Abstract

Travel planning is one of the most important applications for vehicular ad hoc network. The existing travel plan schemes often choose the shortest path to the destination by GPS navigation. However, the current traffic conditions are not considered to modify the trip route while driving. In fact, the traffic jams have increasingly threatened the city traffic so that the shortest path calculated by GPS navigation takes more time to the destination. A congestion adaptive travel planning scheme is presented to adjust the trip route according to the current traffic conditions, which improves the traffic efficiency.

Keywords: Congestion Adaptive, Travel Plan, VANET

1. Introduction

Vehicular ad hoc networks have been envisioned to be promising in road safety and many other commercial applications, such as the Intelligent Transportation System (ITS), one of most important applications of VANET, which has been deployed in U.S., Europe, and Asia. For example, with the ITS, an emergency warning can help drivers behind a crashed vehicle (or incident) to avoid multi-car collisions [1]. Besides, a vehicular network can be used to alert drivers to potential traffic jams, providing more convenience and efficiency.

As more and more vehicles are equipped with communication capabilities that allow for inter vehicle communication, large scale vehicular ad hoc networks are expected to be available in the near future. Equipped with navigation system and electronic map, vehicle nodes can make a travel route choice based on the information they hold. In the most cases, selecting the shortest path by the Dijkstra algorithm with the information of the city map is a direct way of making a travel plan decision. However, the selected shortest path is usually not the speediest one for a vehicle node which is eager to arrive at the destination as soon as possible. For instance, if a driver plans to catch his scheduled flight in 30 minutes, he has to search some potential routes to get the airport in the shortest time.

Making a travel plan or routing decision according to the statistic data is a former related work [2, 3]. In this method, a vehicle node should firstly save the data of the time consumption of each road segment it passed by on the map in different time slots of history, and then calculate the weighted average time for them. With the statistic record, traffic conditions for each road segment of the travel route can be roughly estimated. In this scheme, the driver can decide a travel route before he sets off. However, that may not be the real time one which he wants most. The statistical data may reflect valuable estimation of traffic condition, while this is not a real time scheme. For instance, it cannot match the driver’s demand to arrive at the airport in 30 minutes on a holiday because there are some difference between the special date and work day.

In this paper, a Congestion Adaptive Travel Plan (CATP) scheme is presented to make a reliable travel plan dynamically according to the real time information about the current traffic conditions. In CATP, the information about current traffic conditions is decided by the time driving through the road segment, which is delivered to other vehicles. The vehicles that will enter this road segment may adjust their older route according to the traffic condition about the segment so that they can reach their destination more quickly.

The paper is organized as follows: Section II discusses some related works. Section III gives the definitions and assumptions about network model. Section IV presents the detailed design of our travel plan scheme, followed by the evaluation of CATP in section V. Section VI concludes the paper.
2. Related Works

There have been a lot of research works on vehicular ad hoc networks. In this section, we just discuss some related works about the traffic monitoring and some other travel plan scheme.

Traffic navigation systems enable the effective delivery and presentation of fine-grained traffic information to vehicles and have thus created demand for improved traffic data collection. Conventional traffic monitoring relies on eyewitness reports, traffic cameras, and loop detectors to regulate the driving actions, which have been introduced in [5, 6].

Recently, people care more about the traffic efficiency that roads can provide; this is also the demand of the Intelligent Transportation System (ITS). Traffic flow is a valuable parameter which gets many focuses in past decades including the analysis, simulations, and experiments [1, 8, 9]. However, most of them only talk the traffic flow on the freeway cases.

In our previous work [3], we proposed a distributed real time based travel planning algorithm for VANET. In this method, a vehicle node should firstly collect the data of the time consumption of each road segment the vehicles passed by on the map in different time slots of history, and then calculate the weighted average time for them. With the statistic record, traffic conditions for each road segment of the travel route can be roughly estimated. Then the shortest path is calculated by Dijkstra algorithm. In this scheme, the driver can decide a travel route before he sets off. However, the traffic conditions change over time while the vehicle on their way to the destinations. In fact, the scheme in [3] is a static planning for the trip because it just compute the route before driving without considering the current traffic conditions while driving. So the travel plan made before the journey often leads to more time spent on some crowded road. In addition, if there is a traffic jam on some hot road, the information of the jammed road segment couldn’t be delivered to other vehicles. So the car starts to driving can’t get the real time data and it just calculates its trip route based on the history data, which may navigate it to the jammed road.

