TECHNICAL REPORT 02-04

Peer-to-Peer Cooperative Driving

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

Ramon Lawrence
Computer Science Department
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, 34]. 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 [10, 14]. 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 demonstrate properties of both peer-to-peer (P2P) [20] networks and mobile ad-hoc networks [22]. In peer-to-peer (P2P) 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 ah-hoc network is a collection of mobile hosts with wireless communication capabilities, forming a temporary network without aid of any established infrastructure [22]. 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

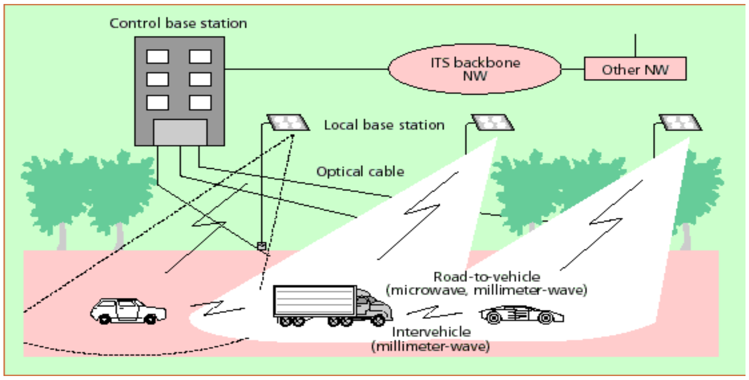

Figure 1: The basic concept of ITS communications

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 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 [3, 7, 9, 24, 25]. 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 1** introduces the general framework, **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 presented in **Section 5**.

2 Problem Description

Before we describe our scenarios, we briefly present the basic concept of intelligent transportation system (ITS) communications. There are two basic types of ITS (see Figure 1 [23]):

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

The road-vehicle communication involves several 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). In the road-assistance applications and services already existing, a person (operator) assists the traveler. Similar scenarios involving agents that offer support to mobile users have been described in [30] and [29]. Mobility offers a high degree of flexibility, but also requires new location-based technologies and assistance services. Agent-based solutions represent one approach for the mobile environment. In the future, optical fiber networks will be implemented along the roadside to convey high speed data to access points, which will connect fiber and wireless links for moving vehicles. Radio on fiber is the future trend in ITS.

The other mode of communication involving inter-vehicle communication has not been explored in as much detail. We are interested in vehicle-to-vehicle communication. In this paper we present two application scenarios of an intelligent transportation system, 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 [5, 33, 34, 36], and we extend the discussion to working scenarios and study the related databases 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. To our knowledge, this is the first comprehensive examination of ITS from a database perspective. We first describe the two cooperative driving scenarios that we considered.

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 considerable 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 to make local decisions (depending on the warning type). We opt for a totally decentralized solution since it offers flexibility and is suitable for our dynamic scenario.

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 [5] 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 meeting again (by *meeting* we mean that vehicles are within communication range). Again, we do not rely on a central point of control, the solution being fully decentralized. This allows group members to equally contribute to the scenario by communicating and exchanging information with each other.

3 An ITS Architecture

In this section we present the technical requirements for the cooperative driving scenario and summarize the technologies involved. We will enumerate the network, security and database issues in this environment.

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. Due to the rapid developments in mobile technologies and the need for devices that support mobility [4], there exists many different GPS receivers [6] on the market, and all provide different functionalities. The more sophisticated the GPS, the more services and facilities are available to the user. For our purpose, the GPS receiver will provide the following functionalities:

- 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
- the ability to plot multiple paths of the form given above.

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. Our assumption is that all the vehicles are trustworthy.

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 an *ad-hoc network* [22], 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. A peer-to-peer (P2P) [20] distributed system is one in which 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. There is no server that offers services to clients. Service and data are provided by the participants in the P2P network at any given time.

In summary, our 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 [22], and group membership management and discovery in peer-to-peer networks [31]. We will be using these results in designing a solution for the cooperative driving problem.

Discovery protocols are used in establishing connectivity among wireless devices. The group membership maintenance problem is defined as the requirement for each host to have knowledge of what other hosts are members of its group and for such knowledge to be consistent across the entire group at all times [31]. To solve this problem, assumptions need to be made about host movement and network delays, membership policy, and a protocol that serializes all configuration changes need to be defined. The notion of group can have two slightly different meanings: in the more general sense, a number of nodes belong to the same group if they are within each other's communication range; in a more specific sense (i.e. when referring to the *mutual group driving*), members of the group are only those vehicles that share the trip.

