

**A Method for Creating a Real-Time  
Distributed Travel History Database:  
The PTC Project**

UCI-ITS-TS-WP-04-7

James E. Marca <sup>1</sup>  
Craig R. Rindt <sup>1</sup>  
R. Jayakrishnan <sup>2</sup>

<sup>1</sup> Institute of Transportation Studies  
University of California, Irvine  
jmarca@uci.edu, crindt@uci.edu

<sup>2</sup> Department of Civil and Environmental Engineering and  
Institute of Transportation Studies  
University of California, Irvine, rjayakri@uci.edu

June 2004

Institute of Transportation Studies  
University of California, Irvine  
Irvine, CA 92697-3600, U.S.A.  
<http://www.its.uci.edu>

# A method for creating a real-time, distributed travel history database: the PTC project

James E. Marca

Craig R. Rindt

R. Jayakrishnan

June 25, 2004

## 1 Introduction

The usual approach to monitoring traffic is to use an infrastructure-based vehicle detection system, transmit that information to a local or centralized analysis center, and then process that data to extract the variables of interest, which can then be employed for determining control actions. The information flow is entirely external to the vehicles. The only interface between the management system and the vehicles is at the point of detection, and that interface is typically a passive one-way link from the vehicle to the sensor.

One major improvement in detection that appears to be gaining traction is the move from anonymous point measurement of aggregate flow, speed, and occupancy variables to vehicle-centric measurements of routes, travel times, and general vehicle dynamics. The added fidelity of the data coming from these types of detection systems, however, comes at a cost. Such monitoring systems must be capable of recording an individual vehicle's progress through the network over time. This presents technical problems of how to (1) uniquely identify a vehicle, (2) track that vehicle as it moves through the system, and (3) efficiently store and process that data (which may be sizable in large systems) for use in traffic management. Furthermore, systems built upon such a vehicle tracking approach raise privacy concerns that cannot be ignored.

This paper describes an approach to solving these problems by making the vehicle a partner in the process of traffic monitoring and control. We propose leveraging the computational power that is available in vehicles, or even on PDAs or similar devices carried by pedestrians or bicyclists, by augmenting such devices with a short-range wireless communication system to permit data exchange with the infrastructure. This system leads to a method for creating a distributed travel history database using in-vehicle computers and short-range wireless communications. Such a database can be employed in a range of traffic management applications by improving the system's ability to monitor and analyze real-time traffic demand and performance.

The use of short-range wireless communications has been studied for a number of applications, including for Automated Highway Systems (AHS) and both centralized and decentralized information dissemination. This paper focuses explicitly on integrating vehicles as *computational resources* that can collaborate with the public traffic management infrastructure to improve individual and system-wide performance.

In the past, this was not feasible due to the expense of the hardware required and the limitations of the communication link. However, embedded computers are now ubiquitous and wireless

technologies are cheap and readily available. With 802.11b (WiFi) chips becoming essentially a commodity product and many other protocols and radio technologies being actively developed, we are confident these requirements will be trivial to meet in coming years. Finally, while a roadside wireless access point will not entirely replace the need for embedded loops, the hardware costs of a WiFi access point are comparable, and the installation costs are far cheaper, since the pavement does not need to be cut up.

Conceptually, the system we propose is quite simple—each mobile device collects a cookie each time it establishes a connection with a roadside device, in a process analogous to a web browser collecting a cookie from a website. These traffic cookies record the current state of the mobile device—where it is and at what time. They are written such that taken together, they describe the complete travel history of the vehicle, and can be read and authenticated by any roadside device in the system. Finally, the cookies are persistent from trip to trip, leading to the name persistent traffic cookies, or PTC.

The PTCs each vehicle carries can be used for a number of interesting applications. One of the most interesting is to use them to obtain an estimate of the current path-level demand in the system. The state of the art approach to estimating real-time path demands—Dynamic Traffic Assignment (DTA)—approximates the current trip table based on point-based flow measurements, some prior notion of origin and destination demand in the system (from historical sources or from a prior iteration), and embedded models of driver routing behavior. Individual vehicle paths are not sampled directly, nor is the underlying historical Origin-Destination (O-D) matrix limited to just the vehicles that are currently on the road. This information gap is a significant source of error in the DTA estimates. To the best of our knowledge, however, there are no robust and scalable methods to determine an approximation of the prevailing real-time path demands using direct vehicle sampling. The main stumbling blocks are sampling vehicle paths, storing that information, and then making use of the stored path information in real time.

The PTC system could close this gap by providing a means for obtaining a direct estimate of the routes (and thus O-Ds) being used by vehicles currently in the system. Using the cookies from each participating vehicle, each roadside controller knows those vehicles' histories, both for past trips and the current trip. The controller can use these histories to predict the most likely downstream paths those vehicles *might* take by conditioning the historical information with the current trip information. The resulting path predictions form the basis for a system-wide O-D and path demand estimate that can be integrated into DTA and other traffic management applications.

The elegance of the PTC approach is that it does not require any centralization of information, and the communications hardware is limited to the (free) wireless link between the vehicles and the roadside device. Although we can use any centralized communications lines, the system can operate just fine without it, since the vehicles carry the right information to the right place as natural side effect of traveling. Further, none of the current detector based approaches are amenable to monitoring pedestrian or bicycle movements.

Finally, because the tracking system is owned and operated by the drivers themselves, many of the privacy concerns can be alleviated. Drivers are under no requirement to participate and can clear the travel memory of their embedded system at any time. Because of this, incentives to participate will likely be necessary—the most obvious being access to traffic information from the same embedded system that is cooperating with the traffic management system. This exchange of information—user behavior data exchanged for value-added information—is similar to the business model used by many of the largest sites on the World Wide Web.

The next section gives an overview of current approaches to traffic monitoring and control, along with a description of what we see as the major shortcomings of centralized approaches. Then section 4 describes the PTC method for tracking vehicles and predicting traffic flow conditions, using a detailed example and results from a simulation to illustrate the concept and its potential. Section 5 discusses some practical applications that can leverage the information gathered and the communications links established by a working, system-wide PTC network. Finally, section 6 concludes this paper with a research roadmap, and a description of planned field tests.

## 2 Related work

### 2.1 Intelligent Transportation Systems and Traffic Management

The conventional view of traffic management envisions a system of sensors providing point-based or link-based state measurements as shown in figure 1. These measurements are passed to a centralized management system, which uses them for estimation of supply and demand states, and applies algorithms to determine optimal control strategies. These control strategies are then implemented using the available control infrastructure.

Despite the years of research on ITS systems, most traffic management systems are not integrated, adaptive systems. Generally, travel demand is estimated off-line using travel demand forecasts from regional and local planning models. Local traffic counts are used to generate turning movement predictions, which are in turn used for one-time optimization of traffic actuated controllers. Fortunately, traffic actuated controllers operate efficiently if prevailing conditions are in line with the forecasts used to optimize the system. However, as systemic demands approach capacity, or when there is an incident in the system that drastically alters travel demand patterns, traffic actuated control rapidly loses efficiency leading to large queues and sizable delays.

To alleviate these shortcomings, truly adaptive traffic control has long been a goal of researchers and practitioners alike. The best known of these systems include Split Cycle Offset Optimization Technique (SCOOT) [Bretherton et al., 1998], Sydney Coordinated Adaptive Traffic System (SCATS) [Orselli, 1995], and Optimized Policies for Adaptive Control (OPAC) [Gartner et al., 2001], among others. These control systems are largely confined to local optimization. OPAC itself has been incorporated into Real-Time Traffic Adaptive Control System (RT-TRACS) [Pooran et al., 1996], which combines a set of adaptive control schemes that are selectively applied depending on prevailing conditions. The long-term goal for RT-TRACS deployment is to link the adaptive control systems with a system-wide management scheme that is based on centralized DTA—recognizing the link between path and O-D demands and traffic supply embodied by measured flows and prevailing traffic control settings. This centralized traffic management application would respond to unusual capacity and demand fluctuations that would cause a breakdown in the system. The assignments proposed by the DTA system would be sanity checked with a fast simulator, and then the optimal adaptive control algorithm would be implemented within the RT-TRACS framework.

