Peer-to-Peer Cooperative Driving

Alina Bejan
Computer Science Department
University of Iowa
Iowa City, USA
abejan@cs.uiowa.edu

Ramon Lawrence
IDEA Lab
University of Iowa
Iowa City, USA
rlawrenc@cs.uiowa.edu

Abstract

In this paper we address database-related issues in the emerging application field of ITS (Intelligent Transportation Systems). In this context we propose and study two cooperative driving scenarios: on-the-fly highway alert scenario and mutual driving group scenario. Vehicles cooperate and coordinate their actions by exchanging information, hence the need for database technologies such as consistency, replication, and query optimization. Technical requirements and suitability of technologies such as wireless and peer-to-peer communication and mobile ad-hoc networks are discussed. We introduce the notion of a peer-dependent query for the application environment.

1 Introduction

The area of ITS, *Intelligent Transportation Systems*, is a fairly new concept which involves an advanced information and telecommunications network for users, roads, and vehicles [1, 12]. Problems of concern addressed by ITS are road safety, detection and avoidance of traffic accidents or traffic congestion, and safe driving assistance. One direction associated with ITS is cooperative driving, a paradigm involving inter-vehicle communication in which road participants exchange information in order to coordinate and support some of their actions. The existence of ITS is due to developments in several technological areas: navigation systems, electronic toll collection systems (ETC), telecommunications, and wireless communication.

Vehicle-to-vehicle communications in an ITS demonstrates properties of both peer-to-peer (P2P) [6] networks and mobile ad-hoc networks [7, 10]. In peer-to-peer (P2P) [6] systems, participants rely on one another for service, rather than solely relying on a dedicated and centralized infrastructure. Peers in the system both provide and consume services. A mobile ad-hoc network is a collection of mobile hosts with wireless communication capabilities, forming a temporary network without aid of any established infrastructure [7]. In such networks, topological connectivity is subject to frequent, unpredictable change. Given the mobility of vehicles on the road, it is obvious that an ITS network demonstrates properties of both network types. Further, in an ad-hoc network two hosts that want to communicate may not be within wireless range of each other, but could still communicate if other hosts between them are also participating in the ad-hoc network and are willing to forward messages for them. We exploit already existing routing algorithms for inter-vehicle communication.

Our contribution consists in studying how peer-to-peer communication in a mobile ad-hoc network would serve as an architecture for two new cooperative driving scenarios, namely on-the-fly highway alert scenario and mutual driving group scenario, and also demonstrate the role for database management in such an architecture. To the best of our knowledge this is the first comprehensive examination of ITS from a database perspective, even though there exists research on component topics of mobile databases [2] and geographic, network routing [5], and P2P [4]. We demonstrate how ITS merges problems from these diverse areas, and give a practical architecture, based on previous work in these fields.

The paper is structured in five sections as follows. **Section 2** describes the two cooperative driving scenarios, and **Section 3** presents the architecture while addressing the technical and networking issues. Database techniques applicable to our scenarios are discussed in detail in **Section 4**, including materialization of views for query answering as well as location and peer dependent queries. Our concluding remarks and future research directions are summarized in **Section 5**.

2 Problem Description

Before we describe our scenarios, we briefly present the basic concept of ITS communications. There are two basic types of ITS (see Figure 1 [8]):

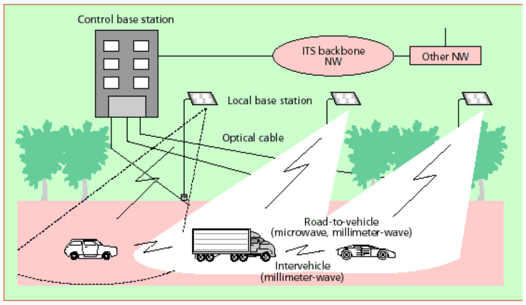

Figure 1. The basic concept of ITS communications

- road-to-vehicle communications
- vehicle-to-vehicle communications

The road-vehicle communication involves base stations located along the road that communicate with the vehicles on the road and are coordinated by several control stations. An example is the warning highway system that warns about congestion or accidents and suggests alternative route(s). Inter-vehicle communication has not been explored in as much detail.

