Lane-Level Localization and Map Matching for Advanced Connected and Automated Vehicle (CAV) Applications

March 2023

A Research Report from the National Center for Sustainable Transportation

Jay A. Farrell, University of California, Riverside

Guoyuan Wu, University of California, Riverside

Wang Hu, University of California, Riverside

David Oswald, University of California, Riverside

Peng Hao, University of California, Riverside

National Center for Sustainable Transportation

UCR | College of Engineering- Center for Environmental Research & Technology

TECHNICAL REPORT DOCUMENTATION PAGE

1. Report No. NCST-UCR-RR-23-13	**2. Government Accession No.** N/A	**3. Recipient's Catalog No.** N/A

4. Title and Subtitle Lane-Level Localization and Map Matching for Advanced Connected and Automated Vehicle (CAV) Applications	**5. Report Date** March 2023
	6. Performing Organization Code N/A
7. Author(s) Jay A. Farrell, Ph.D., https://orcid.org/0000-0002-2077-8691 Guoyuan Wu, Ph.D., https://orcid.org/0000-0001-6707-6366 Wang Hu, https://orcid.org/0000-0002-6416-3875 David Oswald, https://orcid.org/0000-0003-2307-1437 Peng Hao, https://orcid.org/0000-0001-5864-7358	**8. Performing Organization Report No.** N/A
9. Performing Organization Name and Address University of California, Riverside Bourns College of Engineering – Center for Environmental Research & Technology 1084 Columbia Avenue, Riverside, CA 92507	**10. Work Unit No.** N/A
	11. Contract or Grant No. USDOT Grant 69A3551747114
12. Sponsoring Agency Name and Address U.S. Department of Transportation Office of the Assistant Secretary for Research and Technology 1200 New Jersey Avenue, SE, Washington, DC 20590	**13. Type of Report and Period Covered** Final Research Report (April 2021 – March 2022)
	14. Sponsoring Agency Code USDOT OST-R

15. Supplementary Notes DOI: https://doi.org/10.7922/G25T3HSS Dataset DOI: https://doi.org/10.6086/D11M43
16. Abstract Reliable, lane-level, absolute position determination for connected and automated vehicles (CAV's) is near at hand due to advances in sensor and computing technology. These capabilities in conjunction with high-definition maps enable lane determination, per lane queue determination, and enhanced performance in applications. This project investigated, analyzed, and demonstrated these related technologies. Project contributions include: (1) Experimental analysis demonstrating that the USDOT Mapping tool achieves internal horizontal accuracy better than 0.2 meters (standard deviation); (2) Theoretical analysis of lane determination accuracy as a function of both distance from the lane centerline and positioning accuracy; (3) Experimental demonstration and analysis of lane determination along the Riverside Innovation Corridor showing that for a vehicle driven within 0.9 meters of the lane centerline, the correct lane is determined for over 90% of the samples; (4) Development of a VISSIM position error module to enable simulation analysis of lane determination and lane queue estimation as a function of positioning error; (5) Development of a lane-level intersection queue prediction algorithm; Simulation evaluation of lane determination accuracy which matched the theoretical analysis; and (6) Simulation evaluation of lane queue prediction accuracy as a function of both CAV penetration rate and positioning accuracy. Conclusions of the simulation analysis in item (6) are the following: First, when the penetration rate is fixed, higher queue length estimation error occurs as the position error increases. However, the disparity across different position error levels diminishes with the decrease of penetration rate. Second, as the penetration rate decreases, the queue length estimation error significantly increases under the same GNSS error level. The current methods that exist for queue length prediction only utilize vehicle position and a penetration rate estimate. These results motivate the need for new methods that more fully utilize the information available on CAVs (e.g., distance to vehicles in front, back, left, and right) to decrease the sensitivity to penetration rate.

17. Key Words Connected and automated vehicles, lane determination, lane-level queue prediction, intersection traffic management, lane-level positioning	**18. Distribution Statement** No restrictions.

19. Security Classif. (of this report) Unclassified	**20. Security Classif. (of this page)** Unclassified	**21. No. of Pages** 67	**22. Price** N/A

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

About the National Center for Sustainable Transportation

The National Center for Sustainable Transportation is a consortium of leading universities committed to advancing an environmentally sustainable transportation system through cutting-edge research, direct policy engagement, and education of our future leaders. Consortium members include: University of California, Davis; University of California, Riverside; University of Southern California; California State University, Long Beach; Georgia Institute of Technology; and University of Vermont. More information can be found at: ncst.ucdavis.edu.

Disclaimer

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated in the interest of information exchange. The report is funded, partially or entirely, by a grant from the U.S. Department of Transportation's University Transportation Centers Program. However, the U.S. Government assumes no liability for the contents or use thereof.

