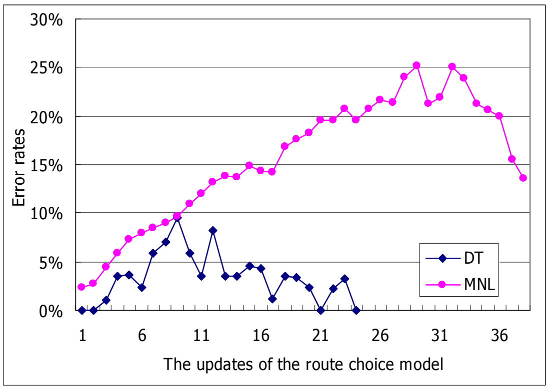
Similar to the significant difference in average error rates of two models in Table 3, the overall error rate of the DT model is lower in most cases and more stable along with the updates, while the error rate of the MNL model is higher on the whole and shows dramatic fluctuation during the adaptive process. Any possible reason of the rapid drops in the error rates of the MNL model is not found at the current stage. However, It appears that the unstable tendency might have a negative impact on predictive accuracy.



(a) Test 1



(b) Test 2



(c) Test 3

Figure 7 The error rates

To sum up, both the DT model and the MNL model appear to be capable of learning user preferences and of adaptively modifying their structures. In this section, advantages and disadvantages of both methods are compared in detail and future work is presented for the purpose of improving the weaknesses.

Firstly, the DT model shows better performance with respect to predictability, which is of importance in practice. Also, the DT model has a comprehendible structure, which is particularly useful when examining and assessing learnt knowledge. In addition, failure of construction of a decision tree in all tests is unlikely, once the size of data becomes larger than the number of attributes. However, the experiments point out some weaknesses of this method. As multiple routes are classified according to the decision tree, in some cases it may be impossible to select a particular route which best fits user preferences. That is, route guidance implementing a DT model may yield multiple recommendations, which may in turn confuse the driver. Thus, it is required to expand the decision tree learning algorithm, so as it becomes capable of producing real-valued output, such as scores. Correlations between attributes may also affect the biased model; although the correlation problem is not found here, some attributes such as travel time and travel distance are strongly related, so one attribute may replace the other one in a decision tree. Consequently, it is necessary to come up with a method to deal with attribute correlations in the model.

As opposed to that, there would not be any possibility of multiple recommendation problems if the MNL model is applied to adaptive route guidance. In the light of behavioural analysis aspects, this approach may be closer to human reasoning. Moreover, once enough data has been collected, the MNL model shows a slightly more stable predictive accuracy, or at least similar to that of the DT model. Apart from these strengths, the MNL model structure is less straightforward compared with the human intelligible structure of the DT model. If interpretability is not necessary, this feature might not be necessary either, but in terms of route choice, a comprehendible model structure may be significant in giving insight into choice mechanisms and therefore confidence in the validity of the model.

Another problem is that coefficient estimates may be insignificant; for example, if the model fails to converge, or if the t-statistics of the coefficient estimates or the ρ^2 are lower than the significance levels, the coefficients cannot be used. This can be caused by either insufficient data or by the use of irrelevant variables. A sensible method to solve the problem caused by the size of the data is to utilise a recursive process, which updates a utility model based on the previously estimated parameters and the new observations. This method is expected to not only reduce the computational effort but to also enable estimation without the need of a panel data of travel history. If coefficient estimates are considered insignificant with respect to confidence measures such as t-statistics or ρ^2, an automation process re-estimating coefficients whilst excluding variables considered as less relevant might be a solution.

5. Conclusion

In the previous sections, a rule-based approach and a utility maximisation approach to modelling route choice have been compared, with a view to implementation in personalised navigation. The C4.5 algorithm and the multinomial logit were employed to construct route choice models. Both approaches make use of the same adaptation process to learn user preferences, which involves continuously updating the route choice model, so as to accommodate knowledge recently learnt. The learning ability of the adaptive route choice models was investigated in experiments, where a single preference was predefined for each test and was subsequently changed during the test. The results showed that it was possible to alter route selection rules in both approaches, when predicted results were not corresponding to actual choices. With respect to predictive accuracy, decision tree models outperform MNL models, and decision tree models appear to possess a strong point in terms of the interpretability of the model structure. The experiments also highlighted several tasks required to improve the performance of adaptive models in practice.