Vehicles periodically send messages to their peers to discover their status. On the other hand, any vehicle joining the network must send a number of housekeeping messages to inform other members about their presence. This will insure basic network connectivity. Each vehicle manages a *routing table*, needed to store information about neighboring vehicles (participating vehicles within its range) which will be used in look-ups. Updates in the routing table are triggered by the receipt of a broadcast message. Because vehicles can fail (go out of range) without prior notification, each vehicle must periodically monitor the state of its neighbors. The membership in mobile ah-hoc networks is dynamic.

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 the time, so clock synchronization is not an issue. Vehicles continuously broadcast these messages. History table management is discussed in detail in Section 4.

GPS-enabled vehicles allow for the development of useful location dependent services. Examples of such services are: direction assistance, navigation, automated road assistance, selective multicasting to specific geographic areas (defined in terms of GPS coordinates), providing a service only to nodes withing a certain geographic range, providing information to mobile users when the information depends on the user's geographic position. Integrating GPS data into network protocols, such as the Internet protocol, to create such services has been studied in [11]. 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 [2, 8, 11, 15, 17, 18, 21]. The idea is that the message address will be specified

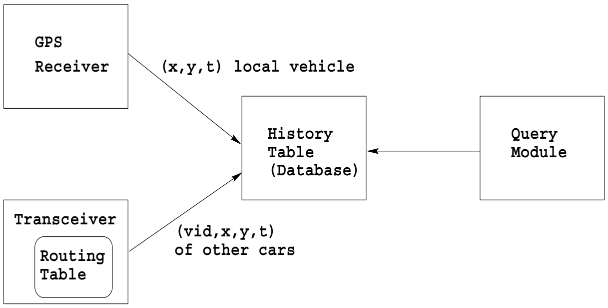

Figure 2: Infrastructure

as a polygon on the map. Then, the polygon will be translated into GPS coordinates and the messages will eventually be multicast to all clients who are located within the bounds of that polygon [11]. For ad-hoc specified areas, the polygon addressing will be used. We can make use of existing algorithms regarding geographic and ad-hoc routing to solve the addressing and routing mechanisms for our scenarios.

As with any P2P architectures 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 [13] 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 (for instance, to find where a particular vehicle which is not within the communication range is located). Instead, each node has some local information regarding its neighboring[1] vehicles. 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 application, 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 plus time t. Further, a global-level historical relation $H(vid, x, y, t)$ stores locations of all vehicles for all times. Each vehicle

---

[1] It is worth mentioning that the issue of proximity (or neighborhood in our case) has been debated in the ad-hoc networks literature.

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 based upon information provided by a central unit. Figure 2 provides the infrastructure.

3.5 Security Issues

Security issues are not our core focus, but we list here some of the important issues.

Our assumption is that the participating members are all trustworthy. For the mutual group driving, it is not unreasonable to assume that the group members are trustworthy. However, since the communication is broadcast-based, messages can be received by malicious traffic participants. To avoid such cases, some security mechanism is needed.

For the highway alert scenario, malicious information can be detected based on the answers received from participants. If contradictory information is detected, then this can be ignored, provided that enough participants collaborate and assuming that malicious cases are rare.

3.6 GPS Issues

Many GPS receivers provide the functionality of displaying the current position of the vehicle on a map. For mutual group driving, the GPS should be able to display the positions of all the group vehicles on the screen. Thus, each driver has a visual picture of where the other vehicles are with respect to his own vehicle and with respect to the final destination. Tracking multiple vehicle routes on the same map is not currently a standard feature of GPS receivers. We assume that given a view of the historical relation H for car i, the GPS can plot the path of each vehicle over time. This view is constructed by only selecting tuples in H_i corresponding to the current driving group. By inputing historical relation H_i into GPS, one vehicle can monitor the route progress of a predefined set of vehicles.

4 Applying Database Techniques to Cooperative Driving

The source data for the database is the history table that contains one tuple per vehicle at time t, and has the following attributes: (vid, x, y, t), where vid is the vehicle identifier, (x, y) represents the GPS coordinates and t is the associated time. The routing table has the same fields, but it stores only the most current information about the direct neighbors of one vehicle, and it is used solely for routing. 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 current tuple (vid, x, y, t) for in H_i. The routing table is managed by the underlying network routing protocol.

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

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-materialize a view.

The information in the routing table can become stale under certain conditions. If some data corresponding to vehicle i has been stored in vehicle j's routing table and it has not been refreshed for the last T units of time, the data becomes *stale* and the record becomes a candidate for deletion. The deletion does not have to happen automatically after interval T. Rather, when T time has elapsed and no new information has been acquired about vehicle X, then i initiates a process of locating X. This may be accomplished by broadcasting a query of the form "*Where is X?*" and waiting for a *reasonable* amount of time for answers from its neighbors. If no answer is received or if the answers are of the form "*No information on X*" or the answers contain location information about X older than i's information on X, then the tuple corresponding to X is deleted from i's routing table.

