Real-Time Decision Making for Urban Vehicle Navigation

C. L. Philip Chen, Fellow, IEEE  
Department of Computer and Information Science  
University of Macau  
Macau, China  
philip.chen@ieee.org

Jin Zhou, Student Member, IEEE  
Department of Computer and Information Science  
University of Macau  
Macau, China  
ise_zhouj@ujn.edu.cn

Wei Zhao, Fellow, IEEE  
Department of Computer and Information Science  
University of Macau  
Macau, China  
weizhao@umac.mo

Abstract—In a large-scale wireless sensor traffic network, collecting and processing of the global real-time traffic information are often unreliable. Making real-time navigation decision becomes an arduous task. To address this issue, an efficient real-time vehicle navigation algorithm is proposed, in which multiple local traffic information are considered to make navigation decision in a quick and accurate way. At the same time, a general distance metric is defined for the processing of both exact and fuzzy data. In addition, the algorithm can provide various navigation decisions according to the choice of different attributes to meet the diverse navigation requirements of drivers. Simulation results show the suitability and efficiency of the proposed algorithm.

Keywords-real-time vehicle navigation; wireless sensor network; hybrid multi-attribute decision making

I. INTRODUCTION

In recent years there has been a resurgence of interest in Cyber-Physical Systems and Intelligent Transportation Systems (ITS). This is directly attributed to the recent developments in the field of Vehicle Navigation System (VNS), where there is an actual need to search the optimal routing path from a source to a destination within a reasonable time for achieving the least cost, i.e., travel time or travel distance. Most of the works in this area focus on developing increasingly efficient optimal algorithms. Dijkstra algorithm [1] and A* algorithm [2] are two well-known shortest path algorithms. Other variants of these two algorithms are further discussed and popularized in [3-5]. However, these approaches are mostly devised for static networks. They are not efficient when directly applied on dynamic shortest path planning in a real-time traffic environment. In practice, the navigation route needs to be re-planned frequently according to the dynamics of the traffic conditions. Some researchers pay more attention to the efficient re-optimization strategies on dynamic graphs [6-7]. Recently, a heuristic dynamic algorithm has been discussed in [8]. This algorithm considers both the changing locations of the vehicle and the changing traffic conditions in the navigation area. Based on multi-path routing strategy, Chen et al. [9] and Bell [10] propose multi-path dynamic vehicle navigation algorithms.

Recently, with the development of microelectronic technologies, low-energy-consumption, low-cost and relatively powerful wireless sensor networks (WSNs) have been implemented in the ITS [11-12]. A great deal of new energy efficient and real-time solutions for traffic data sensing and collecting are discussed in [13-14]. At the same time, based on WSN techniques, a lot of efforts in vehicle navigation have been made. Huang et al. [15] put forward a distance aware epidemic navigation algorithm using the vehicular ad hoc network. Chang et al. [16] apply three metrics (road traffic density, road class and road distance) in ITS to determine the optimal navigation route. Recently, Skog et al. [17] give a survey of car positioning and navigation technologies.

These dynamic vehicle navigation approaches are efficient in information collecting and traffic control, but there are still some problems.

1. Collecting and processing the global real-time traffic information is often unreliable in a large-scale WSN-based traffic network with energy and bandwidth constrained sensors.

2. Most of the existing vehicle navigation algorithms try to pursue the minimum travel distance or the minimum travel time. Usually they can not achieve an effective trade-off.

3. Single traffic information or simple mathematic model using for navigation decision making is not very sufficient. Comprehensive consideration of all information is needed to assist navigation.

An algorithm that can take into account all kinds of local traffic information to make real-time navigation decision in a quick and accurate way is in dire need. In this paper, we propose a novel WSN-based dynamic vehicle navigating algorithm, in which multi-attribute decision making (MADM) method is applied in solving real-time navigation decision making problem. In addition, because the real-time traffic information involved in the navigation decision making should not all denoted by exact data, some of them are more suitable to be denoted by fuzzy data, a general distance metric is presented for processing the hybrid data. As a result, the proposed navigation scheme provides a general method for navigation decision making, namely, it can provide various navigation decisions on the choice of different attributes to meet diverse navigation requirements of drivers, including the
least travel distance, the least travel time or a trade-off of all conditions.

