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

James E. Marca, Craig R. Rindt and R. Jayakrishnan

Institute of Transportation Studies
University of California, Irvine
Irvine, CA 92967
tel: 949-824-6544
fax: 949-824-8385
email: [jmarca,crindt,rjayakri]@uci.edu

March 1, 2006

Keywords: traffic monitoring, probe vehicle systems

Wordcount: 5,400 words + 7 figures = 7,150 words

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

James E. Marca, Craig R. Rindt and R. Jayakrishnan

Abstract. A novel, distributed method for estimating a trip table in real time is described. 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 web browsers navigating a web site. The method uses traffic cookies placed on in-vehicle computers to maintain the state (current trip) of vehicles moving through the system. These cookies are persistent from day to day; taken together they form a complete travel history for a traveler or vehicle. The method leverages the vehicles themselves to store their own travel data, and then physically carry that data around the network. Advantages include scalability in both storage and computational effort, as well as the unique ability to incorporate the travel behavior of individuals into real-time traffic predictions. A small scale simulation is presented to illustrate the concept and its potential applications.

INTRODUCTION

With the advent of increasingly sophisticated traffic management systems, such as those incorporating Dynamic Traffic Assignment, more stringent demands are being placed upon the available input data. Specifically, traffic monitoring inputs are being used to estimate a real-time origin-destination (O-D) matrix. The usual approach to monitoring traffic is to use an infrastructure-based vehicle detection system, transmit that information to a central analysis center, and then process that data to extract the variables of interest. 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.

To address the increased processing needs, one major improvement in detection is the move from anonymous point measurement of aggregate flow, speed, and occupancy variables to vehicle-specific measurements of routes, travel times, and general vehicle dynamics. Examples of these enhanced detection systems include video detection systems, the use of toll tags. 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, (3) store the collected data, (4) process the data (which will be sizable in large systems) for use in traffic management applications, (5) transmit the collected data and/or the processing results to local control hardware, and (6) address privacy concerns that vehicle tracking systems are sure to raise.

This paper describes an approach to solving these problems by making the traveler a partner in the process of traffic monitoring and control. In the system described here, each participating traveler has a device which stores his or her movements through the transportation network. This system (1) uniquely identifies each device over time, (2) tracks each participating traveler over multiple trips, for arbitrary lengths of time, (3) stores travel history data within each compliant device, (4) performs at least some processing of the data within each compliant device, (5) relies upon the natural movement of vehicles to "hand carry" the data to where it is needed the most—at the local controllers, and (6) address privacy concerns by allowing the traveler to turn off, delete, or otherwise restrict participation in the system.

The proposed system is quite simple: each mobile device collects a cookie each time it establishes a connection with a roadside device. These traffic cookies record the current time, and the current roadside controller. The cookies are written such that taken together they describe the complete travel history of the vehicle, and can be read and authenticated by any *other* roadside device in the system. Finally, the cookies are persistent from trip to trip. This leads to the system's name *persistent traffic cookies*, or PTC.

This paper is organized as follows. The next section describes the PTC method for tracking vehicles and predicting traffic flow conditions. This is followed by a section which describes the results of an illustrative simulation. Then a brief overview of current approaches to traffic monitoring is given, followed by a section that describes how the proposed PTC offers a unique approach to traffic monitoring. The penultimate section lists some practical applications of a working, system-wide PTC network, and the paper closes with a short conclusion.

A DESCRIPTION OF THE PTC SYSTEM

Figure 1 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 1 about here.]

Figure 2 shows *one* method for establishing the necessary exchange of data between a participating vehicle and the roadside computer. The first step is to establish a local area wireless communication link. Then 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 verifies the cookie against the upstream node's signature. Upon verification, the roadside computer adds its own node name and the current time to the active cookie, signs it with its own key, and returns this revised cookie and the new signature to the vehicle. This cookie and signature will be processed in turn by the next node along the vehicle's path. A subset of cookies collected in this manner for a single vehicle in a simulation experiment is shown later in figure 3.

[FIGURE 2 about here.]

After writing out the current cookie to the vehicle, the roadside computer can query the vehicle for past travel behavior. 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. The roadside computer might request all stored cookies which contain the node sequence y, x. 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. Other data queries are possible. The conditional sequence could be the current trip origin plus the current intersection. Or to estimate possible demands later in the day, the query might ask for cookies going the other way, from $z_{1..3}$ to x, or from x to the trip's origin. The returned list of cookies can be used to estimate the most likely time the vehicle might return that day.

A SIMULATED EXAMPLE

This section describes the results of a small simulation of a PTC system in operation. The simulation consists of $1,000$ artificial individuals, traveling over a 30 day period. All individuals pursued individual activity patterns in an artificial travel and land use network generated using an experimental activity pattern generator *(13, 9)*.

There are two important items to note concerning this simulation. First, in order to properly exercise the PTC concept, it is not enough to generate random trips between origins and destinations. If this were to be done, there would be no continuity from trip to trip, and day to day. The PTC system presumes that vehicles store their travel history on-board and carry it with them. A random trip generated from an origin to a destination in response to some trip table rate will eliminate the requirement that we be able to track individual vehicles from trip to trip.