Moreover, many papers have talked the additional infrastructure for VANET. In [4, 5], Mohammad proposed task organizers should get information including time mean speed (TMS), space mean speed (SMS), traffic density, and vehicle miles of travel (VMT) and so on. After that they report those data to local traffic management centers in [4]. In [7], Henrik designed a merging assistance application for a freeway lane drop scenario that builds on a message exchange between the vehicles and infrastructure of roadside, which provides drivers with individual speed limits and merging positions for drivers. But that is not fit for the urban environment. In [10], authors examine the problem of sharing information on one directional lane in sparse situation and proposed the information sharing using bidirectional communication, to find the sophisticated route in traffic jams in faster way. In [11], a scheme named SOTIS system has been proposed. It stated that the information of the traffic condition can be carried by vehicles. Then, they send this message to other vehicles to tell them about the traffic condition on this road segment. However, it cannot deliver a real time information if carriers have been stopped by jams.

In our another previous work[12], we proposed a scheme of Flow-Based Travel plan of VANETs (FBTV), which calculates most time-saving routes for vehicles via the real time traffic information of each road segments, and the traffic condition can be classified by the vehicles flow that we collected. In FBTV, it sets up some static watching nodes on some busy road segments as the additional supported devices for VANET, which are used to collect information about the vehicle flow. Before setting off, the vehicle can make a rough route calculation. When the vehicle arrives at each intersection of the formerly decided route, it will receive a warning message from a static watching node we set for each traffic busy road segment. According to the message, the vehicle node can recalculate a route to replace this road segment if it is very busy at that time. However, FBTV needs road-side infrastructures such as the watching nodes deployed on the busy road segments with higher investment cost. Furthermore, traffic congestion may occur on the roads without the watching nodes caused by accidents and the vehicles couldn’t get the current information about these roads so that FBTV can’t make a new route for them.
3. Network Models

Before talking about our scheme, some hypotheses have to be made. For one thing, we suppose that
the number of the vehicles on the city roads keep constant within a reasonable short time, i.e., the
number of vehicles leaving the city can be balanced by vehicles joining in. For another, vehicles are
equipped with the GPS devices and the pre-loaded digital maps, which are popular in today’s cars. In
addition, there is no road-side infrastructure such as deployed in [7] or [12], the information about
traffic conditions is transferred by the vehicle nodes.

As in Fig. 1, the road intersection I is denoted by I_i and the road segment between intersection I_i and
intersection I_j is denoted by R_{ij}.

Definition 1: Neighbor vehicles. For vehicle A, its neighbor vehicles include all vehicles within its
transmission radius. The vehicle nodes discover its neighbor vehicles with beacon messages discussed
in [13]. As shown in Fig. 2, the neighbor vehicles of A are the vehicle node B, C, D and E.

Definition 2: Passing Time. Passing Time of road segment R_{ij} is the interval between the time when
a vehicle enter into the Intersection I_i and the time when it leaves from the Intersection I_j, which
includes the time spent for waiting signals. It could be denoted by PassT_{ij} and there is:

\[ PassT_{ij} = t_j - t_i \]

where \( t_i \) is the time that the vehicle enters into I_i and \( t_j \) is the time that the vehicle leaves
from I_j.

The vehicle keeps the record of Passing Time as a two-tuple denoted by \( \langle PassT_{ij}, t_j \rangle \),
where \( t_j \) is the timestamp of the record of Passing Time. When the vehicle drives though
Intersection I_i, it will broadcast the record of Passing Time to its neighbor vehicles. And the
neighbors spread the information over networks. The vehicle that will enter into road segment
R_{ij} can recalculate its route when it receives the Passing Time of R_{ij} and decides to choose
another route or do nothing.

