Adaptive Vehicle Navigation via En Route Stochastic Traffic Information

Lin XIAO  
PhD student  
Department of Civil and Environmental Engineering  
The Hong Kong University of Science and Technology  
Clear Water Bay, Kowloon, Hong Kong  
Tel: +852-93211969  
E-mail: cexiaol@ust.hk  

Hong K. LO  
Professor  
Department of Civil and Environmental Engineering  
The Hong Kong University of Science and Technology  
Clear Water Bay, Kowloon, Hong Kong  
E-mail: cehklo@ust.hk  

Abstract: This paper develops an adaptive approach for vehicle navigation in a stochastic network with en route traffic information. This stochastic, adaptive approach is formulated as a probabilistic dynamic programming problem and solved through a backward recursive procedure. The formulation is designed to incorporate various sources of information and real time traffic conditions, offering flexibility to incorporate different information types deemed useful in future extensions. In this paper, we prove mathematically that the approach outperforms deterministic formulations in a statistical sense. Results from numerical examples are included to show promise of this adaptive routing policy.

Keyword: adaptive navigation, dynamic programming, real traffic condition, transition probability, conditional probability

1. INTRODUCTION

En route vehicle navigation has become common nowadays. A typical navigation system works like this: the onboard navigation device gets the vehicle’s location from its embedded GPS chip and displays its location on a pre-loaded digital map. In some systems, the information provider also provides near real-time traffic data to its clients, i.e., the driver or onboard navigation device. It is then up to the navigation device or driver to decide how to make use of such data for navigation purposes.

Many algorithms have been developed to generate routes for navigation. The simplest is the shortest path algorithm, such as the Dijkstra algorithm (Dijkstra 1959). Attempts to improve the computational speed of the Dijkstra algorithm were reviewed in Wagner and Willhalm (2006), which can be broadly classified into two classical techniques: bidirectional search and goal-directed search which is more commonly known as the A* algorithm (Hart, Nilsson and Raphael 1968). Basically, these algorithms maintain two node lists, an open list from which potential successor node is selected and a close list of nodes that have already been selected. An important extension of the simple shortest path approach is the generation of alternative paths, such as the k-shortest path approach (Eppstein 1998). Chen, Bell et al. (2007) reviewed methods for generating multiple paths in the context of route guidance. Bell (2009) modified the classical A* shortest path algorithm to generate a set of attractive paths, called hyperpaths. The advantage of hyperpaths is that multi-paths can be generated to learn and accommodate drivers’ preferences (Park et al. 2007).
Besides the development of faster shortest path algorithms, for vehicle navigation, it is essential to acknowledge the fact that traffic networks are neither static; nor perfectly deterministic. Such recognitions open up new dimensions to enrich the consideration. For example, Dell et al. (2008) proposed a way of finding shortest path in piecewise continuous time-dependent networks; Nie (2009) considered shortest paths with on-time arrival probability. Instead of determining a particular path, various adaptive routing policies have been proposed. This kind of consideration was first introduced in the context of transit routing by Spiess (1989). There are other approaches as well, such as Opasanon (2006) and Gao (2006), who developed adaptive routing policy in stochastic, time-dependent networks.

Actually, the quality of routings determined depends very much on the nature of travel time encountered. In the ideal case, if traffic is in steady state, i.e. the travel time of every link remains unchanged over time (for example, during early morning hours, say, 2 a.m.), even the simple shortest path algorithm can produce quality routes. In reality, link travel times are both stochastic and time-variant. These uncertainties should be taken into account when generating paths. Moreover, traffic data collection and processing, and link travel time estimation take time; therefore, the traffic information collected is already obsolete, to different degrees. If such traffic information is used directly for navigation, the delay or time lag between the actual and reported traffic conditions would hamper the routing quality. This is especially so during the peak period or transitions to and from it when the traffic loadings on the network have high variations.

