PARTNERS FOR ADVANCED TRANSPORTATION TECHNOLOGY

INSTITUTE OF TRANSPORTATION STUDIES
UNIVERSITY OF CALIFORNIA, BERKELEY

HYBRID TRAFFIC DATA COLLECTION ROADMAP: OBJECTIVES AND METHODS

FINAL REPORT FOR TASK ORDER 2

Alexandre M. Bayen, Ph.D., Principal Investigator
Mohammad (Ashkan) Sharafsaleh, P.E., Project Manager
Anthony D. Patire, Ph.D., Assistant Research Engineer

PATH Research Report
UCB-ITS-PRR-2013-2

Partners for Advanced Transportation Technology works with researchers, practitioners, and industry to implement transportation research and innovation, including products and services that improve the efficiency, safety, and security of the transportation system.

Partners for Advanced Transportation Technology
University of California, Berkeley
2105 Bancroft Way, Suite 300, Berkeley, CA 94720-3830
Phone: (510) 642-4522   Fax: (510) 642-0910   http://www.path.berkeley.edu

ISSN 1557-2269 OCTOBER 2013

HYBRID TRAFFIC DATA COLLECTION ROADMAP: OBJECTIVES AND METHODS

FINAL REPORT FOR TASK ORDER 2

Prepared by:

Alexandre Bayen
Mohammad (Ashkan) Sharafsaleh
Anthony Patire
Joe Butler
Pieter Abbeel
Sébastien Blandin
Tim Hunter
Tasos Kouvelas
J.D. Margulici
Pierre-Emmanuel Mazare
Ali Mortazavi
Ricardo Reynoso
Samitha Samaranayake
Krute Singa
Olli-Pekka Tossavainen
Fred Winik
Dan Work

For:

CALIFORNIA DEPARTMENT OF TRANSPORTATION
DIVISION OF TRAFFIC OPERATIONS

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the State of California. This report does not constitute a standard, specification, or regulation.

ACKNOWLEDGMENTS

We would like to acknowledge the participation and support of the many people who, in their respective organizations, have contributed to the Hybrid Traffic Data Collection Roadmap project (Task Orders 1 and 2).

California Department of Transportation

This project would not have been possible without the interest, engagement, and financial support of our sponsors at Caltrans. Joan Sollenberger (Chief, Office of System Management Planning) and Nick Compin (Chief, Traffic Data and Performance Measures Branch) provided ongoing consultation and valuable input in an unfailing spirit of collaboration throughout the project. We are grateful for their confidence and involvement in the work.

We also received important assistance from the departments of Operations and Maintenance of Caltrans District 4 and District 8, including Sean Nozzari, Renato Dacquel, Renato Fortaleza, Thomas Ainsworth, Peter Acosta, and their respective groups. We thank them for their cooperation and support.

In addition, the Division of Research and Innovation and the Division of Operations, both at Caltrans Headquarters, and the Operations offices in District 4 and District 7 generously granted us interviews to explore the potential impact of purchasing probe data. We appreciate their time and insights.

Additional contributors

A variety of participants made significant contributions to the project, and we gratefully acknowledge their work. Thanks go to Nokia team members Jane Macfarlane, Andy Lewis, and Jeff Adachi; Iteris team members Karl Petty, Mark Merala, Andrew Moylan, Ricardo Reynoso, and Nick Hartman; and J.D. Margulici of Novavia Solutions.

We also received helpful input from the City of San Jose Transportation Department, the Contra Costa County Transportation Authority, the Metropolitan Transportation Commission, Berkeley Transportation Systems (a division of Iteris Inc.), and Delcan Corporation. Discussions with them expanded our understanding of current practices in the transportation community.

The PATH team

California PATH members at all levels and in all capacities provided the energy, dedication, and teamwork to develop and complete the project. We were fortunate to collaborate with such a talented group.

Management

We want to thank PATH Director Tom West and Program Manager Joe Butler, whose foresight and skillful leadership made the project possible and propelled it forward. The work could not have proceeded without their efforts.

We would also like to express our appreciation to Samer Madanat, who served as original Principal Investigator for the project, and Ali Mortazavi for his contribution as original Project Manager.

Team members (staff, students, alumni, visitors, in alphabetical order)

Saneesh Apte (R&D Engineer), Nate Bailey (Undergraduate Research Apprentice), Agathe Benoit (Visiting Graduate Student Researcher), Sebastien Blandin (Graduate Student Researcher), Joy Bonaguro (Graduate Student Researcher), Paul Borokhov (Graduate Student Researcher), Stephen Carrier (Systems Administrator), Elie Daou (Visiting Graduate Student Researcher), Gunes Dervisoglu (Graduate Student Researcher), Magnus Fransson (Visiting Graduate Student Researcher), Timothy Hunter (Graduate Student Researcher), Raphaël Jatteau (Graduate Student Researcher), Matt Juhn (Applications Developer), Tasos Kouvelas (Postdoctoral Researcher), Tom Kuhn (R&D Engineer), Alex Kurzhanskiy (Research Engineer), Tristan Lall (Graduate Student Researcher), Pierre-Emmanuel Mazare (Visiting Graduate Student Researcher), Greg Merritt (Systems Analyst), Scott Myers (Applications Developer), Boris Prodhomme (R&D Engineer), Joao Rodrigues (Graduate Student Researcher), Samitha Samaranayake (Graduate Student Researcher), Mats Sandin (Visiting Graduate Student Researcher), Bill Sappington (Database Administrator), Krute Singa (Research Data Analyst), Nazy Sobhi (Business Analyst), Jerome Thai (Graduate Student Researcher), Luis Torres (Systems Administrator), Olli-Pekka Tossavainen (Postdoctoral Researcher), Joel VanderWerf (Systems Engineer), Fred Winik (Technical Writer), Dan Work (Graduate Student Researcher), Matt Wright (Graduate Student Researcher), Jeffrey Young (Communications Analyst), Yi Zhou (Undergraduate Research Apprentice).

Many people provided crucial subject matter during the project through conversations and interviews, original project documents, briefing materials, personal notes, emails, and other sources that made it possible to construct the narrative in this report. If, through our own oversight, we neglected to mention anyone who contributed to the project, we ask them to accept both our apologies and our gratitude.

EXECUTIVE SUMMARY

MOTIVATION

Traditionally, Caltrans' automated traffic data collection systems have relied heavily on roadway-embedded sensors, such as loop detectors installed at fixed locations. With the recent prevalence of commercial traffic data sources, however, and the rising cost of maintaining state-sponsored traffic data collection operations, Caltrans is exploring the possibility of purchasing probe data from the commercial sector. To help with this effort, PATH undertook the *Hybrid Traffic Data Collection Roadmap: Objectives and Methods* (Task Order 2) that investigated the processes and algorithms required to assimilate probe data (unaggregated GPS point speeds) and fuse it with Caltrans' existing data for the purpose of estimating travel times. The task order also examined the business case for purchasing and integrating probe data. In conjunction with the related Task Order 1 (*Pilot Procurement of Third-Party Traffic Data*), the intent was to demonstrate an efficient and cost-effective use of alternative traffic data sources to complement the detection systems currently installed and operated by Caltrans.

CONTEXT OF THIS STUDY

Traffic data is used to estimate current traffic conditions so that travelers and agencies can make better decisions about how to use and manage the transportation network. This contract explored the fusion of probe data (vehicle speed and direction) with loop data (density, speed, and count) in the context of producing overall network speed and travel time estimates.

Speed and travel time estimates are useful in many circumstances, but current system control strategies (ramp metering, for example) require density data. The next phase of research will likely focus on new control strategy implementations that use fused probe and loop data for traffic management.

Our analysis of the ability to reduce the density of loops as presented in this report should therefore be viewed in the context of our current research, which focused on using fused probe data to estimate speed and travel time. The results and recommendations may differ when the requirement to control traffic is taken into account.

Additionally, future data procurement decisions need to determine whether the fusion and estimation steps would be done by an outside vendor or by transportation agencies.

OBJECTIVES

The objectives of this project were to:

- Establish definitions, criteria, and guidelines for assessing data quality
- Identify the algorithms and processes needed to use probe data and fuse it with loop data
- Integrate the data in a flow model for the purpose of estimating travel times
- Test the implementation using varying proportions of probe and loop data
- Analyze the results
- Evaluate the business case for purchasing and fusing probe data from the commercial sector

RESEARCH FOCUS

Research efforts focused on the following areas:

Data quality assessment

We investigated the issue of assessing data quality in an era of ubiquitous probe data, outlining the definitions and criteria needed to measure data quality (accuracy, completeness, validity, etc.), a methodology for assessing it, the characteristics needed for a reference state to represent ground truth, and a multi-level validation methodology. A Data Quality Tool was also developed for examining the characteristics of probe data feeds directly.

Probe data quality

To examine the implications of using varying amounts of probe data, we studied data collected during a one-day controlled experiment from 100 GPS-equipped cars used as probe vehicles driving for multiple hours on a section of Bay Area roadway. Using algorithms, we modified the penetration rates and VTL (virtual trip line) locations to study the effects of the changes on the ability to estimate accurate speeds for a roadway. This study was the precursor to, and informed the design and methodology of, the further analysis of data fusion performed under Task Order 1.

Data fusion implementation

We studied how to fuse multiple data sources with various characteristics by running probe and loop data through the Mobile Millennium highway model, which generated velocity maps and travel times. The Mobile Millennium system can accept data from traditional sources (such as occupancy and counts from loop detectors) and point-speed measurements from providers of probe data. This enabled us to evaluate the performance of the data sources both individually and when fused together.

Sensitivity analysis of loop detector spacing and location

To analyze the sensitivity of the spacing and location of fixed sensors along the roadway, we algorithmically simulated the removal of loop detectors to test whether loops could be placed further apart and still provide sufficient information to generate accurate traffic estimates.

Balancing loop and probe data

By adjusting the amount of data from loops and probes in combination, we were able to create multiple scenarios with different proportions of loop and probe data and evaluate the impact on computing travel times. A total of 1,637 scenarios were developed by instantiating all combinations of 9 sets of inductive loop detector data sets (from 0 to 16 detectors), 11 probe penetration rates, and various space-based and time-based sampling strategies.

Hybrid data roadmap

A hybrid traffic data roadmap document was developed that provides an overarching view of the context, objectives, and implementation of a hybrid traffic data system. Drawing on the full scope of work completed for Task Orders 1 and 2, the roadmap includes a business analysis assessing the benefits, trade-offs, and next steps in procuring third-party data and integrating it into Caltrans' existing structure.

CONCLUSIONS

The study led us to the following conclusions:

- **Data quality**—The quality of probe data can be measured and compared to ground truth. Combining the measurements source with a traffic model within an estimation framework can provide levels of accuracy associated with the robustness of the model and the design and purpose of the estimation method (such as estimating travel times).

- **Speed data and travel times**—GPS point-speed data is usable for the intended application (estimating speed data and travel times) and can be successfully processed with the Mobile Millennium system to map velocities.

- **Data filtering and map-matching**—Point speeds from GPS-equipped probe vehicles can be filtered to remove faulty data, and the vehicles and their trajectories accurately mapped to the road network. This is essential for developing reliable estimates of velocities and travel times along the highway.

- **Data fusion**—Probe data can be successfully fused with loop detector data, and meaningful comparisons can be assessed.

- **Loop detector spacing**—In this study, spacing loop detectors less than an average of 0.83 miles apart (i.e., using data from more than eight inductive loop detector stations along the stretch of roadway under study) did not provide extra benefit in the travel time estimation. The error remains constant between 6–13% depending on the time of day, regardless of the added loop detector stations.

- **Quantity of probe data**—In this study, when sampling probe vehicles at a rate of 137.5 veh/hr with more than 2.54 VTL/mi, increasing the number of probe measurements by adding more probe vehicles or additional virtual trip lines caused only small improvement in the travel time accuracy.

- **Mixing probe and loop detector data**—It was found that when complementing loop detector data with probe vehicle data, better estimates for travel times are obtained, especially at low penetration rates. In this study, for example, if using loop detectors spaced more than 2.11 miles apart, probe data can give over 50% increase in the travel time accuracy. These results hold generally, independent of the sampling strategy of the probe vehicles.

- **Confidence in the model**—In this study, it was found that when using a flow model with data assimilation, dynamic travel times can be estimated with less than 10% error by using either inductive loop detector data, probe data, or a mixture of both. The quantitative and graphical results of the research give us confidence in both the modeling approach to roadway estimation and the effectiveness of the Mobile Millennium highway model.

- **Costs, benefits, trade-offs**—When examining the trade-offs between the costs and performance of available data sources, we found that the probe data purchased for the hybrid data roadmap project could often match existing loop detectors in producing consistent travel time estimates and can do so at lower expense, especially if detector maintenance and replacement costs are factored in. The highest quality estimates, however, were achieved by combining both probe data and loop detector data. Moreover, this project focused on only one application of traffic information—travel time estimation—and did not investigate other traffic management applications such as control strategies.

IMPLICATIONS AND FUTURE DIRECTIONS

The successful procurement and processing of probe data for travel time estimation, as demonstrated in Task Orders 1 and 2, illuminates the complexities, challenges, and opportunities of the new world in which transportation agencies find themselves. Some of the implications to emerge from the project include:

Reduced dependency on loop data

While current system control strategies, such as ramp metering, require density data, it seems difficult to significantly increase the quantity of loop detectors on California roads. At the same time, the penetration rate of probe data is continually increasing and far from reaching its limits. This represents a sea change in the types of data available for traffic management and offers the prospect of migrating away from exclusive dependency on loop detectors over time.

Outsourced data collection

Purchasing probe data from the commercial sector means, in effect, outsourcing the collection of traffic data. Any such undertaking comes with new risks (e.g., data quality, privacy protection, business continuity) which would need to be managed through, for example, a careful vetting and data acquisition process and robust data assessment tools, processes, and standards.

Redesigned information systems

The research work in this project was predicated on having a data assimilation and state estimation system in place that would allow the implementation, testing, and analysis of data hybridization. This required:

- Building and calibrating a model to estimate speed and travel times from probe data
- Developing a set of methods to fuse probe and loop detector data
- Creating tools to visualize the data
- Testing the tools and methods on real data from pilot sites
- Building tools to determine the quality of the methods, models, and data

Creating this mathematical and technology infrastructure points the way to the redesign of information systems that would make it possible to implement data fusion and take full advantage of hybrid data in traffic management systems.

New detector strategy

While further research on the position and spacing of loop detectors is needed, our initial results suggest that using the most critical detectors (those that add the most information value) rather than enforcing a minimal spacing between detectors could help optimize the existing stock of detectors and allow Caltrans to selectively focus maintenance efforts or supplement loop data with probe data when certain loops fail.

Augmented traffic measurements

This project studied the use of probe data for estimating travel times. However, the enhanced modeling and estimation accuracy demonstrated by data fusion also lays the foundation for better control strategies and operational decision-making. Augmenting traffic volume measurement with probe data, for example, could be a fruitful area for research in the future. Fused loop and probe data (“hybrid” data) could thus provide a pathway to the development and use of additional traffic measurements, such as arterial estimation, origin-destination information, demand modeling, and, ultimately, integrated corridor management.

The road ahead

The Hybrid Data Roadmap (Chapter 6) outlines a possible implementation path that would enable Caltrans to gradually experiment with and adopt a hybrid traffic data collection system. It involves three phases that could be developed over a time horizon of three to five years:

1. Short-term pilot in a selected district (1–2 years)
2. Using the PATH Connected Corridors project to spearhead information systems innovations (2–3 years)
3. Full-scale pilot in a selected district (3–5 years)

These phases would allow Caltrans to leverage what was learned from Task Orders 1 and 2, build on the mathematical and technology infrastructure already in place at PATH, and test the viability of hybrid traffic data collection in a controlled manner. They would thus provide a way to effectively manage the next steps into a new traffic information environment.

Table of Contents

Chapter 1

Introduction

1 INTRODUCTION

This final report documents Task Order 2, Hybrid Traffic Data Collection Roadmap: Objectives and Methods. The task order falls under the “Hybrid Traffic Data Collection Roadmap” technical agreement number 51A0391, executed in September, 2009 under the parent agreement 22A0486 between the California Department of Transportation (Caltrans) and the Regents of the University of California (UC Regents).

This task order explored the processes, algorithms, and business case for using commercially available probe data (vehicle speed and direction) with loop data (density, speed, and count) in the context of producing overall network speed and travel time estimates. The project was carried out by California PATH (Partners for Advanced Transportation Technology), a research unit of the Institute of Transportation Studies at the University of California, Berkeley.

1.1 MOTIVATION

Data is the lifeblood of effective transportation management. Traffic data is used to estimate current traffic conditions so that travelers and agencies can make better decisions about how to use and manage the transportation network.

Caltrans has traditionally captured traffic data from sensors buried in the road, such as loop detectors installed at fixed locations. While these detectors yield solid results for estimates of traffic volume and occupancy, they do not provide accurate travel time information unless the sensor coverage is very dense. In addition, roadway-embedded sensors currently in use are beginning to age, and the costs of maintaining or replacing them are high. Consequently, to avoid the capital and maintenance costs of new sensor technologies, Caltrans is looking into procuring good-quality data from third-party sources and integrating that data into Caltrans’ existing systems.

As part of that effort, PATH undertook the Hybrid Traffic Data Collection Roadmap: Objectives and Methods (Task Order 2) that investigated the processes and algorithms required to assimilate third-party probe data (unaggregated GPS point speeds) and fuse it with Caltrans’ existing data for the purpose of estimating travel times. The task order also examined the business case for purchasing and integrating probe data. In conjunction with the related Task Order 1 (Pilot Procurement of Third-Party Traffic Data), the intent was to demonstrate an efficient and cost-effective use of alternative traffic data sources to complement the detection systems currently installed and operated by Caltrans.

1.2 NEW SOURCES OF DATA

From Roman roads to GPS-equipped probe vehicles, transportation engineers have built on the achievements of their predecessors. In 2008, the Mobile Century experiment demonstrated the feasibility of monitoring traffic by using data from GPS-enabled cell phones in a controlled environment.

Beginning later that year, the Mobile Millennium project extended that capability to a complex urban environment on a far larger scale, gathering traffic information from the GPS in cell phones, processing it through a flow model, and distributing it back to the phones in real time to give users up-to-the-minute travel information. Since those groundbreaking achievements, the dramatic proliferation of smartphones and on-board GPS units carried by drivers has transformed ordinary cars, trucks, buses, and taxis into a mass of mobile traffic probes, a sea of data collection devices transmitting each vehicle's location, direction, and speed.

The massive availability of traffic data from GPS-equipped probe vehicles will have important consequences for both the traveling public and roadway operators, and Caltrans, in particular, is set to benefit tremendously from its use. Leveraging this data source could drastically cut the ongoing costs of traffic monitoring and expand coverage to thousands of miles of highways and urban arterials for which sensor installations are not considered an option. Moreover, it could be used for improving system management, traffic flow, transportation safety, work zone safety, emergency services, and evacuation management, as well as increasing the efficient movement of goods and people across California. Further, the dissemination of traffic information could promote a form of system "self-management" in which individual commuters can make informed travel decisions. Not only would each user benefit personally, but the entire driving community would enjoy more balanced loads across the road network.

1.3 A HYBRID TRAFFIC DATA SYSTEM

The data collected from cell phones and other GPS devices, however, is limited to velocity information. Caltrans has relied on fixed loop detectors to provide measures of traffic flow and occupancy rate, and current system control strategies (ramp metering, for example) in fact require density data. It is therefore unlikely that probe data will completely replace existing detection systems in the foreseeable future.

However, using mobile probe data to complement existing detectors—a so-called "hybrid" traffic data collection system—has real viability. In such a setup offering ubiquitous availability of speed information, loop detector stations would be needed only to maintain accurate flow counts. Larger spacing intervals could be allowed between stations, and therefore equipment could be deployed much more sparingly than it is today. Caltrans could thus be getting much more information while spending much less for it.

(It should be noted that this project focused on using fused probe and loop detector data to estimate *speed and travel time*, rather than to explore control strategy implementations. Using fused data for traffic control strategies will likely be addressed in future research.)

1.4 PROBLEM STATEMENT

Purchasing traffic data represents a fundamental shift for Caltrans that poses its own set of challenges. As this task order focused both on the methods of evaluating and assimilating third-party data and on examining the value for Caltrans of doing so, the project presented a number of key questions:

- With the growing availability of probe data, can its quality be assessed?
- What guidelines might be used?
- Can it be compared to a benchmark?
- Can it be used and fused with loop detector data?
- How can this be achieved?
- With what results?
- To what extent can probe data supplement or supplant loop data?
- What is the business case for purchasing commercial data and integrating it into Caltrans' existing systems?

1.5 OBJECTIVES

To address these questions, the research team identified the following objectives for this project:

- Establish definitions, criteria, and guidelines for assessing data quality
- Identify the algorithms and processes needed to use purchased data and fuse it with loop data
- Integrate the data in a flow model for the purpose of estimating travel times
- Test the implementation using varying proportions of probe and loop data
- Analyze the results
- Evaluate the business case for purchasing and fusing probe data from the commercial sector

1.6 CONTRACTUAL DELIVERABLES AND ORGANIZATION OF THIS REPORT

Task Order 2 categorizes the proposed work into five work packages:

- Work Package 1. Data quality metrics and measurement procedures
- Work Package 2. Probe data quality study
- Work Package 3. Data fusion implementation and prodecures
- Work Package 4. Sensitivity analysis of loop detector spacing and location
- Work Package 5. Hybrid data roadmap

This report is organized into six chapters and addresses the Task Order work breakdown as follows:

- Chapter 2, *Assessing Data Quality*, addresses the issue of data quality in an era of ubiquitous probe data and speaks to Work Package 1. It outlines the definitions and criteria needed to measure data quality, a methodology for assessing it, the characteristics needed for a reference state to represent ground truth, and a multi-level validation methodology.

- Chapter 3, *Data Quality Tool User Guide*, describes the Data Quality Tool and how to use it. The tool was created to examine the characteristics of probe data feeds directly by displaying their attributes for a variety of metrics.

- Chapter 4, *Path Inference Filter: Map Matching of Probe Vehicle Data*, describes a class of algorithms that accurately map probe vehicles to the road network and plot their trajectories. This vital function makes it possible to use probe data from multiple sources (taxis, buses, truck fleets, and so on) to develop reliable estimates of velocities and travel times along the highway.

- Chapter 5, *Loop and Probe Data: Assimilation and Trade-offs,* presents a study of data assimilation, and details the work done for Work Packages 2, 3, and 4: integrating loop and probe data into a flow model to estimate travel times; fusing data from the different sources; comparing the quality of the output to ground truth; varying the amounts of probe and loop data by adjusting the number and spacing of loop detectors and the penetration rate and sampling strategies of probes; and analyzing the results.

- Chapter 6, *Hybrid Data Roadmap* (Work Package 5), offers an overarching view of the context, objectives, and implementation of a hybrid traffic data system, including a business case that assesses the benefits, trade-offs, and next steps in procuring third-party data and integrating it into Caltrans' existing structure.

- The report concludes with Chapter 7, which reviews the project, summarizes the results and the conclusions that emerged from the work, and considers the implications for the future.

Chapter 2

Assessing Data Quality

Fundamental to purchasing probe data in the commercial sector is the issue of data quality. How good is the data? Good for what purpose? Good compared to what? While the Task Order 1 report (*Pilot Procurement of Third-Party Traffic Data*) details the specifications for purchased data and the metrics for quantifying its usefulness, and Chapter 3 of this report describes the Data Quality Tool for examining it, this chapter addresses the essential definitions, principles, and processes involved in assessing data quality.

1 INTRODUCTION

In one of the seminal chapters of the transportation community dating back to 1935, Greenshields [1] states that there is a linear relationship between speed and density on freeways. This empirical study, which led to the classical model used by a large part of the transportation community today, relied on state-of-the-art data collection techniques at the time. A camera located about 350 feet from a roadway was taking pictures at a rate of 88 frames per minute, and the pictures were *superimposed upon a scale to show the distance traveled* [1].

In the modern age of information technology, during a typical day major traffic information providers collect easily 30 million probe data measurements in the United States alone, which come in addition to millions of data points collected by dedicated sensing infrastructure (loops, cameras, etc.). This significant increase in the volume of the available traffic data and the advances in the collection technology are the most visible part of a paradigm shift which requires the transportation community to rethink the definition of traffic data and to lay down new foundations for data quality assessment.

In a recent reflection from the computer sciences community on the work of Gray [2], it is argued that in the future, the ability of computational engineering researchers to make use of massive amounts of data will play a decisive role in both innovative research and operational efficiency. This seems particularly true in a field such as traffic engineering, which is by nature intricately related to people’s lives and therefore can benefit from synergies with other data-driven fields. This change would not have been possible without major improvements in communication technology in the last two decades. For example, in California the PeMS [3] infrastructure was one of the first to allow efficient and extended access to traffic measurements with analysis capabilities. More recently, the Mobile Millennium traffic estimation system [4], relying mostly on user-generated traffic data, pioneered traffic data collection by the mass public (also called crowdsourcing), with the incentive of improving traffic estimates [5] [6] [7] [8] [9]. It is now clear that user-generated data can be of strategic interest for a wide variety of applications. For instance, commuters’ societal profile and typical social activities allow one to forecast their future locations and thus congestion level more accurately [10]. User-generated data can also provide information about the type of cars being driven and help calibrate pollution maps.

The increasing complexity of traffic data motivates a redefinition of traffic data quality and the use of more complex analysis tools. Because the term *traffic data* now encompasses a broader variety of raw and processed measurements, a more elaborate framework has to be designed to integrate this diversity. Working with massive amounts of data requires well-thought-out standards for categorization and quality assessment of data feeds and appropriate tools to leverage this wealth of information. Recent work proposed *artificial societies for modeling, computational experiments for analysis, and parallel execution for control* [11]. This approach would greatly benefit from a massive volume of traffic data but requires substantial processing power.

From the data quality assessment perspective, following [12], this study proposes to lay the foundation of traffic data understanding at the application level, since the application is the motivation for any engineering process and is assumed to be fully understood and accurately defined. *Data quality is the* *fitness of data for all purposes that require it. Measuring data quality requires an understanding of all intended purposes for that data* [12].

An application-dependent quality assessment process is a significant change from previous approaches to data quality assessment, which essentially amounted to data collection science. The problem of data collection methodologies for specific data types, mostly travel time, has been extensively studied in the literature [13] [14] [15]. The focus of these studies has in general been on operational specifications for data collection procedures. Numerous efficient data processing techniques have been designed in that regard, for instance [16], but the larger picture has been partially ignored in the process. In this chapter, the extensive studies conducted on data collection are leveraged to propose a new paradigm for data quality assessment.

2 DATA QUALITY DEFINITIONS AND CRITERIA

In order to support a clear and coherent discussion of traffic data quality in this study, a set of definitions is introduced below. These terms are used to qualify traffic data at various steps, from its collection to its final usage. These definitions are not necessarily universal, but reflect their context in traffic engineering. While some might appear intuitive, they are necessary to accurately capture some of the concepts which our analysis tools attempt to encompass.

2.1 GLOSSARY

The following terms allow us to accurately define data quality requirements.

- *Data feed*: Collection of raw or processed data points.

  Example: In California, the PeMS loop detectors feed [3]and Mobile Millennium segment speed feed [17], data points from Greenshields article [1].

- *Data source*: Sensing device using a given physical phenomenon to measure a well identified quantity. The word sensor is often used to denote a data source, and in general the physical phenomenon is assumed to be uniquely defined once the sensing device is specified.

  Example: Loop detectors using magnetic induction, cell phones using GPS trilateration, speed radar using Doppler shift.

- *Data type*: Traffic quantity and its level of aggregation in space and time.

  Example: Point counts over 30 seconds, instantaneous point speed, 1-mile segment speed over 5 minutes.

- *Ground truth*: True value of a data type. It is known only by an oracle.

- *Processed data*: Output from the processing of a data feed.

  Example: Five-minute PeMS aggregated counts from loop detectors, segment speeds from GPS point speeds.

- *Processing*: Irreversible computational mechanism aimed at transforming a data feed.

  Example: Averaging, outlier removal, low-pass filtering.

- *Raw data*: Direct output stream from a data source.

  Example: [Time, device identity, position] from a GPS device, [time, device identity, voltage] from a classical loop detector.

- *Reference state*: Best available estimate of ground truth. This is a choice made by the practitioner.

Example: Camera record for point locations, Mobile Millennium highway model output for segment speed.

- *Traffic application*: Activity of interest to traffic managers, agencies, or institutional entities.

  Example: Point speed estimation, ramp metering, traffic information dissemination.

- *Traffic state*: Quantity chosen to characterize traffic conditions.

  Example: Count, occupancy, density, speed, vehicle-miles traveled.

2.2 DATA QUALITY CRITERIA

The main criteria according to which data quality can be assessed have been the subject of extensive studies in the literature [12]. In this chapter the criteria introduced in a 2004 report to the U.S. Department of Transportation [18] are used as a basis for a slightly modified definition:

- *Accuracy*: Degree to which the data type from a data feed matches the ground truth.
- *Accessibility*: Ease of manipulation of the individual elements of a data feed when used by a given traffic application.
- *Completeness*: Degree to which values are present for all the fields of the data feed.
- *Coverage*: Degree to which the data feed encompasses the full extent of what is of interest.
- *Purity*: Degree to which the data feed consists of raw data.
- *Validity*: Degree to which the data feed fields satisfy acceptance requirements.

Two major modifications of the definitions from [18] have been introduced:

- *Accounting for timeliness as part of the validity requirement*: The acceptable delay with which a data feed can be provided is an acceptance criterion and, as such, is accounted for in the validity specification. For different applications, different delays can be tolerated and yield different acceptance requirements in the data feed specifications (real-time applications require a relatively short delay, a posteriori performance analysis can accommodate a much larger delay).
- *Introducing the purity requirement*: One of the major foreseeable changes for traffic data is the spread of fused data from multiple data feeds, where each feed can be raw or processed. This trend creates multiple asymmetric layers between ground truth and the final data feed. For quality assessment purposes this study proposes to make the distinction between raw data and processed data. This is related to the distinction between *original source data* or *archive data* and *traveler information data* in [18]. In this study the motivation stems from the transparency brought by raw data. The quality of a raw data feed can be assessed from fundamental descriptive characteristics of both the measuring data source (physical process and device

specifications), the measuring context (number of data sources, area of deployment, etc.), and some technical specifications (maximal delay, sampling rate, etc.). On the other hand, defining a robust validation methodology for processed data is a more involved problem:

- *Sampling is crucial in the validation of processed data*: A processed data feed does not easily allow one to trace the origin of the value of the data. Because of the periodicity of traffic patterns, similar accuracy performances can be achieved with a good quality raw data feed used by a simple algorithm, and a purely historical data feed with sophisticated algorithms. In fact, in the market of data, historical and real-time are often mixed and sometimes interchanged in their use. Assessing the quality of a processed data feed requires appropriate statistical sampling to identify how well the feed behaves in the case of unforeseeable events (road closure, accident, etc.).

- *Responsibility with respect to society requires transparency*: Assessing the quality of a data feed solely from its final output does not seem to provide the guarantee required by large scale strategies impacting the life of millions of people. Without additional knowledge on the processing used, providing a guarantee level equivalent to the one given by the extensive knowledge of the data sources associated with a raw feed is a difficult problem.

- *Anticipating changes in traffic data world*: The current growth in deployment of sensing devices suggests that the future of traffic data is a combination of more and more layers of processed data, with more and more intricate data feeds and data types (e.g., mobile devices currently providing speed only may soon provide pollution indicators based on the use of additional embedded sensors, and a smart processing of traffic speed, weather, and location).

The following section expands on the key distinction between raw data and processed data.

2.3 DISTINCTION BETWEEN RAW DATA AND PROCESSED DATA

A raw data type consists of the output of a data source, without any modifications. In that regard, the data quality assessment of this data feed is guided by the corresponding data source, and more precisely by:

- the error of the physical mechanism leveraged by the data source for sensing (e.g., GPS multi-path)

- the technical specifications of the data source considered (e.g., GPS range)

- the network properties of the fleet used to collect the data (e.g., coverage)

Processed data, on the other hand, can be generated by a non-transparent method, which can lead to significant loss of correctness in data quality assessment. *GPS measurement* is a common term, widely used within the transportation community. However, for data quality assessment purposes, it is an extremely loose term. Several interpretations can be made, leading to different qualitative quality statements:

- Point location measured by a GPS. The data type is raw. There is a small and unavoidable error term introduced by internal GPS processing. The error term is expected to be high only during urban canyoning, or when occlusion occurs.
- Point location estimated from a given GPS point location, car odometer measurements, and magnetometer readings. The data type is processed from different signals. The computation or process leading to the computed position is not well identified and documented and thus its typical error is difficult to characterize, although it is assumed to be lower than in the case above.
- Point speed computed from consecutive GPS point locations using an upwind or downwind numerical scheme. The data type is processed. The error increases at every abrupt speed change of the probe vehicle.
- Point speed measured by a GPS device using Doppler shift. The data type is raw. The error term is mostly impacted by multipath and atmospheric turbulence. There should be no correlation between speed changes and error variations.

In each of these cases, the error profile has different magnitude and is influenced by different factors. The root cause of any inaccuracy is unknown to the user unless more information than just the appellation *GPS measurement* is provided. The error profile is also unknown. With the most basic models, the error profile for raw data is in general Gaussian, but for processed data it relies heavily on the processing, its nonlinearity type, and the data sources leveraged.

Given the various nature of processed data and its multiple degrees of complexity, this study focuses on raw data types, which allows a more rigorous and generalizable discussion.

The next section presents a data quality assessment framework along with a methodology for traffic data validation, which is later instantiated on two data feeds available through the Mobile Millennium system [4].

3 DATA QUALITY ASSESSMENT METHODOLOGY

In this study data quality assessment is considered as a process relating physical traffic phenomena to a given traffic application (see Figure 2-1). This section describes a data quality assessment methodology centered on this principle. A two-step procedure for validation is also defined, along with requirements for a definition of the appropriate reference state.

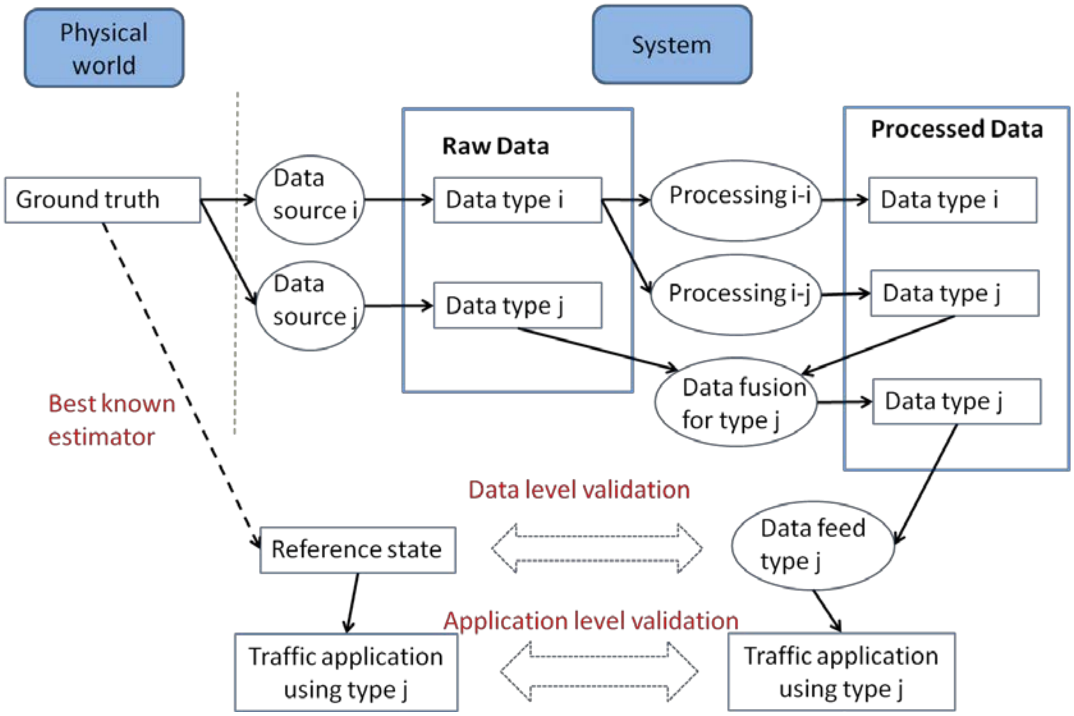


Figure 2-1: Data quality framework: Ground truth is measured by data sources. Data sources output raw data, which can be processed in a subsequent step. Different data feeds can be fused (this figure only presents data fusion across identical data types). Data feed validation is completed at both the data level and the application level.

3.1 TRAFFIC APPLICATIONS

Since data is typically requested with a specific application in mind, it seems natural to build the data quality assessment and validation process from the application side. For instance, a data feed exhibiting a large delay would be useless for any real-time application, but this does not prevent a user from using it for a posteriori performance assessment. Similarly, a data feed reporting the most common car model on a given spatiotemporal discretization would be useless for speed estimation, but very useful for the design of a pollution map.

Main traffic applications can in general be divided into three different categories:

- *Traffic operation and control*: Activities whose goal is to directly modify traffic conditions in order to improve performance indicators
- *Performance management*: Analysis of traffic phenomena and infrastructures in order to improve operational procedures and commuters' experience
- *Traveler information*: Providing commuters with knowledge of the traffic conditions (past, current, or future) to improve their transit quality

Given an application, the definition of the reference state used for validation of the data feed of interest is of crucial importance.

3.2 REFERENCE STATE

Current challenges facing the transportation community, in light of the increasing amount of available data sources, inevitably lead to the necessity of a change in the classical approach to data quality assessment. The classical approach, which amounts to assessing the quality of a data feed by comparing it with measurements from another sensor, should be carefully considered. Such a process simply allows one to provide insights and not exact extensive knowledge, on the accessibility, completeness, and coverage of the data feed. Additionally, the accuracy of the data feed cannot be reliably assessed from such a method. In a sound estimation framework, the accuracy of a traffic state estimate should be compared with the best ground truth estimate available. The methodology used to estimate ground truth should provide the following features:

- *Confidence*: Scientific soundness provided by the use of well known, well studied, and acknowledged techniques supported by significant academic work.
- *Robustness*: The uncertainty in the data quality assessment process should be known and documented, and not rely heavily on the transient characteristics of the data feed (variability patterns, coverage, etc.).
- *Scalability*: Efficiency of the process when the volume of the data feeds, validation accuracy, or size of the coverage area increases.
- *Cost-efficiency*: Limited deployment procedures.
- *Applicability*: The data type should be appropriate for the application of interest.

While these features appear desirable, they are at best only partially present in current traffic data quality assessment procedures. Work from the estimation community could be leveraged to compare a data feed with a well-known, well-calibrated, scientifically supported, and robust estimate, which would be developed by specialists in the community [4] [19] [20] and would improve the comparison relevance and decrease the comparison cost.

The advantage of this approach can be illustrated by a simple comparison between weather and traffic information systems. Meteorologists want to estimate and potentially forecast the state of their system (atmosphere or highway) in real-time. In order to obtain information about the system, a certain number of probes and static sensors can be deployed. However, meteorologists heavily rely on the use of models such as Navier-Stokes equations combined with data assimilation algorithms. This is required by the dimensionality and complexity of the underlying physics. Similar complexity is true for the dynamics of the highway, which highlights the need for model-driven approaches to define the reference state. At least, if one cannot produce more accurate estimates with an a priori approach (compared to dedicated sensor deployments), one can reduce some costs.

4 VALIDATION

This section describes the validation methodology that is used to assess the quality of a data feed.

4.1 METHODOLOGY

In order to assess the quality of a data feed, a two-level validation process is proposed:

- *Data-level validation*: This step consists of assessing how the data feed satisfies the accuracy, accessibility, completeness, coverage, purity, and validity requirements. It consists of a sequence of simple verifications which should be completed both at the data point level (measurement accuracy) and the data feed level (delay). This validation process requires a priori definition of the reference state, according to the criteria defined in previous section.

- *Application-level validation*: This step consists of assessing how the application of interest performs using the considered data feed. This can be determined by comparing how the application performs with the reference state feed and with the data feed considered. Sensitivity analysis of the application output to the data feed is also of great value for validation purposes.

The following section presents the distinct steps proposed for the data-level validation process.

4.2 DATA-LEVEL VALIDATION

Data-level validation evaluates the conformity of the data feed to the requested specifications. The following steps are proposed:

- *Integrity*: The data feed should satisfy the requested specifications for every data point and for the feed as a whole. This enforces accessibility, completeness, coverage, and validity.

- *Consistency*:

  - Checking whether the data is consistent across data points. For example, a sequence of probe measurements should exhibit feasible changes in speed over given space and time intervals.

  - Comparing data across multiple data feeds. This allows the identification of the feeds which significantly differ from the others and thus should be considered more carefully. Further validation is required to determine whether one of the feeds or the set of feeds is providing inaccurate data points.

- *Accuracy*: Comparing the data feed of interest with a data feed that estimates ground truth with a known higher level of accuracy than the data feed to be validated in a statistically sampled subset of the spatiotemporal domain. The state given by this feed is called the *reference state*. Section 4.4 describes how the reference state is determined.

In the following section application-level validation is described.

4.3 APPLICATION-LEVEL VALIDATION

Application-level validation evaluates the appropriateness and value of a data feed for a given application. The following steps are proposed:

- *Compatibility*: Whether the feed data type is appropriate for the application. For instance, generating segment speed maps ideally requires a segment speed data feed.
- *Cross-validation*: Comparing different data types to determine which type is most appropriate for the application. For segment speed maps, the use of a segment speed data feed could be compared to the use of a point speed data feed processed to generate segment speeds. In the latter case the nature of traffic flow would not be accurately captured and thus the point speed feed is assumed to not perform as well.
- *Performance*: Comparing the performance of the application with the data feed to be validated with the performance of the application with the reference state feed. The performance should exceed a predetermined threshold.

4.4 DETERMINING THE REFERENCE STATE

The reference state is the best estimate of ground truth. It is thus contingent on the current theoretical, computational, and operational limitations of the user.

It is common practice to use the output of a sensor such as a GPS or a Bluetooth reader that is deployed by the agency performing the validation or a third party as reference state. The choice of so-called reference devices can be dictated by the relative accuracy of the measurement devices (for example GPS versus cell tower triangulation). Although this approach offers reasonable results in terms of accuracy, it suffers from the fact that the reference state is built from only one source of knowledge, which does not offer any scientific stability or accuracy guarantee. Combining the measurements source with a traffic model within an estimation framework can provide accuracy guarantees associated with the robustness of the model and the design of the estimation method. Moreover, this technique in general makes it possible to reduce the number of deployed sensors and thus to alleviate the cost of the whole validation process.

Since it is in general not possible to estimate ground truth on the whole spatiotemporal domain of interest, the sub-domain on which the reference state is defined has to be chosen according to a careful statistical analysis. The validation method can be greatly impacted by this de facto sampling.

4.5 EXAMPLE: SEGMENT SPEED MAP

Speed map generation can be done using, for instance, the following data types (in decreasing order of accuracy):

- Segment speed
- Point locations generating segment speed
- Point speeds generating segment speeds

In this example, the validation of speed map generation using segment speed data is considered.

Step 1: Data-level validation

- *Integrity*: Check if the data satisfies the specifications for segment speed data.
- *Consistency*: Check whether the data is physically consistent across segments. This is a complex test in the speed map generation application because segment speeds can change abruptly between segments (for example, due to an accident). However, there should be some consistency between segments in general. A simple test would be to count the number of discontinuities in the data and raise a warning if it exceeds some threshold.
- *Accuracy*: Check if the data obtained is correct in terms of the segment speed reference state.

Step 2: Application-level validation

- *Compatibility*: Check whether segment speed data is appropriate for speed map generation. In this case, the check is trivial since speed map generation is a direct display of segment speed.
- *Cross-validation*: Speed map generation can also be done using point location or point speed. Cross-validation involves comparing the results of segment speed–based speed map generation with the results using alternate data types and determining which of them exhibits the highest accuracy. In this case, segment speed is expected to perform better.
- *Performance*: Determine the accuracy of the application output with the data feed considered by comparing it with the application output using the reference state feed. This is similar to the data-level accuracy test, but the application's output accuracy might be different from the feed accuracy itself.

Performing this set of validation steps provides a good assessment of the appropriateness and quality of the data feed for the application.

5 SYSTEM IMPLEMENTATION

We now illustrate the importance of the previous concepts for practitioners. This section presents an implementation of the standards discussed before within the Mobile Millennium [4] system, which is representative of modern traffic information systems.

5.1 MOBILE MILLENNIUM SYSTEM

The Mobile Millennium system [4] is a large scale traffic estimation system which integrates a variety of different data types to provide traffic estimates in the San Francisco Bay Area. The different data types collected by the system follow a sequence of steps whose aim is to assess the data feed quality, filter out low quality data points, and fuse different data types. The output of this process is fed to estimation modules, including partial differential equations based traffic models coupled with ensemble Kalman filtering, and machine learning algorithms. The outline of the system is given in Figure 2-2. Because of the complexity of the overall system, strict data type specifications and data feed quality metrics are enforced. These schema and metrics are described in the following sections.

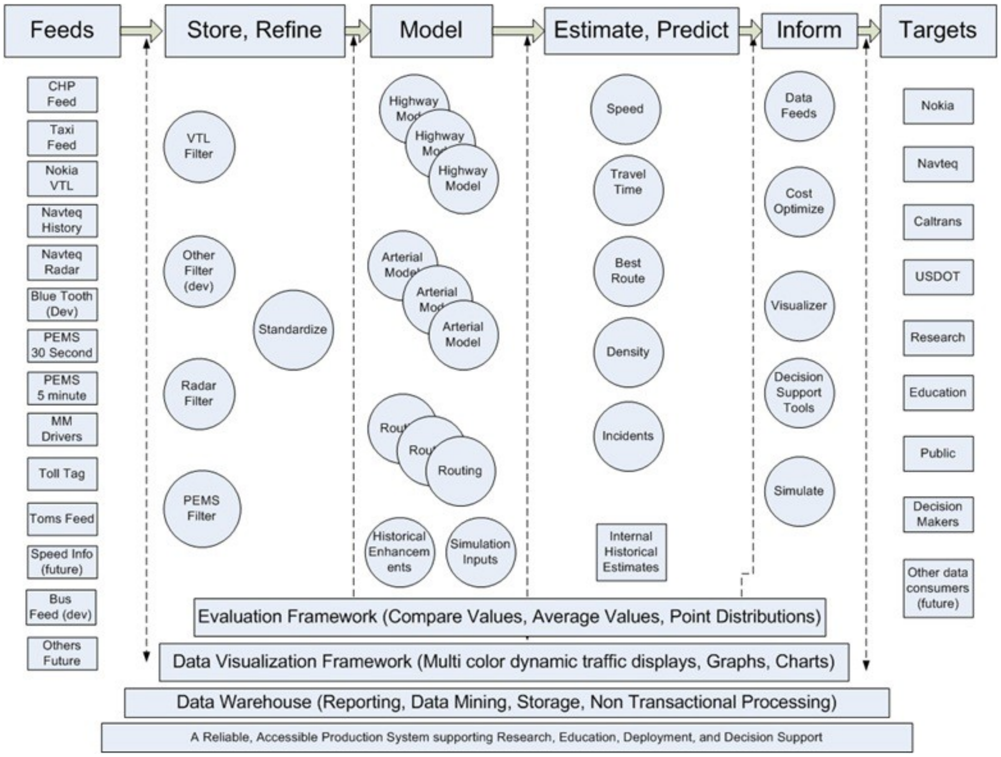


Figure 2-2: Mobile Millennium system: Data feeds entering the system go through a well-defined sequence of computational steps necessary to guarantee the quality of the final outputs.

5.2 DATA FEED REQUIREMENTS

In this section, a basis for the requirements of the most common data types used by the transportation engineering community is proposed. Similar standards [21] for transit applications have greatly contributed to the improvement of commute planning. At the more local level of the Mobile Millennium system, the standards introduced by this study provided a solid common basis for a fast and robust development of traffic estimation activities involving a variety of data feeds and multiple data types. For each data type, the specifications are organized as follows:

- *Data sources*: Most common data sources for this data type.
- *Applications*: Applications for which this data type is the most relevant.
- *Output schema*: Format of the data feed. Two types of fields are present in the output schema, required fields which have to meet the completeness requirement, and optional fields.
- *Validation schema*: Format of the validation data feed. In particular this feed should be independent from the regular data feed described by the output schema. The validation data feed should be available during a defined probationary period and should be available at pre-defined periods for standard assessment of data quality during the data feed lifetime.
- *Processed data requirements*: Requirements which have to be satisfied by processed data. These requirements depend on the processing type.
- *Individual data source requirements*: Specifications on the individual data sources used by the data feed. This includes, for instance, the sensor characteristics and the sampling scheme.
- *Data source network requirements*: Specifications on the data source network used by the feed. This includes the coverage information and the redundancy coming from the spatiotemporal spread of the data sources.
- *Processing requirement*: Extensive description of the nature and properties of the processing algorithm, with appropriate references.

In the following section these specifications are instantiated for the application of point speed estimation using the data type point location.

5.3 INSTANTIATION OF POINT LOCATION DATA FEED: TAXI DATA

In this section two data feeds available in the Mobile Millennium system are considered:

- *Cabspotting* feed [22]: Public data feed consisting of point locations recorded by taxis in the city of San Francisco, California.

- *Info24* feed [23]: Data feed consisting of point locations recorded by taxis in the city of Stockholm, Sweden.

We propose to instantiate some of the quality metrics defined in this section for data-level validation on a specific subset of these two data feeds. We consider a 8.6-mile stretch of highway between downtown San Francisco and San Francisco International Airport for the Cabspotting feed (Figure 2-3), and a 14-kilometer stretch of highway (equivalent length) between downtown Stockholm and Stockholm-Arlanda airport (Figure 2-4) for the Info24 feed. In both cases, we limit our analysis to July 21, 2010.

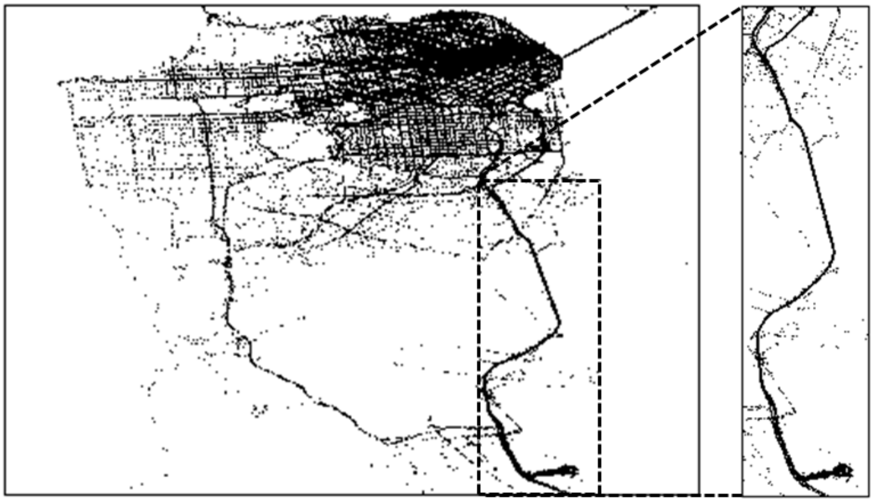


Figure 2-3: Cabspotting feed: 6 × 10^5 points (order of magnitude) received by the Mobile Millennium system on July 21, 2010

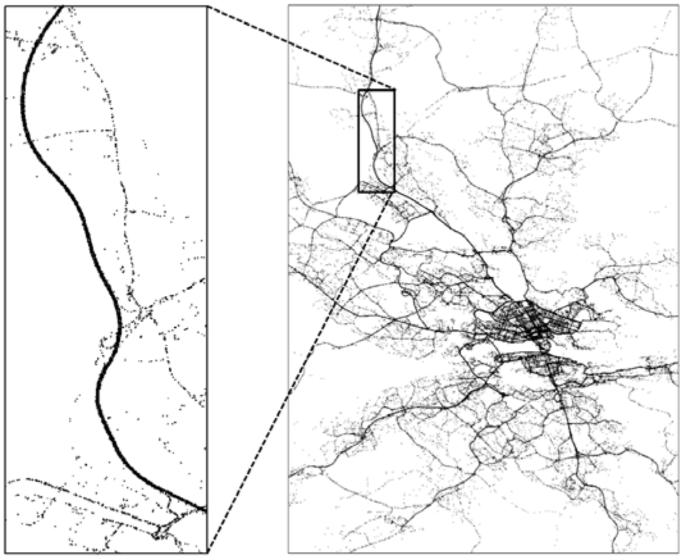


Figure 2-4: Info24 feed: 2 × 10^5 points (order of magnitude) received by the Mobile Millennium system on July 21, 2010

The schema for these two data feeds is defined below.

Data sources: GPS using trilateration

Application: Real-time point speed estimation

Output schema:

Required fields:

<time>: measurement time in GMT
<id>: unique identifier for sensor from which the measurement originates
<position>: location measurement in the reference coordinate system used by the *Global Positioning System*

Optional fields:

<error>: uncertainty on the point location (distance)
<heading>: direction of travel (angle)

The sensor ID allows computing travel time and point speed in the case of high frequency sampling with minimal processing. One may note that without the sensor ID, only the density of probe vehicles can be computed, which has some interest for applications such as queue length estimation, but requires a large volume of data points.

Validation schema:

Since this data type is a direct output of a data source with minor processing inherent to the measurement process, no additional validation data feed is required, and thus both schema match.

Processed data requirement:

Since this data type can be considered raw, no requirement is expressed in this section.

Individual data source requirements:

- *Device error characteristic* (meters): Device error characteristics are crucial parameters for data assimilation schemes in which the observation error is taken into consideration [24] [25] [26]. In an assimilation framework, the observations are accounted for with a weight corresponding to this error. When this metric is unavailable, the distance between the point location from the data feed and the map-matched point on the road network can be used as an indicator of the point location accuracy. Figure 2-5 shows the cumulative distribution of projection error for the Info24 data feed for the area described in Figure 2-4. At the 90th percentile the individual data points have a projection error lower than 8 meters.

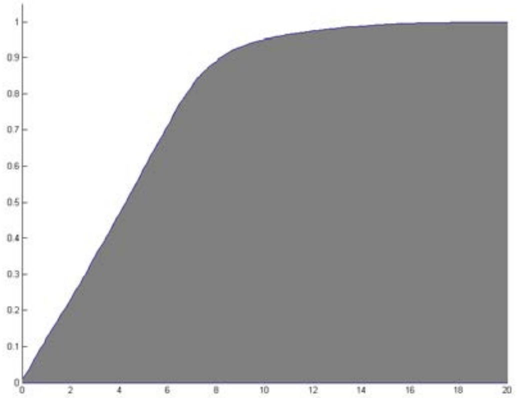


Figure 2-5: Cumulative distribution of projection error (meters)

- *Sampling period* (seconds): Sampling period impacts the usability of the data feed. It is advantageous to have as many data points as possible. The product of the sampling period and the penetration rate expresses the number of data points received on average. The sampling period is important by itself because it gives an indication of the connectivity of the reported trajectory. With a high sampling period, complex methods will have to be used to infer the trajectory of the probe vehicles between two measurements. On the other hand, a low sampling period is not desirable in a privacy-sensitive context.

- *Data transmission delay* (seconds): Duration from the time at which the data source records a measurement to the time at which the corresponding data point is available for the application considered. Figure 2-6 presents the distribution of delay in the Cabspotting data feed for the period of interest. At the 90th percentile the reports have a delay lower than 320 seconds.

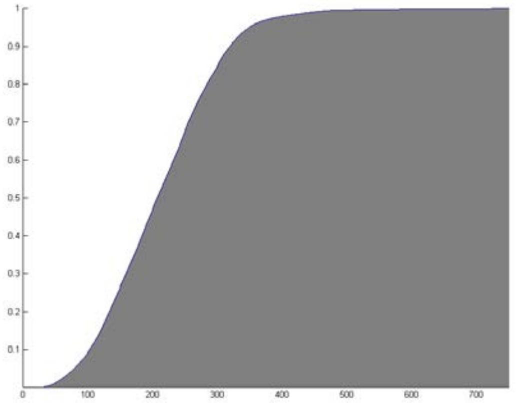


Figure 2-6: Cumulative distribution of the delay (seconds)

Sensor network requirements:

- *Space-time coverage*: The data feed should be available on the roads of major interest for the traffic monitoring authority, at times when traffic is the most uncertain, i.e., during the day and especially during peak hours. The number of measurements collected on each discretization segment of the network is a parameter which directly impacts the accuracy of the estimate. Figure 2-7 presents the distribution of the number of points per segment in the Info24 data feed for the period of interest. At the 35th percentile a network segment receives more than 200 points per day.

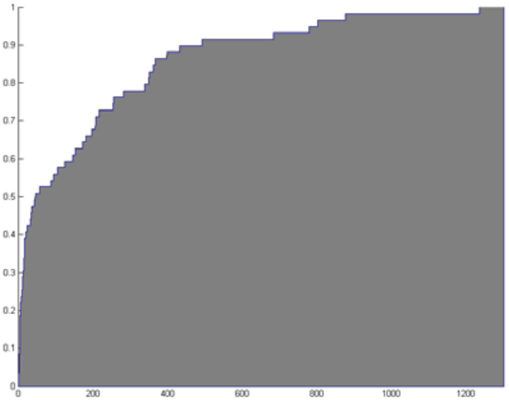


Figure 2-7: Cumulative distribution of number of measurements per network segment for one day

- *Homogeneity*: This requirement expresses that most segments should have data points from at least a given number of sensors. This avoids well-known failure situations arising in the case of loop detectors, where a sensor reporting wrong measurements is difficult to detect.
- *Penetration rate*: Proportion of the total flow consisting of equipped vehicles. The definition of the reference state for penetration rate should be documented, as it is not a straightforward quantity to estimate.

Processing requirement:

This data type is considered to be *raw*, and thus no requirement is expressed in this section.

6 CONCLUSION

This chapter presented a methodology for data quality assessment of traffic data. Recent trends of the data quality market were discussed to support the need for a new perspective on data quality assessment. In particular, it was argued that the increase in volume of traffic data leads to an increasing complexity of available traffic data feeds, and thus new standards should be defined to enable traffic engineers to properly explore the new opportunities offered by this wealth of novel traffic information. The instantiation within the Mobile Millennium system of some of the data quality metrics introduced in this study was used to illustrate the potential of the proposed framework for data quality assessment. These metrics capture the complexity of typical probe measurements and allow traffic practitioners to rate available data feeds according to their application of interest.

Chapter 3

Data Quality Tool User Guide

As part of the effort to assess data quality, the research team developed a tool to directly examine the characteristics of probe data feeds. This chapter describes what the Data Quality Tool does and how to use it.

1 INTRODUCTION

The Data Quality Tool is a web application that visualizes traffic data from multiple data feeds on multiple networks. The data collected from feeds are essentially datapoints: single devices (e.g., mobile phones) detected by feeds at a particular time and location. Each datapoint is associated with a set of attributes, such as date, location, speed, and heading.

The Data Quality Tool provides key insights on these datapoints. These insights are represented as metrics. We will discuss each of the eight metrics provided by the tool in more detail later in this chapter.

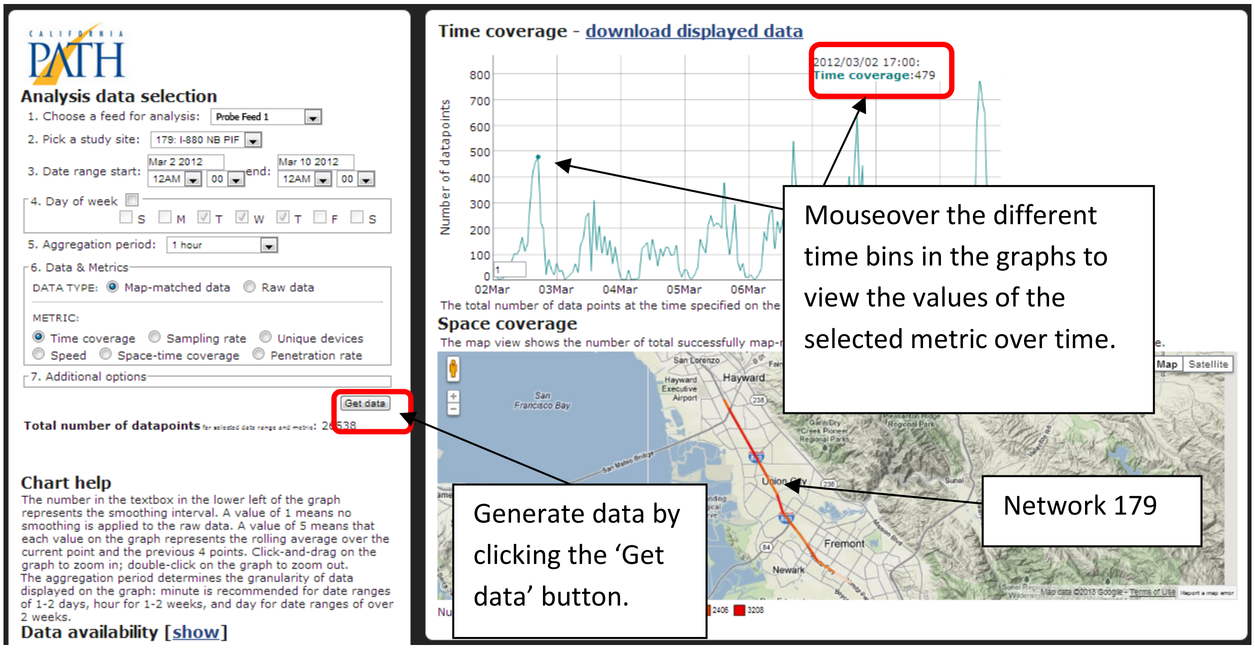
1.1 KEY CONCEPTS

Before using the tool, it is important to understand the concepts of time bins and space bins:

- **Time bins**—The tool represents time bins as periods of time in the following quantities: 1 minute; 5 minutes, 15 minutes, 30 minutes, 1 hour, 2 hours, and 1 day. Here are a few examples of time bins: a 15-minute time bin from from 12:00pm to 12:15pm on Jan 1; a 30-minute bin from 12:30pm to 1pm on Jan 1; a 1-day time bin from 12:00am on Jan 1 to 12:00am on Jan 2; etc.

- **Space bins**—The Data Quality Tool represents space bins as sections of a network. By default the tool divides networks into sections, each measuring 1600 meters (or approximately 1 mile). We will discuss space bins in more detail later.

2 NAVIGATING THE DATA QUALITY TOOL



Analysis data selection

1. Choose a feed for analysis: Probe Feed 1
2. Pick a study site: 179: I-880 NB PIF
3. Date range start: Mar 2 2012 12AM 00 end: Mar 10 2012 12AM 00
4. Day of week □ □ S □ M ☑ T ☑ W ☑ T □ F □ S
5. Aggregation period: 1 hour
6. Data & Metrics
   DATA TYPE: ● Map-matched data ○ Raw data
   METRIC: ● Time coverage ○ Sampling rate ○ Unique devices ○ Speed ○ Space-time coverage ○ Penetration rate
7. Additional options

Get data

Total number of datapoints for selected data range and metric: 26538

Chart help

The number in the textbox in the lower left of the graph represents the smoothing interval. A value of 1 means no smoothing is applied to the raw data. A value of 5 means that each value on the graph represents the rolling average over the current point and the previous 4 points. Click-and-drag on the graph to zoom in; double-click on the graph to zoom out.
The aggregation period determines the granularity of data displayed on the graph: minute is recommended for date ranges of 1-2 days, hour for 1-2 weeks, and day for date ranges of over 2 weeks.

Data availability [show]

Time coverage - download displayed data

2012/03/02 17:00:
Time coverage:479

The total number of data points at the time specified on the

Space coverage

The map view shows the number of total successfully map-r

Mouseover the different time bins in the graphs to view the values of the selected metric over time.

Generate data by clicking the 'Get data' button.

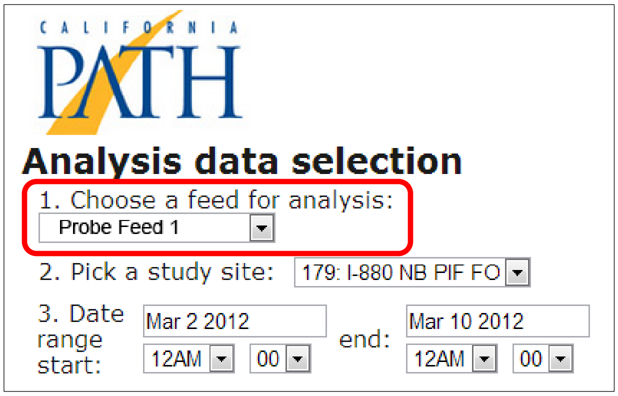
Network 179

Figure 3-1: Data Quality Tool default page

Figure 3-1 shows the default page of the Data Quality Tool. The left panel contains all input fields for the tool. The right panel displays a graph and a map based on the values of the input fields. The output is generated by selecting the inputs in the left panel, then clicking the 'Get Data' button located below the input fields. Output is also generated by selecting any of the options in field 6, 'Data & Metrics'.