Definition 3: Upper bound of Passing Time. The upper bound of Passing Time, denoted by T, is the
metric to determine there is traffic congestion on the specific road segment or no. If Passing Time of R_{ij}
is greater than T, it means there is traffic congestion on R_{ij} or there will be jammed. T is decided by the
design passing time, which is the length of $R_{ij}$ divided by the speed limit of $R_{ij}$, and the waiting time for signal light.

\[ T = \varepsilon DT + ST \]  

(2)

where $\varepsilon$ is a constant and $\varepsilon > 1$; DT is the design passing time; ST is the time for signal light.

**Definition 4:** Staying Time. Staying Time is the time spent on the road segment $R_{ij}$ before the vehicle leaves from $R_{ij}$. If there is no any congestion on the $R_{ij}$, Staying Time of $R_{ij}$ is equivalent to Passing Time of $R_{ij}$. Otherwise, Staying Time is often greater than Passing Time. When Staying Time is greater than $T$ at the moment $t$, then calculates Staying Time of $R_{ij}$ as (3).

\[ StayT_{ij} = t - t_i \]  

(3)

If there is heavy traffic jam on $R_{ij}$, the vehicle has to stay on the road and broadcasts its record of Staying Time periodically. The record of Staying Time is also a two-tuple such as $<StayT_{ij}, t>$ and the vehicles on other road segments could adjust their new route according these records.

4. The Design of CATP

The records of Passing Time or Staying Time will assist vehicle nodes to estimate the traffic condition of the road. While in our model, the real time condition can be known to help vehicles to recalculate a new travel route with lowest time consumption. This is an iterative process. Before setting off, vehicle can calculate a shortest path with the geography information of the city or the statistic records of the history data. It can travel along this route because it is the optimization choice at the beginning. This travel plan can be continued if it never encounters the traffic jams or a road which is too busy to support more vehicles. During the rush hour, however, the vehicle will meet with traffic jams with a high probability, especially on a holiday. In CATP, traffic congestion of a road segment $R_{ij}$ can be estimated by Mean Passing Time (MPT) of $R_{ij}$ or Mean Staying Time (MST) of $R_{ij}$. Once MPT or MST of $R_{ij}$ exceeds the upper bound of Passing Time of $R_{ij}$, the warning message will be broadcasted. The vehicle could recalculate its trip route once it receives the warning message according to the MPT or MST of $R_{ij}$. We will discuss how to get MPT and MST, and present the details of CATP later.

4.1. Mean Passing Time

When a vehicle passes through the Intersection $I_j$, it broadcasts its record of Passing Time of $R_{ij}$, $<PassT_{ijk}, t_j>$. Assuming that vehicle $A$ which will driving through $R_{ij}$ receives $n$ records of Passing Time of $R_{ij}$ from $n$ vehicles, it computes the MPT of $R_{ij}$ as follows:

\[ MPT_{ij} = \frac{\sum_{k=1}^{n} PassT_{ijk} \cdot t_j}{\sum_{k=1}^{n} t_j} \]  

(4)

where $PassT_{ijk}$ and $t_j$ denote Passing Time of $R_{ij}$ and the time when leaving $I_j$ of the $k$th vehicle, respectively. As in (4), the timestamp $t_j$ is the weight of Passing Time. The larger $t_j$, the more updated record of Passing Time of $R_{ij}$ is. So the latest record $<PassT_{ij}, t>$ contributes more to $MPT_{ij}$, while considering the previous records of Passing Time.

However, the computation of $MPT_{ij}$ requires the vehicle to store $n$ records of Passing Time of $R_{ij}$ from $n$ vehicles. To simplicity, a vehicle can calculate the $MPT_{ij}$ of $R_{ij}$ immediately according to the older $MPT_{ij}$ and the newest record $<PassT_{ij}, t>$ once it gets an update record. For example, when vehicle $A$ get a record $<PassT_{ij,n}, t_j>$ from the $n$th vehicles passing through road $R_{ij}$, it updates the $MPT_{ij}$ of $R_{ij}$ as (5):
\[ MPT_{ij}^n = \frac{(MPT_{ij}^{n-1} \cdot SumT_{ij}^{n-1} + PassT_{ijn} \cdot t_{jn})}{(SumT_{ij}^{n} + t_{jn})} \]  

(5)

where \( MPT_{ij}^{n-1} \) denotes the history \( MPT \) of \( R_{ij} \), \( SumT_{ij}^{n-1} \) is the sum of timestamp \( t_j \) from the previous \( n-1 \) records and it is updated by \( SumT_{ij}^n = SumT_{ij}^{n-1} + t_{jn} \).

