Hierarchical En-Route Planning under the Extended Belief-Desire-Intention (E-BDI) Framework

Sojung Kim, Young-Jun Son, Ye Tian, Yi-Chang Chiu
The University of Arizona
Tucson, AZ, USA

C. Y. David Yang
Federal Highway Administration
McLean, VA, USA

Abstract

En-route planning is a dynamic planning process to find the optimal route (e.g., shortest route) while driving. The goal of this paper is to mimic a realistic drivers’ en-route planning behavior under the situations with incomplete information about road conditions using the Extended Belief-Desire-Intention (E-BDI) framework. The proposed E-BDI based en-route planning is able to find a new route to the destination based on the predicted road conditions inferred by drivers’ own psychological reasoning. A main challenge of such a detailed E-BDI model is a high computational demand needed to execute a large scale road network, which is typical in a big city. To mitigate such a high computational demand, a hierarchical route planning approach is also proposed in this work. The proposed approach has been implemented in Java-based E-BDI modules and DynusT® traffic simulation software, where a real traffic data of Phoenix, Arizona is used. To validate the proposed hierarchical approach, the performance of the en-route planning modules under the different aggregation levels is compared in terms of their computational efficiency and modeling accuracy. The validation results reveal that the proposed en-route planning approach efficiently generates a realistic route plan with individual driver’s prediction of the road conditions.

Keywords
Agent-based simulation, En-route planning, Belief-desire-intention, Hierarchical route planning.

1. Introduction

Route planning by searching the optimal route on a road network given pairs of the origin and destination, is one of the major problems in the transportation and supply chain management applications [1]. To achieve accurate prediction and analysis of the traffic system, drivers’ route planning behaviors have been extensively studied considering individual driver’s own preferences (e.g., road safety, traffic volume, willingness to pay) [2]. Recently, Kim et al. [3] has proposed a route planning approach under the Extended Belief-Desire-Intention (E-BDI) framework [4], which has allowed us to mimic the realistic drivers’ en-route planning behavior with various preferences. The en-route planning approach (theme of this paper) based on such an E-BDI framework can generate a dynamic route plan under a more realistic driving environment (e.g., drivers’ learning and interactions while driving) [3].

One of the major difficulties of the E-BDI based en-route planning, however, is high computational requirements, especially for a large scale road network, which is typical in a big city. In order to infer the conditions of each road (or link), the E-BDI based en-route planning needs to execute its underlying algorithms multiple times, such as Bayesian Belief Network (BBN) which is a probabilistic model to handle uncertain perception and reasoning processes of a human under dynamic environments (e.g., perception of travel time on a road network) [5]. One way to resolve such a computation issue is to adopt a hierarchical route planning approach, which is known as efficient for speeding up the route planning [1]. The hierarchical route planning is an aggregation approach by dividing an
original road network into the hierarchical structure. The goal of this paper is to propose an E-BDI based hierarchical en-route planning approach for large scale road networks.

According to Mainali et al. [6], there are two categories of approaches in the hierarchical structure of the road network: (1) the property based aggregation, which generates the hierarchical structure based on the properties of the road network (e.g., road section categories and the number of lanes), and (2) the location based aggregation, which partitions a road network into several subnetworks and aggregates the subnetworks. Since drivers’ route plans in real-life situations often involve various road section categories together (e.g., highway with interstate roads), this study uses the location based aggregation.

The rest of the paper is organized as follows. The background of the E-BDI framework [4] is briefly described. The E-BDI based hierarchical en-route planning is then proposed. The simulation scenario based on the Phoenix road network is discussed, and the experiments and results are then provided. Lastly, conclusions and future works are presented.