In this paper we present two application scenarios of an ITS, in which vehicles can cooperate and coordinate their actions, and enumerate the database issues involved. In order for a set of vehicles to be engaged in a *cooperative driving* scenario, they have to be able to exchange information about each other. Cooperative driving has been briefly introduced in the literature [3], and we extend the discussion to two different working scenarios and study the database aspects. To this point, ITS technology has been exclusively addressed in the domain of networking and private corporations. However, as the functionality of these systems progresses, the need for database technologies such as consistency, replication, and query optimization will have an increasing role. A description of the two cooperative driving scenarios follows.

In the *on-the-fly highway alert* scenario, vehicles transmit to and receive from other participating vehicles warning messages regarding the state of the road (accidents, traffic congestion, closed roads). Although this service is available in a centralized broadcast fashion, it can be enhanced by allowing the vehicles themselves to be involved in detecting and propagating the road status information. This can be useful when a centralized dispatcher is not available for that specific road segment, or it can be used to improve the accuracy of the centralized information (if available). For this scenario to function properly, a larger number of vehicles need to participate in detecting and passing along the information to upcoming vehicles. Otherwise the information might be meaningless or not trustworthy. Each vehicle uses this information to decide an alternative route or make different local decisions (depending on the warning type).

The second scenario is a *mutual driving group* of vehicles that plan a trip together. They leave together and plan to reach the same destination. The goal is that vehicles stay in communication as much as possible throughout the entire trip. They may need to communicate in order to make common decisions (such as dynamically deciding which exit to take). Clearly, it is not always possible that vehicles stay in visual contact, and communication disconnections are quite possible. Unlike other approaches [3] that perform real-time cooperative driving, we will allow vehicles to lose communication contact for periods of time, and still be able to track each other. We would like each vehicle of the group to have sufficient information about the rest of the group, at any given time, and also be able to foresee when they will be within communication range again.

3 An ITS Architecture

This section presents the technical requirements for the cooperative driving scenario, summarizes the technologies involved, and enumerates the network and database issues.

3.1 Assumptions

Location information is essential in the scenarios described. Therefore, each vehicle is equipped with a GPS (Global Positioning System) receiver and a corresponding off-line map. For our purpose, the GPS receiver will provide the following functionalities:

- provide location coordinates (x, y) at time t,
- ability to locate (x, y) on a static map stored in the device memory,
- ability to plot a path on the map given the table (x, y, t), and ability to plot multiple paths[1] in a similar form.

Each participating vehicle is equipped with a wireless communication module (transceiver), that has a limited broadcast range R. Initially, assume all have the same range. Each vehicle has a unique identification number *vid* based on its transceiver, for use in communication and tracking. Though security issues are not addressed in this paper, the vehicle vid may be encoded using encryption techniques during communication to avoid the effects of potential interference of malicious participants. Our assumption is that all the vehicles are trustworthy.

[1]Tracking multiple vehicle routes on the same map is not currently a standard feature of GPS receivers.

3.2 An Ad Hoc Mobile Network

Using the wireless transceivers, vehicles form a *wireless network* given a suitable network protocol. Due to the highly dynamic nature of the environment, vehicles join and leave the network frequently. Therefore, the network is a *mobile ad-hoc network* [7], where hosts travel through physical space and communicate in an opportunistic manner via wireless links. Further, since the communication and control are decentralized, the network can be viewed as a *peer-to-peer network*. Decentralization offers scalability, robustness, and limits requirements for central administration.

The network is characterized by wireless communications, no central control, and cooperation among nodes. These features form a union of P2P, wireless, and ad-hoc networks, therefore our network architecture will be a *peer-to-peer, mobile, ad-hoc, wireless network*.