On the other hand, while a vehicle is traveling, it automatically experiences or collects the actual traffic conditions, which may be different from the information available or reported. This actual information can be incorporated into the optimization as feedbacks, allowing the vehicle to perform better by adaptive rerouting. This idea motivates a class of stochastic optimization, often referred to as adaptive control. The applications of this concept in transportation can be found in Hall (1986); Lo and McCord (1998) applied it in the context of ocean current routing; and Fu (2001) formulated and developed a label-correcting algorithm for solutions. More recently, Miller-Hooks and Yang (2005) proposed a fast technique of updating paths if given arc weight in a time-varying network changes.

In this paper, we propose a vehicle navigation algorithm based on the approach of adaptive control. The objective is to minimize the expected en route travel time. Three types of information are required for this approach: system-wide traffic information broadcast, actual traffic condition (e.g. link travel time) measured by an onboard device, and historical traffic patterns captured as transition probabilities, which will be explained in more detail later. This approach operates on the premise that travel times on links are correlated with other factors. For example, a longer than expected travel time experienced on a particular link would imply higher probabilities of having longer travel times on its downstream links. Furthermore, this kind of link travel condition correlation is time-variant. That is, different correlation structures are applicable for different times, which are to be established based on historical observations. This assumption is consistent with our experience, since congestion is often experienced in peak hours. A general proof is given to demonstrate that the proposed approach can produce a better performance, i.e. shorter path travel time on average as compared with its traditional deterministic counterpart. Finally, we demonstrate the benefit of this adaptive routing approach over the deterministic shortest path approach through a numerical example.
2. FORMULATION

2.1 Modeling Uncertainty in Link Travel Time
A probability model is proposed to capture the uncertainty in link travel time. Let the transportation network be presented by a directed graph \( \mathcal{G} = (\mathcal{N}, \mathcal{A}) \), consisting of a set of \(|\mathcal{N}|\) nodes and a set of \(|\mathcal{A}|\) links. On link \((i,j)\) (from node \(i\) to node \(j\)), denote \(T(i,j)\) as the actual travel time, \(T_M(i,j)\) the broadcast travel time, and \(U_T(i,j)\) a random variable depicting the uncertainty or discrepancy between the broadcast and actual travel times. Thus the actual travel time can be written as:

\[
T(i,j) = T_M(i,j) + U_T(i,j)
\]

The probability distribution \(U_T(i,j)\) captures the accuracy (or inaccuracy) of the broadcast information, and is intrinsically a function of the frequency of information update and the measurement errors of surveillance devices. In the future, as the accuracy of broadcast traffic information is improved, \(T_M(i,j)\) will get closer to \(T(i,j)\), and the significance of \(U_T(i,j)\) will decrease. Meanwhile, based on the available information, the distribution of \(U_T(i,j)\) can be updated regularly to reflect improvements in estimating \(T_M(i,j)\). In any case, in (1), \(U_T(i,j)\) is a random variable, therefore \(T(i,j)\) is a random variable sharing a similar distribution as \(U_T(i,j)\) but with the mean shifted by \(T_M(i,j)\).

Given historical patterns of \(T(i,j)\) and \(T_M(i,j)\), we can derive the \(U_T(i,j)\) distribution by examining factors that influence \(U_T(i,j)\) and incorporating them into the distribution as conditioning or explanatory variable. In other words, the distribution \(U_T(i,j)\) can be written as \(Pr\{U_T(i,j)|CV\}\) where \(CV\) represents the conditioning variables to be determined from historical observations, taking into account factors such as spatial dependency, approach direction, time of arrival at the concerned node, etc. The detailed correlation structure may also depend on the origin-destination flows and diversion patterns. In general, these factors should be selected based on their significance to the probability distribution. For ease of illustration of the approach, in the following, we limit the conditioning variables to be the time of arrival. Nevertheless, it should be clear that the following formulation can be readily modified to incorporate other factors as well, such as, the approach direction, and upstream link travel time.