While such integrated schemes have shown promise, their extensive real-time data requirements and significant complexity make them a difficult option for general deployment. Furthermore, the previously mentioned data gaps that hamstring DTA make it difficult to make a conclusive case that the complete system would have a noticeable impact on system performance. Richer

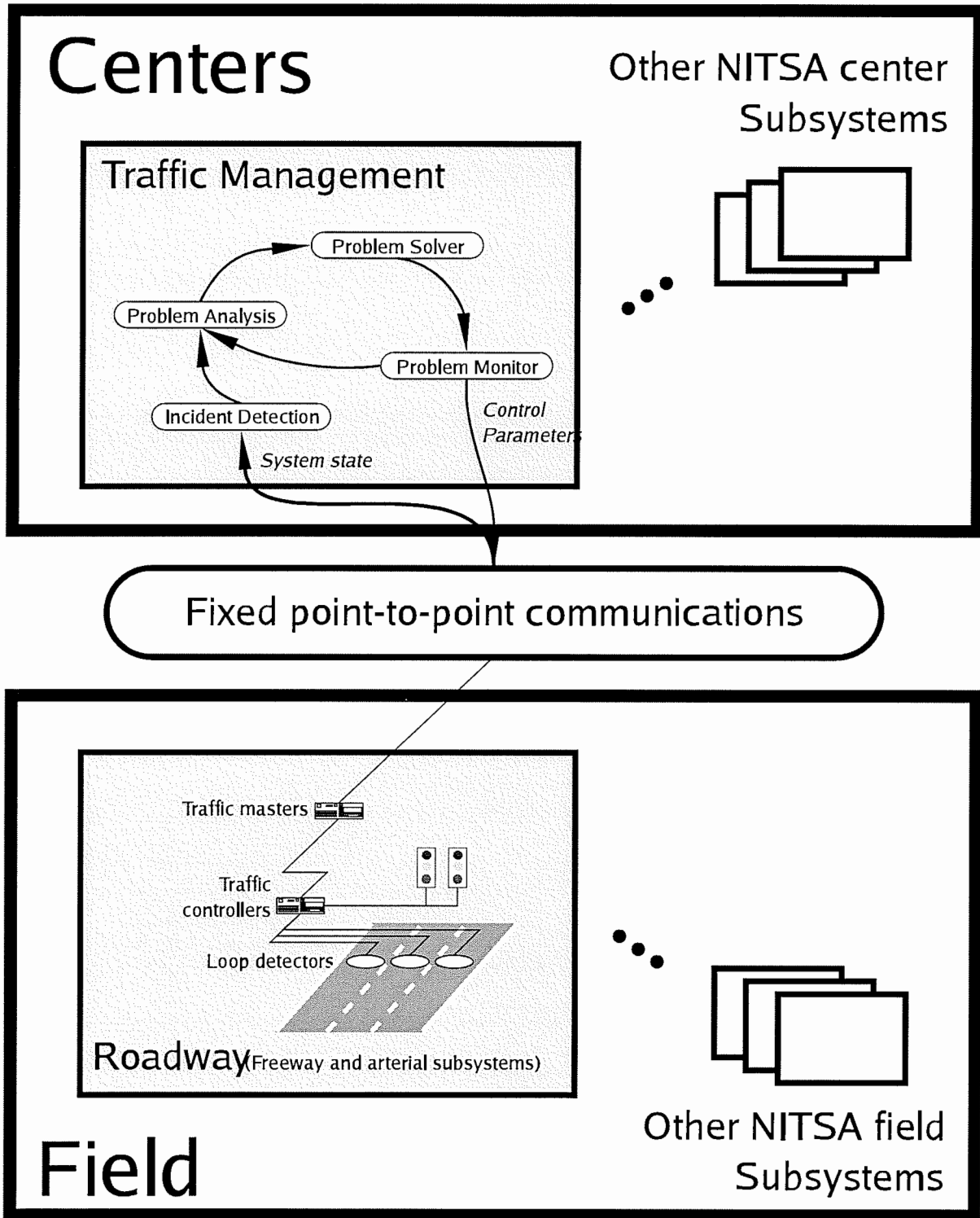


Figure 1: A typical traffic management system.

information about the path and O-D level travel demands would go a long way toward answering these questions.

## 2.2 Traffic measurement

In general, state measurement for traffic systems involves both supply-side measurement (speed, flow, occupancy, and travel times) and demand-side measurement (O-D, path, and link flows). Even in modern day traffic management systems, the venerable Inductive Loop Detector (ILD) provides the bulk of real-time state measurement. The resolution and quality of this data, however, limits it largely to supply side performance measurements that can be used to estimate capacity of nearby sections, and for incident detection, information dissemination, and local traffic control. On the demand side, conventional ILDs can only measure link-level flows at specific points in the network, which are generally equivalent to link-level demands unless the link is operating at capacity with demand actually exceeding the measured flow (resulting in queuing).

A significant amount of research has been produced into improving sensing in transportation systems to obtain the richer information necessary for reliable system-wide travel demand estimates and predictions. The most promising of these technologies are reviewed in the following subsections.

### 2.2.1 Infrastructure-based detector systems

A recent avenue of research showing promise is the refinement of ILDs to identify each vehicle's (presumably) unique inductance signature [Oh et al., 2002b]. By placing these signatures in a database that is accessible to other controllers in the system, or to the Transportation Management Center (TMC), the vehicles can be re-identified as they move through the network [Oh and Ritchie, 2003]. This approach can be used to measure (supply side) link travel times [Coifman, 2002] as well as estimate path and O-D level demands [Oh et al., 2002a]. This system must overcome a number of challenges, however, before it can be deployed in practice. First, there are technical problems related to searching a potentially exponentially growing space of candidate vehicles as the catchment area grows. Second, the cost of providing upload and download capacity between roadside controllers and the central database will grow exponentially with the growth of the network, or else the capacity of the communications network will degrade steadily with the growth of the network. Third, unauthorized, surreptitious vehicle tracking is likely to present significant privacy concerns. Other sensor technologies have been developed over the years, with video detection systems garnering the most attention. But for many reasons, including deployment and maintenance costs and the complexity of the solution methods, none have made any headway at replacing the ILD in the field.

### 2.2.2 Vehicles as Probes

The use of vehicle probes has also long been contemplated as a means of obtaining traffic stream data over space, instead of just point measurements. Properly equipped vehicles can be used as traffic flow data probes to improve both supply and demand side measurements. A review of the literature shows proposed applications in a variety of areas. Most prevalent are network speed and travel time estimation [Cheu et al., 2002], incident detection [Thomas, 1998, Sethi et al.,

1995], and O-D estimation [Hellings, 1997]. The expected results are qualified, however, by a consideration of the necessary size of the probe vehicle fleet. For example, Van Aerde et al. [1991] seem to conclude that to do *effective* O-D estimation using probe vehicles would require a very high participation rate by all of the vehicles on the road. Hellings [1997] conclude that

The statistical analysis of the expected quality of O-D demands, which are estimated solely on the basis of RGS probe vehicle data, indicated that even for levels of market penetration of 30%-50%, the O-D estimates are unlikely to be of sufficient quality to be of practical benefit.

It is noteworthy that these uses of probe vehicles treat the probes as read-only sensors that passively provide information about the state of the system in the recent past (speeds over a link, queue lengths at intersections, and so on). This read-only treatment of the probe vehicle concept appears to be fundamental to the ITS community's perception of the role of vehicles in traffic management as indicated by the definition of the *Probe Surveillance Market Package* in the National Intelligent Transportation Systems Architecture (NITSA) [Iteris, Inc, 2002]:

The [Probe Surveillance Market Package] enables traffic managers to monitor road conditions, identify incidents, analyze and reduce the collected data, and make it available to users and private information providers. . . .

Again, this read-only interface with the vehicle leads directly to technical problems with the amount of data that can be collected and how to fit all that data into single pipe feeding into a central data processing and storage site. The NITSA quote above continues with the following observation.

Due to the large volume of data collected by probes, data reduction techniques are required, such as the ability to identify and filter out-of-bounds or extreme data reports.

This contrasts with our approach of leaving the information on the vehicles, and processing it directly at each intersection. The difference perhaps stems from our belief that all vehicles can be partners in traffic control. This would lead to a more decentralized traffic management system—a concept that has also received significant attention in the literature.

## 2.3 Decentralized Traffic Management

An alternative to the monolithic traffic management architecture described above in section 2.1 is to employ decentralized traffic management. In such systems the measurement, estimation, management, and control architectures are partially or completely distributed among diverse resources. There are numerous examples in the literature, including the October, 2002 issue of *Transportation Research, Part C*, which was dedicated entirely to the use of agent technology in traffic and transportation. These systems all describe different decentralization schemes.

For instance, a natural approach is to use operational criteria to decompose the system into manageable sections. In this vein, Hernandez et al. [2002] describe the TRYS multi-agent traffic management architecture that decomposes a global transportation system into a set of local “problem areas,” which are individually maintained by separate traffic management agents. Because the control strategies applied by one agent may affect the performance of another agent's problem

area, the agents must coordinate their actions to generate better global solutions. Similarly, van Katwijk and van Koningsbruggen [2002] describe an agent-based system for coordinating traffic management by modeling traffic management “instruments” using intelligent agents. In their conceptualization, intelligent software agents represent various management instruments in system, ranging from high-level management functions down to individual traffic controllers. The behavior of the collective of instrument agents is dictated both by their local goals as well as through coordination with other agents.

An alternative is to break the system down into jurisdictional boundaries that reflect the institutional realities of large-scale urban traffic management and therefore offer an attractive practical solution to traffic management. Logi and Ritchie [2002] describe the Coordinated Adaptive Real-Time Expert System for Incident management in Urban Systems (CARTESIUS) system, which is similar to the TRYS system in many ways. Unlike TRYS, however, decomposition of the transportation system is based on jurisdictional divisions rather than arbitrary “problem areas.” Each agent controls a particular jurisdiction’s traffic management infrastructure, but coordinates with the agents representing other jurisdictions to produce globally efficient solutions that are guaranteed to not violate local policies. Similarly, Adler and Blue [2002] argue that agent-based systems are a natural extension to the institutional divisions outlined in the NITSA and go on to describe a hierarchical architecture with agents representing public area managers, commercial information providers, and private travelers. The system provides a framework for supply-side (management) agents and demand-side (driver) agents to negotiate trip assignments through the network in a mutually satisfactory manner.