The following sections describe each of the input fields and options. To view the same data shown in this guide (if available) on your version of the Data Quality Tool, simply select identical input options in the left panel, then click the 'Get Data' button.

2.1 SELECTING A DATA FEED



Analysis data selection

1. Choose a feed for analysis: Probe Feed 1

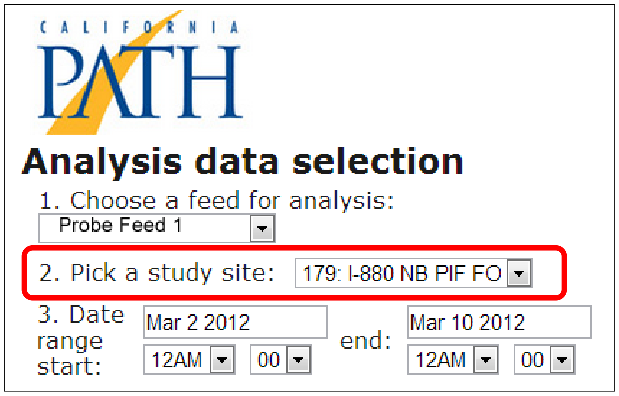
2. Pick a study site: 179: I-880 NB PIF FO

3. Date range start: Mar 2 2012 12AM 00 end: Mar 10 2012 12AM 00

Field 1 contains a dropdown list of all available data feeds. Data feeds usually represent companies or entities that are in the business of collecting probe data, or datapoints.

Users can select only one feed at a time for analysis.

2.2 SELECTING A NETWORK (STUDY SITE)



Analysis data selection

1. Choose a feed for analysis: Probe Feed 1

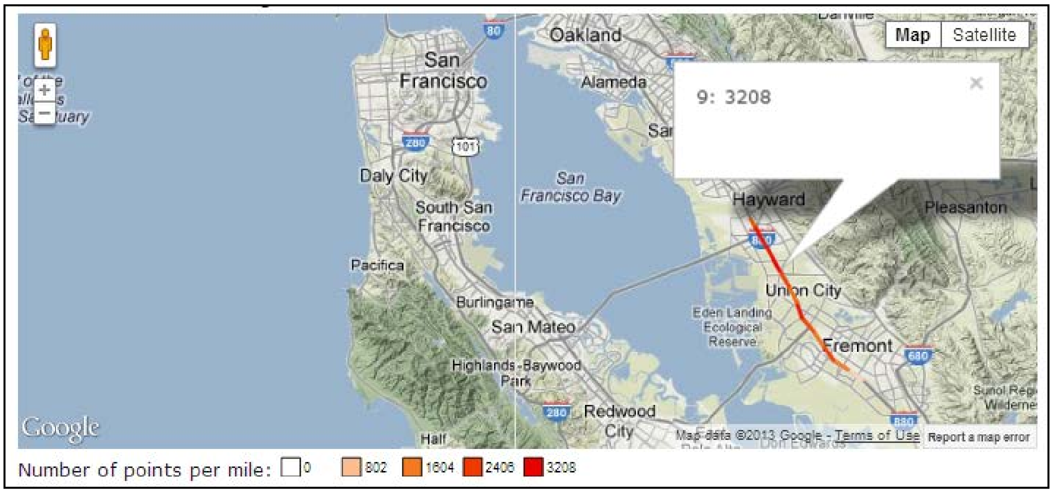
2. Pick a study site: 179: I-880 NB PIF FO

3. Date range start: Mar 2 2012 12AM 00 end: Mar 10 2012 12AM 00

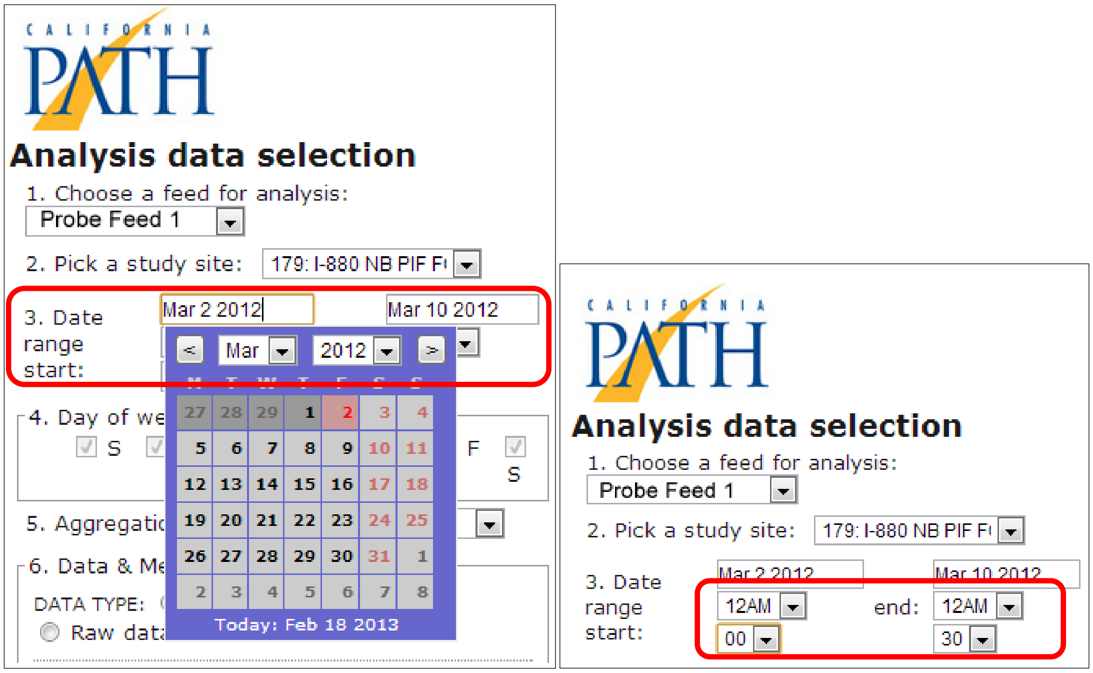
Field 2 lists all available networks. In this user guide, we will be illustrating most examples with data from network 179, an approximately 12-mile section of the I-880 freeway northbound starting from the city of Fremont, California and ending in the city of Hayward, California.

The map in the DQ tool (shown below) highlights the selected network, 179. Clicking on different areas of the highlighted network on the map displays the number of datapoints collected by the selected data feed for the selected section of the network within the selected date range. The map in the tool below shows that there were 3208 datapoints found in section 9 of network 179 from Mar 2 2012 at 12am and before Mar 10 2012 at 12am. In the map below, section 9 roughly represents the ninth mile of network 179.

Note that maps highlighting the selected network should appear in the right panel of all screens in the Data Quality Tool whenever output is generated for valid input. The functionality of the highlighted highway remains exactly the same on every screen where it appears. The highlighted network is clickable at different segments of the route. Each segment of the network represents a 1600-meter stretch of road (roughly one mile) . The number that appears after clicking the segment represents the number of datapoints in that 1600-meter segment of the network within the date range determined by the values in the start and end date fields.



2.3 SELECTING A DATE RANGE



CALIFORNIA PATH

Analysis data selection

1. Choose a feed for analysis: Probe Feed 1

2. Pick a study site: 179: I-880 NB PIF Fi

3. Date range start: Mar 2 2012 Mar 10 2012

4. Day of we

5. Aggregatic

6. Data & Me

DATA TYPE:

○ Raw dat

CALIFORNIA PATH

Analysis data selection

1. Choose a feed for analysis: Probe Feed 1

2. Pick a study site: 179: I-880 NB PIF Fi

3. Date range start: Mar 2 2012 12AM 00 end: Mar 10 2012 12AM 30

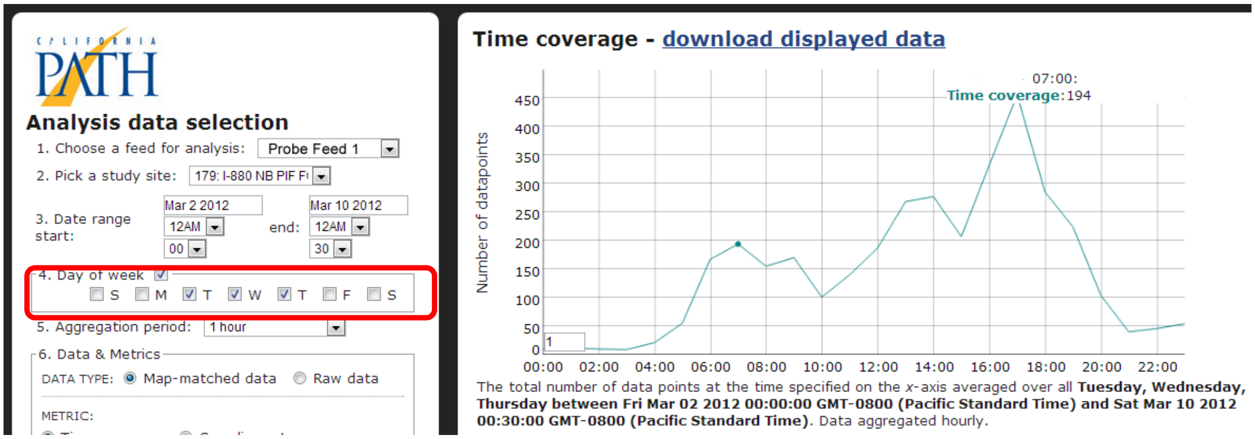
Figure 3-2: Date and Time fields

Figure 3-2 defines the date and time range of the data displayed for analysis. The data being analyzed will include all datapoints whose associated dates are greater than or equal to the start date and whose associated dates are less than the end date (i.e., excluding datapoints at the end date). Users can manually type in dates in the start date and end date fields. If users manually type in dates, the format must be exactly as follows: 1) the three-character alphabetic month (e.g., Mar) followed by one space; 2) one- or two-digit day followed by one space; then 3) the four-digit year. For example, **Mar 2 2012** or **Mar 02 2012** would both be valid strings to enter in the date fields.

Clicking in either the start or end date fields displays a calendar on which users can click and select dates. Users can select the time of day just below the date fields. Time of day on the hour (e.g., 12:00AM), 15 minutes, 30 minutes, and 45 minutes after the hours (e.g., 2:15pm, 2:30pm; 2:45pm), can be selected.

Note that networks commonly only contain data for a short range of days. Therefore, it can be cumbersome to search for valid date ranges using the tool. For example, as of February 2013, only data from Mar 2, 2012 to Mar 20, 2012 existed for network 179. To find valid date ranges for your network of interest, request someone with database access to find distinct days for which data exists for your network of interest.

2.4 AVERAGING DATA OVER THE DAY OF WEEK



Analysis data selection

1. Choose a feed for analysis: Probe Feed 1
2. Pick a study site: 179: I-880 NB PIF F
3. Date range start: Mar 2 2012 12AM 00 end: Mar 10 2012 12AM 30
4. Day of week ☑ □ S □ M ☑ T ☑ W ☑ T □ F □ S
5. Aggregation period: 1 hour
6. Data & Metrics

DATA TYPE: ● Map-matched data ○ Raw data

METRIC:

Time coverage - download displayed data

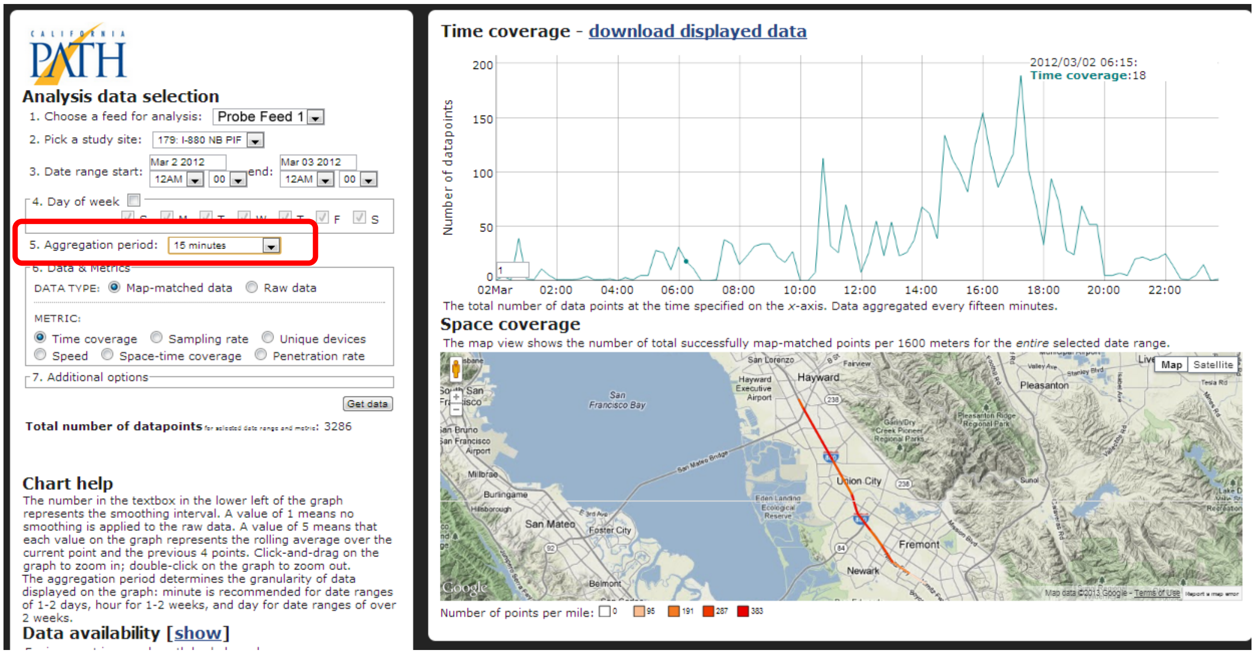
The total number of data points at the time specified on the *x*-axis averaged over all **Tuesday, Wednesday, Thursday between Fri Mar 02 2012 00:00:00 GMT-0800 (Pacific Standard Time) and Sat Mar 10 2012 00:30:00 GMT-0800 (Pacific Standard Time)**. Data aggregated hourly.

Figure 3-3: The Day of Week option averages values over the days of the week.

Checking the Day of Week checkbox in Field 4 generates a graph that displays the average value of a metric over selected days of the week. The report in Figure 3-3 shows an average of 194 datapoints in the 1-hour time bin from 7:00am to 8:00am (excluding datapoints at 8:00am) on all Tuesdays, Wednesdays, and Thursdays between Mar 2 2012 and Mar 10 2012. Here, we know that the time bin has a size of 1 hour because the next field we will describe, ‘Aggregation period’, contains a value of 1 hour. In addition, the description below the graph indicates ‘**Data aggregated hourly’**.

The x-axis will always represent a single 24-hour period. This is expected since the Day of Week feature gives the average measure of a metric over the selected days at a particular time bin. This will be the case for all metrics when the Day of Week feature is selected.

2.5 SELECTING AN AGGREGATION PERIOD



Analysis data selection

1. Choose a feed for analysis: Probe Feed 1
2. Pick a study site: 179: I-880 NB PIF
3. Date range start: Mar 2 2012 12AM 00 end: Mar 03 2012 12AM 00
4. Day of week □ ☑ S ☑ M ☑ T ☑ W ☑ T ☑ F ☑ S
5. Aggregation period: 15 minutes
6. Data & Metrics

DATA TYPE: ● Map-matched data ○ Raw data

METRIC: ● Time coverage ○ Sampling rate ○ Unique devices ○ Speed ○ Space-time coverage ○ Penetration rate

7. Additional options

Total number of datapoints for selected date range and metric: 3286

Chart help

The number in the textbox in the lower left of the graph represents the smoothing interval. A value of 1 means no smoothing is applied to the raw data. A value of 5 means that each value on the graph represents the rolling average over the current point and the previous 4 points. Click-and-drag on the graph to zoom in; double-click on the graph to zoom out. The aggregation period determines the granularity of data displayed on the graph: minute is recommended for date ranges of 1-2 days, hour for 1-2 weeks, and day for date ranges of over 2 weeks.

Data availability [show]

Time coverage - download displayed data

2012/03/02 06:15: Time coverage:18

The total number of data points at the time specified on the *x*-axis. Data aggregated every fifteen minutes.

Space coverage

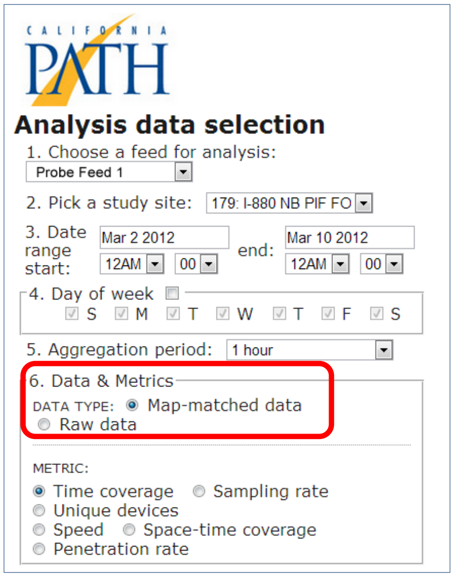
The map view shows the number of total successfully map-matched points per 1600 meters for the *entire* selected date range.

Number of points per mile: □ 0 🟧 95 🟧 191 🟥 287 🟥 383

Figure 3-4: Aggregation Period

The Aggregation Period field determines the size of the time bins on the x-axis. The possible values for this field are: 1 minute, 5 minutes, 15 minutes, 30 minutes, 1 hour, and 1 day. Therefore, if an aggregation period of 15 minutes is chosen, the graph will display data for all 15-minute time bins as shown in Figure 3-4, where we observe 18 datapoints in the time bin from 6:15am to 6:30am on March 2 2012 (excluding datapoints at 6:30am). Together with the date and time range, the Aggregation Period defines the time bins on the x-axis.

2.6 SELECTING A DATA TYPE



Analysis data selection

1. Choose a feed for analysis: Probe Feed 1

2. Pick a study site: 179: I-880 NB PIF FO

3. Date range start: Mar 2 2012 12AM 00 end: Mar 10 2012 12AM 00

4. Day of week □ ☑ S ☑ M ☑ T ☑ W ☑ T ☑ F ☑ S

5. Aggregation period: 1 hour

6. Data & Metrics

DATA TYPE: ● Map-matched data ○ Raw data

METRIC:
● Time coverage ○ Sampling rate
○ Unique devices
○ Speed ○ Space-time coverage
○ Penetration rate

Figure 3-5: Data Type

The 'DATA TYPE' option filters 'Map-matched Data' or 'Raw Data'. (Map-matched data is GPS probe data that has been processed through the Path Inference Filter described in Chapter 4. The filter maps the probe data points to the road network and plots their trajectories.)

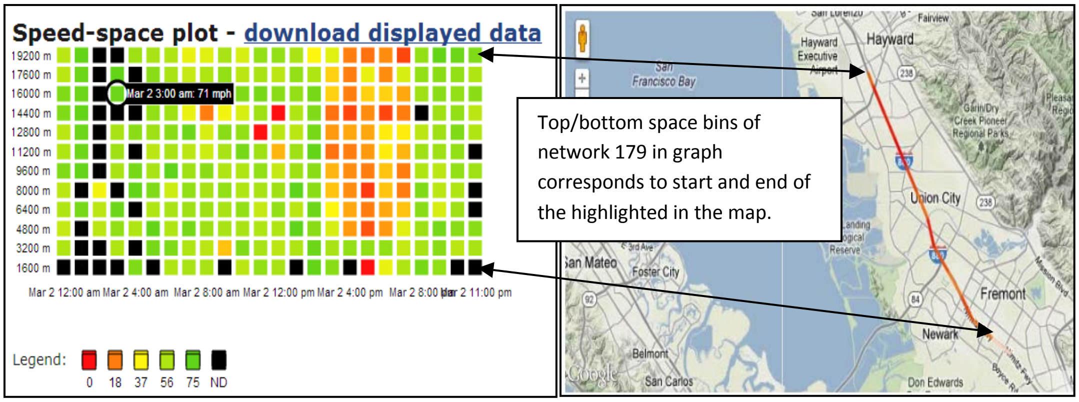
Map-matched data in the DQ Tool refers to datapoints that are associated with sections of networks. We identify these sections of the network as "space bins" that can be visualized on the y-axis of the Speed metric in Figure 3-6 below.

By default, the DQ Tool sets the size of these space bins to 1600 meters (approximately 1 mile).

On the y-axis are 12 space bins. The first space bin (labeled '1600 m') represents the space bin from the beginning of network 179 to the 1600th meter in the network (approximately the 1-mile mark) and the top-most space bin (labeled '19200 m') represents the final bin in the network (approximately the last mile of the network).

Note, the start and end may vary depending on the direction of the network. The representation below in Figure 3-6 is convenient because 179 is a northbound network.

If the network highlighted in the map below ran southbound, the first space bin located at the bottom of the graph of Figure 3-6 would represent the northernmost part of the highlighted network and the space bin at the top of the graph would represent the southernmost point of the highlighted network in the map. If the network were eastbound, the first space bin would correspond to the westernmost part of the highlighted network in the map and the space bin at the top would correspond to the easternmost part; and vice versa if the highlighted network were westbound.



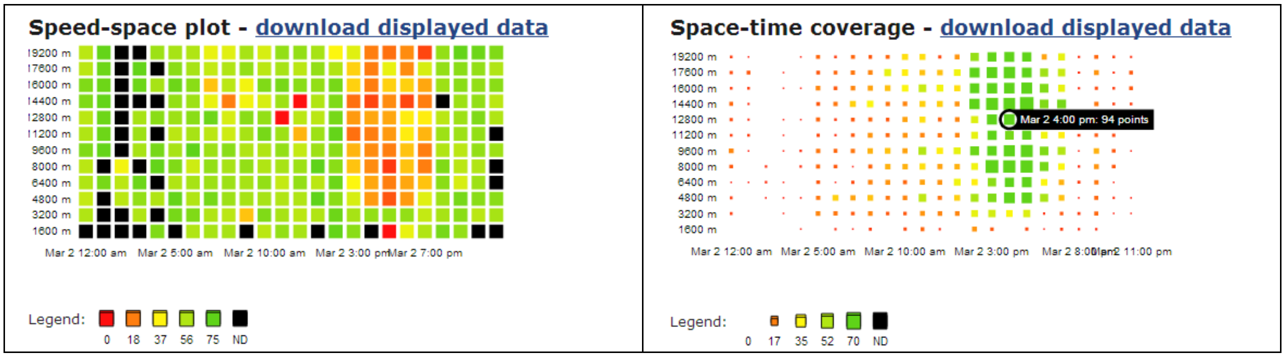
Top/bottom space bins of network 179 in graph corresponds to start and end of the highlighted in the map.

Figure 3-6: Map-Matched graph and corresponding highlighted network on the map

The size of the space bins can be configured to a minimum of 100 meters. Space bins can also be configured to be larger than 1600 meters, but the rule of thumb for sizing space bins for visual purposes is to keep the total number of space bins at a minimum of 7; otherwise, the space bins may not display clearly.

Raw data, on the other hand, are not associated with specific segments of the network; we only know that raw datapoints were located within the bounding polygon region associated with the network of interest.

There are two metrics that are true map-matched metrics, meaning their y-axes represent space bins: 1) Speed; and 2) Space-time Coverage. These metrics can only be displayed using map-matched data, not using raw data.



DQ Tool's two map-matched metrics: Speed and Space-time Coverage

There are three metrics that are not true map-matched metrics (meaning their y-axes do not represent space bins), but can still be displayed in the DQ Tool using map-matched data. These metrics, which will be described in more detail in the next section, are: 1) Time Coverage; 2) Sampling Rate; and 3) Unique

Devices. The purpose of displaying these metrics using map-matched data is to be able to compare map-matched datapoints with raw datapoints for these metrics.

In addition, there are two more metrics which can only be displayed using raw data: 1) Provider Transmission Delay and 2) Total Transmission Delay.

The following table lists each metric of the DQ Tool and indicates what type of datapoints can be used to display each metric.

Metric	Can Be Displayed with Map-Matched Data?	Can Be Displayed with Raw Data?	True Map-matched Metric? (i.e., y-axis represents space bins)
Time Coverage	X	X	
Sampling Rate	X	X	
Unique Devices	X	X	
Speed	X		X
Space-time Coverage	X		X
Provider Transmission Delay		X	
Total Transmission Delay		X	

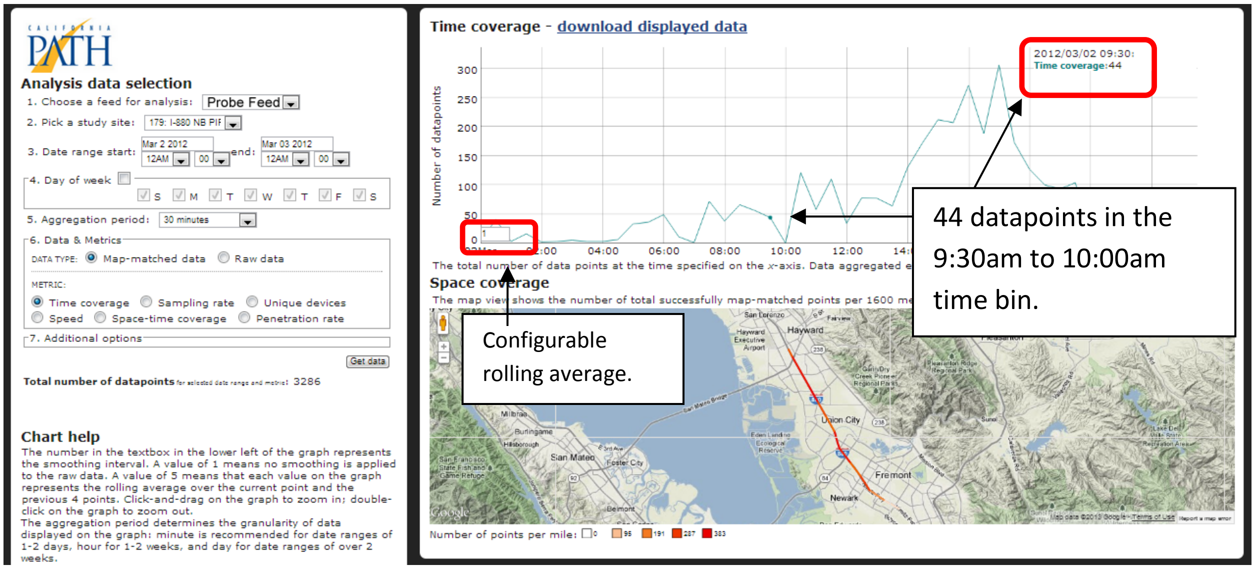
These metrics are described in more detail in the following section.

3 METRICS

To view any of the figures displayed in this document on your version of the Data Quality Tool, simply select identical input options in the left panel (if available), then click the Get Data button.

Note: Although shown in the screenshots, the **Penetration rate** metric is not currently operational in the DQ Tool. It is a placeholder for future development.

3.1 TIME COVERAGE



Analysis data selection

1. Choose a feed for analysis: Probe Feed
2. Pick a study site: 179: I-880 NB PIF
3. Date range start: Mar 2 2012 12AM 00 end: Mar 03 2012 12AM 00
4. Day of week □ ☑ S ☑ M ☑ T ☑ W ☑ T ☑ F ☑ S
5. Aggregation period: 30 minutes
6. Data & Metrics
   DATA TYPE: ● Map-matched data ○ Raw data
   METRIC: ● Time coverage ○ Sampling rate ○ Unique devices ○ Speed ○ Space-time coverage ○ Penetration rate
7. Additional options

Get data

Total number of datapoints for selected date range and metric: 3286

Chart help

The number in the textbox in the lower left of the graph represents the smoothing interval. A value of 1 means no smoothing is applied to the raw data. A value of 5 means that each value on the graph represents the rolling average over the current point and the previous 4 points. Click-and-drag on the graph to zoom in; double-click on the graph to zoom out.
The aggregation period determines the granularity of data displayed on the graph: minute is recommended for date ranges of 1-2 days, hour for 1-2 weeks, and day for date ranges of over 2 weeks.

Time coverage - download displayed data

2012/03/02 09:30:
Time coverage:44

The total number of data points at the time specified on the x-axis. Data aggregated e

Space coverage

The map view shows the number of total successfully map-matched points per 1600 me

Number of points per mile: □ 0 95 191 287 383

44 datapoints in the 9:30am to 10:00am time bin.

Configurable rolling average.

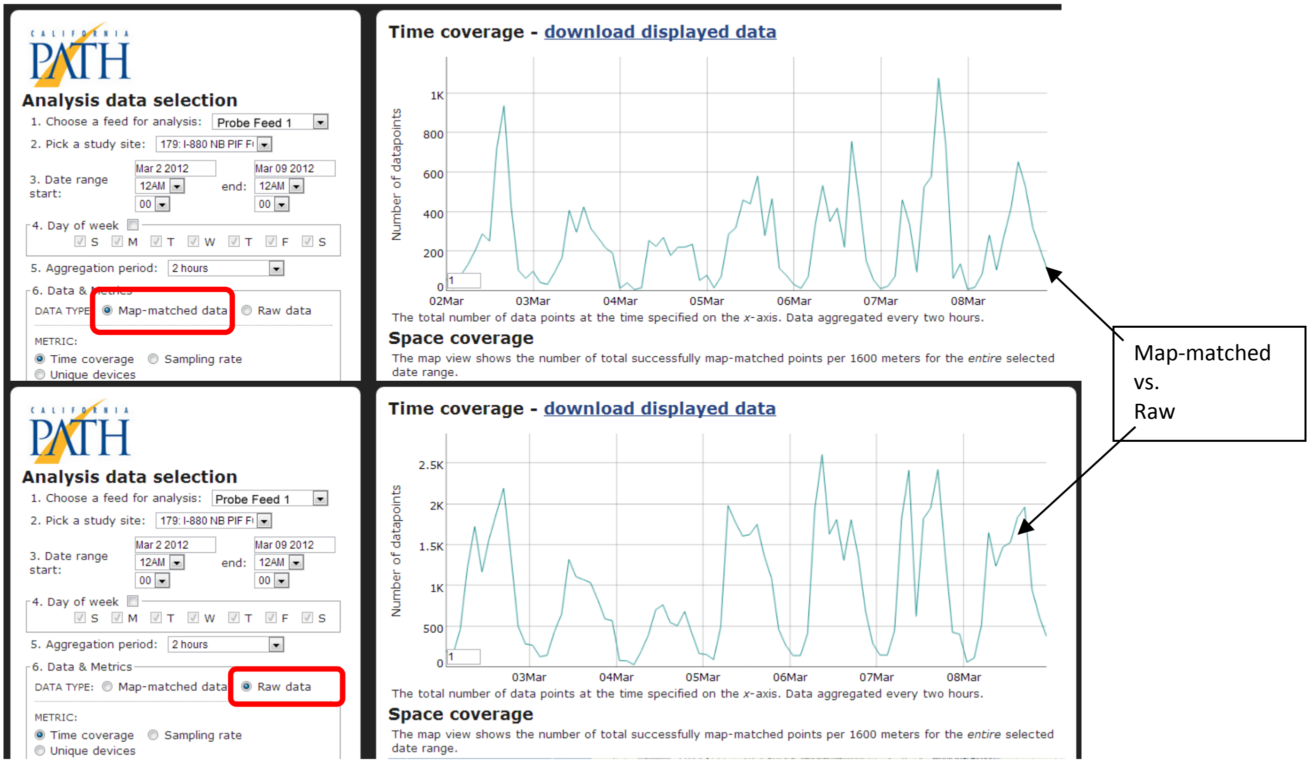
Figure 3-7: Time Coverage Metric

The Time Coverage metric represents the total number of datapoints found in the selected network within a specific time bin. The graph in Figure 3-7 shows that the 30-minute time bin from 9:30am to 10:00am on March 2, 2012 contains 44 datapoints in network 179.

As was discussed in the previous section, 2.6, although the Time Coverage metric can be displayed using both map-matched data as in Figure 3-6, this metric is not truly map-matched since its measure on the y-axis counts all the datapoints in the entire network, not within specified time bins within the network as is the case for the Speed and Time Coverage metrics.

The graph in Figure 3-7 has a configurable field for a rolling average in the graphs. The scale of this field is 1 aggregation Period. For example, a value of 1 in Figure 3-7 represents a rolling average of 30 minutes; a value of 2 represents a rolling average of 60 minutes; a value of 3 would represent 90 minutes; etc.

Being able to display the Time Coverage metric using both map-matched or raw data allows users to compare the two types of data. Using the same date range and aggregation period for network 179 below, we see that the general pattern of this metric is the same for both map-matched and raw data even though the actual number of datapoints (values on the y-axis) may differ:


Analysis data selection

1. Choose a feed for analysis: Probe Feed 1
2. Pick a study site: 179: I-880 NB PIF F
3. Date range start: Mar 2 2012 12AM 00 end: Mar 09 2012 12AM 00
4. Day of week □ ☑ S ☑ M ☑ T ☑ W ☑ T ☑ F ☑ S
5. Aggregation period: 2 hours
6. Data & Metrics

DATA TYPE: ● Map-matched data ○ Raw data

METRIC: ● Time coverage ○ Sampling rate ○ Unique devices

Time coverage - download displayed data

The total number of data points at the time specified on the *x*-axis. Data aggregated every two hours.

Space coverage

The map view shows the number of total successfully map-matched points per 1600 meters for the *entire* selected date range.

Analysis data selection

1. Choose a feed for analysis: Probe Feed 1
2. Pick a study site: 179: I-880 NB PIF F
3. Date range start: Mar 2 2012 12AM 00 end: Mar 09 2012 12AM 00
4. Day of week □ ☑ S ☑ M ☑ T ☑ W ☑ T ☑ F ☑ S
5. Aggregation period: 2 hours
6. Data & Metrics

DATA TYPE: ○ Map-matched data ● Raw data

METRIC: ● Time coverage ○ Sampling rate ○ Unique devices

Time coverage - download displayed data

The total number of data points at the time specified on the *x*-axis. Data aggregated every two hours.

Space coverage

The map view shows the number of total successfully map-matched points per 1600 meters for the *entire* selected date range.

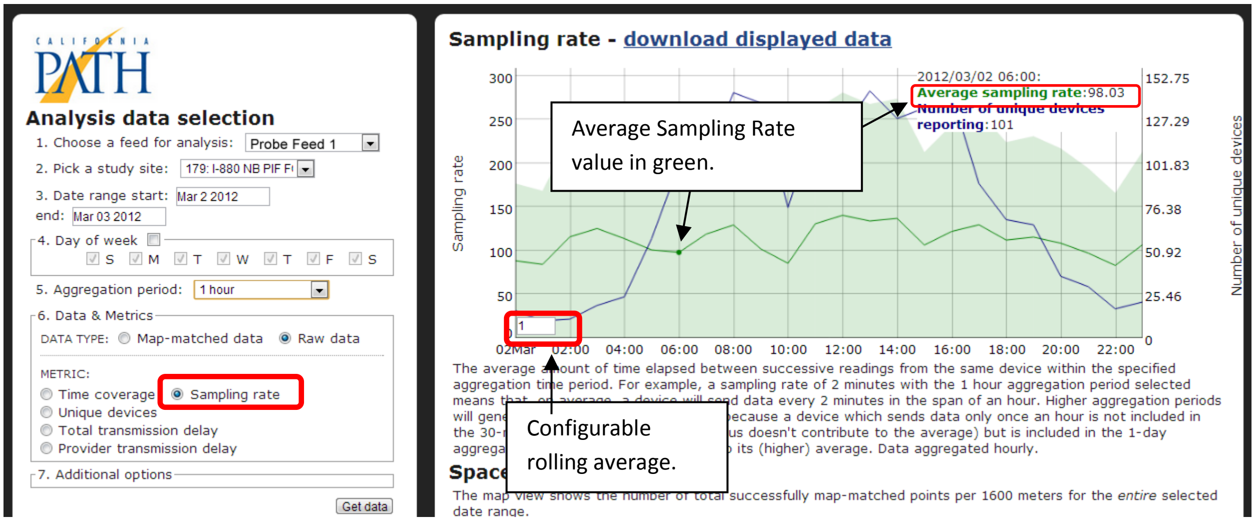
Map-matched
vs.
Raw

Comparing Map-matched versus Raw Data for Time Coverage Metric

Comparisons such as these can be made for the other 2 metrics that can be displayed with both map-matched and raw data: Sampling Rate and Unique Devices.

Also, all metrics in the DQ Tool (except for Speed and Space-time Coverage) have a configurable field for rolling averages in their graphs. The scale of this field is 1 aggregation Period. For example, here a value of 1 represents a rolling average of two hours.

3.2 SAMPLING RATE



Analysis data selection

1. Choose a feed for analysis: Probe Feed 1
2. Pick a study site: 179: I-880 NB PIF F
3. Date range start: Mar 2 2012 end: Mar 03 2012
4. Day of week □ ☑ S ☑ M ☑ T ☑ W ☑ T ☑ F ☑ S
5. Aggregation period: 1 hour
6. Data & Metrics

DATA TYPE: ○ Map-matched data ● Raw data

METRIC: ○ Time coverage ● Sampling rate ○ Unique devices ○ Total transmission delay ○ Provider transmission delay

7. Additional options

Sampling rate - download displayed data

2012/03/02 06:00:
Average sampling rate:98.03
Number of unique devices reporting:101

The average amount of time elapsed between successive readings from the same device within the specified aggregation time period. For example, a sampling rate of 2 minutes with the 1 hour aggregation period selected means that on average, a device will send data every 2 minutes in the span of an hour. Higher aggregation periods will gene… because a device which sends data only once an hour is not included in the 30-… …us doesn't contribute to the average) but is included in the 1-day aggrega… …its (higher) average. Data aggregated hourly.

Space…

The map view shows the number of total successfully map-matched points per 1600 meters for the *entire* selected date range.

Average Sampling Rate value in green.

Configurable rolling average.

Figure 3-8: Sampling Rate Metric

Selecting the Sampling Rate metric in the DQ Tool displays two metrics on a graph: Sampling Rate and Unique Devices.

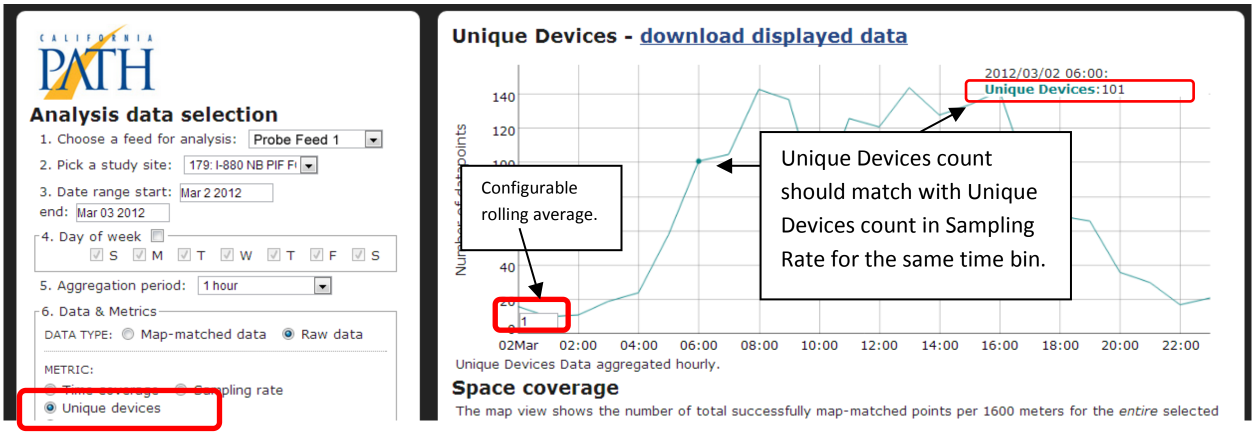
Note: Although labeled as "rate," the Sampling Rate metric presents data as *seconds* rather than *hertz*. It is thus actually displaying the *sampling period* and should be understood as such.

The Sampling Rate metric refers to the average number of seconds between successive detections of a device. In order to calculate a Sampling Rate for a device, there must be at least 2 detections for a device within a single time bin. The more times a device is detected within a time bin, the lower the value of the Sampling Rate.

Figure 3-8 shows 101 unique devices, each of which were detected at least twice in the 6:00am to 7:00am time bin on March 2, 2012. Figure 3-8 indicates that for the 101 Unique Devices that were detected at least twice within this 1-hour time bin, the average number of seconds between successive detections of each device was 98.03 seconds.

The Sampling Rate metric can be viewed using both map-matched and raw data.

3.3 UNIQUE DEVICES



CALIFORNIA PATH

Analysis data selection

1. Choose a feed for analysis: Probe Feed 1
2. Pick a study site: 179: I-880 NB PIF F
3. Date range start: Mar 2 2012
   end: Mar 03 2012
4. Day of week □
   ☑ S ☑ M ☑ T ☑ W ☑ T ☑ F ☑ S
5. Aggregation period: 1 hour
6. Data & Metrics
   DATA TYPE: ○ Map-matched data ● Raw data
   METRIC:
   ○ Time coverage ○ Sampling rate
   ● Unique devices

Unique Devices - download displayed data

2012/03/02 06:00:
Unique Devices: 101

Configurable rolling average.

Unique Devices count should match with Unique Devices count in Sampling Rate for the same time bin.

Unique Devices Data aggregated hourly.

Space coverage

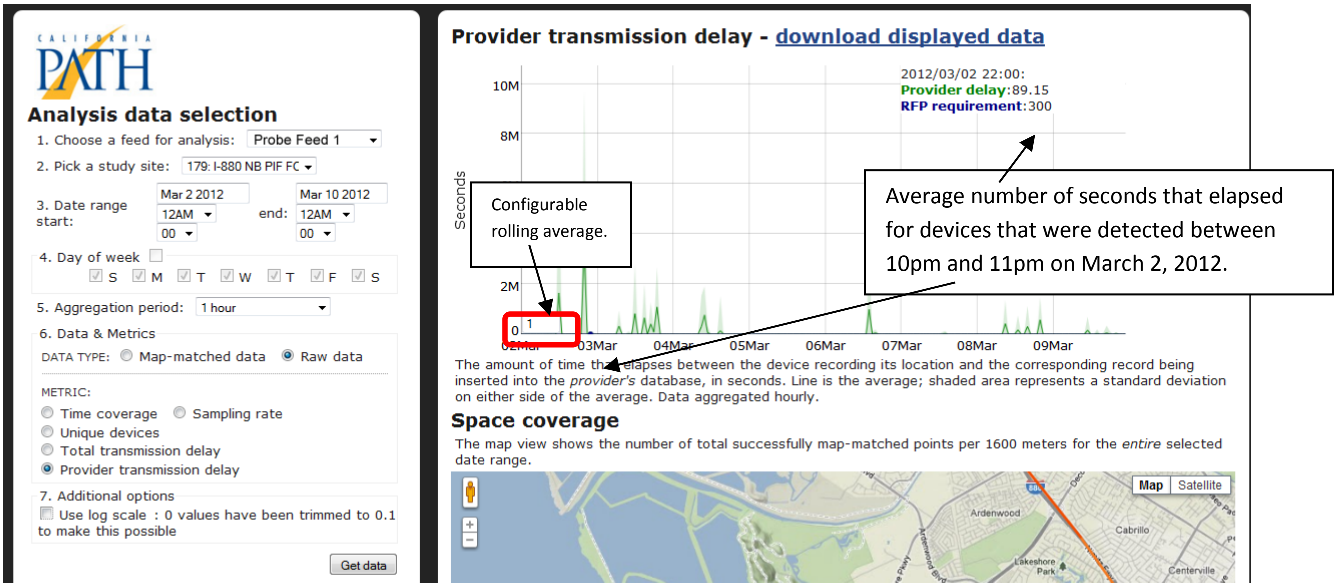
The map view shows the number of total successfully map-matched points per 1600 meters for the *entire* selected

Figure 3-9: Unique Devices Metric

The Unique Devices metric can be displayed by itself. As before, this count represents all devices in the given time bin that were detected at least twice. Figure 3-9 indicates that there are 101 Unique Devices which were detected at least twice in the 6:00am to 7:00am time bin on March 2, 2012. This is the same count shown for Unique Devices shown in the same time bin in the previous example for the Sampling Rate metric.

The Unique Devices metric can be viewed using map-matched and raw data.

3.4 PROVIDER TRANSMISSION DELAY



Analysis data selection

1. Choose a feed for analysis: Probe Feed 1
2. Pick a study site: 179: I-880 NB PIF FC
3. Date range start: Mar 2 2012 12AM 00 end: Mar 10 2012 12AM 00
4. Day of week □ ☑ S ☑ M ☑ T ☑ W ☑ T ☑ F ☑ S
5. Aggregation period: 1 hour
6. Data & Metrics

DATA TYPE: ○ Map-matched data ● Raw data

METRIC: ○ Time coverage ○ Sampling rate ○ Unique devices ○ Total transmission delay ● Provider transmission delay

7. Additional options

□ Use log scale : 0 values have been trimmed to 0.1 to make this possible

Get data

Provider transmission delay - download displayed data

2012/03/02 22:00:
Provider delay:89.15
RFP requirement:300

Configurable rolling average.

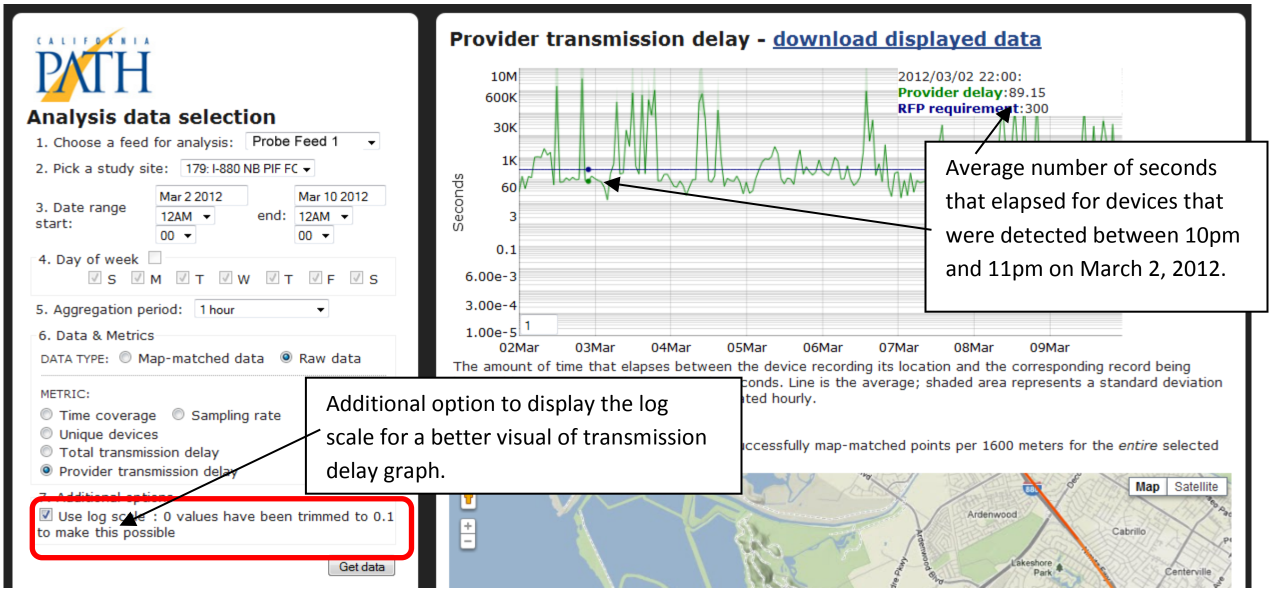
Average number of seconds that elapsed for devices that were detected between 10pm and 11pm on March 2, 2012.

The amount of time that elapses between the device recording its location and the corresponding record being inserted into the *provider's* database, in seconds. Line is the average; shaded area represents a standard deviation on either side of the average. Data aggregated hourly.

Space coverage

The map view shows the number of total successfully map-matched points per 1600 meters for the *entire* selected date range.

Figure 3-10: Provider Transmission Delay Metric



Analysis data selection

1. Choose a feed for analysis: Probe Feed 1
2. Pick a study site: 179: I-880 NB PIF FC
3. Date range start: Mar 2 2012 12AM 00 end: Mar 10 2012 12AM 00
4. Day of week □ ☑ S ☑ M ☑ T ☑ W ☑ T ☑ F ☑ S
5. Aggregation period: 1 hour
6. Data & Metrics

DATA TYPE: ○ Map-matched data ● Raw data

METRIC: ○ Time coverage ○ Sampling rate ○ Unique devices ○ Total transmission delay ● Provider transmission delay

7. Additional options

☑ Use log scale : 0 values have been trimmed to 0.1 to make this possible

Get data

Provider transmission delay - download displayed data

2012/03/02 22:00:
Provider delay:89.15
RFP requirement:300

Average number of seconds that elapsed for devices that were detected between 10pm and 11pm on March 2, 2012.

The amount of time that elapses between the device recording its location and the corresponding record being conds. Line is the average; shaded area represents a standard deviation ted hourly.

Additional option to display the log scale for a better visual of transmission delay graph.

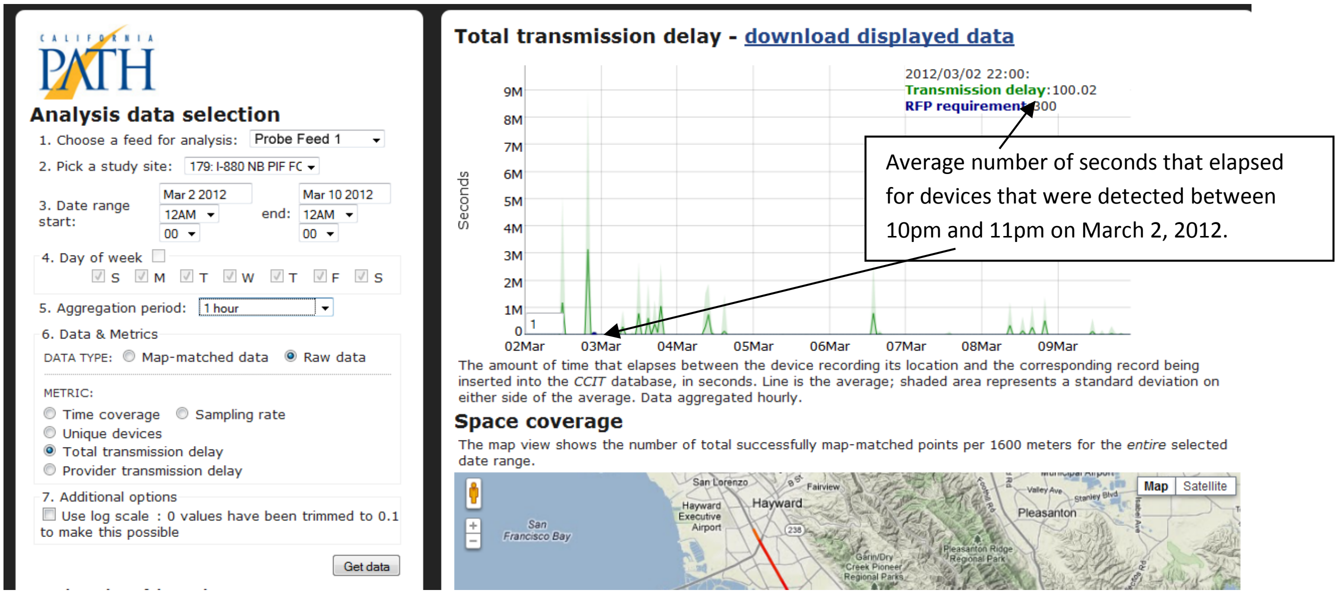
ccessfully map-matched points per 1600 meters for the *entire* selected

Figure 3-11: Provider Transmission Delay Metric (using log scale option)

Provider Transmission Delay represents the average amount of time that elapsed (in seconds) between the time devices were detected and the time the detection was inserted into the **Provider's** database. The green lines in Figure 3-10 and Figure 3-11 represent the average amount of time elapsed for devices detected between 10pm and 11pm on Mar 2, 2012 before the detection was inserted into the PATH database. Figure 3-11 is the same representation of Figure 3-10, but using the “Use log scale” option in Field 7 ‘Addition option’ in order to display the graph more clearly.

The Provider Transmission Delay metric can only be viewed using raw data.

3.5 TOTAL TRANSMISSION DELAY



CALIFORNIA PATH

Analysis data selection

1. Choose a feed for analysis: Probe Feed 1
2. Pick a study site: 179: I-880 NB PIF FC
3. Date range start: Mar 2 2012 12AM 00 end: Mar 10 2012 12AM 00
4. Day of week □ ☑ S ☑ M ☑ T ☑ W ☑ T ☑ F ☑ S
5. Aggregation period: 1 hour
6. Data & Metrics
   DATA TYPE: ○ Map-matched data ● Raw data
   METRIC:
   ○ Time coverage ○ Sampling rate
   ○ Unique devices
   ● Total transmission delay
   ○ Provider transmission delay
7. Additional options
   □ Use log scale : 0 values have been trimmed to 0.1 to make this possible

Get data

Total transmission delay - download displayed data

2012/03/02 22:00:
Transmission delay:100.02
RFP requirement:300

Average number of seconds that elapsed for devices that were detected between 10pm and 11pm on March 2, 2012.

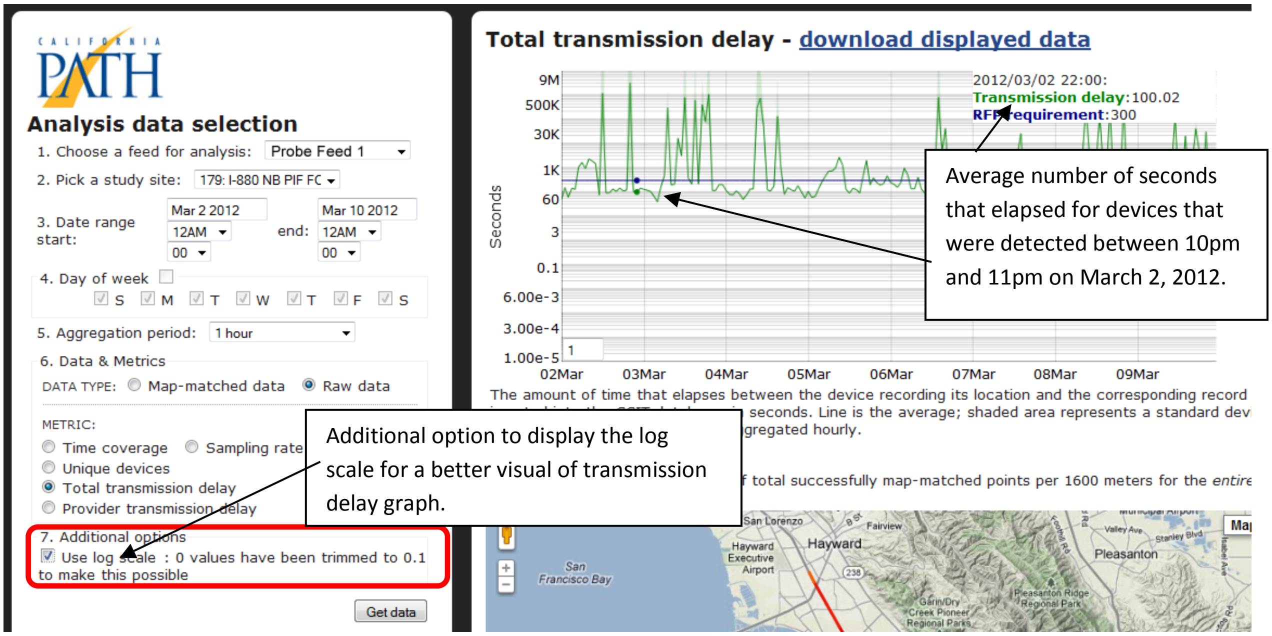
The amount of time that elapses between the device recording its location and the corresponding record being inserted into the *CCIT* database, in seconds. Line is the average; shaded area represents a standard deviation on either side of the average. Data aggregated hourly.

Space coverage

The map view shows the number of total successfully map-matched points per 1600 meters for the *entire* selected date range.

Map
Satellite
San Francisco Bay
San Lorenzo
Hayward Executive Airport
Hayward
Fairview
238
Garin/Dry Creek Pioneer Regional Parks
Pleasanton Ridge Regional Park
Pleasanton
Valley Ave
Stanley Blvd

Figure 3-12: Total Transmission Delay Metric



Analysis data selection

1. Choose a feed for analysis: Probe Feed 1
2. Pick a study site: 179: I-880 NB PIF FC
3. Date range start: Mar 2 2012 12AM 00 end: Mar 10 2012 12AM 00
4. Day of week □ ☑ S ☑ M ☑ T ☑ W ☑ T ☑ F ☑ S
5. Aggregation period: 1 hour
6. Data & Metrics

DATA TYPE: ○ Map-matched data ● Raw data

METRIC:
○ Time coverage ○ Sampling rate
○ Unique devices
● Total transmission delay
○ Provider transmission delay

7. Additional options
☑ Use log scale : 0 values have been trimmed to 0.1 to make this possible

Total transmission delay - download displayed data

2012/03/02 22:00:
Transmission delay:100.02
RFP requirement:300

The amount of time that elapses between the device recording its location and the corresponding record … seconds. Line is the average; shaded area represents a standard dev… …gregated hourly.

…f total successfully map-matched points per 1600 meters for the *entire*

Average number of seconds that elapsed for devices that were detected between 10pm and 11pm on March 2, 2012.

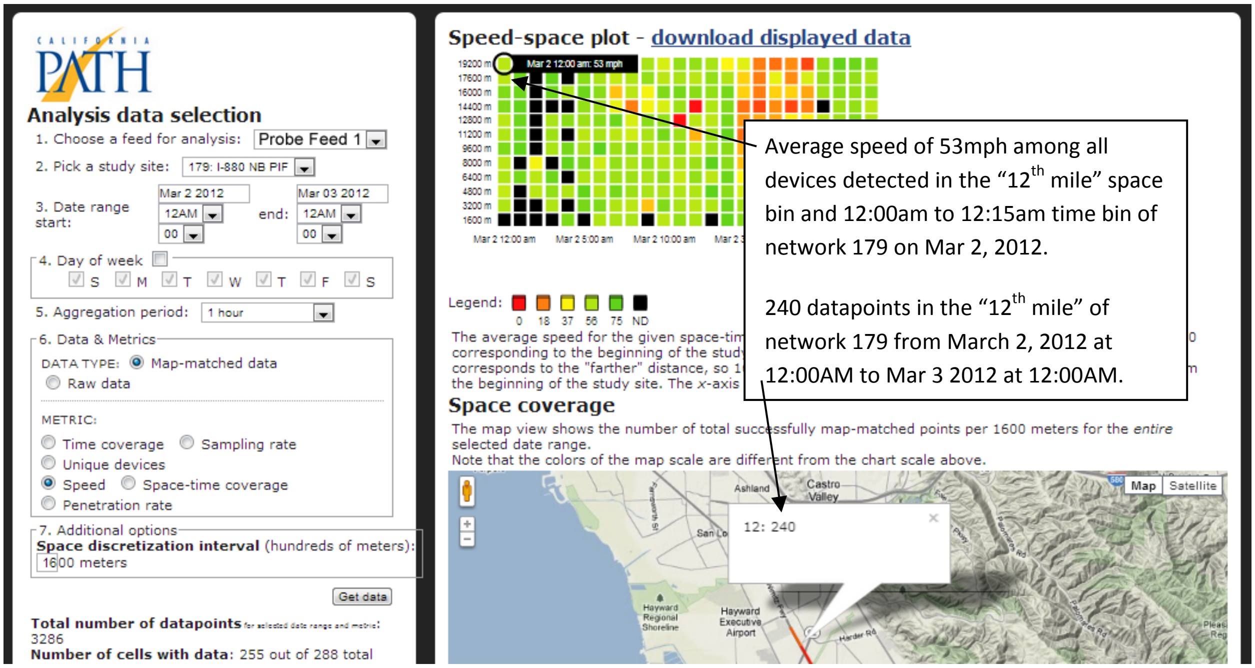
Additional option to display the log scale for a better visual of transmission delay graph.

Figure 3-13: Transmission Delay (Using log scale option)

Total Transmission Delay represents the average amount of time that elapsed (in seconds) between the time devices were detected and the time the detection was inserted into the **PATH** database. The green lines in Figure 3-12 and Figure 3-13 represent the average amount of time elapsed for devices detected between 10pm and 11pm on Mar 2, 2012 before the detection was inserted into the **PATH** database. Figure 3-13 is the same representation of Figure 3-12, but using the "Use log scale" option in Field 7 'Addition option' in order to display the graph more clearly.

The Total Transmission Delay metric can only be viewed using raw data.

3.6 SPEED



Analysis data selection

1. Choose a feed for analysis: Probe Feed 1
2. Pick a study site: 179: I-880 NB PIF
3. Date range start: Mar 2 2012 12AM 00 end: Mar 03 2012 12AM 00
4. Day of week □ ☑ S ☑ M ☑ T ☑ W ☑ T ☑ F ☑ S
5. Aggregation period: 1 hour
6. Data & Metrics
   DATA TYPE: ● Map-matched data ○ Raw data
   METRIC: ○ Time coverage ○ Sampling rate ○ Unique devices ● Speed ○ Space-time coverage ○ Penetration rate
7. Additional options
   Space discretization interval (hundreds of meters): 1600 meters

Total number of datapoints for selected date range and metric: 3286
Number of cells with data: 255 out of 288 total

Speed-space plot - download displayed data

Mar 2 12:00 am: 53 mph

Legend: 🟥 0 🟧 18 🟨 37 🟩 56 🟩 75 ⬛ ND

The average speed for the given space-tim… corresponding to the beginning of the stud… corresponds to the "farther" distance, so 1… the beginning of the study site. The *x*-axis…

Space coverage

The map view shows the number of total successfully map-matched points per 1600 meters for the *entire* selected date range.
Note that the colors of the map scale are different from the chart scale above.

12: 240

Average speed of 53mph among all devices detected in the “12th mile” space bin and 12:00am to 12:15am time bin of network 179 on Mar 2, 2012.

240 datapoints in the “12th mile” of network 179 from March 2, 2012 at 12:00AM to Mar 3 2012 at 12:00AM.

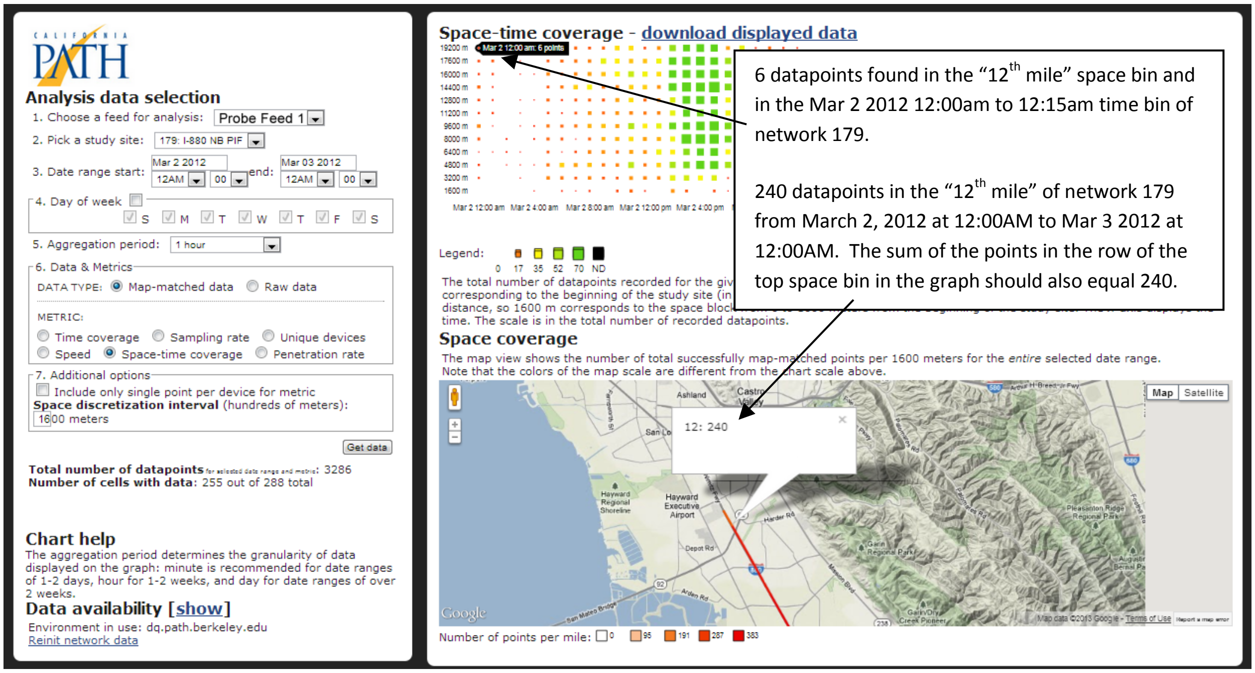
Figure 3-14: Speed Metric

The Speed metric takes the average speed of all datapoints within a given space-time bin. Clicking on the top of the highlighted network of the map in Figure 3-14 shows that 240 devices were detected in the “12th mile” of network 179 from 12AM Mar 2 2012 to 12AM on Mar 3 2012. The graph represents the “12 mile” (12th space bin that approximates 1 mile) in the 12AM to 12:15AM time bin in the top-left block. Placing your mouse over this top-left block shows an average of 53 miles per hour for the devices that were detected in this space-time bin.