2.2 Adaptive Vehicle Routing Formulation
Probabilistic dynamic programming (Bertsekas 1987, 1995) or stochastic optimal control theory can be applied to formulate the adaptive vehicle routing problem. The advantage of control theory is that it offers the possibility of obtaining analytical solutions, while the solution procedure of dynamic programming requires discretization. For realistic problems, analytical solutions are rarely available from control theory; instead, both approaches obtain solutions through numerical approaches. Because the formulation and solution procedure of dynamic programming provide more transparent exposition, it is chosen for this study.

Intersections in the network are places where drivers can make recourses in directions. Accordingly, we divide the routing problem into a finite number of stages or decision points. A stage is defined as a location at which link switching is possible, for example arterial intersections or exits from and entrances to freeways. Denote \(S\) as the number of stages for a
particular origin-destination pair. The state vector $\lambda_i$, which records the conditions of the route up to stage $i$, is:

$$\lambda_i = [T(k,i), t(i)], k \in P(i)$$  \hspace{1cm} (2)

where $T(k,i)$ is the actual travel time experienced on link $(k,i)$; $t(i)$ is the time of arrival at node $i$ by the concerned vehicle; and $P(i)$ is the set of predecessor nodes emerging into node $i$. The decision variable, $U_i$, at each stage is to select the next node to be headed conditioned upon the state vector encountered:

$$U_i = (j|\lambda_i), \ j \in Q(i)$$  \hspace{1cm} (3)

where $Q(i)$ is the set of successor nodes emerging from node $i$. As the objective is to minimize the expected travel time, we define $J^*(\lambda_i)$, as the minimum cumulative expected travel time from stage $i$ onward to destination, and $Pr\{T_{ij}(i,j)|CV\}$ as the probability distribution of $T(i,j)$ given $\lambda_i$. By applying the principle of optimality for dynamic programming (Bellman, 1961), the backward recursive relationship in (4) determines the minimum cumulative expected travel time from node $i$ onward, via node $j$, to the destination.

$$J^*(\lambda_i) = \min_{j \in U_i} \sum_{Pr(j)} \{Pr\{T(i,j)|\lambda_i\}[T(i,j)+J^*(\lambda_j)]\}$$  \hspace{1cm} (4)

The optimal decision rule can be obtained by starting the recursive relationship in (4) backward from the destination, and performing the calculations iteratively towards the origin. When the process reaches the origin, a set of optimal decision rules is obtained, which instructs the vehicle to take the appropriate heading based on the traffic condition encountered.

The optimization does not produce a pre-determined route for the vehicle. Instead, the next direction to be taken is a function of the arrival state at a node, including arrival time at the node and the travel time encountered in the preceding link. Hence the decision rule is adaptive to the most recent traffic conditions encountered. The decision rule may take the form as in the following:

<table>
<thead>
<tr>
<th>Time of arrival at node $i$</th>
<th>Next node choice $U_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_1$</td>
<td>Node$_1$</td>
</tr>
<tr>
<td>$T_2$</td>
<td>Node$_2$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>$T_n$</td>
<td>Node$_n$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

This approach avoids the potential problem associated with the deterministic approach of route planning and guidance, which recommends the same routes to users and hence overloads those recommended routes, in a situation sometimes referred to as system-induced congestion. One crucial element of the formulation is the travel time dependency (or correlation) structure, expressed as travel time discrepancy distribution ($T_{ij}$), which is to be extracted from historical traffic patterns. To do that, the conditioning factors that influence $T_{ij}$ will have to be established first. This can be achieved via data-mining the surveillance traffic flow data which are routinely collected. To illustrate the approach and control for
different kinds of variability for future studies, we will rely on traffic simulation to generate
the various flow patterns and derive the underlying correlation or dependency patterns, as will
be further discussed in the section on numerical studies.

One more property of this formulation is that it is consistent with the classical deterministic
shortest path algorithm in the sense that if the travel time discrepancy variable $T_{U}$ gradually
diminishes, as when the technology of travel time prediction gradually matures, the proposed
adaptive formulation will reduce back to the deterministic approach. This aspect will be
further explained in the numerical example.