Second, tracking vehicles from trip to trip is also not sufficient to exercise the PTC concept. If a vehicle were assigned a random trip start time and trip destination (taking the current location as the trip origin), then the PTC system will have nothing to "learn." That is, each vehicle will carry with it a history that is comprised of random trips. Instead, the PTC concept presumes that people typically do not make random trips, but instead have an origin, a desintation, and an activity goal in mind.

For this reason the random activity simulator was used as a trip generation module. Since a personal travel database only makes sense in the context of the habitual, systematic behavior generated by "realistic" activities, the activity simulator includes a fixed schedule of activities for each household, which the household members attempt to complete. In addition, each household also has a number of random activities, some of which have high priority, others of which have low priority and are optional within a day. The real-world analogues are going to the dentist (random, high priority) or buying milk (low priority, more or less optional on any given day). While the acitivity simulator is far from perfect, it provides a nice way to generate trips that "look like" real trips. Further explanation can be found in *(13)*.

Finally, for this example simulation, it is assumed 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 are always perfect.

Figure 3 shows an example of the history a single vehicle might collect. This figure shows data stored for a single vehicle on day 24 of the simulation. Records have been removed from the figure to conserve space.

[FIGURE 3 about here.]

Figure 4 shows an example of the history a single vehicle might collect over 30 days, mapped onto a network representation to aid visualization. The data shown are simply the summation of travel by the vehicle on each link, and have not been conditioned on prior links, time of day, etc.

[FIGURE 4 about here.]

Figure 5 shows a similar network representation of the travel history, but this time for all vehicles. This figure 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. Furthermore, an individual's view of a link's conditions can be decidedly different than the global average. Compare the link from node 84 to node 92 in figure 4 and figure 5. 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).

[FIGURE 5 about here.]

At any given moment in the simulation, each roadside device will only query the vehicles that are present at any given time, and will compute expectations 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 6. In this figure, the querying node is node 81, and the immediate prior node is node 93.

[FIGURE 6 about here.]

The conditional vehicle histories are aggregated to obtain an estimate of the most likely turning fractions, routes, and ultimate destinations for a whole 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 7. 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 6. 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. The downstream travel probabilities can also be computed as running averages, and so on, depending upon the needs of the traffic control application.

[FIGURE 7 about here.]

Figure 7 underscores the natural filtering of data that occurs in the PTC system. The route and destination probability tree is very different from the complete historical average shown in figure 5. This is due to two effects. First, the system is polling only the current travelers—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 those individuals' have traveled at the same times in prior trips. Further research will explore whether this natural conditioning is good enough, or whether the roadside data query should trip origins, time of day, day of week, or other conditions.

BRIEF REVIEW OF CURRENT TRAFFIC MONITORING APPROACHES

The move towards sophisticated traffic management applications demand an equivalent improvement in traffic measurement technologies. Limitations of point-based measurement form significant roadblocks on the way towards truly adaptive traffic control. In general, state measurement for traffic systems involves both supply-side measurement (speed, flow, occupancy, and travel times) and demand-side measurement (Origin-Destination (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. 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 concerned with improving sensing in transportation systems to obtain the richer information necessary for reliable system-wide travel demand estimates and predictions. Most of these require expensive new hardware, such as video surveillance cameras, toll-tag reading gantries, and so on. One very interesting avenue of research is to leverage the existing installed base of ILDs to identify each vehicle's (presumably) unique inductance signature to effect vehicle reidentification from sensor to sensor *(12, 10)*. The reidentification data can then be used to measure (supply side) link travel times *(3)* as well as estimate path and O-D level demands *(11)*.

Another way to improve traffic monitoring is to use vehicles as traffic probes. This is very similar in spirit to the PTC system, and so deserves special attention. There are several proposed applications for probe vehicle fleets, such as network speed and travel time estimation *(2)*, incident detection *(15, 14)*, and O-D estimation *(4)*. The expected results are qualified, however, by a consideration of the necessary size of the probe vehicle fleet. For example, *(16)* 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. In this vein, *(4)* 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.

The difference between the PTC system and probe vehicle systems is the treatment of probe vehicles as read-only, passive sensors. This view appears to be fundamental to the ITS community's perception of probe vehicles as embodied in the National Intelligent Transportation Systems Architecture (NITSA) *(6)*. The PTC approach, in contrast, argues that vehicles should be considered active, writable sensors with significant data storage and processing capabilities. In a perfect world, this eliminates the need to upload and store each vehicle's travel paths, which in turn eliminates a large portion of the financial and computational costs of a probe vehicle system. Further, it is hoped that the presence of a standard format for storing travel histories will spur third-party applications for route guidance, etc. These applications will add value to the system, and possibly boost the adoption rate of PTC-compliant devices.

INTEGRATING PTC WITH EXISTING APPROACHES

The PTC system is grounded in a decentralized view of traffic management. It is closest to an operational decomposition of traffic management components, such as those proposed by *(17)*, *(5)*, and *(18)*. However, since the basic unit is the roadside controller, multiple, simultaneous systems could exist in parallel, which would allow a jurisdictional decomposition of the system along the lines of *(8)* and *(1)*. Within the decentralized framework, each vehicle communicates with the nearest roadside controller. Each such wireless network acts independently of all others. New roadside nodes can be added to the system without needing to modify any existing nodes. This project is not yet considering vehicle to vehicle exchanges of information, as is proposed by *(18)* and *(7)*. Instead, the wireless communication system requires only a *single* wireless hop, which is identical to what most entry-level WiFi cards can handle. Trying to route messages in an ad hoc manner through other vehicles, while interesting, greatly complicates the networking software layer.