Obviously, if there is heavy traffic congestion on the road segment \( R_{ij} \), there is no any vehicle passing through the Intersection \( I_j \) so that there is no latest record of Passing Time of \( R_{ij} \) and other vehicles will enter into road segment \( R_{ij} \) can’t know the traffic jam on the road \( R_{ij} \). At this moment, the records of Staying Time should be delivered by the vehicles on the road \( R_{ij} \). Other vehicles calculate the Mean Staying Time of \( R_{ij} \) to determine whether change their trip route or not just the same as the Mean Passing Time of \( R_{ij} \).

4.2. Mean Staying Time

The calculation of \( MST \) is similar to \( MPT \). Noticeably, the broadcasting of the record of Staying Time only occurs when Staying Time of a vehicle exceeds the upper bound of Passing Time \( T \). If the traffic flow of \( R_{ij} \) is normal, the Staying Time on the road is obviously less than \( T \) and the record of Staying Time of \( R_{ij} \) will not be delivered. On the contrary, the road \( R_{ij} \) is considered to be jammed and the record \( \langle StayT_{ij}, t \rangle \) should be scattered over the networks. The record \( \langle StayT_{ij}, t \rangle \) will be broadcasted periodically. The more \( MST \) of \( R_{ij} \), the heavier the traffic congestion on road \( R_{ij} \) is. Once other vehicles receive the record \( \langle StayT_{ij}, t \rangle \), they update \( MST_{ij} \) as (6):

\[ MST_{ij}^n = \frac{(MST_{ij}^{n-1} \cdot SumT_{ij}^{n-1} + StayT_{ijn} \cdot t_{jn})}{(SumT_{ij}^{n} + t_{jn})} \]  

(6)

4.3. Algorithm of CATP

As described before, the \( MPT_{ij} \) or \( MST_{ij} \) of road segment \( R_{ij} \) is used to decide whether there is a potential traffic jam or not. To each road segment on the map, there is an upper bound of Passing Time \( T_{ij} \) which calculated by the length of road segment \( R_{ij} \) divided by the speed limit of road segment \( R_{ij} \). If a vehicle enters into road segment \( R_{ij} \), it starts a timer to record its passing time or staying time. Once the vehicle’s passing time or staying time is greater than \( T_{ij} \), there may be potential traffic congestion and it broadcast a warning message containing the record of Passing Time or Staying Time of road segment \( R_{ij} \). After receiving the warning message, the vehicle computes the the \( MPT_{ij} \) or \( MST_{ij} \) as equation (5) or (6), which is considered as the weight of road segment \( R_{ij} \). Then it recalculates the new trip route by Dijkstra algorithm and makes a decision to bypass the potential jammed road segment or do nothing.

For example, as shown in Fig. 1, vehicle A’s destination is D and it acquires the optimal trip route according to GPS navigator based on the statistic data about the map or the history \( MPT_{ij} \) in which road \( R_{ij} \) is one of the segment on the route. While vehicle A is on its way to the destination D, A will receive the record of the Passing Time or the Staying Time of \( R_{ij} \) from vehicles passing through road segment \( R_{ij} \). Then vehicle A computes \( MPT_{ij} \) or \( MST_{ij} \) of \( R_{ij} \) and recalculate the route to the destination D based on the real-time traffic conditions. If the new route is better than the former route, vehicle A will choose the better route. Obviously, if the traffic is light, there is no traffic jam on any road segment and it doesn’t need to change the former route. If the traffic flow on road segment \( R_{ij} \) gets heavier and heavier, leading to the Passing Time and Staying Time of \( R_{ij} \) longer and longer. In the worst case, the Staying Time of \( R_{ij} \) exceeds \( T_{ij} \) firstly and the warning message is delivered. Of cause, the Passing Time of \( R_{ij} \) is greater than \( T_{ij} \) if the vehicle passed through the intersection, too. At this moment, the recomputed new route may spend less time than former route. As in Fig. 1, vehicle A will bypass road segment \( R_{ij} \) and select a new route to the destination D.