2. E-BDI Framework

The E-BDI framework is a generic framework for representing the decision-making and decision-planning processes with psychological natures of the human [4]. Figure 1 shows the four major modules of the E-BDI framework: Belief module, Desire module, Decision making module, and Emotional module. The Belief module represents a perceptual process of a human and generates beliefs which are perceived information from the environment (e.g. travel time and road risk for drivers). Then, the Desire module determines desires, such as goals or intended outcomes, based on the beliefs. If a human agent wishes to achieve a certain desire among alternatives, the selected desire becomes the intention [7]. To achieve the intention, the Decision making module creates a multi-stage plan (e.g., route plan of a driver) via Real-time Planner, and then executes the multi-stage plan in the physical environment through the Decision Executor. The E-BDI framework adopts algorithms and techniques to realize each submodule: (1) Bayesian Belief Network (BBN) as the Perceptual Processor in the Belief module which infers the states of attributes (i.e., Beliefs) from environmental information, (2) Extended Decision Field Theory (EDFT) and Probabilistic Depth First Search (PDFS) algorithms as the Real-time Planner in Decision Making module. EDFT computes the choice probability of each option (e.g., a path), and then PDFS generates a multi-stage plan (e.g., a route plan) based on the choice probabilities of options. As a result, the E-BDI framework is capable of mimicking real drivers’ route planning behaviors based on perceived road conditions involving psychological natures of drivers. Especially, unlike the constant time based shortest path finding algorithms (e.g., Dijkstra’s algorithm, a widely used algorithm to find the shortest path with a low computational complexity by eliminating infeasible paths [9]), the E-BDI framework allows driver agents to generate various route plans with choice probabilities of paths via BBN, EDFT, and PDFS. More detailed information about the E-BDI framework is illustrated in [4].

Figure 1: Components of the E-BDI framework [4, 8]
3. E-BDI based Hierarchical En-route Planning

The proposed hierarchical en-route planning approach consists of two levels: (1) a high-level (aggregated) planning with the E-BDI framework, and (2) a low-level planning with the shortest path finding algorithm (e.g., Dijkstra’s algorithm). The basic concept of this approach is to reduce the computational demand of the E-BDI framework by considering an aggregated, high-level network. According to Sedgewick [9], the run time of Depth First Search (DFS) algorithm is linearly increased with the number of vertices (V) and edges (E) in graph (i.e., O(|V|+|E|)). This means that the computation time of PDFS algorithm in the E-BDI framework can be significantly reduced with fewer nodes and paths in the high-level network given by the proposed hierarchical approach. Furthermore, the location based aggregation is applied to generate aggregated nodes from the low-level road network topology.

In the proposed hierarchical en-route planning algorithm, the first step is to generate a high-level network from the original road network. To this end, the road network is partitioned into several zones, where each zone has several borders with outer nodes connecting to adjacent zones. Once the zones and borders are defined, the shortest paths between borders are determined using the shortest path algorithm. In this paper, the Dijkstra’s algorithm is adopted to find the shortest path between the borders. However, any other shortest path algorithms such as A* could also be considered. Once the high-level network is established via finding shortest paths between borders, the E-BDI based en-route planning is performed. The E-BDI based en-route planning is conducted in a probabilistic manner based on the origin and destination of a driver agent with the processed data. Next, the high-level route plan is merged with the related shortest paths at the low level. As a result, the existing route plan of a driver agent is substituted by the new route plan. Figure 2 shows pseudo codes of the proposed E-BDI based hierarchical en-route planning approach.

1: LOAD an original road network topology information
2: GENERATE borders as aggregate nodes from the original road network
3: FIND the shortest path between all pairs of the borders to generate aggregate paths
4: IF the E-BDI based en-route planning is triggered THEN
5: COMPUTE the attributes’ values of all pairs of the aggregate paths
6: LOAD the origin and destination of an individual driver
7: RUN the E-BDI based en-route planning using the attributes’ values
8: COMBINE the aggregate route plan with the shortest path sets of the aggregate paths
9: RETURN a route plan