Similar to Adler and Blue’s [2002] driver agents, the idea of getting the vehicles themselves involved in traffic management functions, beyond their use as simple read-only probes, has been proposed by a number of researchers. For instance, [Ziliaskopoulos and Zhang, 2003] describe the Zero Public Infrastructure (ZPI) traffic information system in which properly equipped vehicles share traffic information using vehicle to vehicle wireless communications. In a similar vein, Kher et al. [2002] present the kernel of a traffic management system which relies on vehicle-to-vehicle information passing, as with [Ziliaskopoulos and Zhang, 2003], but focused on the potential for vehicles to change their routes based on the available information. Within the vehicle control literature, there is an interesting approach proposed in a series of papers [Moriarty et al., 1998, Moriarty and Langley, 1988] in which vehicles “learn” lane changing strategies and lane organization for optimal flow. These research projects are beginning to demonstrate that the cooperative approach to traffic management can be generally superior to hand crafted control schemes.

### 3 Filling the gaps with PTC

The PTC system is grounded in a decentralized view of traffic management, and solves a number of problems faced by existing integrated traffic management schemes. In this approach, we view mobile, vehicle based computers as valuable resources within the context of traffic management and control. It is in our best interests to use these computers as best as we are able, and to reward participating drivers both directly (perhaps through better information provision) and indirectly (through better system-wide efficiency). Second, this research stresses the interaction of roadside devices with in-vehicle devices, evolving the read-only detection paradigm to a read-write communication paradigm.

Third, this project is not yet considering vehicle to vehicle exchanges of information, as is proposed by Ziliaskopoulos and Zhang [2003] and Kher et al. [2002]. While there is nothing preventing vehicle to vehicle communication in future generations of the PTC system, the initial stages of our research have been carefully designed to establish the feasibility of a simple system that can be demonstrated with a field test using current, off-the-shelf technology. The in-vehicle devices need only be slightly more capable than common electronic toll collection transponders, and the communications between the vehicle and the roadside controller requires only slightly more information to be exchanged than the standard toll collection case. More importantly, a vehicle to roadside wireless communication system requires only a *single* wireless hop. That is, the participating mobile devices only communicate directly with the roadside device, and do not try to route messages to or from other, remote peers. Ad hoc, peer to peer routing is an area of active research in the Computer Science literature. While there are several candidate schemes for such a system, none have achieved widespread acceptance as an official or de facto standard.

Finally, by focusing on an incremental expansion of read-only detection to read-write communication, we can firmly establish practical examples of the utility of letting vehicles cooperate in the traffic control system. The end result of the PTC research program will be a completely new and very important source of data, as well as algorithms to embed that data into existing and state of the art traffic control schemes, such as DTA. Jumping directly to vehicle to vehicle exchange of information would break the link with the current infrastructure-centric traffic control paradigms.

Three specific state measurement problems are addressed by the PTC system: vehicle identification, vehicle tracking, and maintaining and querying a travel history for each vehicle. These are discussed in turn in the following subsections.

### 3.1 Vehicle identification

**Problem** First, there is the problem of identifying each vehicle uniquely. Current approaches use external surveillance techniques, such as examining each vehicle's magnetic signature or parsing video data. These techniques are error prone and computationally intensive. Another technology that is promising for vehicle tracking is the use of radio frequency identification (RFID) tags.

**Solution** The PTC system does not need any vehicle identification, as such. One can argue that by using modern wireless communications technology one could simply have each vehicle announce its presence to the roadside computer. (Note that other wireless communications techniques, such as using RFID tags, also uniquely identify each vehicle in this way.) However, in the PTC system, since travel cookies are being written to the vehicle for storage, there is no need to collect a unique identifier from the vehicle for tracking purposes. All that is necessary is the ability to differentiate between various simultaneous wireless connections, something that is already handled nicely within wireless communications protocols such as 802.11b.

Finally, the user can also turn the device off, reset the memory, or otherwise enforce privacy guards. This is not possible with RFID, video monitoring, and so on. These advantages reduce the invasion of privacy aspects of vehicle identification, which may help make the PTC system more palatable to the traveling public than an external detection system.

### 3.2 Vehicle tracking

**Problem** Once a vehicle is *identified* at a particular point in the road network, its travel to other points in the network must be determined. The prevailing infrastructure-based bias that exists in all detector related research implicitly or explicitly requires communication between all detection points and a central computer. The detected vehicle's unique identifiers (RFID tag, magnetic signature, image of license plate, etc) must be matched and tracked through each detector point, which requires sorting or searching all of the vehicles identified at every detection point downstream from the original point in a geometrically expanding set. Even for technologies which uniquely and absolutely identify each vehicle, such as RFID tags, determining a vehicle's path through the network is computationally expensive, approximately increasing with the square of the number of nodes and the numbers of vehicles. For some technologies, such as magnetic signature processing or image analysis, the vehicle signature is not guaranteed to be unique, which further compounds the computational size of the problem.

**Solution** The PTC approach is to track vehicles by having the vehicles track themselves. The problem of storing and transmitting the vehicle tracking data that arises with other, detector-based approaches to vehicle tracking is solved here by storing the data on the vehicles, and having them *carry* the data to the points (in time and space) at which the data are needed. While obvious in hindsight, this is a completely novel approach to collecting, storing and retrieving vehicle-based travel data, and at the heart of the PTC system's efficiency and scalability. To eliminate false data being implanted in the system, only data written and signed by the roadside computers is trusted by the roadside computers. (We intend to study the impact of allowing untrusted, vehicle-reported data using simulation studies, but have not yet done so.)

### 3.3 Maintaining and querying a travel history for each vehicle

**Problem** The most significant problem with the current, centralized approaches to estimating a trip table based on detector data is the difficulty of storing the data, and then querying and forming predictions from the resulting database. In a centralized solution, the cost of storing each vehicle's trip from day to day is huge. The more trips are made, and the more data is collected, the larger the central database becomes and the more difficult it becomes to generate conditional predictions for each intersection. Even if the central database and processing costs could be met, the cost of installing the communications network that would enable each intersection to upload its detected vehicles every signal cycle, and to then download those vehicles' most likely destinations is likely to be even more expensive. In short, considering the size of the benefit, the costs are likely to be prohibitive. Finally, maintaining a *central* database of individual vehicles' travel patterns is a tremendous invasion of privacy, and will face stiff opposition from social and political forces, even if it could eventually be made technically and economically feasible.

**Solution** The PTC system stores all of the required information exactly where it is needed—in the vehicles that present themselves at the intersection. In the *worst* case from a communications and computational complexity case, the roadside computer will have to read *all* of the vehicle's records, and then parse those records to consider only those which contain the particular roadside computer's identification code and which are aligned with the current trip direction to determine

the vehicle's most likely downstream travel paths. In the best case, each read of the past history would be paired with a corresponding write summarizing the intersection's interpretation of the past history, thus entailing just two reads and writes per vehicle (one for the history, one for the current trip). Other optimizations are possible, especially since two consecutive roadside devices are likely to submit nearly identical queries to the vehicle, allowing responses to be precomputed and cached. Finally, if the decision is made to trust the computations made by the vehicles themselves, even greater efficiencies can be gained by relying on in-vehicle processing power.

## 4 A detailed example describing the PTC system

We have constructed a detailed example to demonstrate the properties of our proposed system. We generated 1,000 artificial individuals, all pursuing individual activity patterns in an artificial travel and land use network. The primary purpose of using this activity-based simulator was to gather travel data that was repeatable but with some random elements. We fully expect that truly random travel will generate uniformly random travel histories for each vehicle, which is completely uninteresting for the PTC system. In contrast, any non-random component to travel behavior should be detectable by examining a vehicle's history. For the purposes of this paper, we have further supposed that all vehicles in the network are participating in the PTC scheme, that every intersection has been instrumented with a roadside beacon, and that communications in all cases works perfectly. We are fully aware that all of these assumptions are unlikely in practice. However, we are not trying to evaluate specific operational plans or determine the minimum adoption rate necessary for the system to be effective. Rather we are just trying to explain the basic idea, and will leave implementation details and ancillary services to future papers.

Figure 2 shows an overview of how the PTC system works. A single vehicle's trip-table history is grown over time by repeated collection of messages from roadside computers. As more data is collected, the historical average becomes more representative of the true propensity for different paths and destinations the vehicle might take. As the history becomes more refined, it departs significantly from both the naive model (all links are equally likely) and the global average.

Figure 3 shows one method for establishing the necessary exchange of data between a participating vehicle and the roadside computer. As the figure shows, the first step is for an approaching vehicle and the roadside computer to establish a local area wireless communication link. In our modeling and testing this link will be over 802.11b, although any of the many local area wireless communications standards (DSRC, IR, Bluetooth, any of the 802.11 protocols, etc) can be used for this purpose, provided the link is has enough capacity and range to exchange the necessary data. In the example of figure 3, each cookie is cryptographically signed to establish authenticity. Thus once the connection has been established, the roadside computer asks the vehicle for the current active traffic cookie and its associated cryptographic signature. The current cookie contains information on the vehicle's current trip. The roadside computer inspects the cookie to determine the immediate upstream node, and verifies that the upstream node did indeed sign this cookie with the associated signature. Any other suitable verification and authentication method can be used here, as long as the roadside computer can be certain that another trusted device wrote the vehicle's cookie. When this has been verified, the roadside computer adds its own node name and the current time to the active cookie, signs it with its own key, and returns the new active cookie and the new signature to the vehicle. This cookie and key will be processed in turn by the next node