Note that it is recommended that at least seven space bins appear for the Speed metric; otherwise, the distinction between the space bins may be difficult to visualize. Also, users should limit the length of the x-axis so that fewer than 1440 time bins appear on the x-axis. Otherwise, rendering the graph can take more than several minutes. For example, generating graphs for this metric with an Aggregation Period of 1 minute and a date range of 24 hours can take approximately 2 minutes to render.

The Speed metric can only be viewed using map-matched data.

3.7 SPACE-TIME COVERAGE



Analysis data selection

1. Choose a feed for analysis: Probe Feed 1
2. Pick a study site: 179: I-880 NB PIF
3. Date range start: Mar 2 2012 12AM 00 end: Mar 03 2012 12AM 00
4. Day of week □ ☑ S ☑ M ☑ T ☑ W ☑ T ☑ F ☑ S
5. Aggregation period: 1 hour
6. Data & Metrics
   DATA TYPE: ● Map-matched data ○ Raw data
   METRIC: ○ Time coverage ○ Sampling rate ○ Unique devices ○ Speed ● Space-time coverage ○ Penetration rate
7. Additional options
   □ Include only single point per device for metric
   Space discretization interval (hundreds of meters): 1600 meters

Total number of datapoints for selected date range and metric: 3286
Number of cells with data: 255 out of 288 total

Chart help
The aggregation period determines the granularity of data displayed on the graph: minute is recommended for date ranges of 1-2 days, hour for 1-2 weeks, and day for date ranges of over 2 weeks.

Data availability [show]
Environment in use: dq.path.berkeley.edu
Reinit network data

Space-time coverage - download displayed data

The total number of datapoints recorded for the giv
corresponding to the beginning of the study site (in
distance, so 1600 m corresponds to the space block
time. The scale is in the total number of recorded datapoints.

Space coverage
The map view shows the number of total successfully map-matched points per 1600 meters for the *entire* selected date range. Note that the colors of the map scale are different from the chart scale above.

6 datapoints found in the "12th mile" space bin and in the Mar 2 2012 12:00am to 12:15am time bin of network 179.

240 datapoints in the "12th mile" of network 179 from March 2, 2012 at 12:00AM to Mar 3 2012 at 12:00AM. The sum of the points in the row of the top space bin in the graph should also equal 240.

Figure 3-15: Space-time Coverage Metric

The Space-time Coverage metric counts the number of datapoints in each space-time bin. Clicking on the top of the highlighted portion of the map in Figure 3-15 shows that 240 datapoints were detected in the "12th mile" of network 179 from 12AM Mar 2 2012 to 12AM on Mar 3 2012. The point selected in the top-left of the graph represents the "12th mile" (12th space bin of network 179 that approximates 1 mile) in the 12AM to 12:15AM time bin. Placing your mouse over this top-left point shows that a total of 6 datapoints were detected in this space-time bin. The sum of the top row in the graph should be equal to the total shown in the map.

Note that it is recommended that at least seven space bins appear for the Space-time Coverage metric, otherwise, the distinction between the space bins may be difficult to visualize. In addition, it is recommended to limit the length of the x-axis so that fewer than 1440 time bins appear on the x-axis. Otherwise, rendering the graph can take more than several minutes. For example, generating graphs for this metric with Aggregation Period of 1 minute and a date range of 24 hours (60 minutes * 24 hours = 1440 1-minute time bins) can take approximately 2 minutes.

The Space-time Coverage metric can only be viewed using map-matched data.

Chapter 4

Path Inference Filter: Map Matching of Probe Vehicle Data

In Mobile Century [27], a controlled experiment which demonstrated that GPS-equipped cell phones could be used for traffic monitoring, there was no question where the cars were located. Probe vehicles with identical GPS devices were sent along designated routes with known origins and destinations over a specific time span.

With commercially available data, such as that procured through the *Pilot Procurement of Third-Party Traffic Data* (Task Order 1), that is not the case. Although the vendors responded to our RFP in good faith, the supplied data comes from multiple sources (truck fleets, taxis, delivery services, etc.) equipped with various types of GPS devices that report vehicle speed and location with different levels of precision. Therefore, before we could use the data to develop reliable estimates of velocities and travel times along the highway, it was critical first to accurately map the probe vehicles to the road network. For example, we needed to develop the capability to distinguish between a vehicle traveling slowly on the highway from a vehicle traveling on a parallel arterial or access road.

To accomplish this vital step, we employed a class of algorithms called the *path inference filter* (PIF). Originally deployed as part of the Mobile Millennium project [28], we applied PIF to the probe data acquired for the Pilot Procurement. This chapter details the design, components, and mathematical processes of the path inference filter.

1 INTRODUCTION

GPS receivers have enjoyed a widespread use in transportation, and they are rapidly becoming a commodity. They offer unique capabilities for tracking fleets of vehicles (for companies), and routing and navigation (for individuals). These receivers are usually attached to a car or a truck, also called a *probe vehicle*, and they relay information to a base station using the data channels of cellphone networks (3G, 4G). A typical datum provided by a probe vehicle includes an identifier of the vehicle, a (noisy) position, and a timestamp[1]. In addition to these geolocalization attributes, data points contain other attributes such as heading, speed, etc.

The two most important characteristics of GPS data for traffic estimation purposes are:

- the *GPS localization accuracy* (how accurate is the vehicle's reported position)
- the *sampling strategy* used by the probe vehicle

In order to reduce power consumption or transmission costs, probe vehicles do not continuously report their location to the base station. The probe data currently available are generated using a *temporal sampling* strategy, where GPS probes send their position at a fixed rate over time, such as every 30 seconds or every 5 minutes. The critical factor is then the *temporal resolution* of the probe data (how frequently the location is reported). A low temporal resolution (low frequency) carries some uncertainty as to which trajectory was followed. A high temporal resolution (high frequency) gives access to the complete and precise trajectory of the vehicle. However, the device usually consumes more power and communication bandwidth.

GPS observations in cities are noisy [29], and are usually provided at low sampling rates (on the order of one minute) [22]. One common problem that occurs when dealing with GPS traces is the correct mapping of these observations to the road network, and the reconstruction of the trajectory of the vehicle. Even at higher sampling rates and along highways where a vehicle's trajectory is less ambiguous, GPS observations can differ from the mapped roadway.

This chapter presents a class of algorithms, together called the Path Inference Filter (PIF), that solves this problem in a principled and efficient way. The PIF algorithm solves the problems of both map matching and path inference of GPS points in real time, for a variety of trade-offs and scenarios, and with a high throughput. These processes represent a different notion of filtering, where instead of correcting values, the data is actually being translated from one spatial reference system (GPS position on the Earth) to another (link identifiers and position along the link using a network representation of the road).

Filtering in the context of this chapter is understood as the process of cleaning data that is inherently noisy (as opposed to estimation techniques such as Kalman filtering and extensions). A filter for large

[1] The experiments in this chapter use GPS observations only. However, nothing prevents the application of the PIF algorithms to other types of localized data.

scale GPS data like PIF requires both filtering in the more traditional sense of the word (removing outliers, smoothing, etc.) as well as map matching for every data point that arrives. This type of filter is computationally intensive, as performing map matching is a time-consuming process because it relies on a spatial query to the database for each GPS point. PIF has been implemented in Scala and in such a way as to leverage parallel computing technology as more hardware resources become available to our system. Choosing this design strategy means that this computationally intensive filter can scale well when the volume of data substantially increases as is expected in the coming years.

Specific instantiations of this algorithm were deployed as part of the *Mobile Millennium* system, and PIF has been used with all the probe data acquired for the Hybrid Traffic Data Procurement pilot. Figure 4-1 shows a subset of probe data collected by *Mobile Millennium*. Figure 4-2 shows GPS data along highway I-880 before and after PIF processing.

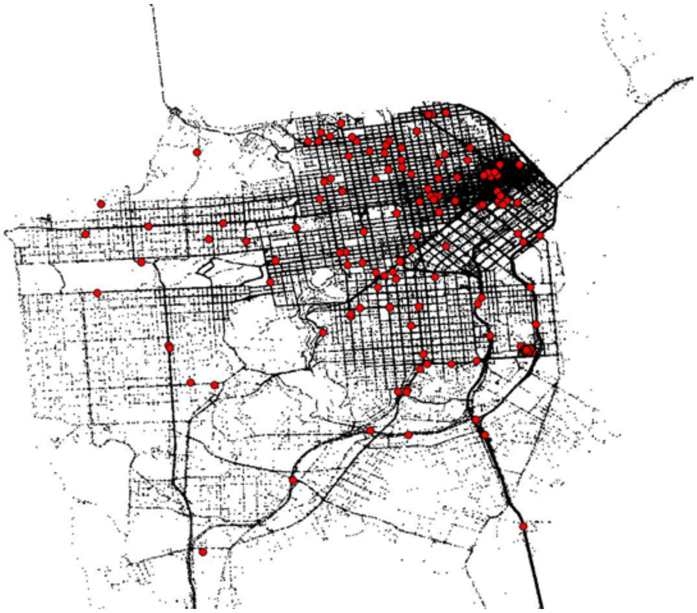


Figure 4-1: An example dataset available to *Mobile Millennium* and processed by the path inference filter: taxicabs in San Francisco from the Cabspotting program [22]. Large circles in red show the position of the taxis at a given time and small dots (in black) show past positions (during the last five hours) of the fleet. The position of each vehicle is observed every minute.

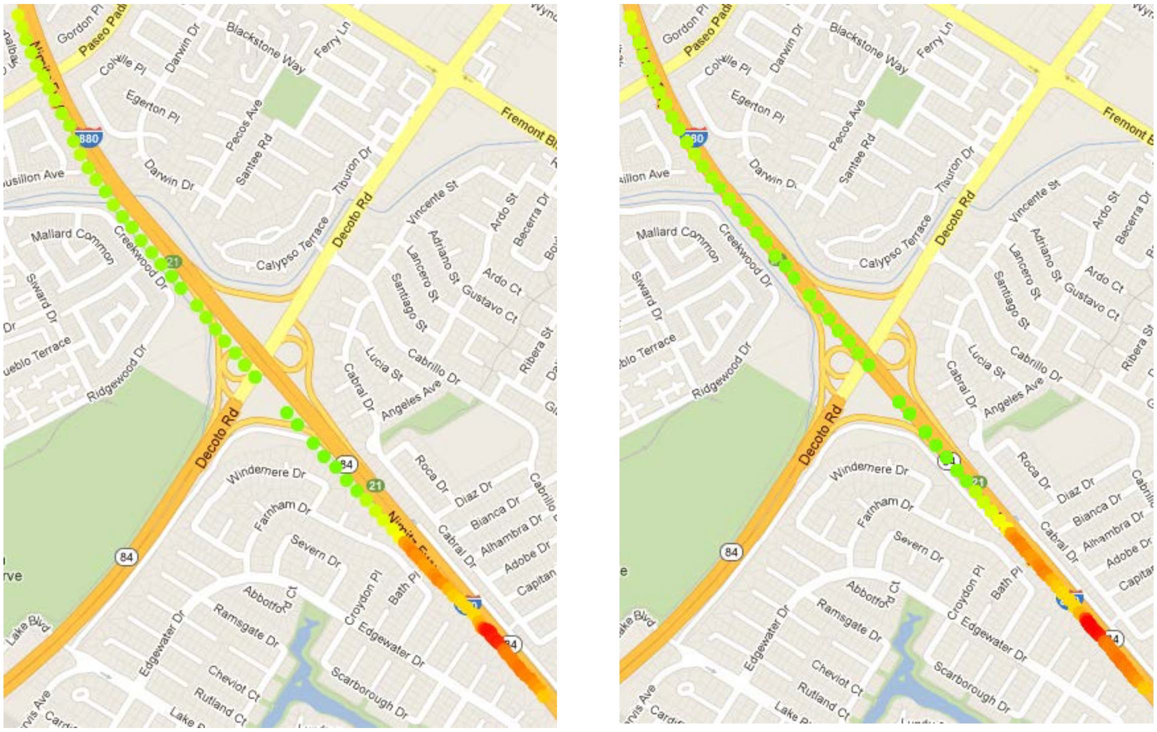


Figure 4-2: Green dots show GPS data points along a section of highway I-880, before (left) and after (right) being processed by the path inference filter to correct GPS location errors and accurately map vehicle locations to the road network.

1.1 SPARSELY SAMPLED GPS

Sparsely sampled probe GPS data refers to the case where probe vehicles send their current GPS location at a fixed frequency, which is not frequent enough to directly measure velocities or link travel times (i.e., sampling period is greater than about 10 seconds). There are several challenges associated with this type of data. First, GPS measurements must be mapped to the road network representation used by the traffic information system, which means that the correct position on the road as well as the path in between successive measurements must be determined. This process is known as map matching and path inference, which is described in more detail in the following sections. Second, probe vehicles can often travel multiple links between measurements when the sampling frequency is low, which means that one must infer what the likely travel times on each link of the path were.

Sparsely-sampled probe GPS data is currently the most ubiquitous data source on the arterial network. This data clearly has some privacy issues as it is possible to track the general path of the vehicle. However, the majority of this data today comes from fleets of various sorts (such as UPS, FedEx, taxis, etc.). Most of this data is privately held among several companies, but between all sources there are millions of records per day in many major urban markets.

1.2 HIGH-FREQUENCY GPS

High-frequency probe GPS data refers to the case where probe vehicles send their current GPS location relatively often (at a frequency greater than 0.5 Hz). This kind of data is generally the most accurate kind of vehicle trajectory data possible, especially when sampling every second with a high-quality GPS chip. From this data, one can directly infer velocities and short distance travel times. The issue of map matching is still present as there can be ambiguity around intersections, but the path is usually easy to determine when examining the entire trace. This data can provide a high level of detail (due to the high frequency); however, the percentage of vehicles that are being tracked is usually relatively low. Sampling a vehicle's position every few seconds is clearly very privacy invasive and it also comes with large communication costs to send the high volume of data. For these reasons, it is not common to receive this data with any kind of regularity. This data is often collected for specific experimental studies, but is not generally available for real-time traffic information systems.

GPS tracking data (both sparse and high-frequency) is not intended to preserve the privacy of the driver. Our system collects this data specifically from sources who have agreed to provide it in that form and are not concerned with privacy of the drivers (the primary function of the data is generally to track service vehicles). This is generally restricted to fleet delivery vehicles or taxis, but if an individual driver wanted to participate in this manner, the data can be collected in that form.

Fixed-location sensors also require map matching, although the task is generally much easier than for GPS data. For these types of sensors, a spatial database is again required to identify the closest links to the GPS location of the sensor (which is generally how fixed-location sensors are identified). The GPS location often comes with a description of the location as text, and this text is used in the case where the GPS location is close to several possible links. In that situation, the text acts as a discriminator for choosing the correct mapping.

1.3 SHORTCOMINGS OF CURRENT APPROACHES

In the case of a high temporal resolution (typically, a frequency greater than an observation per second), some highly successful methods have been developed for continuous estimation [30] [31] [32]. However, most data collected at large scale today is generated by commercial fleet vehicles. It is primarily used for tracking the vehicles and usually has a low temporal resolution (1 to 2 minutes) [33] [7] [34] [22]. In the span of a minute, a vehicle in a city can cover several blocks. Therefore, information on the precise path followed by the vehicle is lost. Furthermore, due to GPS localization errors, recovering the location of a vehicle that just sent an observation is a non-trivial task: There are usually several streets that could be compatible with any given GPS observation. Simple deterministic algorithms to reconstruct trajectories fail due to misprojection (Figure 4-3) or shortcuts (Figure 4-4). Such shortcomings have motivated our search for a principled approach that jointly considers the mapping of observations to the network and the reconstruction of the trajectory.

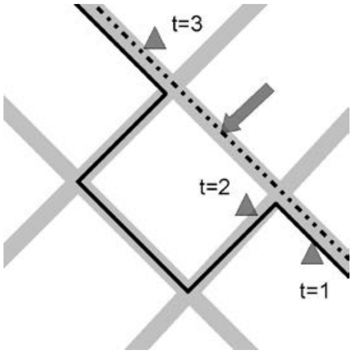


Figure 4-3: Example of failure when using an intuitive algorithm projects each GPS measurement to the closest link. The raw GPS measurements are the triangles, the actual true trajectory is the dashed line, and the reconstructed trajectory is the continuous line. Due to noise in the observation, the point at $t = 2$ is closer to the orthogonal road and forces the algorithm to add a left turn, while the vehicle is actually going straight. This problem is frequently observed for GPS data in cities. The *path inference filter* provides one solution to this problem.

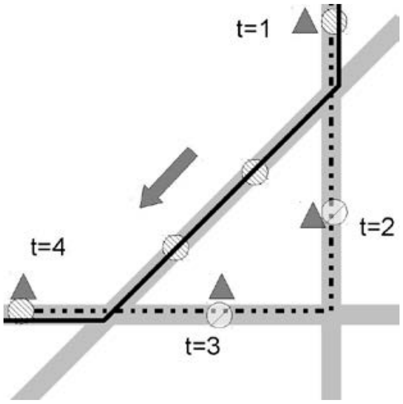


Figure 4-4: Example of failure when trying to minimize the path length between a sequence of points. The raw observations are the triangles, the actual true trajectory is the dashed line, and the reconstructed trajectory is the continuous line. The circles are possible locations of the vehicle corresponding to the observations. The hashed circles are the states chosen by this reconstruction algorithm. Due to GPS errors that induce problems explained in Figure 4-3, we must consider point projections on all links within a certain distance from the observed GPS points. However, the path computed by a shortest path algorithm may not correspond to the true trajectory. Note how, for $t = 2$ and $t = 3$, the wrong link and the wrong states are elected to reconstruct the trajectory.

The problem of mapping data points onto a map can be traced back to 1980 [35]. Researchers started systematic studies after the introduction of the GPS system to civilian applications in the 1990s [36]. These early approaches followed a *geometric* perspective, associating each observation datum to some point in the network [37]. Later, this projection technique was refined to use more information such as heading and road curvature. This greedy matching, however, leads to poor trajectory reconstruction since it does not consider the path leading up to a point [38]. New deterministic algorithms emerged to directly match partial trajectories to the road by using the topology of the network [39] and topological metrics based on the Fréchet distance [40] [41]. These deterministic algorithms cannot readily cope with ambiguous observations [31], and were soon expanded into probabilistic frameworks. A number of implementations were explored: particle filters [42] [43], Kalman filters [44], Hidden Markov Models [45], and less mainstream approaches based on Fuzzy Logic and Belief Theory.

1.4 MAP MATCHING AND PATH INFERENCE

Two types of information are missing in a sequence of GPS readings: the exact location of the vehicle on the road network when the observation was emitted, and the path followed from the previous location to the new location. These problems are correlated. The aforementioned approaches focus on high-frequency sampling observations, for which the path followed is extremely short (less than a few hundred meters, with very few intersections). In this context, there is usually a dominant path that starts from a well-defined point, and Bayesian filters accurately reconstruct paths from observations [44] [30] [43]. When sampling rates are lower and observed points are further apart, however, a large number of paths are possible between two points. Researchers have recently focused on efficiently identifying these correct paths and have separated the joint problem of finding the paths and finding the projections into two distinct problems. The first problem is path identification and the second step is projection matching [46] [45] [38] [47] [48]. Some interesting trajectories mixing points and paths that use a voting scheme have also recently been proposed [38]. Our filter aims at solving the two problems at the same time, by considering a single unified notion of *trajectory*.

The *path inference filter* is a probabilistic framework that aims at recovering trajectories and road positions from low-frequency probe data in real time, and in a computationally efficient manner. As will be shown, the performance of the filter degrades gracefully as the sampling frequency decreases, and it can be tuned to different scenarios (such as real time estimation with limited computing power or offline, high accuracy estimation).

The filter is justified from the Bayesian perspective of the noisy channel and falls into the general class of *Conditional Random Fields* [49]. Our framework can be decomposed into the following steps:

- *Map matching*: each GPS measurement from the input is projected onto a set of possible candidate states on the road network.
- *Path discovery:* admissible paths are computed between pairs of candidate points on the road network.
- *Filtering*: probabilities are assigned to the paths and the points using both a stochastic model for the vehicle dynamics and probabilistic driver preferences learned from data.

According to the very exhaustive review by Quddus et al. [50], most map-matching approaches fall into one of the four categories:

1. "Geometric" methods, which pick the closest matching point. The distance metric itself is the subject of variations by different authors.

2. "Weighted topological" methods, which use connectivity information between links and various ways to weight the different paths.

3. "Probabilistic" methods, which combine variance information about the points and topological information about the paths in a simple way.

4. "Advanced" methods, which encompass everything more complicated: Kalman Filtering, Particle Filtering, Belief Theory [51] and Fuzzy Logic [52].[2]

The path inference filter presents a number of compelling advantages over the work found in the current literature:

1. The approach presents a general framework grounded in established statistical theory that encompasses, as special cases, most techniques presented as "geometric", "topological" or "probabilistic". In particular, it combines information about paths, points and network topology in a single unified notion of *trajectory.*

2. Nearly all work on Map Matching is segmented into (and presents results for) either high-frequency or low-frequency sampling. The path inference filter performs as well as the current state-of-the-art approaches for sampling rates less than 30 seconds, and improves upon the state of the art [46] [38] by a factor of more than 10% for sampling intervals greater than 60 seconds[3]. We also analyze failure cases and we show that the output provided by the path inference filter is always "close" to the true output for some metric.

3. As will be seen in Section 3, most existing approaches (which are based on Dynamic Bayesian Networks) do not work well at lower frequencies due to the *selection bias problem*. Our work directly addresses this problem by performing inference on a Random Field.

4. The path inference filter can be used with complex path models such as those used in [45] and [47]. In the present study, we restrict ourselves to a class of models (the exponential family distributions) that is rich enough to provide insight on the driving patterns of the vehicles.

[2] Note that "probabilistic" models, as well as most of the "advanced" models (Kalman Filtering, Particle Filtering, Hidden Markov Models) fall under the general umbrella of *Dynamic Bayesian Filters*, presented in great detail in [30]. As such, they deserve a common theoretical treatment, and in particular all suffer from the same pitfalls detailed in Section 3.

[3] Performance comparisons are complicated by the lack of a single agreed-upon benchmark dataset. Nevertheless, the city we study is complex enough to compare favorably with cities studied with other works.

Furthermore, when using this class of models, the learning of new parameters leads to a convex problem formulation that is fast to solve. These parameters can be learned using standard Machine Learning algorithms, even when no ground truth is available.

5. With careful engineering, it is possible to achieve high throughput on large-scale networks. Our reference implementation achieves an average throughput of hundreds of GPS observations per second on a single core in real time. Furthermore, the algorithm scales well on multiple cores and has achieved average throughput of several thousands of points per second on a multicore architecture.

Algorithms often need to trade off accuracy for timeliness, and are considered either “local” (greedy) or “global” (accumulating some number of points before returning an answer) [38]. The path inference filter is designed to work across the full spectrum of accuracy versus latency. As we will show, we can still achieve good accuracy by delaying computations by only one or two time steps.

2 PATH DISCOVERY

The road network is described as a directed graph $\mathcal{N} = (\mathcal{V}, \mathcal{E})$ in which the nodes are the street intersections and the edges are the streets, referred to in the text as the *links* of the road network. Each link is endowed with a number of physical attributes (speed limit, number of lanes, type of road, etc.). Given a link of the road network, the links into which a vehicle can travel will be called *outgoing links*, and the links from which it can come will be called the *incoming links.* Every location on the road network is completely specified by a given link l and offset o on this link. The offset is a non-negative real number bounded by the length of the corresponding link, and represents the position on the link. At any time, the *state* x of a vehicle consists of its location on the road network and some other optional information such as speed, or heading. For our example we consider that the state is simply the location on one of the road links (which are directed). Additional information such as speed, heading, lane, etc. can easily be incorporated into the state:

$$x = (l, o)$$

Furthermore, for the remainder of this chapter we consider trajectory inference for a single probe vehicle.

2.1 FROM GPS POINTS TO DISCRETE VEHICLE STATES

The points are mapped to the road following a Bayesian formulation. Consider a GPS observation g. We study the problem of mapping it to the road network according to our knowledge of how this observation was generated. This generation process is represented by a probability distribution $\omega(g|x)$ that, given a state x, returns a probability distribution over all possible GPS observations g. Such distributions ω will be described in Section 3.4. Additionally, we may have some *prior knowledge* over the state of the vehicle. For example, some links may be visited more often than others, and some positions on links may be more frequent, such as when vehicles accumulate at the intersections. This knowledge can be encoded in a *prior distribution* $\Omega(x)$. Under this general setting, the state of a vehicle, given a GPS observation, can be computed using Bayes' rule:

$$\pi(x|g) \propto \omega(g|x)\Omega(x)$$

The letter π will define general probabilities, and their dependency on variables will always be included. This probability distribution is defined up to a scaling factor in order to integrate to 1. This posterior distribution is usually complicated, owing to the mixed nature of the state. The state space is the product of a discrete space over the links and a continuous space over the link offsets. Instead of representing it in closed form, some sampled values are considered: for each link l_i, a finite number of states from this link are elected to represent the posterior distribution of the states on this link $\pi(o|g, l = l_i)$. A first way of accomplishing this task is to grid the state space of each link, as illustrated in Figure 4-5.

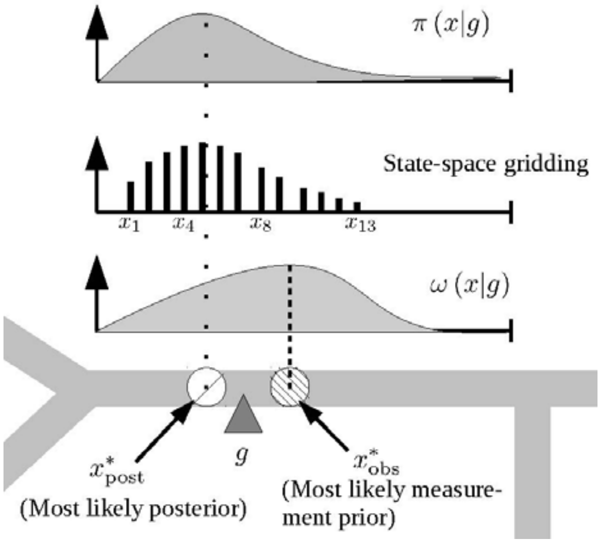


Figure 4-5: Example of a measurement g on a link and two strategies to associate state projections to that measurement on a particular link (gridding and most likely location). The GPS measurement is the triangle denoted g. For this particular measurement, the observation distribution $\omega(x|g)$ and the posterior distribution $\pi(x|g)$ are also represented. When gridding, we select a number of states $x_1, \cdots x_I$ spanning each link at regular intervals. This allows us to use the posterior distribution and have a more precise distribution over the location of the vehicle. However, it is more expensive to compute. Another strategy is to consider a single point at the most probable offset x^*_{post} according to the posterior distribution $\pi(x|g)$. However, this location depends on the prior, which is usually not available at this stage (since the prior depends on the location of past and future points, for which do not also know the location). A simple approximation is to consider the most likely point x^*_{obs}according to the observation distribution.

This strategy is robust against the observation errors described in Section 2.2, but it introduces a large number of states to consider. Furthermore, when new GPS values are observed every minute, the vehicle can move quite extensively between updates. The grid step is usually small compared to the distance traveled. Instead of defining a coarse grid over each link, another approach is to use some *most likely state* on each link. Since our state is the pair of a link and an offset on this link, this corresponds to selecting the most likely offset on each state:

$$\forall l_i, \quad o^*_{i_{\text{posterior}}} = \underset{o}{\operatorname{argmax}}\, \pi(o|g, l = l_i)$$

In practice, the probability distribution $\pi(x|g)$ decays rapidly, and can be considered overwhelmingly small beyond a certain distance from the observation g. Links located beyond a certain radius need not be considered valid projection links, and may be discarded.

In the rest of this chapter, the boldface symbol $\boldsymbol{x}$ will denote a (finite) collection of states associated with a GPS observation g that we will use to represent the posterior distribution $\pi(x|g)$, and the integer

I will denote its cardinality: $\boldsymbol{x} = (x_i)_{1:I}$. These points are called *candidate state projections for the GPS measurement g.* These discrete points will then be linked together through trajectory information that takes into account the trajectory and the dynamics of the vehicle. We now mention a few important points for a practical implementation.

The prior distribution. A Bayesian formulation requires that we endow the state x with a prior distribution $\Omega(x)$ that expresses our knowledge about the distribution of points on a link. When no such information is available, since the offset is continuous and bounded on a segment, a natural non-informative prior is the uniform distribution over offsets: $\Omega \sim U([0, \text{length}(l)])$. In this case, maximizing the posterior is equivalent to maximizing the conditional distribution from the generative model:

$$\forall l_i, o^*_{i_{\text{observation}}} = \underset{o}{\operatorname{argmax}}\, \omega\big(g | x = (o, l_i)\big)$$

Having mapped GPS points into discrete points on the road network, we now turn our attention to connecting these points by paths in order to form trajectories.

2.2 FROM DISCRETE VEHICLE STATES TO TRAJECTORIES

At each time step t, a GPS point g^t (originating from a single vehicle) is observed. This GPS point is then mapped onto I^t different candidate states denoted $\boldsymbol{x}^t = x_1^t \cdots x_{I^t}^t$. Because this set of projections is finite, there is only a (small) finite number J^t of paths that a vehicle can have taken while moving from some state $x_i^t \in \boldsymbol{x}^t$ to some state $x_{i'}^{t+1} \in \boldsymbol{x}^{t+1}$. We denote the set of *candidate paths* between the observation g^t and the next observation g^{t+1} by $\boldsymbol{p}^t$:

$$\boldsymbol{p}^t = \left(p_j^t\right)_{j=1:J^t}$$

Each path p_j^t goes from one of the projection states x_i^t of g^t to a projection state $x_{i'}^{t+1}$ of g^{t+1}. There may be multiple pairs of states to consider, and between each pair of states, there are typically several paths available (see Figure 4-6). Lastly, a *trajectory* is defined by the succession of states and paths, starting and ending with a state:

$$\tau = x_1 p_1 x_2 \cdots p_{t-1} x_t$$

where x_1 is one element of $\boldsymbol{x}^1$, p_1 of $\boldsymbol{p}^1$, and so on.

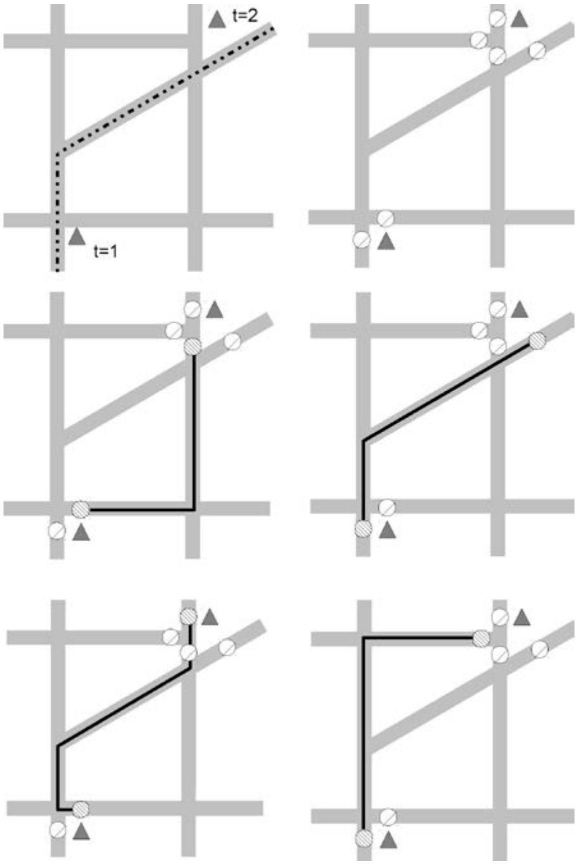


Figure 4-6: Example of path exploration between two observations. The true trajectory and two associated GPS observations are shown on the upper left corner. The upper right corner figure shows the set of candidate projections associated with each observation. A path discovery algorithm computes every acceptable path between between each pair of candidate projections. The four figures at the bottom show a few examples of such computed paths.

Due to speed limits leading to lower bounds on achievable travel times on the network, there is only a finite number of paths a vehicle can take during a time interval Δt. Such paths can be computed using standard graph search algorithms. The depth of the search is bounded by the maximum distance a vehicle can travel on the network at a speed $v_{\max}$ within the time interval between each observation. An algorithm that performs well in practice is the A* algorithm [53], a common graph search algorithm that makes use of a heuristic to guide its search. The cost metric we use here is the expected travel time on each link, and the heuristic is the shortest geographical distance, properly scaled so that it is an admissible heuristic.

The case of backward paths. It is convenient and realistic to assume that a vehicle always drives *forward*, i.e., in the same direction of a link[4]. In our notation, a vehicle enters a link at offset 0, drives along the link following a non-decreasing offset, and exits the link when the offset value reaches the total length of the link. However, due to GPS noise, the most likely state projection of a vehicle waiting at a red light may appear to go backward, as shown in Figure 4-7. This leads to incorrect transitions if we assume that paths only go forward on a link.

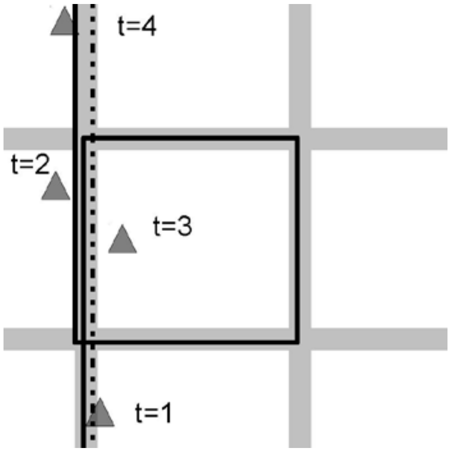


Figure 4-7: Example of failure when observing strict physical consistency: due to the observation noise, the observation (3) appears physically behind (2) on the same link. Without considering backward paths, the most plausible explanation is that the vehicle performed a complete loop around the neighboring block.

Three approaches to solve this issue are discussed, depending on the application:

1. It is possible to keep a single state for each link (the most likely) and explore some backward paths. These paths are assumed to go backward because of observation noise. This solution provides *connected states at the expense of physical consistency*: all the measurements are correctly mapped to their most likely location, but the trajectories themselves are not physically acceptable. This is useful for applications that do not require connectedness between pairs of states, for example when computing a distribution of the density of probe data per link.

2. It is also possible to disallow backward paths and consider multiple states per link, such as a grid over the state space. A vehicle never goes backward, and in this case the filter can generally account for the vehicle not moving by associating the same state to successive observations. All the trajectories are physically consistent and the posterior state density is the same as the probability density of the most likely states, but is more burdensome from a computational perspective (the number of paths to consider grows quadratically with the number of states).

[4] Reverse driving is in some cases even illegal. For example, the laws of Glendale, Arizona, prohibit reverse driving.

3. Finally it is possible to disallow backward paths and use a sparse number of states. The path connectivity issue is solved using some heuristics. Our implementation creates a new state projection on a link l using the following approach:

Given a new observation g, and its most likely state projection $x^* = (l, o^*)$:

1. If no projection for the link l was found at the previous time step, return x^*
2. If such a projection $x_{\text{before}} = (l, o_{\text{before}})$ existed, return $x = (l, \max{(o_{\text{before}}, o^*)})$

With this heuristic, all the points will be well connected, but the density of the states will not be the same as the density of the most likely reconstructed states.

In summary, the first solution is better for density estimations and the third approach works better for travel time estimations. The second option is currently only used for high-frequency offline filtering, for which paths are short, and for which more expensive computations is an acceptable cost.

Handling errors. Maps may contains some inaccuracies, and may not cover all the possible driving patterns. Two errors were found to have a serious effect on the performance of the filter:

- Out of network driving: This usually occurs in parking lots or commercials driveways.
- Topological errors: Some links may be missing on the base map, or one-way streets may have changed to two-way streets. These situations are handled by running *flow analysis* on the trajectory graph. For every new incoming GPS point, after computing the paths and states, it is checked if any candidate position of the first point of the trajectory is reachable from any reachable candidate position on the latest incoming point, or equivalently if the trajectory graph has a positive flow. The set of state projections of an observation may end up being disconnected from the start point even if at every step, there exists a set of paths between each points. In this situation, the probability model will return a probability of 0 (non-physical trajectories) for any trajectory. If a point becomes unreachable from the start point, the trajectory is broken, and restarted again from this point. Trajectory breaks were few (less than a dozen for our dataset), and a visual inspection showed that the vehicle was not following the topology of the network and instead made U-turns or breached through one-way streets.

3 DISCRETE FILTERING USING A CONDITIONAL RANDOM FIELD

In the previous section, we reduced the trajectory reconstruction problem to a discrete selection problem between sets of candidate projection points, interleaved with sets of candidate paths. A probabilistic framework can now be applied to infer a reconstructed trajectory τ or probability distributions over candidate candidate states and candidate paths. Without further assumptions, one would have to enumerate and compute probabilities for every possible trajectory. This is not possible for long sequences of observations, as the number of possible trajectories grows exponentially with the number of observations chained together. By assuming additional independence relations, we turn this intractable inference problem into a tractable one.

3.1 CONDITIONAL RANDOM FIELDS TO WEIGHT TRAJECTORIES

The observation model provides the joint distribution of a state on the road network given an observation. We have described the *noisy generative model* for the observations in Section 2.1. Assuming that the vehicle is at a point x, a GPS observation g will be observed according to a model ω that describes a noisy observation channel. The value of g only depends on the state of the vehicle, i.e. the model reads $\omega(g|x)$. For every time step t, assuming that the vehicle is at the location x^t, a GPS observation g^t is created according to the distribution $\omega(g^t|x^t)$.

Additionally, we endow the set of all possible paths on the road network with a probability distribution. The *transition model* η describes the preference of a driver for a particular path. In probabilistic terms, it provides a distribution $\eta(p)$ defined over all possible paths p across the road network. This distribution is not a distribution over actually observed paths as much as a model of the *preferences* of the driver when given the choice between several options.

We introduce the following *Markov assumptions*.

- Given a start state x_{start} and an end state x_{end}, the path p followed by the vehicle between these two points will only depend on the start state, the end state and the path itself. In particular, it will not depend on previous paths or future paths.
- Consider a state x followed by a path p_{next} and preceded by a path p_{previous}, and associated to a GPS measurement g. Then the paths taken by the vehicle are independent from the GPS measurement g if the state x is known. In other words, the GPS measurement does not add subsequent information given the knowledge of the state of the vehicle.

Since a state is composed of an offset and a link, a path is completely determined by a start state, an end state and a list of links in between. Conditional on the start state and end state, the number of paths between these points is finite (it is the number of link paths that join the start link and the end link).

Not every path is compatible with given start point and end point: the path must start at the start state and must end at the end state. We formally express the compatibility between a state x and the start state of a path p with the compatibility function $\underline{\delta}$:

$$\underline{\delta}(x,p) = \begin{cases} 1 & \text{if the path } p \text{ starts at point } x \\ 0 & \text{otherwise} \end{cases}$$

Similarly, we introduce the compatibility function $\bar{\delta}$ to express the agreement between a state and the end state of a path:

$$\bar{\delta}(p,x) = \begin{cases} 1 & \text{if the path } p \text{ ends at point } x \\ 0 & \text{otherwise} \end{cases}$$

Given a sequence of observations $g^{1:T} = g^1 \cdots g^T$ and an associated trajectory $\tau = x^1 p^1 \cdots x^T$, we define the *unnormalized score,* or *potential*, of the trajectory as:

$$\phi(\tau|g^{1:T}) = \left[\prod_{t=1}^{T-1} \omega\,(g^t|x^t)\underline{\delta}(x^t,p^t)\eta(p^t)\bar{\delta}(p^t,x^{t+1})\right] \cdot \omega(g^T|x^T)$$

The non-negative function ϕ is called the *potential function.* A trajectory τ is said to be a *compatible trajectory with the observation sequence* $g^{1:T}$ if $\phi(\tau|g^{1:T}) > 0$. When properly scaled, the potential ϕ defines a probability distribution over all possible trajectories, given a sequence of observations:

$$\pi(\tau|g^{1:T}) = \frac{\phi(\tau|g^{1:T})}{Z}$$

The variable Z, called the *partition function*, is the sum of the potentials over all the compatible trajectories:

$$Z = \sum_{\tau} \phi\,(\tau|g^{1:T})$$

We have combined the observation model ω and the transition model η into a single potential function ϕ, which defines an unnormalized distribution over all trajectories. Such a probabilistic framework is called a *Conditional Random Field* (CRF) [49]. A CRF is an undirected graphical model which is defined as the unnormalized product of factors over cliques of factors (see Figure 4-8). There can be an exponentially large number of paths, so the partition function cannot be computed by simply summing the value of ϕ over every possible trajectory. As will be seen in Section 4, the value of Z needs to be computed only during the training phase. Furthermore it can be computed efficiently using dynamic programming.

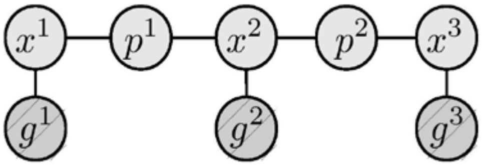


Figure 4-8: Illustration of the Conditional Random Field defined over a trajectory $\tau = x^1 p^1 x^2 p^2 x^3$ and a sequence of observations $g^{1:3}$. The gray nodes indicate the observed values. The solid lines indicate the factors between the variables: $\omega(g^t|x^t)$ between a state x^t and an observation g^t, $\underline{\delta}(x^t, p^t)\eta(p^t)$ between a state x^t and a path p^t and $\bar{\delta}(p^t, x^{t+1})$ between a path p^t and a subsequent state x^{t+1}.

The case against the Hidden Markov Model approach. The classical approach to filtering in the context of trajectories is based on Hidden Markov Models (HMMs), or their generalization, Dynamic Bayesian Networks (DBNs) [54]: a sequence of states and trajectories form a trajectory, and the coupling of trajectories and states is done using transition models $\hat{\pi}(x|p)$ and $\hat{\pi}(p|x)$. See Figure 4-9 for a representation.

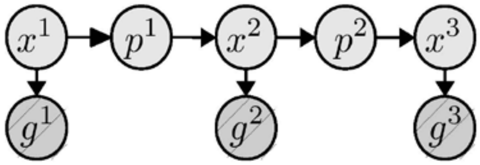


Figure 4-9: A Dynamic Bayesian Network (DBN) commonly used to model the trajectory reconstruction problem. The arrows indicate the directed dependencies of the variables. The GPS observations g^t are generated from states x^t. The unobserved paths p^t are generated from a state x^t, following a transition probability distribution $\hat{\pi}(p|x)$. The transition from a path p^t to a state x^t follows the transition model $\hat{\pi}(x|p)$.

This results in a chain-structured directed probabilistic graphical model in which the path variables p^t are unobserved. Depending on the specifics of the transition models, $\hat{\pi}(x|p)$ and $\hat{\pi}(p|x)$, probabilistic inference has been done with Kalman filters [44] [42], the forward algorithm or the Viterbi algorithm [45] [55], or particle filters [43].

Hidden Markov Model representations, however, suffer from the *selection bias problem*, first noted in the labeling of words sequences [49], which makes them not the best fit for solving path inference problems. Consider the example trajectory $\tau = x^1 p^1 x^2$ observed in our data, represented in Figure 4-10.

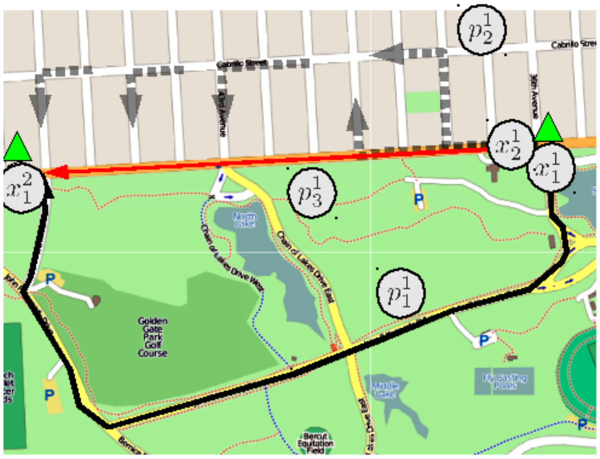


Figure 4-10: Example of a failure case when using a Hidden Markov Model: the solid black path will be favored over all the other paths.

For clarity, we consider only two states x_1^1 and x_2^1 associated with the GPS reading g^1 and a single state x_1^2 associated with g^2. The paths $\left(p_j^1\right)_j$ between x^1 and x^2 may either be the lone path p_1^1 from x_1^1 to x_1^2 that allows a vehicle to cross the Golden Gate Park, or one of the many paths between Cabrillo Street and Fulton Street that go from x_2^1 to x^1, including p_3^1 and p_2^1. In the HMM representation, the transition probabilities must sum to 1 when conditioned on a starting point. Since there is a single path from x_2^1 to x^2, the probability of taking this path from the state x_1^1 will be $\hat{\pi}(p_1^1|x_1^1) = 1$ so the overall probability of this path is $\hat{\pi}(p_1^1|g^1) = \hat{\pi}(x_1^1|g^1)$. Consider now the paths from x_2^1 to x_1^2: a lot of these paths will have a similar weight, since they correspond to different turns and across the lattice of streets. For each path p amongst these N paths of similar weight, Bayes' assumption implies $\hat{\pi}(p|x_2^1) \approx \frac{1}{N}$ so the overall probability of this path is $\hat{\pi}(p|g^1) \approx \frac{1}{N}\hat{\pi}(x_2^1|g^1)$. In this case, N can be large enough that $\hat{\pi}(p_1^1|g^1) \geq \hat{\pi}(p|g^1)$, and the remote path will be selected as the most likely path.

Due to their structures, all HMM models will be biased towards states that have the least expansions. In the case of a road network, this can be pathological. In particular, the HMM assumption will carry the effect of the selection bias as long as there are long disconnected segments of road. This can be particularly troublesome in the case of road networks since HMM models will end up being forced to assign too much weight to a highway (which is highly disconnected) and not enough to the road network alongside the highway. Our model, which is based on Conditional Random Fields, does not have this problem since the renormalization happens just once and is over all paths from start to end, rather than renormalizing for every single state transition independently.

Efficient filtering algorithms. Using the probabilistic framework of the CRF, we wish to infer:

- the most likely trajectory τ:

$$\tau^* = \underset{\tau}{\operatorname{argmax}} \quad \pi(\tau|g^{1:T}) \tag{1}$$

- the posterior distributions over the elements of the trajectory, i.e. the conditional marginals $\pi(x^t|g^{1:T})$ and $\pi(p^t|g^{1:T})$

As will be seen, both elements can be computed without having to obtain the partition function Z. The solution to both problems is a particular case of the *Junction Tree algorithm* [54] and can be computed in time complexity linear in the time horizon by using a dynamic programing formulation. Computing the most likely trajectory is a particular instantiation of a standard dynamic programing algorithm called the *Viterbi algorithm* [56]. Using a classic Machine Learning algorithm for chain-structured junction trees (the *forward-backward algorithm* [57] [58]), all the conditional marginals can be computed in two passes over the variables. In the next section, we detail the justification for the Viterbi algorithm and in Section 3.3 we describe an efficient implementation of the forward-backward algorithm in the context of this application.

3.2 FINDING THE MOST LIKELY PATH

For the rest of this section, we fix a sequence of observations $g^{1:T}$. For each observation g^t, we consider a set of candidate state projections $\boldsymbol{x}^t$. At each time step $t \in [1 \cdots T-1]$, we consider a set of paths $\boldsymbol{p}^t$, so that each path p^t from $\boldsymbol{p}^t$ starts from some state $x^t \in \boldsymbol{x}^t$ and ends at some state $x^{t+1} \in \boldsymbol{x}^{t+1}$. We will consider the set ς of valid trajectories in the Cartesian space defined by these projections and these paths:

$$\varsigma = \left\{ \tau = x^1 p^1 \cdots p^{T-1} x^T \middle| \begin{array}{r} x^t \in \boldsymbol{x}^t \\ p^t \in \boldsymbol{p}^t \\ \underline{\delta}(x^t, p^t) = 1 \\ \overline{\delta}(p^t, x^{t+1}) = 1 \end{array} \right\}$$

In particular, if I^t is the number of candidate states associated with g^t (i.e. the cardinal of $\boldsymbol{x}^t$) and J^t is the number of candidate paths in $\boldsymbol{p}^t$, then there are at most $\prod_1^T I^t \prod_1^{T-1} J^t$ possible trajectories to consider. We will see, however, that most likely trajectory τ^* can be computed in $O(TI^*J^*)$ computations, with $I^* = \max_t I^t$ and $J^* = \max_t J^t$.

The partition function Z does not depend on the current trajectory τ and need not be computed when solving Equation 1:

$$\begin{aligned} \tau^* &= \underset{\tau \in \varsigma}{\operatorname{argmax}} \quad \pi(\tau|g^{1:T}) \\ &= \underset{\tau \in \varsigma}{\operatorname{argmax}} \quad \phi(\tau|g^{1:T}) \end{aligned}$$

Call $\phi^*(g^{1:T})$ the maximum value over all the potentials of the trajectories compatible with the observations $g^{1:T}$:

$$\phi^*(g^{1:T}) = \max_{\tau \in \varsigma} \phi\,(\tau | g^{1:T})$$

The trajectory τ that realizes this maximum value is found by tracing back the computations. For example, some pointers to the intermediate partial trajectories can be stored to trace back the complete trajectory, as done in the referring implementation [59]. This is why we will only consider the computation of this maximum. The function ϕ depends on the probability distributions ω and η, left undefined so far. These distributions will be presented in depth in Sections 3.4 and 3.5.

It is useful to introduce notation related to a *partial trajectory*. Call $\tau^{1:t}$ the *partial trajectory* until time step t:

$$\tau^{1:t} = x^1 p^1 \cdots x^t$$

For a partial trajectory, we define some partial potentials $\phi(\tau^{1:t} | g^{1:t})$ that depend only on the observations seen so far:

$$\phi(\tau^{1:t} | g^{1:t}) = \omega(g^1 | x^1) \prod_{t'=1}^{t-1} \underline{\delta}\left(x^{t'}, p^{t'}\right) \eta\left(p^{t'}\right) \cdot \bar{\delta}\left(p^{t'}, x^{t'+1}\right) \omega\left(g^{t'+1} | x^{t'+1}\right) \tag{2}$$

For each time step t, given a state index $i \in [1, I^t]$ we introduce the potential function for trajectories that end at the state x_i^t:

$$\phi_i^t = \max_{\tau^{1:t} = x^1 p^1 \cdots x^{t-1} p^{t-1} x_i^t} \phi\,(\tau^{1:t} | g^{1:t})$$

One sees:

$$\phi^* = \max_{i \in [1, I^T]} \phi_i^T$$

The partial potentials defined in Equation (2) follow an inductive identity:

$$\phi_i^1 = \omega\left(g^1 | x_i^t\right)$$

$$\forall t, \phi_i^{t+1} = \max_{\substack{i' \in [1, I^t] \\ j \in [1, J^t]}} \left[\phi_{i'}^t \underline{\delta}\left(x_{i'}^t, p_j^t\right) \eta\left(p_j^t\right) \cdot \bar{\delta}\left(p_j^t, x_i^{t+1}\right) \omega\left(g^{t+1} | x_i^{t+1}\right)\right] \tag{3}$$

By using this identity, the maximum potential ϕ^* can be computed efficiently from the partial maximum potentials ϕ_i^t. The computation of the trajectory that realizes this maximum potential ensues by tracing back the computation to find which partial trajectory realized ϕ_i^t for each step t.

3.3 TRAJECTORY FILTERING AND SMOOTHING

We now turn our attention to the problem of computing the marginals of the posterior distributions over the trajectories, i.e. the probability distributions $\pi(x^t|g^{1:T})$ and $\pi(p^t|g^{1:T})$ for all t. We introduce some additional notation to simplify the subsequent derivations. The posterior probability $\bar{q}_i^t$ of the vehicle being at the state $x_i^t \in \boldsymbol{x}^t$ at time t, given all the observations $g^{1:T}$ of the trajectory, is defined as:

$$\overline{q}_i^t \propto \pi\left(x_i^t|g^{1:T}\right) = \frac{1}{Z} \sum_{\tau = x^1 \cdots p^{t-1} x_i^t p^t \cdots x^T} \phi\,(\tau|g^{1:T})$$

The operator $\propto$ indicates that this distribution is defined up to some scaling factor, which does not depend on x or p (but may depend on $g^{1:T}$). Indeed, we are interested in the probabilistic weight of a state x_i^t relative to the other possible states $x_{i'}^t$ at the state time t (and not to the actual, unscaled value of $\pi\left(x_i^t|g^{1:T}\right)$). This is why we consider $\left(\bar{q}_i^t\right)_i$ as a choice between a (finite) set of discrete variables, one choice per possible state x_i^t. A natural choice is to scale the distribution $\bar{q}_i^t$ so that the probabilistic weight of all possibilities is equal to 1:

$$\sum_{1 \le i \le I^t} \bar{q}_i^t = 1$$

From a practical perspective, $\bar{q}^t$ can be computed without the knowledge of the partition function Z. Indeed, the only required elements are the unscaled values of $\pi\left(x_i^t|g^{1:T}\right)$ for each i. The distribution $\bar{q}^t = \left(\bar{q}_i^t\right)_i$ is a multinomial distribution between I^t choices, one for each state. The quantity $\bar{q}_i^t$ has a clear meaning: it is the probability that the vehicle is in state x_i^t, when choosing amongst the set $\left(x_{i'}^t\right)_{1 \le i \le I^t}$, given all the observations $g^{1:T}$.

For each time t and each path index $j \in [1 \cdots J^t]$, we also introduce (up to a scaling constant) the discrete distribution over the paths at time t given the observations $g^{1:T}$:

$$\overline{r}_j^t \propto \pi\left(p_j^t|g^{1:T}\right)$$

which are scaled so that $\sum_{1 \le j \le J^t} \overline{r}_j^t = 1$.

This problem of smoothing in CRFs is a classic application of the Junction Tree algorithm to chain-structured graphs [54]. For the sake of completeness, we derive an efficient smoothing algorithm using our notations.

The definition of $\pi\left(x_i^t|g^{1:T}\right)$ requires summing the potentials of all the trajectories that pass through the state x_i^t at time t. The key insight for efficient filtering or smoothing is to make use of the chain structure of the graph, which lets us factorize the summation into two terms, each of which can be computed much faster than the exponentially large summation. Indeed, one can show from the structure of the clique graph that the following holds for all time steps t:

$$\pi(x^t|g^{1:T}) \propto \pi(x^t|g^{1:t})\pi(x^t|g^{t+1:T}) \tag{4}$$

The first term of the pair corresponds to the effect that the *past and present observations* ($g^{1:t}$) have on our belief of the present state x^t. The second term corresponds to the effect that the *future observations* ($g^{t+1:T}$) have on our estimation of the present state. The terms $\pi(x^t|g^{1:t})$ are related to each other by an equation that propagates *forward* in time, while the terms $\pi(x^t|g^{t+1:T})$ are related through an equation that goes *backward* in time. This is why we call $\pi(x^t|g^{1:t})$ the *forward distribution for the states*, and we denote it [5] by $\left(\overrightarrow{q}_i^t\right)_{1\leq i\leq I^t}$:

$$\overrightarrow{q}_i^t \propto \pi\left(x_i^t|g^{1:t}\right)$$

The distribution $\overrightarrow{q}_i^t$ is proportional to the posterior probability $\pi\left(x_i^t|g^{1:t}\right)$ and the vector $\overrightarrow{q}^t = \left(\overrightarrow{q}_i^t\right)_i$ is normalized so that $\sum_{i=1}^{I^t}\overrightarrow{q}_i^t = 1$. We do this for the paths, by defining the *forward distribution for the paths*:

$$\overrightarrow{r}_j^t \propto \pi\left(p_j^t|g^{1:t}\right)$$

Again, the distributions are defined up to a normalization factor so that each component sums to 1.

In the same fashion, we introduce the *backward distributions for the states and the paths:*

$$\overleftarrow{q}_i^t \propto \pi\left(x_i^t|g^{t+1:T}\right)$$

$$\overleftarrow{r}_j^t \propto \pi\left(p_j^t|g^{t+1:T}\right)$$

Using this set of notations, Equation (4) can be rewritten:

$$\begin{aligned} \overline{q}_i^t &\propto \overrightarrow{q}_i^t \cdot \overleftarrow{q}_i^t \\ \overline{r}_j^t &\propto \overrightarrow{r}_j^t \cdot \overleftarrow{r}_j^t \end{aligned}$$

Furthermore, $\overrightarrow{r}^t$ and $\overrightarrow{q}^t$ are related through a pair of recursive equations:

[5] The arrow notation indicates that the computations for $\overrightarrow{q}_i^t$ will be done forward in time.

$$\overrightarrow{q}_i^1 \propto \pi\left(x_i^1 | g^1\right)$$

$$\overrightarrow{r}_j^t \propto \eta\left(p_j^t\right)\left(\sum_{j:\underline{\delta}\left(x_i^t, p_j^t\right)=1} \overrightarrow{q}_i^t\right) \tag{5}$$

$$\overrightarrow{q}_i^t \propto \omega\left(x_i^t | g^t\right)\left(\sum_{j:\bar{\delta}\left(p_j^{t-1}, x_i^t\right)=1} \overrightarrow{r}_j^{t-1}\right) \tag{6}$$

Similarly, the backward distributions can be defined recursively, starting from $t = T$:

$$\overleftarrow{q}_i^T \propto 1$$

$$\overleftarrow{r}_j^t \propto \eta\left(p_j^t\right)\left(\sum_{j:\bar{\delta}\left(p_j^t, x_i^{t+1}\right)=1} \overleftarrow{q}_i^{t+1}\right) \tag{7}$$

$$\overleftarrow{q}_i^t \propto \omega\left(x_i^t | g^t\right)\left(\sum_{j:\underline{\delta}\left(x_i^t, p_j^t\right)=1} \overleftarrow{r}_j^t\right) \tag{8}$$

Details of the forward algorithm and backward algorithm are provided in Algorithm 1 and Algorithm 2 below. The complete algorithm for smoothing is detailed in the Algorithm 3.

Algorithm 1: Description of forward recursion

```
Given a sequence of observations $g^{1:T}$, a sequence of sets of candidate projections $\boldsymbol{x}^{1:T}$ and a sequence of sets of candidate paths $\boldsymbol{p}^{1:T-1}$:

Initialize the forward state distribution:
    $\forall i = 1 \cdots I^1$: $\overrightarrow{q}_i^1 \leftarrow \omega\left(x_i^1 | g^1\right)$
    Normalize $\overrightarrow{q}^1$

For every time step $t$ from 1 to $T-1$:
    Compute the forward probability over the paths:
    $\forall j = 1 \cdots J^t$:
     $\overrightarrow{r}_j^t \leftarrow \eta\left(p_j^t\right)\left(\sum_{j:\underline{\delta}\left(x_i^t,p_j^t\right)=1} \overrightarrow{q}_i^t\right)$
    Normalize $\overrightarrow{r}^t$
    Compute the forward probability over the states:
    $\forall i = 1 \cdots I^{t+1}$:
     $\overrightarrow{q}_i^{t+1} \leftarrow \omega\left(x_i^{t+1} | g^{t+1}\right)\left(\sum_{j:\bar{\delta}\left(p_j^t,x_i^{t+1}\right)=1} \overrightarrow{r}_j^t\right)$
    Normalize $\overrightarrow{q}^{t+1}$

Return the set of vectors $\left(\overrightarrow{q}^t\right)_t$ and $\left(\overrightarrow{r}^t\right)_t$
```

Algorithm 2: Description of backward recursion

```
Given a sequence of observations $g^{1:T}$, a sequence of sets of candidate projections $\boldsymbol{x}^{1:T}$ and a sequence of sets of candidate paths $\boldsymbol{p}^{1:T-1}$:

Initialize the backward state distribution
$\forall i = 1 \cdots I^T: \overleftarrow{q}_i^T \leftarrow 1$

For every time step $t$ from $T - 1$ to 1:
    Compute the forward probability over the paths:
    $\forall j = 1 \cdots J^t$:
        $\overleftarrow{r}_j^t \leftarrow \eta(p_j^t)\left(\sum_{j:\bar{\delta}\left(p_j^t, x_i^{t+1}\right)=1} \overleftarrow{q}_i^{t+1}\right)$
    Normalize $\overleftarrow{r}^t$
    Compute the forward probability over the states:
    $\forall i = 1 \cdots I^t$:
        $\overleftarrow{q}_i^t \leftarrow \omega\left(x_i^{t+1} | g^{t+1}\right)\left(\sum_{j:\underline{\delta}\left(x_i^t, p_j^t\right)=1} \overleftarrow{r}_j^t\right)$
    Normalize $\overleftarrow{q}^t$

Return the set of vectors $\left(\overleftarrow{q}^t\right)_t$ and $\left(\overleftarrow{r}^t\right)_t$
```

Algorithm 3: Trajectory smoothing algorithm

```
Given a sequence of observations $g^{1:T}$, a sequence of sets of candidate projections $\boldsymbol{x}^{1:T}$ and a sequence of sets of candidate paths $\boldsymbol{p}^{1:T-1}$:

Compute $\left(\overrightarrow{q}^{t}\right)_{t}$ and $\left(\overrightarrow{r}^{t}\right)_{t}$ using the forward algorithm.

Compute $\left(\overleftarrow{q}^{t}\right)_{t}$ and $\left(\overleftarrow{r}^{t}\right)_{t}$ using the backward algorithm.

For every time step $t$:
    $\forall j = 1 \cdots J^{t}: \overline{r}_{j}^{t} \leftarrow \overrightarrow{r}_{j}^{t} \cdot \overleftarrow{r}_{j}^{t}$
    Normalize $\overline{r}^{t}$
    $\forall i = 1 \cdots I^{t}: \overline{q}_{i}^{t} \leftarrow \overrightarrow{q}_{i}^{t} \cdot \overleftarrow{q}_{i}^{t}$
    Normalize $\overline{q}^{t}$

Return the set of vectors $\left(\overline{q}^{t}\right)_{t}$ and $\left(\overline{r}^{t}\right)_{t}$
```

The above smoothing algorithm requires all the observations of a trajectory in order to run. We have presented so far an *a posteriori* algorithm that requires full knowledge of measurements $g^{1:T}$. In this form, it is not directly suitable for real-time applications that involve streaming data, for which the data is available up to t only. However, this algorithm can be adapted for a variety of scenarios:

- *Smoothing*, also called *offline filtering*. This corresponds to getting the best estimate given all observations, i.e. to computing $\pi(x^t|g^{1:T})$. Algorithm 3 describes this procedure.
- *Tracking, filtering*, or *online estimation.* This usage corresponds to updating the current state of the vehicle as soon as a new streaming observation is available, i.e., to computing $\pi(x^t|g^{1:t})$. This is exactly the case the forward algorithm (Algorithm 1) is set to solve. If one is simply interested in the most recent estimate, then only the previous forward distribution $\vec{q}^{\,t}$ needs to be kept, and all distributions $\vec{q}^{\,t-1} \cdots \vec{q}^{\,1}$ at previous times can be discarded. This application minimizes the latency and the computations at the expense of the accuracy.
- *Lagged smoothing*, or *lagged filtering*. A few points of data are stored and processed before returning a result. Algorithm 4 details this procedure, which involves computing $\pi\left(x^t|g^{1:t+k}\right)$ for some $k > 0$. A trade-off is being made between the latency and the accuracy, as the information from the points $g^{t+1:t+k}$ is used to update the estimate of the state x^t. As shown in Section 4, even for small values of k, such a procedure can bring significant improvements in the accuracy while keeping the latency within reasonable bounds. A common ambiguity solved by lagged smoothing is presented in Figure 4-11.

Algorithm 4: Lagged smoothing algorithm

```
Given an integer $k > 0$, and a LIFO queue of observations:

Initialize the queue to the empty queue.

When receiving a new observation $g^t$:
    Push the observation in the queue
    Run the forward filter on this observation
    If $t > k$:
        Run the backward filter on the queue
        Compute $\overline{q}^{t-k}, \overline{r}^{t-k}$ on the first element of the queue
        Pop the queue and return $\overline{q}^{t-k}$ and $\overline{r}^{t-k}$
```

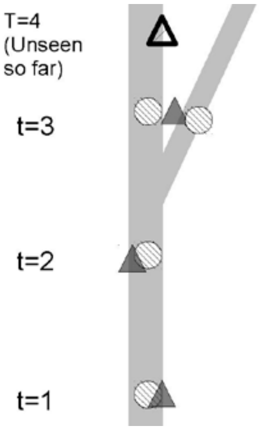


Figure 4-11: Example of case handled by lagged smoothing which disambiguates the results provided by tracking. An observation is available close to an exit ramp of a highway, for which the algorithm has to decide if it corresponds to the vehicle exiting the highway. Lagged smoothing analyzes subsequent points in the trajectory and can disambiguate the situation.