Figure 2: Pseudo codes of the E-BDI based hierarchical en-route planning

3.1 Partitioning for a Hierarchical Network

A partitioning method has been employed to generate a high-level network. The method consists of two processes: (1) decomposition process, and (2) aggregation process. The decomposition process divides an original road network into m zones (or subnetworks) via the Breadth First Search (BFS) algorithm, which is a searching algorithm to find the shortest path by inspecting all possible paths. Although BFS algorithm is computationally expensive, the algorithm guarantees to visit all possible nodes until reach a destination node [9]. Thus, all nodes in a road network can be contained by one of zones. In this stage, m BFS algorithms are executed in parallel to define the m zones. For the BFS algorithms, K-means clustering method [10] is employed to determine a root node of each BFS algorithm. Let \( x_i \) be a node in the entire road network with two dimensional real vector (i.e., x coordinate and y coordinate) and C be a set of clusters \( C = \{ C_1, C_2, \ldots, C_m \} \) (1 ≤ i ≤ m) where \( C_i \) is the ith cluster. \( \mu_i \) is the mean of points in \( C_i \) (or a centroid in \( C_i \)). Equation (1) reveals the K-means clustering for the partitioning:

\[
\arg \min_c \left\{ \sum_{i=1}^{m} \sum_{x \in C_i} \| x - \mu \|^2 \right\}
\]

where all \( x_i \)'s are connected in cluster \( C_i \). Then, in each cluster \( C_i \), the closest node from \( \mu_i \) is selected as a root node for each BFS algorithm by the Equation (2):

\[
\arg \min_{x \in C_i} \left\{ \| x - \mu \|^2 \right\}
\]

As a result, since each root node is likely to be located at the center of a cluster, all nodes in the road network can be evenly distributed in the zones. Once the m root nodes in the road network are selected, BFS algorithms are executed with each of the selected root nodes. From each selected root node \( j \), the BFS selects adjacent nodes of \( j \) from the original road network if \( k \) is not included in any other BFS trees. If \( k \) is already included in other BFS
trees, \( j \) is marked as the border node. This process is repeated until all nodes are selected. Finally, \( m \) zones are created from the original network [6].

In the aggregation process, the hierarchical mapping algorithm [11] is applied to the \( m \) zones given by the decomposition process. The high-level network consists of borders and paths between the borders. Each path between borders is the shortest path given by the Dijkstra’s algorithm. Consequently, a high-level network can be attained. Figure 3 depicts an overview of the partitioning procedure.

![Figure 3: Bayesian-Belief Network of commuter and explorer](image)

In the original road network (see Figure 3 (a)), a root node for each cluster is selected via \( k \)-means clustering (see Figure 3 (b)). Then, two different BFS algorithms are conducted at the same time and generate two zones in Figure 3 (c). Once zones are defined, the nodes connected with other zones are selected as borders (red nodes in Figure 3 (d)). By applying the Dijkstra’s algorithm to the borders, the original network is aggregated to a high-level network as shown in Figure 3 (e).

### 3.2 E-BDI based En-route Planning for High-Level Road Network

The E-BDI based en-route planning has been employed to provide a new route plan in the aggregate road network. Figure 4 shows pseudo codes of the route planning procedure which includes the three major algorithms implemented in the E-BDI framework.

```
1: CALL PDFS to generate a route plan
2: SET a current intersection \( v \) as a staring node of ROUTE PLAN \( S \)
3: SET a planning horizon \( q \) as a zero
4: REPEAT
5: SET \( t \) is the latest intersection of \( S \) and ADD one to \( q \)
6: IF \( t \) is a destination or \( q \) is the same as the given number of planning horizons THEN return \( S \)
7: CALL BBN and EDFT to calculate choice probability for all connected PATHs of \( t \)
8: FOR the all connected PATHs
9: IF a PATH has been selected THEN set a choice probability as zero CONTINUE with the next PATH
10: SELECT a PATH based on the probability distribution given by BBN and EDFT
11: SET \( w \) as an adjacent intersection
12: IF \( w \) is not discovered and not explored
13: SET a PATH as tree edge and \( w \) as discovered
14: ADD \( w \) to \( S \) as a latest intersection and CONTINUE at line 5
15: ELSEIF \( w \) is discovered
16: SET a PATH as back edge
17: ELSE
18: SET a PATH as forward edge
19: SET \( t \) as explored
20: DELETE \( t \) from \( S \)
21: UNTIL \( S \) is not empty
```

Figure 4: E-BDI based en-route planning algorithm [3, 4]
Figure 5 shows an exemplary BBN structure of a driver agent. In this study, the travel time and road risk are considered as attributes. The two attributes and risk weight are influenced by drivers’ perceived information about each path in a road network such as distance to destination, free flow travel time, traffic flow, and accident frequency.