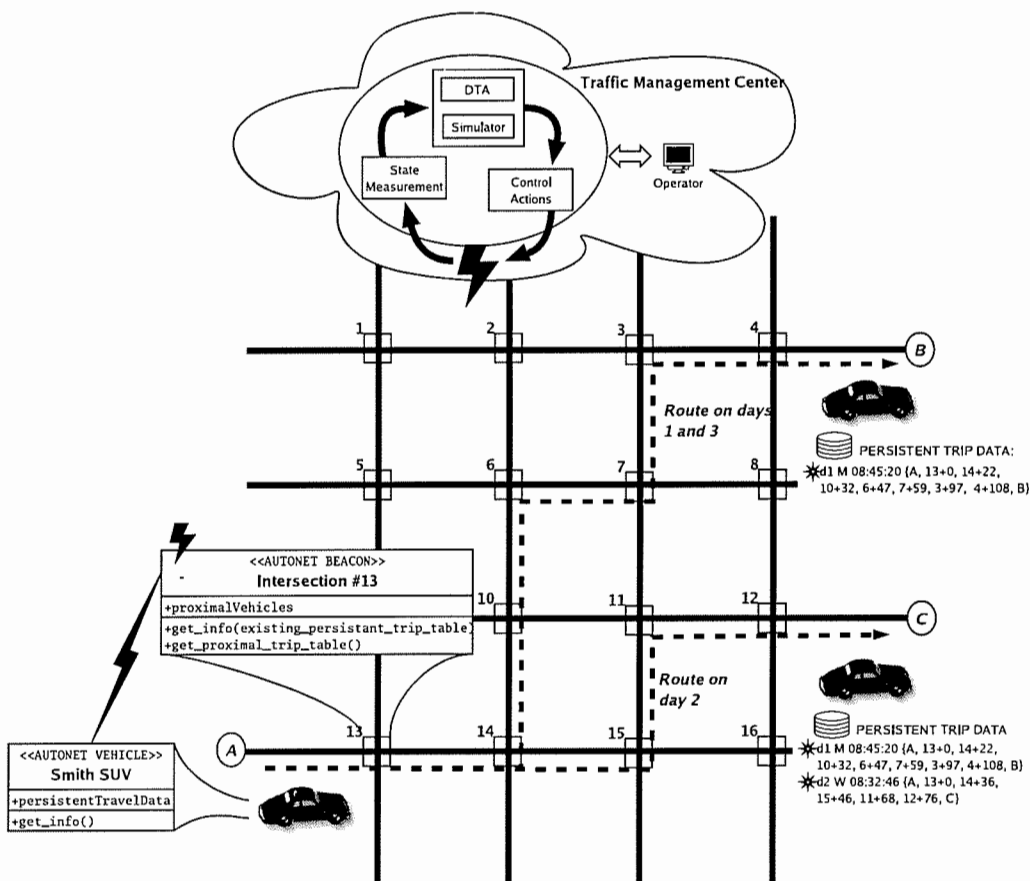


Figure 2: Example of a cooperative traffic management system. In the example shown, beacon #13 interacts with the Smith SUV on three consecutive days. The first day, the beacon retrieves no information from the vehicle, but writes an authenticated, timestamped “traffic cookie” to the vehicle’s on-board computer. Similar interactions occur as the vehicle passes through each intersection, leaving the network at B. The second day beacon#13 retrieves the previously written cookie, and uses it to update its local view of the network’s current travel demand. Based on the available information, a local demand estimation algorithm in the beacon would expect the Smith SUV to follow the same route and exit the network at B. Instead, it takes a different path and exits at C. Events on the third day are similar, except that now the demand estimation algorithm that the trip will either take the first path to B 50% of the time and the second path to C 50% of the time. After day three, the splits are 67% and 33%, and so on. A TMC might also query the beacon population to supply travel demand estimates to the DTA application.

down the road. A subset of cookies collected in this manner for a single vehicle in a simulation experiment is shown in figure 4.

Next, the roadside computer asks the vehicle for its past travel history. One possible query is to ask for all past traffic cookies, and then to parse those cookies. Another, more efficient request is to ask for all persistent traffic cookies which contain the node in question. This can be used to determine the possible downstream routes of the vehicle. However this kind of a request ignores readily available information on heading. Instead, in the example, the roadside computer requests all past cookies which contain the sequence of the current node and the immediate prior node.

Suppose the current roadside computer is called  $x$ , the immediate upstream roadside computer is  $y$ , and the immediate downstream nodes are  $z_1$ ,  $z_2$ , and  $z_3$ . Then the vehicle computer is asked to transmit all past cookies which contain the sequential pair  $x, y$ . This can be used to predict the most likely downstream node out of  $z_1$ ,  $z_2$ , and  $z_3$ , as well as the path beyond those nodes. The computational requirement for this request for the roadside computer is simply to be able to parse the current traffic cookie to determine the immediate prior node, and then to form the query string. The vehicle computer may have some difficulty responding to this request in a timely fashion if it is a small, embedded processor, but the request is quite simple and easily predicted. Further, the answer is likely to be approximately the same as the answer to the upstream node's request, and so the answer can be cached or prefetched.

There are many other data queries that can be made that are not shown. For example, suppose the roadside computer wants to estimate possible demands later in the day. Suppose the initial request produced a downstream tree in which node  $z_1$  was the most likely subsequent node for the vehicle in question. A second data query might be formed to request any cookies containing the sequential pair  $z_1, x$ . That returned history could be further filtered by time of day, to estimate the most likely time the vehicle might return, if at all.

After the data query has been sent to the vehicle, it responds by uploading all of its persistent traffic cookies which match the conditions of the query, along with each cookie's signature. The roadside device processes these cookies, first verifying that they are authentic, and then extracting downstream probabilities for the vehicle in question.

Figure 4 shows an example of the history a single vehicle might collect. This is in fact the data generated for a single vehicle on day 24 of a simulation of 1,000 vehicles over 30 days. Obviously records have been removed from the figure to conserve space. Note that while the travel data for a vehicle fills several pages, storing that data is trivial for a computer, and transmitting the data consumes very little bandwidth—especially if the history is compressed. Also not shown are the cryptographic signatures that are paired with each string, generated in all cases by the last node to append to a cookie. The cookies are stored as text strings, and cannot be modified by the vehicle without violating match between the string and its cryptographic signature. This allows drivers to inspect the cookies, but prevents them from modifying them or creating artificial cookies. Further research will examine alternate possibilities for verifying the authenticity of cookies, and more importantly, the possibility of using vehicle-originated data (GPS readings) or computations (custom data processing algorithms) to provide richer information to the system with a built-in “trust and verify” mechanism provided by the signed traffic cookies.

Figure 5 shows an example of the history a single vehicle might collect, rendered as a network. In the proposed process, this data is stored as text strings, such as those shown in figure 4, which are far more flexible and descriptive. However, in order to make those cookie files easily understood to person, the data has been mapped onto a network representation. Note that the data shown have