3.4 OBSERVATION MODEL

We now describe the observation model ω. The observation probability is assumed only to depend on the distance between the point and the GPS coordinates. We take an isoradial Gaussian noise model:

$$\begin{aligned}\omega(g|x) &= p\big(\mathrm{d}(g,x)\big)\\ &= \frac{1}{\sqrt{2\pi}\sigma}\left(-\frac{1}{2\sigma^2}\,\mathrm{d}(g,x)^2\right)\end{aligned}$$

in which the function d is the distance function between geocoordinates. The standard deviation σ is assumed to be constant over all the network. This is not true in practice because of well documented urban canyoning effects [29] [60] [48] and satellite occlusions. Updating the model accordingly presents no fundamental difficulty, and can be done by geographical clustering of the regions of interest. Using the estimation techniques described later in Section 4.3 and Section 4.4, an estimate of σ between 10 and 15 meters could be estimated for data of interest in this study.

3.5 DRIVER MODEL

The second model to consider is the driver behavior model. This model assigns a weight to any acceptable path on the road network. We consider a model in the *exponential family*, in which the weight distribution over any path p only depends on a selected number of features $\varphi(p) \in \mathbb{R}^K$ of the path. Possible features include the length of the path, the number of stop signs, and the speed limits on the road. The distribution is parametrized by a vector $\mu \in \mathbb{R}^K$ so that the logarithm of the distribution of paths is a linear combination of the features of the path:

$$\eta(p) \propto \exp\left(\mu^T \varphi(p)\right)$$

The function φ is called the *feature function,* and the vector μ is called the behavioral *parameter vector*, and simply encodes a weighted combination of the features.

In a simple model the vector $\varphi(p)$ may be reduced to a single scalar, such as the length of the path. Then the inverse of μ, a length, can be interpreted as a characteristic length. This model simply states that the driver has a preference for shorter paths, and μ^{-1} indicates how aggressively this driver wants to follow the shortest path. Such a model is explored in Section 4. Other models considered include the mean speed and travel times, the stop signs and signals, and the turns to the right or to the left.

4 TRAINING PROCEDURE

The procedure detailed so far requires the calibration of the observation model and the path selection model by setting some values for the weight vector μ and the standard deviation σ. Using standard machine learning techniques, we maximize the likelihood of the observations with respect to the parameters, and we evaluate the result against held-out trajectories using several metrics detailed in Section 5. Computing likelihood will require the computation of the partition function (which depends on μ and σ). We first present a procedure that is valid for any path or point distributions that belong to the *exponential family*, and then show how the models we presented in Section 3 fit into this framework.

4.1 LEARNING WITHIN THE EXPONENTIAL FAMILY AND SPARSE TRAJECTORIES

There is a striking similarity between the state variables $x^{1:T}$ and the path variables $p^{1:T}$ especially between the forward and backward distributions introduced in Equation (5). This suggests to generalize our procedure to a context larger than states interleaved with paths. Indeed, each step of choosing a path or a variable corresponds to making a *choice* between a finite number of possibilities, and there is a limited number of pairwise compatible choices (as encoded by the functions $\underline{\delta}$ and $\bar{\delta}$). Following a trajectory corresponds to choosing a new state (subject to the compatibility constraints of the previous state). In this section, we introduce the proper notation to generalize our learning problem, and then show how this learning problem can be efficiently solved. In the next section, we will describe the relation between the new variables we are going to introduce and the parameters of our model.

Consider a joint sequence of multinomial random variables $\boldsymbol{z}^{1:L} = \boldsymbol{z}^1 \cdots \boldsymbol{z}^L$ drawn from some space $\prod_{l=1}^{L}\{1 \cdots K^l\}$ where K^l is the dimensionality of the multinomial variable $\boldsymbol{z}^l$. Given a realization $z^{1:L}$ from $\boldsymbol{z}^{1:L}$, we define a non-negative potential function $\psi(z^{1:L})$ over the sequence of variables. This potential function is controlled by a parameter vector $\theta \in \mathbb{R}^M$: $\psi(z^{1:L}) = \psi(z^{1:L};\theta)$ [6]. Furthermore, we assume that this potential function is also defined and non-negative over any subsequence $\psi\left(z^{1:l}\right)$. Lastly, we assume that there exists at least one sequence $z^{1:L}$ that has a positive potential. As in the previous section, the potential function ψ, when properly normalized, defines a probability distribution of density π over the variables $\boldsymbol{z}$, and this distribution is parametrized by the vector θ:

$$\pi(z;\theta) = \frac{\psi(z;\theta)}{Z(\theta)} \tag{9}$$

with $Z = \sum_z \psi\,(z;\theta)$ called the *partition function*. We will show the partition function defined here is the partition function introduced in Section 3.1.

[6] The semicolon notation indicates that this function is parametrized by θ, but that θ is not a random variable.

We assume that ψ is an unscaled member of the *exponential family*: it is of the form:

$$\psi(z;\theta)=h(z)\prod_{l=1}^{L}e^{\theta\cdot T^{l}(z^{l})} \tag{10}$$

In this representation, h is a non-negative function of z which does not depend on the parameters, the operator $\cdot$ is the vector dot product, and the vectors $T^{l}(z^{l})$ are vector mappings from the realization z^{l} to $\mathbb{R}^{M}$ for some $M\in\mathbb{N}$, called *feature vectors*. Since the variable z^{l} is discrete and takes on values in $\{1\cdots K^{l}\}$, it is convenient to have a specific notation for the feature vector associated with each value of this variable:

$$\forall i\in\{1\cdots K^{l}\},T_{i}^{l}=T^{l}(z^{l}=i)$$

The sequence of variables $\boldsymbol{z}$ represents the choices associated with a single trajectory, i.e. the concatenation of the xs and ps. In general, we will observe and would like to learn from multiple trajectories at the same time. This is why we need to consider a collection of variables $\left(\boldsymbol{z}^{(u)}\right)_{u}$, each of which follows the form above and each of which we can define a potential $\psi\left(z^{(u)};\theta\right)$ and a partition function $Z^{(u)}(\theta)$ for. There the variable u indexes the set of sequences of observations, i.e. the set of consecutive GPS measurements of a vehicle. Since each of these trajectories will take place on a different portion of the road network, each of the sequences $\boldsymbol{z}^{(u)}$ will have a different state space. For each of these sequences of variables $\boldsymbol{z}^{(u)}$, we observe the respective realizations $z^{(u)}$ (which correspond to the observation of a trajectory), and we wish to infer the parameter vector θ^{*} that maximizes the likelihood of all the realizations of the trajectories:

$$\begin{aligned}\theta^{*}&=\arg\max_{\theta}\sum_{u}\log\,\pi^{(u)}\left(z^{(u)};\theta\right)\\&=\arg\max_{\theta}\sum_{u}\log\,\psi\left(z^{(u)};\theta\right)-\log\,Z^{(u)}\left(\theta\right)\\&=\arg\max_{\theta}\sum_{u}\sum_{l=1}^{L^{(u)}}\theta\cdot T^{l^{(u)}}\left(z^{l^{(u)}}\right)-\log\,Z^{(u)}\left(\theta\right)\end{aligned} \tag{11}$$

where again the indexing u is for sets of measurements of a given trajectory. Similarly, the length of a trajectory is indexed by u: $L^{(u)}$. From Equation 11, it is clear that the log-likelihood function simply sums together the respective likelihood functions of each trajectory. For clarity, we consider a single sequence $z^{(u)}$ only and we remove the indexing with respect to u. With this simplification, we have for a single trajectory:

$$\log\,\psi\left(z;\theta\right)-\log\,Z\left(\theta\right)\quad=\sum_{l=1}^{L}\theta\cdot T^{l}(z^{l})-\log\,Z\left(\theta\right) \tag{12}$$

The first part of Equation (12) is linear with respect to θ and $\log\ Z\left(\theta\right)$ is concave in θ (it is the logarithm of a sum of exponentiated linear combinations of θ [61]) . As such, maximizing Equation (12) yields a unique solution (assuming no singular parametrization), and some superlinear algorithms exist to solve this equation [61]. These algorithms rely on the computation of the gradient and the Hessian matrix of $\log\ Z\,(\theta)$. We now detail some closed-form recursive formulas to compute these elements.

1. Efficient estimation of the partition function

A naive approach to the computation of the partition function $Z(\theta)$ leads to consider exponentially many paths. Most of these computations can be factored using dynamic programming [7]. Recall the definition of the partition function:

$$Z(\theta) = \sum_{z} h\,(z) \prod_{l=1}^{L} e^{\theta \cdot T^{l}(z^{l})}$$

So far, the function h was defined in in a generic way (it is non-negative and does not depend on θ). We consider a particular shape that generalizes the functions $\underline{\delta}$ and $\bar{\delta}$ introduced in the previous section. In particular, the function h is assumed to be a binary function, from the Cartesian space $\prod_{l=1}^{L}\{1 \cdots K^{l}\}$ to $\{0,1\}$, that decomposes to the product of binary functions over consecutive pairs of variables:

$$h(z) = \prod_{l=1}^{L-1} h^{l}\left(z^{l}, z^{l-1}\right)$$

in which every function h^{l} is a binary indicator $h^{l}\colon \{1 \cdots K^{l}\} \times \{1 \cdots K^{l-1}\} \to \{0,1\}$. These functions h^{l} generalize the functions $\underline{\delta}$ and $\overline{\delta}$ for arguments z equal to either the xs or the ps. It indicates the compatibility of the values of the instantiations z^{l} and z^{l-1}

Finally, we introduce the following notation. For each index $l \in [1 \cdots L]$ and subindex $i \in [1 \cdots K^{l}]$, we call Z_{i}^{l} the partial summation of all partial paths $\boldsymbol{z}\ ^{1:l}$ that terminate at the value $z^{l} = i$:

$$\begin{aligned} Z_{i}^{l}(\theta) &= \sum_{z^{1:l}:z^{l}=i} h\left(z^{1:l}\right) \prod_{m=1}^{l} e^{\theta \cdot T^{m}(z^{m})} \\ &= \sum_{z^{1:l}:z^{l}=i} e^{\theta \cdot T^{1}(z^{1})} \prod_{m=2}^{l} h^{m}\,(z^{m}, z^{m-1}) e^{\theta \cdot T^{m}(z^{m})} \end{aligned}$$

[7] This is—again—a specific application of the junction tree algorithm. See [54] for an explanation of the general framework.

This partial summation can also be defined recursively:

$$Z_i^l(\theta) \quad = e^{\theta \cdot T_i^l} \sum_{j \in [1 \dots K^{l-1}] : h^l(z^i, z^j) = 1} Z_j^{l-1}(\theta) \tag{13}$$

The start of the recursion is for all $i \in \{1 \cdots K^1\}$:

$$Z_i^1(\theta) = e^{\theta \cdot T_i^1}$$

and the complete partition function is the summation of the auxiliary values:

$$Z(\theta) = \sum_{i=1}^{K^L} Z_i^L(\theta)$$

Computing the partition function can be done in polynomial time complexity by a simple application of dynamic programming. By using sparse data structures to implement h, some additional savings in computations can be made[8].

2. Estimation of the gradient

The estimation of the gradient for the first part of the log likelihood function is straightforward. The gradient of the partition function can also be computed using Equation (13):

$$\nabla_\theta Z_i^l(\theta) \quad = Z_i^l(\theta) T_i^l + e^{\theta \cdot T_i^l} \sum_{j : h^l(z^i, z^j) = 1} \nabla_\theta Z_j^{l-1}(\theta)$$

The Hessian matrix can be evaluated in similar fashion:

[8] In particular, care should be taken to implement all the relevant computations in log-domain due to the limited precision of floating point arithmetic on computers. The reference implementation [59] shows one way to do it.

$$\begin{aligned}
\triangle_{\theta\theta}\, Z_i^l(\theta) = \quad & Z_i^l(\theta)\left(T_i^l\right)\left(T_i^l\right)' \\
& + e^{\theta \cdot T_i^l}\left(\sum_{j:h^l\left(z^i,z^j\right)=1} \nabla_\theta\, Z_j^{l-1}(\theta)\right)\left(T_i^l\right)' \\
& + e^{\theta \cdot T_i^l}\left(T_i^l\right)\left(\sum_{j:h^l\left(z^i,z^j\right)=1} \nabla_\theta\, Z_j^{l-1}(\theta)\right)' \\
& + e^{\theta \cdot T_i^l} \sum_{j:h^l\left(z^i,z^j\right)=1} \triangle_{\theta\theta}\, Z_j^{l-1}(\theta)
\end{aligned}$$

4.2 EXPONENTIAL FAMILY MODELS

We now express our formulation of Conditional Random Fields to a form compatible with Equation (10).

Consider $\epsilon = \sigma^{-2}$ and θ the stacked vector of the desired parameters:

$$\theta = \binom{\epsilon}{\mu}$$

There is a direct correspondence between the path and state variables with the $\boldsymbol{z}$ variables introduced above. Let us pose $L = 2T - 1$, then for all $l \in [1, L]$ we have:

$$z^{2t} = r^t$$

$$z^{2t-1} = q^t$$

and the feature vectors are simply the alternating values of φ and d, completed by some zero values:

$$T_i^{2t} = \begin{pmatrix} 0 \\ \varphi\left(p_i^t\right) \end{pmatrix}$$

$$T_j^{2t-1} = \begin{pmatrix} -\frac{1}{2}\mathrm{d}\left(g, x_j^t\right)^2 \\ \mathbf{0} \end{pmatrix}$$

These formulas establish how we can transform our learning problem that involves paths and states into a more abstract problem that considers a single set of variables.

4.3 SUPERVISED LEARNING WITH KNOWN TRAJECTORIES

The most straightforward way to learn μ and σ, or equivalently to learn the joint vector θ, is to maximize the likelihood of some GPS observations $g^{1:T}$, knowing the complete trajectory followed by the vehicle. For all time t, we also know which path p_{observed}^{t} was taken and which state x_{observed}^{t} produced the GPS observation g^{t}. We make the assumption that the observed path p_{observed}^{t} is one of the possible path amongst the set of candidate paths $\left(p_{j}^{t}\right)_{j}$:

$$\exists j \in [1 \cdots J^{t}] : \; p_{\text{observed}}^{t} = p_{j}^{t}$$

and similarly, that the observed state x_{observed}^{t} is one of the possible states:

$$\exists i \in [1 \cdots I^{t}] : \; x_{\text{observed}}^{t} = x_{i}^{t}$$

In this case, the values of r^{t} and q^{t} are known (they are the matching indexes), and the optimization problem of Equation (11) can be solved using methods outlined in Section 4.1.

4.4 UNSUPERVISED LEARNING WITH INCOMPLETE OBSERVATIONS: EXPECTATION MAXIMIZATION

Usually, only the GPS observations $g^{1:T}$ are available; the values of $r^{1:T-1}$ and $q^{1:T}$ (and thus $z^{1:L}$) are hidden to us. In this case, we estimate the *expected likelihood* $\mathcal{L}$, which is the expected value of the likelihood under the distribution over the assignment variables $\boldsymbol{z^{1:L}}$:

$$\mathcal{L}(\theta) = \mathbb{E}_{\boldsymbol{z}\sim\boldsymbol{\pi}(\cdot|\boldsymbol{\theta})}\left[\log\left(\pi(z;\theta)\right)\right] \tag{14}$$

$$= \sum_{z} \pi\,(z;\theta) \log\left(\pi(z;\theta)\right) \tag{15}$$

The intuition behind this expression is quite natural: since we do not know the value of the assignment variable z, we consider the *expectation* of the likelihood over this variable. This expectation is done with respect to the distribution $\pi(z;\theta)$. The challenge lies in the dependency in θ of the very distribution used to take the expectation. Computing the expected likelihood becomes much more complicated than simply solving the optimization problem of (11).

One strategy is to find some “fill in” values for z that would correspond to our guesses of which path was taken, and which point made the observation. However, such a guess would likely involve our model for the data, which we are currently trying to learn. A solution to this chicken and egg problem is the Expectation Maximization (EM) algorithm [62]. This algorithm performs an iterative projection ascent by assigning some *distributions* (rather than singular values) to every z^{l}, and uses these distributions to updates the parameters μ and σ using the procedures seen in Section 4.3. This iterative procedure performs two steps:

1. Fixing some value for θ, it computes a distribution $\tilde{\pi}(z) = \pi(z;\theta)$
2. It then uses this distribution $\tilde{\pi}(z)$ to compute some new value of θ by solving the approximate problem in which the expectation is fixed with respect to θ:

   $$\max_{\theta} \mathbb{E}_{z\sim\tilde{\pi}(\cdot)}\left[\log\left(\pi(z;\theta)\right)\right] \tag{16}$$

   This problem is significantly simpler than the optimization problem in Equation (14) since the expectation itself does not depend on θ and thus is not part of the optimization problem.

Under this procedure, the expected likelihood is shown to converge to a local maximum [54]. It can be shown that good values for the plug-in distribution $\tilde{\pi}$ are simply the values of the posterior distributions $\pi(p^t|g^{1:T})$ and $\pi(x^t|g^{1:T})$, i.e. the values $\overline{q}^t$ and $\bar{r}^t$. Furthermore, owing to the particular shape of the distribution $\pi(z)$, taking the expectation is a simple task: we simply replace the value of the feature vector by its *expected value* under the distribution $\tilde{\pi}(z)$. More practically, we simply have to consider:

$$\begin{aligned} T^{2t}(z^{2t}) &= \mathbb{E}_{p\sim\pi(\cdot|\theta,g^{/1:T})}\left[\begin{pmatrix} 0 \\ \varphi(p_r^t) \end{pmatrix}\right] \\ &= \begin{pmatrix} 0 \\ \mathbb{E}_{p\sim\pi(\theta,g^{1:T})}[\varphi(p_r^t)] \end{pmatrix} \end{aligned} \tag{17}$$

in which

$$\mathbb{E}_{p\sim\pi(\theta,g^{1:T})}[\varphi(p_r^t)] = \sum_{i=1}^{I^t} \overline{r}_i^t\, \varphi_i^t$$

and

$$T^{2t-1}(z^{2t-1}) = \begin{pmatrix} -\frac{1}{2}\mathbb{E}_{x\sim\pi(\cdot|\theta,g^{1:T})}\left[\mathrm{d}\left(g, x_{q^t}^t\right)^2\right] \\ \mathbf{0} \end{pmatrix} \tag{18}$$

so that

$$\mathbb{E}_{x\sim\pi(\cdot|\theta,g^{1:T})}\left[\mathrm{d}\left(g, x_{q^t}^t\right)^2\right] = \sum_{i=1}^{J^t} \bar{q}_i^t\, \mathrm{d}\left(g, x_i^t\right)^2$$

These values of feature vectors plug directly into the supervised learning problem in Equation (11) and produce updated parameters μ and σ, which are then used in turn for updating the values of $\bar{q}$ and $\bar{r}$ and so on.

Algorithm 5: Expectation maximization algorithm for learning parameters without complete observations

```
Given a set of sequences of observations, an initial value of $\theta$

Repeat until convergence:
    For each sequence, compute $\overline{r}^t$ and $\overline{q}^t$ using Algorithm ?.
    For each sequence, update expected values of $T^t$ using (17) and (18).
    Compute a solution of Problem (11) using these new values of $T^t$.
```

5 RESULTS FROM FIELD OPERATIONAL TEST

The path inference filter and its learning procedures were tested using field data through the *Mobile Millennium* system. Ten San Francisco taxicabs were fit with high frequency GPS (1 second sampling rate) in October 2010 during a two-day experiment. Together, they collected about seven hundred thousand measurement points that provided a high-accuracy ground truth. Additionally, the unsupervised learning filtering was tested on a significantly larger dataset: one day one-minute samples of 600 taxis from the same fleet, which represents 600 000 points. For technical reasons, the two datasets could not be collected the same day, but were collected the same day of the week (a Wednesday) three weeks prior to the high-frequency collection campaign. Even if the GPS equipment was different, both datasets presented the same distribution of GPS dispersion. Thus we evaluate two datasets collected from the same source with the same spatial features: a smaller set at high frequency, called “Dataset 1”, and a larger dataset sampled at 1 minute for which we do not know ground truth, called “Dataset 2”.

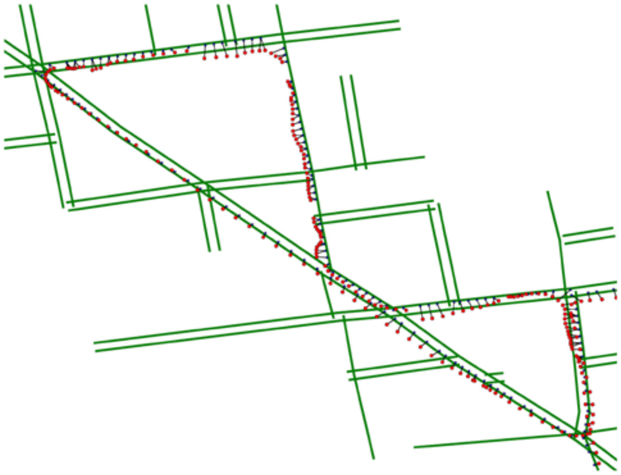


Figure 4-12: Example of points collected in “Dataset 1”, in the Russian Hill neighborhood in San Francisco. The (red) dots are the GPS observations (collected every second), and the green lines are road links that contain a state projection. The black lines show the most likely projection of the GPS points on the road network, using the Viterbi algorithm on a gridded state-space with a 1-meter grid for the offsets.

5.1 EXPERIMENT DESIGN

The testing procedure is described in Algorithm 6: The filter was first run in trajectory reconstruction mode (Viterbi algorithm) with settings and-tuned for a high-frequency application, using all the samples, in order to build a set of ground truth trajectories. The trajectories were then downsampled to different temporal resolutions and were used to test the filter in different configurations.

Algorithm 6: Evaluation procedure

Given a set of high-frequency sequences of raw GPS data:

1. Map the raw high-frequency sequences on the road network
2. Run the Viterbi algorithm with default settings
3. Extract the most likely HF trajectory on the road network for each sequence
4. Given a set of projected HF trajectories:
    (a) Decimate the trajectories to a given sampling rate
    (b) Separate the set into a training subset and a test subset
    (c) Compute the best model parameters for a number of learning methods (most likely, EM with a simple model or a more complex model)
    (d) Evaluate the model parameters with respect to different computing strategies (Viterbi, online, offline, lagged smoothing) on the test subset

The following features were tested:

- The sampling rate. The following values were tested: 1 second, 10 seconds, 30 seconds, one minute, one and a half minute and two minutes

- The computing strategy: pure filtering (“online” or forward filtering), fixed-lagged smoothing with a one- or two-point buffer (“1-lag” and “2-lag” strategies), Viterbi and smoothing (“offline”, or forward-backward procedure).

- Different models:

    - “Hard closest point”: A greedy deterministic model that computes the closest point and then finds the shortest path to reach this closest point from the previous point. This non-probabilistic model is the baseline against which we make comparison on [39]. This greedy model may lead to non-feasible trajectories, for example by assigning an observation to a dead end link from which it cannot recover.

- “Closest point” : A non-greedy version of “Hard closest point”. Among all the feasible trajectories, this (naive, deterministic) model projects all the GPS data to their closest projections and then selects the shortest path between each projection. The computing strategy chosen is important because the filter may determine that some projections lead to dead end and force the trajectory to break.

- “Shortest path”: A naive model that selects the shortest path. Given paths of the same length, it will take the path leading to the closest point. The points projections are then recovered from the paths. This is similar to [47] [37].

- “Simple” A simple model that considers two features that could be tuned by hand:

  1. ξ_1 : The length of the path

  2. ξ_2 : The distance of a point projection to its GPS coordinate

  This model was trained on learning data by two procedures:

  - Supervised learning, in which the true trajectory is provided to the learning algorithm leading to the “MaxLL-Simple” model

  - Unsupervised learning, which produced the model called “EM-Simple”

- “Complex” : A more complex model with a more diverse set of features, which is complicated enough to discourage manual tuning:

  1. The length of the path

  2. The number of stop signs along the path

  3. The number of signals (red lights)

  4. The number of left turns made by the vehicle at road intersections

  5. The number of right turns made by the vehicle at road intersections

  6. The minimum average travel time (based on the speed limit)

  7. The maximum average speed

  8. The maximum number of lanes (representative of the class of the road)

  9. The minimum number of lanes

10. The distance of a point to its GPS point

    This model was first evaluated using supervised learning leading to the model called "MaxLL-Complex". The unsupervised learning procedure was also tried but failed to properly converge when using "Dataset 1", obtained from high-frequency samples. Unsupervised learning was run again with "Dataset 2", using the simple model as a start point and converged properly this time. This set of parameters is presented under the label "EM-Complex".

All the models above are specific cases of our framework:

- "Simple" is a specific case of "Complex", by restricting the complex model to only two features.
- "Shortest path" is a specific case of "Simple" with $|\xi_1| \gg 1$, $|\xi_2| \ll 1$. We used $\xi_1 = -1000$ and $\xi_2 = -0.001$
- "Closest point" is a specific case of "Simple" with $|\xi_1| \ll 1$, $|\xi_2| \gg 1$. We used $\xi_1 = -0.001$ and $\xi_2 = -1000$
- "Hard closest point" can be reasonably approximated by running the "Closest point" model with the Online filtering strategy.

Thanks to this observation, we implemented all the model using the same code and simply changed the set of features and the parameters [59].

These models were evaluated under a number of metrics:

- The proportion of path misses: for each trajectory, it is the number of times the most likely path was not the true path followed, divided by the number of time steps in the trajectory.
- The proportion of state misses: for each trajectory, the number of times the most likely projection was not the true projection.
- The log-likelihood of the true point projection. This is indicative of how often the true point is identified by the model.
- The log-likelihood of the true path.
- The entropy of the path distribution and of the point distribution. This statistical measure indicates the confidence assigned by the filter to its result. A small entropy (close to 0) indicates that one path is strongly favored by the filter against all the other ones, whereas a large entropy indicates that all paths are equal.
- The miscoverage of the route. Given two paths p and p' the coverage of p by p', denoted $\mathrm{cov}(p, p')$ is the amount of length of p that is shared with p' (it is a semi-distance since it is not symmetric). It is thus lower than the total length $|p|$ of the path p. We measure the dissimilarity

of two paths by the *relative miscoverage*: $\mathrm{mc}(p) = 1 - \frac{cov(p^*,p)}{|p^*|}$. If a path is perfectly covered, its relative miscoverage will be 0.

For about 0.06% of pairs of points, the true path could not be found by the A* algorithm and was manually added to the set of discovered paths

Each training session was evaluated with k-fold cross-validation, using the following parameters:

Sampling rate (seconds)	Batches used for validation	Batches used for training
1	1	5
10	3	5
30	6	5
60	6	5
90	6	5
120	6	5

5.2 RESULTS

Given the number of parameters to adjust, we only present the most salient results here.

The most important practical result is the raw accuracy of the filter: for each trajectory, which proportion of the paths or of the points was correctly identified? These results are presented in Figure 4-13 and Figure 4-14. As expected, the error rate is 0 for high frequencies (low sampling period): all the points are correctly identified by all the algorithms. In the low frequencies (high sampling periods), the error is still low (around 10%) for the trained models, and also for the greedy model ("Hard closest point"). For sampling rates between 10 seconds and 90 seconds, trained models ("Simple" and "Complex") show a much higher performance compared to untrained models ("Hard closest point", "Closest point" and "Shortest path").

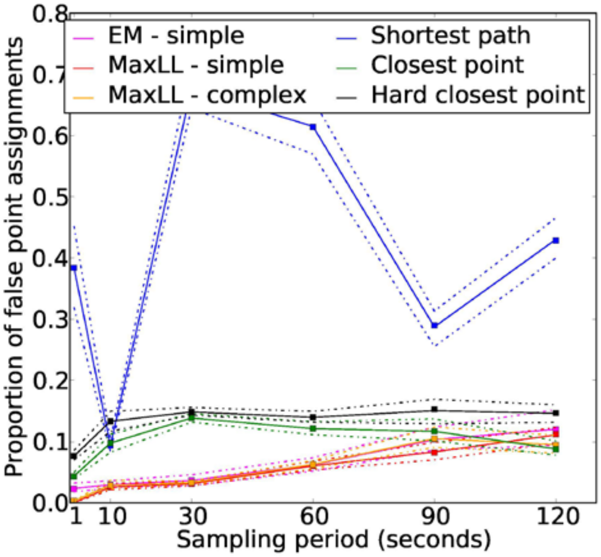


Figure 4-13: Point misses using trajectory reconstruction (Viterbi algorithm) for different sampling rates, as a percentage of incorrect point reconstructions for each trajectory (positive, smaller is better). The solid line denotes the median, the squares denote the mean and the dashed lines denote the 94% confidence interval. The black curve is the performance of a greedy reconstruction algorithm, and the colored plots are the performances of probabilistic algorithms for different features and weights learned by different methods. As expected, the error rate is close to 0 for high frequencies (low sampling rates): all the points are correctly identified by all the algorithms. In the low frequencies (high sampling rates), the error still stays low (around 10%) for the probabilistic models, and also for the greedy model. For sampling rates between 10 seconds and 90 seconds, tuned models show a much higher performance compared to greedy models (Hard closest point, closest point and shortest path). However, we will see that the errors made by tuned models are more benign than errors made by simple greedy models.

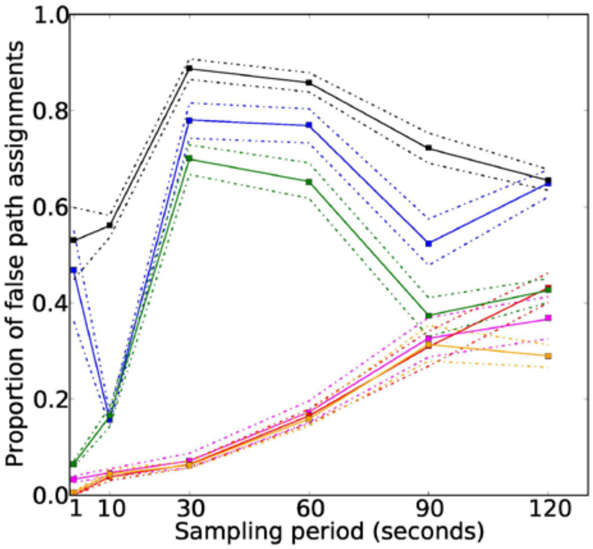


Figure 4-14: Path misses using the Viterbi reconstruction for different models and different sampling rates, as a percentage on each trajectory (lower is better). The solid line denotes the median, the squares denote the mean and the dashed lines denote the 98% percentiles. The error rate is close to 0 for high frequencies: the paths are correctly identified. In higher sampling regions, there are many more paths to consider and the error increases substantially. Nevertheless, the probabilistic models still perform very well: even at 2 minute intervals, they are able to recover about 75% of the true paths. In particular, in these regions the shortest path becomes a viable choice for most paths. Note how the greedy path reconstruction fails rapidly as the sampling increases. Also note how the shortest path heuristic performs poorly.

We now turn our attention to the resilience of the models, i.e. how they perform when they make mistakes. We use two statistical measures: the (log) likelihood of the true paths (Figure 4-15) and the entropy of the distribution of points or paths (Figure 4-16 and Figure 4-17). Note that in a perfect reconstruction with no ambiguity, the log likelihood would be zero. Interestingly, the log likelihoods appear very stable as the sampling interval grows: our algorithm will continue to assign high probabilities to the true projections even when many more paths can be used to travel from one point to the other. The performance of the simple and the complex models improves greatly when some backward filtering steps are used, and stays relatively even across different time intervals.

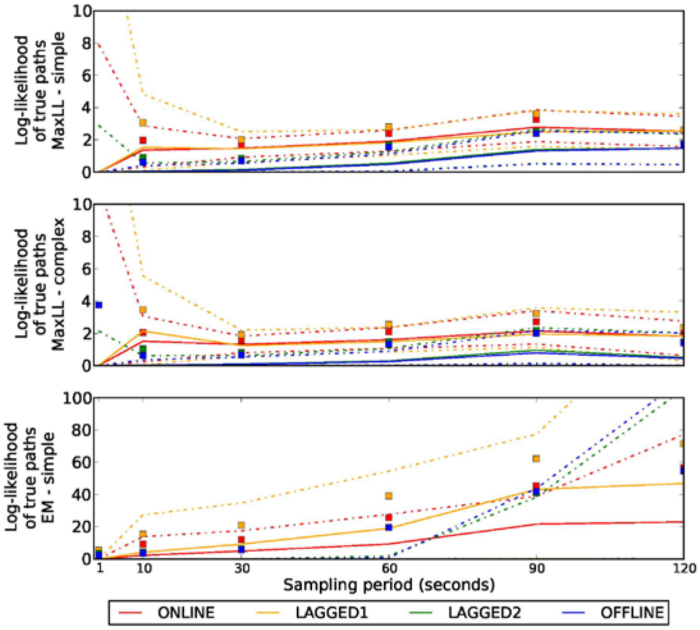


Figure 4-15: (Negative of) Log likelihood of true paths for different strategies and different sampling rates (positive, lower is better). The error bars denote the first and last quartiles (the 25th and 75th percentiles). The solid line denotes the median, the squares denote the mean and the dashed lines denote the 98% confidence interval. The likelihood decreases as the sampling interval increases, which was to be expected. Note the relatively high mean likelihood compared to the median : a number of true paths are assigned very low likelihood by the model, but this phenomenon is mitigated by using better filtering strategies (2-lagged and smoothing). The use of a more complex model (that accounts for a finer set of features for each path) brings some improvements on the order of 25% of all metrics. The behavior around high frequencies (1 and 10 second time intervals) is also very interesting. Most of the paths are chosen nearly perfectly (the median is 0), but the filters are generally too confident and assign very low probabilities to their outputs, which is why the likelihood has a very heavy tail at high frequency. Note also that in the case of high frequency, the use of an offline filter brings significantly more accurate results than a 2-lagged filter. This difference disappears rapidly (it becomes insignificant at 10 second intervals). Note how the EM trained filter performs worse in the low frequencies (note the difference of scale). The points for online strategy (red) and for 2-lagged filtering (green) do not appear because they are too close to the 1-lagged and offline strategies, respectively. Again in the EM setting, the offline and 2-lagged filters perform considerably better than the cruder strategies.

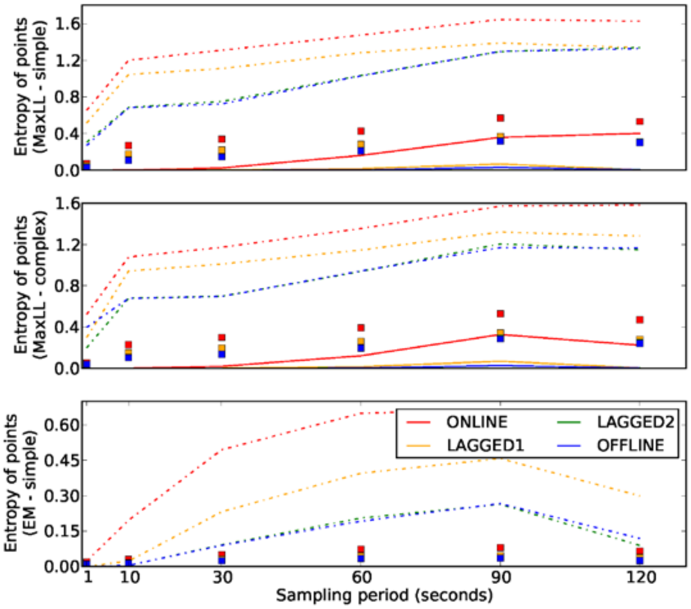


Figure 4-16: Distributions of point entropies with respect to sampling and for different models. The colors show the performance of different filtering strategies (pure online, 1-lag, 2-lag and offline). The entropy is a measure of the confidence of the filter on its output and quantifies the spread of the probability distribution over all the candidate points. The solid line denotes the median, the squares denote the mean and the dashed lines denote the 95% confidence interval. The entropy starts at nearly zero for high frequency sampling : the filters are very confident in their outputs. As sampling time increases, the entropy at the output of the online filter increases notably. Since the online filter cannot go back to update its belief, it is limited to pure forward prediction and as such cannot confidently choose a trajectory that would work in all settings. For the other filtering strategies, the median is close to zero while the mean is substantially higher. Indeed, the filter is very confident in its output most of the time and assigns a weight of nearly one to one candidate, and nearly zero to all the other outputs, but it is uncertain in a few cases. These few cases are at the origin of the fat tail of the distributions of entropies and the relatively wide confidence intervals. Note that using a more complex model improves the mean entropy by about 15%. Also, in the case of EM, the entropy is very low (note the difference of scale): the EM model is overconfident in its predictions and tends to assigns very large weights to a single choice, even if it not the good one.

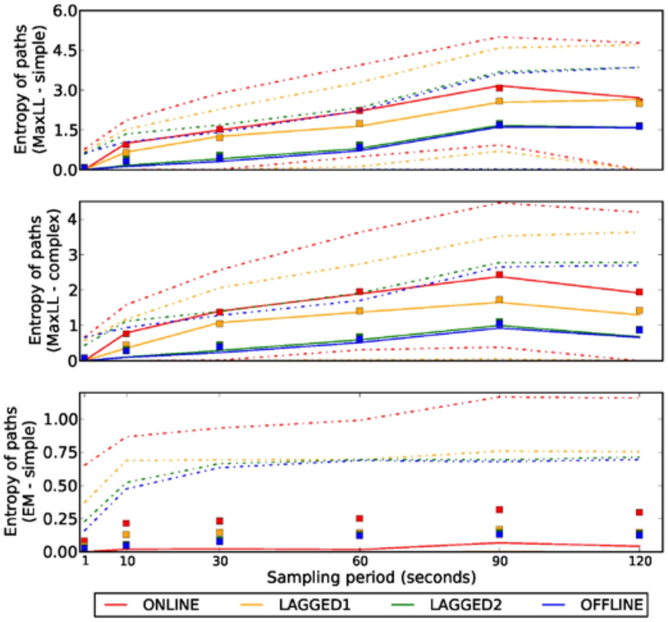


Figure 4-17: Distributions of path entropies with respect to sampling period and for different models (positive, lower is better). The colors show the performance of different filtering strategies (purely online, 1-lag, 2-lag and offline) The entropy is a measure of the confidence of the filter on its output and quantifies the spread of the probability distribution over all the candidate paths. The solid line denotes the median, the squares denote the mean and the dashed lines denote the 95% confidence interval. Compared to the points, the paths distributions have a higher entropy: the filter is much less confident in choosing a single path and spreads the probability weights across several choices. Again, the use of 2-lagged smoothing is as good as pure offline smoothing, for the same computing cost and a fraction of the data. Online and 1-lagged smoothing perform about as well, and definitely worse than 2-lagged smoothing. The use of a more complex model strongly improves the performance of the filter: it results in more compact distribution over candidate paths. Again, the model learned with EM is overconfident and tends to offer favor a single choice, except for a few path distributions.

We conclude the performance analysis by a discussion of the miscoverage (Figure 4-18). The miscoverage gives a good indication of how far the path chosen by the filter differs from the true path. Even if the paths are not exactly the same, some very similar path may get selected, that may differ by a turn around a block. Note that the metric is based on length covered. At high frequency however, the vehicle may be stopped and cover a length 0. This metric is thus less useful at high frequency. A more complex model improves the coverage by about 15% in smoothing. In high sampling resolution, the error is close to zero: the paths considered by the filter, even if they do not match perfectly, are very close to the true trajectory for lower frequencies. Two groups clearly emerge as far as computing strategies are concerned: the online/1-lag group (orange and red curves) and the 2-lag and offline group (green and blue curves). The relative miscoverage for the latter group is so low that more than half of the probability mass is at zero. A number of outliers still raise the curve of the last quartile as well as the mean, especially in the lower frequencies. The paths inferred by the filter are never dramatically different: at two minute time intervals (for which the paths are 1.7km on average), the returned path spans more than 80% of the true path on average. The use of a more complicated model decreases the mean miscoverage as well as all quartile metrics by more than 15%.

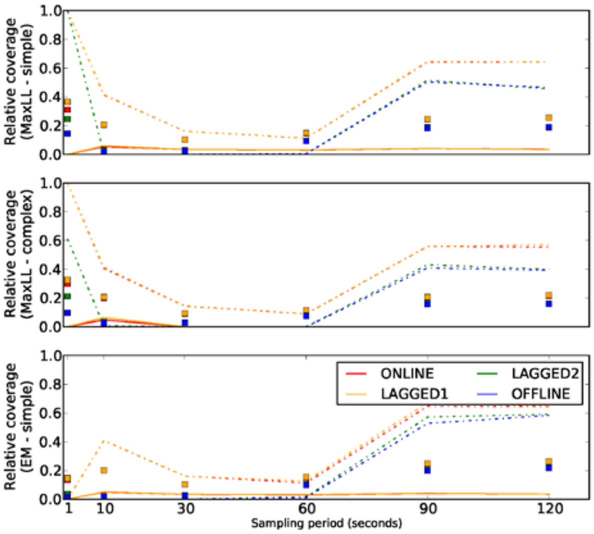


Figure 4-18: Distribution of relative miscoverage of the paths (between 0 and 1, lower is better). The solid line denotes the median, the squares denote the mean and the dashed lines denote the 98% confidence interval. This metric evaluates how much of the true path the most likely path covers , with respect to length (0 if it is completely different, 1 if the two paths overlap completely). Two groups clearly emerge as far as computing strategies are concerned: the online/1-lag group (orange and red curves) and the 2-lag and offline group (green and blue curves). The relative miscoverage for the latter group is so low that more than half of the mass is at the 0 and cannot be seen on the curve. There are still a number of outliers that raise the curve of the last quartile as well as the mean, especially in the lower frequencies. Note that the paths offered by the filter are never dramatically different: at two minute time intervals (for which the paths are 1.7km on average), the returned path spans more than 80% of the true path on average. The use of a more complicated model decreases the mean miscoverage as well as the quartile metrics by more than 15%. Note that there is a large spread of values at high frequency: indeed the metric is based on length covered and at high frequency, the vehicle may be stopped and cover 0 length. This metric is thus less indicative at high frequency.

In the case of the complex model, the weights can provide some insight into the features involved in the decision-making process of the driver. In particular, for extended sampling rates (t=120s), some interesting patterns appear. For example, the drivers do not show a preference between driving through stop signs ($w_3 = -0.24 \pm 0.07$) or through signals ($w_4 = -0.21 \pm 0.11$). However, drivers show a clear preference to turn on the right as opposed to the left, as seen in Figure 4-19. This is may be attributed, in part, to the difficulty in crossing an intersection in the United States.

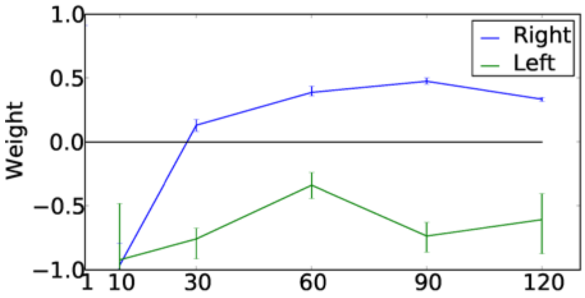


Figure 4-19: Learned weights for left or right turns preferences. The error bars indicate the complete span of values computed for each time (0th and 100th percentile). For small time intervals, any turning gets penalized but rapidly the model learns how to favor paths with right turns against paths with left turns. A positive weight even means that - all other factors being equal! - the driver would prefer turning on the right than going straight.

From a computation perspective, given a driver model, the filtering algorithm can be dramatically improved for about as much computations by using a full backward-forward (smoothing) filter. Smoothing requires backing up an arbitrary sequence of points while 2-lagged smoothing only requires the last two points. For a slightly greater computing cost, the filter can offer a solution with a lag of one or two interval time units that is very close to the full smoothing solution. Fixed-lag smoothing will be the recommended solution for practical applications, as it strikes a good balance of computation costs, accuracy and timeliness of the results.

It should be noted the algorithm continues to provide decent results even when points grow further apart. The errors steadily increase with the sampling rate until the 30 seconds time interval, after which most metrics reach some plateau. This algorithm could be used in tracking solutions to improve the battery life of the device by up to an order of magnitude for GPSs that do not need extensive warm up. In particular, the tracking devices of fleet vehicle are usually designed to emit every minute as the road-level accuracy is not a concern in most cases.

5.3 UNSUPERVISED LEARNING RESULTS

The filter was also trained for the simple and complex models using Dataset 2. This dataset does not include true observations but is two orders of magnitude larger than Dataset 1 for the matching sampling period (1 minute). We report some comparisons between the models previously trained with Dataset 1 (“MaxLL-Simple”, “EM-Simple”, “MaxLL-Complex”) and the same simple and complex models trained on Dataset 2: “EM-Simple large” and “EM-Complex large”. The learning procedure was calibrated using cross-validation and was run in the following way: all unsupervised models were initialized with a hand-tuned heuristic model involving only the standard deviation and the characteristic length (with the weight of all the features set to 0). The Expectation Maximization algorithm was then run for 3 iterations. Inside each EM iteration, the M-step was run with a single Newton-Raphson iteration at each time, using the full gradient and Hessian and a quadratic penalty of 10^{-2}. During the E step, each sweep over the data took 13 hours 400 thousand points on a 32-core Intel Xeon server.

We limit our discussion to the main findings for brevity. The unsupervised training finds some weight values similar to those found with supervised learning. The magnitude of these weights is larger than in the supervised settings. Indeed, during the E step, the algorithm is free to assign any sensible value to the choice of the path. This may lead to a self-reinforcing behavior and the exploration of a bad local minimum.

As Figure 4-22, Figure 4-23, and Figure 4-24 show, a large training dataset puts unsupervised methods on par with supervised methods as far as performance metrics are concerned. Also, the inspection of the parameters learned on this dataset corroborates the finding made earlier. One is tempted to conclude that given enough observations, there no need to collect expensive high-frequency data to train a model.

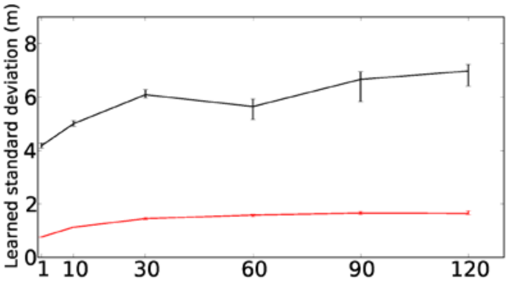


Figure 4-20: Standard deviation learned by the simple models, in the supervised (Maximum Likelihood) setting and the EM setting. The error bars indicate the complete span of values computed for each time. Note that the maximum likelihood estimator rapidly converges toward a fixed value of about 6 meters across any sampling time. The EM procedure also rapidly converges, but it is overconfident and assigns a lower standard deviation overall.

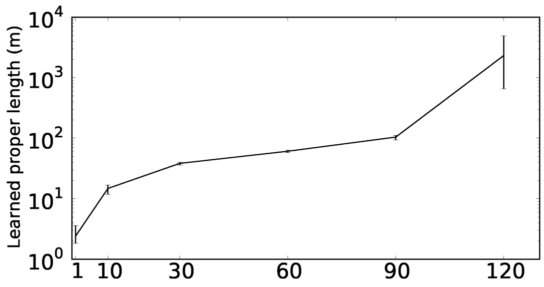


Figure 4-21: Characteristic length learned by the simple models, in the supervised (Maximum Likelihood) setting and the EM setting. As hoped, it roughly corresponds to the expected path length. The error bars indicate the complete span of values computed for each time (0th and 100th percentile). Note how the spread increases for large time intervals. Indeed, vehicles have different travel lengths at such time intervals, ranging from nearly 0 (when waiting at a signal) to more than 3 km (on the highway) and the models struggle to accommodate a single characteristic length. This justifies the use of more complicated models.

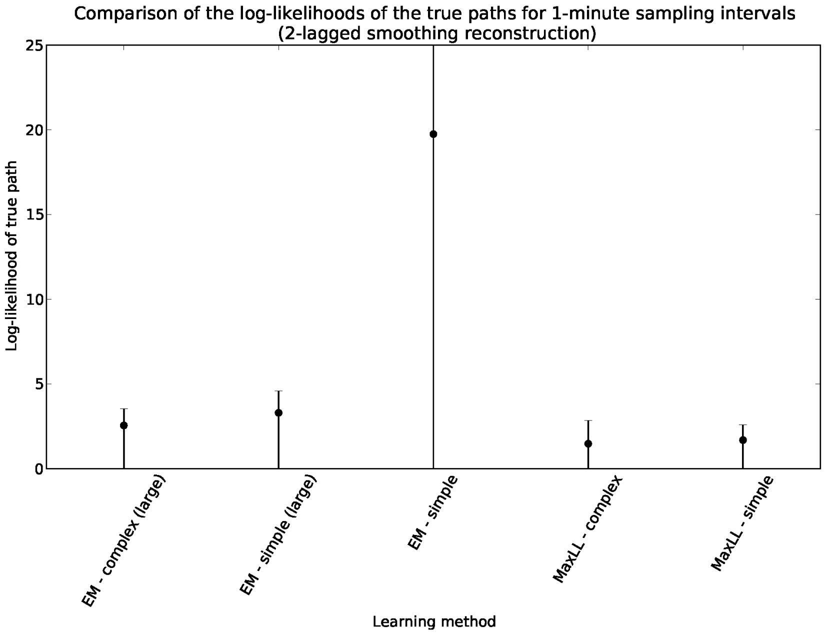


Figure 4-22: Expected likelihood of the true path. The central point is the mean log-likelihood, the error bars indicate the 70% confidence interval. Note that the simple model trained unsupervised with the small dataset has a much larger error, i.e. it assigns low probabilities to the true path. Both unsupervised models tend to express the same behavior but are much more robust.

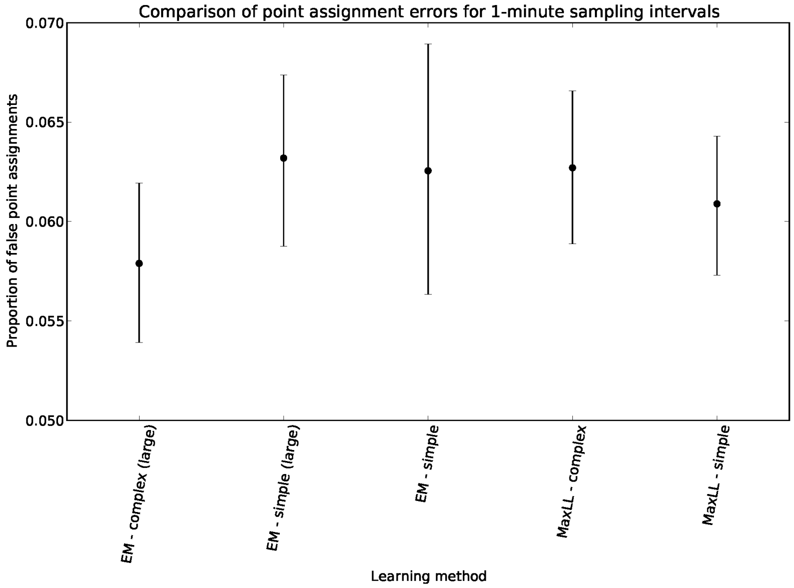


Figure 4-23: Proportion of true points incorrectly identified, for different models evaluated with 1-minute sampling (lower is better). The central point is the mean proportion, the error bars indicate the 70% confidence interval. Unsupervised models are very competitive against supervised models, and the complex unsupervised model slightly outperforms all supervised models.

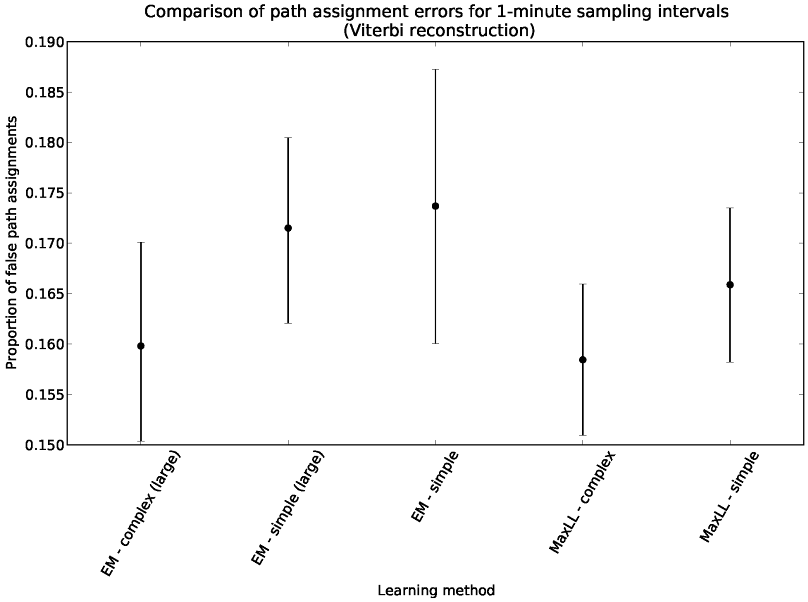


Figure 4-24: Proportion of true paths incorrectly identified, for different models evaluated with 1-minute sampling (lower is better). The central point is the mean proportion, the error bars indicate the 70% confidence interval. The complex unsupervised model is as good as the best supervised model.

5.4 KEY FINDINGS

Our algorithm can reconstruct a sensible approximation of the trajectory followed by the vehicles analyzed, even in complex urban environments. In particular, the following conclusions can be drawn:

- An intuitive deterministic heuristic (“Hard closest point”) dramatically fails for paths at low frequencies, less so for points. It should not be considered for sampling intervals larger than 30 seconds.
- A simple probabilistic heuristic (“closest point”) gives good results for either very low frequencies (2 minutes) or very high frequencies (a few seconds) with more than 75% of paths and 94% points correctly identified. However, the incorrect values are not as close to the true trajectory as they are with more accurate models (“Simple” and “Complex”).
- For the medium range (10 seconds to 90 seconds), trained models (either supervised or unsupervised) have a greatly improved accuracy compared to untrained models, with 80% to 95% of the paths correctly identified by the former.
- For the paths that are incorrectly identified, trained models (“Simple” or “Complex”) provide better results compared to untrained models (the output paths are closer to the true paths, and the uncertainty about which paths may have been taken is much reduced). Furthermore, using a complex model (“Complex”) improves these results even more by a factor of 13-20% on all metrics.
- For filtering strategies: online filtering gives the worst results and its performance is very similar to 1-lagged smoothing. The slower strategies (2-lagged smoothing and offline) outperform the other two by far. Two-lagged smoothing is nearly as good as offline smoothing, except in very high frequencies (less than 2 second sampling) for which smoothing clearly provides better results.
- Using a trained algorithm in a purely unsupervised fashion provides an accuracy as good as when training in a supervised setting - within some limits and assuming enough data is available. The model produced by EM (“EM-Simple”) is equally good in terms of raw performance (path and point misses) but it may be overconfident.
- With more complex models, the filter can be used to infer some interesting patterns about the behavior of the drivers.

6 CONCLUSIONS AND FUTURE WORK

We have presented a novel class of algorithms to track moving vehicles on a road network: the *path inference filter*. This algorithm first projects the raw points onto candidate projections on the road network and then builds candidate trajectories to link these candidate points. An observation model and a driver model are then combined in a Conditional Random Field to find the most probable trajectories.

The algorithm exhibits robustness to noise as well as to the peculiarities of driving in urban road networks. It is competitive over a wide range of sampling rates (1 seconds to 2 minutes) and greatly outperforms intuitive deterministic algorithms. Furthermore, given a set of ground truth data, the filter can be automatically tuned using a fast supervised learning procedure. Alternatively, using enough regular GPS data with no ground truth, it can be trained using unsupervised learning. Experimental results show that the unsupervised learning procedure compares favorably against learning from ground truth data. One may conclude that given enough observations, there is no need to collect expensive high-frequency data to train a model.

This algorithm supports a range of trade-offs between accuracy, timeliness, and computing needs. In its most accurate settings, it extends the current state of the art [46] [38]. This result is supported by the theoretical foundations of Conditional Random Fields. Because no standardized benchmark exists, an open-source implementation of the filter has been released to foster comparison with other methodologies using other datasets [59].

In conjunction with careful engineering, this program can achieve high map-matching throughput. An industrial-strength version in the Scala programming language, the PIF deployed in the *Mobile Millennium* system maps GPS points at a rate of about 400 points per second on a single core for the San Francisco Bay area (several hundreds of thousands of road links) and has been scaled to multicore architecture to achieve an average throughput of several thousand points per second [63].

A number of extensions could be considered to the core framework. In particular, more detailed models of the driver behavior as well as algorithms for automatic feature selection should bring additional improvements in performance. Another line of research is the mapping of very sparse data (sampling intervals longer than two minutes). Although the filter already attempts to consider as few trajectories as possible, more aggressive pruning may be necessary in order to achieve good performance. Finally, the EM procedure presented for automatically tuning the algorithm requires large amounts of data to be effective, and could be tested on larger datasets than what we have presented here.

7 NOTATION

Symbol	Meaning
$\underline{\delta}(x,p)$	Compatibility function between a state x and the start state of a path p
$\bar{\delta}(p,x)$	Compatibility function between an end state x and the end state of a path p
$\epsilon = \sigma^{-2}$	Stacked inverse variance
$\eta = \eta(p\|x)$	Transition model
θ	Stacked vector of parameters
μ	Weight vector
ξ_1, ξ_2	Simple features (path length and distance of a point projection to its GPS coordinate)
π	Probability distribution, the variables are always indicated to disambiguate which variables are involved
$\hat{\pi}$	Probability distribution in the case of a dynamic Bayesian network, the variables are always indicated to disambiguate which variables are involved
$\tilde{\pi}$	Expected plug-in distribution
ς	Set of valid trajectories
σ	Standard deviation
$\tau = x^1 p^1 x^2 \ldots p^{T-1} x^T$	Trajectory of a vehicle
τ^*	Most likely trajectory given a model (ω, η) and a GPS track $\mathrm{g}^{1:\mathrm{T}}$
$\phi(\tau\|g^{1:T})$	Potential, or unnormalized score, of a trajectory
ϕ_i^t	Maximum of all the potentials of the partial trajectories that end in the state $\mathrm{x}_\mathrm{i}^\mathrm{t}$
ϕ^*	Maximum value over all the potentials of the trajectories compatible with $\mathrm{g}^{1:\mathrm{T}}$
$\varphi(p)$	Feature function
$\psi(z^{1:L})$	Generalized potential function
$\omega = \omega(g\|x)$	Observation model
$\Omega(x)$	Prior distribution over the states x
g	GPS coordinate (pair of latitude and longitude)

$(g^t)^{1:T}$	Sequence of all T GPS observations of a GPS track
I^t	Number of projected states of the GPS point at time index t onto the road network
I	Number of mappings of the GPS point onto the road network
J	Number of all candidate trajectories between the mappings x and x'
J^t	Number of all trajectories between the mappings at time t (i.e. x^{t}) and the mappings at time $\mathrm{t}+1$ $\mathrm{x}^{\mathrm{t}+1}$
(l,o)	Location in the road network defined by a pair of a road link l and an offset position o on this link
$L = 2T - 1$	Complete length of a trajectory
$\mathcal{L}$	Expected likelihood
$\mathcal{N} = (\mathcal{V},\mathcal{E})$	Road network, comprising some vertices (nodes) $\mathcal{N}$ and edges (roads) $\mathcal{E}$
$x = (l,o)$	State of the vehicle (typically a location on the road network)
p	Path between one mapping x and one subsequent mapping x'
$\boldsymbol{p} = (p_j)_{1:J}$	Collection of all J candidate trajectories between a set of candidate statesx and a subsequent set x'
$\boldsymbol{p}^t = (p_j^t)_{1:J^t}$	Collection of all J candidate trajectories between the set of candidate states at time t x^{t} and the subsequent set $\mathrm{x}^{\mathrm{t}+1}$
$\bar{q}_i^t$	Probability that the vehicle is in the discrete state $\mathrm{x}_{\mathrm{i}}^{\mathrm{t}}$ at time t given all observations
$\overrightarrow{q}_i^t$	Probability that the vehicle is in the discrete state $\mathrm{x}_{\mathrm{i}}^{\mathrm{t}}$ at time t given all observations up to time t
$\overleftarrow{q}_i^t$	Probability that the vehicle is in the discrete state $\mathrm{x}_{\mathrm{i}}^{\mathrm{t}}$ at time t given all observations after time $\mathrm{t}+1$
$\overline{r}_j^t$	Probability that the vehicle uses the (discrete) path $\mathrm{p}_{\mathrm{j}}^{\mathrm{t}}$ at time t given all observations
$\overrightarrow{r}_j^t$	Probability that the vehicle uses the (discrete) path $\mathrm{p}_{\mathrm{j}}^{\mathrm{t}}$ at time t given all observations up to time t
$\overleftarrow{r}_j^t$	Probability that the vehicle uses the (discrete) path $\mathrm{p}_{\mathrm{j}}^{\mathrm{t}}$ at time t given all observations after time $\mathrm{t}+1$
T	Number of GPS observations for a track
$T^l(z^l)$	Generalized feature vector
Z	Partition function

Chapter 5

Loop and Probe Data: Assimilation and Trade-offs

This chapter presents a case study in the fusion of probe and loop detector data. Based on probe data gathered during the Mobile Century experiment, this case study was the precursor to the data fusion investigation detailed in the final report for Task Order 1, *Pilot Procurement of Third-Party Traffic Data*. Mobile Century was a controlled field experiment in which drivers were hired to follow pre-determined routes and both spatial and temporal data sampling schemes were evaluated. The results of the data fusion algorithm were compared to the drivers' true travel times obtained through license plate re-identification. This precursor informed the design and methodology of the subsequent data fusion study described in the TO1 final report.

1 INTRODUCTION

1.1 OBJECTIVE

Probe data will likely become ubiquitous in the not too distant future, due to the rapid expansion of consumer-generated probe data from cell phones and personal navigation devices. In addition, due to low acquisition and maintenance costs compared to fixed sensors, probe data will become an increasingly important source of real-time traffic data, augmenting fixed sensor data streams when they are available, and adding coverage to areas which are currently unmonitored. As the use of probe data increases, so does the need to understand the benefits and trade-offs between GPS data and conventional data sources. Yet, a complete analysis of the trade-offs between probe data and fixed sensors is difficult, because the value of the data from any sensor (probe, loops, etc.) is dependent on the specifics of the sensing technology, the method used to process the data, and the specific traffic monitoring objective in question.

The goal of this study is to make a first step towards answering the larger question:

> *To what degree can GPS probe data act as a substitute for conventional traffic monitoring technologies such as inductive loop detectors?*

In particular, this study addresses the trade-offs between (i) velocity data collected from GPS smartphones in probe vehicles and (ii) velocity data obtained from inductive loop detectors, for the purpose of computing travel times on a stretch of roadway. It is a case study which uses experimental data collected on one day on a stretch of roadway in the San Francisco Bay Area, obtained as part of a field experiment known as Mobile Century [17]. The data set collected during this experiment and used in this study is unique because of the large number of GPS-equipped probe vehicles representing 2-5% of the traffic flow, the dense coverage of working inductive loop detectors on the experiment site, and the availability of travel time data obtained from video license plate re-identification.[9]

1.2 LITERATURE REVIEW

Several field experiments have been conducted to assess the applicability of cell phone–based measurements for traffic monitoring, including data generated from cell phone towers, which is less accurate than GPS. Bar-Gera [64] compared several months of network data from cell phones to inductive loop detector data on a 14-km freeway segment in Israel and found them to be in good agreement. Liu et al. [65]evaluated a different network-based cell phone system in Minnesota, and

[9] Further research following this study, using commercially-purchased probe data as the data set and Bluetooth-measured travel times as ground truth, is detailed in the final report for Task Order 1, *Pilot Procurement of Third-Party Traffic Data*.

compared travel times to license plate re-identification, and found the system generated results with varying accuracies. A summary of the major network-based cell phone experiments to date can be found in Liu et al. [65].

Several studies have also been conducted to assess the trade-offs between inductive loop detector data and data collected from GPS equipped probe vehicles. In Kwon et al. [66], it is shown that annual estimates of total delay, average duration of congestion, and average spatial extent of congestion can be made with less than 10% error by using either inductive loop detectors placed with half-mile spacing, or by using probe vehicle runs at a rate of about three vehicles an hour. Approximately four to six days of data is needed for reliable estimates from either data source.

The work of Herrera et al. [67] compares a nudging algorithm and a mixture Kalman filtering algorithm to examine how the addition of probe vehicle measurements sampled at a fixed time interval can decrease errors in estimating traffic velocity. On a 0.4 mile stretch of roadway, sampling 5% of the traffic at 150 second intervals with inductive loops at both ends of the domain lead to a 16% improvement over the inductive loop detector data alone. The article also uses the Mobile Century experiment data to compare three scenarios of time-based sampling of probe vehicles, finding that probe data outperforms inductive loop detector data for estimating traffic velocity if a sufficient number of measurements can be obtained from probe vehicles. This work uses the same data set from Mobile Century, but we now consider several thousand scenarios to compare probe data to inductive loop detector data.

1.3 METHODOLOGY

In order to assess the trade-offs between velocity data collected from GPS smartphones and velocity data obtained from inductive loop detectors, it is necessary to define the process by which the data is transformed into an estimate of travel time. In this study, we rely on a velocity estimation algorithm developed at Berkeley as part of the Mobile Millennium project [4]. The algorithm combines velocity measurements from GPS smartphones or inductive loop detectors with a model of traffic evolution, using a technique known as *ensemble Kalman filtering* (EnKF) to produce an improved estimate of the velocity field, from which the travel time is computed. The resulting travel time computed from this process is then compared to the travel times recorded from the license plate re-identification video data.