3. COMPARISON BETWEEN INSTANTANEOUS AND ADAPTIVE ROUTING

A directed graph, or any sub-graphs of it, in an abstract form like Figure 1 is used to represent
a network $G = (M, A)$ where $M$ represents the set of notes and $A$ the set of links.

![Figure 1 Abstract representation of a traffic network](image)

In Figure 1, node 0 and path 0 are artificial node and path (link) created to capture different
departure times from node k. In the discussion below, nodes $k$ and $k'$, respectively, are
taken as real origin and destination of any connected transportation network (graphs or its
sub-graphs).

Notations and assumptions:

$t_{k-k}^{n}$: travel time from node $k$ to node $k'$ via path $n$

It is a random variable dependent on the arrival time at the origin of this path, i.e. node
$k$. For example, $t_{0-k}^{0}$ represents the travel time from node 0 to node $k$ via path 0. For
computational purposes, travel time is discretized into $I_n$ values, covering all possible
states.

$t_{k-k}^{i} = \{t_{k-k}^{(1)}, t_{k-k}^{(2)}, \ldots, t_{k-k}^{(i)} \}$, where $i \in 1: I_n$

$T_{k}$: arrival (departure) time at (from) node k, assuming no nodal delay.

In particular, the departure time from node 0 or the artificial origin, is $T_{0}$. The
relationship between arrival times at two adjacent nodes is represented as: $T_{k} = T_{k} + t_{k-k}^{n}$. 
\( T_k \) is also a random variable.

\( p_{k-k}^{n(i)} \): conditional probability of travel time from node \( k \) to node \( k' \) via path \( n \) equals \( t_{k-k}^{n(i)} \) in terms of arrival time; the conditioning variable is the time of arrival at node \( k \).

So \( p_{k-k}^{n(i)} \) can be expressed as a function of the arrival time at node \( k \), \( p_{k-k}^{n(i)}(T_k) \). The time dimension will be omitted in the following discussion for notational simplicity.

\( E_{k-k}^n \): expected travel time from node \( k \) to node \( k' \) via path \( n \).

\[
E_{k-k}^n = \sum_{i=1}^{l_k} t_{k-k}^{n(i)} \cdot p_{k-k}^{n(i)}.
\]

Since \( p_{k-k}^{n(i)} \) is conditional probability given the time of arrival at node \( k \), \( E_{k-k}^n \) can be expressed as a function of arrival time, \( E_{k-k}^n(T_k) \).

\( TT_{SP} \): the expected experienced cumulative travel time from origin to destination via a shortest path algorithm.

\( TT_{DP} \): the expected experienced cumulative travel time from origin to destination via the adaptive routing policy.

**Proposition:**

In a given stochastic, time-dependent traffic network, the expected travel time from origin to destination via the adaptive routing policy is at least as short as that of the shortest path determined with instantaneous travel time at the time of departure.

**Proof:**

The procedure of the shortest path determined with instantaneous travel time at the time of departure is illustrated first. The approach calculates the shortest path at the time of journal departure \( T_0 \) based on instantaneous traffic information available at that time. For the traffic network shown in Figure 1, the algorithm can be formulated as:

\[
\min \left( TT_{SP} \right) = E_{0-k}^0(T_0) + \min \left\{ E_{k-k}^n(T_k) \right\}.
\]

(5)

Suppose that the solution to eqn (5) is path \( X \). This solution, i.e. the path, is fixed once (5) is solved. As the journal progresses, the travel times of the downstream links will be gradually realized. Therefore, the travel time on this path is a random variable, whose expectation can be determined by accounting for its different departure times from origin, \( T_k \). Consequently, the expected experienced cumulative travel time from origin node 0 to destination node \( k' \) can be expressed like this:

\[
E( TT_{SP} ) = \sum_{i=1}^{l_k} \left( t_{0-k}^{0(i)} + E_{k-k}^{X} (T_k) \right) \cdot p_{0-k}^{0(i)} = E_{0-k}^0(T_0) + \sum_{i=1}^{l_k} E_{k-k}^{X} (T_k) \cdot p_{0-k}^{0(i)}
\]