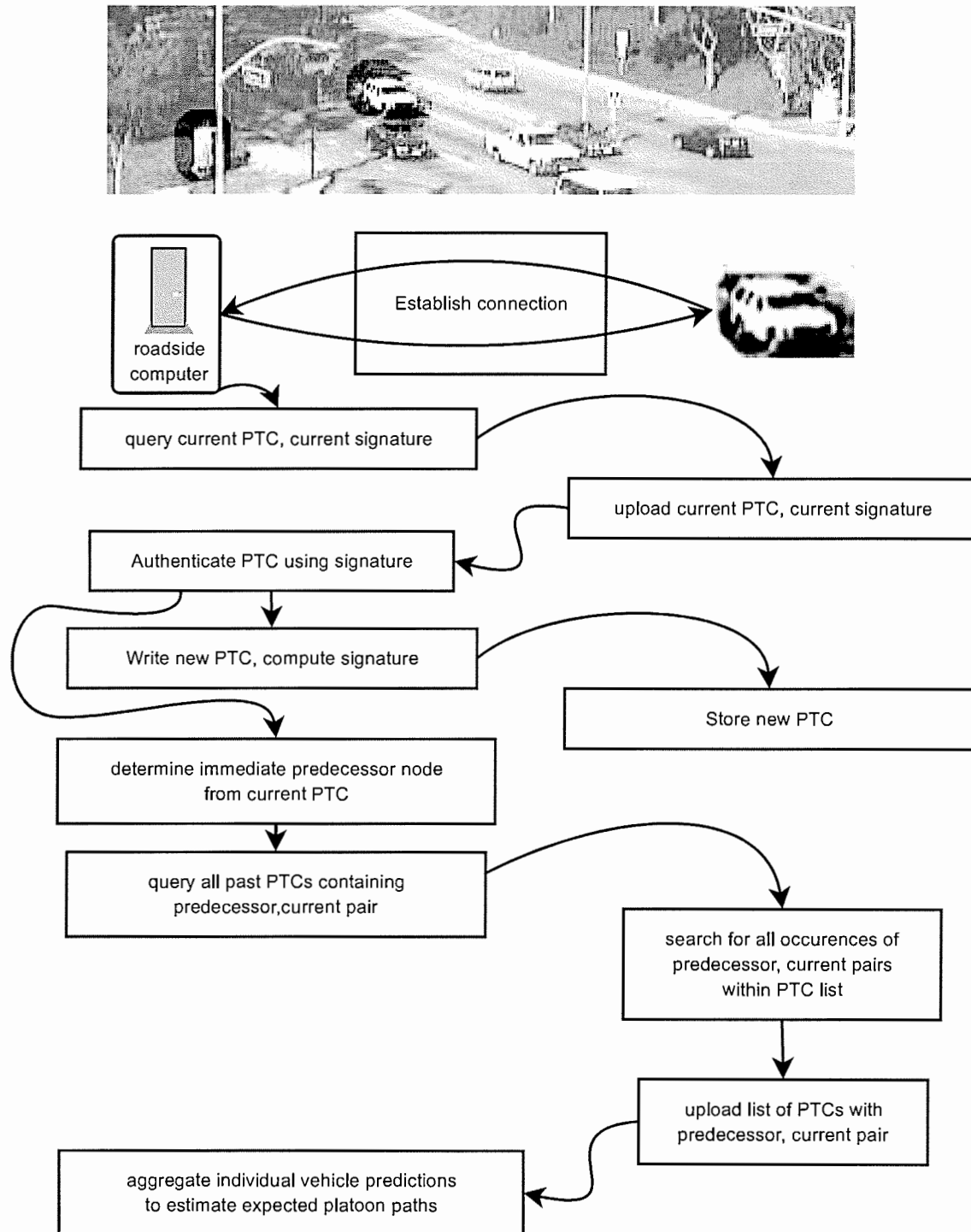


Figure 3: Flow of negotiation between the in-vehicle computer and the roadside computer

```

Vehicle 234 = 2003-07-31 14:00:06 START:94|2003-07-31 14:04:18 IN:92|2003-07-31 14:04:18 OUT:92|2003-07-31 14:06:55 FINISH:84
Vehicle 234 = 2003-07-31 19:00:06 START:84|2003-07-31 19:02:36 IN:92|2003-07-31 19:02:36 OUT:92|2003-07-31 19:08:51 FINISH:94
Vehicle 234 = 2003-07-31 19:45:06 START:94|2003-07-31 19:49:56 IN:92|2003-07-31 19:49:56 OUT:92|2003-07-31 19:52:34 FINISH:84
Vehicle 234 = 2003-08-01 00:00:06 START:84|2003-08-01 00:00:57 IN:92|2003-08-01 00:00:57 OUT:92|2003-08-01 00:07:11 FINISH:94
Vehicle 234 = 2003-08-01 14:00:06 START:94|2003-08-01 14:04:18 IN:92|2003-08-01 14:04:18 OUT:92|2003-08-01 14:06:55 FINISH:84
Vehicle 234 = 2003-08-01 19:00:06 START:84|2003-08-01 19:01:12 IN:92|2003-08-01 19:01:12 OUT:92|2003-08-01 19:07:26 FINISH:94
Vehicle 234 = 2003-08-01 19:45:06 START:94|2003-08-01 19:47:20 IN:92|2003-08-01 19:47:20 OUT:92|2003-08-01 19:49:58 FINISH:84
Vehicle 234 = 2003-08-02 00:00:06 START:84|2003-08-02 00:02:34 IN:92|2003-08-02 00:02:34 OUT:92|2003-08-02 00:08:49 FINISH:94
Vehicle 234 = 2003-08-02 14:00:06 START:94|2003-08-02 14:05:10 IN:92|2003-08-02 14:05:10 OUT:92|2003-08-02 14:07:47 FINISH:84
Vehicle 234 = 2003-08-02 19:00:06 START:84|2003-08-02 19:00:56 IN:92|2003-08-02 19:00:56 OUT:92|2003-08-02 19:07:10 FINISH:94
Vehicle 234 = 2003-08-02 19:45:06 START:94|2003-08-02 19:45:42 IN:92|2003-08-02 19:45:42 OUT:92|2003-08-02 19:48:19 FINISH:84
Vehicle 234 = 2003-08-03 00:00:06 START:84|2003-08-03 00:00:44 IN:92|2003-08-03 00:00:44 OUT:92|2003-08-03 00:06:58 FINISH:94
Vehicle 234 = 2003-08-03 13:30:06 START:94|2003-08-03 13:31:53 FINISH:61
Vehicle 234 = 2003-08-03 14:01:54 START:61|2003-08-03 14:06:08 IN:83|2003-08-03 14:06:08 OUT:83|2003-08-03 14:08:46 FINISH:84
Vehicle 234 = 2003-08-03 19:00:06 START:84|2003-08-03 19:01:49 IN:92|2003-08-03 19:01:49 OUT:92|2003-08-03 19:08:04 FINISH:94
Vehicle 234 = 2003-08-03 19:45:06 START:94|2003-08-03 19:45:14 IN:92|2003-08-03 19:45:14 OUT:92|2003-08-03 19:47:51 FINISH:84
Vehicle 234 = 2003-08-04 00:00:06 START:84|2003-08-04 00:02:23 IN:92|2003-08-04 00:02:23 OUT:92|2003-08-04 00:08:37 FINISH:94
Vehicle 234 = 2003-08-04 14:00:06 START:94|2003-08-04 14:00:38 IN:92|2003-08-04 14:00:38 OUT:92|2003-08-04 14:03:14 FINISH:84
Vehicle 234 = 2003-08-04 19:00:06 START:84|2003-08-04 19:01:08 IN:92|2003-08-04 19:01:08 OUT:92|2003-08-04 19:07:22 FINISH:94
Vehicle 234 = 2003-08-04 19:45:06 START:94|2003-08-04 19:50:06 IN:92|2003-08-04 19:50:06 OUT:92|2003-08-04 19:52:43 FINISH:84
Vehicle 234 = 2003-08-05 00:00:06 START:84|2003-08-05 00:01:44 IN:92|2003-08-05 00:01:44 OUT:92|2003-08-05 00:07:59 FINISH:94
Vehicle 234 = 2003-08-05 14:00:06 START:94|2003-08-05 14:01:02 IN:92|2003-08-05 14:01:02 OUT:92|2003-08-05 14:03:39 FINISH:84
Vehicle 234 = 2003-08-05 19:00:06 START:84|2003-08-05 19:00:55 IN:92|2003-08-05 19:00:55 OUT:92|2003-08-05 19:07:08 FINISH:94
Vehicle 234 = 2003-08-05 19:45:06 START:94|2003-08-05 19:49:46 IN:92|2003-08-05 19:49:46 OUT:92|2003-08-05 19:52:24 FINISH:84
Vehicle 234 = 2003-08-06 00:00:06 START:84|2003-08-06 00:00:12 IN:92|2003-08-06 00:00:12 OUT:92|2003-08-06 00:06:26 FINISH:94
Vehicle 234 = 2003-08-06 14:00:06 START:94|2003-08-06 14:03:01 IN:92|2003-08-06 14:03:01 OUT:92|2003-08-06 14:05:38 FINISH:84
Vehicle 234 = 2003-08-06 19:00:06 START:84|2003-08-06 19:02:28 IN:92|2003-08-06 19:02:28 OUT:92|2003-08-06 19:08:42 FINISH:94
Vehicle 234 = 2003-08-06 19:45:06 START:94|2003-08-06 19:49:02 IN:92|2003-08-06 19:49:02 OUT:92|2003-08-06 19:51:40 FINISH:84
Vehicle 234 = 2003-08-07 00:00:06 START:84|2003-08-07 00:02:19 IN:92|2003-08-07 00:02:19 OUT:92|2003-08-07 00:08:33 FINISH:94
Vehicle 234 = 2003-08-07 14:00:06 START:94|2003-08-07 14:04:19 IN:92|2003-08-07 14:04:19 OUT:92|2003-08-07 14:06:57 FINISH:84
Vehicle 234 = 2003-08-07 19:00:06 START:84|2003-08-07 19:01:41 IN:92|2003-08-07 19:01:41 OUT:92|2003-08-07 19:07:56 FINISH:94
Vehicle 234 = 2003-08-07 19:45:06 START:94|2003-08-07 19:48:05 IN:92|2003-08-07 19:48:05 OUT:92|2003-08-07 19:50:42 FINISH:84
...
Vehicle 234 = 2003-08-23 00:00:06 START:84|2003-08-23 00:01:49 IN:92|2003-08-23 00:01:49 OUT:92|2003-08-23 00:08:03 FINISH:94
Vehicle 234 = 2003-08-23 14:00:06 START:94|2003-08-23 14:05:04 IN:92|2003-08-23 14:05:04 OUT:92|2003-08-23 14:07:41 FINISH:84
Vehicle 234 = 2003-08-23 19:00:06 START:84|2003-08-23 19:02:25 IN:92|2003-08-23 19:02:25 OUT:92|

```

Figure 4: Example of cookies accumulated by a simulated vehicle. In this example, each intersection in the network is equipped with a roadside controller, and communicates the time that the vehicle enters the range of the controller, and the time that the vehicle accelerates to driving speed on its way to the next intersection. The very last cookie in the list is the current “active” cookie, while all other cookies in the list are used only in response to information queries.