With the data processing algorithm determined, we create a number of scenarios in which the volume of probe data and number of inductive loop detectors made available to the processing algorithm are adjusted. For example, this allows us to compare the accuracy of computing travel times when all of the probe data is made available, to travel times which are computed when only some of the probe data is available, to travel times when some probe data is available and some inductive loop detector data is available. In this way, we can quantify the trade-offs of various amounts of data from probes and inductive loop detector data in terms of increased or decreased accuracy of the computed travel times.

In order to describe and quantify what probe data is made available to the travel time processing algorithm, we introduce two metrics of importance to probe data: the *penetration rate* and the *sampling strategy*. The penetration rate is defined as the percentage of cars on the roadway reporting probe data compared to the overall traffic flow, including the vehicles which do not send data. In addition to increasing the number of measurements, as the penetration rate increases, the sample of vehicles which generate measurements is more likely to be representative of the total traffic flow.

The sampling strategy refers to how data is collected from the probe vehicles, and can be used to increase or decrease the number of measurements made available for estimating travel times. Two sampling strategies are discussed in this work. The first strategy collects data from probe vehicles at fixed points in space using a new technique known as *Virtual Trip Lines* (VTLs) [5] invented by Nokia. By decreasing spacing between the VTLs, the probe vehicles will send more measurements, with smaller spacing between measurements. The second strategy is a *temporal sampling* strategy in use by many probe data providers. In a time-based sampling strategy, vehicles send measurements at a fixed frequency. By increasing the frequency of measurements, a fixed number of probe vehicles will generate additional measurements.

In order to modify the amount of data obtained from inductive loop detectors, the number of inductive loop detectors which are made available to the processing algorithm is adjusted. Because this is a case study of a real highway, it is not possible to modify the location of the inductive loop detectors. Instead, given a fixed number of inductive loop detectors to include for a given scenario, we select the specific loop detectors such that they achieve as uniform a spacing along the highway as possible.

The remainder of this chapter describes in detail the specific components of this case study. In Section 2, the Mobile Century experiment is described, which serves as the source of data for the case study. The processing algorithm used for velocity estimation is given in Section 3, and the methods for computing travel times from the velocity field are described. In Section 4, the techniques for generating scenarios with various amounts of input data from probe vehicles and inductive loops are presented. In Section 5, the results of nearly 1,700 different scenarios using various amounts of inductive loop detector data and probe data for travel time estimation is presented and summarized. Finally, the discussion in Section 6 concludes the study.

2 MOBILE CENTURY EXPERIMENT

The Mobile Century field experiment serves as a case study to determine the trade-offs between inductive loop detector data and probe data for estimating travel time. In this section, the key features of the Mobile Century experiment are reviewed, and the features of the data collected during the experiment are presented.

The Mobile Century field experiment was a one-day test in the San Francisco Bay Area which collected GPS data from cell phones in probe vehicles, inductive loop detector data, and travel time data from license plate re-identification video data. The experiment took place on February 8th, 2008, and involved 100 probe vehicles equipped with Nokia N95 cell phones which repeatedly drove a stretch of the I-880 freeway near Union City, CA. The GPS cell phones recorded the position and velocity of each vehicle at three-second intervals throughout the day.

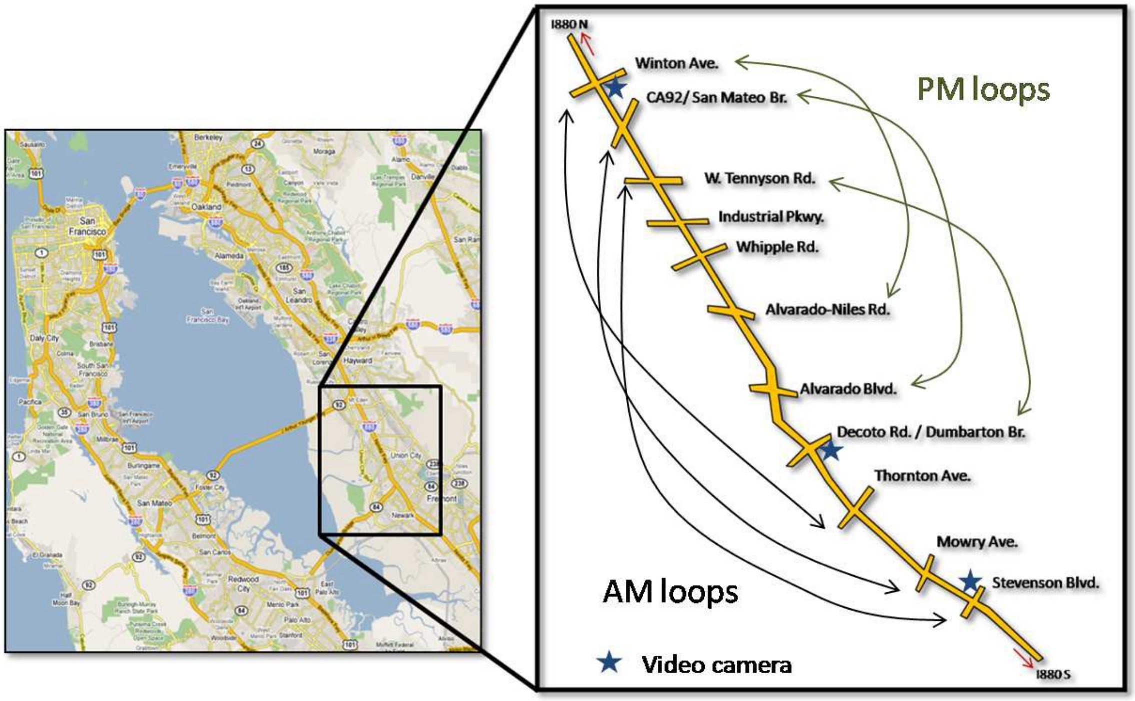
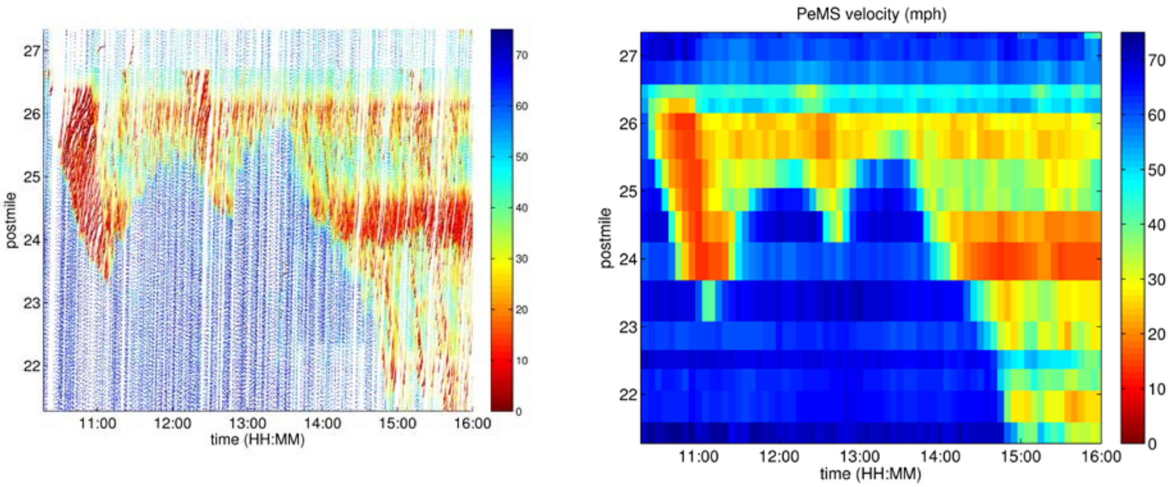


Figure 5-1: Mobile Century experiment site in the San Francisco Bay Area. Vehicles drove a subset of an 11.4 mile stretch of highway I-880.

The experiment location is shown in Figure 5-1, and covers Stevenson Blvd. to the south and Winton Ave. to the north. During the experiment, the 100 vehicles were divided into three groups, and each group covered a different subset of the stretch of freeway for experimental reasons. For example, as shown in Figure 5-1, one third of the vehicles drove north starting at Stevenson Blvd. to W. Tennyson Rd., before exiting, turning around, and driving south from W. Tennyson Rd. to Stevenson Blvd. A second group of vehicles drove in loops covering the freeway between Mowry Ave. and CA 92 / San Mateo Br., while the third group covered the stretch between Thornton Ave. and Winton Ave. In the afternoon, the three groups drove a shorter stretch of roadway labeled “PM loops” in Figure 5-1 to maintain a penetration rate between 2-5% as the traffic volume increased. When the experiment was concluded, it was identified that 77 of the cell phones running the experimental software were able to properly record the probe vehicles’ positions and velocities. Thus, the GPS data recorded from these 77 vehicles is available for input to compute travel times for this study. The data obtained from these vehicles on the northbound stretch of roadway is shown in Figure 5-2a.

The experiment site is also covered with 17 *inductive loop detector* (ILD) stations which feed measurements into the PeMS system [3]. The inductive loop detectors record the sensor occupancy and vehicle counts every 30 seconds, which is processed by a Mobile Millennium filtering algorithm in order to obtain the 30-second average velocity at the sensor. At 5-minute intervals, the PeMS system produces an estimate of the 5-minute average velocity at the sensor, which is shown in Figure 5-2b for the northbound traffic. The locations of the inductive loop detector stations are shown in Figure 5-3.



(a) (b)

Figure 5-2: I-880N experiment data. (a) Vehicle trajectory logs stored locally on the phone. (b) Velocity contour plot from the PeMS system. Color denotes speed in mph. *x*-axis: time of day. *y*-axis: postmile.

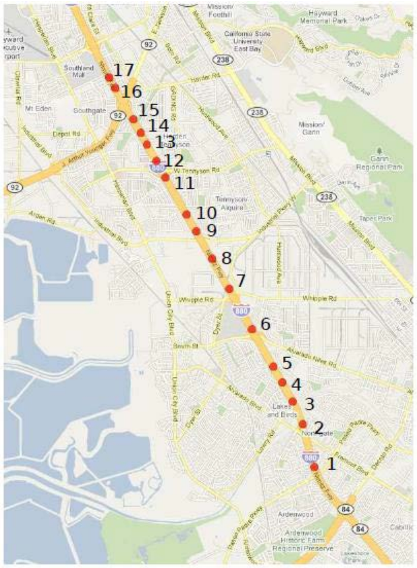


Figure 5-3: Location of the northbound inductive loop detector stations on the area where travel times are to be estimated

Finally, as part of the experiment, high definition video cameras were temporarily installed on three bridges to record license plates of northbound traffic. The locations of the video cameras are shown in Figure 5-1 and are marked by stars. The travel times recorded from the re-identified vehicles traveling northbound between Decoto Rd. to the south and Winton Ave. to the north are shown in Figure 5-4. During the morning, a 5-car accident caused significant delay, and some drivers experienced travel times in excess of 20 minutes around 10:48am. Between 11:50am and 1:20pm, vehicles experience travel times between 8 and 10 minutes on the same stretch of roadway, which steadily increased from 1:20pm to 3:20pm. By 3:20pm, most re-identified drivers experienced heavy evening congestion with travel times increasing to 15–20 minutes.

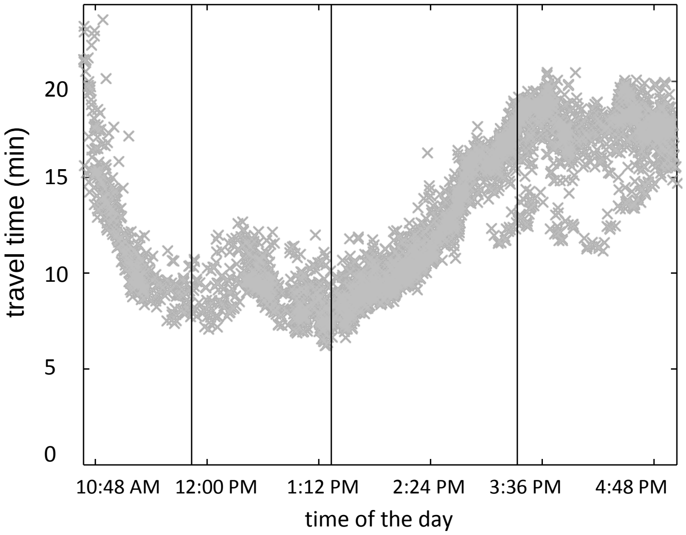


Figure 5-4: Mobile Century northbound travel times divided into four time bins from left to right: morning accident (10:00am-11:50am), free flow (11:50am-1:20pm), congestion building (1:20pm-3:20pm), and full congestion (3:20pm-). The travel times obtained from the license plate re-identification video recordings are marked with crosses.

3 ALGORITHM FOR ESTIMATING TRAVEL TIMES

Given the velocity data obtained from inductive loop detectors and GPS-equipped probe vehicles, a processing algorithm is needed to convert the velocity data into an estimate of travel time. The processing algorithm used in this study is based on a velocity estimation algorithm developed in the Mobile Millennium system. The algorithm takes velocity data from inductive loop detectors and probe vehicles as input, combines the data with a physical model of traffic evolution, and produces an improved estimate of the velocity along the full stretch of roadway. Using this improved estimate of velocity, an estimated travel time is computed using (i) an instantaneous method and (ii) a dynamic method, to compare against the travel times recorded from video data. A brief overview of this process is described in this section.

3.1 MOBILE MILLENNIUM VELOCITY ESTIMATION ALGORITHM

The velocity estimation algorithm developed in the Mobile Millennium system is based on a discretization of a traffic flow model known as the *Lighthill-Whitham-Richards* (LWR) partial differential equation [68] [69] which describes the evolution of traffic density on the highway. In its discrete form, this model is also known as the *Cell Transmission Model* [11, 12]. In order to simplify the velocity estimation problem, this model is transformed into an equivalent velocity evolution equation [70]. More specifically, the highway is discretized into approximately 300-meter-long segments known as cells, and the model produces an estimate of the average velocity in each cell, every 30 seconds.

The model is developed to handle realistic highway properties. For example, the model can take into account splitting and merging of highway segments; changes in the number of lanes, capacity and speed limit; and the flows on the on-ramps and off-ramps.

The model produces a first guess of the current velocity along the roadway, given the immediate history of the velocity on the roadway. In order to improve the mean speed estimate given by the velocity evolution equation, velocity measurement data gathered from probe vehicles and inductive loop detectors is combined with the model. The methodology in which the output of a mathematical model for the physical phenomena is improved with measurements is often called data assimilation or state estimation. The complete mathematical details of (i) the employed traffic velocity evolution equation and (ii) the fusion of velocity measurement data with the evolution equation using *ensemble Kalman filtering* (EnKF) are presented in the article of Work et al. [70].

A few remarks on the performance of the velocity estimation algorithm described above are in order. First, it is noted that the algorithm was designed as part of the Mobile Millennium system, where it is not possible to track probe vehicles for privacy reasons. In other words, it assumed that the probe vehicles send measurements to the system only from pre-selected locations on the highway and, thus, no continuous GPS records from probes are available for the estimation algorithm. Hence, in this study, we also make the assumption that tracking of the vehicles is prohibited. In practice, it is expected that the performance of the estimation algorithm could be improved when tracking of individual probe vehicles is allowed.

Second, the Mobile Millennium algorithm does not directly estimate travel times. Instead, travel times are computed from the estimated velocity field, assuming a vehicle travels at the mean speed reported in each cell. Again, it is expected that the performance of the estimation algorithm could be further improved by directly estimating the travel times in addition to estimating the velocity field. Regardless of the potential for further improvement, preliminary studies of the processing algorithm to compute travel times on the Mobile Century experimental data suggest the approach used in this study works well in practice.

Third, it should be noted that the flow model requires some historical flow information to help calibrate the model. In this study, historical inductive loop detector data from PeMS was used to estimate a constant flow value for the Dumbarton (CA-84) and San Mateo bridge (CA-92) on-ramps feeding traffic to the experiment site. Also, loop detectors outside of the actual experiment site were used to estimate a constant flow value for the north end and south end of the experiment site. Results describing estimates from probe data only still use inductive loop detector data in this way.

Next, the methods for computing the instantaneous and dynamic travel times from an estimated velocity field are described.

3.2 METHODS FOR COMPUTING TRAVEL TIMES

The instantaneous method of computing an estimate of the travel time along a stretch of roadway is as follows. At the time when the instantaneous travel time estimate is produced, the current estimate of the velocity field is recorded. The travel time of a vehicle is simulated, assuming the vehicle travels at the estimated velocity in each cell. The velocity field is assumed to remain constant in time, as the simulated vehicle travels through the velocity field. Thus, the method is an approximation of the true travel time a vehicle would experience, because in practice the velocity field would change as the simulated vehicle completes the trip. The main advantage of the instantaneous travel time is that it does not require a prediction of the evolution of the velocity field, and it should produce accurate travel times when the velocity does not change significantly during the computation.

The dynamic method of computing an estimate of the travel time is obtained similarly, with one important modification. Unlike the instantaneous method which assumes the velocity field does not change during the computation, in the dynamic method, the velocity field is updated during the computation. Thus, the traffic conditions are allowed to evolve while the vehicle travels along the roadway. In practice, the computation of a dynamic travel time has to be done a posteriori, since the method requires knowledge of the speed evolution from the future time steps. Yet, under rapidly changing traffic conditions, the dynamic method will result in more accurate estimates for the travel times actually experienced by the drivers compared to instantaneous travel times.

It is worth noting that the dynamic travel time for individual vehicles is measurable via the license plate recognition performed on the video data collected during the experiment. The differences in accuracy between the two travel time computation methods are further discussed in Section 5.

4 DATA SELECTION

The core topic of this study is to assess the trade-offs between different amounts of probe data and inductive loop detector data for the purpose of estimating travel times. To achieve this, we algorithmically select different subsets of the inductive loop detector data and GPS probe data from the Mobile Century experiment, and use these subsets as inputs to the data processing algorithm described in the previous section. This section describes the various scenarios which modify the type and amount of the data which is made available for estimation, and the selection criteria which are used to generate the scenarios.

4.1 DESCRIPTION OF SCENARIOS TO BE CONSIDERED

The variables for modifying the amount of the input data for computing travel times considered in this study are as follows.

- **Number of inductive loop detectors.** We modify the number of inductive loop detectors which send data into the data processing algorithm.
- **Sampling strategy of probe data.** We consider two sampling strategies for probe vehicles. The first is a space-based sampling strategy using virtual trip lines, where vehicles send measurements at fixed locations on the roadway. The second is a time-based sampling strategy, where vehicles send measurements at a fixed time interval.
- **Number of probe data measurements.** The amount of probe data can be modified in two ways.
    - **Penetration rate.** We modify the number of measurements by increasing or decreasing the penetration rate of the probe vehicles. This is achieved indirectly by changing the number of vehicles from which measurements are collected.
    - **Number of measurements per vehicle**. The second method of modifying the amount of probe data is to change the number of measurements made available from each vehicle. For space-based sampling, this is achieved by changing the number of locations where vehicles report measurements, which is encoded by the number of virtual trip lines. For time-based sampling, the amount of measurements is modified by changing the frequency at which vehicles report measurements.

By modifying the type and amount of inductive loop detector data and probe data through the techniques described above, various scenarios are created to test the impact of the data on computing travel times. In total, the number of scenarios run in this case study is 1,637. They are generated by creating combinations of the following input data:

- Nine different sets of inductive loop detector data, ranging from scenarios with 0 inductive loop detectors to 16 inductive loop detectors, increasing by increments of two detectors.

- Eleven different penetration rates, ranging from scenarios when none of the 2,200 probe vehicle trajectories of Mobile Century are used, to scenarios when 100% of the probe vehicle trajectories are used, increasing by increments of 10%. This corresponds to an average rate of probe vehicles between 27.5 veh/hr and 275 veh/hr.
- Two different sampling strategies. One is a space-based sampling strategy, one is a time-based sampling strategy. Only one sampling strategy is used at a time, meaning for a fixed scenario, all probe data is either collected using a time-based strategy or a space-based strategy, but not both.
- Ten different sets of locations to collect space-based measurements, encoded by scenarios with nine evenly spaced virtual trip lines covering the experiment site (about 8.68 VTL/mi), to scenarios with 99 virtual trip lines (about 0.79 VTL/mi), increasing by increments of 10 virtual trip lines.
- Eight different sampling intervals for each probe vehicle, ranging from 3 second sampling intervals, to 384 second sampling intervals, doubling on each increment.

Thus the 1,637 scenarios are created by instantiating scenarios with all combinations of the 9 sets of inductive loop detector data sets, 11 probe penetration rates, and either one of the 10 space-based strategies or one of the 8 time-based strategies. The scenarios tested are summarized in Table 1. In the remainder of this section, we describe the specific algorithms which select the data for each scenario.

Table 1: A subset of runs used in the study

Run	ILD stations	Probe type	Probe rate (veh/hr)	VTL/mi	Sampling interval(s)
1	1	No Probe	-	-	-
2	2	No Probe	-	-	-
3	3	No Probe	-	-	-
.	.	.	.	.	.
101	0	Space	27.5	0.79	-
102	0	Space	27.5	1.67	-
103	0	Space	27.5	2.54	-
104	0	Space	27.5	3.42	-
.	.	.	.	.	.
478	6	Space	220	6.93	-
479	6	Space	220	7.81	-
480	6	Space	220	8.68	-
481	6	Space	247.5	0.79	-
482	6	Space	247.5	1.67	-
.	.	.	.	.	.
918	16	Space	55	6.93	-
919	16	Space	55	7.81	-
920	16	Space	55	8.68	-
.	.	.	.	.	.
1001	0	Time	27.5	-	3
1002	0	Time	27.5	-	6
1003	0	Time	27.5	-	12
.	.	.	.	.	.
1493	12	Time	55	-	48
1494	12	Time	55	-	96
1495	12	Time	55	-	192
				.	.
1719	16	Time	275	-	192
1720	16	Time	275	-	384

4.2 ALGORITHMS FOR DATA SELECTION

The remainder of this section discusses:

- inductive loop detector sensor selection algorithm
- penetration rate of the probe vehicles used in this study
- method for placing the virtual trip lines

4.2.1 SELECTION OF INDUCTIVE LOOP DETECTOR DATA

In order to modify the number of inductive loop detector stations which are made available for computing travel times, a simple selection criterion is developed for determining the sensors which are made available for estimation. Specifically, given a fixed number of inductive loop detectors to include, the inductive loop detector stations are selected in order to minimize the variance of the distance between consecutive sensors. This allows us to pick the sensors such that they are as uniformly distributed across the experiment site as possible, given the fixed locations of the candidate inductive loop detector stations. We describe this criterion in detail in this section.

We consider a stretch of highway of length *L*, starting at *x* = 0 and ending at *x* = *L*, with *n* inductive loop detector stations located at $x_1, x_2, \cdots, x_n$, as shown in Figure 5-5:

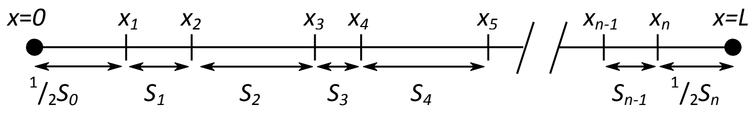


Figure 5-5: Highway segment of length *L*, with *n* inductive loop detector stations located at x_i

Let S_i denote the spacing between sensor *i* and *i* + 1. In order to treat the boundaries without explicit knowledge of sensors outside the domain $x \in [0, L]$, it is assumed only half of the first inter-station spacing S_0 and the last inter-station spacing S_n is in the domain of interest. The weighted average spacing between the sensors is given by:

$$\bar{S} = \frac{\frac{1}{2}S_0 + S_1 + S_2 \ldots + S_{n-1} + \frac{1}{2}S_n}{n} = \frac{L}{n}$$

where the first and last spacings have a weight $\frac{1}{2}$, since only half of these spacings actually lie within the [0, *L*] domain. Note that the average spacing is independent of the specific locations of the sensors x_i and consequently cannot be used as a selection criterion.

Instead, we use a selection criterion which explicitly takes the uniformity of the inter-sensor distances S_i into account. This is achieved by minimizing the variance σ^2 of the inter-station spacings S_k, $0 \le k \le n$, given by:

$$\sigma^2 = \frac{1}{2n}(S_0 - \bar{S})^2 + \frac{1}{n}\sum_{1 \le i \le n}(S_i - \bar{S})^2 + \frac{1}{2n}(S_n - \bar{S})^2$$

Again, the first and last spacings have a weight $\frac{1}{2}$, since only half of these spacings actually lie within the [0, *L*] domain.

In practice, rather than minimizing the variance σ^2, it is convenient to minimize an equivalent loop detector placement criterion denoted $\tilde{S}$:

$$\tilde{S}(x_1, x_2, \ldots, x_n) = 2\sqrt{\frac{x_1^2}{2n} + \frac{(L - x_n)^2}{2n} + \sum_{1 \le k < n}\frac{\left(\frac{x_{k+1} - x_k}{2}\right)^2}{n}}$$

which is equal to σ^2plus a constant offset. The best set of *k* inductive loop detector stations is then given by:

$$U^*(k) = \operatorname{argmin}\{\tilde{S}(U) \quad | \quad U \subset \{x_1, x_2, ..., x_n\} \text{and } |U| = k\}$$

where $|U|$ represents the number of elements in the set U . The resulting selections for the inductive loop detector stations are shown in Table 2.

Table 2: Inductive loop detector selection results. Given a number *k*, the selection algorithm returns the set *U∗(k)* of *k* inductive loop detector stations which minimizes the inductive loop detector placement index $\tilde{S}$ *(U ∗(k))*. The labels in *U ∗(k)* correspond to the labels of the inductive loop detectors in Figure 5-3.

k	$\tilde{S}$ $(U^*(k))$ (mi)	$U^*(k)$
0	∞	∅
1	6.51	{ 8 }
2	3.25	{ 4, 11 }
3	2.17	{ 3, 8, 14 }
4	1.63	{ 2, 6, 9, 15 }
5	1.33	{ 2, 6, 8, 11, 16 }
6	1.11	{ 1, 3, 6, 8, 11, 16 }
7	0.95	{ 1, 3, 6, 8, 10, 13, 16 }
8	0.83	{ 1, 3, 6, 7, 9, 11, 14, 16 }
9	0.73	{ 1, 2, 4, 6, 7, 9, 11, 14, 16 }
10	0.66	{ 1, 2, 4, 6, 7, 8, 10, 11, 14, 16 }
11	0.60	{ 1, 2, 3, 5, 6, 7, 8, 10, 11, 14, 16 }
12	0.55	{ 1, 2, 3, 5, 6, 7, 8, 10, 11, 13, 15, 16 }
13	0.51	{ 1, 2, 3, 5, 6, 7, 8, 9, 10, 11, 13, 15, 16 }
14	0.48	{ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 13, 15, 16 }
15	0.46	{ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 13, 15, 16, 17 }
16	0.43	{ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 15, 16, 17 }
17	0.41	{ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17 }

In the case when the chosen inductive loop detector stations are uniformly spaced within the section of interest, the criterion $\tilde{S}$ is equal to the average spacing $\bar{S}$. Because $\bar{S}$ serves as a lower bound for $\tilde{S}$, the difference between $\tilde{S}$ and $\bar{S}$ indicates the degree of non-uniformity of the sensor spacings caused by the fixed set from which the sensors are selected. Figure 5-6 shows the difference between the inductive loop detector placement criterion $\tilde{S}\,(U^*(k))$ and its lower bound; the average inductive loop detector spacing $\bar{S}(k)$, is small, indicating that the sensors are uniformly spaced:

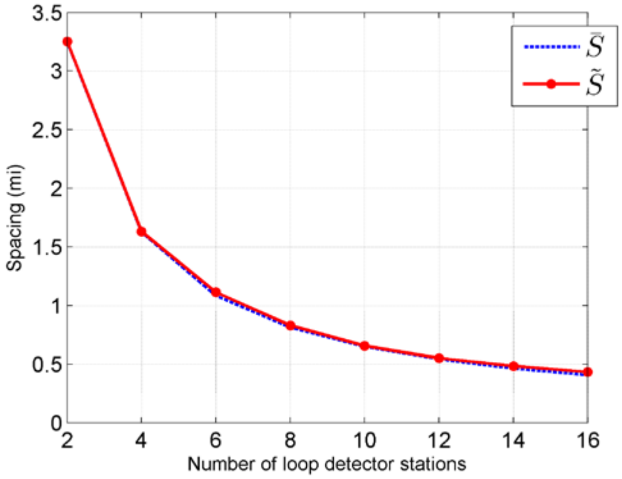


Figure 5-6: Results from the inductive loop detector selection algorithm.

4.2.2 PENETRATION RATE FOR PROBE DATA

During the Mobile Century experiment, GPS data was successfully obtained from 77 GPS-equipped probe vehicles, yielding a total of 2,200 vehicle trajectories on I-880 in the northbound direction. Each vehicle trajectory consists of the estimated vehicle position and velocity recorded at three-second intervals. The trajectory data is filtered to guarantee physically meaningful acceleration and velocity data based on assumed vehicle dynamics. Less than 0.01% of the data were identified as outliers, which were replaced by an interpolated value.

Throughout the day, these 2,200 vehicle trajectories represent a 5-minute penetration rate which ranges between approximately 0% and 5%, depending on the time and location at which the penetration rate is estimated [17]. Figure 5-7 shows the 20-minute penetration rate estimated at the center of the Mobile Century experiment site (near inductive loop detector station 9 in Figure 5-3), which varies between 1.5% and 3%:

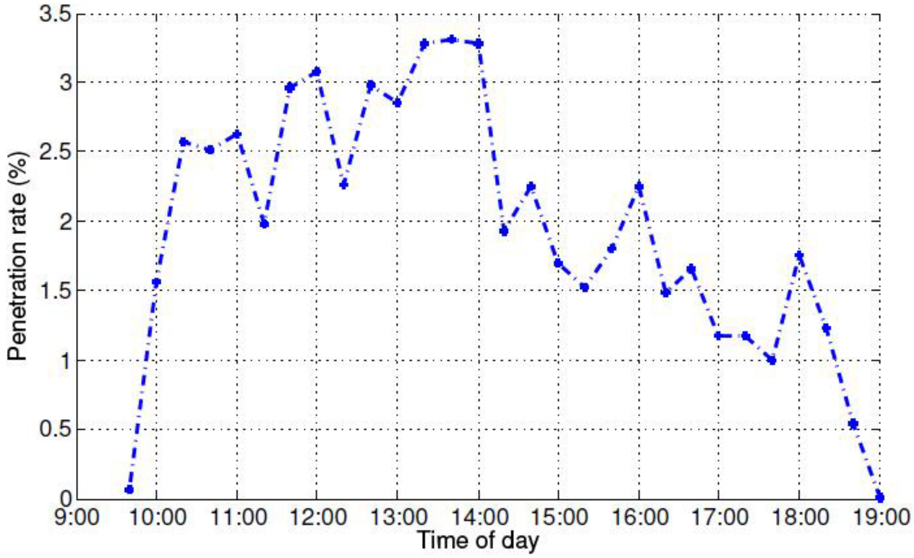


Figure 5-7: 20-minute average penetration rate in the center of the Mobile Century experiment site on I-880 northbound

A few remarks about probe penetration rates are in order, before criteria to modify the penetration rate are discussed. In general, the penetration rate is difficult to determine for probe vehicles specifically because it depends on (i) the number of equipped probe vehicles, (ii) the total traffic flow, and (iii) the evolution of the traffic flow in space and time. Typically, only the total number of equipped probe vehicles is known to probe data providers. Similarly, the total traffic flow can only be estimated from counts recorded by inductive loop detectors at predefined locations. Finally, because the evolution of the traffic flow is not under the control of the probe vehicles, it is nearly impossible to a priori specify a penetration rate which is both uniform in space and uniform in time.

Because of the inherent difficulty in specifying the penetration rate a priori, we instead elect to directly modify the number of equipped probe vehicles as a proxy for modifying the penetration rate. The number of equipped probe vehicles in this study varies from 0% of the 2,200 vehicle trajectories to 100% of the 2,200 vehicle trajectories, increasing by increments of 10%. Over the eight-hour experiment, this corresponds to an average rate of probe vehicles between 27.5 veh/hr and 275 veh/hr. When a subset of the vehicle trajectories is required, the subset is determined by selecting the trajectories at random. For example, 50% of the collected probe data corresponds to exactly 1,100 vehicle trajectories (137.5 veh/hr), which are selected at random before the simulation. The corresponding 20-minute penetration rate at the center of the experiment site would then be half of the penetration rate shown in Figure 5-7, but only in the expected sense, since the trajectories are selected at random.

4.2.3 SPACE–BASED SAMPLING

In order to modify the number of measurements used from each probe vehicle trajectory under spatial sampling, the number of locations where measurements are collected is modified. The locations where measurements are obtained are encoded through the placement of virtual trip lines (VTLs), which can be viewed as virtual geographic markers which trigger vehicles to send measurements when the vehicle trajectory intersects the VTL. A complete description of the VTL sampling strategy is described in detail in Hoh et al. [5].

Because the VTLs are virtual, it is possible to place them anywhere on the experiment site. The determination of optimal VTL placement is complex, so instead we elect to place the VTLs uniformly across the experiment site. The number of VTLs n_{VTL} tested in our scenarios varies from nine VTLs to 99 VTLs, increasing by increments of 10 VTLs. This corresponds to an average spacing between 0.72 to 7.1 VTL/mi. Note the number of VTLs used on the experiment site is significantly higher than the number of inductive loop detector stations. This is possible because unlike inductive loop detector stations, the marginal cost of virtual trip lines is small.

4.2.4 TIME–BASED SAMPLING

To modify the number of measurements used from each probe vehicle trajectory under temporal sampling, the sampling intervals at which the data are reported are modified. Because the vehicle trajectories are sampled at 3-second intervals, all sample intervals are multiples of 3 seconds, ranging from 3 to 384 seconds. To simplify computations and avoid unnecessary interpolations on the data, only time samples equal to $2^k \cdot 3$ seconds were simulated, with $0 \leq k \leq 7$. Thus, for a given trajectory, simulating a 6-second time sampling is achieved by using every other data point, while simulating a 12-second time sampling uses every fourth data point.

5 RESULTS AND DISCUSSION

In this section, we present the results of 1,637 runs with varying amounts of probe and inductive loop detector data. We also vary the type of travel time computed (instantaneous or dynamic). The quantification of error is described in Section 5.1, and the computational results are shown in Section 5.2.

5.1 ERROR QUANTIFICATION

Since validation data is available for dynamic travel times (see Section 3.2), an error metric is used to compare the velocity estimation algorithm output that has been converted to travel times with the travel time measured from video recordings. By using the travel time error as a performance metric, estimation algorithm results can be compared with the results obtained when using different types and quantities of the input data.

Since the license plate re-identification data provides a distribution of individual vehicle travel times (see Figure 5-4), we define the true travel time as a one-minute moving average of the recorded travel times. Figure 5-4 also shows the division of the experiment into four time periods that represent the different phases of the traffic during the experiment. These periods are (i) the *morning accident*, where travel times are decreasing as an incident clears, (ii) a *free flow* period during the middle of the day when travel times are low, (iii) a *congestion building* period before the evening rush hours, and (iv) *full congestion* during the evening rush hours. Because of the different traffic conditions present in these time intervals, in addition to computing the error across the full day, the error is also computed for each time interval.

The error is quantified as follows. Let *n* be the number of estimates given in a period for which the error is to be computed, with each estimate indexed by *i*. The travel time error is computed as follows. Let $T_v(i)$ be the mean travel time from the video data at time *i*, $T_{inst}(i)$ be the estimated mean travel time computed with the instantaneous method at time *i*, and $T_{dyn}(i)$ be the estimated mean travel time computed with the dynamic method at time *i*, as in Section 3.2. The *mean absolute percent error* (MAPE) for the travel time computed with the instantaneous method is:

$$\varepsilon_{\text{inst,MAPE}} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{T_v(i) - T_{\text{inst}}(i)}{T_v(i)}\right|$$

while the MAPE for the travel time computed with the dynamic method is:

$$\varepsilon_{\text{dyn,MAPE}} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{T_v(i) - T_{\text{dyn}}(i)}{T_v(i)}\right|$$

The MAPE is used in this study to aggregate the error in the model estimates over a given time period in order to produce a single value for the error with given input data and time period. Similarly a non-aggregated error would result in a representation of the error that is dependent on the time dimension of the results. Also, note that the spatial dimension of the error vanishes when the travel times are produced from the model estimated mean speed fields.

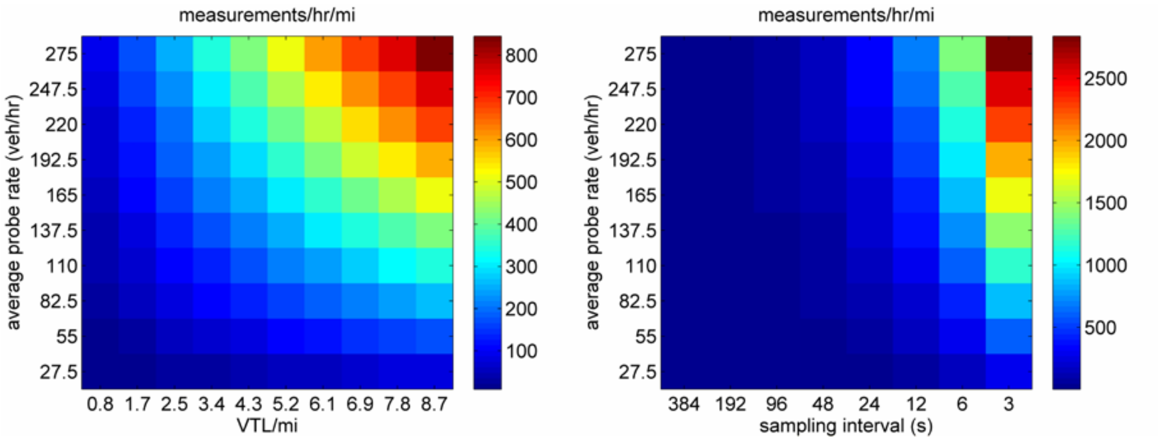
5.2 COMPUTATIONAL RESULTS

5.2.1 IMPLEMENTATION

The estimates were computed using the existing Mobile Millennium highway model. The model was run 1,637 times with various data inputs. Given the computationally intensive nature of the algorithms utilized, each run (an eight-hour simulation) took approximately 20 minutes to complete. Each run consisted of the computation of the mean speed field evolution and computation of both instantaneous and dynamic travel time every 30 seconds, from 10am to 6pm on the day of the experiment.

The runs took 630 CPU-hours, and were distributed on 8 servers equipped with 2.2 GHz dual core AMD Opteron CPUs and 8 GB RAM, which reduced the computation time to 48 hours. The travel time data was then extracted manually from the servers. Finally, this data was analyzed in Matlab.

To give an idea of the input data variability between the runs, a representative subset of the input data combinations is shown in Table 1. The table shows the travel time type, number of inductive loop detector stations, average rate of probe vehicles, and number of VTLs per mile. These parameters are presented as a function of the run number. The number of probe vehicle measurements used in each simulation is shown in Figure 5-8:



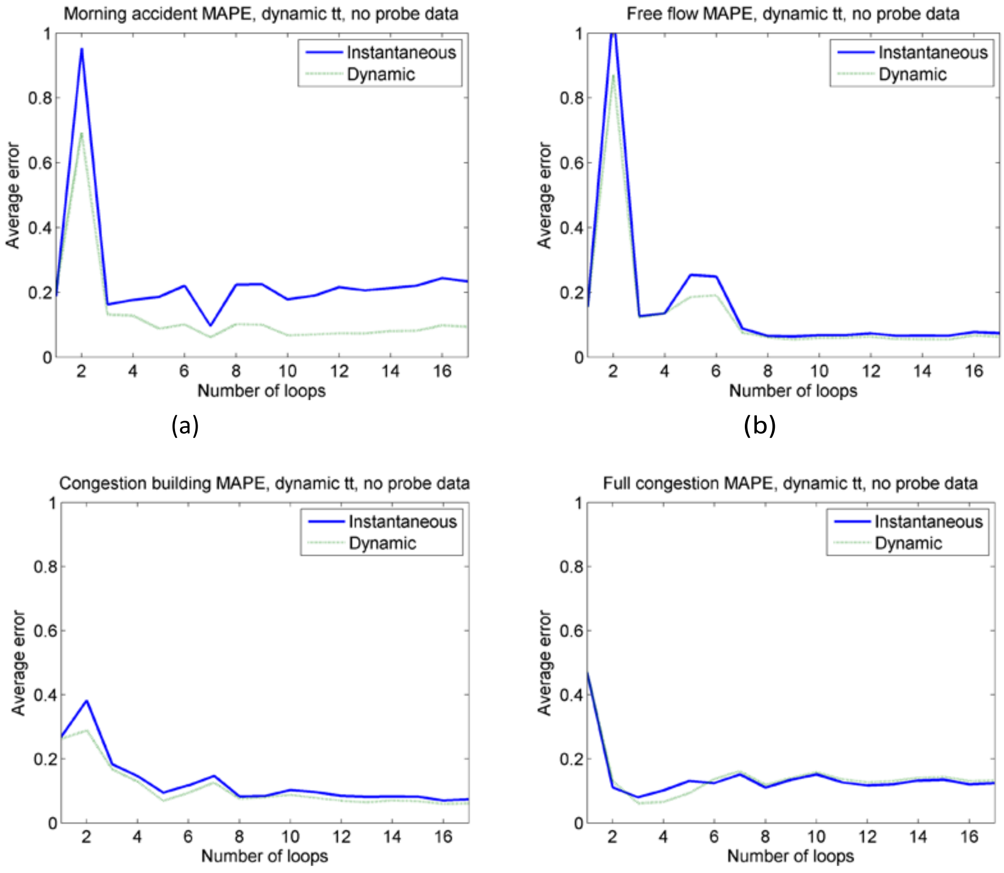
(a) (b)

Figure 5-8: Number of probe vehicle measurements used in the simulations when using (a) VTL data and (b) time-sampled data. See also Table 3 and Table 4.

5.2.2 USING ONLY INDUCTIVE LOOP DETECTOR DATA IN THE MODEL

The first analysis of the traffic estimates is based on the results obtained when using inductive loop detector data as the only input to the model. These results give us a baseline for the comparison between probe and loop detector data. A total number of 17 runs were conducted based only on the inductive loop detector data by varying the number of sensors according to the selection algorithm in Section 4.2.1. Both instantaneous and dynamic travel times were computed. The labels of the inductive loop detector stations used in the estimation are presented in Table 2 as a function of the number of stations selected (see also Figure 5-3).

The results of these runs are shown Figure 5-9. The subfigures show the estimation error broken down by time of the day, as defined in Figure 5-4. During the morning accident (Figure 5-9a), the dynamic travel times converge to estimates with 7% error, while the instantaneous estimates remain above 20% error. The instantaneous and dynamic estimates have between 6% and 7% error during the free flow and congestion-building periods (Figure 5-9b and Figure 5-9c), and 13% error during the full congestion period (Figure 5-9d), with the instantaneous and dynamic estimates performing similarly.



(a)

(b)

(c)

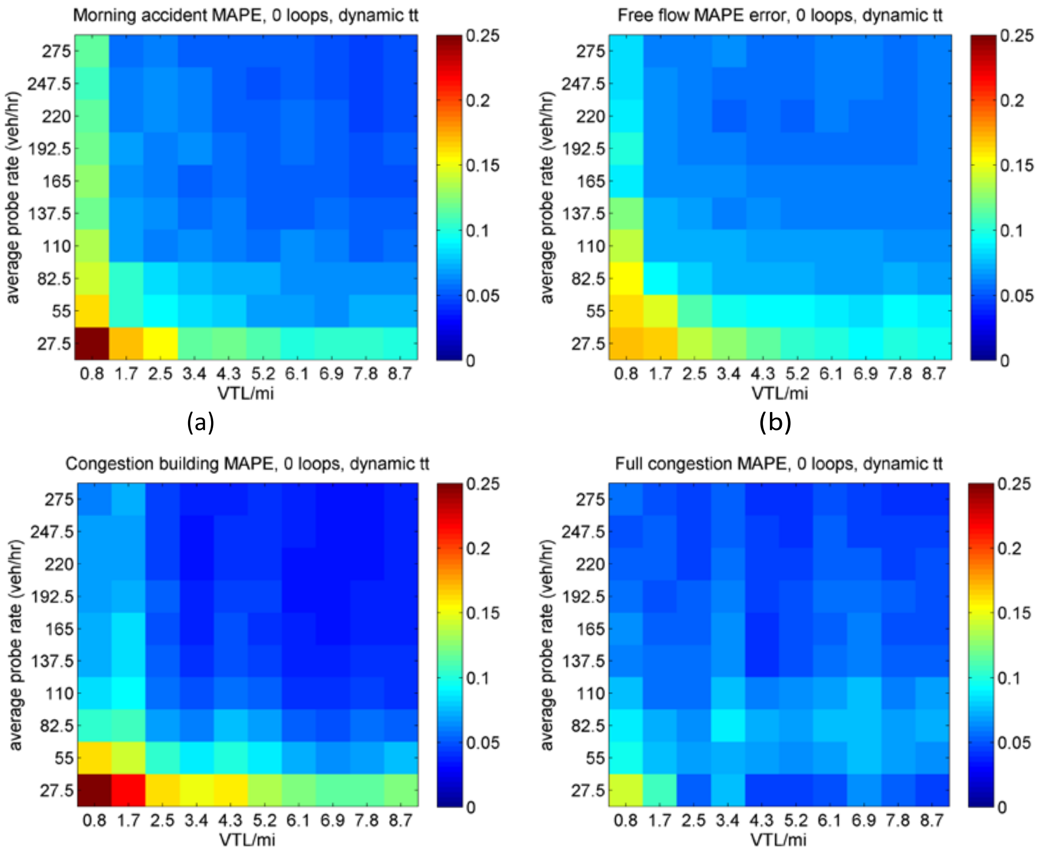
(d)

Figure 5-9: MAPE computed using inductive loop detector data only, no probe data. Travel time is computed using the dynamic method (green dash) and instantaneous method (solid blue). x-axis: number of inductive loop detector sensors, y-axis: MAPE (a) morning incident; (b) free flow; (c) afternoon as congestion increases; (d) evening congestion.

The number of inductive loop detector stations used tends to have a positive impact on the quality of the estimate when less than eight inductive loop detector stations are used. Note that the curve is not monotonic decreasing. This is because when only a few sensors are deployed, the error becomes highly dependent on the placement of the sensors. It is expected that an optimal sensor placement algorithm would reduce the error. The threshold of eight inductive loop detector stations corresponds to the inductive loop detector placement index $\widetilde{S}\,(U^*(8)) = 0.83$ mi (Table 2). However, using data from more than eight inductive loop detector stations does not improve the quality of the estimates. If fewer than three inductive loop detectors are used, the estimation error is unacceptably high, at some points reaching as high as 100% error.

5.2.3 USING ONLY VTL DATA IN THE MODEL

The second part of the analysis consists of the travel time estimates obtained when using VTL data only. The changing parameters of the input data are the number of VTLs deployed on the experiment site and the rate of the probe vehicles used to produce speed measurements at the locations of VTLs (see Table 1). The estimation errors of the travel times obtained with the dynamic method are shown in Figure 5-10:



(a)

(b)

(c)

(d)

Figure 5-10: MAPE contours computed using VTL data only, no inductive loop detector sensors. Travel time is computed using the dynamic method. x-axis: number of VTLs, y-axis: average probe data rate; (a) morning accident; (b) free flow; (c) congestion building; (d) full congestion. Color scale limited to 0.25. See also Table 5 to Table 8.

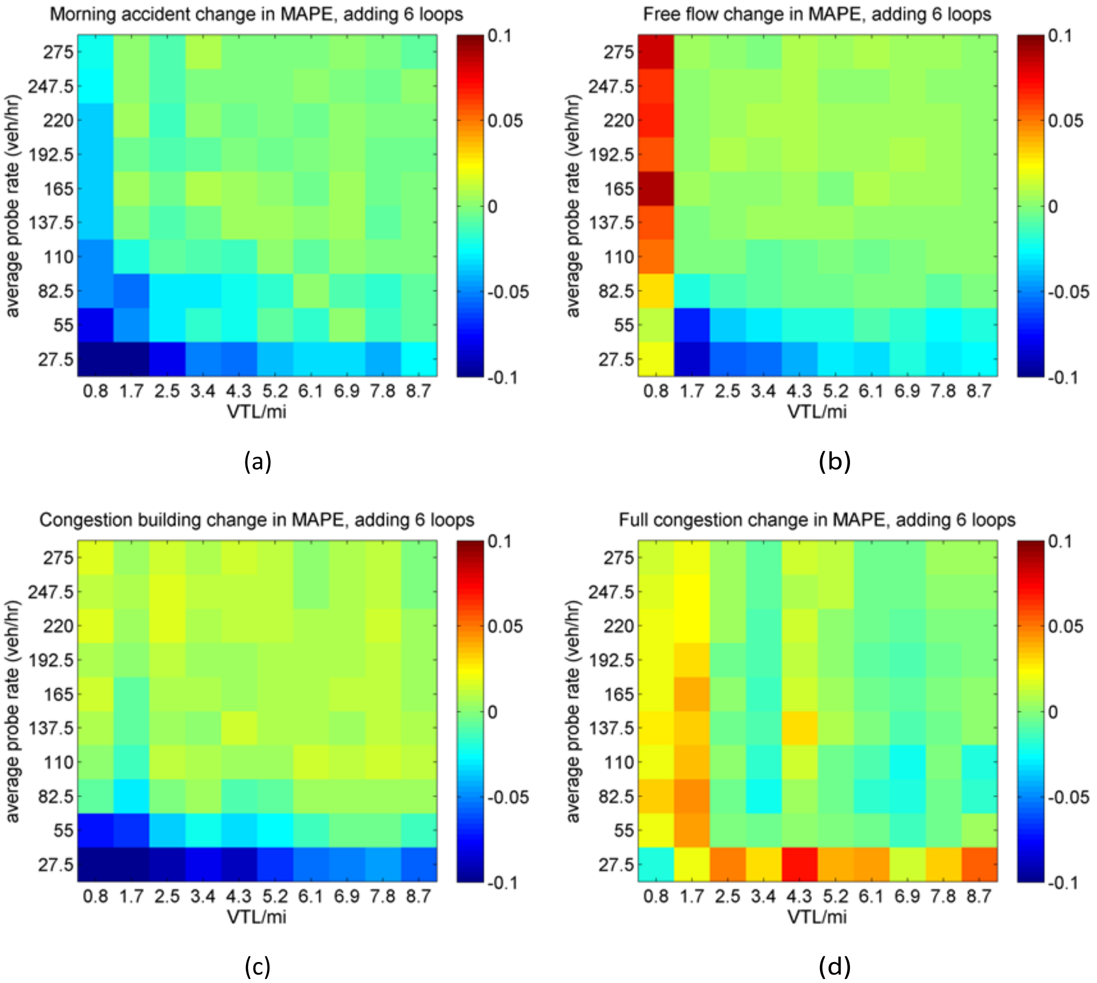
In each of the time periods, estimates of the travel time can be achieved with less than 6% MAPE, with sufficient probe vehicles and virtual trip lines. However, when more than 137.5 veh/hr are used with more than 2.54 VTL/mi, only small improvements in the accuracy of the estimates can be achieved. When compared with inductive loop detectors, using 137.5 veh/hr and 2.54 VTL/mi performs as well as the estimates using more than eight inductive loop detector stations during the morning accident, free flow, and congestion-building periods, but has less than half the error of inductive loops during the full congestion period. When 137.5 veh/hr are used, the overall probe penetration rate as a percentage of the total number of vehicles is one-half the values shown in Figure 5-7.

5.2.4 MIXING VTL AND LOOP DETECTOR DATA

The dynamic travel time estimation errors using both VTL and loop detector data simultaneously is assessed in Figure 5-11, where the change in the dynamic travel time MAPE due to the addition of data from six inductive loop detectors is computed. The results shown are a representative subset of all the runs performed when mixing the two data types.

At low probe data rates during the morning accident, free flow, and congestion-building periods, adding inductive loop detector data increases the accuracy of the dynamic travel time estimates. For example, during the morning accident (Figure 5-11a), with a probe rate of 27.5 veh/hr and a VTL spacing of 0.79 VTL/mi, adding inductive loop detector data reduced the error from 29% to 8%. During the full congestion period, the dynamic travel time estimate accuracy decreased when inductive loop detector data was added at low probe rates (27.5 veh/hr). This is likely due to the fact that the estimates based on virtual trip line data only were unusually accurate, even performing better than simulations with more probe vehicles.

At higher penetration rates (above 137.5 veh/hr) adding data from the six inductive loops has negligible effect, increasing or decreasing the accuracy only slightly. The exception is during the free flow period, when the MAPE increased (between 0.05 and 0.08) even at high probe rates, when 0.79 VTL/mi were used. The errors in the free flow period are magnified due to the small base travel time, which is under 10 minutes, and it is in fact not constant during the period (see Figure 5-4). Moreover, it is clear from Figure 5-2 that there is an area of heavy congestion around postmile 26 even during the free flow period, which is difficult to capture correctly with sparse sampling.



(a)

(b)

(c)

(d)

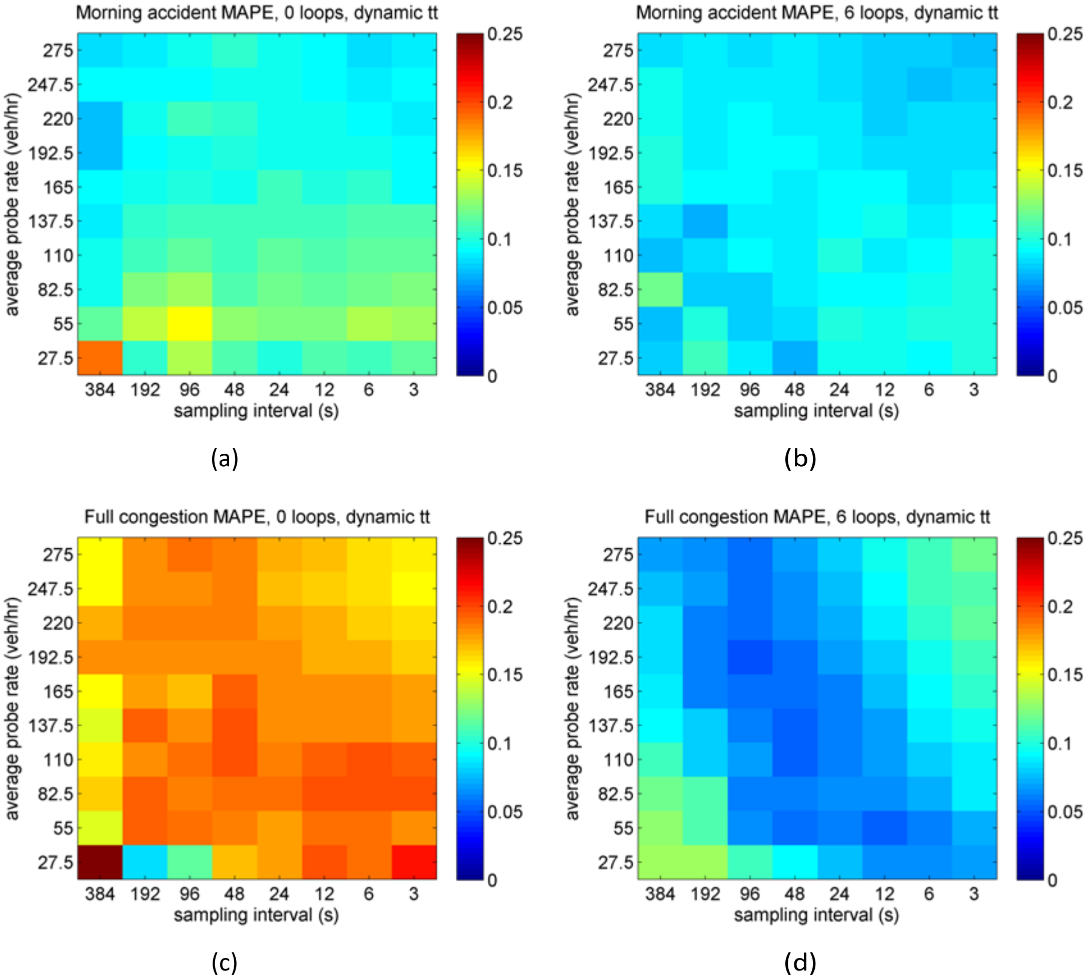
Figure 5-11: Change in MAPE contours when adding six inductive loop detectors to VTL data. x-axis: number of VTLs, y-axis: average probe data rate; (a) morning accident; (b) free flow; (c) congestion building; (d) full congestion. Color scale limited to ±0.1. See also Table 9 to Table 12.

5.2.5 MIXING TIME-SAMPLED PROBE DATA AND LOOP DETECTOR DATA

Figure 5-12 shows the average error estimate using time-sampled probe data. Although the data is collected from the same vehicles as the VTL data, the estimates obtained using time-sampled data are different from the estimates using VTL data. The differences can be attributed to the number of measurements used in the estimation (Figure 5-8), the location of where the measurements are collected, and biases associated with spatial sampling of average velocity data. The Mobile Millennium velocity estimation algorithm is calibrated for accepting virtual trip line and inductive loop detector data, and thus the estimates in this section could be improved with additional calibration for time-sampled data.

During the morning accident (Figure 5-12a), when the probe rate is 55 veh/hr and vehicles are sampled every 32 seconds, the estimation error is 14% when no inductive loop detectors are used. Adding data from six inductive loop detectors (Figure 5-12b) reduces the estimation error to 10%. When enough probe data is added, an error of under 10% is achievable using time-sampled data only.

During the full congestion (Figure 5-12c), the estimation error remains between 15–20% regardless of the amount of probe data used. Adding six loop detectors (Figure 5-12d) improves the quality of the estimates, reducing the error to 5–10%.



(a)

(b)

(c)

(d)

Figure 5-12: MAPE contours computed using time-sampled probe data and inductive loop detector sensor data. Travel time is computed using the dynamic method. x-axis: vehicle sampling interval in seconds, y-axis: average probe data rate; (a) morning accident, no inductive loop detector data; (b) morning accident, data from six inductive loop detector stations; (c) full congestion, no inductive loop detector data; (d) full congestion, data from six inductive loop detector sensors. Color scale limited to 0.25.

From the results not shown here, it is shown that the time-sampled data tends to converge to either similar or a higher travel time error compared to estimates produced using VTL data when the same number of measurements are used in each simulation. Again, this is expected due to the current calibration of the Mobile Millennium system. In conclusion, regardless of the offset in the results obtained using the time-sampled data, all the results suggest that when using either type of probe data with low penetration rates, adding loop detector data usually helps in travel time estimation, although there are some exceptions. With higher penetration rates the impact is not as significant.

5.2.6 USING INSTANTANEOUS TRAVEL TIME AS AN ESTIMATE FOR DYNAMIC TRAVEL TIME

Figure 5-13 shows a comparison of the estimation errors when using instantaneous and dynamic travel times for the morning accident period. Instantaneous travel times can be determined at any time on any route using the speed estimates, and used as a proxy for dynamic travel times. As was shown for the inductive loop detector data in Figure 5-9b and Figure 5-9d, instantaneous and dynamic travel time estimates are very similar when traffic conditions change sufficiently slowly. The same holds when estimating travel times from probe data.

By looking at the instantaneous travel time errors in Figure 5-13a, an interesting result can be seen. The results suggest that adding more probe data results in an increased travel time estimation error. However, this result is expected, and can be explained by focusing on the scenarios (in Figure 5-13a) in which the penetration rate of the probe vehicles is low and no loop detectors are used. Here, the instantaneous travel time estimate performs well, and may seem like a valid estimate of the true travel time during the incident. However, this gives a misleading indication of the quality of these travel time estimates. The good performance of the instantaneous estimate is caused by the fact that the current state of the traffic (speed field) is very poorly captured in the underlying scenario, and the speed of the traffic is heavily overestimated. This causes the instantaneous travel time estimate to perform as a good predictor of the future traffic conditions, namely, as a predictor of the clearing incident. When the number of probe measurements increases, the speed field estimate is captured more accurately, and the increased error in the travel time estimate is caused by the instantaneous approximation.

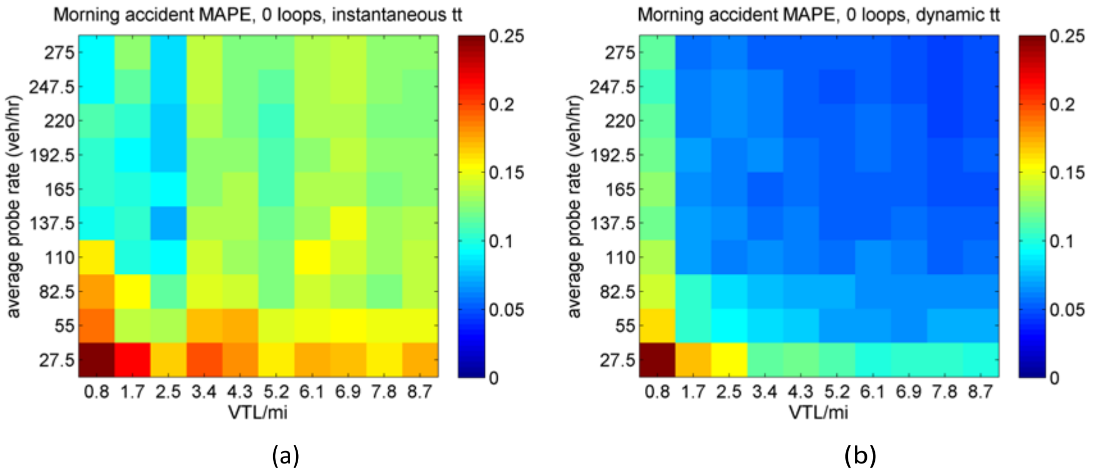


Figure 5-13: MAPE contours computed for the morning accident using VTL data and inductive loop detector sensors. x-axis: number of VTLs, y-axis: average probe data rate; (a) 0 inductive loop detector sensors, instantaneous travel time; (b) 0 inductive loop detector sensors, dynamic travel time. Color scale limited to 0.25. See also Table 17 and Table 18.

6 SUMMARY AND FUTURE WORK

In this study, trade-offs between velocity data collected from GPS smartphones in probe vehicles and velocity data obtained from inductive loop detectors, for the purpose of computing travel times on a stretch of highway, were studied.

This work was a case study that used experimental probe data obtained from the Mobile Century field experiment. The loop detector data was obtained from PeMS. The measurements were combined with a mathematical traffic model in a highway traffic estimation algorithm using a data assimilation technique called ensemble Kalman filtering, developed as a part of the Mobile Millennium project. The results of the algorithm were compared against the true travel times experienced by the drivers, obtained through license plate re-identification. A number of scenarios were created in which the volume of the probe data and number of inductive loop detector stations available for the estimation algorithm could be adjusted.

The following is a summary of the key results found in this study:

1. **Achieving 10% error for dynamic travel times.** In this study, it was found that the dynamic travel time estimates can be achieved with less than 10% error when using a flow model with data assimilation, by using either inductive loop detector data, probe data, or a mixture of both inductive loop detector data and probe data. Moreover, the estimates from virtual trip line–based probe data can achieve a higher degree of accuracy when all available probe data is used compared to estimates from inductive loop detectors when all inductive loops on the experiment site are used, although in general the performance is similar.

2. **Minimum loop detector spacing for travel time estimation.** In this study, using data from more than eight inductive loop detector stations (average spacing 0.83 miles) did not give extra benefit in the travel time estimation. The error remains constant between 6–13% depending on the time of day, regardless of the added loop detector stations.

3. **Diminishing travel time accuracy improvement.** When sampling probe vehicles at a rate of 137.5 veh/hr with more than 2.54 VTL/mi, increasing the number of probe measurements by adding more probe vehicles or additional trip lines causes only small improvement in the travel time accuracy.

4. **A mixture of probe and loop detector data in travel time estimation.** It was found that when complementing loop detector data with probe vehicle data, better estimates for travel times are obtained, especially at low penetration rates. For example, if using loop detectors spaced more than 2.11 miles apart, probe data can give over 50% increase in the travel time accuracy. These results hold generally, independent of the sampling strategy of the probe vehicles.

Based on the results found in this study, several additional areas should be explored as part of future work, including:

1. **Need for a traffic forecasting model.** One of the topics stressed throughout this study is the difference between travel time information that can be made available for the driving public in real time and information available a posteriori. The dynamic travel time estimates, while more accurate, cannot be estimated in real time.

   Especially during rapidly changing traffic conditions, instead of collecting additional measurement data in order to provide more accurate instantaneous travel time estimates, it is beneficial to develop a novel forecasting algorithm that can utilize the real-time data obtained until the point at which the forecast is made. The volumes of data needed to use these types of forecasting algorithms can be reasonably assumed to correspond to the guidelines presented in this study. It was demonstrated that adding large volumes of probe or loop detector data does not improve the quality of real-time instantaneous travel time information (derived from speed) after a certain threshold.