(6)

where \( T_k = T_0 + t_{0-k}^{0(i)} \)

On the other hand, by adopting the adaptive dynamic programming procedure, the decision rule is determined by

\[
\min \left( TT_{DP} \right) = \sum_{i=1}^{l_k} \left( t_{0-k}^{0(i)} + \min \left\{ E_{k-k}^n(T_k) \right\} \right) \cdot p_{0-k}^{0(i)}
\]

(7)

This is equivalent to
\[
\min (TT_{DP}) = \sum_{i=1}^{I_0} t_{0-k}^{(i)} \cdot p_{0-k}^{(i)} + \sum_{i=1}^{I_0} \min \left\{ E_{k-k}^{n} (T_k) \right\} \cdot p_{0-k}^{(i)} \n
= E_{0-k}^{0} (T_0) + \sum_{i=1}^{I_0} \min \left\{ E_{k-k}^{n} (T_k) \right\} \cdot p_{0-k}^{(i)}
\]

Suppose the solution is of the form below:

<table>
<thead>
<tr>
<th>Time of arrival at node k</th>
<th>Next path to enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>(T_0 = T_0 + t_{0-k}^{(1)})</td>
<td>(Y_1)</td>
</tr>
<tr>
<td>(T_k = T_0 + t_{0-k}^{(2)})</td>
<td>(Y_2)</td>
</tr>
<tr>
<td>(\ldots)</td>
<td>(\ldots)</td>
</tr>
<tr>
<td>(T_k = T_0 + t_{0-k}^{(i)})</td>
<td>(Y_i)</td>
</tr>
<tr>
<td>(\ldots)</td>
<td>(\ldots)</td>
</tr>
<tr>
<td>(T_k = T_0 + t_{0-k}^{(I_0)})</td>
<td>(Y_{I_0})</td>
</tr>
</tbody>
</table>

The expected experienced cumulative travel time of the realization of such strategy is:

\[
E (TT_{DP}) = E_{0-k}^{0} (T_0) + \sum_{i=1}^{I_0} E_{k-k}^{Y_i} (T_k) \cdot p_{0-k}^{(i)} \tag{8}
\]

Comparing eqns (6) and (8), the first items on right hand side of both are the same, meanwhile, it’s noted that both equations have \(I_0\) items. The difference or discrepancy between them therefore is:

\[
D = E (TT_{DP}) - E (TT_{SP}) = \sum_{i=1}^{I_0} (E_{k-k}^{Y_i} (T_k) - E_{k-k}^{X_i} (T_k)) \cdot p_{0-k}^{(i)}
\]

For each \(E_{k-k}^{Y_i} (T_k)\), the superscript \(Y_i\) is actually the solution to \(\min \left\{ E_{k-k}^{n} (T_k) \right\}\) which implies that \(E_{k-k}^{Y_i} (T_k) \leq E_{k-k}^{X_i} (T_k)\). Moreover, for each \(i\), \(p_{0-k}^{(i)}\) is positive by definition. As a result, \(D = \sum_{i=1}^{I_0} (E_{k-k}^{Y_i} (T_k) - E_{k-k}^{Y_i} (T_k)) \cdot p_{0-k}^{(i)} \leq 0\).

According to the above results, it can be concluded that the expected travel time from node \(k\) to node \(k'\) by adopting the adaptive routing policy is at least as short as that of the shortest path determined with instantaneous travel time at the time of departure.

This proof can be extended to a more general network. Suppose a directed graph \(G = (M, A)\) consists of a set of \(|M|\) nodes and a set of \(|A|\) links is used to represent a general network. Node 1 and node \(M\) are the origin and destination of the directed graph, respectively. Following the discussion hereinbefore, one can gain the same conclusion for that traffic network by replacing node \(k\) and node \(k'\) by node 1 and node \(M\).