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

Vehicle identification

Problem: Unique identification First, there is the problem of identifying each vehicle uniquely. Current approaches use external surveillance techniques (video processing, magnetic signature analysis, etc.), or else use radio frequency identification (RFID) tags. The former present computational difficulties, while the latter introduce privacy concerns.

Solution: IDs are not necessary The PTC system does not need any vehicle identification, as such. 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 to guarantee that the history the vehicle reports is authentic.

Not requiring unique ids is an aid to privacy. The user can reset the PTC device at any time, can change the wireless radio, and so on. Systems that require a unique match between the traveler and past data collection, such as RFID tags, video monitoring, and so on, do not allow these privacy enhancing steps to take place. This may help make the PTC system more palatable to the traveling public than an external detection system.

Vehicle tracking

Problem: Vehicle tracking has large storage and computation requirements 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 current approach 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 uploaded to a central server in order to be matched and tracked through each detector point. This is expensive in terms of bandwidth, computation, and storage.

Solution: Let vehicles track themselves The PTC solution is to track vehicles by having the vehicles track themselves. The problem of storing and transmitting the vehicle tracking data is eliminated entirely. The vehicles literally *carry* the data to the points (in time and space) at which the data are needed. 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.

Maintaining and querying a travel history for each vehicle

Problem: Creating a scalable, efficient travel history database The final problem with the current approaches to estimating a real-time trip table is the difficulty of storing and querying the data. If the advanced detector data (observed paths, etc) are stored over time to improve the trip table estimates, then that growing database will have to be queried for every intersection for every discrete time period. The results then need to be transmitted to all intersections to tweak local control strategies. Considering the modest size of the benefit (especially if the traffic system is already close to capacity), the storage, processing, and networking costs are likely to be prohibitive.

Solution: Decentralized storage and processing 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 and parse *all* of the records from each vehicle currently situated at the intersection. This has a natural upper bound based on the size and capacity of the intersection. Any existing wireline communication network capacity can be devoted to letting downstream nodes know what is coming, based on the immediate sample of vehicle cookies.

BENEFITS AND APPLICATIONS OF THE PTC SYSTEM

Beside being practical and computationally feasible, the PTC method has a number of unique benefits and applications. For example:

- a standard set of algorithms can be developed to provide travelers with real-time travel advice based on the consistent, locally stored data set;
- information service providers can compete to customize information that is particularly valuable to a vehicle, given its history;
- system operators can attempt to encourage participation by documenting the benefits of the system for each unique traveler, much like the supermarket club cards do.

The PTC system could be operated by an independent commercial entity. A company could invest in the roadside devices, and then sell the aggregated information to the local traffic authority. The presence of an intermediary between the traveler and the traffic agency would further improve driver confidence that the information about their particular trips would not be used to issue them a traffic citation or worse.

Finally, the PTC system allows the traffic control system to truly anticipate demands. Each intersection will "know" which downstream intersections are most affected by its local control decisions. This can be used to generate better system-wide optimization of traffic control.

CONCLUSION

This paper describes a novel distributed method for estimating a trip table in real time. The system is called persistent traffic cookies (PTC), and is analogous to the use of cookies by webservers. The method uses traffic cookies to maintain the state (current trip) of vehicles moving through the system. These cookies are persistent from day to day, and over time form a complete travel history for a traveler. The method leverages the vehicles themselves to store their own travel data, and then physically carry that data around the network. 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.

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

[1] 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.

[2] 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.

[3] 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.

[4] Bruce Robert Hellinga. 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`.

[5] 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.

[6] Iteris, Inc. National ITS architecture. Technical report, US Department of Transportation, April 2002. URL `http://itsarch.iteris.com/itsarch/`.

[7] Shubhalaxmi Kher, Sanjiv Tokekar, and P.K. Chande. Self sustainging traffic management system and its compartmental modeling. In *Proceedings. The IEEE 5th International Conference on Intelligent Transportation Systems*, pages 639–644. IEEE, 2002.

[8] 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.

[9] James E. Marca, Craig R. Rindt, and Michael G. McNally. A simulation framework and environment for activity based transportation modeling. *presented at the 79th Annual Meeting of the Transportation Research Board, Washington, DC*, 2000.

[10] 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.

[11] 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 Intellegent Transportation Systems*, pages 582–585. IEEE Press, 2002.

[12] Seri Oh, Stephen G. Ritchie, and Cheol Oh. Real-time traffic measurement from single loop inductive signatures. *Transportation Research Record 1804*, pages 98–106, 2002.

[13] Craig R. Rindt, James E. Marca, and Michael G. McNally. An agent-based activity microsimulation kernel using a negotiation metaphor. In *Proceedings of the 81st annual meeting of the Transportation Research Board*, Washington, D. C., 2003. TRB.

[14] 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`.

[15] 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`.