   A powerful forecasting algorithm enables the delivery of almost dynamic travel time–quality estimates of the travel times for the driving public. Statistical inverse problems theory and machine learning theory, for example, offer tools to tackle this problem.

2. **Generalization of the results for different sites.** Before more generalization of the results obtained in this study can be made, the methodology needs to be tested on different sites. Different sites provide more information about the variability of penetration rate and loop detector spacing that can be used to verify and improve the data collection guidelines suggested in this study.

3. **Optimal loop detector station placement.** The guidelines regarding the loop detector station spacing were made assuming a uniform spacing and that no prior information about the traffic features of the site is available. Studies concerning the optimal placement of loop detectors for the travel time estimation purposes have already been conducted. However, in the presence of probe data, the optimal placement of loop detector stations can possibly result in a more cost efficient installation.

4. **Other performance metrics.** The model used in this study was developed for high resolution average speed estimation. It was not calibrated or developed to estimate many other important highway performance metrics such as vehicle miles traveled, vehicle hours traveled, delays, etc. Thus, the ability of inductive loop detector data or probe data to estimate these performance metrics should be considered in future studies.

7 SUPPLEMENTARY TABLES

Table 3: Number of probe vehicle measurements used in the simulations when using VTL data. See also Figure 5-8a.

measurements/hr/mi										
probe rate	VTL/mi									
veh/hr	0.79	1.67	2.54	3.42	4.30	5.18	6.05	6.93	7.81	8.68
275.00	7977	15589	23363	31265	38559	46154	54615	62568	69802	77073
247.50	7147	13960	20923	27998	34529	41330	48902	56035	62496	69012
220.00	6383	12469	18685	25005	30836	36907	43671	50035	55809	61622
192.50	5588	10911	16356	21886	26996	32301	38218	43792	48847	53942
165.00	4811	9393	14076	18838	23239	27801	32892	37692	42032	46429
137.50	4012	7842	11749	15733	19410	23214	27465	31473	35102	38775
110.00	3237	6334	9489	12708	15681	18750	22185	25425	28358	31326
82.50	2435	4773	7149	9579	11821	14126	16712	19159	21367	23606
55.00	1631	3195	4784	6408	7908	9451	11187	12822	14297	15792
27.50	817	1587	2377	3185	3930	4698	5555	6373	7104	7848

Table 4: Number of probe vehicle measurements used in the simulations when using time-sampled data. See also Figure 5-8b.

measurements/hr/mi								
probe rate	sampling interval(s)							
veh/hr	384.00	192.00	96.00	48.00	24.00	12.00	6.00	3.00
275.00	2027	4049	8094	16184	32365	64723	129440	258872
247.50	1817	3630	7257	14511	29019	58032	116059	232111
220.00	1622	3240	6477	12951	25900	51794	103583	207160
192.50	1418	2832	5662	11322	22642	45278	90552	181099
165.00	1222	2441	4880	9758	19514	39022	78040	156076
137.50	1026	2050	4098	8194	16387	32769	65534	131064
110.00	831	1661	3320	6639	13277	26550	53097	106190
82.50	627	1253	2505	5009	10018	20033	40063	80123
55.00	420	839	1677	3354	6708	13414	26826	53650
27.50	212	424	848	1696	3392	6783	13565	27129

Table 5: Travel time MAPE (in %) using VTL probe data and 0 loop detector sensors. Travel time is computed using the dynamic method. See also Figure 5-10a.

Morning accident MAPE, 0 loops, dynamic tt										
probe rate	VTL/mi									
veh/hr	0.79	1.67	2.54	3.42	4.30	5.18	6.05	6.93	7.81	8.68
275.00	11.34	5.67	5.92	5.18	5.30	5.11	5.27	5.03	4.52	5.05
247.50	10.81	6.12	6.35	5.88	5.13	5.00	5.11	5.00	4.60	4.76
220.00	11.67	6.06	6.34	5.87	5.22	5.32	5.74	5.19	4.60	4.69
192.50	11.73	6.71	6.03	6.29	5.70	5.11	5.52	5.33	4.80	5.11
165.00	12.72	6.39	5.98	5.41	5.73	5.29	5.46	5.37	5.03	4.73
137.50	11.98	6.75	6.41	5.78	6.13	5.23	5.46	5.70	5.36	5.36
110.00	13.28	6.74	6.19	6.46	6.10	5.47	6.49	6.18	5.32	5.79
82.50	14.30	10.21	8.42	7.77	7.41	7.10	6.30	6.35	6.46	6.51
55.00	16.09	10.50	9.30	8.26	7.86	6.99	6.90	6.60	7.09	7.29
27.50	28.90	16.88	15.37	11.34	11.81	11.23	9.91	10.30	10.30	10.06

Table 6: Travel time MAPE (in %) using VTL probe data and 0 loop detector sensors. Travel time is computed using the dynamic method. See also Figure 5-10b.

Free flow MAPE, 0 loops, dynamic tt										
probe rate	VTL/mi									
veh/hr	0.79	1.67	2.54	3.42	4.30	5.18	6.05	6.93	7.81	8.68
275.00	8.28	6.03	5.91	6.26	5.84	5.79	5.87	5.88	5.79	6.13
247.50	8.53	6.30	5.98	5.80	5.75	5.69	5.96	5.94	5.83	6.03
220.00	8.72	6.25	6.00	5.38	5.51	5.42	5.94	5.59	5.62	5.97
192.50	9.79	6.53	5.95	5.88	5.82	5.81	5.75	5.82	5.65	5.89
165.00	8.94	6.56	6.57	6.39	5.97	6.16	5.90	6.06	5.92	6.09
137.50	12.22	7.28	6.71	6.18	6.31	5.95	6.14	6.11	5.94	6.14
110.00	13.77	7.25	7.21	7.16	6.91	6.65	6.67	6.80	6.39	6.40
82.50	15.43	9.17	8.15	7.16	7.17	7.22	6.89	6.84	7.19	6.80
55.00	16.37	14.48	11.21	9.38	9.10	9.32	8.68	8.31	9.11	8.75
27.50	16.89	16.51	14.02	12.82	11.51	10.54	9.93	9.22	10.06	9.74

Table 7: Travel time MAPE (in %) using VTL probe data and 0 loop detector sensors. Travel time is computed using the dynamic method. See also Figure 5-10c.

Congestion building MAPE, 0 loops, dynamic tt										
probe rate	VTL/mi									
veh/hr	0.79	1.67	2.54	3.42	4.30	5.18	6.05	6.93	7.81	8.68
275.00	6.10	7.42	4.56	3.56	3.74	3.94	3.67	3.42	3.43	3.76
247.50	6.69	6.93	4.66	3.50	4.06	4.00	3.76	3.39	3.45	3.85
220.00	6.66	6.94	4.59	3.49	4.28	4.02	3.33	3.34	3.37	3.70
192.50	6.67	7.41	5.25	3.85	4.52	4.38	3.51	3.37	3.74	3.86
165.00	7.04	8.25	4.84	3.65	4.71	4.26	3.58	3.40	3.75	3.89
137.50	7.36	8.55	5.40	4.26	4.85	4.68	3.81	3.67	4.05	4.08
110.00	8.35	9.01	5.60	4.86	5.70	5.42	4.14	4.05	4.37	4.18
82.50	10.16	10.57	6.80	5.89	7.57	6.66	5.15	5.01	5.59	5.18
55.00	16.38	14.07	10.34	8.79	9.99	8.70	7.40	6.40	6.95	7.80
27.50	30.18	21.70	16.30	15.05	15.98	13.41	12.34	11.38	11.47	12.47

Table 8: Travel time MAPE (in %) using VTL probe data and 0 loop detector sensors. Travel time is computed using the dynamic method. See also Figure 5-10d.

Full congestion MAPE, 0 loops, dynamic tt										
probe rate	VTL/mi									
veh/hr	0.79	1.67	2.54	3.42	4.30	5.18	6.05	6.93	7.81	8.68
275.00	5.50	4.97	4.58	5.25	4.21	3.99	4.77	4.54	4.18	4.21
247.50	4.97	5.39	4.36	5.18	4.62	4.15	5.09	4.67	4.31	4.39
220.00	5.18	5.13	4.68	5.59	4.36	4.56	5.20	5.16	4.59	4.79
192.50	5.47	4.74	5.25	6.01	4.31	5.06	5.59	5.52	5.23	4.80
165.00	6.31	5.27	5.27	6.30	4.29	4.92	5.44	5.86	5.08	4.88
137.50	6.24	5.63	5.48	6.35	4.01	4.90	5.54	6.13	5.23	5.20
110.00	7.61	5.82	5.78	7.52	6.11	6.38	6.65	7.68	6.21	7.02
82.50	8.82	7.17	6.45	8.64	7.34	6.98	7.44	7.73	6.80	7.09
55.00	9.51	7.53	6.71	7.42	6.69	6.57	7.02	7.65	6.67	6.54
27.50	14.42	10.90	5.18	7.54	4.35	4.58	4.82	6.50	5.22	4.34

Table 9: Travel time MAPE (in %) change when 6 loop detector sensors are used. Travel time is computed using the dynamic method. See also Figure 5-11a.

Morning accident change in MAPE, adding 6 loops, dynamic tt										
probe rate	VTL/mi									
veh/hr	0.79	1.67	2.54	3.42	4.30	5.18	6.05	6.93	7.81	8.68
275.00	-2.44	0.22	-1.08	0.72	-0.08	-0.18	-0.11	0.25	-0.22	-0.71
247.50	-2.63	0.13	-1.15	-0.07	-0.14	-0.21	0.14	-0.08	-0.34	0.01
220.00	-3.45	0.62	-1.38	0.12	-0.41	-0.20	-0.46	0.29	-0.09	-0.13
192.50	-3.59	-0.46	-1.13	-0.71	-0.29	-0.32	-0.30	0.27	-0.49	-0.41
165.00	-3.47	0.35	-0.51	0.66	0.46	0.10	-0.37	0.34	-0.45	-0.01
137.50	-3.65	-0.22	-1.22	-0.62	0.33	0.31	0.28	0.42	-0.64	-0.22
110.00	-4.88	-2.11	-0.71	-1.17	-1.28	0.26	-0.85	0.15	-0.29	-0.29
82.50	-4.87	-5.34	-3.08	-3.02	-2.32	-1.83	0.09	-0.97	-1.83	-0.70
55.00	-7.93	-4.75	-3.04	-1.81	-2.31	-0.69	-1.72	0.19	-1.54	-0.87
27.50	-20.84	-10.62	-7.86	-5.22	-5.47	-3.86	-3.28	-3.18	-4.35	-2.67

Table 10: Travel time MAPE (in %) change when 6 loop detector sensors are used. Travel time is computed using the dynamic method. See also Figure 5-11b.

Free flow change in MAPE, adding 6 loops, dynamic tt										
probe rate	VTL/mi									
veh/hr	0.79	1.67	2.54	3.42	4.30	5.18	6.05	6.93	7.81	8.68
275.00	8.19	0.44	0.27	-0.24	0.63	0.32	0.70	0.59	0.38	0.15
247.50	6.32	0.04	0.53	0.40	0.71	0.20	0.23	0.54	0.15	0.27
220.00	6.79	0.17	0.39	0.92	0.79	0.61	0.34	0.61	0.47	0.26
192.50	5.90	0.19	0.75	0.45	0.75	0.36	0.55	0.89	0.55	0.24
165.00	8.82	0.52	0.14	0.04	0.53	-0.02	0.69	0.47	0.47	0.13
137.50	5.82	-0.20	0.13	0.43	0.47	0.60	0.24	0.29	0.10	0.29
110.00	5.16	-0.31	-0.15	-0.79	-0.45	-0.07	-0.53	-0.08	0.03	0.24
82.50	3.08	-1.99	-1.02	-0.79	-0.48	-0.62	-0.23	-0.25	-0.78	-0.08
55.00	1.05	-6.99	-3.64	-2.85	-1.89	-1.94	-1.24	-1.59	-2.71	-1.96
27.50	1.94	-8.70	-5.85	-5.60	-4.09	-3.10	-3.17	-2.10	-3.10	-2.78

Table 11: Travel time MAPE (in %) change when 6 loop detector sensors are used. Travel time is computed using the dynamic method. See also Figure 5-11c.

Congestion building change in MAPE, adding 6 loops, dynamic `tt`										
probe rate	VTL/mi									
veh/hr	0.79	1.67	2.54	3.42	4.30	5.18	6.05	6.93	7.81	8.68
275.00	1.56	0.58	1.40	0.64	1.30	1.07	0.04	0.66	1.02	-0.09
247.50	1.02	0.73	1.71	0.96	1.05	1.09	0.15	0.70	1.16	-0.11
220.00	1.78	0.48	1.64	0.66	1.15	0.98	0.70	0.65	1.34	0.50
192.50	0.78	0.06	0.99	0.46	0.57	0.76	0.73	0.78	1.13	0.41
165.00	1.38	-0.83	0.89	0.93	0.46	0.68	0.63	0.97	1.01	0.45
137.50	0.88	-0.72	0.44	0.22	1.36	0.70	0.91	0.46	1.08	0.63
110.00	0.21	-1.46	1.00	0.75	0.47	0.36	1.32	1.17	1.53	1.22
82.50	-0.84	-2.88	-0.21	0.37	-1.04	-0.83	0.44	0.55	0.52	0.41
55.00	-7.37	-6.70	-3.63	-2.40	-3.19	-2.58	-1.46	-0.47	-0.61	-1.30
27.50	-21.00	-13.57	-9.24	-7.96	-9.02	-6.66	-5.58	-5.03	-4.64	-5.74

Table 12: Travel time MAPE (in %) change when 6 loop detector sensors are used. Travel time is computed using the dynamic method. See also Figure 5-11d.

Full congestion change in MAPE, adding 6 loops, dynamic `tt`										
probe rate	VTL/mi									
veh/hr	0.79	1.67	2.54	3.42	4.30	5.18	6.05	6.93	7.81	8.68
275.00	1.33	2.02	0.53	-0.78	1.30	1.05	-0.34	-0.52	0.37	0.44
247.50	1.69	2.46	0.53	-0.70	0.82	1.10	-0.49	-0.54	0.20	0.28
220.00	1.96	2.39	0.34	-0.98	1.28	0.22	-0.47	-0.87	-0.30	-0.19
192.50	1.97	2.95	-0.36	-1.23	1.09	0.08	-0.68	-1.23	-0.49	-0.13
165.00	1.90	3.87	0.04	-1.34	1.44	0.47	-0.45	-0.89	-0.05	0.13
137.50	2.56	3.19	-0.43	-1.13	2.90	0.65	-0.20	-1.18	-0.39	0.02
110.00	1.95	3.49	0.09	-1.59	1.43	-0.60	-1.24	-2.34	-0.23	-2.02
82.50	3.37	4.51	-0.59	-2.31	0.37	-0.42	-1.60	-2.05	-0.53	-1.59
55.00	2.17	4.21	-0.27	-0.49	0.30	-0.10	-0.53	-1.48	-0.31	0.33
27.50	-1.96	2.02	4.93	3.03	7.04	3.77	4.20	1.33	3.35	5.42

Table 13: Travel time MAPE (in %) using time-sampled probe data and 0 loop detector sensors. Travel time is computed using the dynamic method. See also Figure 5-12a.

Morning accident MAPE, 0 loops, dynamic tt								
probe rate	sampling interval(s)							
veh/hr	384.00	192.00	96.00	48.00	24.00	12.00	6.00	3.00
275.00	8.20	8.80	9.40	10.30	9.57	9.06	8.39	8.64
247.50	9.19	9.06	9.10	9.56	9.46	9.02	8.96	9.33
220.00	7.78	9.52	10.72	10.19	9.64	9.66	9.18	8.71
192.50	7.75	9.15	9.40	10.10	9.55	9.41	9.43	9.15
165.00	9.12	9.74	10.12	9.60	10.92	10.06	10.33	9.14
137.50	8.77	10.30	10.62	10.64	10.91	10.90	11.03	10.96
110.00	9.75	10.55	11.63	10.93	11.46	11.23	11.58	11.63
82.50	9.71	12.34	13.17	11.18	11.91	11.69	11.98	12.25
55.00	11.67	14.02	15.26	12.83	12.29	12.31	13.41	12.98
27.50	19.10	10.52	13.61	11.06	9.96	10.95	10.58	11.57

Table 14: Travel time MAPE (in %) using time-sampled probe data and 6 loop detector sensors. Travel time is computed using the dynamic method. See also Figure 5-12b.

Morning accident MAPE, 6 loops, dynamic tt								
probe rate	sampling interval(s)							
veh/hr	384.00	192.00	96.00	48.00	24.00	12.00	6.00	3.00
275.00	8.46	8.71	8.44	8.84	8.49	7.86	8.07	7.77
247.50	9.58	8.92	8.81	8.71	8.39	8.11	7.71	7.86
220.00	9.76	8.81	9.20	8.71	8.63	8.18	8.32	8.42
192.50	10.10	8.94	9.08	9.23	8.60	8.41	8.38	8.41
165.00	10.05	9.07	9.32	8.70	9.15	9.15	8.33	8.60
137.50	8.37	7.32	8.94	8.93	9.31	9.52	8.89	9.36
110.00	7.69	8.28	9.07	8.84	9.80	8.68	9.37	10.14
82.50	11.80	8.10	7.84	8.87	9.22	9.31	9.68	10.15
55.00	7.78	9.77	7.92	8.26	10.15	9.43	9.91	9.86
27.50	8.06	10.60	8.78	7.24	9.72	9.60	9.19	9.95

Table 15: Travel time MAPE (in %) using time-sampled probe data and 0 loop detector sensors. Travel time is computed using the dynamic method. See also Figure 5-12c.

Full congestion MAPE, 0 loops, dynamic tt								
probe rate	sampling interval(s)							
veh/hr	384.00	192.00	96.00	48.00	24.00	12.00	6.00	3.00
275.00	15.41	18.29	18.77	18.66	17.47	16.99	16.09	15.69
247.50	15.46	18.06	18.18	18.39	17.14	16.73	16.29	15.28
220.00	17.32	18.69	18.66	18.58	17.60	17.34	16.55	16.34
192.50	18.04	17.99	17.99	18.08	18.04	17.45	17.23	16.73
165.00	15.41	17.68	17.05	19.17	18.14	18.10	18.05	17.80
137.50	14.73	19.38	18.05	19.60	18.02	18.16	18.27	17.84
110.00	15.71	18.19	18.98	19.61	18.67	19.34	19.84	19.38
82.50	16.58	19.19	18.56	19.11	18.79	19.72	19.92	19.81
55.00	14.77	19.41	19.10	18.55	17.83	18.79	19.12	18.28
27.50	50.48	8.43	11.71	16.99	17.72	19.62	19.11	21.24

Table 16: Travel time MAPE (in %) using time-sampled probe data and 6 loop detector sensors. Travel time is computed using the dynamic method. See also Figure 5-12d.

Full congestion MAPE, 6 loops, dynamic tt								
probe rate	sampling interval(s)							
veh/hr	384.00	192.00	96.00	48.00	24.00	12.00	6.00	3.00
275.00	6.93	6.63	5.67	6.85	7.93	9.54	10.64	11.94
247.50	7.56	6.70	5.59	6.56	7.64	9.14	10.58	11.08
220.00	8.24	5.98	5.69	6.43	7.24	8.79	10.31	11.56
192.50	8.53	5.88	4.92	5.58	6.87	8.08	9.49	10.88
165.00	8.70	6.14	5.59	5.57	6.22	7.66	9.32	10.44
137.50	9.15	7.83	6.09	5.23	5.97	6.79	8.81	9.69
110.00	10.84	8.01	6.86	5.22	6.04	6.85	7.99	8.95
82.50	12.00	11.31	5.89	5.92	6.25	6.34	7.28	8.84
55.00	12.63	11.16	6.36	5.73	5.92	5.28	6.03	7.12
27.50	13.12	13.04	10.62	9.18	7.58	6.44	6.61	6.83

Table 17: Travel time MAPE (in %) using VTL probe data and 0 loop detector sensors. Travel time is computed using the instantaneous method. See also Figure 5-13a.

Morning accident MAPE, 0 loops, instantaneous tt										
probe rate	VTL/mi									
veh/hr	0.79	1.67	2.54	3.42	4.30	5.18	6.05	6.93	7.81	8.68
275.00	9.34	12.77	8.29	14.04	12.43	12.17	13.34	13.69	12.83	12.54
247.50	9.20	11.51	8.33	13.97	12.37	11.55	13.61	13.69	12.52	12.44
220.00	10.99	10.25	8.09	13.67	12.27	10.80	13.57	13.39	12.15	12.30
192.50	10.49	9.03	8.06	12.86	12.59	11.31	12.70	13.73	12.52	12.74
165.00	10.52	9.78	9.35	12.73	13.60	10.95	13.62	13.52	12.32	12.68
137.50	9.39	10.33	7.17	13.30	13.49	11.95	12.97	15.15	13.03	13.49
110.00	15.69	10.12	9.17	14.29	13.08	12.28	15.52	14.21	13.05	14.01
82.50	17.95	15.48	11.51	14.83	14.29	12.30	14.39	13.50	12.46	13.72
55.00	19.11	13.90	13.53	17.01	17.37	14.73	15.10	15.60	15.11	15.23
27.50	28.37	21.72	16.57	19.68	18.25	15.67	17.51	17.11	15.70	17.49

Table 18: Travel time MAPE (in %) using VTL probe data and 0 loop detector sensors. Travel time is computed using the dynamic method. See also Figure 5-13b.

Morning accident MAPE, 0 loops, dynamic tt										
probe rate	VTL/mi									
veh/hr	0.79	1.67	2.54	3.42	4.30	5.18	6.05	6.93	7.81	8.68
275.00	11.34	5.67	5.92	5.18	5.30	5.11	5.27	5.03	4.52	5.05
247.50	10.81	6.12	6.35	5.88	5.13	5.00	5.11	5.00	4.60	4.76
220.00	11.67	6.06	6.34	5.87	5.22	5.32	5.74	5.19	4.60	4.69
192.50	11.73	6.71	6.03	6.29	5.70	5.11	5.52	5.33	4.80	5.11
165.00	12.72	6.39	5.98	5.41	5.73	5.29	5.46	5.37	5.03	4.73
137.50	11.98	6.75	6.41	5.78	6.13	5.23	5.46	5.70	5.36	5.36
110.00	13.28	6.74	6.19	6.46	6.10	5.47	6.49	6.18	5.32	5.79
82.50	14.30	10.21	8.42	7.77	7.41	7.10	6.30	6.35	6.46	6.51
55.00	16.09	10.50	9.30	8.26	7.86	6.99	6.90	6.60	7.09	7.29
27.50	28.90	16.88	15.37	11.34	11.81	11.23	9.91	10.30	10.30	10.06

Chapter 6

Hybrid Data Roadmap

This chapter was prepared for PATH by Novavia Solutions as part of a subcontract. It presents an overarching view of the context, objectives, and implementation of a hybrid traffic data system, drawing on the full scope of work completed for Task Orders 1 and 2, including the many weeks of data and the in-depth analysis of data fusion detailed in the final report for Task Order 1, *Pilot Procurement of Third-Party Traffic Data*. This roadmap chapter also contains a business analysis that assesses the benefits, trade-offs, and next steps in procuring third-party data and integrating it into Caltrans' existing structure. It can be read either as part of the Task Order 1 and 2 reports or as a stand-alone document that incorporates the PATH findings into a broad synthesis of the issues surrounding hybrid traffic data.

EXECUTIVE SUMMARY

In recent years, new methods of roadway traffic data collection have become available. In particular, data streams originating from smartphones represent a radical breakthrough in terms of scope and extent. They also involve a completely different supply chain and raise relatively novel issues such as individual privacy. This roadmap examines the vision of a so-called "hybrid" data system in which legacy data collected by roadway sensors coexist with third-party data supplied by private sector entities, presumably, though not necessarily exclusively, originating from cellular phones.

DATA COLLECTION METHODS

Traffic data collection methods can be broken down into three categories. Point-based data collection methods employ sensors that pick up traffic volumes and/or speeds at one dedicated location. That is the method most commonly employed by Caltrans today. The second category of traffic data collection methods provides trip times for preset road segments. Segment-based data collection is achieved by vehicle reidentification, that is, the ability to uniquely match records of a traveling vehicle obtained at two different locations. The third available category of traffic data collection methods relies on mobile data networks to extract the velocity of individual vehicles. This offers two key advantages: 1) no infrastructure deployment is necessary (save for cellular network infrastructure, but that is exogenous), and 2) data can be obtained from virtually any location on the roadway network, as long as there is cellular coverage.

Each category of traffic data collection methods has its own strengths. Only point-based collection provides reliable traffic counts. On the other hand, producing accurate travel time estimates between two network locations from point-based data has always proven difficult because it offers no information about conditions between sensors. This can be overcome by installing a high density of detectors, but that obviously comes at a cost. Segment-based detection works well to generate traveler information or establish performance measures, particularly on signalized arterials where traffic signal timing plays a determining factor in maintaining adequate flows. It is superior to mobile data sources in that it normally offers much higher sample sizes. Mobile data offers unprecedented coverage with no need for infrastructure deployment and maintenance. As of today, the data is limited to speed estimates, which makes it most suitable for traveler information.

HYBRID TRAFFIC DATA

A hybrid traffic data environment would result from the incorporation of third-party data (from mobile data or other sources) into transportation management systems, to complement the data currently collected by Caltrans. If employed sensibly, this strategy could augment information availability and quality while costing less than current business practices. In turn, increasing access to information would improve the ability of Caltrans and its local partners to manage roadway traffic in ways that balance demand and flow in order to minimize delays and make the most of their infrastructure investments.

The objectives of a hybrid traffic data strategy would be as follows:

1. Filling gaps in the current traffic detector infrastructure and/or relaxing maintenance on secondary detection stations

2. Increasing coverage and providing uniform congestion reports

3. Enhancing information with data fusion, i.e., using multiple data sources and making the whole greater than the sum of the parts

Currently, the most common type of third-party traffic data available consists of aggregated speed data. Such data can contribute to more accurate and robust traveler information and generally costs less than using fixed detectors. It can also expand the coverage and sometimes the accuracy of traffic performance measures. On the other hand, it remains inadequate for more advanced traffic management applications such as ramp metering.

It is conceivable that mobile data sources can become a primary element of Caltrans' traffic information infrastructure and power even the most demanding traffic management applications. So far, however, that concept has not yet been proven. At a minimum, this would require that mobile data be processed at an elementary level, that is, starting with individual vehicle observations. Further, no public agency has yet procured unaggregated mobile data, except on an experimental basis. Clearly, additional research and development into traffic estimation procedures and systems is still required to give substance to the possibility of using mobile data for advanced traffic management.

IMPLICATIONS FOR CALTRANS

Systematically bringing third-party traffic data into Caltrans' information mix would have key implications for Caltrans as an organization:

- **Outsourcing data collection**—Procurement specifications and methods must be devised for that purpose.

- **Managing new risks**—Specific risks would arise from the incorporation of third-party data.

  - *Going concern risk:* Caltrans would have to insure against the default of traffic data providers.

  - *Data quality risk:* Data quality would need ongoing monitoring, in step with corresponding contractual terms.

  - *Data Privacy risk:* Any data that originates from a personal electronic device poses privacy risks that will need to be addressed with adequate policies and practices.

- **Designing and implementing new information systems**—Current information systems will require significant changes in order to integrate third-party data.

- **Adopting a new detector strategy**—The implementation of a hybrid traffic data roadmap can only make sense if it helps save money on traffic detector deployment and maintenance, and thus a new strategy would be needed in that area.

IMPLEMENTATION PATH

This roadmap suggests an implementation path that would enable Caltrans to gradually experiment with and adopt a hybrid traffic data collection system. It involves three steps that could be developed over a time horizon of three to five years:

1. Short-term pilot in a selected district (1-2 years)
2. Using the PATH Connected Corridors project to spearhead information systems innovations (2-3 years)
3. Full-scale pilot in a selected district (3-5 years)

The implications corresponding to each implementation step are listed in the following table:

	Short-term Pilot	Connected Corridors	Full-scale Pilot
Outsourcing	Pilot procurement and contract management for third-party traffic data	Incremental improvements	Incremental improvements
Risk management	Limited to setting policies on data privacy and security	Limited by implementing Connected Corridors on an experimental basis	Third-party traffic data dependence can be initially limited by roll-back option
Information systems	Integration of mobile data into PeMS	Prototype integration of mobile data into traffic management applications	Pilot operations-grade integration of mobile data into traffic management applications
Detection strategy	No impact	No impact	Start shifting priorities based on mobile data availability
Target benefits	Enhanced traveler information and performance measures	Traffic state estimation from hybrid data; novel/optimized integrated corridor traffic management strategies	Novel/optimized traffic management strategies; save on detector maintenance costs; open architecture

TRADE-OFFS AND PROCUREMENT

The primary rationale for shifting traffic data collection from the current detector-based paradigm to a hybrid system that incorporates third-party data is to improve information quality for a given budget allocation. Decisions should therefore be driven by trade-offs between the costs and performance of available data sources. This roadmap incorporates summary findings from PATH's hybrid data roadmap project (Task Orders 1 and 2) regarding the relative performance yielded by inductive loop detectors and mobile data sources in estimating travel times. Probe data sources procured by PATH from responsive vendors as part of the hybrid data project could often match extant detectors in producing consistent travel time estimates. They can also do it at less expense, especially if the maintenance and replacement costs of keeping traffic detectors in perpetuity are factored in. Still, the highest level of traffic estimation quality could only be reached with a combination of probe data and detector data. An important caveat to this analysis is that travel time estimations only constitute one application of traffic information, which is less stringent than advanced traffic management applications.

Procuring third-party traffic data has become more mainstream over the last few years. In this respect the I-95 corridor coalition, which started providing traveler information with probe-based data in 2008, established a watershed. Nonetheless, the market for privately provided traffic information remains somewhat immature, which makes the procurement process more difficult. A key to success is the identification of well-formed service specifications, including functional specifications (data sources, data processing methods, geographical coverage and metadata), performance specifications (accuracy, completeness, validity, timeliness, coverage, and accessibility), vendor data management practices, licensing terms, and pricing structure. Further, in the absence of independent benchmarks, it is necessary for buyers to evaluate the quality of traffic data services as part of the vendor selection process and to continue monitoring data quality once a vendor is under contract.

LIST OF ACRONYMS

ATDM	Advanced Transportation and Demand Management
ATMS	Advanced Transportation Management System
CMA	Congestion Management Agency
CMS	Changeable Message Sign
DOT	Department of Transportation
DSS	Decision Support System
ETC	Electronic Toll Collection
FEP	Front End Processor
FSR	Feasibility Study Report
GHG	Greenhouse Gases
GPS	Global Positioning System
HAR	Highway Advisory Radio
HOT	High-Occupancy Tolling
HPMS	Highway Performance Monitoring System
ICM	Integrated Corridor Management
ITIP	Intelligent Transportation Infrastructure Program
JSON	JavaScript Object Notation
MAPE	Mean Absolute Percentage Error
MPO	Metropolitan Planning Organization
MTC	Metropolitan Transportation Commission
PATH	Partners for Advanced Transportation TecHnologies
PeMS	Performance Measurement System
PMATE	Per-Mile Absolute Time Error
RFP	Request for Proposals

TMC	Transportation Management Center
TMC	Traffic Messaging Channel
TMS	Transportation Management Systems
TPEG	Transport Protocol Experts Group
VAR	Value Added Reseller
XML	eXtensible Markup Language

1 INTRODUCTION

1.1 BACKGROUND

Managing roadway traffic, which includes responding to incidents, controlling flows to balance demand and capacity across times and network locations, and informing drivers to avoid gridlock, requires timely and accurate data. Similarly, the quality of transportation improvement plans is largely a function of the information that supports them. In recent years, new methods of roadway traffic data collection have become available. Some of them are technological advances that have improved the cost-to-performance ratio over legacy technologies. Others, most notably data streams originating from smartphones, represent a radical breakthrough in terms of scope and extent. They also involve a completely different supply chain and raise relatively novel issues such as individual privacy. This presents both opportunities and challenges for infrastructure managers such as Caltrans.

In 2010, Caltrans commissioned the California Partners for Advanced Transportation Technology (PATH) at UC Berkeley to study these opportunities and challenges. The corresponding vision is that of a so-called "hybrid" data system in which legacy data collected by roadway sensors coexist with third-party data supplied by private sector entities, presumably, though not necessarily exclusively, originating from cellular phones. The PATH project, named *Hybrid Traffic Data Collection Roadmap*, consisted of carrying out detailed investigations while procuring privately-provided traffic data that was subsequently evaluated. A complete report of the methods and findings of the project is available from PATH.

1.2 DOCUMENT OVERVIEW

This document is intended as a broad synthesis that can be read either as a chapter within the complete PATH project report or as a stand-alone complement to it. While some portions, especially in section 4, are quite technical, the bulk of it is written for a relatively wide audience of transportation engineers and planners, policy-makers, finance officers, and other professional parties who have a stake or an interest in the topic of traffic information. The document generally speaks to Caltrans' position and needs, which are transposable to other Departments of Transportation (DOTs) and large infrastructure managers, but obviously the information is applicable to different types of organizations.

Section 2 describes the existing state of practice. It starts by defining the primary applications of traffic information. It then gives an overview of Caltans' current traffic information infrastructure. The last sub-section provides a comprehensive review of available traffic data collection methods, whether employed by Caltrans or not.

Section 3 gets to the core of the issue by discussing the implementation of a hybrid data system starting with Caltrans' current practices. First, a vision statement and some definitions are provided. The document then examines the benefits and challenges of migrating to a hybrid traffic data collection system across a broad range of aspects, such as organizational, contractual, legal, budgeting, and information technology implications. Finally, an actual roadmap for implementation is proposed.

Section 4 addresses the question of the costs and benefits associated with traffic data derived from mobile sources when compared with fixed detectors. There are several caveats to that analysis, however. First of all, traffic information performance is only evaluated with regard to travel time estimations. While accurate travel time estimates make for good driver information, they are not sufficient to enable advanced traffic management applications such as ramp metering, which require measurements of traffic volumes. Second, the analysis presented in section 4 is based on a limited set of observations on two specific freeway sections. While the results were established rigorously and provide insights applicable to any other freeway section, they cannot authoritatively carry over the entire network—in other words, the relative costs and benefits of fixed detectors and mobile sources vary by location.

Section 5 focuses on third-party traffic data procurement, drawing from PATH’s experience with the hybrid data roadmap project as well as interviews with practitioners. It highlights the primary market players as of this writing, lists the specifications that must be factored into the procurement and evaluation process, and adds a section on contract management.

2 EXISTING PRACTICE AND METHODS

2.1 TRAFFIC INFORMATION AND ITS APPLICATIONS

Traffic information can be defined elementarily as a set of variables including traffic volumes, traffic speed, and vehicle classes that describe prevailing conditions at a particular roadway location at a given moment in time. These basic traffic variables, combined with additional information such as weather forecasts or incident reports, can be transformed and aggregated in a variety of ways to create higher-level representations that help roadway operators, transportation planners, or individual drivers make decisions. For instance, a commuter may want to know current and predicted travel times in order to decide when to leave for work. Or an engineer needs reliable information about travel demand and delays on a corridor in order to adjust traffic signal synchronization in a way that minimizes future congestion.

Accurate, relevant traffic information is indispensable to the proper operations of roadway networks. As the saying goes, you cannot manage what you cannot measure. Traffic information needs are particularly crucial on urban and suburban roads that are affected by frequent congestion. Congestion can be mitigated by implementing a range of active traffic management strategies such as signal synchronization, freeway ramp metering, fast incident response, or changeable message signs that inform motorists about network conditions and alternative itineraries. The effectiveness of any of these strategies can be directly tied to the availability and quality of traffic information in the affected areas.

For most of the last century, roadway authorities around the world focused primarily on the engineering feats required to build and maintain structures that could safely carry tens or hundreds of thousands of vehicles every day. Yet we have known since at least the 1960s that this approach alone was not sufficient: Congestion inevitably sets in when demand exceeds roadway capacity, but capacity cannot be expanded indefinitely because of costs or sheer lack of space. In the past 20 years or so, several factors have contributed to impress that reality onto the organizations that oversee road networks: Capacity building has slowed as networks have reached near-completion, congestion has worsened due to economic expansion and new social habits, and information and communications technology has matured enough to offer effective traffic and demand management tools.

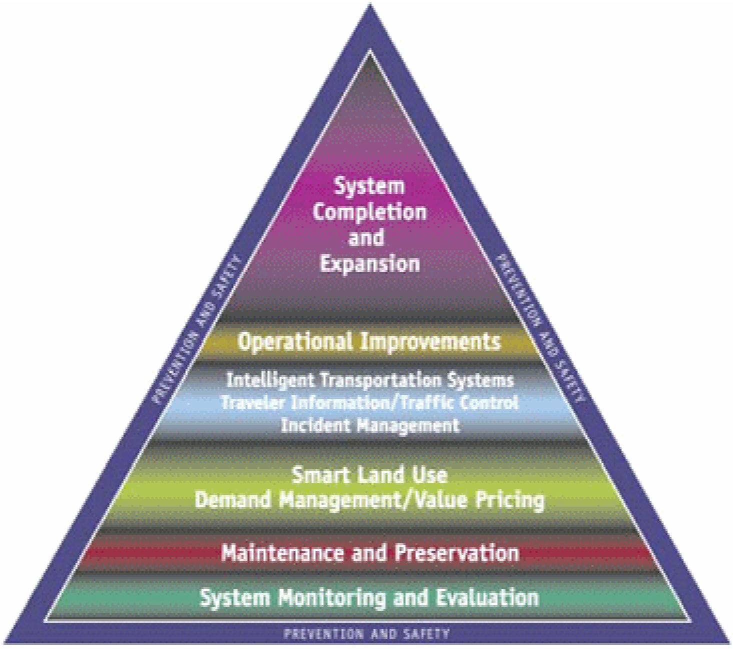

System Completion and Expansion
Operational Improvements
Intelligent Transportation Systems
Traveler Information/Traffic Control
Incident Management
Smart Land Use
Demand Management/Value Pricing
Maintenance and Preservation
System Monitoring and Evaluation
PREVENTION AND SAFETY
PREVENTION AND SAFETY
PREVENTION AND SAFETY

Figure 6-1: Caltrans system management pyramid

Accordingly, Caltrans has embraced mobility management as a key organizational function and adopted a system-level approach that emphasizes efficient use of existing transportation assets and the ability to dynamically balance demand and capacity before resorting to additional construction. Caltrans' system management paradigm is effectively conveyed by the pyramid in Figure 6-1. Each slice of the pyramid represents a particular function, and the sequence establishes a hierarchy of strategies, starting with the most ubiquitous at the bottom and ending with the most exceptional at the top. As can be seen, "System Monitoring and Evaluation" forms the base of the pyramid and is therefore the precursor to any other type of action. The monitoring and evaluation functions rely almost entirely on collecting information about the transportation network, of which traffic data is a key constituent.

Caltrans' Division of Traffic Operations has formally committed to system management with the adoption of a business plan for transportation management systems[10] (TMS) that was updated at the end of 2011. In this recent update, the TMS business plan lists five high-level goals. Of those, three directly require the availability of traffic information:

i. *Adopt and implement a performance-based framework for all TMS work activities and funding prioritization*—Establishing performance measures requires information.

ii. *Establish a well-maintained and high-performing TMS infrastructure that supports real-time traffic management*—Real-time traffic management requires the availability of real-time traffic information.

iii. *Cooperatively develop and implement real-time traffic management strategies that optimize flow and safety (and aid in regional and state requirements to reduce greenhouse gases (GHG) from transportation)*—Again, this goal revolves around real-time traffic management and makes traffic information availability a prerequisite.

[10] Caltrans Division of Traffic Operations: "Transportation Management Systems Business Plan Update" (December 2011)

In this context, the importance of traffic information to the effective management of transportation infrastructure that moves people and goods throughout the state of California can hardly be overstated. This is akin to any modern business, which cannot function well without substantial amounts of data about its operations. In the following sub-sections, we further elaborate on the needs and applications of traffic information with respect to its three primary uses: traffic operations, transportation planning and engineering, and traveler information.

2.1.1 TRAFFIC OPERATIONS

All aspects of traffic operations use some form of traffic information. At a minimum, operators must respond to roadway incidents and clear their impact, which requires timely information about incident occurrences. Beyond that, most traffic operations implement some form of traffic control: This includes traffic signals at intersections or at freeway ramps, dynamic lane management, and, increasingly, road pricing. At a high level, these traffic control strategies can be implemented in three different ways, each one carrying its own set of information requirements:

- **Static control**: In this form of control, the settings of traffic control devices are set statically. For instance, many traffic signals operate on a time-of-day basis with cycles that are pre-programmed. In this case, traffic engineers ideally need large sets of representative data about travel volumes to establish optimal settings.

- **Responsive control:** In contrast to static control, responsive traffic control devices constantly receive new information about current traffic conditions to adjust their settings. This situation is more typical of high-volume intersections and freeway ramp meters. It obviously requires real-time information about traffic volumes and traffic speeds. Traffic operators may also exploit additional information about weather, special events, and incidents to manually adjust control strategies.

- **Proactive control:** Proactive control is a particular form of responsive control in which settings depend not only on current conditions but also on anticipations about future conditions. In addition to needing real-time traffic information, proactive strategies require historical travel demand profiles as well as information about how drivers respond to given conditions and control actions. This form of control is still in its infancy, but some High-Occupancy Tolling lane deployments represent early examples.

2.1.2 TRANSPORTATION PLANNING AND ENGINEERING

Broadly speaking, transportation planners and engineers are charged with determining transportation needs, selecting projects among investment alternatives, and designing infrastructure improvements. These activities involve much data-intensive modeling that requires travel demand information, that is, where people or goods travel from and to, when they travel, and which mode they use. This information is not needed in real time but must be accurate enough to reproduce traffic conditions over large networks, both now and in a predicted future. Further, travel demand is not static and shifts based on supplied capacity; the relationship between transportation capacity and demand must therefore be captured. Both travel demand and travel behavior have to be inferred from traffic volumes over extended periods of time.

Transportation planning takes place at the local, state, and federal level. Therefore, traffic data collected by Caltrans is consumed by many other public agencies. In this respect, a fact of particular importance is that federal transportation dollars are apportioned to states by a formula that factors in traffic volumes. Additionally, grant funding is often tied to congestion reduction, and establishing congestion levels as well as demonstrating improvements requires traffic information. In effect, extensive and accurate traffic information can become a source of revenue in a competitive government funding environment.

2.1.3 TRAVELER INFORMATION

Traveler information is delivered by roadway operators through changeable message signs (CMS) or highway advisory radio (HAR), and by traveler information service providers (both public and private) through FM radio, phone lines (511 in particular), and digital interfaces (navigation units, smartphones, and the like). Traveler information is also available pre-travel through other media channels including television and internet websites. Traveler information is an important service in that it not only enhances individual experience but also sets a form of system self-management by helping drivers make decisions about their route, departure time, and mode of travel, which balances demand and capacity and ultimately benefits everybody.

The basic currency of traveler information is traffic speeds and corresponding delays, which is what matters most from an individual driver’s standpoint. At the same time, traveler information is most helpful when it incorporates some elements of prediction, since what you really want to know is what traffic conditions will be like once you get there, rather than what they are now. Predicting requires modeling, and the modeling of future traffic speeds ideally accounts for pending demand (which again, suggests that traffic volumes have been measured over time). However, statistical techniques exist that produce valuable traffic speed predictions exclusively from past observations of the traffic speeds themselves.

2.2 CALTRANS' TRAFFIC INFORMATION INFRASTRUCTURE

This section presents Caltrans' current traffic information infrastructure, including the field devices that collect primary data and the set of software applications that process that data.

2.2.1 TRAFFIC DATA COLLECTION DEVICES

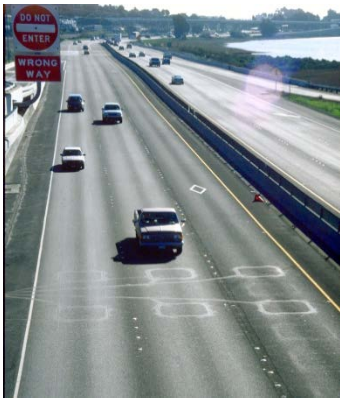


Figure 6-2: A typical array of inductive loops forming a Vehicle Detection Station

The backbone of Caltrans' traffic information infrastructure consists of approximately 25,000 inductive loop detectors statewide. These detectors pick up vehicle presence from small variations in the local magnetic field. In turn, this data is aggregated to provide vehicle counts over 30-second periods, along with the occupancy rate, which is the proportion of time that a section of road is occupied by a vehicle. When two inductive loops are coupled longitudinally on the roadway, they can jointly provide speed measurements which are calculated from the time difference between the detection events of the two loops. Inductive loop technology has been available since the 1960s and has dominated traffic information ever since because of its high reliability and relatively low cost compared to other technologies. Its main drawback is that it must be buried in pavement. The installation costs are significant and the loops are subject to fracture from pavement deformation.

Caltrans also employs other technologies for traffic detection. These include multiple types of radars that can be installed off-pavement and provide functionalities similar to inductive loops, but whose reliability suffers from calibration errors and drift. Wireless in-pavement detectors are another popular choice. These are a modern version of inductive loops with lower installation costs, lesser vulnerability to pavement deformations, and more flexible deployment options.

Most traffic detectors that are deployed on Caltrans' right-of-way are procured, installed, operated, and maintained by Caltrans or one of its contractors. However, there are a few notable exceptions:

- In the San Francisco Bay Area, approximately 100 toll tag readers pick up the passage time of vehicles carrying FasTrak transponders. These passage times are turned into travel time measurements between key locations. The readers and the data processing are owned and operated by the Metropolitan Transportation Commission's (MTC) 511 program, but the estimated travel times are available to Caltrans.

- MTC's 511 program was also the first local agency to purchase traffic speed measurement radars from a Silicon Valley company named SpeedInfo. The Santa Barbara County Association of Governments similarly bought and deployed 30 of those sensors on US-101 in 2009.

- Through the Federal Intelligent Transportation Infrastructure Program (ITIP), several metropolitan areas partnered with Traffic.com (now part of NAVTEQ, itself a subsidiary of Nokia) to install federally funded traffic radars. These radars are operated by NAVTEQ, and the data is licensed to Caltrans and other public agencies under the terms of the ITIP program.

Caltrans has had difficulties maintaining its traffic data collection devices;[11] available resources are directed in priority to mission-critical systems such as traffic signals. As of this writing, about one third of traffic detectors statewide are not reporting data on a given day.[12]

2.2.2 TRAFFIC DATA PROCESSING AND DISSEMINATION

Field traffic monitoring devices are connected to telecommunications networks of various kinds so that the data they collect can be transmitted to one of Caltrans' twelve Transportation Management Centers (TMC), one for each Caltrans district. The detailed implementation of traffic data processes varies by district, but the high-level functions are consistent. Figure 6-3 provides an overview of those functions. In each TMC, the data collected from field devices is first treated by a Front End Processor (FEP) which serves as an interface between the traffic detection infrastructure and software applications.

[11] Ram Rajagopal and Pravin Varaiya, "Evaluating the Health of California's Loop Sensor Network," Transportation Research Board, 89th Annual Meeting (January 11-15, 2009)

[12] California Freeway Performance Measurement System (http://pems.dot.ca.gov/), consulted on December 12, 2012.

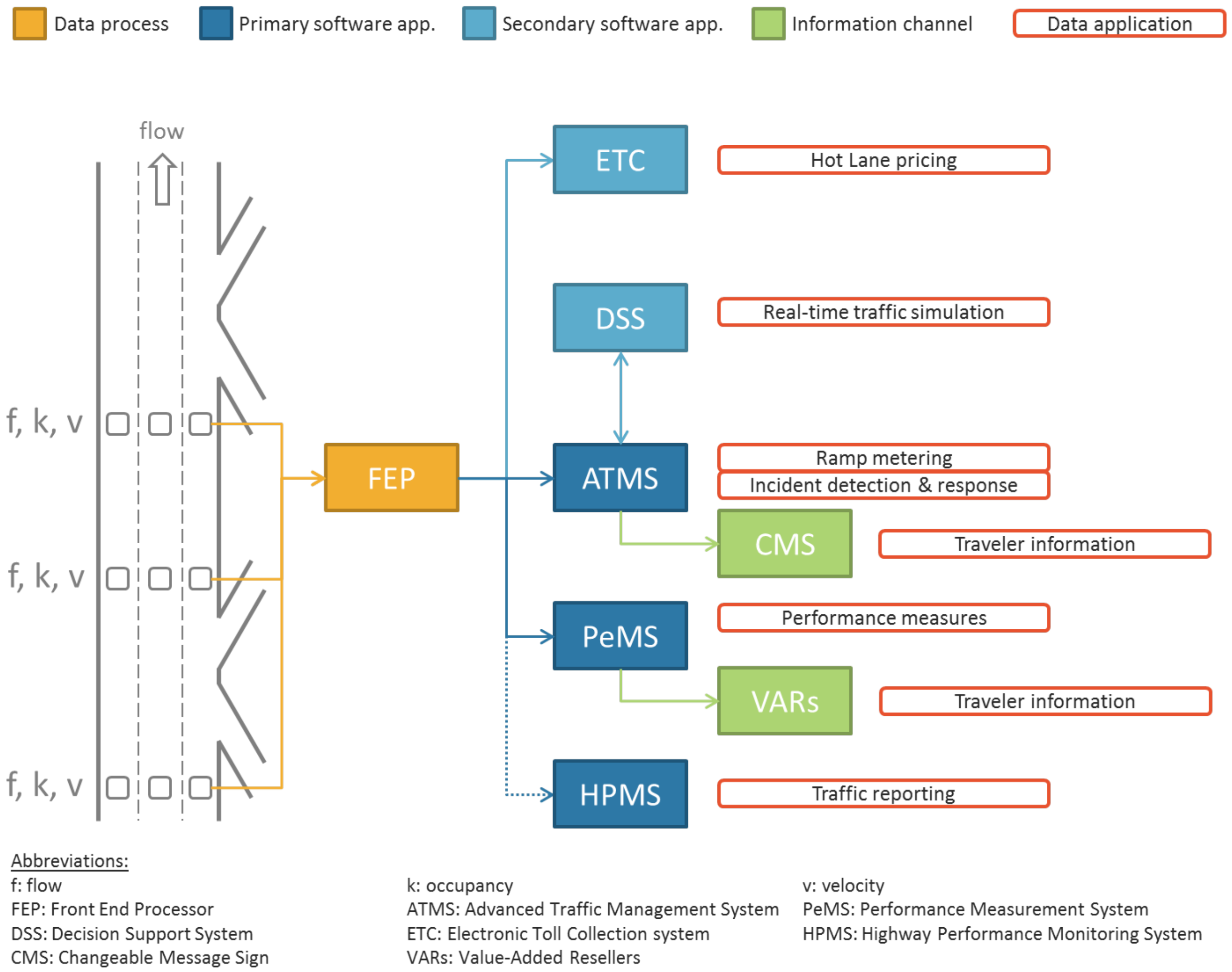


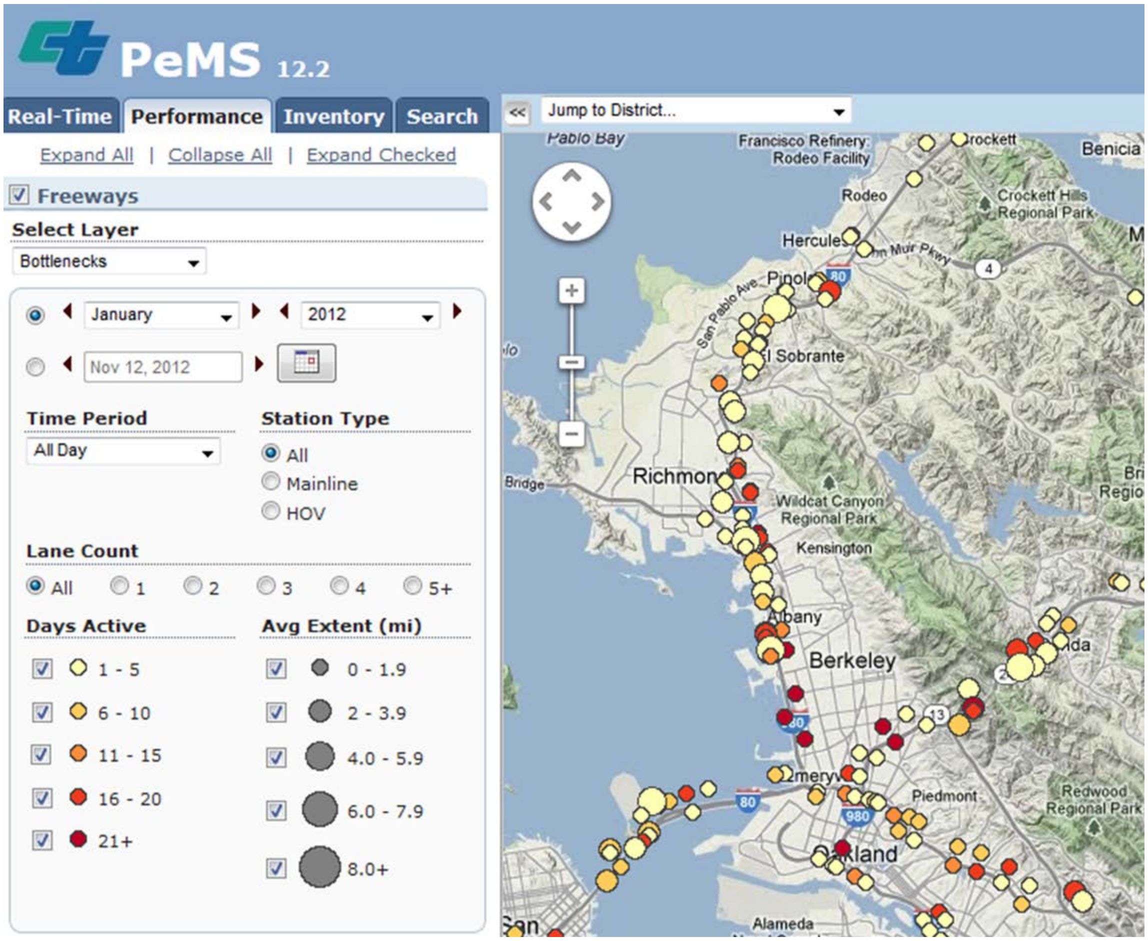
Figure 6-3: Traffic data processing and applications

The most important of these applications are as follows:

- **Advanced Traffic Management System (ATMS):** The ATMS is the core operating system that Caltrans districts use for managing traffic. It gathers traffic information in real time and enables TMC operators to monitor current conditions and act as needed, for instance by updating changeable message signs. The ATMS also computes and executes traffic control actions automatically, such as metering highway access based on pre-set algorithms.

- **Performance Measurement System (PeMS):** Data from each district's FEP also flows to PeMS, a statewide archival, retrieval, and data analytics platform that has been storing California traffic information for over a decade. PeMS also serves as the interface to private Value-Added Resellers (VARs) of traffic information who use it as a one-stop shop to obtain up-to-date California freeway data.

- **Highway Performance Monitoring System (HPMS):** HPMS is a federal system that provides consistent reporting across state DOTs to the US DOT. The dotted line in Figure 6-3 indicates that there is not a direct data connection from the FEP to HPMS, but traffic volume information is input into HPMS.



☑ Freeways

Select Layer
Bottlenecks

● January 2012
○ Nov 12, 2012

Time Period
All Day

Station Type
● All
○ Mainline
○ HOV

Lane Count
● All ○ 1 ○ 2 ○ 3 ○ 4 ○ 5+

Days Active		Avg Extent (mi)	
☑	● 1 - 5	☑	● 0 - 1.9
☑	● 6 - 10	☑	● 2 - 3.9
☑	● 11 - 15	☑	● 4.0 - 5.9
☑	● 16 - 20	☑	● 6.0 - 7.9
☑	● 21+	☑	● 8.0+

Figure 6-4: A screenshot taken from PeMS, showing significant freeway bottlenecks over a period of a year

Although not currently deployed by Caltrans, two additional roadway management applications make use of traffic information:

- **Decision Support System (DSS):** The evolution of traffic operations toward Active Transportation Demand and Management (ATDM) strategies and technologies is setting the need for sophisticated software tools that can simulate traffic scenarios and tactical response in real time. Two projects in California exemplify that trend: the federally-funded I-15 Integrated Corridor Management (ICM) demonstration in San Diego, and the I-80 ICM project in the San Francisco East Bay. Both projects aim to actively manage traffic events by coordinating ramp meters and traffic signals over wide areas and computing alternative itineraries that are suggested to motorists via dynamic signage. The I-15 project will run simulation software during regular operations to determine optimal strategies, requiring up-to-date traffic information. These projects are deployed on Caltrans' right of way, but they are led by local agencies (the San Diego Association of Governments and the Alameda County Transportation Commission, respectively).

- **Electronic Toll Collection (ETC):** Unlike traditional electronic tolling on turnpikes and bridges, high-occupancy tolling (HOT) projects typically resort to dynamic pricing that is set based on current traffic conditions. In essence, the pricing adapts to ensure that the HOT lane never becomes congested, thereby delivering commensurate value over general purpose lanes. HOT lanes with dynamic pricing are operated or being rolled out in the San Diego region, the Los Angeles region, and the San Francisco Bay Area, under the authority of Metropolitan Planning Organizations (MPOs) or county Congestion Management Agencies (CMAs).

In sum, there is a very rich suite of traffic management applications that rely on accurate, timely traffic information to ease congestion, both now and in the future. These applications are primarily operated by Caltrans but also increasingly by its local agency partners.

2.3 TRAFFIC DATA COLLECTION METHODS

The impetus for this roadmap document is that over the past decade the availability of options to collect traffic information has multiplied. First, sensing technologies have evolved significantly, resulting in a broad array of new products that offer improved price-to-performance ratios. Second, the growth of cellular telecommunication networks and the advent of smartphones have opened a new source of data, commonly referred to as crowdsourcing. As a result, we can classify available traffic data collection methods into three broad categories, each with its own strengths and limitations.

2.3.1 POINT-BASED COLLECTION METHODS

Inductive loops, magnetometers, and radars that form the bulk of Caltrans' traffic detection infrastructure fall in a first category that we call point-based. Point-based data collection methods employ sensors that pick up traffic volumes and/or speeds at one dedicated location. Other technologies that are used to collect point-based traffic data include video cameras, piezoelectric sensors, and acoustic sensors.

The strength of these methods is that they provide a comprehensive survey of all vehicles passing by a given location, and therefore a reliable measure of volume and speed—within the capabilities of each technology. The disadvantage is that traffic managers remain blind to what takes place between those locations. Various studies have attempted to determine the optimal spacing between vehicle detection stations, with estimates ranging between ¼ and ½ mile[13]. However these estimates must be corrected to take detector failures into account. At a 30% malfunction rate, the effective number of stations required to obtain reliable information rises to 3 to 5 per mile.

2.3.2 SEGMENT-BASED COLLECTION METHODS

The second category of traffic data collection methods provides trip times for preset road segments. Segment-based data collection is achieved by vehicle reidentification, that is, the ability to uniquely match records of a traveling vehicle obtained at two different locations. In practice, this can be done in one of four ways:

i. **Toll tag readers:** Toll tag readers can acquire a unique identifier stored on transponders that vehicles carry to pay electronic tolls. This technique is notably used by MTC's 511 program in the San Francisco Bay Area and by the TranStar program in the Houston, Texas, area.

ii. **License plate readers:** A combination of video cameras and online character recognition software is able to extract and store license plate numbers. This technology is widely used for automated ticketing linked to toll enforcement, but it can be repurposed to capture trip times between two or more locations.

iii. **Magnetometers:** Besides being able to pick up vehicle presence, magnetometers and inductive loops can also measure a more detailed set of magnetic field disturbances associated with each vehicle. This data can be turned into a "signature" that is unique enough to perform reidentification. Sensys Networks in Berkeley, California, has deployed this technology on signalized arterial streets around the United States for a few years, and Caltrans has been funding research to make its freeway inductive loops more capable thanks to this technique.

iv. **Bluetooth readers:** Roadside Bluetooth readers are a relatively recent innovation, dating back five years or so as of this writing. They exploit the massive growth in personal electronic devices such as GPS units and smartphones that contain Bluetooth technology. This technology makes those devices "discoverable" by readers, which means that they transmit a unique identifier even though no meaningful exchange of data may take place. That identifier is exactly what is needed to perform reidentification and travel time estimation. Several vendors are marketing this technology, which is primarily used on signalized arterial streets.

Each of these methods provides accurate travel times between two locations. The number of vehicles that get reidentified varies with each technology (toll tag readers and Bluetooth readers require vehicles

[13] See for instance X. Ban et al. "Optimal sensor placement for freeway travel time estimation," presented at the 18th International Symposium on Traffic and Transportation Theory (July 2009).

to carry compatible equipment, which restricts the pool; license plate readers and magnetometers suffer from misreads which effectively reduce the number of matches), but in practice the sample size is almost always sufficient to provide a reliable median travel time. Just like the point-based collection methods, segment-based methods require dedicated physical infrastructure.

2.3.3 MOBILE DATA SOURCES

The third available category of traffic data collection methods relies on mobile data networks to extract the velocity of individual vehicles. This offers two key advantages: 1) no infrastructure deployment is necessary (save for cellular network infrastructure, but that is exogenous), and 2) data can be obtained from virtually any location on the roadway network, as long as there is cellular coverage. Mobile data sources can be further divided into three methods:

i. **Cellular network data analysis:** This method first appeared around 2000 and is offered commercially by a few vendors today. It requires a deep partnership with one or more cellular operators and relies on direct analysis of the flow of connected phones through the cellular network. In practice, phones cannot be located very accurately with this method, and therefore traffic speed estimates tend to lag, although they do capture significant events such as major bottlenecks or accidents. On the other hand this technique has great potential to measure travel demand in terms or origins and destinations.

ii. **Fleet telematics:** Satellite communications and cellular networks have enabled operators of vehicle fleets (including commercial trucking operations, taxi fleets, transit bus fleets, etc.) to track each vehicle’s position (provided by Global Positioning Sytem (GPS) receivers), often in real time. Many of these fleets agree to let traffic information aggregators use that data to estimate current traffic conditions and archive it for historical reference. One of the drawbacks with this technique is that professional fleets tend to follow specific driving patterns not necessarily representative of general public road users.

iii. **Smartphone applications:** Thanks to GPS and other technologies, smartphones can determine their own location. As a result, most providers of mapping or traffic applications poll location data from their customers, typically at fixed time intervals that may vary from one second to one minute. This technique clearly has the most long-term potential since over 50% of the US adult population now has a smartphone—and that number could approach 100% within a few years. There are several limitations, however. One limitation is that polling is restricted to phones on which the provider’s application is currently running. In effect, only the most often used applications such as Google Maps may generate sufficient data to produce consistently reliable traffic estimations over an entire network. The second limitation is that providers must tread carefully with privacy concerns. The fact that data gets polled only while an application is in use offers a quid-pro-quo which has functioned so far. Yet more systematic data collection schemes may be hard to implement without explicit user consent. Finally, data collection from smartphones takes place in a competitive environment. While there is little doubt that the data collected by all application providers put together could serve today's traffic information needs, that aggregation is unlikely to happen.

Traffic information based on mobile data sources is almost exclusively available from private sector providers, which is a departure from historical practice that was more akin to meteorological information: The public sector was collecting raw data which private companies could package for consumers. Now a reversal of roles is taking hold: Private sector companies have access to traffic data that is of interest to public sector organizations who manage roadway networks. So far the information available from mobile data sources has been limited to traffic speeds, and volume estimation remains difficult to do. This may change, however, as cars themselves become connected to wireless networks.