![Bayesian Belief Network (BBN) of a driver agent](image)

In the E-BDI based en-route planning algorithm, the states of the two considered attributes and the risk weight are first inferred from the BBN via probabilistic inference process. If the inferred state of the road risk (or travel time) is high, then 2.0 was assigned as the attribute value. Otherwise, the value of road risk (or travel time) is set as 1.0. Similarly, the risk weight is provided in the form of a two dimensional weight vector based on the state of the risk weight (the sum of each weight vector is 1). If the inferred state of the risk weight is high, the road risk has a higher weight than that of the travel time (e.g., a weight vector for risk-averse drivers is 0.7 road risk and 0.3 for travel time). Otherwise, the travel time has a higher weight than the road risk (e.g., a weight vector for risk-taking drivers is 0.3 for road risk and 0.7 for travel time). Each of the weight values for the road risk and travel time in the weight vector is multiply by each attribute value via EDFT. As a result, these values (i.e., weight values and attribute values) are used as the input of EDFT algorithm so that a driver agent can obtain a route plan based on its own preferences and observations on the road network [3, 4].

Secondly, EDFT calculates the preference values for all the connected paths at an intersection with the values of attributes and risk weight inferred by BBN. Then, EDFT selects the most preferred path among the connected paths based on the preference values. This selection procedure is repeated to compute the choice probabilities of all the connected paths at the intersection. At the end, by multiple replications, EDFT provides the choice probabilities for the connected paths.

Finally, the choice probabilities obtained by EDFT are used for PDFS to select a path among all the connected paths at the intersection. In other words, the choice probabilities are used as the direction information for PDFS to search a destination (i.e., an informed search process). After selecting a path, the planning horizon of PDFS moves onto the next stage to select a next path. Then, the entire computation procedure of the choice probabilities involving EDFT and BBN is repeated at the next intersection. As a result, the PDFS algorithm generates an en-route plan with multiple paths when the PDFS searching process finds a destination within the considered planning horizon.

### 3.3 Sample Complexity of E-BDI Framework

As mentioned in Section 3.2, the E-BDI based en-route planning is a probabilistic route planning algorithm with BBN. It means that constructing a robust BBN, which represents driver’s perceptual process, significantly affects the en-route planning process. By considering the sufficient number of samples, the E-BDI based en-route planning can provide a reliable route plan. According to Hoeffding’s inequality [12], the sufficient number of samples for each environmental variable can be calculated.

**Lemma 3.3.1 (Sufficient number of samples) [12]:** Let $X_1, \ldots, X_d$ be i.i.d. observations such that $\bar{X} = (X_1 + \cdots + X_d)/d$ and $P(X_i \in [a,b]) = 1$. Then, for any $\epsilon > 0$ with confidence $1-\delta$ in the estimate, the sufficient number of samples $d$ satisfies Equation (3).  

$$
\frac{1}{2 \epsilon^2} \frac{2}{\delta} (b-a)^2 \leq d.
$$

Equation (3)
Kim, Son, Tian, Chiu, and Yang

Proof. According to Hoeffding’s inequality [12], for any \( \varepsilon > 0 \),

\[
P\left( \left| \overline{X} - E(\overline{X}) \right| \geq \varepsilon \right) \leq 2 \exp\left( -\frac{2d^2 \varepsilon^2}{d(b-a)^2} \right)
\]

Since we desire confidence \( 1-\delta \) in the estimate, we want the right-hand side in the above to be at most \( \delta \).

\[
\delta \leq 2 \exp\left( -\frac{2d^2 \varepsilon^2}{d(b-a)^2} \right) \iff \frac{2d^2 \varepsilon^2}{(b-a)^2} \geq \ln \frac{2}{\delta} \iff d \geq \frac{1}{2\varepsilon^2} \ln \frac{2}{\delta} (b-a)^2
\]