4. NUMERICAL STUDIES
Simulation tests are conducted to investigate the performance of the proposed adaptive routing algorithm. This section first describes the illustrative scenario and simulation method. After that, based on the simulation, we examine the distribution of link travel time and verify the assumption of whether it relies on the time of arrival at the node concerned. The main analysis in this section is the comparison between the classical deterministic shortest path algorithm and this proposed adaptive routing policy.

4.1 Traffic Scenario Setting and Simulation
We construct a directed network, \( G = (25, 40) \), of 25 nodes and 40 links, as shown in Figure 2. Each node represents a signalized intersection while each arc represents link between intersections. The entire network is modeled with the cell-transmission model (Daganzo 1994, 1995), which includes queue spill and junction blockage. According to the cell transmission model (CTM), the travel time for vehicles traveling at free flow speed on a link is set to be 3 minutes. At each intersection, there is a fixed four-phase signal plan based on Webster’s formula (Webster 1958), including protected right turning (Hong Kong case). The cycle time is 90 seconds. This four-phase signal plan is summarized in Figure 3. The traffic scenario involves a typical morning peak from 7:00 a.m. to 9:00 a.m. This 120-min period is discretized into 24 5-min time intervals. We track the average link travel time on each link during each time interval. The navigation guidance applies to vehicles traveling from node 1 to node 25. Traffic enters the network with an average volume of 1000 veh/hr or 1400 veh/hr (peak volume). Three peaks appear from 7:15-7:45, 8:00-8:15 and 8:30-8:45. During these peaks, traffic volume jumps from 1000veh/h to 1400veh/hr. The uncertainty of link travel time mainly comes from stochastic turning movements. The probability of vehicles making left-turns at an intersection is set to be 0.3 while that of right-turns is 0.2.

The simulation for the entire peak period is repeated 100 times (representing 100 days) to generate and gather the historical traffic pattern.
4.2 Link Travel Time Distribution
As discussed earlier, the conditioning variables are limited to be the time of arrival at an intersection. After the 100-day simulation, the average travel time on each link for each time interval is collected in a 24 x 40 matrix. In the interest of space, we do not show them all here but just illustrate the results for one link (link 15), whose link travel time distributions are shown in Figure 4. The X-axis represents the delay, which is equivalent to travel time minus 3 minutes in this case, on link 15, while the Y-axis is the probability density.
4.3 Results Comparison
In this part, we study the potential benefit of the adaptive routing policy. From a deterministic point of view, the shortest path from node 1 to node 25 was first solved by eqn (5). In this particular network, the solution was \{link5, link14, link19, link24, link29, link30, link35, link40\}, as shown in Figure 5:

![Figure 5 Deterministic shortest path, the darker line (red in color)](image-url)
If the transition probability of link travel time was taken into consideration, eqn (7) was adopted to make an adaptive optimal routing policy. Since at each stage (intersection or node in this example), different states (time of arrival at this node) may lead vehicles to different link to enter, the final solution would be a 24-by-40 look up table. Table 3 is the visualization of part of the final solution.

Table 3 Illustration of the adaptive routing policy lookup table

<table>
<thead>
<tr>
<th>node 1</th>
<th>time interval</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>next link</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>node 7</th>
<th>time interval</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>next link</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>11</td>
<td>11</td>
<td>15</td>
<td>15</td>
<td>11</td>
<td>11</td>
<td>1</td>
</tr>
</tbody>
</table>

For example, if a vehicle arrives at node 7 during the 3rd interval, which is equivalent to 7:10am – 7:15am, it should proceed to link 15. If the vehicle arrives 10 minutes later, it should proceed to link 11.

Suppose vehicles leave node 1 for node 25 at different times during the morning peak. The departure time is rounded to a certain 15-minute time interval. To compare the performance between the deterministic instantaneous shortest path and the adaptive routing policy, we look at the resultant travel times between them, as shown in Table 4 and Figure 6.