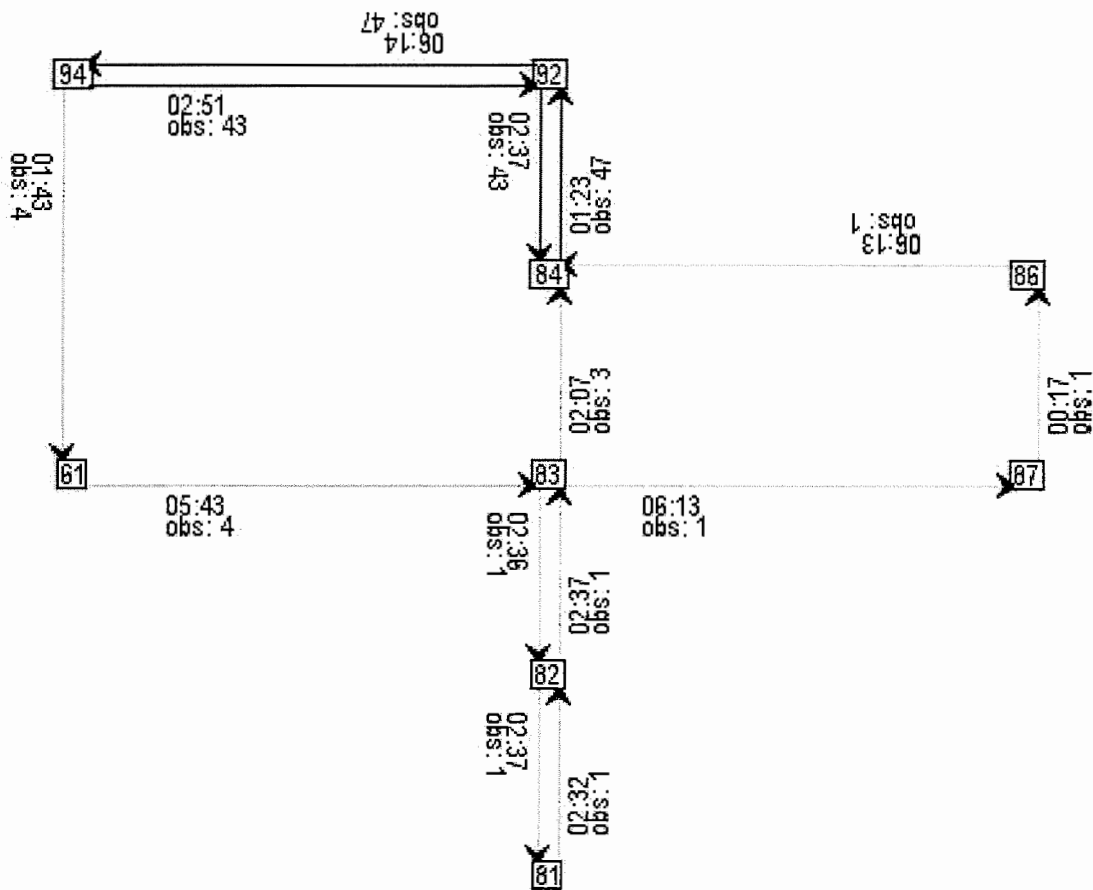


Figure 5: A single vehicle’s history of traffic cookies, rendered as a network without regard to time or prior nodes in a trip (global aggregate). (Note this is not the same vehicle shown in figure 7.) This figure is analogous to predicting a trip when one only knows *which* vehicle is traveling, but not when and not where it is at the moment, or what its prior nodes were. This kind of an aggregation would not be used in practice, but is helpful to demonstrate how each vehicle’s collection of PTCs could be used.

not been conditioned on any prior information. That is, it is simply the summation of travel by the vehicle on each link. This map would never be used in practice, because once some information about a trip is known, it can be used to condition the full database to generate the most likely downstream paths and conditions.

Figure 6 shows a similar network representation of the travel history, but this time for all vehicles. Again, this figure is not directly used by the PTC system, but rather is included to illustrate some basic concepts. First, a single vehicle does not hold as much information as the entire population of vehicles, even after 30 days of simulation. Rather, it only contains information on the traffic conditions and routes that it has observed in the past. Compare the link from node 84 to node 92 in figure 5 and figure 6. Even though the single vehicle has traversed that link 47 times in the simulated 30 days, its average observed travel time is different than the average as seen by the entire population of vehicles (in this case, 1000 vehicles traveling for 30 days). This is obvious,

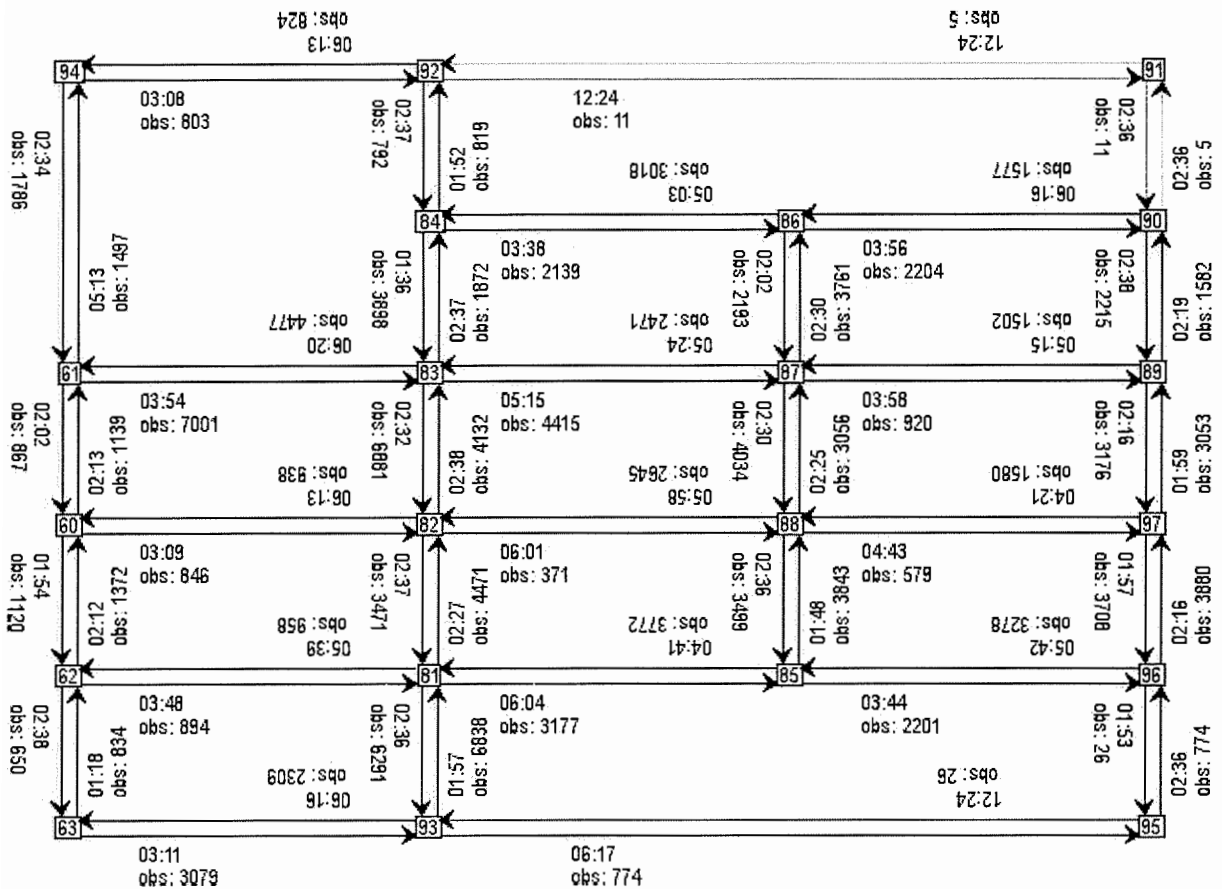


Figure 6: The aggregation of all 1000 simulated vehicles' travel histories for the 30 simulated days of travel. Comparing to the previous figure, all links in the network are used.

but important. When a vehicle arrives at a detection point, its historic information is likely to be highly correlated with the conditions prevalent at that moment, even without conditioning the data request on the time of day. Travelers who generally travel during rush hour have information about rush hour, travelers who generally travel in off peak times or traveling against the prevailing flow of traffic have a different view of traffic conditions (and consequently most likely make different route choices). In practice it may be necessary to explicitly condition queries on time of day at very busy intersections. This is not a problem, as adding extra conditioning information when calculating probabilities is conceptually very easy to add to the proposed system.

This illustrates an important strength of the PTC system that to our knowledge has never been done before. At all times, the current estimate of the state of the system is based only on information that is relevant to the *current* vehicles in the system. Sunday drivers or contra-flow commuters are *not* used in an estimate of how a congested corridor is behaving. This filtering of the data is provided for free from a computational standpoint. The vehicles that show up are the ones that are important and the ones who provide the data being used. Vehicles that don't show up are irrelevant, but their data is conveniently unavailable.

In practice, instead of the complete historical averages shown in figure 6, each roadside device will only query the vehicles that are present at any given time, and compute averages based

on those vehicles. For a single vehicle, this will generate a downstream likelihood tree based on the particular vehicle's immediately preceding path, such as that shown in figure 7. In this figure, the querying node is node 81, and the immediate prior node is node 93. If the vehicle was moving in the opposite direction, then another tree would result.

The conditional vehicle histories can then be aggregated to obtain an estimate of the most likely destinations for an entire platoon of vehicles. Counting each vehicle as one vehicle (equal weighting), and then assigning a fractional vehicle to each downstream link based on each vehicle's historical percentages gives a probability tree similar to the one shown in figure 8. That figure includes all vehicles passing through intersection 81 from intersection 93, for the *ten minute* period which contains the trip of the vehicle shown in figure 7. Similar diagrams can be made for any period of time, from one minute to one hour or longer, for all intersections for any time step in the simulation. We used ten minutes in this figure only because our simulated travel data is rather sparse, containing as it does only 1,000 total travelers. The downstream travel probabilities can also be computed as running averages, and so on, depending upon the needs of the traffic control application.