2.3.4 SUMMARY

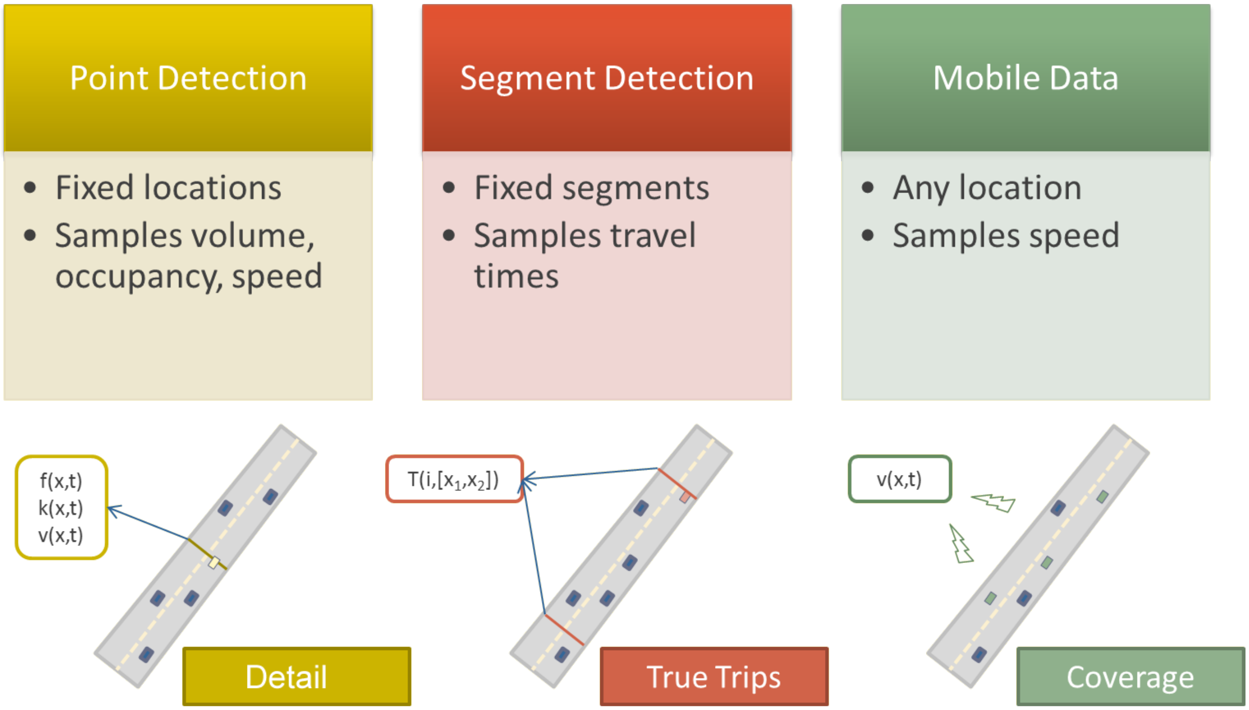


Figure 6-5: A summary comparison of traffic data collection methods, highlighting the key advantage of each one

It truly can be said that each category of traffic data collection methods has its own strengths. Only point-based collection provides reliable traffic counts. On the other hand, producing accurate travel time estimates between two network locations from point-based data has always proven difficult because it offers no information about conditions between sensors. This can be overcome by installing a high density of detectors, but that obviously comes at a cost.

Segment-based detection works effectively to generate traveler information or establish performance measures, particularly on signalized arterials where traffic signal timing plays a determining factor in maintaining adequate flows. It is superior to mobile data sources in that it normally offers much higher sample sizes.

Finally, mobile data offers unprecedented appeal to traffic managers in that it provides ubiquitous coverage with no need for infrastructure deployment and maintenance. As of today, the data is limited to speed estimates, which makes it most suitable for traveler information. Yet even traffic management applications can benefit from adding mobile data into the information mix, as described in the next section.

3 HYBRID SYSTEM: OBJECTIVES AND IMPLEMENTATION

3.1 DEFINITION AND VISION

A hybrid traffic data environment would result from the incorporation of third-party data into transportation management systems in complement to the data currently collected by Caltrans. If employed sensibly, this strategy could augment information availability and quality while costing less than current business practices. In turn, increasing access to information would improve the ability of Caltrans and its local partners to manage roadway traffic in ways that balance demand and flow in order to minimize delays and make the most of their infrastructure investments.

Today, the great majority of traffic data that is used to manage roadway operations around the state is collected by detectors that are owned and operated by Caltrans. In fact, Caltrans has been a national leader among state DOTs in deploying traffic monitoring technology in a systematic way. Close to a third of all freeway detectors in the United States are in California. However, there are now several reasons why this model needs to be reexamined.

The first and most obvious reason is that new technology and new business models are now available to Caltrans. As this roadmap aims to demonstrate, there are circumstances under which incorporating mobile data sources into Caltrans' traffic information infrastructure would be a net benefit. Mobile data sources can complement and in some instances substitute for detector data with less cost. A second reason is that under current practices, Caltrans is not able to keep up with the data requirements that ATDM strategies impose. Freeway mainlines in urban and suburban locations are well instrumented, but ramps and signalized arterials for the most part are not. In order to support HOT lane and ICM projects, local agencies are deploying their own data solutions, which are not always integrated into Caltrans' information flows. Therefore, there is both a need to scale up data collection more rapidly and to make information systems more open so that they can benefit from investments made by local agencies. A third reason is that Caltrans has struggled to keep existing detectors running. A traffic detector that is broken or not transmitting is like no detector at all. No more than two-thirds of traffic detectors statewide are up on any given day[14], although this figure varies by district. Unless more resources can be committed by the Department to detector maintenance, Caltrans would be better off with fewer detectors so that all can be operated nominally. Alternative solutions would be selected in areas not covered by detectors, leading to a hybrid traffic data collection system.

In effect hybridization is already happening, as demonstrated by the examples provided in section 2.2.1. However, this is the result of isolated initiatives, not a system-wide program. Establishing a hybrid data roadmap consists of acknowledging the opportunities provided by the diversification of traffic data sources and outlining a course of action that Caltrans could take to benefit broadly and systematically.

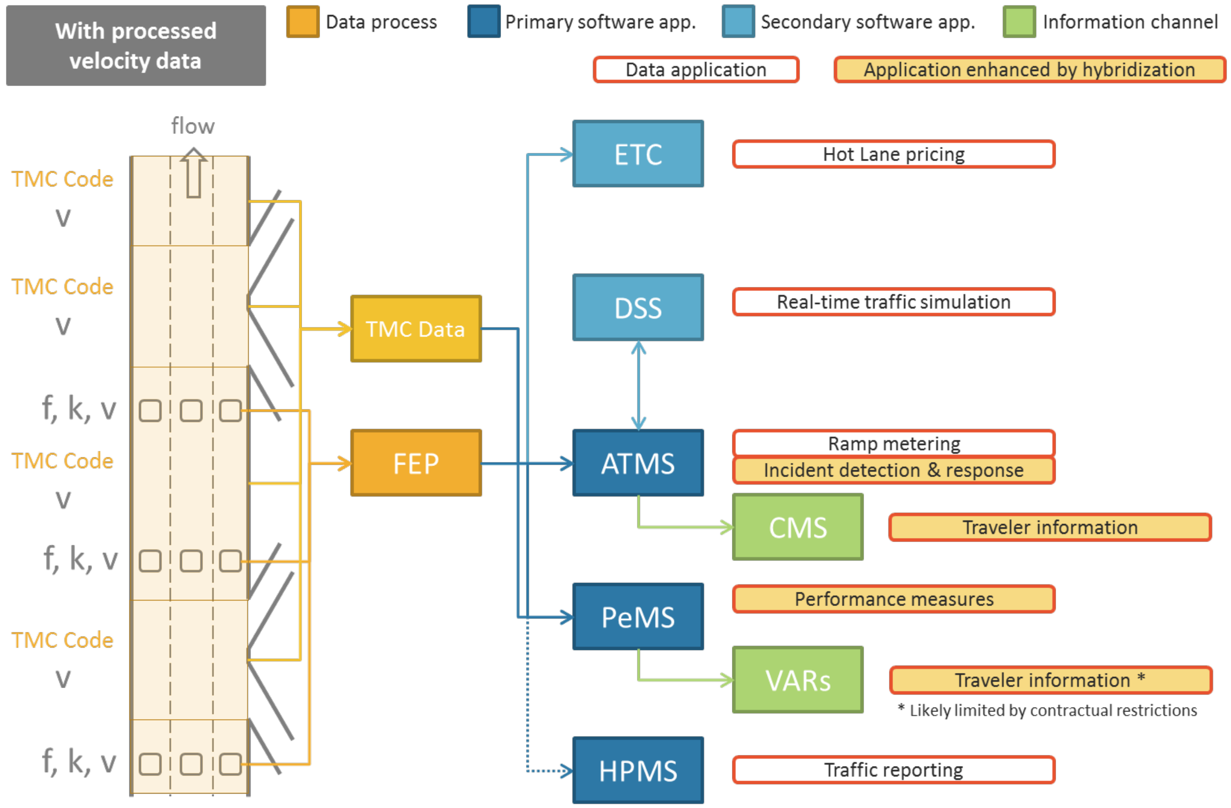
[14] Op. cit., California Freeway Performance Measurement System

This roadmap document focuses somewhat more intently on the specific opportunity afforded by the availability of mobile data provided by private enterprises. Today, that mobile data is primarily marketed as a source of traveler information. It is usually reported as aggregated velocity data for set roadway sections (the industry standard is so-called "TMC location codes"[15]). A somewhat more exploratory alternative that was tackled by PATH's hybrid data roadmap project consists of processing individual vehicle observations that are "fused" with detector data in a generalized traffic flow model. Data fusion refers to the ability to create a single data set from multiple sources, in such a way that these sources correct and complement one another. While the use of individual vehicle observations raises privacy issues, this granularity of data offers significantly more potential to improve the accuracy of traffic state estimates and also provides a pathway to the measurement of travel demand and origin-destination information, which is critical to the deployment of ATDM strategies. These two scenarios are further described in the subsections that follow.

[15] TMC stands for Traffic Messaging Channel, a semantic protocol for broadcasting traveler information

3.1.1 USING AGGREGATED VELOCITY DATA

In the first case, illustrated by Figure 6-6, the supplemental data is aggregated velocity data, typically reported by TMC location. This is the data commonly offered by the primary providers of traffic data services that use mobile sources. It provides overlapping values for segment speeds that may or may not be consistent with existing fixed sensors. (This type of data was not considered in detail by this study, and no recommendations are explicitly made here.)



Abbreviations:

f: flow
FEP: Front End Processor
DSS: Decision Support System
CMS: Changeable Message Sign

k: occupancy
ATMS: Advanced Traffic Management System
ETC: Electronic Toll Collection system
VARs: Value-Added Resellers

v: velocity
PeMS: Performance Measurement System
HPMS: Highway Performance Monitoring System
TMC: Traffic Messaging Channel

Figure 6-6: Traffic data processing and applications with procured velocity-only data

Under this scenario, aggregated velocity data might be merged with traditional detector data at the application level in PeMS or ATMS and used to augment and possibly improve extant roadway speed estimates (although a proof of concept would be required). However, such data is not helpful in improving volume and occupancy estimates, which are critical for the most stringent freeway management applications where an underlying flow model is required. As a result, a limited range of applications might be enhanced. These include:

- **Incident detection and response:** Traditional incident detection procedures rely on abrupt changes in the speed gradient along freeway sections. Under that assumption, improvements in speed estimates may help detect incidents faster and more reliably. However, the advent of cell phones has changed this paradigm to some extent: Caltrans and CHP typically get notified of traffic accidents through phone calls that come faster than the algorithms can operate.

- **Traveler information:** The primary purpose of aggregated mobile data is to provide traveler information. As a result of purchasing this data, Caltrans might be able to improve the travel time estimations it displays on Changeable Message Signs (although a proof of concept would be required). There are, however, some caveats. In the San Francisco Bay Area, the data displayed on those signs is provided by the local 511 service, which itself purchases data from third-party vendors. In addition, the other prong of the traveler information application, disseminating data through PeMS, is likely to be restricted by vendors' licensing terms: Making data available through PeMS would serve it to those vendors' competitors. Furthermore, Caltrans' role in providing traveler information is somewhat small today. That role has been deliberately handed over to local agencies and the private sector. Finally, this use case does not leverage existing investment in fixed infrastructure. Indeed, it raises the question of what the best policy is when discrepancies appear between measurements from existing fixed sensors and estimates from aggregated velocities from a third party?

- **Performance measures:** To some extent, performance measure calculations might benefit from more accurate speed estimates. This is true in particular of the mandatory HICOMP reporting from Caltrans to the legislature, which sets a threshold of 35 mph to define congestion. Supplemental speed measurements might also help with more accurate and reliable identification and assessment of bottleneck formation.

In summary, while aggregated mobile data has been a great boon to traveler information services, it would only bring limited benefits to Caltrans given its core mission.

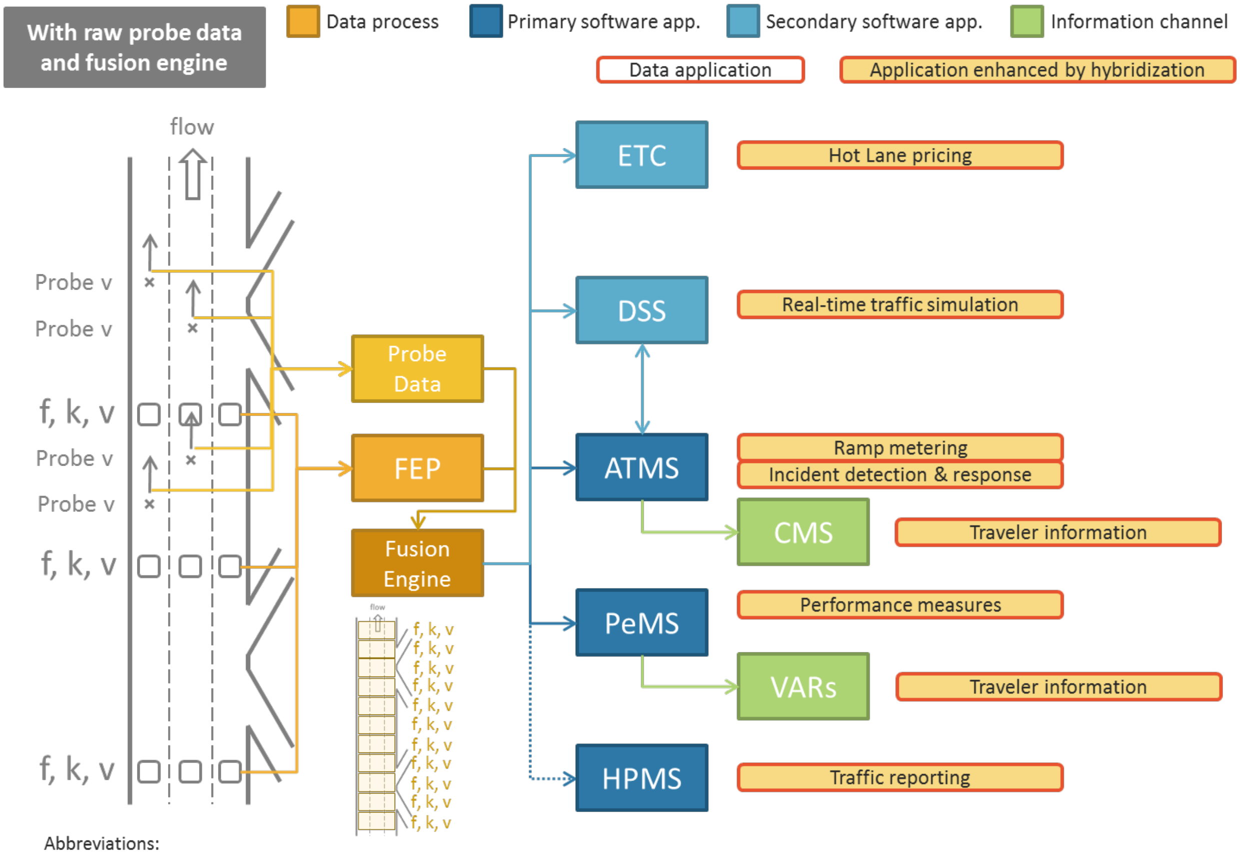
3.1.2 INCORPORATING DATA FROM MOBILE SOURCES

The second scenario is the incorporation of "raw" or unaggregated data coming from mobile probes. This scenario is depicted in Figure 6-7. Obtaining individual vehicle observations provides significantly more flexibility in how to process the additional data. In particular, it enables the calibration and operation of a full-fledged traffic state estimation model similar to the one implemented by PATH as part of the Mobile Millennium and hybrid data roadmap projects. This is the working assumption adopted in Figure 6-7, which shows the data from mobile probes being fused with detector data upstream of freeway management applications, as part of a new module labeled "fusion engine." That engine turns disparate data sources into a complete representation of the highway traffic flow, yielding estimated volume, occupancy, and velocity values for each “cell” in a partition of the road network.

The advantage of such a scheme is that it would require only minimal design changes to the freeway management applications themselves, since they would receive similar data to the one provided by traffic detectors today. There are still a few differences:

a. The number of cells would be vastly greater than the number of existing traffic detectors. This is overall good news for running those traffic management applications, provided that data quality does not suffer in the transition. However, this may weigh on computing performance.

b. While the data could be presented in the same way as detector data, its essential nature would be different. Rather than correspond directly to an identifiable source, it would result from a complex algorithm that pulls from multiple sources. This means that robust data quality diagnostics should accompany it, and that freeway management applications should be modified to have the capability to take those diagnostics into account.



Abbreviations:
f: flow
FEP: Front End Processor
DSS: Decision Support System
CMS: Changeable Message Sign
k: occupancy
ATMS: Advanced Traffic Management System
ETC: Electronic Toll Collection system
VARs: Value-Added Resellers
v: velocity
PeMS: Performance Measurement System
HPMS: Highway Performance Monitoring System

Figure 6-7 - Traffic data processing and applications with procured unaggregated probe data

The great benefit of this approach is that all applications currently deployed by Caltrans could be enhanced. All would be fed by a common traffic flow model that contains a complete description of the state of the network. Of course, the benefit would only be as great as allowed by the quality of that traffic flow model, which would itself depend on the accuracy of incoming data sources and the intelligence of the fusion engine. Such a fusion engine does not exist commercially today, but numerous efforts to design it are underway in the academic world. For that reason, a gradual roadmap as described further in this document should be employed in order to give life to a beneficial hybrid data environment.

3.2 IMPLICATIONS AND BENEFITS

In practice, a hybrid data environment would be manifested in different ways depending on the local context and objectives, future capabilities of transportation information systems, and the available supply of third-party data. The following is a list of prominent examples or “use cases” that would offer benefits:

- **Filling gaps in the current traffic detector infrastructure and/or relaxing maintenance on secondary detection stations:** This use case is premised on the fact that the full array of data provided by fixed detectors may not be needed with a very high spatial resolution. As was pointed out in section 2.3.1, three to five detection stations per mile are often necessary to estimate accurate travel times. However, travel times on freeways can now be provided with often greater accuracy by mobile data sources. What mobile data sources lack is traffic volume information, which still needs to be collected by physical detectors. But traffic volumes propagate on a freeway in very predictable patterns. If that becomes the rationale for having traffic detectors, then one station every three to five miles (rather than three to five per mile) could often be sufficient. In effect, where detectors are currently this sparse, mobile data sources could fill the information gap. On roadway segments where detector density is high, some stations could be allowed to fail and not be repaired or replaced if mobile data sources were available. This strategy would save money and enable Caltrans’ maintenance workforce to focus on the most critical detectors.

- **Increasing coverage and providing uniform congestion reports:** Mobile data sources present the unique advantage of covering the entirety of the roadway network. This includes roads that historically have not been instrumented with any detection, such as rural highways and signalized arterial streets. The transportation performance of signalized streets is becoming more important as advanced traffic management strategies in urban areas seek to balance demand and flow across all parts of the network. Another feature of ubiquitous coverage is that it provides de facto uniform reporting on the state of the road network. Currently, Caltrans relies on a mixture of traffic detectors that are unevenly deployed, and ad-hoc vehicle runs that are both costly and inadequate representations of year-round conditions.

- **Enhancing information with data fusion:** Where both detector and third-party data is collected, Caltrans' information systems could make the whole superior to the sum of the parts by fusing these data. As a result, greater value can be extracted for the same cost. As described in section 3.1, mobile data can supplement detector data in one of two ways. Most information providers that employ mobile data sources broadcast an estimated speed on discrete road segments. This information could augment traditional detector data to fill gaps in detection coverage and enhance traffic performance measures to some extent. However an even greater opportunity lies in fusing detector data directly with the location data polled from individual mobile phones.

The transition from current business practices to a hybrid data environment involves numerous changes, both organizational and technical. On the whole, these changes would be largely beneficial to Caltrans, but some of them will be challenging to carry out because they require new ways of thinking, experimentation, and even missing skills. This section summarizes key implications of a hybrid data environment and further highlights their benefits and pitfalls. Some of the changes will need to be voluntary on the part of the organization in order to initiate action, while others will take place organically once the transition is underway.

3.2.1 OUTSOURCING DATA COLLECTION

Purchasing third-party data requires particular contractual arrangements that are mostly foreign to Caltrans and state governments in general. Caltrans excels at designing roads and hiring builders based on a complete set of project plans with concrete milestones. By contrast, information is a somewhat intangible product that is delivered on a continuous basis. The novelty of purchasing information challenges the procurement process, and this severely hinders action. In 2005, a pilot assessment of third-party traffic data collected from cell phones became an element of the state governor's GoCalifornia initiative with a budget of $2 million. The publication of a request for proposals (RFP) took over two years, in part because of the insistence of the state's administrative officers to treat it as a software purchase. This interpretation meant that a Feasibility Study Report (FSR) should be written as a justification, an onerous procedure out of proportion to the scope of the project. When the RFP was finally published, the service specifications were written in such a way that the bidder considered most responsive was one that offered to install inductive loops. Caltrans management ended up deciding not to award a contract and returned the allocated funds to the state's budget.

Successfully procuring traffic data would require the following changes:

- Clear specifications must be articulated that reflect not only Caltrans' needs but also the ability of vendors to meet them.

- Procurement officers must be educated about the nature of the service in order to make adequate decisions that facilitate the administrative process.

- Contractual agreements must be written that reflect the particular features of the service, possibly including payment terms that are tied to continuous quality monitoring.
- Funds must be identified, which may mean redirecting some of the budget allocated to traffic detection on a somewhat permanent basis. This is guaranteed to upset those who normally spend that budget, but it could be attempted one district at a time based on voluntary participation.

These are difficult changes that can only take place with dedicated staff and upper management champions.

3.2.2 MANAGING NEW RISKS

Procuring traffic information from vendors introduces specific risks that must be taken into account. We outline three such risks: going concern risk, data quality risk, and privacy risk. Simple strategies exist to manage these risks and to prevent or mitigate their potential impacts.

Going concern risk

If Caltrans purchases part of its traffic information from a vendor, one obvious risk is that the contractor might go out of business. This would leave gaps in traffic information, some of which may be critical if the information is used to actively manage traffic (for instance, as part of ramp metering or signal coordination systems). However, there are some factors that limit that risk. The first is that the provision of traffic information, once the realm of start-up ventures, has evolved into a robust industry with relatively large players (Nokia and TomTom are billion-dollar, publicly traded companies). This makes business failures easier to anticipate and prepare for, and also means that alternative suppliers will likely be available should a contractor go out of business. Second, switching costs associated with changing suppliers need not be very high, given that the industry has historically adhered to standards for data publication (TMC location codes in particular, and more recently “TPEG”). Standards will evolve and some vendors may try proprietary approaches, but there are reasons to remain optimistic: a) Automotive manufacturers currently have the market power to demand standardization, and b) the future of travel services involves mixing disparate data, which sets an incentive among industry participants to maintain interoperability and compete on features and data quality.

In order to manage this risk, Caltrans could enact the following provisions:

- Start experimenting with hybrid data to improve information coverage rather than substitute for existing coverage. This way, the risks associated with contractors’ defaults will have limited consequences.
- If and when hybrid data is employed in critical traffic management systems, monitor contractors’ financial health and require periodic access to books as necessary. Penalties can be

imposed on large companies that decide to interrupt or divest their traffic information service while remaining in business.

- Monitor industry standards and adhere to them as much as necessary to minimize switching costs in the event of a contractor's inability to provide continued service.

Data quality risk

A more specific risk resulting from the ongoing provision of traffic information by external suppliers lies in a relative loss of control over the quality of the data. Good data quality on a portion of the network and for certain time periods does not guarantee that this quality pervades other portions of the network and time periods. Therefore, measuring quality on sample data will not sufficiently determine overall data quality. This is especially true of information derived from mobile data, whose quality constantly depends on the number of connected users at a given location and time, and can therefore vary widely.

The value of a traffic information service is directly tied to its accuracy and timeliness, and therefore understanding and measuring data quality is absolutely integral to the success of the hybrid traffic data model. Accordingly, sections 4 and 5 of this document provide more in-depth treatment of data quality monitoring and its inclusion in procurement contracts.

Privacy risk

Finally, the collection of traffic data from personal mobile devices creates risks around individual information security and invasion of privacy. Under any realistic scenario, Caltrans would not be collecting this data directly. Vendors would provide anonymous data that is used to predict and manage roadway conditions, while omitting or masking personally identifiable information. However, this does not annihilate all risks. If there is any way to derive the whereabouts of individuals, even if only with sophisticated algorithms, from data that Caltrans has stored or used at some point, then the potential for liability exists. The risk could be manifested either by a security breach into Caltrans' information systems or by a legal subpoena. The former could expose Caltrans to civil lawsuits, while the latter could engender public outcry.

Traffic data sources that rely on vehicle tracking (such as certain segment-based data collection methods and mobile data sources) can be deceptive in that stripping them of identifying data fields does not automatically render them anonymous. To take an extreme example, if a tracked vehicle drives on a rural road on the Modoc Plateau, anyone who has evidence of the presence of that vehicle there would be able to identify it in a data log with a high degree of certainty and then probably follow its itinerary over any number of miles. Making traffic information truly anonymous requires some additional, proactive steps to ensure that the data cannot be "reverse-engineered."

The possibility to mine processed traffic data to identify the source means that using that data poses non-negligible risk. To minimize this risk, Caltrans would need to impose a set of minimum data management practices on its contractors. Subcontractors should be targeted as well because the provision of traffic information often relies on the aggregation of many different sources. Failure to render any one source's data anonymous can propagate to the final product. There is no published and broadly recognized method for post-processing traffic data in order to make it truly anonymous, but routines exist to do so that can provide good results without too much complexity. As the example above illustrates, the key is to avoid situations in which vehicles can too easily be singled out, and this can be done on the basis of relatively simple criteria. Other risk-mitigating factors would include: a) a data retention policy that limits its availability on Caltrans' servers to short amounts of time, except in an aggregated form, and b) data security measures that minimize the possibility of capture by a malicious third party.

3.2.3 DESIGNING AND IMPLEMENTING NEW INFORMATION SYSTEMS

The bulk of new data that a hybrid collection system would bring to Caltrans would originate from mobile sources. This data only provides speed information, unlike inductive loops and other types of detectors that capture traffic volumes and occupancy rates. Traffic volumes may be estimated from mobile data sources if enough vehicles are reporting, but that is still a long way off, and at any rate this volume of information would likely require a different treatment than the direct measurements performed by highway sensors. In either case, a crucial consideration for Caltrans and other traffic management agencies is that the traffic data available from private providers is different from the detector data upon which they have built existing traffic management systems.

As was illustrated in Figure 6-3, information systems such as ATMS and PeMS read streams of volume, occupancy, and speed data collected at fixed locations. Applications such as ramp metering algorithms or the calculation of freeway performance metrics are therefore designed to use those inputs. By contrast, data captured from mobile sources may come in either one of two forms: unaggregated data consisting of single speed observations at arbitrary network locations or aggregate data consisting of a prevalent speed estimate for set sections of roadway. There are multiple reasons why ATMS or PeMS cannot readily use this data:

- A lot of the algorithms use volume or occupancy as inputs, which makes the speed data useless, even though it contains valuable information. To effectively capture that information would require new algorithms.
- The spatial referencing systems used by existing traffic systems to represent traffic states may differ from those used by industry vendors, requiring reconciliation.
- If vendor data is used to supplement existing detector data, a fusion engine must first be devised that merges both sources of data into a single set of traffic information.

The need to redesign the information systems that support traffic management functions is probably the highest hurdle to overcome in order to take advantage of a hybrid data environment. It largely explains why mobile data sources have been so successful at providing travel information to consumers in the last five years while not making any inroads in the field of traffic management.

3.2.4 ADOPTING A NEW DETECTOR STRATEGY

One of the key reasons to migrate to a hybrid traffic data system is to reduce the long-term costs associated with the installation, operation, and maintenance of traffic detectors. Therefore, a hybrid data roadmap would have to incorporate a new detector strategy. What this means is a reassessment of what technologies to invest in, where to place new detectors, and which detectors to maintain in priority. While it is not the object of this roadmap document to fully articulate what that new detector strategy should be, we can outline its tenets.

The majority of Caltrans' traffic detectors are installed on freeway lanes. This has made sense historically because this is where traffic volumes are highest and therefore where congestion has the most adverse impact, which justifies monitoring and countermeasures. However, if Caltrans adopts a hybrid traffic data strategy by adding mobile sources, investments in traffic detectors should deemphasize freeway mainline and concentrate instead on freeway ramps (both on-ramps and off-ramps) and signalized intersections. Mobile data sources provide the most coverage where traffic volumes are highest, and in fact they can already offer adequate coverage on urban freeways today. Further, flow models of freeway mainline traffic can be calibrated from a combination of mobile data sources that provide speed information only and sparse detectors that collect volume and occupancy as well. This argument is developed in section 4, and it basically suggests that Caltrans could obtain the same quality information with a vehicle detection station every 3 to 5 miles complemented by mobile data sources as it gets with a vehicle detection station every half-mile. Conversely, mobile data sources cannot do a good job of monitoring individual ramps where volume information is critical to metering algorithms and other freeway management applications that require precise demand measurements. Likewise, signalized intersections on major thoroughfares operated by Caltrans or its partners lack traffic detection that could help optimize signal timing and coordinate daily operations between freeways and surface streets. It could be a long time before mobile data sources offer a credible alternative to infrastructure sensors in those settings.

In a broad sense, the detector strategy that should accompany the transition to hybrid traffic data collection system should play to the strengths of each data source. Mobile traffic data sources shine in terms of ubiquity of coverage. However, they only provide speed information, and the coverage is unequally distributed as it hinges on traffic volume. Ideally, the use of traffic detectors should be designed to complement these attributes by leveraging their own strength, which is detailed data, at locations where it matters most. If mainline information can be derived from fewer detectors thanks to the addition of mobile data sources, then the next locations of interest are ramps and intersections. In terms of technological choices, this also means that detectors should be selected for their ability to accurately measure traffic volumes, whereas speed measurements should become less important. In particular, the practice of installing dual inductive loops to obtain speed readings may be abandoned or at least moderated.

3.2.5 REAPING THE BENEFITS

Sections 3.2.1 through 3.2.4 described technical and organizational changes that are prerequisites to the transition to a hybrid traffic data collection system. In this section, we bundle a few additional recommendations that may be less critical elements of the roadmap but are still important if Caltrans wants to reap the benefits of adopting a hybrid data strategy. These recommendations include: i) selective maintenance, ii) traffic management augmentations, and iii) an open information architecture. The corresponding benefits all tend to a clear outcome: reduced costs and increased utility of traffic information.

Selective maintenance

This recommendation echoes the need to develop a new detector strategy which is described in section 3.2.4. In the margins of that new detector strategy, Caltrans could start saving money short-term by allowing a certain number of freeway mainline detectors to fail. In other words, the new detector strategy could be kick-started by selectively maintaining detectors based on their location and criticality to traffic management applications.

Traffic management augmentations

On the other side of cost savings, a key benefit of a hybrid traffic data strategy is to increase the usefulness of information. This means improving traffic safety and flows wherever possible by using available data, whether in real time to power adaptive traffic management systems, or off-line to optimize tactics and the settings of traffic systems. However, lack of data is not the only barrier to more responsive and effective traffic management systems. Legacy technologies as well as limited connectivity between traffic management centers and field elements are additional limitations that would need to be addressed in order to create benefits. Short of doing that, richer information may not always translate into safer and more fluid roadways.

Open information architecture

Section 3.2.3 highlights the need to redesign certain information systems in order to create the capacity to ingest hybrid traffic data, some of which does not originate from fixed detection stations. This need can be generalized into a recommendation to set a direction toward a more open information architecture. In effect, speed and flow data are only one of the many sets of data that is required for comprehensively managing transportation networks. All kinds of operational data emanating from local agencies could be added to the picture. Information about special events or school and employer schedules also needs to be incorporated for integrated corridor management. The future of traffic management systems, and indeed of any other information system, involves transactions with third-party systems of various types.

Tending toward this direction will generally help Caltrans realize the transition to a hybrid traffic data collection system, and the proper architectural choices may reduce the costs of implementing and maintaining gateways to data providers. In the same vein, accepting vendor data into traffic management systems can be seen as the first step toward a more open environment in which data is traded back and forth with multiple vendors, local agencies, and even the traveling public. This more open environment could lead to dramatic traffic management performance improvements over a 5-to-10-year horizon. PATH's *Connected Corridors* project, sponsored by Caltrans, provides the basis for exploration of the transformation suggested here.

3.3 PATH TO IMPLEMENTATION

In this section, we suggest an implementation path that would enable Caltrans to gradually experiment with and adopt a hybrid traffic data collection system. The critical changes listed in the previous section cannot realistically be adopted in a wholesale fashion, and at any rate they must be shaped incrementally from direct experience with third-party data. The path we propose here involves three steps that could be developed over a time horizon of three to five years:

1. Short-term pilot in a selected district
2. Using the PATH Connected Corridors project to spearhead information systems innovations
3. Full-scale pilot in a selected district

3.3.1 SHORT-TERM DISTRICT PILOT

The first implementation step that could be taken without delay would consist of purchasing third-party data in one or more districts on a pilot basis. As part of its hybrid data roadmap project, PATH has already taken a similar step and obtained mobile data from two vendors in Districts 4 and 8. For that purpose, PATH followed its own procurement process with Caltrans' sponsorship. This experiment should serve as a springboard for Caltrans to independently contract with one or more vendors. Doing so would enable Caltrans to work out a method for procuring the data and managing the corresponding contracts. In the following list, we link this initiative to the key implications described in section 3.2.

i. **Procurement and contract management:** The short-term pilot would give Caltrans a chance to procure third-party traffic data and manage the ensuing contracts. This would provide a template for future procurement, and the lessons learned from this first iteration would be incorporated moving forward.

ii. **Risk management:** In this initial pilot, third-party data would be collected in addition to existing detector data, and it would not feed any critical application. In this way, Caltrans would not be exposed to substantial going-concern or data quality risk. The privacy risk would have to be addressed with selected vendors.

iii. **Information systems integration:** Our suggestion for an initial pilot would be to integrate the third-party data in PeMS by fusing it with existing detector data to enhance roadway speed estimations and performance measure calculations. On the other hand, there would be no change to the selected district's ATMS and therefore no integration of third-party traffic data into TMC operations. This proposal leverages recent efforts funded by Caltrans to make PeMS mobile-data ready. It would advance the hybrid data agenda while limiting implementation risks.

iv. **Detector strategy:** At this stage, changes to Caltrans' detector strategy would not yet be necessary.

v. **Targeted benefits:** The functional benefits of the proposed short-term pilot would focus on improved traveler information and highway performance measures. Both benefits would be realized through the PeMS software suite by either fusing existing detector data with third-party data to enhance accuracy, or by providing information where none existed previously.

3.3.2 CONNECTED CORRIDORS

The second step on our proposed path to implementation would consist of utilizing PATH's Connected Corridors project as an experimentation platform for deeper systems integration and enhanced traffic management methods that leverage third-party data. The initial outcomes of the Connected Corridors project will likely unfold on a pilot basis. Deployment could take place in step with the short-term hybrid data pilot or in a different district altogether. A radical innovation brought by the Connected Corridors project will be the development of a fully operational system featuring hybrid traffic data assimilation, state estimation, and decision support in connection with predefined traffic management strategies. While not intended to replace an ATMS, this system would operate in parallel and either complement ATMS or provide a roadmap for future ATMS evolution. In the context of Connected Corridors, third-party data would be fused with existing detector data to power advanced traffic management applications, thus offering a glimpse of the full benefits of a hybrid data collection system. As in the previous subsection, we analyze the implications of this implementation step by step:

i. **Procurement and contract management:** No significant development would take place in this area. Third-party data used in the Connected Corridors project may be collected by either Caltrans or UC Berkeley.

ii. **Risk management**: Because the Connected Corridors project would run alongside nominal ATMS operations, risks would remain quite limited: The district TMC would normally rely on ATMS and existing operational practices, thus removing any dependency on third-party data. However, having Connected Corridors run in parallel would allow the PATH team and Caltrans managers to develop a clearer appreciation of the risks involved with relying on third-party data for daily traffic management.

iii. **Information systems integration:** Because Connected Corridors is building a complete data assimilation and state estimation system, it will provide a template for the kind of data fusion that would eventually make sense to fully take advantage of hybrid data in traffic management systems. As such, Connected Corridors would represent a first step toward a full redesign of information systems as described in section 3.2.3.

iv. **Detector strategy:** In this step, detector strategy would still not become a critical consideration, although the identification of detection gaps and overlaps would be an outcome of implementing Connected Corridors.

v. **Targeted benefits:** The implementation of Connected Corridors in a selected district would showcase at least two sets of benefits of a hybrid traffic data collection system: Novel or optimized traffic management strategies would be supported by comprehensive state estimation techniques, and a more open data architecture than is currently the case with ATMS would be employed, tying directly to the concept of integrated corridor management.

3.3.3 FULL-SCALE PILOT IN SELECTED DISTRICT

The third and final step we propose before Caltrans can elevate a hybrid traffic data collection strategy to a statewide practice would be a full-scale pilot in a selected district. Under this scenario, a fully operational traffic management system fed by a hybrid set of traffic data would be rolled out in the district's TMC. That traffic management system may either be an evolution of current ATMS software, a module thereof, or a brand new implementation. In any case, the critical innovation would be to let go of an information model in which there is a rigid mapping between the sources of data (i.e., traffic detectors) and the inputs used to run traffic management applications. Instead, multiple data sources would feed a state estimator which would serve those applications.

The new traffic management system could be initially rolled out in parallel to the existing ATMS to ensure a smooth transition and the ability to roll back if needed. This implementation step would bring the following changes:

i. **Procurement and contract management:** Procurement and contract management for third-party vendor data would leverage previous initiatives and possibly enhance them with lessons learned.

ii. **Risk management:** A traffic management system that uses hybrid data as a source of information would be more powerful than if it only uses loops but also more dependent on external vendors. Data quality would need to be monitored on a continuous basis, and attention should be paid to switching costs when selecting a data provider. However, the two preceding steps proposed on the implementation path should provide substantial experience to help manage these new risks.

iii. **Information systems integration:** This step would largely consist of bringing a fully integrated traffic management system into operations, thus realizing the core benefits of hybrid traffic data.

iv. **Detector strategy:** The deployment of a traffic management system that uses hybrid traffic data will provide the opportunity to start altering the selected district's detector strategy. It would enable defining critical detectors that need continued maintenance and reassess needs beyond that.

v. **Targeted benefits:** At this stage, the district selected for pilot implementation would be in a position to reap all of the potential benefits of a hybrid traffic data collection system.

3.3.4 SUMMARY

Table 19 summarizes the recommended implementation path described in this section. It lists all three implementation steps as columns and their effects against the main implications previously described in section 3.2.

Table 19: Hybrid data roadmap: implementation steps and implications

	Short-term Pilot	Connected Corridors	Full-scale Pilot
Outsourcing	Pilot procurement and contract management for third-party traffic data	Incremental improvements	Incremental improvements
Risk management	Limited to setting policies on data privacy and security	Limited by implementing Connected Corridors on an experimental basis	Third-party traffic data dependence can be initially limited by roll-back option
Information systems	Integration of mobile data into PeMS	Prototype integration of mobile data into traffic management applications	Pilot operations-grade integration of mobile data into traffic management applications
Detection strategy	No impact	No impact	Start shifting priorities based on mobile data availability
Target benefits	Enhanced traveler information and performance measures	Traffic state estimation from hybrid data; novel/optimized integrated corridor traffic management strategies	Novel/optimized traffic management strategies; save on detector maintenance costs; open architecture

4 DATA MIX: TRADE-OFFS AND OPTIMALITY

The primary rationale for shifting traffic data collection from the current detector-based paradigm to a hybrid system that incorporates third-party data is to improve information quality for a given budget allocation. Decisions should therefore be driven by trade-offs between the costs and performance of available data sources. For instance, traffic information on a stretch of freeway can be derived from any number of detectors and the addition of third-party mobile data: For a set budget, we would like to know the mix of detectors and third-party data that would result in the most reliable and accurate information. Alternatively, if we can formulate the data quality that is sought, how can it be achieved at the lowest possible cost using a combination of detectors and third-party data?

In practice, establishing trade-offs between data sources and determining an optimum is a very complicated proposition for the following reasons:

a. **Data quality is not a trivial concept:** There is much argument about how to precisely define data quality. In broad terms, “good data” is plentiful, accurate, and reliable. However, a formal definition needs to translate all desirable features of a data stream across network locations and time slices into computable metrics. Various such metrics have been defined and used in practice, but none is completely authoritative.

b. **Requirements vary by application:** Part of the reason why practitioners use a range of different metrics is because they typically do it in the context of specific applications such as estimating travel times or computing vehicle-miles traveled. Even if a single set of metrics could be established, different applications require different levels of quality. If, say, ramp metering is the most stringent application, setting data quality requirements accordingly is only appropriate for a freeway where ramp metering is effectively deployed.

c. **Measurements are imperfect and costly:** Defining a data quality metric is not the end game, because then it still needs measurements. So-called “ground truth” is another object of debates among practitioners and researchers. In practice, available measurements end up determining data quality metrics rather than the other way around. Measurement errors and costs have to be factored in as well.

d. **Context is a huge matter:** On a rural highway with little traffic, there is no need for instrumentation to obtain very accurate travel times—free flow travel times prevail all the time. On the other hand, an urban interchange may require very dense instrumentation. To know exactly how much, we would have to be monitoring local traffic variations in the first place. Hence, finding an optimal data mix is always context-specific, and there is no universal formula for it (which doesn’t mean, however, that we can’t establish rules of thumb).

e. **Data quality is governed by diminishing returns:** On a 10-mile stretch of freeways instrumented with 10 detector stations, adding another 10 stations will provide much less incremental value than the first 10 did. This means that data quality trade-offs shift with each new data point that gets added, creating a very complex relationship.

f. **Cost comparisons are difficult as well:** Even on the cost side of the equation, which one would hope is more tangible, complications arise. Detectors have high initial costs and relatively lower ongoing costs, whereas third-party data will typically be purchased as a service with fixed periodic costs. Comparisons call for estimating the lifecycle costs of detector stations, but this requires sophisticated accounting to keep track of maintenance efforts which are lumped together with the maintenance of other types of freeway equipment. Mobile data sources are still a novelty and therefore their long-term costs are hard to predict. Moreover, mobile data volumes fluctuate widely and are growing overall. Thus the cost of third-party traffic data per unit of information is even harder to anticipate.

Against this backdrop, attempts to characterize trade-offs between data sources have to be infused with modesty. They are nonetheless necessary to provide a priori justification for procuring third-party traffic data rather than continuing to deploy detectors. This section draws from existing literature and investigations conducted by PATH as part of the *Hybrid Data Roadmap* project to provide a general sense of data mix trade-offs, at least at the order-of-magnitude level. It is organized as follows:

- It first describes useful traffic data quality measures that can be used in this context.
- It then provides cursory results on the general trade-offs between inductive loop detectors and mobile data sources in terms of information quality.
- Next, it examines traffic data collection costs to the extent allowed by available information.
- It concludes with a case study conducted on highway I-880 as part of the *Hybrid Data Roadmap* project to show what an optimal data mix might look like.

4.1 TRAFFIC DATA QUALITY MEASURES

Generally, there are a number of desirable attributes of traffic data. The Federal Highway Administration (FHWA) identifies the following criteria[16]:

- **Accuracy:** the measure or degree of agreement between a data value or set of values and a source assumed to be correct.
- **Completeness (also referred to as availability):** the degree to which data values are present in the attributes (e.g., volume and speed are attributes of traffic) that require them. Completeness is typically described in terms of percentages or number of data values.

[16] Federal Highway Administration, "Data Quality White Paper," (Washington, D.C., 2008)

- **Validity:** the degree to which data values satisfy acceptance requirements of the validation criteria or fall within the respective domain of acceptable values. Data validity can be expressed as the percentage of data values that either pass or fail data validity checks.

- **Timeliness:** the degree to which data values or a set of values are provided at the time required or specified. Timeliness can be expressed in absolute or relative terms.

- **Coverage:** the degree to which data values in a sample accurately represent the whole of that which is to be measured. As with other measures, coverage can be expressed in absolute or relative units.

- **Accessibility (also referred to as usability):** the relative ease with which data can be retrieved and manipulated by data consumers to meet their needs. Accessibility can be expressed in qualitative or quantitative terms.

In addition, in the context of a hybrid data system that fuses multiple sources, it becomes necessary to qualify the purity of a set of data:

- **Purity:** degree to which the data feed consists of raw data.

In the context of determining trade-offs between data sources, our primary focus is on accuracy. In reality, fusing multiple sources of data should help with all dimensions of data quality. For instance, complementing existing detectors with mobile data would almost certainly improve coverage because it would enable traffic observations at “blind” locations between detectors. Intelligent algorithms would also enhance validity by checking one source of data against another for consistency. Yet it can be argued that all quality measures ultimately boil down to accuracy: In a real-time context, practitioners have information available, and the key question is how closely this information matches current traffic conditions. If data is incomplete or has high latency (i.e., timeliness is low), the answer to that question would be rather negative, even if individual measurements are highly accurate. The dimensions listed above provide a useful decomposition of the discrepancies that may exist between the data and what it is meant to represent, but a ‘generalized accuracy’ that compares a data stream with a source assumed to be correct essentially captures all of the dimensions.

In the remainder of section 4, our assumption is that we want to capture the accuracy of traffic information (which may be created from either one or multiple data sources) for a given facility, i.e., a stretch of highway, over an extended period of time (typically hours or days). To establish ground truth, we need continuous measurements in both space and time. The practical solution identified by PATH was to deploy Bluetooth readers along selected highways, in relative proximity to one another (about a mile apart or so). This effectively breaks down a section of highway into short segments across which travel times are collected. Note that while the Bluetooth readers are spread across the highway in similar fashion to inductive loop detectors, they record traversal times rather than local velocity: This means that the traffic conditions that prevail between readers are captured by the Bluetooth readers (technically, the readers integrate the velocity field on each segment). Using those Bluetooth readers as ground truth means that we can divide up space and time into small units and compare independently acquired traffic information to that ground truth on a per-unit basis. Figure 6-8 provides a schematic representation of this mechanism.

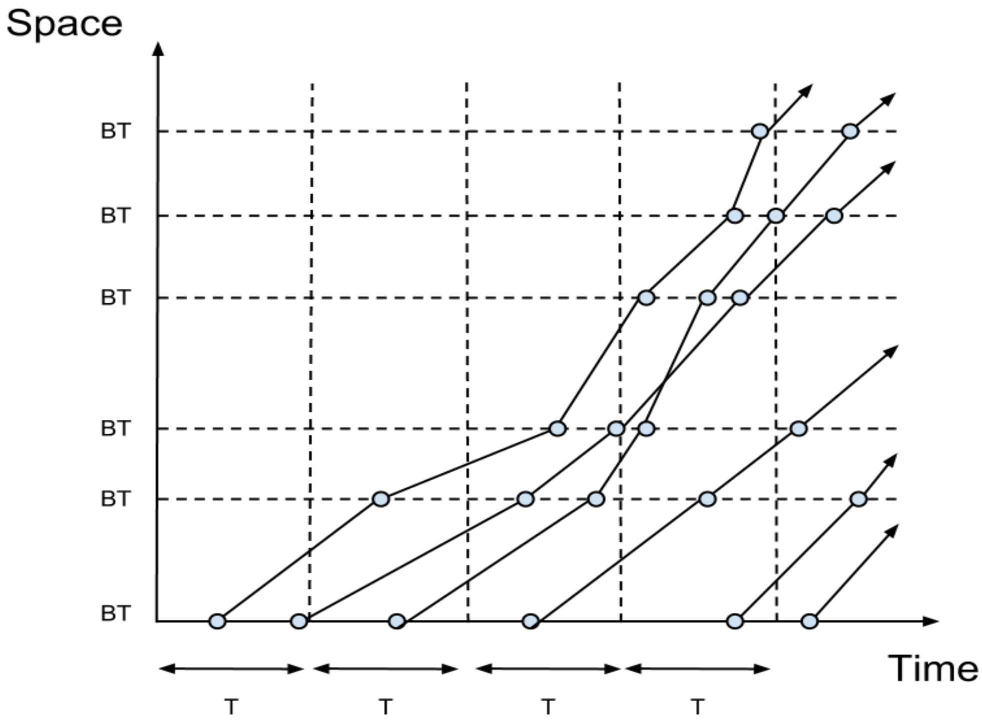


Figure 6-8 - Schematic representation of ground truth capture by Bluetooth readers (labeled on the y-axis). Each solid line is a vehicle trajectory and circles represent detection events by the readers. The dashed lines represent the division of time and space into individual units, which correspond to the resulting rectangles.

Bluetooth readers deployed in this way enable a somewhat granular capture of speed variations, both in time and in space, along a given facility. This setup is therefore well-suited to measuring the accuracy of traffic speed information reported by other sources. One limitation, however, is that Bluetooth readers, like other mobile data sources, say little or nothing about traffic volumes. Traffic volume, or traffic occupancy, is an important variable in traffic management procedures. It is directly captured by inductive loops and other types of fixed detectors. However, the extent to which traffic volume measurement can be enhanced by mobile data sources remain a topic of research, with very few answers to date. One way this could be assessed would be by using loop detectors as ground truth and removing the corresponding data from the information that is fused with mobile data. In this manner, one could test, on a facility where fixed detectors are deployed every mile, how information would fare with only one detector every five miles and the addition of mobile data to offset the loss of fixed sources. From the perspective of a long-term hybrid data roadmap in which mobile data becomes an integral part of the data mix used for traffic management, such a determination is necessary and should be the object of future investigations. Short of that analysis, the present document focuses on the accuracy of speed and travel time estimates.

Having discretized time and space into elementary units, the definition of comprehensive accuracy metrics becomes relatively straightforward. In particular, the mean absolute percentage error (MAPE) is used to globally assess the discrepancies between a traffic information source and the Bluetooth ground truth:

$$MAPE = \frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}\frac{\left|T_{hwy}(i,j) - T_{bt}(i,j)\right|}{T_{bt}(i,j)}$$

where *i* and *j* index time and space, respectively. This metric averages the absolute percentage error between the estimated travel time from a traffic information source, T_{hwy}, and measured travel times T_{bt}, for each combination of segments (defined by the position of Bluetooth readers) and time periods (arbitrarily defined but typically 15 minutes. These correspond to the intervals noted T on Figure 6-8.)

One problem with this metric, however, is that it averages all time-space units with the same weight. In practice, there is little traffic congestion for most of the day on most facilities. During those times, there is little chance that traffic estimates will be wrong. If we were to calculate MAPE over a day, important discrepancies between traffic information and ground truth that take place during a limited time at rush hours may then be lost in the process of averaging them with the more extensive periods of time during which traffic is free-flowing. One way to remedy this problem is to limit analysis to congested periods. This is what the PATH team effectively did. Figure 6-9 shows a record of ground truth data captured on I-880 between Fremont and Castro Valley over a period of a week. The dark boxes represent space-time regions during which congestion took place. Consequently, MAPE calculations that aim to estimate the accuracy of an independent traffic information source are only carried out within those boxes.

Another, related metric to help gain an intuitive feel for the results is the per-mile average time error (PMATE):

$$PMATE = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M}\frac{L(j)}{\sum_{j=1}^{M} L(j)}\frac{\left|T_{hwy}(i,j) - T_{bt}(i,j)\right|}{L(j)}$$

$$= \frac{1}{N\sum_{j=1}^{M} L(j)}\sum_{i=1}^{N}\sum_{j=1}^{M}\left|T_{hwy}(i,j) - T_{bt}(i,j)\right|$$

where $L(j)$ is the length of the route indexed by *j*. PMATE normalizes travel time estimation errors in terms of seconds per mile, which may give a more immediate appreciation of the discrepancy between actual and estimated travel times, especially as it relates to the driver's experience in the case of a traveler information application.

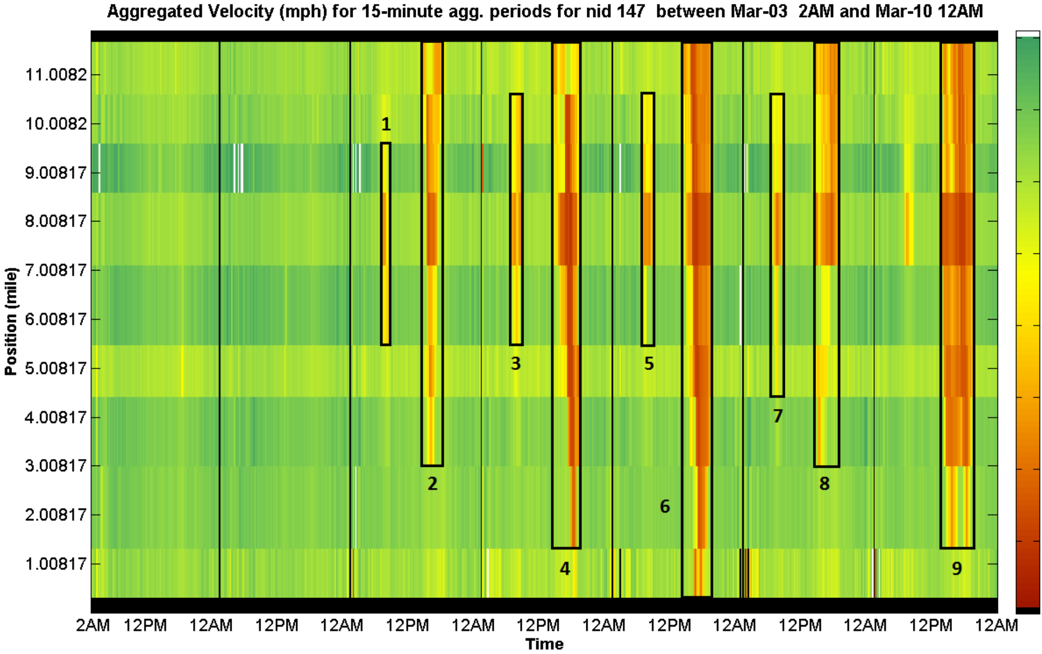


Figure 6-9 - Ground truth traffic conditions as derived from Bluetooth travel times and presented as average velocities. The data is for one week from Saturday, March 3 to Friday, March 9, on a 12-mile stretch of I-880. Black boxes represent time-space regions that are included in the calculation of global accuracy metrics such as MAPE.

4.2 TRADE-OFFS BETWEEN DATA ELEMENTS

This section explores the results of investigations carried out by PATH as part of the hybrid data roadmap project to characterize the trade-offs between inductive loop detectors and commercially available mobile data. These results focus on a section of I-880 between Fremont and Castro Valley, where data was collected for a period of two weeks. The relatively limited scope of the study means that its results cannot be generalized in an authoritative manner, but they nonetheless provide robust insights obtained from a rigorous methodology. The details of that methodology are described in the hybrid data roadmap project report. The gist of it hinges on the fact that the freeway section that was studied is very densely instrumented with inductive loop detectors. Because of this, it is possible to compare the quality of the traffic information obtained by such dense coverage with that derived from only a subset of detectors, or from a combination of detectors and mobile data. In any case, ground truth is provided by Bluetooth detectors as described in section 4.1.

Trading-off between data elements means that we want to compare the quality of information obtained under a range of technology deployments. In effect, we are asking whether it is more desirable to maintain a large number of inductive loop detectors and consider other sources as secondary, or to susbscribe to a mobile data service at the expense of loops. Ultimately, that comparison needs to be informed by costs assumptions. What we are seeking is either the cheapest technical solution that can meet a set level of information quality, or the best information quality for a given budget.

In order to do this, we first need to characterize the use of each information source in relevant units. For inductive loops, we choose the number of loops per mile as a logical choice: It is a normalized metric of the quantity of detectors deployed on a facility. For mobile data, the choice is less obvious. In effect, the number of data points collected from a vehicle fleet varies constantly in both time and space. Data collectors may also sample information from vehicles at different rates; therefore, more data points does not necessarily mean better data (for instance, there is little incremental value in sampling vehicle speed every second as opposed to, say, every 10 seconds, but it generates ten times more data). Further, traffic information services are priced in an ad-hoc manner and may only loosely correlate to the volume and quality of the data provided. Finally, as with loop detectors, we want to express the use of mobile data in a metric that is normalized, meaning that it does not depend on the length of the facility that we are examining or the duration of the study.

To resolve these conceptual challenges, we propose the use of a metric that reflects the typical flow of probe vehicles across a given roadway section during times of substantial traffic activity. Formally, this metric is defined as the 85th percentile hourly flow of probe vehicles reported by a mobile data service over a long enough period (a week or more is ideal). This metric has the following features:

- It is normalized with respect to both duration and facility length. For a facility that is not too long, the flow can simply be measured at or near the median point and still be representative of the entire facility. For a very long facility, it would usually make sense to break down the problem by defining segments.
- It does not depend on the mobile data collector’s sampling rate. What is counted is the number of unique vehicles. This is more meaningful than the number of data points for two reasons: a) as previously mentioned, the sampling rate has diminishing rates of return on information quality, and b) the real challenge of creating information from probe vehicles is to have access to a large enough fleet, whereas the sampling rate is something that can be adjusted. Therefore, the flow of unique vehicles is a more intrinsic property of a mobile traffic data service than the sampling rate.
- It is not too sensitive to extreme variations in data availability. By taking a high percentile, we essentially discard the time periods during which little or no data is collected, especially at night. Conversely, this metric is not affected by a few “lucky strikes” (selected hours during which a particularly high volume of vehicle data is collected, but that are not statistically representative).

In the remainder of this section, those two metrics—the number of loops per mile and the 85th percentile hourly probe flow—are used to quantify the level of use of each information source. They can be used to represent facility configurations; for instance, a possible configuration would combine one loop per mile and mobile data amounting to an 85th percentile flow of 10 vehicles/hour. Variations in both metrics were simulated by removing some of the available data (either detectors or a set proportion of mobile data observations). Further, we will assume that the cost of a mobile data service depends on the 85th percentile flow. There is absolutely no evidence of that, but it is nonetheless a reasonable assumption: the larger a fleet of probe vehicles, the more valuable the corresponding information service. In practice, mobile data services are provided either in whole or not at all, so there isn't really an opportunity to fine-tune a configuration with respect to the volume of mobile data. However, there could be situations in which Caltrans can pick between two or more providers, each featuring different volumes, with the additional option of combining providers to obtain a higher volume of hourly probe data.

Figure 6-10 shows the values of the MAPE performance metric, which is the discrepancy between estimated travel times and ground truth travel times, expressed in percent (where 0.25 = 25%), under several scenarios. The quantity on the x-axis captures the number of unique vehicles reporting information as part of the mobile data streams across a given section. Specifically, it is the 85th percentile of that hourly flow across all collected hours, meant to represent typical good conditions. Each curve corresponds to a given subset of inductive loops (as reported by the PeMS system).

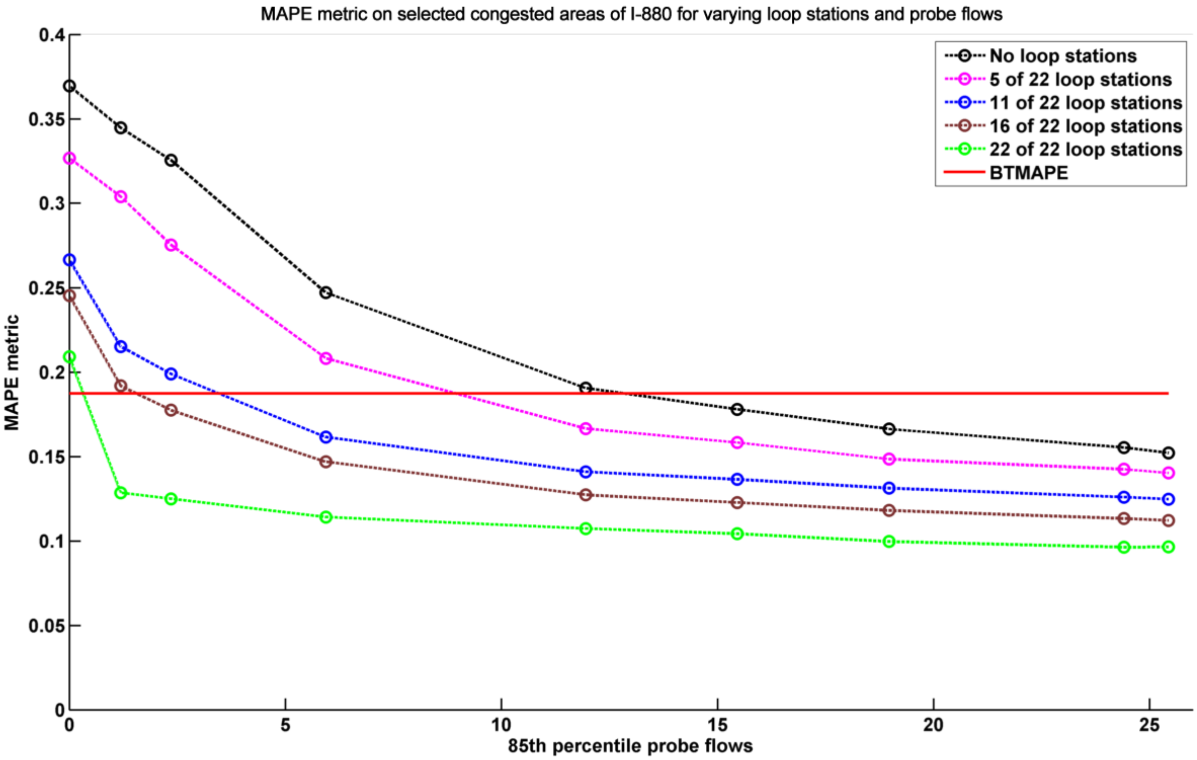


Figure 6-10 - Global travel time MAPE plotted against the 85th percentile hourly probe flows. Each dashed line corresponds to the performance achievable when fusing mobile data with a set proportion of existing loop detectors.

When all detectors are in use (green curve) and no mobile data is incorporated, the difference from Bluetooth measurements is 0.21 (21%). This is high, but it should be noted that the metric is calculated only over congested segments and time periods, when travel times are inherently volatile. The red line around 0.18 indicates a so-called “noise floor,” which corresponds to the dispersion in the ground truth travel times (with outliers removed). Adding mobile data from even a small number of reporting vehicles (85th percentile flow < 5) lowers the metric to 0.13, well below the noise floor.

The other curves show results with no detectors, one out of four detectors, one out of two detectors, and three out of four detectors, respectively. When no detectors are used in travel time estimations, an 85th percentile hourly flow of 10 yields results comparable to those obtained with all detectors and no mobile data.

A more direct representation of the trade-off between data sources is provided by Figure 6-11, which uses the concept of iso-lines with respect to the MAPE metric. The MAPE metric describes estimation errors in terms of relative percentages rather than absolute differences. For instance, the iso-line labeled 0.18 corresponds to a global error of 18% (18% discrepancy from Bluetooth measurements). This level of accuracy can be obtained with 1.5 loops per mile (y-axis intersection) or alternatively with 15 reporting vehicles per hour (85th percentile basis, x-axis intersection). Intermediate combinations can be readily assessed: Half that number of loops (0.8) and 6 vehicles per hour would also yield 18% error.

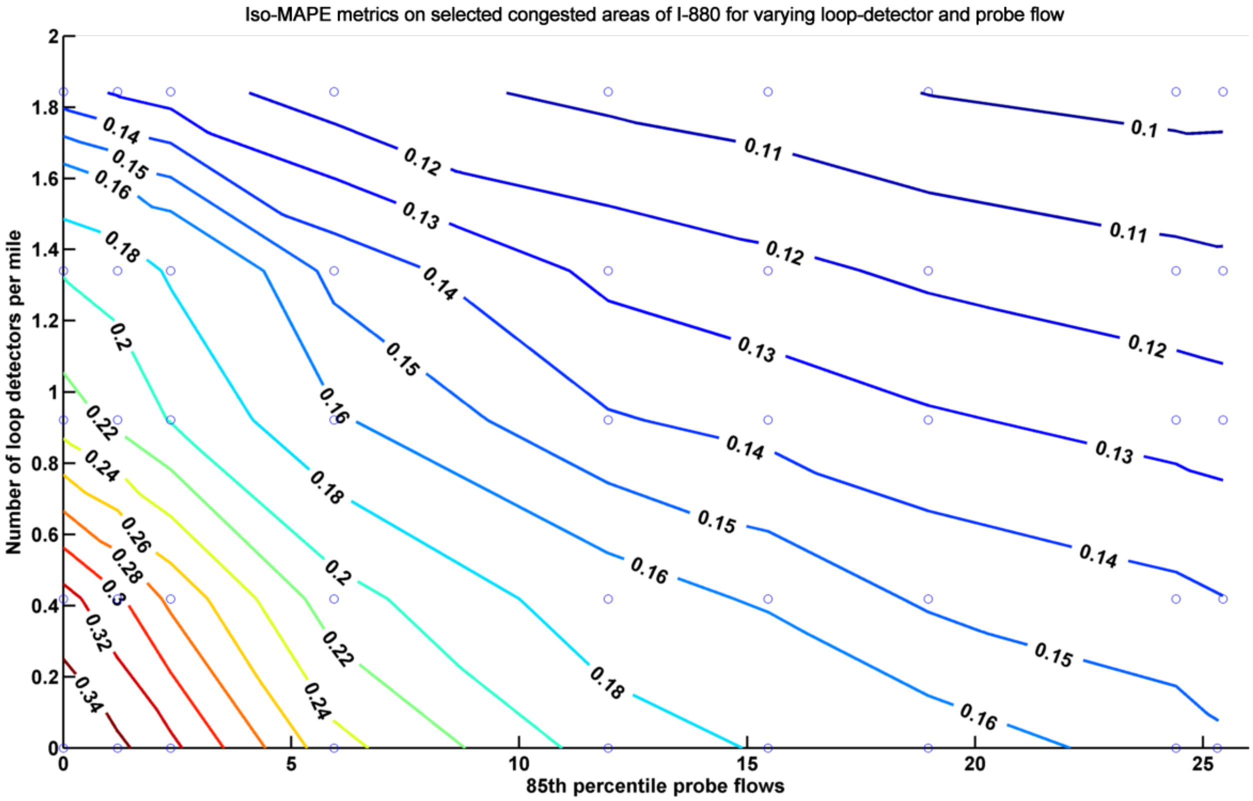


Figure 6-11 - Contours of constant travel time estimation performance as a function of both the number of loop detectors and the 85th percentile probe flow (vehicles/hour)

4.3 TRAFFIC DATA COLLECTION COSTS

In 2007, the California Center for Innovative Transportation at UC Berkeley estimated the lifecycle cost of installing and maintaining a typical freeway vehicle detection station with inductive loops to monitor four lanes of traffic flowing in one direction[17]. The estimated figure broke down into $26,100 for installation and $23,400 for maintenance over a 15-year span, factoring in the likelihood of replacing loops as they break down. Along with those dollar figures, the deployment and operation of each vehicle detection station resulted in an estimated 56 lane-hours of traffic closures. These figures mean that the lifecycle cost of a detection station lies in a range of $2,500 to $3,500 annually. Other types of sensors may be more or less expensive. For instance, the same study found that the 15-year lifecycle cost of the same detection station using wireless magnetometers would approximate $22,500, which translates to an annual cost of around $1,500 if the detectors are maintained in perpetuity. Of these costs, approximately 20%, or $300 per year, correspond to ongoing operations and maintenance.