[16] M. Van Aerde, G. MacKinnon, and B. Hellinga. 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.

[17] 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.

[18] Athanasios Ziliaskopoulos and Jiang Zhang. A zero public infrastructure vehicle based traffic information system. In *Proceedings of the* 82^{nd} *annual meeting of the Transportation Research Board*, Washington, D. C., 2003.

LIST OF FIGURES

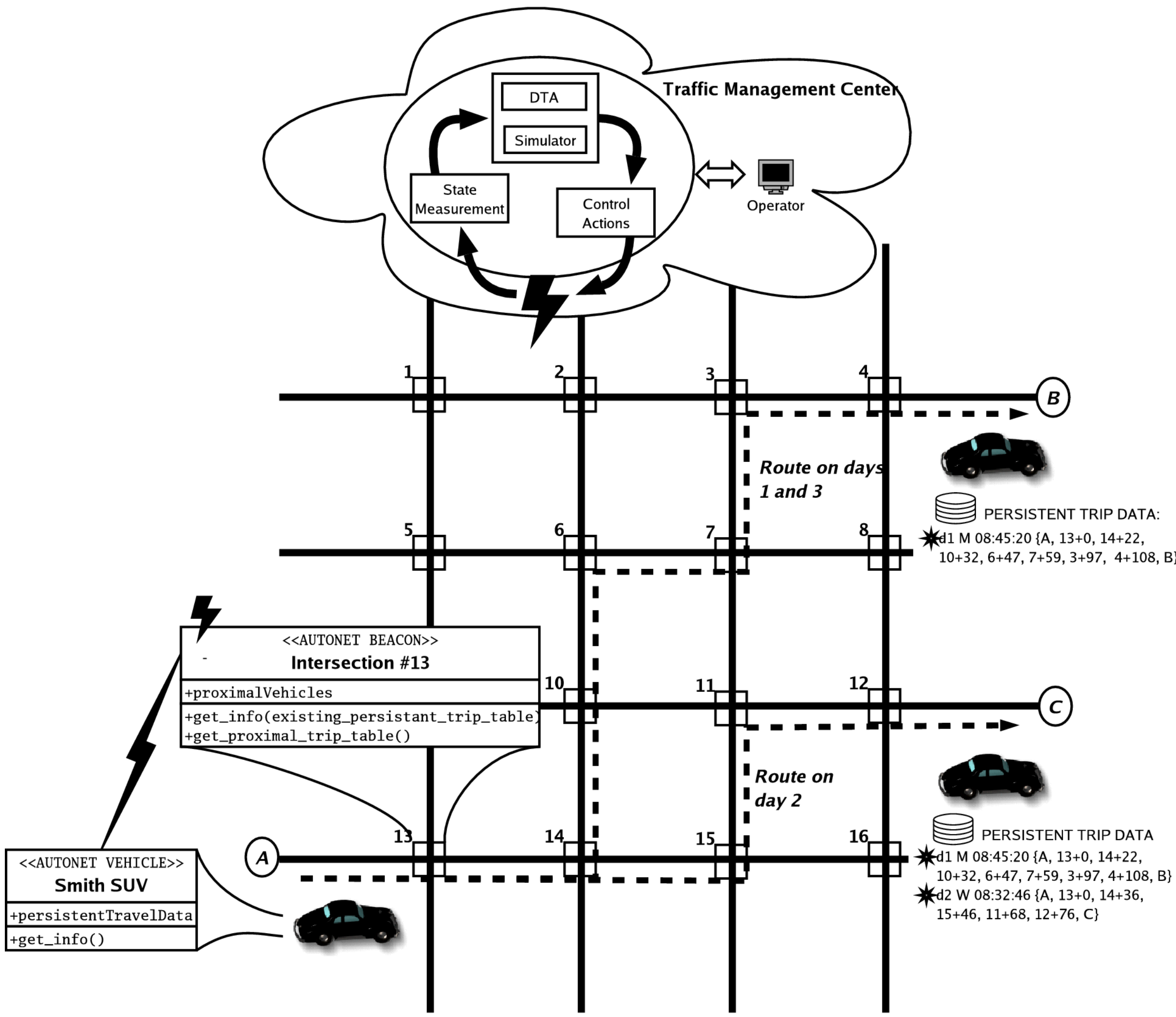


FIGURE 1: Example of the PTC system. Beacon #13 interacts with the vehicle on three consecutive days. The first day, the beacon retrieves no information, and writes a “traffic cookie” to the vehicle. Similar interactions occur at the other instrumented intersections. By the third day, beacon #13 can predict that the trip will end up at either B or C, with a 50/50 split. After day three, the splits are 67% and 33%, and so on.