3.3 Network and Routing Issues

In this section we will look at membership, discovery and routing issues in the context of cooperative driving. These topics have been well studied in the context of system networks. Algorithms exist for routing in ad-hoc networks [7], and group membership management and discovery in peer-to-peer networks [10]. We will be using these results in designing a solution for the cooperative driving problem.

Vehicles periodically broadcast messages containing their location and time information. This will insure basic network connectivity. Each vehicle manages a *routing table*, needed to store information about neighboring vehicles (participating vehicles within its range). Updates in the routing table are triggered by the receipt of a broadcast message. A broadcast message is a quadtuple (vid, x, y, t), where *vid* is the vehicle identification number, (x, y) is vehicle's current GPS coordinates, and t is the value of the local clock. Note that GPS tells extremely accurately time, so clock synchronization is not an issue.

GPS-enabled vehicles allow for the development of useful location dependent services, such as navigation and automated road assistance. Integrating GPS data into network protocols, such as the Internet protocol, to create such services has been studied in [5]. Addressing based on physical location of nodes instead of a logical address is called *geographic addressing*, and corresponding *GPS-based routing protocols* have been proposed [5]. We make use of existing geographic and ad-hoc routing algorithms to solve the addressing and routing mechanisms for our scenarios.

As with any P2P architecture the system's robustness, availability, performance and accuracy depend on the number of peers: the more participating peers, the more available and accurate the information. Vehicles would not benefit much if the network size is small and if new members join and leave frequently.

3.4 Database Issues

One important aspect in our application is the availability of a *consistent view* of the system to all of the network members. It is not trivial to compute the global network image due to the highly dynamic behavior of the network which may induce *stale* data in the routing tables. The routing table is in fact a database relation that stores information about the neighboring vehicles. This relation can be seen as a view of a distributed database, since all the members of the ad-hoc network contribute information. The entire configuration of the network is most likely not known by each individual node, although this is desirable. Instead, each node has some local information regarding its neighbors. Putting all local information together creates the global view of the system.

A vehicle with a GPS and no transceiver is limited on the types of queries that can be performed. Data from the GPS can be used for static mapping and location-dependent queries. Some examples of such queries are: "*Where am I?*", "*Where is the closest gas station?*", "*How far is the closest restaurant?*". However, by introducing a mechanism for communication between GPS-equipped vehicles, database issues of distribution and consistency arise. From a global-level view, at any point in time, there exists a relation $L(vid, x, y, t)$ storing the current location of all vehicles and the corresponding time information. Further, a global-level historical relation $H(vid, x, y, t)$ stores locations of all vehicles for all times. Each vehicle i has its own local views L_i and H_i of the global relations. The views of both relations are continually dynamic, especially L, whereas H may contain incomplete or inconsistent information. We highlight that views are in fact constructed based upon the information exchanged by the peers and not upon information provided by a central unit. History table management is discussed in detail in Section 4, and Figure 2 provides the system infrastructure.

4 Database Issues in Cooperative Driving

Each time a vehicle receives a message, it accordingly updates the two tables: the tuple corresponding to the vehicle whose vid is in the message received will be updated in both L_i and H_i. On an update from a broadcast, the history table represented by relation $H(vid, x, y, t)$ is updated by inserting the current tuple (vid, x, y, t) in H_i. The routing table is managed by the underlying network protocol.

Important issues are dealing with disconnections, stale data, and garbage collection, which are discussed in the next section.

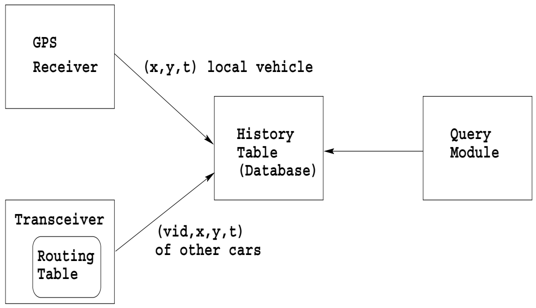

Figure 2. Infrastructure

4.1 View Materialization for Query Answering

In order to compute the global view of the network, individual vehicles need to query other members to gather information about non-neighboring nodes. Dynamic routing protocols are used to re-materialized a view.