The rest of the paper is organized as follows. The WSN-based real-time vehicle navigation algorithm is presented in section 2. The real-time navigation decision making strategy based on hybrid MADM method is discussed in section 3. Section 4 shows the simulation results. Finally, conclusions are given in section 5.

II. WSN-BASED REAL-TIME VEHICLE NAVIGATION ALGORITHM

A. System Architecture

A WSN-based vehicle navigation system can be modeled as a directed graph, \( G=(V,E) \), which consists of a set of intersection nodes \( V \), and a set of links \( E \) (a road exists between two intersection). The optimal navigation path from the source to the destination can be defined as a sequential list of directed links with the least cost.

Without loss of generality, giving a sensor network that is capable of transmitting and receiving signals, as shown in Fig. 1, there are two types of sensor nodes in our system, i.e., the vehicle sensor nodes in the vehicles and the intersection sensor nodes at the intersections. Each vehicle sensor node collects the host vehicle’s position, travel direction and velocity data (obtained from different sensors such as GPS and odometer installed in the vehicle) and transmits them to the intersection sensor nodes. The intersection sensor nodes are responsible for collecting the information coming from vehicle sensor nodes and computing the average vehicle velocity and the road density of each adjacent road. All the traffic information will be transmitted between sensor nodes using a simple “hello” protocol [18]. A two-layer hierarchical communication channel is established among sensor nodes as shown in Fig. 1. One layer is the vehicle-intersection sensor network between the vehicle and intersection sensor nodes, and the other layer is the in-intersection sensor network among intersection sensor nodes.

![Two-level hierarchical network structure in WSN-based vehicle navigation system.](image)

B. Traffic Information

To achieve an efficient and robust routing operation, five kinds of traffic information are considered in our system. They are classified into two categories: the benefit attributes (the-larger-the-better type) and the cost attributes (the-smaller-the-better type) as follows.

- **Road Distance**: It indicates the actual distance of a road. Navigation algorithms aim to achieve the least travel distance. It is a cost type attribute.
- **Road Width**: It denotes the number of lanes of a road. In this paper, we classify it into four priorities, i.e., 1, 2, 3 and 4, the higher of the lane status, the better of the convenient travel condition. It is a benefit type attribute.
- **Vehicle Velocity**: It indicates the value of vehicle velocity on a road. It is important for the dynamic vehicle navigation which aims to achieve the least travel time. It is a benefit type attribute.
- **Road Density**: It stands for the number of vehicles per unit distance on a road. A high road density usually indicates the congested traffic and vice versa. It is a cost type attribute.
- **Navigation Cost**: It means the expected cost (denoted as distance in this paper) for the driver to travel from an intersection position to the destination. This information can be obtained by an “up-down” broadcast protocol through the in-intersection sensor networks (the details can be seen in the next subsection). All intersection sensor nodes with navigation cost in the network form a cost field, which provides the global travel direction towards the destination implicitly. It is a cost type attribute.

Among the five traffic information, the road distance and road width are fixed ones kept in intersection sensor nodes initially; the vehicle velocity and road density are to be measured in real time; the proposed navigation cost will be obtained through an “up-down” broadcast protocol at the beginning of navigation.

C. Real-time Navigation Algorithm

This sub-section gives a detailed description of the WSN-based vehicle navigation algorithm. The algorithm consists of four phases.

1. **Static road information collecting phase**

The static road information, i.e., road distance and road width, should be loaded in the intersection sensor nodes in advance.