This figure underscores the natural filtering of data that occurs in the PTC system. The probability tree of figure 8 is very different from the complete historical average shown in figure 6. This is due to two effects. First, the system naturally is polling only the current travelers for the most likely current destinations and trip paths—non-travelers are not included. Second, those travelers generally will have historic data that is most relevant to the current time of day—to the extent that individuals' travel and activity patterns have daily and weekly periodicity. For example, if traffic is light in the morning and heavy in the evening, then a historical average will estimate a middle time for the expected travel time. But if individual vehicles go one way in the morning and the other way at night, then from the point of view of the vehicles, traffic conditions are fairly consistent from day to day, and are much closer to the "heavy" numbers (slow speeds and long durations). One area that we have not yet tested adequately is determining how much additional specificity is gained by including time (day of the week, or as a time of day, and so on) when conditioning the past histories of each vehicle.

## 5 Some practical applications for the PTC system

Beside being practical and computationally feasible, the method has a number of benefits over other approaches and methods to estimating trip tables.

- The in-vehicle devices are a transparent-to-the-traveler method for enabling vehicle tracking, since each driver can decide whether or not to keep the device on, whether to delete sensitive information, and so on.
- By only using data that has been cryptographically signed, the roadside computers can trust the authenticity data that they are using to develop their predictions. This also removes the need for the vehicles to have GPS antennas, as that data is (currently) unverifiable and therefore untrustworthy.
- The written data is cryptographically *signed*, but not encrypted. This enhances the system's transparency to the driver, and hopefully boosts each driver's confidence in the data with

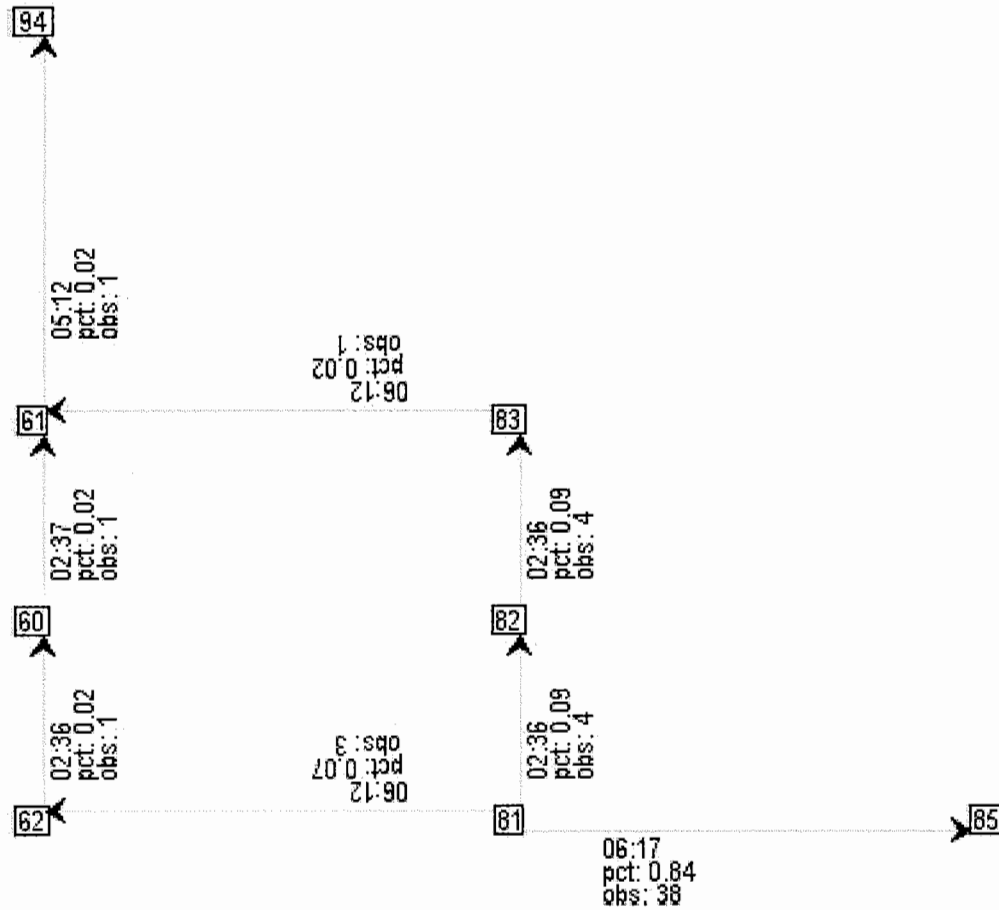


Figure 7: A vehicle’s downstream probability tree, for a vehicle approaching intersection **81** from intersection **93** (not shown here, but compare with figure 8). The probability tree has been rendered as a network for ease of viewing. The most likely movement for this vehicle is to turn right and proceed to node 85, where it will end its trip, but there is a small (2 percent) chance that it will travel to the exact opposite corner of the network, due to one past trip to node 94. The time entries are the average historical travel times observed, the percents are the percentage of total trips which contained a particular link, and the observations are the absolute count of the number of times a link was traversed.

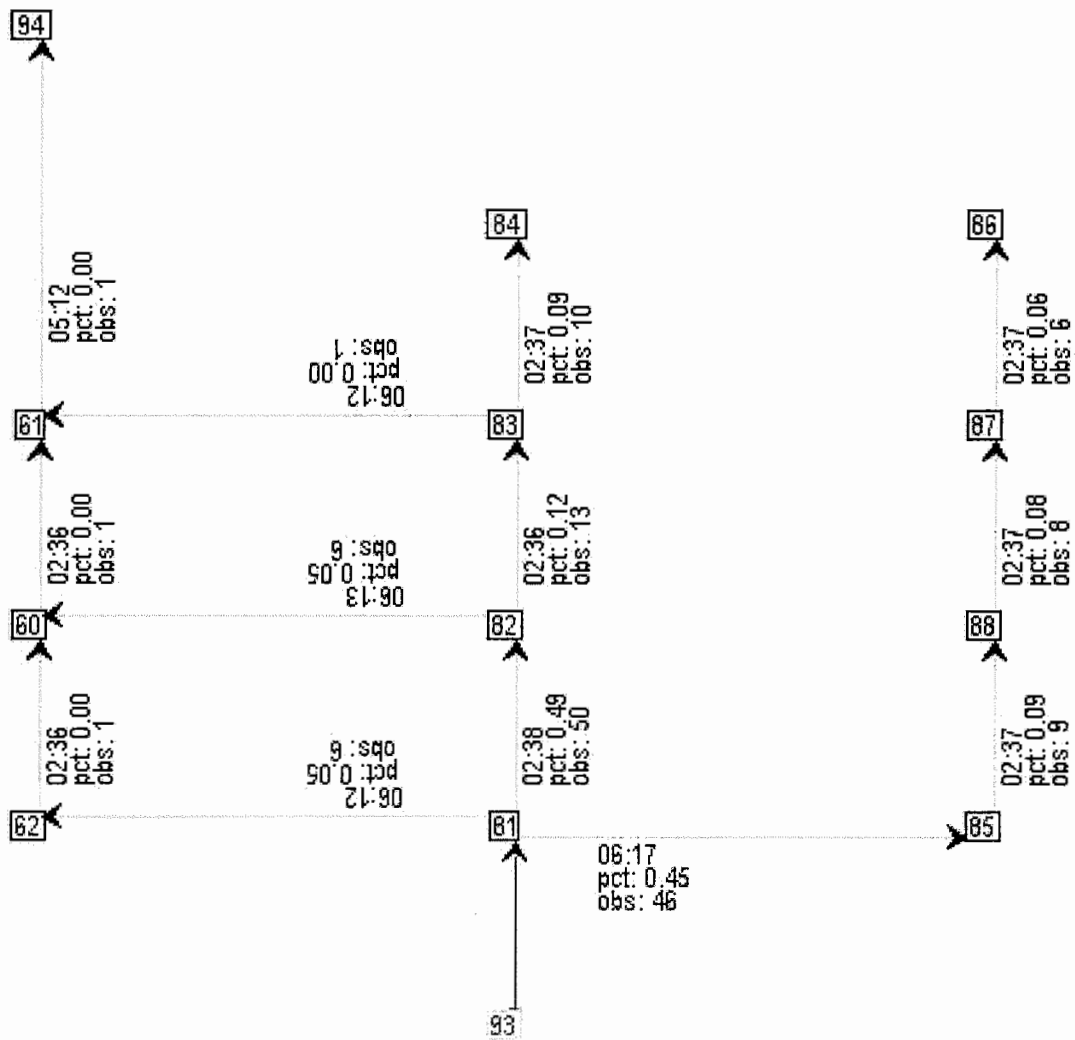


Figure 8: The probable destinations of all vehicles approaching intersection **81** from intersection **93**, observed over a ten-minute period. This figure is for the same day and time in the simulation as figure 7, and includes that vehicle’s history. The group splits almost evenly between right turns and through movements, and about one-third stop traveling at node 85, and one-third stop at node 82. The time entries are the average historical travel times observed by each vehicle (weighting each vehicle’s average observations equally), the percents are the percentage of the original platoon of vehicles which can be expected to traverse the link in question, and the observations are the sum of the fractional vehicles assigned to each link based on each vehicle’s own history, rounded up to the nearest integer.

respect to privacy concerns. Also this transparency may allow third parties to develop applications which can make use of the data.