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 routing 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.*

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 has change 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 a velocity vector information is 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.

We did not specify how this database is indexed, but various algorithms exist [26, 27, 28, 32, 38] for indexing mobile objects, and efficient query answering protocols can be developed based on these indexing techniques.

4.2 Location Dependent Queries

In this dynamic scenario, queries depend on the geographical position of a vehicle, and/or on the relative position of two or more vehicles.

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 by (mostly) any GPS receiver and do not involve peer interaction, include:

- *What is your current position?*
- *How far have you traveled? (odometer)*
- *How long have you been traveling? (timer)*
- *What is your current speed? (speedometer)*
- *What is your average speed?*
- *Draw the journey trajectory showing exactly where you have traveled on the map*
- *Show the estimated time of arrival at your destination if you maintained the current speed.*

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 we should be able ask *peer-dependent queries*, such as:

- *Who are my neighbors?*
- *Where is vehicle X?*
- *Are all the mutual group vehicles within my range?* (of course this query is similar to the first one, because one can check the neighbors list and look for certain vehicle *vid*s).

Peer-dependent queries require peer knowledge and 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*[2], then the information is considered valid. If the information is *stale* [3], 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.

An interesting feature that may be implemented is the one that predicts the next meeting[4] time and location for two vehicles which are not direct neighbors. This may be useful in the mutual group scenario, when one group member gets disconnected from the others, and plan to rejoin the group. The predicted information can be generated by taking into account several parameters: current local time, last update time, last known position and driving direction of the disconnected vehicle.

---

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

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

[4] meeting means the vehicles are withing communication range

```
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 = $\emptyset$ then rec := (X, 0, 0, 0); endif;
send "Where is X?" to $N_i$ (broadcast);
receive { $rec_j$ | $j$ ın $N_i$ };
for each $rec_j$ do
        if $rec_j.t > rec.t$ then
              $H_i := H_i \cup (X, x, y, t)$;
        endif
        $H_i := H_i \cup (j, x_j, y_j, t_j)$;
enddo;
```

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

4.3 Problem Alert Queries

Alert queries involve passing warning messages among vehicles. Similar ideas were presented in [35, 37]. 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 and/or movement 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. The 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. Different types of warnings may exist, corresponding to different degrees of potential hazard.

Note that the message propagation wave can reach vehicles located very remotely from the actual incident, and might 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. A message can be ignored or, on the contrary, actions can be taken appropriately by the driver.

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 a peer-to-peer interaction.

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 routing table management, namely constructing materialized views of the system, and how this information can be used in answering location dependent queries. A protocol for answering 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 (can easily reach tens of thousands of records when refreshing period is one second, the trip duration is at least one hour and the number of neighbors is ten on average). Techniques for reducing table size are important.

References

[1] T. Brinkhoff. Requirements of traffic telematics to spatial databases. In *Proceedings of the 6th International Symposium on Advances in Spatial Databases (SSD99)*, volume 1651, pages 365–369, 1999.

[2] CATSS. Center for advanced telecommunications systems and services: GPS enabled ad hoc routing and multicast project. URL: catss.utdallas.edu/research/gps.htm.

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

[4] X. Defago. Distributed computing on the move: From mobile computing to cooperative robots and nanorobotics. In *Proceedings of the First Annual ACM Workshop on Principles of Mobile Computing (POMC2001)*, pages 49–56, 2001.

[5] 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, Gemeinsame Jahrestagung der GI und OCG, 2001. FleetNet Project: http://www.et2.tu-harburg.de/fleetnet/english/documents.html.

[6] GPS. *How GPS Receivers Work*. http://www.howstuffworks.com/gps.htm.

[7] S. Gribble, A. Halevy, Z. Ives, M. Rodrig, and D Suciu. What can databases do for peer-to-peer? In *Proceedings of the 4th Workshop on Databases and the Web (WebDB2001)*, 2001.

[8] GRP. Geographic routing project. URL: http://www.cs.rutgers.edu/ navas/dataman/papers/.

[9] A. Heuer and A. Lubinski. Database access in mobile environments. In *Database and Expert Systems Applications*, pages 544–553, 1996.

[10] Q. Huang, R. Miller, P. MacNeille, G.-C. Roman, and D. DiMeo. Development of a peer-to-peer collision warning system. Technical report, submitted to Ford Technical Journal, 2001.

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

[12] ITTS-Research. Intelligent transportation and telematics reasearch group at UCSD. URL: http://cvrr.ucsd.edu/projects.html. CVVRR Computer Vision and Robotics Research Laboratory.