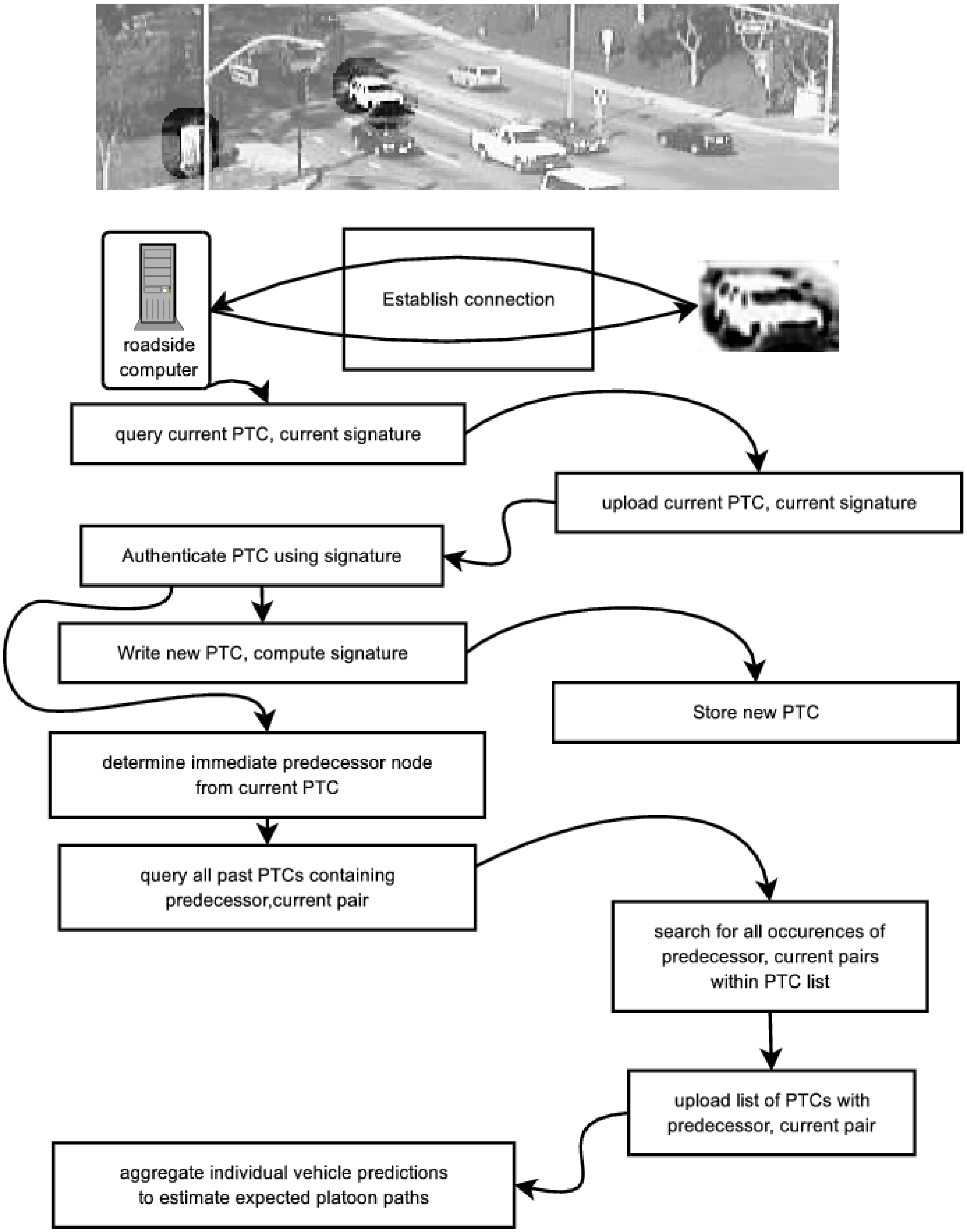


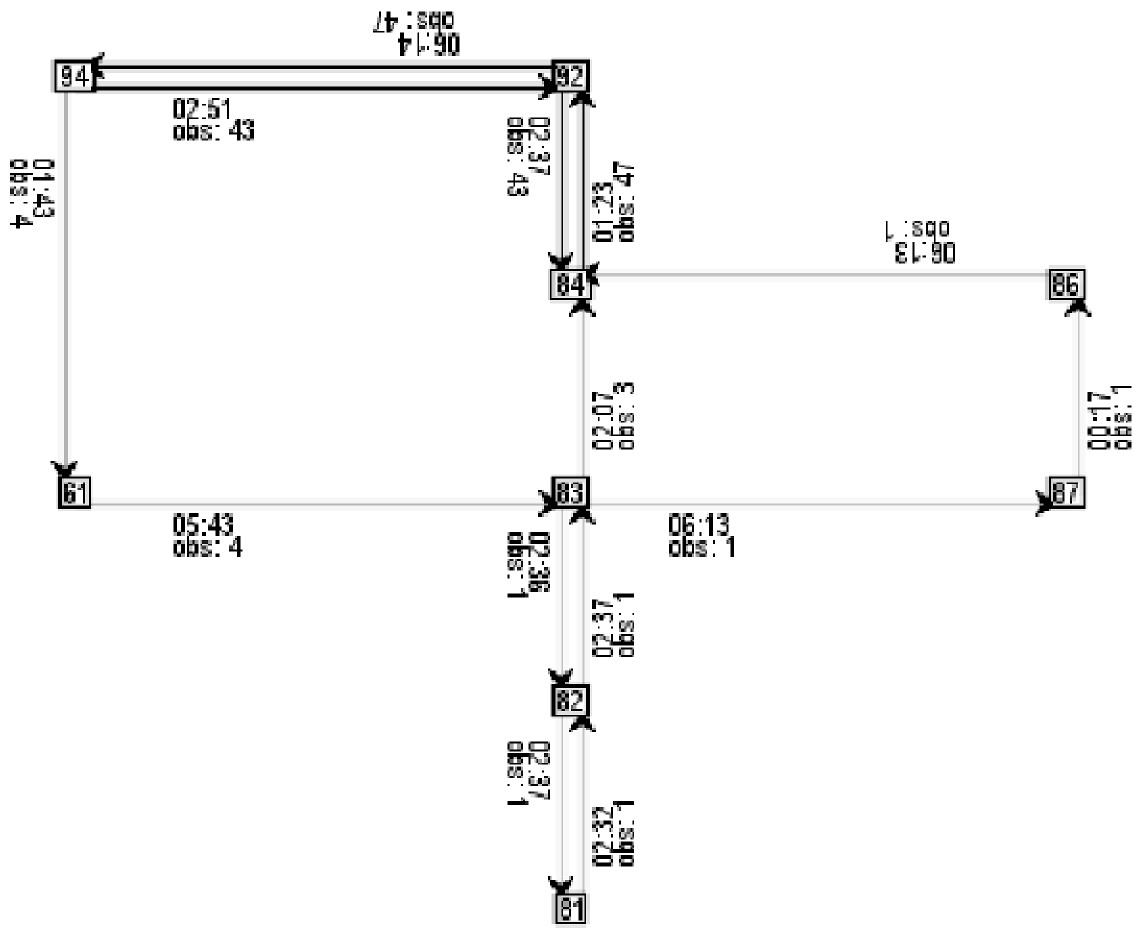
FIGURE 2: Flow of negotiation between the in-vehicle computer and the roadside computer

```
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|
```