In the meantime, the models will be considerably improved if a number of points are complemented; firstly, attributes that are unavailable due to the lack of information about the transport network and traffic/geographical information should be obtained and incorporated into the model, taking into account levels of information provision. A comprehensive investigation of attributes important to users of navigation systems, obtainable through a process of data mining, would be necessary. The data mining process entails not only quantitative attributes (e.g travel time reliability, travel cost) but also qualitative attributes (e.g. aesthetics, driving comfort, safety). Accordingly, multi-criteria route guidance will be required at that stage. In addition, essential information will be obtained by the conduction of an empirical study where stated preferences of experienced drivers will be surveyed with an improved version of the user interface developed in this study. Lastly, field trials with a vehicle equipped with a navigation system in which the adaptive method described here is incorporated, will be carried out to investigate contributions to user satisfaction and efficiency in use.

Acknowledgement

The authors would like to thank BMW AG for supporting the research work reported here..

References

Arentze, T. and Timmermans, H. (2003), ' Modeling learning and adaptation processes in activity-travel choice: A framework and numerical experiment', Transport, 30, pp. 37-62.

Arentze, T. A. and Timmermans, H. J. P. (2004), 'A learning-based transportation oriented simulation system', Transportation Research Part B: Methodological, 38(7), pp. 613-633.

Arentze, T. A. and Timmersmans, H. (2005) 'Parametric action decision trees: Incorporating continuous attribute variables into rule-based models of activity-travel behavior', Paper presented at the 84th Annual Meeting of the Transportation Research Board, Wachington D.C., January 2005.

Bovy, P. H. L. and Stern, E. (1990) *Elements of route choice behaviour*, Kluwer Academic Publishers, Dordrent, The Netherlands.

Breiman, L., Freidmand, J., Olshen, R. and Stone, C. (1984) *Classification and regression trees*, Wadsworth International Group, CA, USA.

Kaparias, I., Bell, M., Chen, Y. and Bogenberger, K. (2006) ' ICNavS: A tool for reliable route guidance', Paper presented at the 13th World Congress on ITS, London, October, 2007.

Kaparias, I., Bell, M., Chen, Y. and Bogenberger, K. (2006) 'Autonomous reliable dynamic route guidance with re-routing', Transportation Research Part C: Emerging Technologies. (In process)

Kaparias, I., Bell, M., Bogenberger, K. and Chen, Y. (2007) 'An approach to time-dependence and reliability in dynamic route guidance', Paper presented at the 86th Annual Meeting of the Transportation Research Board, Washington, D.C., January, 2007.

Kass, G. V. (1980) 'An exploratory technique for investing large quantities of categorical data', Applied Statistics, 29, pp. 119-127.

Mitchell, T. (1997) *Machine learning*, McGrow-Hill International Editions, Singapore.

Nakayama, S. and Kitamura, R. (2000) 'Route choice model with inductive learning', Transportation Research Records, 1725, pp. 63-70.

Nakayama, S., Kitamura, R. and Fujii, S. (2001) 'Drivers' route choice rules and network behavior', Transportation Research Records, 1752, pp. 62-68.

Pang, G. K. H., Takabashi, K., Yokota, T. and Takenaga, H. (1995), 'Drivers route selection : A philosophical consideration and user-Interface', Paper presented at the 6th Vehicle Navigation and Information Systems Conference, pp. 147-154.

Pang, G. K. H., Takabashi, K., Yokota, T. and Takenaga, H. (1999), 'Adaptive route selection for dynamic route guidance system based on fuzzy-neural approaches', IEEE Transactions on Vehicular Technology, 48(6), pp. 2028-2041.

Park, K., Bell, M., Kaparias, I. and Bogenberger, K. (2006) ' Learning user preferences of route choice behaviour for adaptive route guidance', Paper presented at the 13th World Congress on ITS, London, October, 2007.

Park, K., Bell, M., Kaparias, I. and Bogenberger, K. (2007) ' An adaptive route choice model by a rule-based approach for intelligent route guidance', Paper presented at the 86th Annual Meeting of the Transportation Research Board, Washington, D.C., January, 2007.

Quinlan, R. 1993, *C4.5 : Programs for machine learning*, Morgan Kaufmann Publishers, Inc., CA, USA.

Rogers, S., Fiechter, C. and Langley, P. (1999) 'An adaptive interactive agent for route advice', Paper presented at the 3rd International Conference on Autonomous Agents, ACM Press, Seattle, pp. 198-205.

Schofer, J. L., Koppleman, F. S. and Charlton, W. A. (1997) 'Perspectives on driver preferences for dynamic route guidance systems', Transportation Research Records, 1588, pp. 26-31.

Timmermans, H., Arentze, T. and Ettema D. (2003) 'Learning and adaptation behavior : empirical evidence and modelling issues', Paper presented at the Euro-Workshop on Behavior Responses to ITS.