However, a BBN has a conditional probability to describe the dependencies between variables. Thus, Kullback-Leibler divergence (KL-divergence) [13] is considered to get the sufficient number of samples regarding the conditional probability (see Equation (4)).

\[
d_{kl}(P,h) = \sum_{x \in X} P(x) \ln \frac{P(x)}{h(x)}
\]

It is the expectation of the logarithmic difference between the probabilities \( P \) and \( h \) over \( x \). Using the KL-divergence, Dasgupta [14] and Friedman and Yakhini [15] proposed the error of any hypothesis distribution \( h \) compared to the best hypothesis distribution \( q \) (or optimal) of all the variables given a BBN. Since those studies mainly discussed cases with binary variables, we extend the case with bounded random variables in this work based on the theorem proposed by Zuk et al. [16].

Theorem 3.3.2 (Sufficient number of samples for BBN): Let \( X_{i,1}, \ldots, X_{i,d} \) be i.i.d. observations such that \( X_j = \{X_{i,1} + \ldots + X_{i,d}\} / d' \) and \( P(X_{i,j} \in [a_j, b_j]) = 1 \). \( X_i \) has \( k \) parent variable \( \pi_i \in \Pi_i \) in BBN \( G \).

\[
d' \geq \frac{|a_i - b_i|^2}{2\varepsilon^2} \ln \frac{2|a_i - b_i| \prod_j |a_j - b_j|}{\delta}
\]

Proof. Let \( q_i \) be the target distribution over \( X_i \) and \( h_i \) be hypothesis distribution from \( d' \) samples. We seek for the optimal hypothesis which is close to distribution \( q_i \). Let KL-divergence of \( X_i \) be the following equation:

\[
d_{kl}(P,q_i,h_i) = \sum_{x \in X_i} P(x_i) \sum_{\pi_i \in \Pi_i} P(x_i | \pi_i) \ln \frac{q_i(x_i | \pi_i)}{h_i(x_i | \pi_i)}
\]

From the property of KL-divergence proposed by Zuk et al. [16], the Hoeffding’s inequality can be rewritten as follows:

\[
P\left( d_{kl}(P,q_i,h_i) = d_{kl}(P,q_i,h_i) \right) \geq \eta |a_i - b_i| \prod_j |a_j - b_j| \leq 2|a_i - b_i| \left\{ \ln \frac{2d^2 \varepsilon^2}{|a_i - b_i|^2} \prod_j |a_j - b_j| \right\}
\]

where \( \eta = \left[ \ln (\gamma - \delta + 1) \right] \), \( \gamma = \min_j q_j > 0 \) over \( d' \) samples, and \( \|q_i - h_i\|_2 < \gamma / 2 \). The equation considers the number of combinations between parent variables and the number of states of variable \( X_i \). From Lemma 3.3.1, Equation (5) can be calculated.

The sufficient number of samples for BBN \( (d') \) is higher than the sufficient number of samples for each variable \( (d) \) because the states of variables in BBN are influenced by given states of parent variables (i.e., conditional probability between attributes and environmental variables).

4. Experimental Scenario
The performance of the proposed approach is evaluated using a Phoenix, Arizona road network with 11,550 nodes and 24,869 paths (see Figure 6). In this work, several experiments have been conducted on the DynusT (Dynamic Urban Systems in Transportation) simulation platform. DynusT is a Simulation-Based Dynamic Traffic Assignment (SBDTA) software that is capable of performing mesoscopic simulation and dynamic traffic assignment for large-scale and regional networks for long time periods [17]. In order to control vehicles’ en-route planning behavior in DynusT, the proposed E-BDI based en-route planning has been integrated with DynusT via web service, which enables communication between different platforms and languages (e.g. C/C++, Java) over the World Wide Web. Thus, once the web service loads the E-BDI library written in Java, the .DLL interface of DynusT written in C++ can call the E-BDI based hierarchical en-route planning algorithm when the en-route planning is triggered. In this
scenario, there are two trigger conditions for the E-BDI based hierarchical en-route planning algorithm: (1) the absolute delay threshold, which is the difference between the experienced travel time and the idealized travel time exceeds a certain user defined value, and (2) the relative delay threshold, which is the difference between the experienced travel time and the idealized travel time exceeds a certain percentage of the idealized travel time. Whenever the prevailing experienced travel time exceeds both the absolute delay threshold and the relative delay threshold, the en-route planning behavior will be triggered and the vehicle’s current status will be passed into the E-BDI model for en-route planning.