The costs of traffic data services derived from mobile sources are more difficult to establish because the market is still recent and constantly evolving. Further, meaningful cost comparisons must take data quality into account, which in the case of mobile sources depends primarily on the density of reporting vehicles. Evidence assembled by PATH as part of the hybrid data roadmap project points to costs of a few hundreds of dollars per mile per year for a feed of good quality. Given its size, Caltrans would be able to negotiate favorable pricing terms if it decided to purchase traffic data services on a broad scale.

4.4 OPTIMAL MIX

In order to provide insights into the optimal mix between fixed detectors and mobile sources, we compared performance and costs under a range of possible configurations on I-880. Performance trade-offs are described in section 4.2.

4.4.1 COST ASSUMPTIONS

While different cost/benefit models might use different assumptions, we used the following in this case study:

- Installing or maintaining detector stations in perpetuity on a major freeway segment costs about $1,500 per year on a lifecycle basis.
- Operating that same detector station once it has been installed, with no regard for eventual replacement, corresponds to a yearly cash outlay of approximately $300 per year.

[17] Ellen Robinson and J.D. Margulici, “Monitoring of High-Occupancy Vehicle Lanes in Caltrans Districts 3 & 4,” California Center for Innovative Transportation Report prepared for Caltrans, 2008

- Traffic data services have a fixed cost of at least $200 per mile per year and some type of variable cost which we calculated using three cost assumptions:
    - **Linear**—A simple model in which additional data is valued proportionately. For example, if one vendor provides data from 10 probe vehicles per hour with an additional variable cost of $200 per mile per year, another vendor offering twice as much data (20 probes/hour) could charge an additional $400 per mile per year.
    - **Quadratic**—A model in which the largest data providers can command disproportionately larger prices for their services. This approach might capture a market situation dominated by a few players, where only the largest providers can can supply consistently large volumes of probe data, giving them exceptional market power.
    - **Exponential**—Reflects the fact that increasing amounts of probe data provide diminishing marginal gains in travel time estimation performance.

4.4.2 OVERALL FINDINGS

We found that costs are primarily driven by the number of loops per mile and that even with sparse detectors every few miles, the annual cost of maintaining those detectors remains significantly higher than the cost of purchasing probe data, all for comparable information quality outcomes. Overall, for moderate travel time estimation accuracy, the most cost-effective solution is to purchase probe data. As greater accuracy is required, it might be necessary to install loop detectors.

This was especially clear for the linear and quadratic cases where, in seeking greater accuracy, there will typically be a price point at which a hybrid solution (with both probe and loop data) becomes desirable. For the exponential case, it is typically best to purchase as much probe data as possible, and only to add loop data when the performance requirements demand it.

It is important to remember that these results only considered travel time estimation applications. The presence of ramp meters would otherwise justify maintaining vehicle detection stations, but that was not within the scope of this study.

4.4.3 READING THE GRAPHS

The following graphs (Figure 6-12 through Figure 6-15) illustrate the relationship of performance and cost by showing two sets of lines and marking the optimal mix of loops and probes on each:

- **Isometric performance contours** (shown in blue), similar to the performance contours in Figure 6-11. As with Figure 6-11, the isometrics show the combinations of loop detectors and probe data (85th percentile flow) needed to achieve a given performance level. The numbers along the contour lines specify the percent discrepancy from Bluetooth measurements of travel time (where 0.14 = 14%). Lower numbers indicate better performance (accuracy).

- **Contour lines of constant cost** (black lines labeled with dollar amounts), superimposed on the performance contours. The dollar amounts vary depending on the cost assumptions of each graph, but in all the graphs the costs increase with the number of loops per mile.

- **Lowest cost for a fixed level of performance.** Each blue performance line includes a *green circle*, marking the point of lowest cost for the performance that line represents, such as 0.14. The combination of loop data (y-axis) and probe data (x-axis) located at the green circle is the most cost-effective way of achieving that level of performance.

- **Best performance for a fixed cost.** Each black cost line includes a *green square*, indicating where, for that dollar amount (such as $500 per mile per year), the best performance can be achieved. The combination of loop data (y-axis) and probe data (x-axis) located at the green square will yield the best performance for that fixed cost.

4.4.4 PERPETUAL BASIS

We compared the costs and benefits of maintaining freeway detectors in perpetuity with those of purchasing third-party data services, according to the following quadratic formula:

$$\text{Total Cost} = 1500{*}n + (200 + \max(0, f - 15)^2)$$

where **n** is the number of loops and **f** is the probe flow. Figure 6-12 shows the results:

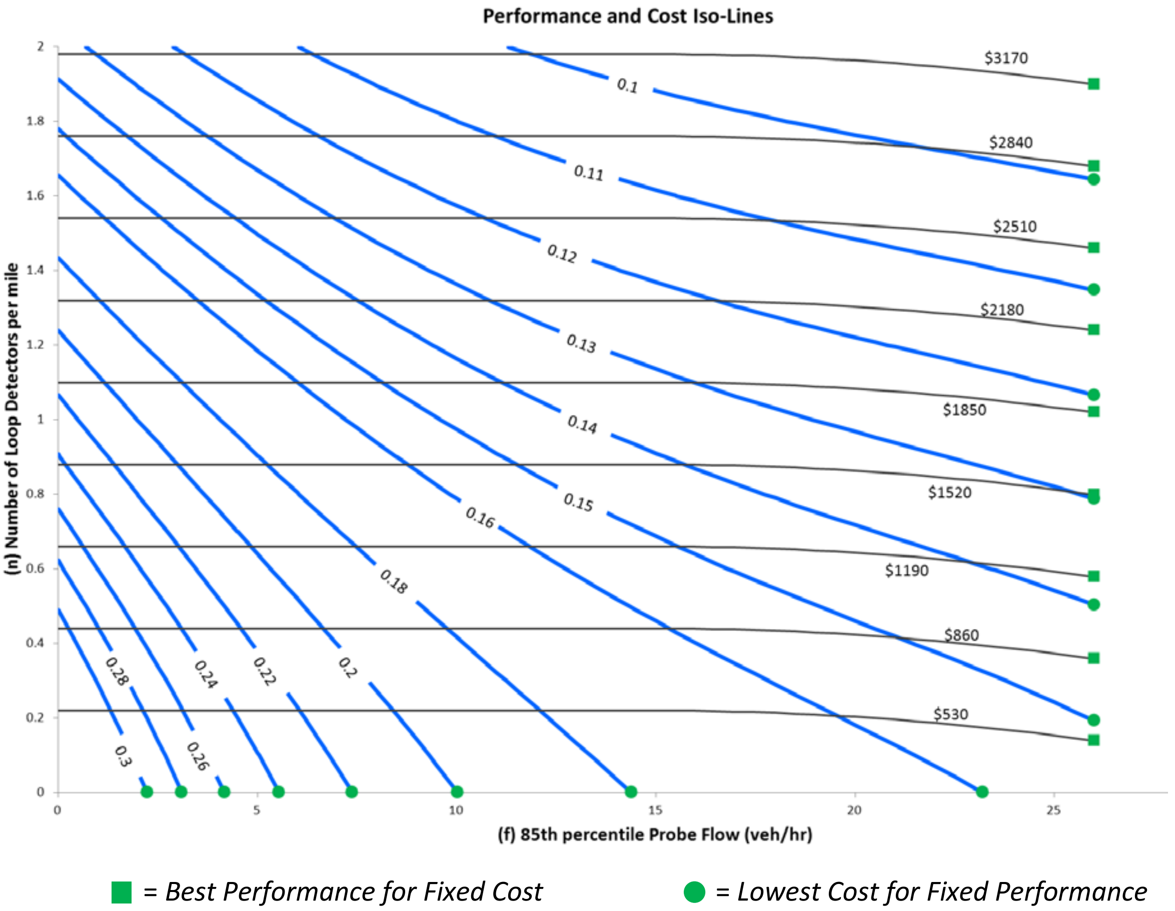


Figure 6-12: Comparing performance and costs for different mixes of information sources, assuming vehicle detection stations are maintained in perpetuity. Quadratic cost assumption for probe data.

The green circles show the combination of loops and probes that provides the lowest cost for a fixed performance level. For example, a performance metric of 0.18 (18% discrepancy from Bluetooth measurements) can be obtained with a probe flow of 15 vehicles per hour without loop data, for an estimated cost of $200 per mile per year. (In this example, the cost lines increase in increments of $330.)

The green squares show the combination of loops and probes that provides the best performance for a fixed cost. For example, for a fixed cost of $530 per mile per year, a performance metric slightly higher than 0.15 can be achieved with a combination of 0.2 loop detectors per mile and a probe flow of 26 vehicles per hour.

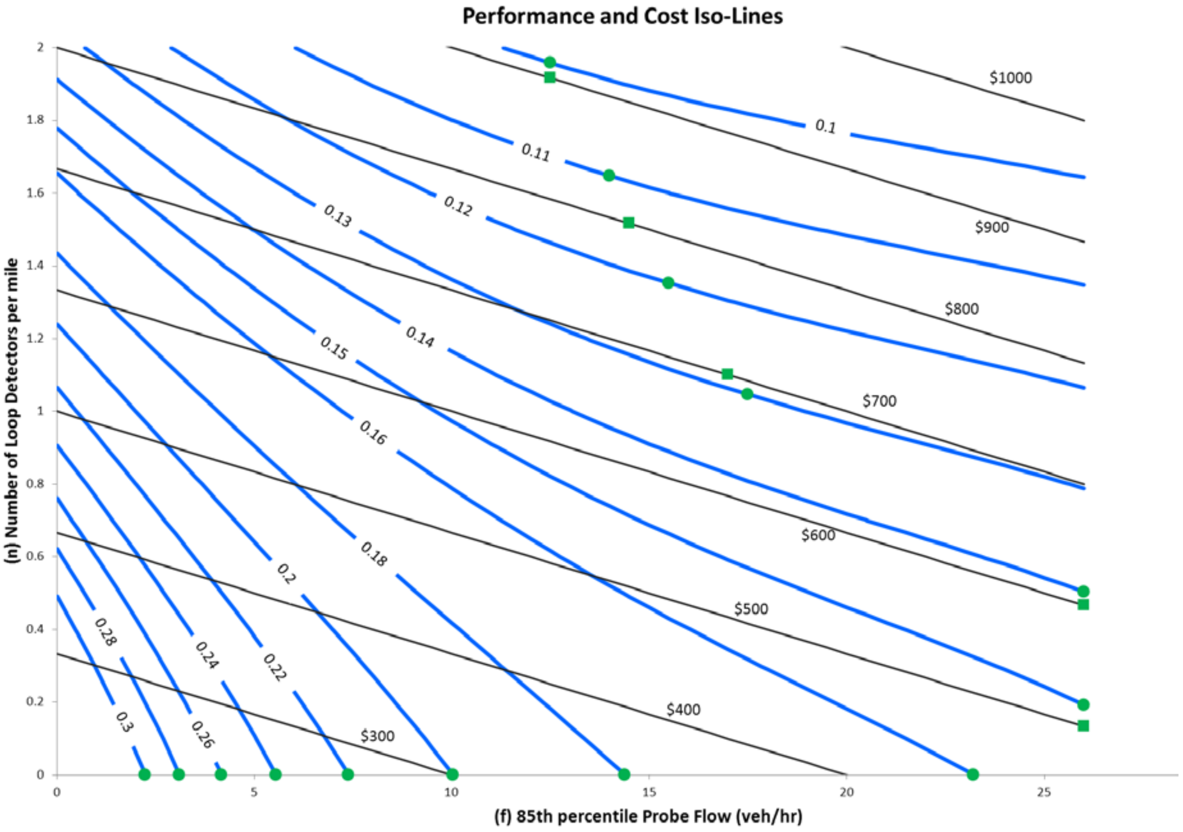
4.4.5 OPERATING BASIS

In considering only the operating costs of the vehicle detection stations, we performed the calculations using the three cost assumptions for purchasing probe data.

Linear. A linear formula for the costs of probe data assumes that if one vendor provides data from 10 probe vehicles per hour with an additional variable cost of $200 per mile per year, another vendor offering twice as much data (20 probes/hour) could charge an additional $400 per mile per year, according to the following equation:

$$\text{Total Cost} = 300*n + (200 + 10*f)$$

where **n** is the number of loops and **f** is the probe flow. Figure 6-13 shows the results:



🟩 = *Best Performance for Fixed Cost* ● = *Lowest Cost for Fixed Performance*

Figure 6-13: Comparing performance and costs for different mixes of information sources, assuming existing vehicle detection stations are not maintained beyond their useful life. Linear cost assumption for probe data.

In this figure, the green circles (lowest cost for a fixed performance level) show that a performance level of 0.16 (16% discrepancy from Bluetooth measurements) can be obtained with a probe flow of 23 vehicles per hour without loop data, for an estimated cost of roughly $450 per mile per year. A performance level of 0.12, however, would require a combination of 1.4 loop detectors per mile and a probe flow of 15 vehicles per hour, at a cost of more than $750 per mile per year.

The green squares (best performance for a fixed cost) show that for $600 per mile per year, for example, a performance level of 0.14 could be achieved with a mixture of 0.5 loops per mile and 26 probe vehicles per hour.

Quadratic. The quadratic formula for operating costs assumes that in a data market dominated by a few players, the largest providers can charge disproportionately more for their data services. The total cost is quantified like this:

$$\text{Total Cost} = 300*n + (200 + \max(0, f - 15)^2)$$

where **n** is the number of loops and **f** is the probe flow. Figure 6-14 shows the results:

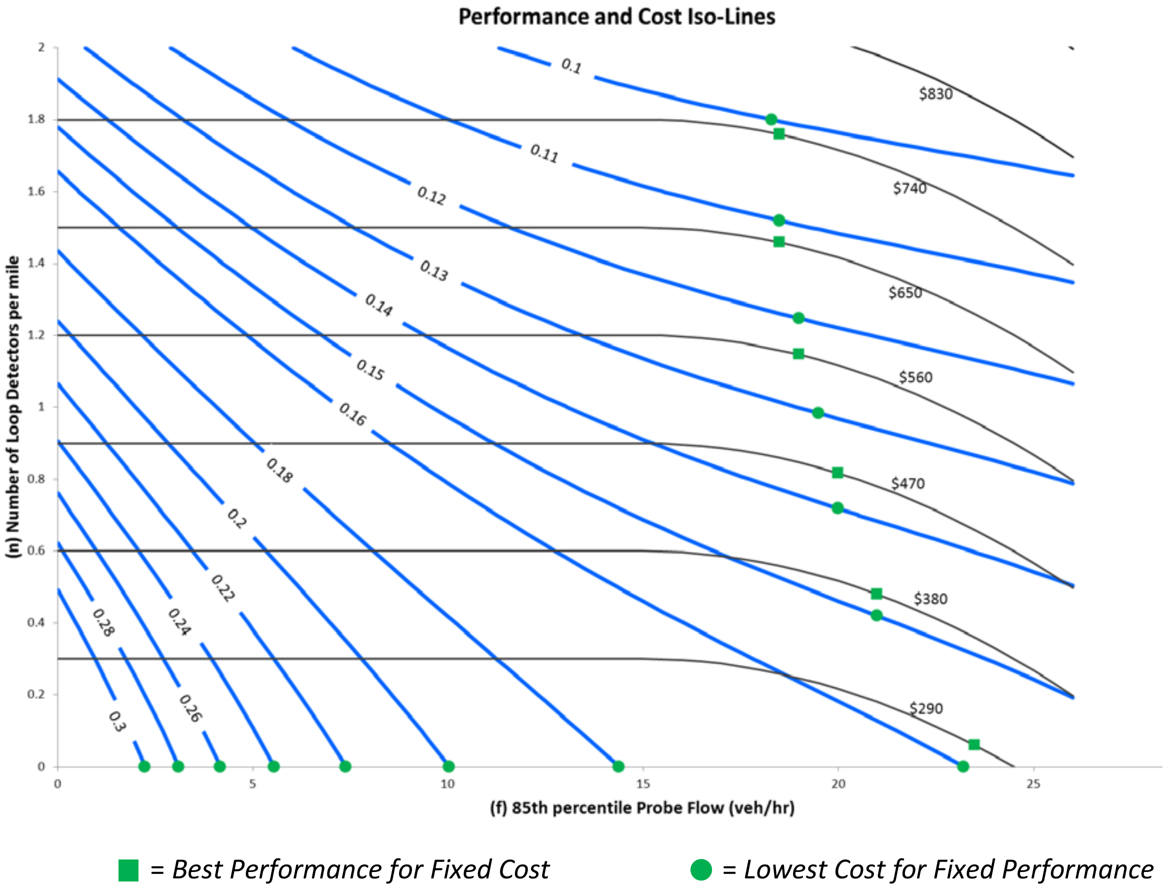


Figure 6-14: Comparing performance and costs for different mixes of information sources, assuming existing vehicle detection stations are not maintained beyond their useful life. Quadratic cost assumption for probe data.

With the green circles representing the lowest cost for a fixed performance level, we can see in this example that a performance metric of 0.16 (16% discrepancy from Bluetooth measurements) can be obtained with a probe flow of 23 vehicles per hour without loop data, for an estimated cost of under

$290 per mile per year. (In this example, the cost lines increase in increments of $90.) For a higher performance level, it would be necessary to add loop detectors. The lowest cost for the 0.12 metric, for example, comes from roughly 1.2 loop detectors per mile plus a probe flow of 19 vehicles per hour, at a cost of approximately $600 per mile per year.

The green squares, representing the best performance for a fixed cost, show, for instance, that for $380 per mile per year, a performance metric slightly lower than 0.15 (15% discrepancy from Bluetooth measurements) can be achieved with a combination of 0.3 loop detectors per mile and a probe flow of 21 vehicles per hour.

Exponential. If we assume an exponential cost for probe data, the total cost per mile per year is calculated as:

$$\text{Total Cost} = 300*n + (200 + 300*(1 - \exp(-f / 8)))$$

where **n** is the number of loops and **f** is the probe flow. Figure 6-15 shows the results:

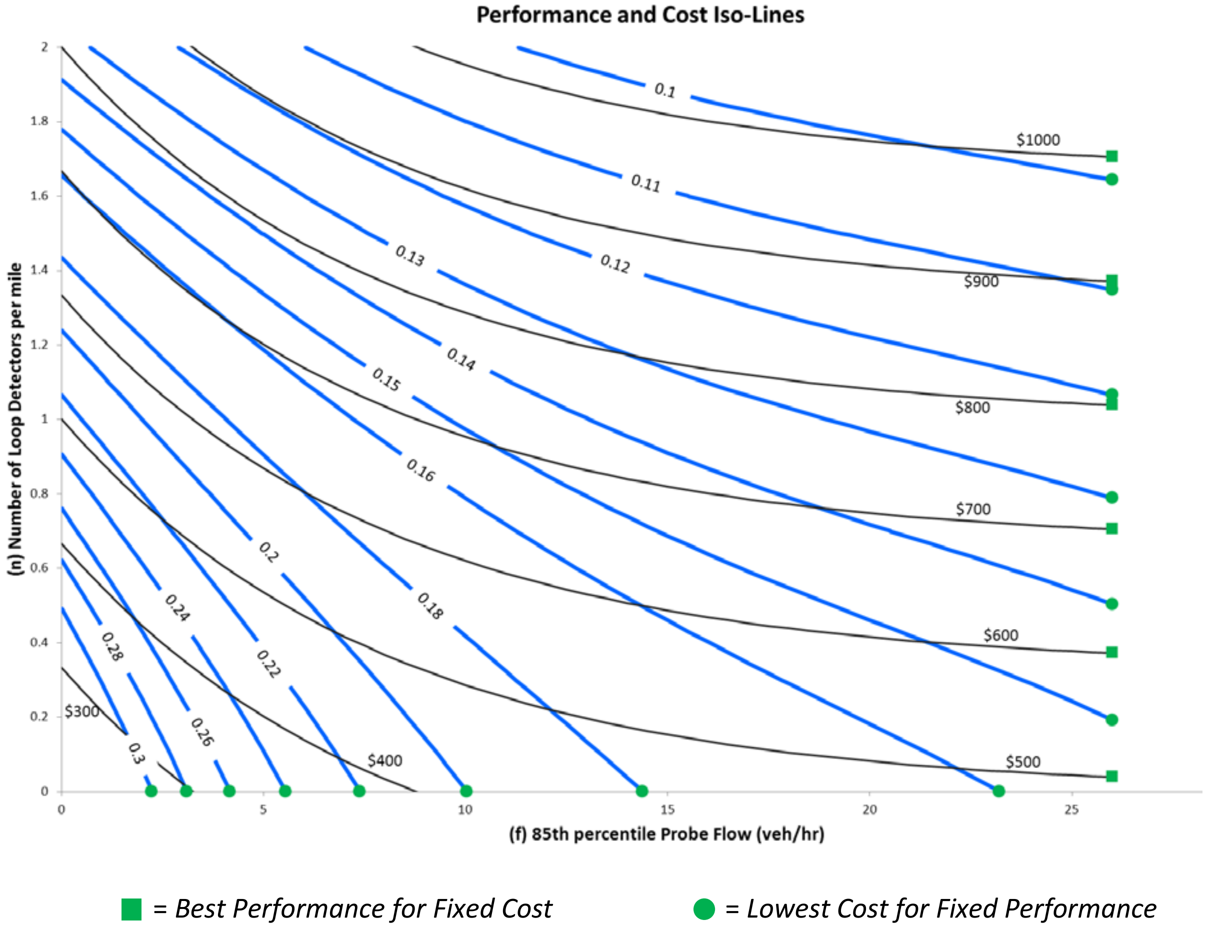


Figure 6-15: Comparing performance and costs for different mixes of information sources, assuming existing vehicle detection stations are not maintained beyond their useful life. Exponential cost assumption for probe data.

Figure 6-15 shows, for example, that the green circle (lowest cost for a fixed performance level) for a performance level of 0.16 (16% discrepancy from Bluetooth measurements) can be obtained with a probe flow of 23 vehicles per hour without loop data, for an estimated cost of less than $500 per mile per year. A performance level of 0.12, however, would require a combination of 1.1 loop detectors per mile and a probe flow of 26 vehicles per hour, at a cost of $800 per mile per year.

The green squares (best performance for a fixed cost) show that for $600 per mile per year, for example, a performance level between 0.14 and 0.15 could be achieved with a mixture of 0.4 loops per mile and 26 probe vehicles per hour.

5 PROCUREMENT OF THIRD-PARTY TRAFFIC DATA

This section provides guidance on the procurement of third-party traffic data. As pointed out in section 3.2, outsourcing data collection represents a leap from Caltrans' current business practices. On the roadmap to a hybrid data system, effective procurement constitutes a major item. As part of the hybrid data roadmap project, PATH orchestrated the procurement of private traffic information extracted from mobile sources. This procurement served a dual purpose. First, it provided data that was required to study the trade-offs between fixed detectors and mobile sources. Second, it represented a pilot effort to determine the most effective ways to structure similar procurements that Caltrans may need to conduct in the future. Some of the lessons learned are incorporated in this section.

5.1 MARKET PLAYERS

The traffic information business ecosystem is not huge (representing a couple of billions of dollars annually at most), but it is relatively complex. It is common to break down the process of providing traffic information into three stages: data production (i.e., collecting primary data from the transportation network), data processing (i.e., employing algorithms and professional expertise to interpret data and turn it into actionable information), and data dissemination (i.e., delivering information to its end users, be it through radio broadcasts, connected electronic devices, and a variety of other media). Different industry players may be involved in one or several of these stages. They may compete in one stage and collaborate in another. Or it may be that one provider sells data to a rival provider in selected geographical markets.

The list below only includes those market players that produce and commercialize traffic data for government agencies. Some of them deploy and operate fixed traffic detectors that they either manufacture or purchase from third parties. Others collect traffic data from mobile applications used by consumers. Finally, the largest volumes of data are collected through third-party agreements that some of these companies have established with vehicle fleets, fleet management software publishers, and cellular network operators.

- **Airsage:** Founded in 2000, Airsage was one of the first companies to credibly promote data collected from cellular networks. The technology taps into the flow of information managed by cellular operators to estimate the progression of swarms of vehicles through the road network (see section 2.3.3), which doesn't require drivers to carry GPS-equipped smartphones.
- **Cellint:** Like Airsage, Cellint provides data collected from cellular network analysis. The company operates internationally and has experience with a few state DOTs in the US.
- **INRIX:** Since its founders left Microsoft in 2005, INRIX has gradually risen to become the world's leading traffic information provider. In the United States, INRIX aggregates data from commercial fleets that it fuses with publicly available sources through its proprietary algorithms. However, the company claims that it cannot provide unprocessed mobile data because of the

nature of its agreements with fleet owners.

- **NAVTEQ**: NAVTEQ is the world’s leader in digital mapping and was acquired by Nokia in 2007, within which it continues to operate as a subsidiary. The previous year NAVTEQ had acquired Traffic.com, which operates traffic detectors in 50 US metropolitan areas. NAVTEQ also collects fleet data as well as smartphone data through an agreement with a cellular operator.

- **Sensys Networks:** Located in Berkeley, California, Sensys Networks manufactures innovative traffic detection technology that mimics inductive loop detectors and can also perform segment-based travel time data collection.

- **SpeedInfo:** SpeedInfo is another manufacturer of traffic detection devices based in the San Francisco Bay Area. The company’s preferred mode of operation is to deploy its Doppler radars on a DOT’s right of way and sell the data as a service on a subscription basis.

- **Telenav:** Telenav publishes a connected navigation and traffic information service for smartphones and other connected devices. The company collects traffic data from end users, and even though it does not appear to actively market data services to government agencies, it responded to the recent PATH Request for Proposals.

- **TomTom:** TomTom is the leading provider of in-vehicle aftermarket GPS units. It also acquired digital map maker TeleAtlas in 2007. Although mostly focused on consumer products, TomTom has started marketing traffic information services to government agencies in the past couple of years, based on data collected from its connected navigation units.

- **Traffax:** A spin-off from the University of Maryland, Traffax manufactures and deploys Bluetooth detectors, either on a permanent or temporary basis. It then provides customers with segment-based travel time data.

- **TrafficCast:** TrafficCast started as an aggregator and processor of traffic information and has now diversified to offer BlueTOAD, another Bluetooth-based solution to collect trip times.

- **Travelers Network:** Travelers Networks is a nascent service recently launched by Canadian weather information Pelmorex. The company acquired Triangle Software/Beat The Traffic, a provider of traffic information services to media outfits that collects its own data through a smartphone and tablet application.

Note that when PATH published an RFP as part of the hybrid data roadmap project, four companies responded. Those responses are included in the Appendices to the project’s final reports.

5.2 SPECIFICATIONS

In this section, we highlight the primary characteristics of a traffic data service that need to be considered as part of the procurement process. These include:

- Functional specifications, i.e., the nature of the data that will be provided
- Performance specifications, which refers to data quality
- Data management, which includes provisions aimed at protecting both Caltrans and its data providers from legal or public perception challenges
- Licensing terms, which describe the allowed uses of the data
- Pricing structure

In considering these specifications, Caltrans does not need to spell out exact requirements in an RFP. Because the market for traffic data services is not very mature, providers can have vastly different approaches, each with their own merits. What is important is to collect accurate information from proposers in each area and to be able to evaluate offers across all dimensions.

5.2.1 FUNCTIONAL SPECIFICATIONS

Functional specifications must describe the nature of the data that a provider can offer. Following is a list of dimensions that need to be examined:

Data sources

The first dimension is the origin of the data that is being provided. Section 2.3 describes the traffic data collection methods available to date and covers this aspect of the data specifications. Details are required to further understand the nature of the data. For instance, if data is collected from detectors, what is the technology being used? If the data comes from cellular network analysis, what existing agreements with cellular networks is the provider relying on? For fleet data, a good understanding of the type and the size of the fleets that are monitored is warranted.

Data processing

Data provision necessarily undergoes some level of processing. Understanding up front how data gets treated and delivered is another important requirement of the procurement process.

The most critical aspect of data processing is the aggregation of elementary data records. Aggregation is usually performed spatially and/or temporally. For instance, Caltrans aggregates individual data records collected by inductive loop detectors into 30-second samples. Likewise, a provider of mobile data may aggregate the information collected from individual vehicles in different ways: Data from the same vehicle may be converted into travel time observations by bundling records collected over defined roadway segments or at fixed sampling intervals. Most providers on the market today will go one extra step by aggregating the data collected from multiple vehicles over given segments (TMC location codes are relatively ubiquitous in this respect) at set time intervals (usually between 30 seconds and 5 minutes). This practice works well for the provision of traveler information applications, and it has the added benefit of offering a higher level of end-user privacy protection. However, once data has been aggregated, disaggregation becomes impossible, or very difficult at best.

Disaggregated data offers a lot more flexibility, though this flexibility comes at the cost of having to perform additional processing in-house. For the hybrid data roadmap project, PATH elected to require disaggregated mobile data, which restricted the pool of potential providers. Yet this was necessary to perform sophisticated data fusion algorithms and to effectively study the trade-offs between mobile data sources and fixed detectors. In order to realize the roadmap set forth in section 3.3, similar data fusion would need to be performed between fixed detectors and third-party data so that a complete flow model including volume and occupancy estimates can be derived. It is quite possible that one or several private traffic data providers will eventually fuse data to produce such a flow model, but that is not the case today. In the meantime, fully aggregated data can serve traffic information applications. On the other hand, if third-party data is to be used to complement existing data and power more complex freeway management applications such as performance monitoring and ramp metering, then it should be purchased in a disaggregated state.

Other aspects of data processing may be important to understand. Mobile data requires matching individual records onto the road network. This process is not trivial and providers should at a minimum shed some light on the algorithms they employ to do this. Likewise, data collection always involves outliers. As part of describing their data, providers should explain whether and how outliers are removed. Finally, Caltrans will need to know the formatting and delivery mechanism employed by vendors. Most providers deliver data over the internet using web standards, including various publish-subscribe protocols that stream XML or JSON files. Even with disaggregated data, one needs to consider the publication frequency, as it introduces latency in real-time applications. However, this should rarely be a concern today.

Geographical coverage

One obvious specification of a traffic data service is its intended area of coverage. With mobile data, intended coverage does not tell the whole story, because effective coverage will vary by location and time of day. However, in this context, coverage should be construed as a performance issue rather than a functional specification. This is addressed in section 5.2.2. More generally, providers should indicate which parts of the network their data will cover.

Metadata

Providers will often supplement their streams with metadata, which describes the attributes of the core traffic data that is delivered. For instance, a desirable metadata would be an indication of the estimated quality (or alternatively, an uncertainty level) of each data record. In the case of mobile data, this could consist of transmitting the number of probe vehicle observations that are included in each record. Likewise, fixed detector data should preferably include diagnostic information about the operational status of each detector. There are no set requirements for metadata, but it should be included in the evaluation criteria, the more complete the better.

5.2.2 PERFORMANCE SPECIFICATIONS

Performance specifications are the most important criteria that should drive decision-making about third-party traffic data. Section 4.1 lists the quality metrics that determine the performance of the data and should be employed in the context of data procurement. A more complex topic, however, is to establish a fair and efficient benchmark against which competing data services can be evaluated and compared. We touch on this topic in section 5.3.

Accuracy

Accuracy specifications require the use of ground truth data. As already pointed out, this is likely to be controversial because there are no cost-effective methods that can establish a perfect ground truth. As a result, the choice of ground truth could skew results toward one provider at the expense of another. Unfortunately, this is a necessary evil. The ground truth methodology needs to be spelled out up front as part of an RFP so that vendors can independently assess their willingness to compete with full knowledge of the rules.

For the hybrid data roadmap project, PATH employed Bluetooth detectors to establish ground truth. This methodology seems to meet a relatively high level of acceptance among data providers. Two recently published sets of guidelines recommend a similar approach[18, 19].

Completeness

Completeness can be easily assessed from a sample of the data. It is basically expressed as a percentage of complete records over a given time span and geographical area.

Validity

Measuring validity requires a robust definition that should be spelled out up front. In one view, every complete record can be measured for accuracy, whether or not it is deemed “valid”. This essentially eliminates the need to measure validity. Conversely, the inclusion of obvious outliers in accuracy measurements can severely downgrade a particular service even though it produces good data. One approach can consist of pushing the decision back onto vendors by requiring them to self-rate valid and invalid records. Only valid records are then included in accuracy measurements, but the percentage of invalid records should be examined carefully since it corresponds to unusable data.

[18] North American Traffic Working Group, "Traffic Information Benchmarking Guidelines" (2010)

[19] Virgina Department of Transportation, "Guidelines for Evaluating the Accuracy of Travel Time and Speed Data," (Richmond, VA, 2011)

Timeliness

Another label for timeliness is data latency. This is a relatively less important measure in the sense that data that is consistently late will also tend to be less accurate in real-time applications. It may therefore be appropriate to ask vendors to specify the timeliness of their data for information purposes, without the need to actually measure it.

Coverage

In the realm of fixed traffic detectors, coverage is a very straightforward concept: Coverage is determined by the location of those detectors. On the other hand, mobile data collection systems rely on a typically small sample of traveling vehicles. Because of this sampling, actual coverage varies by location and time of day: Most roadway segments may be completely "dark" during off-hours, whereas freeway coverage at peak times may be high. Measuring coverage of mobile data may be further complicated if the data is delivered in aggregate form since there is no way to assess with certainty the number of individual observations that make up each record. In fact, even disaggregated data may be spoofed with fake observations to create the illusion of volumes. Because of these difficulties, it may be advisable to require coverage estimates from providers of mobile probe data (which would likely be modulated by location and time of day), but to only use these estimates for informational purposes. Evaluation should instead rely on accuracy measures, which correspond to the desired outcome.

Accessibility

Accessibility is a relatively subjective metric, but it should nonetheless be assessed. In short, the question that needs to be asked is the level of effort required to integrate a given data service into existing information systems. This question would be best answered by a combination of information technology staff and data users.

5.2.3 DATA MANAGEMENT

The next area of specifications is concerned with the handling of personally identifiable data. This, of course, is only meaningful if the traffic data collected by a provider originates from a source that presents the potential for personal identification. Referring to section 2.3, the collection methods that have that potential include segment-based data collection and mobile data collection.

There are two aspects to this issue. The first is whether or not personal information can be extracted from the data that is provided to Caltrans. If that is the case, storing and using that data exposes Caltrans to potential liability. That is not to say that the corresponding risk is considerable, but it exists at least in theory and needs to be handled accordingly. The second aspect is whether or not the data provided to Caltrans presents any objective possibility of inferring personal information, how the provider handles, stores, and protects its data sources. Bad privacy protection practices or a security breach with a provider of traffic data could have implication for Caltrans either in terms of public relations or possible lawsuits. It is therefore important to review (and reserve the right to audit) both of these aspects: how the source data is handled by the provider, and what can be found in the data delivered to Caltrans.

Unfortunately these are fairly complicated topics, even from a theoretical standpoint. However, in the area of data security, best practices exist that could serve as a reference to evaluate various providers. The ability to identify an individual vehicle's path from the data and the measures that can protect this information are less well charted and should call for expert review.

5.2.4 LICENSING TERMS

Another important aspect of a data service's specifications is the set of licensing terms that are attached to it. In the first place, different providers allow different end uses of the data. The more common practice is to grant a public agency a license that lets the buyer share the data with other public agencies and with contractors, but bars open publishing of the data in raw format or sharing with third-party traffic providers. This is not insignificant for Caltrans since the PeMS systems also serves as a gateway for sharing data with value-added resellers. Some providers go a step further by licensing the data for specific uses, for instance a one-time planning exercise versus packaging the data for real-time traveler information.

In addition to the terms of the data service, Caltrans needs to require prospective vendors to disclose third-party agreements that govern the collection and processing of the source data. Such agreements may include agreements with fleet operators or cellular network operators. At any rate, they carry their own sets of terms that may have implications for Caltrans, and they can also be telling of the provider's ability to supply data in a sustainable way.

5.2.5 PRICING STRUCTURE

The pricing structure for traffic data services is widely variable, in large part because the market is relatively immature. Providers that operate roadway sensors will typically charge per station, making the pricing structure simple and transparent. On the other hand, suppliers of mobile data are providing value that is uncorrelated to their marginal costs, a situation similar to that of the software industry. Most of them charge a monthly or yearly subscription fee on a per mile basis. This pricing structure ignores the fact that the volume of data collected varies by network location and time of day, but taking those variations into account would be impractical.

As part of the hybrid data roadmap project, PATH also considered assessing fees on a per-data point basis. However, this approach has some severe drawbacks: The total costs are less predictable, making budgeting more difficult, and it would be easy for vendors to game this—for instance, by pulling back data during evaluation and negotiation, and/or adding fake data points once payments are flowing (auditing would be nearly impossible in either case). At the same time, given the intangible nature of a data service delivery, it may be unwise to commit up front to a set price. One possibility would be to agree to a nominal fee per unit of network length covered by the data service, but to tie payments to data quality metrics that are assessed by Caltrans. The attainment of data quality objectives in a given month would mean that the vendor receives the full fee, whereas a lower data quality would decrease their payment according to a preset schedule.

5.3 CONTRACT MANAGEMENT

Besides writing good specifications, Caltrans also needs to be concerned with the management of the RFP process and the ongoing contractual relationship with data suppliers. This section highlights a few recommended practices and takeaways from the procurement conducted by PATH as part of the hybrid data roadmap project.

5.3.1 PROCUREMENT PROCESS

As previously mentioned, a salient characteristic of the traffic data market is its relative lack of maturity. Government procurement rules are primarily designed to purchase commodity goods and services, and the innovative nature of traffic data services imposes a burden on the agencies that attempt to procure them. Nonetheless, several of them have done so successfully in the past few years. The PATH team conducted a series of interviews with state DOTs (Georgia, Minnesota, Wisconsin, and I-95 Coalition) that have worked with private traffic data providers. Although only two of the projects went through a standard RFP process, the team was able to identify the following recommendations based on overall experiences:

- Consider a Request for Information (RFI) to develop specifications for the RFP**.** The I-95 Coalition used two RFIs to refine their RFP specifications. For the first RFI, they shared their vision for the project and elicited ideas from the vendors. They used the responses to develop specifications for the traffic data for a second RFI, which gave the vendors a chance to comment. The result was that the 5–6 proposals submitted for the RFP were all on target.

- Meet early with the procurement staff**.** Since procuring ongoing data services is somewhat different from either goods (computers, office equipment, etc.) or services (catering, consulting, etc.), the procurement staff may need to be educated on some of the differences and the purpose of the ongoing service.

- Include a clear exit strategy. Be explicit about how you can cancel the contract if you are not getting what you want.

- Define optional versus required features. To minimize the chance that no contract is awarded, a set of required features should be distinguished from a set of optional or desired provisions. This will allow the vendors to have some flexibility in how they respond to the RFP. An alternative way of implementing this would be to have levels of compliance with feature requests—the higher the level of compliance, the more points awarded.

- Establish your validation process in advance of the data procurement. Some of the project managers we spoke with mentioned that it was time-consuming to develop the validation process after the data arrived. However, other groups, such as the I-95 Coalition, have continually refined their validation process.

- Create benchmarks for delivering the data feeds. Several of the previous projects experienced delay in receiving the data feeds. Incorporating timely benchmarks into the RFP will force vendors to account for them in their proposal. Vendors should also be asked to describe their current coverage areas and the rate at which they plan to expand geographic coverage and to what areas.

- Create milestones for different phases in the project**.** In addition to timely data delivery, some DOTs specified milestones for acceptance testing and integration into existing DOT systems.

- Consider using a testing stage before moving to contract. Wisconsin DOT did not proceed to a contract with Airsage after the data performed poorly during acceptance testing. This saved the DOT the costs and time of proceeding with a contract that would not bring value.

- Consider an ongoing monitoring system. Although acceptance and validation testing are essential, some DOTs noticed that traffic data providers would sometimes revert to historical or sensor data.

- Specify customer service response times. Only Wisconsin DOT required customer service provisions, i.e., high priority problems being responded to within two hours. Some of these may be part of the standard contract offered by private traffic data providers, but this should be confirmed.

The PATH team also reached its own conclusions about the features of a good RFP after going through that process itself:

- It provides a way of determining the quality (appropriateness) of the product to be purchased in a way that can be compared to others.

- It stimulates the market to respond: It is easy to respond to, open to some interpretation, and does not ask for too much sensitive information. We need to think about how this data could be used to destroy the credibility of an organization if used badly. For pragmatic reasons we need to be considerate of the risks our responders (hopefully, partners) are taking.

- It promotes good discussion: It does not appear to provide a solution, merely states a need.

- It helps tease out appropriate price points by being written to encourage segmentation on aspects that are important to us: loops versus no loops, time of day, and probably complex road areas that require more data regardless of the application.
- It is legally sound: We follow good procedures and identify the areas that will affect our decisions.

As previously mentioned, the evaluation of proposals must necessarily improve a data quality benchmark procedure. This requires a proper methodology that has been communicated clearly to prospective vendors. The cost of designing, setting up, carrying out, and analyzing such a benchmark will not be insignificant and should therefore be factored into budget decisions. All the specifications listed in section 5.2 should be assessed, along with more traditional criteria such as corporate information, past experience, and overall price.

Negotiations between Caltrans and short-listed vendors are also likely, again in large part due to the relatively uncharted nature of this type of procurement. The PATH team experience involved back and forth discussions of contract terms to align the legal, regulatory, business, and technical requirements of both parties and reach agreement on such issues as:

- Licensing a perpetual use of the purchased data
- Permitting mixing the data with that from other sources
- Confidentiality of the service
- Overall price

5.3.2 CONTRACT MONITORING

A final note on procurement concerns contract monitoring. Designing robust and well thought-out contract monitoring procedures is essential owing to the intangible nature of the service delivered. There is almost no way to tell from the data itself whether it is worth what one is paying for it. The only way to make that assessment is by implementing an ongoing data quality monitoring program. That program need not be very extensive because quality can be checked based on small samples. The I-95 Corridor Coalition has been monitoring the quality of the data it receives from INRIX on a continuous basis by using Bluetooth readers that are rotated across various locations at regular intervals. This approach is similar to the one used by PATH to assess ground truth as part of the hybrid data roadmap project. If need be, the ongoing data quality program can be implemented by a contractor, although strict controls of the ongoing relationship that the contractor may have with data vendors should be exercised. Data quality could be tied to vendor fees to set an incentive, or otherwise be part of regular reviews with vendors, in which case the incentive could be contract renewal, which is less direct but more conducive to a long-term partnership.

In addition to ongoing traffic data quality monitoring, contracts should incorporate a service level agreement that specifies service availability. This is a traditional feature of online services contracts and can also be tied to payments.

Chapter 7

Conclusion

1 SUMMARY OF THE PROJECT

Faced with the rising cost of maintaining its traditional traffic data collection systems, such as loop detectors installed at fixed locations, and recognizing the growing prevalence of commercial traffic data sources, Caltrans is looking into purchasing probe data from the commercial sector. To help with that effort, PATH undertook the *Hybrid Traffic Data Collection Roadmap: Objectives and Methods* (Task Order 2) that investigated the processes and algorithms required to assimilate probe data (unaggregated GPS point speeds) and fuse it with Caltrans' existing data for the purpose of estimating travel times. The task order also examined the business case for purchasing and integrating probe data. In conjunction with the related Task Order 1 (*Pilot Procurement of Third-Party Traffic Data*), the intent was to demonstrate an efficient and cost-effective use of alternative traffic data sources to complement the detection systems currently installed and operated by Caltrans.

Research efforts focused on the following areas:

- **Data quality assessment**—We investigated the issue of assessing data quality in an era of ubiquitous probe data, outlining the definitions and criteria needed to measure data quality (accuracy, completeness, validity, etc.), a methodology for assessing it, the characteristics needed for a reference state to represent ground truth, and a multi-level validation methodology. A Data Quality Tool was also developed for examining the characteristics of probe data feeds directly.

- **Probe data quality**—To examine the implications of using varying amounts of probe data, we studied data collected from 100 GPS-equipped cars used as probe vehicles driving for multiple hours on a section of Bay Area roadway. Using algorithms, we modified the penetration rates and VTL (virtual trip line) locations to study the effects of the changes on the ability to estimate accurate speeds for a roadway. This study was the precursor to, and informed the design and methodology of, the further analysis of data fusion performed under Task Order 1.

- **Data fusion implementation**—We studied how to fuse multiple data sources with various characteristics by running probe and loop data through the Mobile Millennium highway model, which generated velocity maps and travel times. The Mobile Millennium system can accept data from traditional sources (such as occupancy and counts from loop detectors) and point-speed measurements from providers of probe data. This enabled us to evaluate the performance of the data sources both individually and when fused together.

- **Sensitivity analysis of loop detector spacing and location**—To analyze the sensitivity of the spacing and location of fixed sensors along the roadway, we algorithmically simulated the removal of loop detectors to test whether loops could be placed further apart and still provide sufficient information to generate accurate traffic estimates.

- **Balancing loop and probe data**—By adjusting the amount of data from loops and probes in combination, we were able to create multiple scenarios with different proportions of loop and probe data and evaluate the impact on computing travel times. A total of 1,637 scenarios were developed by instantiating all combinations of 9 sets of inductive loop detector data sets (from 0 to 16 detectors), 11 probe penetration rates, and various space-based and time-based sampling strategies.

- **Hybrid data roadmap**—A hybrid traffic data roadmap document was developed that provides an overarching view of the context, objectives, and implementation of a hybrid traffic data system. Drawing on the full scope of work completed for Task Orders 1 and 2, the roadmap includes a business analysis assessing the benefits, trade-offs, and next steps in procuring third-party data and integrating it into Caltrans' existing structure.

2 SUMMARY OF FINDINGS

Data quality

The quality of probe data can be measured and compared to ground truth. As outlined in Chapter 2, combining the measurements source with a traffic model within an estimation framework can provide levels of accuracy associated with the robustness of the model and the design and purpose of the estimation method (such as estimating travel times). In addition, the Data Quality Tool described in Chapter 3 can visually display the the attributes of probe data feeds for a variety of metrics.

Speed data and travel times

GPS point-speed data is usable for the intended application (estimating speed data and travel times) and can be successfully processed with the Mobile Millennium system to map velocities.

Data filtering and map-matching

Point speeds from GPS-equipped probe vehicles can be filtered to remove faulty data, and the vehicles and their trajectories accurately mapped to the road network, as detailed in Chapter 4. This is essential for developing reliable estimates of velocities and travel times along the highway.

Loop detector spacing

In this study, as described in Chapter 5, spacing loop detectors less than an average of 0.83 miles apart (i.e., using data from more than eight inductive loop detector stations along the stretch of roadway under study) did not provide extra benefit in the travel time estimation. The error remains constant between 6–13% depending on the time of day, regardless of the added loop detector stations.

Quantity of probe data

In this study, when sampling probe vehicles at a rate of 137.5 veh/hr with more than 2.54 VTL/mi, increasing the number of probe measurements by adding more probe vehicles or additional virtual trip lines caused only small improvement in the travel time accuracy.

Mixing probe and loop detector data

Probe data can be successfully fused with loop detector data, and meaningful comparisons can be assessed. It was found that when complementing loop detector data with probe vehicle data, better estimates for travel times are obtained, especially at low penetration rates. In this study, for example, if using loop detectors spaced more than 2.11 miles apart, probe data can give over 50% increase in the travel time accuracy. These results hold generally, independent of the sampling strategy of the probe vehicles.

Confidence in the model

In this study, it was found that when using a flow model with data assimilation, dynamic travel times can be estimated with less than 10% error by using either inductive loop detector data, probe data, or a mixture of both. The quantitative and graphical results of the research give us confidence in both the modeling approach to roadway estimation and the effectiveness of the Mobile Millennium highway model.

Costs, benefits, trade-offs

When examining the trade-offs between the costs and performance of available data sources, we found that the probe data purchased for the hybrid data roadmap project could often match existing loop detectors in producing consistent travel time estimates and can do so at lower expense, especially if detector maintenance and replacement costs are factored in. The highest quality estimates, however, were achieved by combining both probe data and loop detector data. Moreover, this project focused on only one application of traffic information—travel time estimation—and did not investigate other traffic management applications such as control strategies.

3 IMPLICATIONS AND FUTURE DIRECTIONS

The successful procurement and processing of probe data for travel time estimation, as demonstrated in Task Orders 1 and 2, illuminates the complexities, challenges, and opportunities of the new world in which transportation agencies find themselves. Some of the implications to emerge from the project include:

Reduced dependency on loop data

While current system control strategies, such as ramp metering, require density data, it seems difficult to significantly increase the quantity of loop detectors on California roads. At the same time, the penetration rate of probe data is continually increasing and far from reaching its limits. This represents a sea change in the types of data available for traffic management and offers the prospect of migrating away from exclusive dependency on loop detectors over time.

Outsourced data collection

Purchasing probe data from the commercial sector means, in effect, outsourcing the collection of traffic data. Any such undertaking comes with new risks (e.g., data quality, privacy protection, business continuity) which would need to be managed through, for example, a careful vetting and data acquisition process and robust data assessment tools, processes, and standards.

Redesigned information systems

The research work in this project was predicated on having a data assimilation and state estimation system in place that would allow the implementation, testing, and analysis of data hybridization. This required:

- Building and calibrating a model to estimate speed and travel times from probe data
- Developing a set of methods to fuse probe and loop detector data
- Creating tools to visualize the data
- Testing the tools and methods on real data from pilot sites
- Building tools to determine the quality of the methods, models, and data

Creating this mathematical and technology infrastructure points the way to the redesign of information systems that would make it possible to implement data fusion and take full advantage of hybrid data in traffic management systems.

New detector strategy

While further research on the position and spacing of loop detectors is needed, our initial results suggest that using the most critical detectors (those that add the most information value) rather than enforcing a minimal spacing between detectors could help optimize the existing stock of detectors and allow Caltrans to selectively focus maintenance efforts or supplement loop data with probe data when certain loops fail.

Augmented traffic measurements

This project studied the use of probe data for estimating travel times. However, the enhanced modeling and estimation accuracy demonstrated by data fusion also lays the foundation for better control strategies and operational decision-making. Augmenting traffic volume measurement with probe data, for example, could be a fruitful area for research in the future. Fused loop and probe data ("hybrid" data) could thus provide a pathway to the development and use of additional traffic measurements, such as arterial estimation, origin-destination information, demand modeling, and, ultimately, integrated corridor management.

The road ahead

The Hybrid Data Roadmap (Chapter 6) outlines a possible implementation path that would enable Caltrans to gradually experiment with and adopt a hybrid traffic data collection system. It involves three phases that could be developed over a time horizon of three to five years:

1. Short-term pilot in a selected district (1–2 years)

2. Using the PATH Connected Corridors project to spearhead information systems innovations (2–3 years)

3. Full-scale pilot in a selected district (3–5 years)

These phases would allow Caltrans to leverage what was learned from Task Orders 1 and 2, build on the mathematical and technology infrastructure already in place at PATH, and test the viability of hybrid traffic data collection in a controlled manner. They would thus provide a way to effectively manage the next steps into a new traffic information environment.

Bibliography

REFERENCES

[1] B. Greenshields, "A study of traffic capacity," in *Proceedings of the 14th annual meeting of the Highway Research Board*, 1935.

[2] T. Hey, S. Tansley and K. Tolle, Eds., The Fourth Paradigm: Data-Intensive Scientific Discovery, Redmond, Washington: Microsoft Research, 2009.

[3] Freeway Performance Measurement System, 2010. [Online]. Available: http://pems.eecs.berkeley.edu.

[4] Mobile Millennium, [Online]. Available: http://traffic.berkeley.edu.

[5] B. Hoh, M. Gruteser, R. Herring, J. Ban, D. Work, J.-C. Herrera, A. Bayen and Q. Jacobson, "Virtual trip lines for distributed privacy-preserving traffic monitoring," in *6th International Conference on Mobile Systems, Applications, and Services*, Breckenridge, CO, 2008.

[6] Google, 2010. [Online]. Available: http://www.google.com.

[7] Inrix, 2010. [Online]. Available: http://www.inrix.com.

[8] Waze, 2010. [Online]. Available: http://www.waze.com.

[9] D. Work and A. Bayen, "Impacts of the mobile internet on transportation cyberphysical systems: traffic monitoring using smartphones," in *National Workshop for Research on High-Confidence Transportation Cyber-Physical Systems: Automotive, Aviation and Rail*, Washington, D.C., 2008.

[10] Sense Networks, 2010. [Online]. Available: http://www.sensenetworks.com.

[11] F. Wang, "Toward a paradigm shift in social computing: The ACP approach," *IEEE Intelligent Systems,* p. 65–67, 2007.

[12] S. Turner, "Defining and measuring traffic data quality: White paper on recommended approaches," *Transportation Research Record: Journal of the Transportation Research Board,* vol. 1870, p. 62–69, 2004.

[13] J.-D. Margulici and X. Ban, "Benchmarking travel time estimates," *Intelligent Transport Systems,* vol. 2, no. 3, p. 228–237, 2008.

[14] B. Smith, "Purchasing travel time data: Investigation of travel time data service requirements," *Transportation Research Record: Journal of the Transportation Research Board,* vol. 1978, p. 178–

183, 2006.

[15] A. Toppen and K. Wunderlich, "Travel time data collection for measurement of advanced traveler information systems accuracy," in *Proceedings of the14th ITS America Annual Meeting*, San Antonio, 2004.

[16] X. Ban, Y. Li and A. Skabardonis, "Local mad method for probe vehicle data processing," in *Proceedings of the 14th world congress on intelligent transport systems*, Beijing, 2007.

[17] J.-C. Herrera, D. Work, X. Ban, R. Herring, Q. Jacobson and A. Bayen, "Evaluation of traffic data obtained via gps-enabled mobile phones: the mobile century field experiment," *Transportation Research Part C,* vol. 18, p. 568–583, 2009.

[18] Office of Highway Policy Information, Federal Highway Administration, "Final Report. Traffic data quality measurement," US Department of Transportation, Washington, D.C., 2004.

[19] Paramics, [Online]. Available: http://www.paramics.com.

[20] TOPL, [Online]. Available: http://path.berkeley.edu/topl.

[21] "Google transit feed specification," 2010. [Online]. Available: http://code.google.com/transit/spec/transit.

[22] Cabspotting, 2010. [Online]. Available: http://cabspotting.org.

[23] Info24, 2010. [Online]. Available: http://www.info24.se.

[24] G. Evensen, Data Assimilation: The Ensemble Kalman Filter, New York: Springer, 2009.

[25] Y. Wang, M. Papageorgiou and A. Messmer, "Real-time freeway traffic state estimation based on extended Kalman filter: A case study," *Transportation Science,* vol. 41, no. 2, p. 167, 2007.

[26] D. Work, S. Blandin, O.-P. Tossavainen, B. Piccoli and A. Bayen, "Highway traffic velocity estimation on networks," *Applied Mathematics Research Express,* 2010.

[27] A. Bayen, "Mobile Century Final Report: A Traffic Sensing Field Experiment Using GPS Mobile Phones," University of California, Berkeley, 2010.

[28] A. Bayen, J. Butler and A. Patire, "Mobile Millennium final report," University of California, Berkeley, 2011.

[29] Y. Cui and S. Ge, "Autonomous vehicle positioning with gps in urban canyon environments," *Robotics and Automation,* vol. 19, no. 1, p. 15–25, 2003.

[30] S. Thrun, "Probabilistic robotics," *Communications of the ACM,* vol. 45, no. 3, p. 52–57, 2002.

[31] T. Miwa, T. Sakai and T. Morikawa, "Route identification and travel time prediction using probe-car data," *International Journal,* 2004.

[32] J. Du, J. Masters and M. Barth, "Lane-level positioning for in-vehicle navigation and automated vehicle location (avl) systems," in *Proceedings of the 7th International IEEE Conference on Intelligent Transportation Systems*, 2004.

[33] NAVTEQ Inc., [Online]. Available: http://www.navteq.com.

[34] Telenav Inc., [Online]. Available: http://www.telenav.com.

[35] J. Bentley and H. Maurer, "Efficient worst-case data structures for range searching," *Acta Informatica,* vol. 13, p. 155–168, 1980.

[36] M. Quddus, W. Ochieng, L. Zhao and R. Noland, "A general map matching algorithm for transport telematics applications," *GPS solutions,* vol. 7, no. 3, p. 157–167, 2003.

[37] C. E. White, D. Bernstein and A. L. Kornhauser, "Some map matching algorithms for personal navigation assistants," *Transportation Research Part C: Emerging Technologies,* vol. 8, no. 1-6, p. 91–108, 2000.

[38] J. Yuan, Y. Zheng, C. Zhang, X. Xie and G. Sun, "An interactive-voting based map matching algorithm," in *Eleventh International Conference on Mobile Data Management (MDM)*, 2010.

[39] J. Greenfeld, "Matching GPS observations to locations on a digital map," in *81th Annual Meeting of the Transportation Research Board*, 2002.

[40] S. Brakatsoulas, D. Pfoser, R. Salas and C. Wenk, "On map-matching vehicle tracking data," in *Proceedings of the 31st international conference on Very large data bases*, 2005.

[41] C. Wenk, R. Salas and D. Pfoser, "Addressing the need for map-matching speed: Localizing global curve-matching algorithms," in *18th International Conference on Scientific and Statistical Database Management*, 2006.

[42] J. Pyo, D. Shin and T. Sung, "Development of a map matching method using the multiple hypothesis technique," in *Proceedings of the IEEE conference on Intelligent Transportation Systems*, 2001.

[43] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson and P. Nordlund, "Particle filters for positioning, navigation, and tracking," *IEEE Transactions on Signal Processing,* vol. 50, no. 2, p. 425–437, 2002.

[44] W. Ochieng, M. Quddus and R. Noland, "Map-matching in complex urban road networks," *Revista Brasileira de Cartografia,* vol. 2, no. 55, 2009.

[45] M. Bierlaire and G. Flötteröd, "Probabilistic multi-modal map matching with rich smartphone data," *STRC 2011,* 2011.

[46] Y. Zheng and M. Quddus, "Weight-based shortest-path aided map-matching algorithm for low-frequency positioning data," in *Transportation Research Board 90th Annual Meeting*, 2011.

[47] L. Giovannini, A Novel Map-Matching Procedure for Low-Sampling GPS Data with Applications to Traffic Flow Analysis, Universita di Bologna, 2011.

[48] A. Thiagarajan, L. Ravindranath, K. LaCurts, S. Toledo, J. Eriksson, S. Madden and H. Balakrishnan, "Vtrack: Accurate, energy-aware traffic delay estimation using mobile phones," in *7th ACM Conference on Embedded Networked Sensor Systems (SenSys)*, Berkeley, 2009.

[49] J. Lafferty, A. McCallum and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of the Eighteenth International Conference on Machine Learning*, San Francisco, 2001.

[50] M. Quddus, W. Ochieng and R. Noland, "Current map-matching algorithms for transport applications: State-of-the art and future research directions," *Transportation Research Part C: Emerging Technologies,* vol. 15, no. 5, p. 312–328, 2007.

[51] M. El Najjar and P. Bonnifait, "A road-matching method for precise vehicle localization using belief theory and kalman filtering," *Autonomous Autonomous,* vol. 19, no. 2, p. 173–191, 2005.

[52] S. Syed and M. Cannon, "Fuzzy logic-based map matching algorithm for vehicle navigation system in urban canyons," in *ION National Technical Meeting*, San Diego, 2004.

[53] P. Hart, N. Nilsson and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE transactions on Systems Science and Cybernetics,* vol. 4, no. 2, p. 100–107, 1968.

[54] K. Murphy, *Dynamic Bayesian Networks: Representation, Inference and Learning,* PhD thesis, University of California at Berkeley, 1994.

[55] M. Bierlaire and E. Frejinger, "Route choice modeling with network-free data," *Transportation Research Part C: Emerging Technologies,* vol. 16, no. 2, p. 187–198, 2008.

[56] G. Forney Jr., "The Viterbi algorithm," in *Proceedings of the IEEE*, 1973.

[57] K. Seymore, A. McCallum and R. Rosenfeld, "Learning hidden Markov model structure for information extraction," in *AAAI-99 Workshop on Machine Learning for Information Extraction*,

1999.

[58] J. Bilmes, "A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models," *International Computer Science Institute,* vol. 4, 1998.

[59] "Supporting code for the path inference filter," [Online]. Available: https://github.com/tjhunter/Path-Inference-Filter.

[60] A. Thiagarajan, J. Biagioni, T. Gerlich and J. Eriksson, "Cooperative transit tracking using smart-phones," in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, 2010.

[61] S. Boyd and L. Vandenberghe, Convex optimization, Cambridge University Press, 2004.

[62] T. Moon, "The expectation-maximization algorithm," *Signal Processing Magazine, IEEE,* vol. 13, no. 6, p. 47–60, 1996.

[63] T. Hunter, T. Moldovan, M. Zaharia, J. Ma, S. Merzgui, M. Franklin and A. Bayen, "Scaling the mobile millennium system in the cloud," in *ACM Symposium on Cloud Computing*, 2011.

[64] H. Bar-Gera, "Evaluation of a cellular phone-based system for measurements of traffic speeds and travel times: A case study from Israel," *Transportation Research Part C,* vol. 15, no. 6, p. 380–391, 2007.

[65] H. Liu, A. Danczyk, R. Brewer and R. Starr, "Evaluation of cell phone traffic data in Minnesota," *Transportation Research Record,* no. 2086, p. 1–7, 2008.

[66] J. Kwon, K. Petty and P. Varaiya, "Probe vehicle runs or loop detectors?," *Transportation and Research Record,* no. 2012, p. 57 – 63, 2007.

[67] J.-C. Herrera and A. Bayen, "Incorporation of lagrangian measurements in freeway traffic state estimation," *Transportation Research B,* vol. 44, no. 4, p. 160–481, 2010.

[68] M. Lighthill and G. Whitham, "On kinematic waves. II. A theory of traffic flow on long crowded roads," in *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 1955.

[69] P. Richards, "Shock waves on the highway," *Operations Research,* vol. 4, no. 1, p. 42–51, 1956.

[70] D. Work, S. Blandin, O.-P. Tossavainen, B. Piccoli and A. Bayen, "A distributed highway velocity model for traffic state reconstruction," *Applied Mathematics Research eXpress (AMRX),* vol. 2010, p. 1–35, 2010.

[71] A. Downs, Still stuck in traffic: coping with peak-hour traffic congestion, Brookings Institution Press, 2004.

[72] K. Liu, T. Yamamoto and T. Morikawa, "Study on the cost-effectiveness of a probe vehicle system at lower polling frequencies," *International Journal of ITS Research,* vol. 6, no. 1, p. 29–36, 2008.

[73] D. Schrank, T. Lomax and Texas Transportation Institute, "2009 Urban mobility report," The Texas A&M University, 2009.

[74] D. Work, O. Tossavainen, S. Blandin, A. Bayen, T. Iwuchukwu and K. Tracton, "An ensemble Kalman filtering approach to highway traffic estimation using GPS enabled mobile devices," in *Proceedings of the 47th IEEE Conference on Decision and Control*, Cancun, 2008.

[75] C. Daganzo, "The cell transmission model: a dynamic representation of highway traffic consistent with the hydrodynamic theory," *Transportation Research Part B,* vol. 28, no. 4, p. Transportation Research Part B, 1994.

[76] C. Daganzo, "The cell transmission model, part II: network traffic," *Transportation Research Part B,* vol. 29, no. 2, p. 79–93, 1995.