[13] C.S. Jensen, A. Friis-Christensen, T.B. Pedersen, D. Pfoser, S. Saltenis, and N. Tryfona. Location-based services - A database perspective. In *Proceedings of the 8th Scandinavian Research Conference on Geographical Information Science (ScanGIS2001)*, 2001.

[14] M.O. Killijian, R. Cunningham, R. Meier, L. Mazare, and V. Cahil. Towards group communication for mobile participants. In *Proceedings of the First Annual ACM Workshop on Principles of Mobile Computing (POMC2001)*, pages 75–82, 2001.

[15] X. Lin and I. Stojmenovic. GPS based distributed routing algorithms for wireless networks. Unpublished manuscript.

[16] N. Marmasse. comMotion: a context-aware communication system. In *Proceedings of the Conference on Human Factors in Computing Systems CHI99*, 1999.

[17] M. Mauve, J. Widmer, and H. Hartenstein. A survey on position-based routing in mobile ad-hoc networks. *IEEE Network*, November 2001.

[18] J. Mehaffey. Automatic address to address routing GPS automobile navigators: What to expect from the currently available equipment. URL: http://www.gpsinformation.net/autogps2.htm.

[19] T. Mitakos, M. Gergathouslis, Y. Stavrakas, and E. Ioannidis. Representing time-dependent information in multidimensional XML. In *Proceedings of the 23rd International Conference on Information Technology Interfaces*, June 2001.

[20] M. Naraghi-Pour, M. Hegde, and R. Pallapotu. Peer-to-peer communication in wireless local area networks. In *Proceedings of the Seventh IEEE International Conference on Computer Communications and Networks (ICCCN98)*, pages 432–437, 1998.

[21] C.J. Navas and T. Imielinski. Geographic addressing and routing. In *Proceedings of the Third ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'97)*, 1997.

[22] S. Nesargi and R. Prakash. Issues pertaining to service discovery in mobile ad hoc networks. In *Proceedings of the First Annual ACM Workshop on Principles of Mobile Computing (POMC2001)*, pages 41–48, 2001.

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

[24] Oracle. Leveraging location-based services for mobile applications - An Oracle Technical White Paper. Technical report, Oracle Co., June 2001.

[25] G. Ozsoyoglu and R.T. Snodgrass. Temporal and real-time databases: A survey. *Knowledge and Data Engineering*, 7(4):513–532, 1995.

[26] D. Pfoser and C.S. Jensen. Querying the trajectories of on-line mobile objects. In *Proceedings of the Second ACM International Workshop on Data Engineering for Wireless and Mobile Access (MobiDE2001)*, 2001.

[27] D. Pfoser, C.S. Jensen, and Y. Theodoridis. Novel approaches to the indexing of moving object trajectories. In *Proceedings of the 26th International Conference on Very Large Databases (DLDB2000)*, 2000.

[28] L. Porkaew, I. Lazaridis, and S. Mehrotra. Querying mobile objects in spatio-temporal databases. In *Proceedings of the The Seventh International Symposium on Spatial and Temporal Databases, SSTD2001*, pages 59–78, 2001.

[29] Impulse Project. Location-based agent assistance. URL: agents.www.media.mit.edu/groups/agents/projects/impulse.

[30] O. Ratsimore, V. Korolev, A. Joshi, and T. Finin. Agents2go: an infrastructure for location-dependent service discovery in the mobile electronic commerce environment. In *Proceedings of the First ACM Mobile Commerce Workshop (AMCW01)*, July 2001.

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

[32] S. Saltenis, C.S. Jensen, S.T. Leutenegger, and M.A. 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.

[33] S. Shimizu, K. Asano, and K. Tokuda. Development of an ROF Road-Vehicle Communication System. *OKI Technical Review*, 68(3), September 2001. Special Edition on ITS of OKI Electric Industry Co., Ltd.

[34] 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.

[35] C.J. Soulliard and M.C. Soulliard. Vehicle detection goes wireless. *IMSA Journal*, Jan/Feb 2002. url: www.imsasafety.org/.

[36] S. Tsugawa. An introduction to 2000: The cooperative driving scenario. *OKI Technical Review*, 68(3), September 2001. Special Edition on ITS of OKI Electric Industry Co., Ltd.

[37] U. Vissr and H. Stuckenschmidt. Intelligent, location-dependent acquisition and retrieval of environmental information. Intelligent Systems Group, Center for Computing Technologies, University of Bremen, Germany.

[38] O. Wolson, B. Xu, and S. Chamberlain. Location prediction and queries for tracking moving objects. In *Proceedings of the 16th International Conference on Data Engineering (ICDE2000)*, 2000.