Figure 6: Snapshot of a Phoenix road network (with 11,550 nodes and 24,869 paths) in DynusT

The performance of the proposed approach is evaluated in terms of computational efficiency and accuracy. For the efficiency evaluation, computations will be discussed when the original road network is aggregated to different levels. Then, the route plans generated at different aggregation levels will be compared for accuracy assessment. Both experiments are conducted on a PC with Intel Core2 Duo P8400 @ 2.26GHz. Table 1 gives the experiment setting for this performance analysis on the Phoenix network. In this section, the following notations are used: (1) \( R_0 \) is a route plan obtained when the E-BDI based en-route planning is applied to the original road network without aggregation, and (2) \( R_m \) is a route plan obtained by the proposed E-BDI based hierarchical en-route planning when the original road network is aggregated with \( m \) zones (\( m \geq 2 \)).

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of replications</td>
<td>10</td>
</tr>
<tr>
<td>Number of vehicles in network</td>
<td>3,102,771</td>
</tr>
<tr>
<td>Number of vehicles with given O-D pair</td>
<td>113</td>
</tr>
<tr>
<td>Number of en-route planning vehicles</td>
<td>78</td>
</tr>
<tr>
<td>Simulation time horizon</td>
<td>360 min.</td>
</tr>
<tr>
<td>Origin node</td>
<td>4048</td>
</tr>
<tr>
<td>Destination node</td>
<td>7093</td>
</tr>
<tr>
<td>Delay threshold for the en-route planning</td>
<td>Absolute</td>
</tr>
<tr>
<td></td>
<td>Normal (10, 2) min.</td>
</tr>
<tr>
<td></td>
<td>Relative</td>
</tr>
<tr>
<td></td>
<td>0.5</td>
</tr>
</tbody>
</table>
4.1 Efficiency Comparison
The proposed E-BDI based hierarchical en-route planning with \( m \) zones (\( R_m \)) is compared with the E-BDI based en-route planning (\( R_0 \)) using the average computation time of individual en-route planning vehicles. Figure 7 shows the average computation time of the proposed E-BDI based hierarchical en-route planning approach under different aggregation levels. The computation time for the partitioning process is not considered since it is off-line process which does not affect the simulation run time.

![Average computation time (seconds)](image)

Figure 7: Average computation time of the E-BDI based hierarchical en-route planning

In Figure 7, the zero zone case (\( R_0 \)) has a higher average computation time than any other cases with \( m \) zones (\( R_m \)). Additionally, the average computation time increases as the number of zones increases. This is because the complexity of the high-level network increases when the original network is divided into more zones with a small number of nodes in each zone. A small number of nodes in each zone improves the computation speed of Dijkstra’s algorithm, which has \( O(|E|+|V| \log(|V|)) \) where \( V \) is the number of vertices and \( E \) is the number edges in a graph. However, in this case, since there are many borders in the high-level network, the execution number of Dijkstra’s algorithm is increased to generate the paths between borders. Thus, the Dijkstra’s algorithm cannot reduce the entire computation complexity of the proposed E-BDI based hierarchical en-route planning approach. The computation complexity of the Dijkstra’s algorithm in the proposed E-BDI based hierarchical en-route planning is shown below:

\[
o(\sum_{i=1}^{m} |A_i|(|E_i|+|V_i| \log(|V_i|)))
\]

where \( A_i \) is the number of aggregate paths in zone \( i \), \( V_i \) is the number of nodes in zone \( i \), and \( E_i \) is the number of paths in zone \( i \).