Table 4 Difference in average total travel time between deterministic shortest path and adaptive routing policy

<table>
<thead>
<tr>
<th>Departure time interval</th>
<th>Average total travel time(min)</th>
<th>Departure time interval</th>
<th>Average total travel time(min)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Deterministic shortest path</td>
<td>Adaptive routing policy</td>
<td>Deterministic shortest path</td>
</tr>
<tr>
<td>1</td>
<td>38.7</td>
<td>37.9</td>
<td>13</td>
</tr>
<tr>
<td>2</td>
<td>39.3</td>
<td>40.1</td>
<td>14</td>
</tr>
<tr>
<td>3</td>
<td>40.2</td>
<td>41.1</td>
<td>15</td>
</tr>
<tr>
<td>4</td>
<td>41.3</td>
<td>40.4</td>
<td>16</td>
</tr>
<tr>
<td>5</td>
<td>46.8</td>
<td>42.1</td>
<td>17</td>
</tr>
<tr>
<td>6</td>
<td>48.6</td>
<td>42.1</td>
<td>18</td>
</tr>
<tr>
<td>7</td>
<td>46.9</td>
<td>42.4</td>
<td>19</td>
</tr>
<tr>
<td>8</td>
<td>45.8</td>
<td>41.6</td>
<td>20</td>
</tr>
<tr>
<td>9</td>
<td>45.4</td>
<td>41.2</td>
<td>21</td>
</tr>
<tr>
<td>10</td>
<td>45.9</td>
<td>39.7</td>
<td>22</td>
</tr>
<tr>
<td>11</td>
<td>38.2</td>
<td>39.1</td>
<td>23</td>
</tr>
<tr>
<td>12</td>
<td>38.1</td>
<td>39.4</td>
<td>24</td>
</tr>
</tbody>
</table>
The red (darker) curve in Figure 6 is the average total travel time if drivers choose to use the deterministic shortest path. Three peaks appear since the traffic scenario was set to have 3 time periods with higher traffic volumes. It is clear that the average path travel time by adopting proposed routing policy is shorter than that by following a deterministic path, especially when the traffic volume is very high.

4.4 Further Discussions

This part is used to show how our adaptive routing method reduces to a deterministic one when the effect of $T_u$ gradually diminishes. To simulate this effect, for example, each stage has only one state. In our example and discussion, this assumption implies that the whole time period, 7:00am-9:00am, is considered as one time interval. However, this is an extreme case. Here we investigated the situation when the 2-hour morning peak was discretized into 8 15-minute time intervals. Figure 7 shows the outcome by taking the similar procedure above.

Benefit is still significant but much less than previous 5min time interval case. The red (darker) and green (lighter) curves are now closer to each other. This result can be explained
in the following. A longer time interval gives the adaptive method less flexibility; and smaller effects of the conditioning variable. In other words, less stochastic information is available to help the algorithm establish the adaptive routing policy and the policy is consequently not as prominent.

5. CONCLUSIONS

We note that the adaptive routing approach is not guaranteed to outperform its deterministic counterpart in every trial. However, it is guaranteed that its average performance is better in the presence of stochastic link travel times. Moreover, the results show that the benefit of the adaptive routing policy is higher in heavy traffic condition. One crucial assumption to the adaptive vehicle routing formulation is that the dependency (or correlation) structure, expressed as travel time discrepancy distribution \( T_{ij} \), can be captured from historical traffic patterns. To do that, the conditioning factors that influence \( T_{ij} \) will have to be established first. Traffic patterns generated by simulation models under different scenarios may be used to explore the dependency relationship. Factors that have significant influence on \( T_{ij} \) will be selected as the conditioning elements. Hence, in order to gain more improvement, a more refined approach of estimating the probabilities and expanding the formulation to a multi-stage process should be necessary.

Summarizing, we have formulated a vehicle routing approach that is adaptive to the traffic conditions encountered. We have also demonstrated the benefits of this adaptive approach over the deterministic approach. In this platform, instead of considering only one kind of stochastic information, incorporating various sources of traffic information is possible, which may include not only the time of arrival at an intersection but also volumes of neighboring links.

REFERENCES