2. **Navigation cost information collecting phase**

In this phase, an “up-down” broadcast protocol as shown in Fig. 2 is introduced to set up the navigation cost field. The protocol includes two steps. In the first step, a source vehicle’s navigation query will be transformed into an “interest” and transferred to the adjacent intersection sensor nodes. These intersection sensor nodes then flood the “interest” throughout
the in-intersection sensor network towards the destination, where the squared nodes represent vehicle sensor nodes and the circles nodes are the intersection sensor nodes. A dashed circle represents the broadcasting area of a sensor. The black arrows indicate the directions of information flow. When one intersection sensor node in the destination region matches the “interest”, it activates itself as the destination node \( N_d \). The second step of the protocol is a reverse process of the first step. The red arrows indicate the directions of information flow. The destination node \( N_d \) broadcasts an advertisement packet announcing a navigation cost of 0 (\( \text{Cost}(N_d) = 0 \)). The rest of intersection sensor nodes initially have a navigation cost of \( \infty \) (\( \text{Cost}(N_j) = \infty \), for \( i \neq d \)). Referring to Fig. 3, upon hearing an advertisement packet containing the navigation cost from the sender \( N_j \), the adjacent node \( N_i \) calculates a new cost \( \text{Cost}(N_i) + \text{LinkCost}(N_j, N_i) \), where \( \text{LinkCost}(N_j, N_i) \) is the link distance between node \( N_j \) and node \( N_i \). The node \( N_i \) compares its current cost \( \text{Cost}(N_i) \) with \( \text{Cost}(N_j) + \text{LinkCost}(N_j, N_i) \), and chooses the smaller one as its new navigation cost. Then the node \( N_i \) continues to broadcast advertisement with the new cost towards the source vehicle. Eventually, every intersection sensor node may calculate the optimal navigation cost to the destination through flooding. Final navigation cost field is set up.

**Figure 2.** An “up-down” broadcast protocol for building navigation cost field.

**Figure 3.** An illustrative example for calculating navigation cost of each intersection node.

### (3) Real-time local traffic information collecting phase

The real-time local traffic information is sensed and collected in the vehicle-intersection sensor network in this phase, and these data will be used to guide the navigation decision making. Each intersection sensor node broadcasts messages periodically including its ID and position. Normally, the vehicle sensor nodes are in the listening state. When a vehicle sensor node comes into the broadcast range of an intersection sensor node and receives its message, the vehicle sensor node will switch to the active state. Then it collects the host vehicle’s location, travel direction and velocity data, and sends the information to the intersection sensor nodes nearby. The intersection sensor nodes receive the real-time road information from the vehicle sensor nodes around, and compute the average vehicle velocity and the road density of each adjacent road.

### (4) Real-time navigation decision making phase

When the vehicle approaches an intersection, the vehicle sensor node can obtain all the traffic information promptly from the intersection sensor nodes nearby using a simple “hello” protocol. Multiple traffic attributes can be used to make real-time navigation decision, which help the driver to select one of the roads connected to the intersection (the details will be described in the following section). The phase will be repeated until the end of navigation.

### III. REAL-TIME NAVIGATION DECISION MAKING STRATEGY

When the vehicle approaches an intersection, it will select one of the adjacent intersection nodes as its direction of next move. Five kinds of traffic information, i.e., road distance, road width, vehicle velocity, road density and navigation cost, are considered as attributes, with which alternatives’ performance are measured. Then the node selection or the navigation decision can be considered as a MADM problem and concisely expressed in a decision matrix format \( A = [a_{ij}]_{n \times m} \).

\[
A = \begin{bmatrix}
    a_{11} & a_{12} & \cdots & a_{1m} \\
    a_{21} & a_{22} & \cdots & a_{2m} \\
    \vdots & \vdots & \ddots & \vdots \\
    a_{n1} & a_{n2} & \cdots & a_{nm}
\end{bmatrix}
\]

where \( a_{11}, a_{12}, \ldots, a_{nm} \) are traffic attribute vectors for the \( n \) possible adjacent intersection sensor nodes among which the decision maker should to choose. Each intersection node has \( m \) attributes. The \( a_{ij} \) (\( 1 \leq i \leq n \), \( 1 \leq j \leq m \)) is the \( j \)-th traffic attribute value of the \( i \)-th intersection node.

Technique for order preference by similarity to ideal solution (TOPSIS) has been proven to be a useful technique in dealing with MADM problems in many applications [19]. It bases upon the concept that the chosen alternative should have the shortest distance from the positive ideal solution and the farthest distance from the negative ideal solution. Unfortunately, the traditional TOPSIS method is used just for exact data. But in vehicle navigation, the traffic attributes involved in the decision problem should not all denoted by exact numbers, some of them are more suitable to be denoted by fuzzy numbers, such as vehicle velocity and road density. In this paper, we further extend the concept of TOPSIS and develop a new method named Hybrid-TOPSIS, where a general distance metric is defined for processing both exact and fuzzy data. In the rest of the section, we start with some basic definitions. Then the process of Hybrid-TOPSIS method is presented.
A. Preliminaries

**Definition 1.** A fuzzy set $a$ in a universe of discourse $X$ is characterized by a membership function $\mu_a(x)$ which associates with each element $x$ in $X$, a real number in the interval $[0, 1]$. The function value $\mu_a(x)$ is termed the grade of membership of $x$ in $a$ [20].