On the other hand, the PDFS algorithm in the E-BDI framework has a significant influence on the computation complexity of the proposed E-BDI based hierarchical en-route planning. This is because the DFS algorithm usually has a higher computation complexity than the Dijkstra’s algorithm. Besides, the PDFS algorithm is performed with computationally complex algorithms such as EDFT and BBN. Therefore, the PDFS algorithm, which has \( O(|V|+|E|) \), is the major factor impacting the entire computation complexity of the proposed E-BDI based en-route planning approach so that the average computation time is almost linearly increased with the higher number of zones. As a result, the computation time of the E-BDI based en-route planning can be enhanced by the proposed hierarchical approach with a high aggregation level.

4.2 Accuracy Comparison
In order to measure the accuracy of the proposed E-BDI based hierarchical en-route planning, another experiment is designed to evaluate the similarity between outcomes obtained from the proposed hierarchical planning approach and the en-route planning approach without aggregation using the following equation:

\[
\text{Accuracy} = \frac{|R_m \cap R_o|}{|R_m \cup R_o|}
\]

The accuracy is the proportion of the number of matched paths between \( R_m \) and \( R_o \) to the number of paths in \( R_m \) or \( R_o \). If \( R_m \) and \( R_o \) have exactly same paths, the accuracy value is 1.0.
Figure 8 shows the compassion of the accuracy values for different aggregation levels. In fact, since the E-BDI based en-route planning is a probabilistic approach using PDFS, EDFT, and BBN, it is unlikely to have the exactly same route plans even if the route plans are generated under the same traffic condition. That is why the accuracy for the case with zero zone is less than 1.0. Furthermore, the accuracy increases as the number of zones increases (i.e. a high-level network containing more details). If the number of borders in a high-level network is the same as the number of nodes in a road network, the proposed hierarchical en-route planning becomes the en-route planning with the original road network (without aggregation) at the low-level.

Figure 10 reveals the average travel time of vehicles with the proposed E-BDI based hierarchical en-route planning. In terms of accuracy, the average travel time with $m$ zones ($R_m$) becomes similar to that of the case with zero zone ($R_0$). Besides, the average travel time is reduced for the cases with a small number of zones. This is because most of paths in a route plan are determined by the Dijkstra’s algorithm which only considers the travel time to find the shortest path between borders. In this case, since only a small number of paths in the route plan are related to the E-BDI framework, it is possible to not utilize the advantage of the E-BDI based en-route planning (i.e. consideration of the individual driver’s perception and preference on multiple attributes (e.g., travel time and road risk)) when generating a route plan.

5. Conclusions and Future Works
In this paper, we have proposed an E-BDI based hierarchical en-route planning approach to mimic the drivers’ route planning behavior in a large scale road network. The original road network has been aggregated to a high-level road network in which the E-BDI based en-route planning algorithm is employed to mimic more realistic planning behavior without dramatic increase of the computational demand. The proposed hierarchical en-route planning approach had been implemented in Java-based E-BDI modules and DynusT® traffic simulation software. To demonstrate and validate our proposed approach, we have compared the performance in terms of computational...
efficiency and accuracy under various aggregation levels for a road network in Phoenix. The experiments have revealed that the average computation time is reduced as the aggregation level increases. On the other hand, the accuracy was increased as the aggregation level decreased. In other words, the proposed E-BDI based hierarchical en-route planning allows us to balance between the computational efficiency and estimation accuracy based on specific requirements of different applications.

For future works, the proposed E-BDI based hierarchical en-route planning can be improved by identifying the optimum aggregation level which helps achieve both the efficiency as well as the accuracy at the same time. A multi-objective optimization approach such as Pareto-based multi-objective metaheuristics, which is a widely used approach to resolve trade-off problems between two or more conflicting objectives, will be adopted to solve such a problem.

Acknowledgements
This research is supported by the FHWA EAR Project DTFH61-11-H-00015 “VASTO - Evolutionary Agent System for Transportation Outlook”. The viewpoints expressed in this paper are solely those of the authors and don’t represent the opinions of the funding agency.

References