The algorithm of CATP is as follows:
Enter Intersection $I_j$:
Record $t_i$
$t = \text{current time}$
While stay in $R_{ij}$ if vehicle stays in road segment $R_{ij}$
if $(t - t_i) > T$ // Staying Time is greater than T
  Broadcasting $\langle \text{StayT}_{ij}, t \rangle$ // broadcasting warning message
  Get $t$ periodically
End if
If receive $\langle \text{PassT}_{ma}, t \rangle$ or $\langle \text{StayT}_{ma}, t \rangle$ // received the record of
  // $\langle \text{PassT}_{ma}, t \rangle$ or $\langle \text{StayT}_{ma}, t \rangle$
  // from other road segment $R_{ma}$
  Calculate $\text{MPT}_{ij}$ or $\text{MST}_{ij}$ // updating $\text{MPT}_{ij}$ or $\text{MST}_{ij}$
  $\text{Weight}_{ij} = \max(\text{MPT}_{ij}, \text{MST}_{ij})$ // the larger as the weight
  // of road segment $R_{ij}$
  Recalculate driving route by Dijkstra algorithm
  make decision
End if
End while
Leave the Intersection $I_j$
Record $t_f$
$\text{PassT}_{ij} = t_f - t_i$
Broadcasting $\langle \text{PassT}_{ij}, t \rangle$

As in Fig. 3, the record of Passing Time $\langle \text{PassT}_{ij}, t \rangle$ is broadcasted when the vehicle leaves from the Intersection $I_j$. However, the warning message containing the record of Staying Time $\langle \text{StayT}_{ij}, t \rangle$ is just broadcasted when the Staying Time exceeds $T$. Vehicles may receive two or more records. If a vehicle receives the record of Passing Time and Staying Time of $R_{ij}$ from the same vehicle at the same time, it only computes the $\text{MPT}_{ij}$. Otherwise, it updates $\text{MPT}_{ij}$ and $\text{MST}_{ij}$ simultaneously, the larger of which is used as the weight of road segment $R_{ij}$ to recalculate the new route.

5. Performance Evaluation

We choose VanetMobiSim+NS2 to evaluate the performance of CATP, in which VanetMobiSim is used to produce the motion model of the vehicles. The FBTV and the shorts path scheme are simulated to compare with the CATP.

The experiment is based on the portion of the map of Chengdu metropolitan area, the biggest city in the western of China, which covers 4500m×3500m street area, including 36 intersections and 62 road segments. The snap topology of the map is shown in Fig. 4.

![Figure 4. The topology of the map](image)
The speed of vehicle is differentiated from 40km/h ~80km/h and is set before the vehicle enters into the road segment, which is restricted by the speed limit of the road segment. MAC protocol of the vehicles uses 802.11 and the transmission radius is set to 200m. In order to simulate the different traffic conditions of the city transportation system, we design 6 kinds of density of vehicle in our experiment, which listed in Table 1. The vehicle nodes have a random distribution initially, and they try to arrive at their destination with a shortest path which is calculated by the Dijkstra algorithm at first. Other simulation parameters are listed in Table 1.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulation area</td>
<td>4500m×3500m</td>
</tr>
<tr>
<td>Intersections</td>
<td>36</td>
</tr>
<tr>
<td>Numbers of road segments</td>
<td>62</td>
</tr>
<tr>
<td>Number of vehicles</td>
<td>100,200,400,600,800,1000</td>
</tr>
<tr>
<td>Transmission radius</td>
<td>200m</td>
</tr>
<tr>
<td>Speed of vehicles</td>
<td>40km/h-80km/h</td>
</tr>
<tr>
<td>MAC protocol</td>
<td>802.11</td>
</tr>
<tr>
<td>Waiting time for traffic lights</td>
<td>30seconds</td>
</tr>
<tr>
<td>ε</td>
<td>1-5</td>
</tr>
<tr>
<td>Watch nodes</td>
<td>30 (only for FBTV)</td>
</tr>
</tbody>
</table>