The history table is used to store information about all the neighbors throughout the journey. The database queries will be performed on this table. We consider two possible cases: when a node receives the query "*Where is X?*", it can either answer immediately with whatever information that node has by inspecting its history table, or it can itself initiate a look-up process to gather information from its own neighbors. Since the sets of neighbors of these two nodes can be distinct, more valuable information may be collected. A mechanism is used to detect query replication. Suppose vehicle i sends this query to vehicle j. Then j will perform a query like *SELECT * FROM H_j WHERE (SELECT MAX(t) FROM H_j WHERE vid = X) = t* to retrieve the most current record for X. This query is used in constructing and updating the history table, since they are used to provide up-to-date information when *stale* data is detected.

One valid question is how far should the query process go. The number of steps should be finite and also the time allotted to this operation should not exceed a certain threshold, since in that case the answer might be already outdated by the time it reaches the initiator (the queried vehicle had changed its position considerably, making the information useless). To improve efficiency, when a vehicle replies to a query it appends its own current information to the reply. This allows the query initiator to update two tuples when receiving the answer: the one corresponding to the queried vehicle and the one corresponding to the answering neighbor.

In addition to storing time information, derived information such as velocity vector are also useful. By using interpolation techniques, this would allow us to make location-time predictions or answer queries corresponding to time values for which no records exist in H. Efficient query answering can be achieved using known indexing protocols [9, 11].

4.2 Location Dependent Queries

Queries can be classified into *GPS specific queries* and *location dependent queries* in the context of cooperative driving. Examples of GPS specific queries, that can be answered without peer interaction include make use of base data (current location, speed, time), or derived data (trip time, average speed, estimated arrival time). These queries are simple in nature as they can be answered by applying mathematical formulas to the stored data in the GPS receiver in addition to information from the history. Database techniques are not required.

In addition to these queries *peer-dependent queries* require peer knowledge to be answered, such as:

- *Who are my neighbors?*
- *Where is vehicle X?*
- *Are all the mutual group vehicles within my range?*[2].

Peer-dependent queries can always be "trivially" answered by querying over the history table (relation H). To answer "*Where is vehicle X?*", a vehicle i could check its history table. If it finds the corresponding entry there, and if the tuple's timestamp is *recent enough*[3], then the information is considered valid. If the information is *stale*[4], then a look-up procedure is performed. The protocol for detecting the position of vehicle X is presented in Figure 3.

The answer to the query "*Who are my neighbors?*" can be easily obtained by scanning the routing table. If it is essential that the information is very accurate, a record may be checked for stale data, and appropriate actions should be taken if this is detected. By inputing historical relation H_i into the GPS receiver, one vehicle can monitor the route progress of a predefined set of vehicles.

4.3 Problem Alert Queries

Alert queries involve passing warning messages between vehicles. Instead of drivers initiating and sending warning messages directly, given each vehicle's local view, data mining can be performed on the historical relation H to decide if the traffic ahead has significantly changed in speed

[2] Of course this query is a combination of the previous two, because one can check the neighbors list and look for certain vehicle vids.

[3] It means that the difference between i's clock and the tuple timestamp is less than a predefine value $tmax$.

[4] It means that the difference between i's clock and the tuple timestamp is greater than $tmax$.

Protocol for vehicle i to detect vehicle X

```
{ i first checks its own table H_i, then broadcasts }
N_i := { i's set of neighbors };
rec = SELECT * FROM H_i WHERE
      (SELECT MAX(t) FROM H_j WHERE vid = X) = t;
if rec = ∅ then rec := (X, 0, 0, 0); endif;
send "Where is X?" to N_i (broadcast);
receive { rec_j | j in N_i };
for each rec_j do
      if rec_j.t > rec.t then
            H_i := H_i ∪ (X, x, y, t);
      endif
      H_i := H_i ∪ (j, x_j, y_j, t_j);
enddo;
```

Figure 3. Protocol to answer "Where is X?"