- A standard set of algorithms can be developed to provide travelers with real-time travel advice based on the consistent, locally stored data set. This might encourage wider adoption of suitable in-vehicle devices.
- Additional revenue streams are possible for device providers who wish to customize information that is particularly valuable to a vehicle, given its past history.
- Since the vehicle route predictions are not required for operation of intersection signals, etc., and since the detection and communication with vehicles does not require cutting up pavement as with loop-detectors, there might be a business model for a third party to invest in the roadside devices, and sell the aggregated information to the local traffic authority. This would further improve driver confidence that the information about their particular trips would not be used to issue them a traffic citation.
- Instead of asking travelers detailed questions about where they went, travel demand surveys can become rather painless. Surveys could be accomplished simply by downloading the data from an in-vehicle computer. The cryptographic signatures would give much more confidence in the accuracy of the survey data.
- As ad-hoc local area wireless networking becomes more sophisticated, the signed data can form the basis of an inter-vehicle cooperative control system, in which vehicles with similar destinations or travel patterns could identify each other and self-organize into coherent platoons.

The PTC system has applications that extend far beyond the estimation of the instantaneous network-wide trip table. For traffic control purposes, each intersection will “know” which downstream intersections are most affected by its local control decisions, leading to better system-wide optimization of traffic control. Each in-vehicle device might be provided to the drivers as a way to offer real-time travel advice. That travel advice can be tailored to fit the vehicle’s travel histories, and can use real-time network information that might also be communicated by the roadside network of computers.

## 6 Conclusion

A novel distributed method for estimating a trip table in real time has been proposed. The system is called persistent traffic cookies, or PTC, by analogy with the use of cookies by web servers to keep track of the current state of all web browsers navigating a web site. Our method uses traffic cookies to maintain the state (current trip) of vehicles through the system. These cookies are persistent from day to day, and therefore can be used to generate historical information about a vehicle. The method leverages the vehicles themselves to store their own travel data, and then physically carry that data around the network. Serendipitously, wherever a vehicle travels is also exactly where the network control system needs to know the history of that exact vehicle, in order to predict downstream impacts of local control actions. Finally, the system as proposed *could* be

implemented using today's off the shelf computers, and standard local area wireless protocols such as 802.11b, rather than relying upon cutting edge technologies and custom protocols.

We foresee this method being used to generate trip tables which can be used as an input to a DTA type control algorithm. Without 100% adoption rates, the trip tables estimated are just a sample of the complete trip table, and therefore it makes sense to plug these estimates into a DTA-type process to derive the most likely trip table. In that way, any small amount of information can be used to assist the DTA solution. In addition, there are many other potential applications for the PTC system, including applications which provide services to the driver.

This paper has described the system, and presented the results of an initial simulation study designed to illustrate the concept. Future work will include more careful simulation of travel, using the real Orange County freeway and arterial networks, limited adoption rates, and a sparser network of roadside nodes, to determine the scalability and practical feasibility of the system. In addition, work is planned to integrate the decentralized information that is available into a DTA system.

## 7 Acknowledgment

This work is currently funded by a grant from the National Science Foundation Award number 0339307, "A Distributed On-Line Database System for Transportation Management Using Cooperating Roadside and In-Vehicle Communication Devices."

## References

- Jeffrey L. Adler and Victor J. Blue. A cooperative multi-agent transportation management and route guidance system. *Transportation Research, Part C: Emerging Technologies*, 10C(5-6): 433-454, October 2002.
- David Bretherton, Keith Wood, and Neil Raha. Traffic monitoring and congestion management in the scoot urban traffic control system. *Transportation Research Record*, 1634, 1998.
- Ruey Long Cheu, Chi Xie, and Der-Horng Lee. Probe vehicle population and sample size for arterial speed estimation. *Computer-Aided Civil & Infrastructure Engineering*, 17(1):53-60, January 2002.
- Benjamin Coifman. Estimating travel times and vehicle trajectories on freeways using dual loop detectors. *Transportation Research, Part A: Policy and Practice*, 36A(4):351-364, May 2002.
- Nathan H. Gartner, Farhad J. Pooran, and Christina M. Andrews. Implementation of the opac adaptive control strategy in a traffic signal network. In *2001 IEEE intelligent transportation systems proceedings. Piscataway, N.J.*, pages 195-200. IEEE Press, 2001.
- Bruce Robert Hellenga. Estimating dynamic origin-destination demands from link and probe counts. *Transportation Research Part A: Policy and Practice*, 31(1):83-83, January 1997. URL <http://www.sciencedirect.com/science/article/B6VG7-3SWT2RP-44/2/1daa1e0%7a359a885efcb7bf6f8967123>.

- Josefa Z. Hernandez, Sascha Ossowski, and Ana García-Serrano. Multiagent architectures for intelligent traffic management systems. *Transportation Research, Part C: Emerging Technologies*, 10C(5–6):473–506, October 2002.
- Iteris, Inc. National ITS architecture. Technical report, US Department of Transportation, April 2002. URL <http://itsarch.iteris.com/itsarch/>.
- Shubhalaxmi Kher, Sanjiv Tokekar, and P.K. Chande. Self sustaining traffic management system and its compartmental modeling. In *Proceedings. The IEEE 5th International Conference on Intelligent Transportation Systems*, pages 639–644. IEEE, 2002.
- Filippo Logi and Stephen G. Ritchie. A multi-agent architecture for cooperative inter-jurisdictional traffic congestion management. *Transportation Research, Part C: Emerging Technologies*, 10C(5–6):507–527, October 2002.
- D. Moriarty and P. Langley. Learning cooperative lane selection strategies for highways. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence*, pages 684–691, Madison, WI, 1988. AAAI Press.
- D. E. Moriarty, S. Handley, and P. Langley. Learning distributed strategies for traffic control. In *Proceedings of the Fifth International Conference of the Society for Adaptive Behavior*, pages 437–446, Zurich, Switzerland, 1998. Society for Adaptive Behavior.
- Cheol Oh and Stephen G. Ritchie. Anonymous vehicle tracking for real-time traffic surveillance and performance on signalized arterials. *Transportation Research Record*, in press, 2003.
- Cheol Oh, Stephen G. Ritchie, Jun-Seok Oh, and R. Jayakrishnan. Real-time origin-destination (od) estimation via anonymous vehicle tracking. In *The IEEE 5th International Conference on Intelligent Transportation Systems*, pages 582–585. IEEE Press, 2002a. URL [http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?isNumber=22289&prod=CNF&arnumber=1041283&arSt=+582&ared=+586&arAuthor=Cheol+Oh%3B+Ritchie%2C+S.G.%3B+Jun-%Seok+Oh%3B+Jayakrishnan%2C+R.&arNumber=1041283&a\\_id0=1041283&a\\_id1=1041284&a\\_id2=1041285&a\\_id3=1041286&a\\_id4=1041287&a\\_id5=1041288&a\\_id6=1041289&a\\_id7=1041290&a\\_id8=1041291&a\\_id9=1041292&a\\_id10=1041293&a\\_id11=1041294&a\\_id12=1041295&a\\_id13=1041296&a\\_id14=1041297&count=15](http://ieeexplore.ieee.org/xpls/abs_all.jsp?isNumber=22289&prod=CNF&arnumber=1041283&arSt=+582&ared=+586&arAuthor=Cheol+Oh%3B+Ritchie%2C+S.G.%3B+Jun-%Seok+Oh%3B+Jayakrishnan%2C+R.&arNumber=1041283&a_id0=1041283&a_id1=1041284&a_id2=1041285&a_id3=1041286&a_id4=1041287&a_id5=1041288&a_id6=1041289&a_id7=1041290&a_id8=1041291&a_id9=1041292&a_id10=1041293&a_id11=1041294&a_id12=1041295&a_id13=1041296&a_id14=1041297&count=15).
- Seri Oh, Stephen G. Ritchie, and Cheol Oh. Real-time traffic measurement from single loop inductive signatures. *Transportation Research Record 1804*, pages 98–106, 2002b.
- Jean Orselli. Scats : Sydney coordinated adaptive traffic system. *Transport, environment, circulation*, Sept.-Oct. 1995.
- Farhard J. Pooran, Philip J. Tarnoff, and Ravi Kalaputapu. RT-TRACS, development of real-time control logic. In *Proceedings of the 6th annual meeting of ITS America*, volume 1, 1996.
- Vaneet Sethi, Nikhil Bhandari, Frank S. Koppelman, and Joseph L. Schofer. Arterial incident detection using fixed detector and probe vehicle data. *Transportation Research Part C: Emerging Technologies*, 3(2):99–112, April 1995. URL

<http://www.sciencedirect.com/science/article/B6VGJ-3YK014K-N/2/3954a406%57e761121019db1ea5dfe9dd>.

- N. E. Thomas. Multi-state and multi-sensor incident detection systems for arterial streets. *Transportation Research Part C: Emerging Technologies*, 6(5-6):337–357, December 1998. URL <http://www.sciencedirect.com/science/article/B6VGJ-3WM4P0T-4/2/a5f36264%0e99e24446c4d732f1f32ae0>.
- M. Van Aerde, G. MacKinnon, and B. Hellenga. The generation of synthetic o-d demands from real-time vehicle probe data : potential and limitations. In *Vehicle Navigation & Information Systems Conference proceedings, Dearborn, MI, 1991*, volume 2. Society of Automotive Engineers, Warrendale, PA, 1991.
- Ronald van Katwijk and Paul van Koningsbruggen. Coordination of traffic management instruments using agent technology. *Transportation Research, Part C: Emerging Technologies*, 10C(5–6):455–471, October 2002.
- Athanasios Ziliaskopoulos and Jiang Zhang. A zero public infrastructure vehicle based traffic information system. In *Proceedings of the 82<sup>nd</sup> annual meeting of the Transportation Research Board*, Washington, D. C., 2003.