In this paper, the fuzzy data is denoted as a triangular fuzzy number $a$, which is a fuzzy set defined by a triplet $(e_1, e_2, e_3)$ shown in Fig. 4.

![A triangular fuzzy number $a$.](image)

Seven linguistic terms, i.e., very low (VL), low (L), medium low (ML), medium (M), medium high (MH), high (H) and very high (VH), are used to express the fuzzy data.

**Definition 2.** A triangular fuzzy number $a$ can be defined as one case of $L-R$ fuzzy number [21]. The left and right parts of membership function can be expressed as $\mu^L_a(x)$ and $\mu^R_a(x)$, and $x^L_a(u)$ and $x^R_a(u)$ are their corresponding inverse functions.

In the decision matrix $A$, the attribute value may be an exact number or a triangular fuzzy number, the basic operation corresponding distance between them is defined as

$$d(a_{ij}, a_{kj}) = 2 \left| a_{ij} - a_{kj} \right|$$

**Definition 3.** Let $a_{ij}$ and $a_{kj}$ are two exact attribute numbers, the distance between them is defined as

$$d(a_{ij}, a_{kj}) = 2 \left| a_{ij} - a_{kj} \right|$$

**Definition 4.** Let $a_{ij}$ and $a_{kj}$ are two triangular fuzzy attribute numbers, the distance between them [21] is defined as

$$d(a_{ij}, a_{kj}) = \int_0^1 \left[ \left| x^L_{ij}(u) - x^L_{kj}(u) \right| + \left| x^R_{ij}(u) - x^R_{kj}(u) \right| \right] du$$

**Definition 5.** Let $a_{ij}$ is a general attribute number for the $j$-th attribute of the $i$-th alternative, we denote it as a triplet $a_{ij} = (a^{(1)}_{ij}, a^{(2)}_{ij}, a^{(3)}_{ij})$, subject to $a^{(1)}_{ij} \leq a^{(2)}_{ij} \leq a^{(3)}_{ij}$. If $a_{ij}$ is an exact number, then $a_{ij} = a^{(1)}_{ij} = a^{(3)}_{ij} = a^{(3)}_{ij}$, while if $a_{ij}$ is a triangular fuzzy number, then $a^{(1)}_{ij} = e_1$, $a^{(2)}_{ij} = e_2$, $a^{(3)}_{ij} = e_3$.

**Definition 6.** Let $a_{ij}$ and $a_{kj}$ are two general attribute numbers, the general distance between them is defined as

$$d(a_{ij}, a_{kj}) = \int_0^1 \left| (a^{(2)}_{ij} - a^{(1)}_{ij}) - (a^{(2)}_{kj} - a^{(1)}_{kj}) \right| u + (a^{(3)}_{ij} - a^{(2)}_{ij}) \right| du$$

We can prove that the distance $d$ is a generalization of $d_1$ and $d_2$. In the following Hybrid-TOPSIS decision scheme, $d$ is used as the general distance metric of two traffic attributes.

**Definition 7.** Let $w$ is a numerical data, $a_{ij}$ are a general attribute number, the basic operation $w \cdot a_{ij}$ is defined as

$$w \cdot a_{ij} = (w \cdot a_{ij}^{(1)}, w \cdot a_{ij}^{(2)}, w \cdot a_{ij}^{(3)})$$

**B. Hybrid-TOPSIS Decision Scheme**

The process of Hybrid-TOPSIS decision scheme is shown as follows.

**Step1:** Normalize the decision matrix $A = [a_{ij}]_{n \times m}$ to get the normalized decision matrix $R = [r_{ij}]_{n \times m}$

Normalization seeks to obtain comparable scales, which is a linear transformation as defined in (6).

$$r_{ij} = \frac{a_{ij}^{(1)}}{a_{ij}^{(2)}}, \frac{a_{ij}^{(2)}}{a_{ij}^{(3)}}, \frac{a_{ij}^{(3)}}{a_{ij}^{(1)}}, j \in S_1$$

where $a_{ij} = \text{max}(a_{ij}^{(1)}, a_{ij}^{(2)}, a_{ij}^{(3)})$, $S_1$ is the set of benefit attributes (the-larger-the-better type), $S_2$ is the set of cost attributes (the-smaller-the-better type), $1 \leq i \leq n$, $1 \leq j \leq m$.