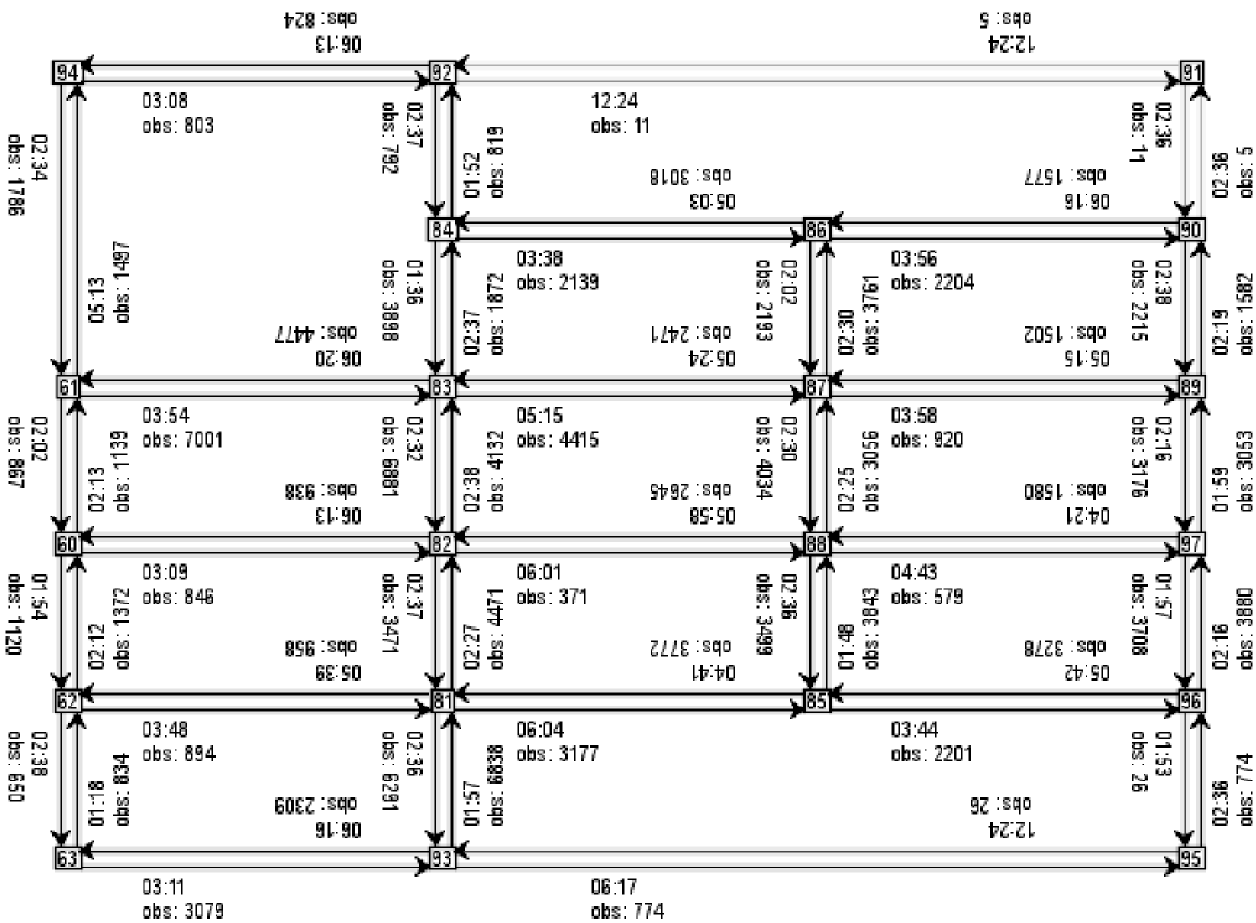
FIGURE 3: Example of cookies accumulated by a simulated vehicle, on day 24 of a 30 day simulation of 1000 vehicles. This is a list of records (cookies), one record for each complete trip that the vehicle has taken for each of the elapsed 24 days. Each record can be interpreted as a sequence of time stamps, from START to FINISH. Each time stamp is written by roadside beacons at each intersection. The intersection beacons write the time that the vehicle enters the range of the beacon (IN), and the time that the vehicle accelerates to driving speed on its way to the next intersection (OUT). The very last cookie in the list is the current "active" cookie, and does not have a 'FINISH' record; all other cookies represent past trips, and are used only in response to information queries. When a beacon decides to START a new trip, it also FINISHes the prior trip.