(and maybe direction). A notion of *progress* can be established. Progress occurs when vehicles maintain roughly the same speed and direction. Any changes in this setting signals a potentially wrong event and an alert message is broadcast. The neighbors receive the message and retransmit it further to their neighbors (geographic routing is involved), thus alerting the upcoming vehicles of potential problems ahead. Drivers can then become more cautious, adjust their speed, and make route changing decisions ahead of time if necessary. Such capability contributes to a safer driving environment.

Note that the message propagation wave can reach vehicles located very remotely from the actual incident, and it is possible that these will not be at all affected by a sudden traffic slow down. The vehicles can make local decisions with respect to the importance of the notice, by considering the location, time, gravity of the incident from the broadcast message and their own time, location, and direction of movement.

5 Conclusion and Future Work

In this paper we have studied how protocols that deal with routing in mobile ad-hoc networks and routing based on geographical location can be used in cooperative driving scenarios, where road participants become engaged in offering and consuming services while driving. The environment is a totally decentralized architecture, based on peer-to-peer interaction. The presented scenarios functionality relies entirely on the willingness of the vehicles to cooperate, and on the quality of the information exchanged. In addition to enumerating the issues in the emerging application area, we characterized some of the database issues involved. By moving from static, single vehicle queries to dynamic, multi-vehicle communications, the need for database technologies increases. We also discussed database issues related to history table management, namely constructing materialized views of the system, and how this information can be used in answering location and peer dependent queries. A protocol for answering basic queries by dynamic view materialization using existing network protocols has been proposed.

Our future work will focus on adding security constraints to our scenarios and by optimizing the management of the history table. This table can grow considerably and techniques for reducing its size are important.

References

[1] T. Brinkhoff. Requirements of traffic telematics to spatial databases. In *Proceedings of SSD99*, volume 1651, pages 365–369, 1999.

[2] M. Choy, M.-P. Kwan, and H. Leong. Distributed database design for mobile geographical applications. *Journal of Database Management*.

[3] W. Franz, H. Hartenstein, and B. Bochow. *Internet on the Road via Inter-Vehicle Communications*. Workshop der Informatik 2001: Mobile Communications over Wireless LAN: Research and Applications, 2001.

[4] S. Gribble, A. Halevy, Z. Ives, M. Rodrig, and D. Suciu. What can databases do for peer-to-peer? In *Proceedings of WebDB2001*, 2001.

[5] T. Imielinski and C. Navas. Geographic addressing, routing, and resource discovery with the global positioning system. *Communications of the ACM Journal (CACM)*, 1997.

[6] M. Naraghi-Pour, M. Hegde, and R. Pallapotu. Peer-to-peer communication in wireless local area networks. In *Proceedings of the ICCCN98*, pages 432–437, 1998.

[7] S. Nesargi and R. Prakash. Issues pertaining to service discovery in mobile ad hoc networks. In *Proceedings of POMC2001*, pages 41–48, 2001.

[8] S. Ohmori, Y. Yamao, and N. Nakajima. The future generations of mobile communications based on broadband access technologies. *IEEE Communications*, December 2000.

[9] D. Pfoser, C. Jensen, and Y. Theodoridis. Novel approaches to the indexing of moving object trajectories. In *Proceedings of VLDB2000*, 2000.

[10] G.-C. Roman, Q. Huang, and A. Hazemi. Consistent group membership in ad hoc networks. In *Proceedings of the 23rd International Conference in Software Engineering (ISCE)*, 2001.

[11] S. Saltenis, C. Jensen, S. Leutenegger, and M. Lopez. Indexing the positions of continuously moving objects. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, pages 331–342, 2000.

[12] Y. Shiraki, T. Ohyama, S. Nakabayashi, and K. Tokuda. Development of an inter-vehicle communications system. *OKI Technical Review*, 68(3), September 2001. Special Edition on ITS of OKI Electric Industry Co., Ltd.