### 5.1. The influence of \( \varepsilon \)

The value of \( \varepsilon \) is vital for determining whether there is traffic jam on the road or not. In simulation, we set \( \varepsilon \) from 1 to 5 and record the average driving time of the five vehicles from their departure to the destination, respectively. The results are shown in Fig. 5.

Clearly, when \( \varepsilon \) is small, the upper bound of Passing Time \( T \) is small, too. The staying time of a vehicle is easier to exceed \( T \) so that the warning message will be broadcasted. Some vehicles would change their route according to the message. Unfortunately, it is a false warning message for that \( T \) is too small. Actually, there isn’t any traffic jam on the road. The false warning message leads to more time spent for driving by changing the former route. On the other hand, if \( \varepsilon \) is set to a large value, \( T \) will be very large so that any warning message wouldn’t be delivered even though there is traffic congestion on the road. In this scenario, the vehicles don’t recalculate the new route to the destination and will be navigated by the former route with the jammed road segment, leading to more driving time for that the vehicles spent more time driving through the jammed road segment. As shown in Fig. 5, the average driving time is smaller when \( \varepsilon \) is from 2~2.5. When \( \varepsilon \) is less than 2, the average driving time increases because the route is adjusted frequently caused by the false warning message. When \( \varepsilon \) is greater than 3, the average driving time increases, too. In the following simulations, we set \( \varepsilon \) to 2.2.

![Figure 5. The influence of \( \varepsilon \)](image-url)
5.2. Average driving time

In this section, we compare the average driving time consumption of the CATP with the FBTV and the shortest path under different density of vehicle nodes. We track 5 vehicle nodes and record their average driving time to their destination, respectively. Fig. 6 shows the results. It is can be seen from Fig. 6 that there is little difference among the three schemes when the density of vehicle nodes is small, i.e., the number of vehicle nodes is from 100-200. This because the traffic is light when the density of vehicle nodes is small and the probability of traffic congestion is small either. Accordingly, it doesn’t need to change the driving route for that the running of FBTV or CATP is equivalent to that of the shortest-path scheme in this case.

![Figure 6. Average driving time under different density of vehicles](image)

However, with the increasing density of vehicle nodes, traffic congestion would occur on some road segment. To the shortest-path scheme, the route can’t be renewed anymore after it was calculated before the departure. The average driving time increases because vehicles have to passing through the jammed route. But for FBTV and CATP, the route could be adjusted by the information about traffic congestion on the specific road segment according to the mean passing time of that road segment, leading to less driving time than the shortest-path scheme. Moreover, the average driving time of FBTV may be less than CATP. The reason is that FBTV can decide traffic jam by road-side infrastructure clearly, which is more accurate than CATP which there may be some false jam warning.

If the density of vehicle nodes is very high, i.e., the number of vehicle nodes reach to 1000, the advantage of CATP and FBTV is disappeared and even worse than the shortest-path scheme. That is because there are too many roads were trapped by traffic jams and vehicle nodes will adjust their travel route frequently, spending more time for driving.

6. Conclusions

In this paper, we proposed a scheme of travel plan named CATP, for urban area vehicles. With CATP, vehicle nodes could recalculate their travel route according the information about the traffic jams on the road segment, which decided by the mean passing time or mean staying time of the road segment. Simulation results show that CATP performs better than the shortest-path scheme when the density of vehicles is medium, while there is no any road-side infrastructure compared to FBTV.

7. Acknowledgement

This work is supported by National Science Foundation under grant number 60903158, 61170256, 61173172, 61103226 and the Fundamental Research Funds for the Central Universities under grant number ZYGX2010J074 and ZYGX2011J102.
8. References