**Step2:** Calculate the weight vector $w = [w_1, w_2, \ldots, w_m]$ based on the maximizing deviation method [22].

$$w_j = \frac{\sum_{i=1}^n d(r_{ij}, s_{ij})}{\sum_{j=1}^m \sum_{i=1}^n d(r_{ij}, s_{ij})}, \text{ for } 1 \leq j \leq m$$

**Step3:** Construct the weighted normalized decision matrix $Z = [z_{ij}]_{n \times m}$ as (8), and identify positive ideal ($z^*$) and negative ideal ($z^-$) solutions.

$$z_{ij} = w \cdot r_{ij} = (w_1 r_{ij}^{(1)}, w_2 r_{ij}^{(2)}, w_3 r_{ij}^{(3)})$$

$$z^* = [z^*_1, z^*_2, \ldots, z^*_m]$$

$$z^- = [z^-_1, z^-_2, \ldots, z^-_m]$$

**Step4:** Obtain the $m$-dimensional Euclidean distance of each intersection node to $z^*$ and $z^-$ as $s$ and $s^*$.

$$s_i = \sqrt{\sum_{j=1}^m d(z_{ij}, z^*_j)^2}, \text{ for } 1 \leq i \leq n$$

$$s^*_i = \sqrt{\sum_{j=1}^m d(z_{ij}, z^-_j)^2}, \text{ for } 1 \leq i \leq n$$

**Step5:** For each intersection node calculate the closeness $c_i$ according to (12).
\[ c_i = \frac{s_{\text{max}}}{s_i + s_{\text{max}}}, \quad \text{for } 1 \leq i \leq n \]  

We rank all the adjacent intersection nodes according to their closeness \( c_i \). The node with the maximum \( c_i \) is selected as the best one for navigation.

IV. SIMULATION RESULTS

To demonstrate the suitability and efficiency of the proposed algorithm, an 8 by 8 grid network followed by a real-world urban road network from Beijing are employed for simulation. SUMO (Simulation of Urban Mobility) [23], a suite of microscopic and continuous road traffic simulation software tools developed by German Aerospace Center, is used to model the movement and behavior of vehicles in traffic networks and help us obtain dynamic traffic data that is similar to those in the real world. Four navigation requirements are discussed, including the least travel distance, the least travel time, the best lane status (the maximum road width), and the comprehensive consideration of multiple attributes. Three performance metrics are introduced for comparative analysis, i.e., travel distance (TD), travel time (TT) and the mean of velocity (MV).

A well designed 8 by 8 grid network shown in Fig. 5 is first given as an example to show the validity of the proposed algorithm. In the figure, an intersection node is represented by a solid circle, and the node number is displayed nearby. A link between two intersection nodes is denoted by a black line, and its length indicates the road distance. All links can be traveled in either direction, leading to 224 directional paths. In order to show the effects of different traffic conditions on each navigation requirement, some parameters are provided as follows. The length of horizontal link is 200\( \text{m} \), and the length of vertical link is 100\( \text{m} \). The number of lanes is set to 4 for the roads on (9-16) and (2-58). Here \((a-b)\) denotes the line from node \(a\) to node \(b\). The number of lanes for other roads is set to 1. The traffic on (1-8) and (9-16) is set as congested, while the traffic on (17-24) and (25-32) is set as relatively crowded. The maximum velocity on each road is 40\( \text{km/h} \).

When the start of navigation is set at node 1 and the destination is set at node 8, four navigation paths marked with different colors and the marks are shown in Fig. 5. The corresponding navigation performances are listed in Table 1. Without considering road traffic, the path 1 (the carmine line) can achieve the minimum travel distance (1400\( \text{m} \)), but with the maximum travel time (282.11\( \text{s} \)) and the lowest mean of velocity (17.87\( \text{km/h} \)). On the contrary, the path 2 (the green line) chooses the smoothest road traffic, resulting in the shortest travel time (198.02\( \text{s} \)) and the highest mean of velocity (40.00\( \text{km/h} \)). But it has the longest travel distance (2200\( \text{m} \)).