Key:

Avg. Link Time (MM:SS)
obs: count of link usage

FIGURE 4: A single vehicle's history of traffic cookies, rendered as a network (global aggregate). Refer to figure 6 for details on what each cookie contains. This figure is the aggregate of all of trips, formed by aggregating the information from each cookie. The links are labeled with the average observed travel time, and the numbers of observations made for that travel time. Links with very high numbers of observations have correspondingly good estimates of the mean travel time on a link. In this figure, no attempt has been made to condition the data based on time of day, or past nodes in a trip.



Key:

Avg. Link Time (MM:SS)
obs: count of link usage

FIGURE 5: The aggregation of all 1000 simulated vehicles' travel histories for the 30 simulated days of travel. Each link is labeled with the total mean travel time observed by all vehicles using the link, as well as the total numbers of observations (numbers of times each link was used as part of a trip). Compared to figure 4, the support for the mean travel time (the numbers of observations) is much stronger, and all links in the network are used.

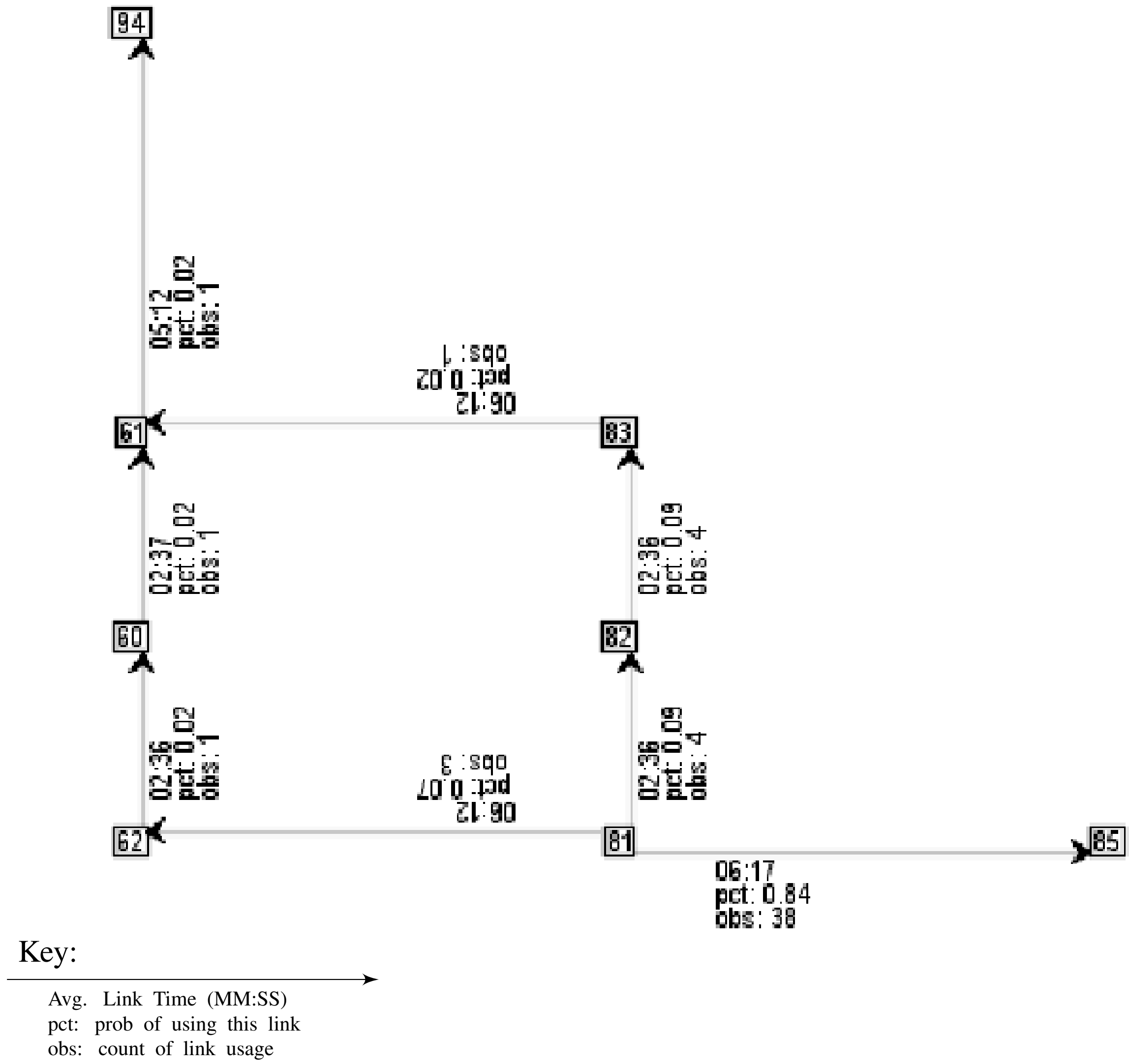
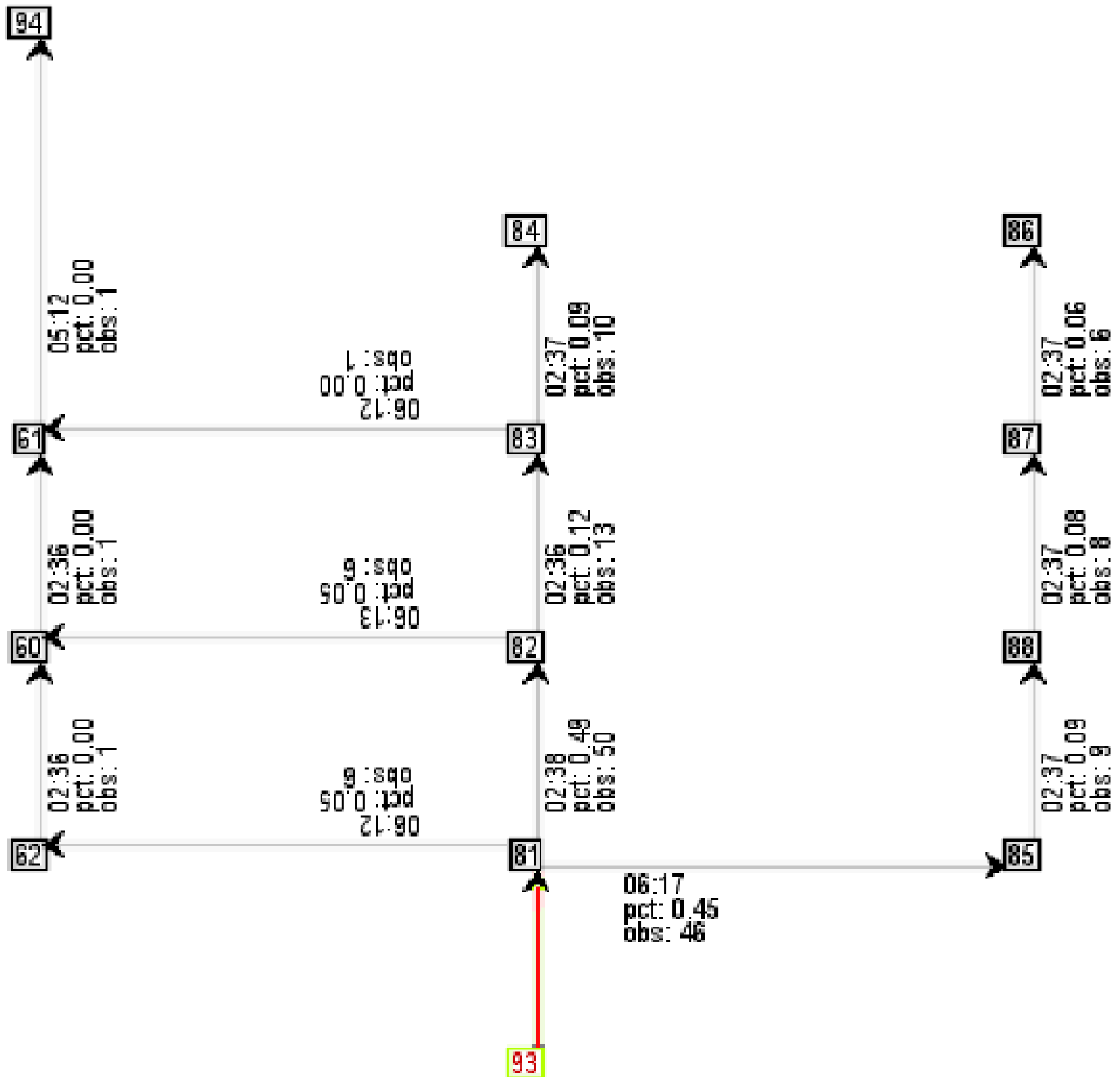


FIGURE 6: A vehicle's downstream probability tree, for a vehicle approaching intersection **81** from intersection **93** Compare with figure 7. The most likely movement for this vehicle is to turn right and proceed to node 85, where it will end its trip. The time entries are the average historical travel times observed using only the conditionally selected trips. The percents are the percentage of total trips which contained a particular link. The observations are the absolute count of the number of times a link was traversed in the conditionally observed set.



Key:

Avg. Link Time (MM:SS)
pct: prob of using this link
obs: count of link usage

FIGURE 7: 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 6, and includes that vehicle's conditional 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 conditional historical travel times observed by each vehicle (weighting each vehicle's conditional 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.