In the following, a real-world urban road network from Beijing is shown in Fig. 6. It is obtained from http://www.openstreetmap.org/. Fig. 7 illustrates the corresponding traffic network for the simulation. The traffic network of Beijing contains 115 nodes and 321 links. The coordinate of each node and the length of each link in the traffic networks are in accordance with real map data. The gray areas in Fig. 7 indicate the heavy traffic situations in these areas. Here the dark gray denotes the crowded traffic, and the light gray stands for the relatively crowded condition. The navigation results shown in Table 2 are very similar to that of the path 1, and the result of dynamic A* algorithm is same as the path 2. In conclusion, our algorithm not only can obtain an effective and compromised route by considering multiple traffic information to make navigation decision, but it can also select a route with the least travel distance or the least travel time similar to traditional navigation approaches.

<table>
<thead>
<tr>
<th>Paths</th>
<th>Requirements</th>
<th>TD (m)</th>
<th>TT (s)</th>
<th>MV (km/h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Travel Distance (as Dijkstra algorithm [1])</td>
<td>1400.00</td>
<td>282.11</td>
<td>17.87</td>
</tr>
<tr>
<td>2</td>
<td>Travel Time (as dynamic A* algorithm [6])</td>
<td>2200.00</td>
<td>198.02</td>
<td>40.00</td>
</tr>
<tr>
<td>3</td>
<td>Lane Status</td>
<td>1600.00</td>
<td>318.38</td>
<td>18.09</td>
</tr>
<tr>
<td>4</td>
<td>Multi-attribute</td>
<td>1800.00</td>
<td>213.27</td>
<td>30.38</td>
</tr>
</tbody>
</table>

Table I. Statistics of Different Navigation Requirements on 8 by 8 Grid Network

In the following, a real-world urban road network from Beijing shown in Fig. 6 is employed in the simulation. The coordinate of each node and the length of each link in the traffic networks are in accordance with real map data. The gray areas in Fig. 7 indicate the heavy traffic situations in these areas. Here the dark gray denotes the crowded traffic, and the light gray stands for the relatively crowded condition. The navigation results shown in Table 2 are very similar to that of the path 1, and the result of dynamic A* algorithm is same as the path 2. In conclusion, our algorithm not only can obtain an effective and compromised route by considering multiple traffic information to make navigation decision, but it can also select a route with the least travel distance or the least travel time similar to traditional navigation approaches.

<table>
<thead>
<tr>
<th>Paths</th>
<th>Requirements</th>
<th>TD (m)</th>
<th>TT (s)</th>
<th>MV (km/h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Travel Distance (as Dijkstra algorithm [1])</td>
<td>663.53</td>
<td>116.20</td>
<td>20.56</td>
</tr>
<tr>
<td>2</td>
<td>Travel Time (as dynamic A* algorithm [6])</td>
<td>1015.69</td>
<td>91.42</td>
<td>40.00</td>
</tr>
<tr>
<td>3</td>
<td>Lane Status</td>
<td>1152.31</td>
<td>180.61</td>
<td>22.97</td>
</tr>
<tr>
<td>4</td>
<td>Multi-attribute</td>
<td>840.93</td>
<td>99.52</td>
<td>30.42</td>
</tr>
</tbody>
</table>

Table II. Statistics of Different Navigation Requirements on Beijing Road Network
This paper develops a novel WSN-based real-time vehicle navigation algorithm, in which the two-layer sensor network architecture is introduced, five kinds of traffic attributes are defined, and four phases of the proposed algorithm are discussed. Hybrid MADM method is presented for the real-time navigation decision making, where a new general distance metric is proposed for the processing of both exact and fuzzy traffic data. Based on multiple traffic information and navigation requirements including the least travel distance, the least travel time, and the best lane status, the proposed algorithm selects the most effective and comprised navigation route. Simulation results demonstrate the good performance of the proposed algorithm in the WSN-based real-time traffic networks.

V. CONCLUSIONS

ACKNOWLEDGMENT

Figure 6. Urban road network of Beijing.

Figure 7. Different navigation paths on Beijing road network.

REFERENCES