The U.S. Department of Transportation requires that all University Transportation Center reports be published publicly. To fulfill this requirement, the National Center for Sustainable Transportation publishes reports on the University of California open access publication repository, eScholarship. The authors may copyright any books, publications, or other copyrightable materials developed in the course of, or under, or as a result of the funding grant; however, the U.S. Department of Transportation reserves a royalty-free, nonexclusive and irrevocable license to reproduce, publish, or otherwise use and to authorize others to use the work for government purposes.

Acknowledgments

This study was funded, partially or entirely, by a grant from the National Center for Sustainable Transportation (NCST), supported by the U.S. Department of Transportation (USDOT) through the University Transportation Centers program. The authors would like to thank the NCST and the USDOT for their support of university-based research in transportation, and especially for the funding provided in support of this project.

NCST

Lane-Level Localization and Map Matching for Advanced Connected and Automated Vehicle (CAV) Applications

A National Center for Sustainable Transportation Research Report

March 2023

Jay A. Farrell, **Wang Hu, and David Oswald**

Department of Electrical and Computer Engineering, University of California, Riverside

Peng Hao and Guoyuan Wu

College of Engineering –Center for Environmental Research and Technology, University of California, Riverside

[page intentionally left blank]

TABLE OF CONTENTS

List of Figures

Lane-Level Localization and Map Matching for Advanced Connected and Automated Vehicle (CAV) Applications

EXECUTIVE SUMMARY

Connected and Automated Vehicles (CAVs) have the potential for significant impact on numerous socio-economic and environmental issues of contemporary transportation systems. CAV applications are intended to decrease the frequency and severity of accidents, mitigate congestion, reduce energy consumption and pollutant emissions, and enhance system resilience and efficiency.

Many CAV applications achieve enhanced performance from reliable, lane-level, absolute position determination, which is near at hand due to advances in sensor and computing technology. The advent of this capability enables lane determination and applications that rely upon it. Another precondition of such applications is accurate digital maps for each lane of the roadway.

This project investigated, analyzed, and demonstrated these related technologies. Project contributions include:

- Experimental data to analyze the lane-level accuracy of the USDOT Mapping tool. Its internal accuracy is better than 0.2 meters (standard deviation). However, some accuracy is lost in the conversion to the J2735 file. See Task 2 and Appendix A.
- Theoretical analysis of lane determination accuracy as a function of both distance from the lane centerline and positioning accuracy. See Task 3.
- Experimental demonstration and analysis of lane determination along the Riverside Innovation corridor. Conclusions of the experimental demonstration include the following. (1) For a vehicle driven within 0.9 meters of the lane centerline, the correct lane is determined for over 90% of the samples (See Figure 6); and, (2) Lane width is unequal on the left and right sides of the lane centerline and varies with position along the centerline. These facts must be addressed in lane determination to achieve high performance. See Task 3 and Task 5.
- Development of a VISSIM position error module to enable simulation analysis of lane determination and lane queue estimation as a function of positioning error. See Appendix B.
- Development of a lane-level intersection queue prediction algorithm. See Appendix C.
- Simulation evaluation of lane determination accuracy which matched the theoretical analysis of Task 3. See Task 6.
- Simulation evaluation of lane queue prediction accuracy as a function of both CAV penetration rate and positioning accuracy. Conclusions of this analysis are the following. 1) When the penetration rate is fixed, higher queue length estimation error occurs as the position error increases. However, the disparity across different position error levels

diminishes with the decrease of penetration rate; 2) As the penetration rate decreases, the queue length estimation error significantly increases under the same GNSS error level. See Task 6.

All data related to this project is available as described in the Data Summary portion of this report.

Introduction

Connected and Automated Vehicles (CAVs) have received significant attention from industry, government, and academia as an emerging solution to numerous socio-economic and environmental issues in our contemporary transportation systems. A variety of CAV applications have been developed to decrease the frequency and severity of accidents, mitigate congestion, monitor the traffic network, reduce energy consumption and pollutant emissions, as well as enhance system resilience and efficiency [Tian et al., 2019; Williams et al., 2021].

Many CAV applications achieve enhanced performance from increasingly accurate and reliable vehicle position determination. Due to rapid decrease in the cost and size of sensors, vehicle position accuracy and reliability is rapidly increasing. Accuracy and reliability is typically achieved via the use of Global Navigation Satellite Systems (GNSS) for absolute position determination in conjunction with other sensors (e.g., cameras, Lidar, wheel encoders, Inertial Measurement Units) [Farrell et al., 1999; Farrell 2008; Toledo-Moreo et al., 2009; Groves, 2013; Nedevschi et al., 2013; Rose et al., 2014] for relative position information, and contextual information (e.g., high-definition maps) [Toledo-Moreo et al., 2009; Groves, 2013; Nedevschi et al., 2013]. With these advances, the advent of reliable absolute lane determination and lane-level positioning is at hand.

This project investigated and demonstrated the utility of lane-level map-matching and localization through the following tasks:

1. Literature review: This literature review focuses on the question of how recent and ongoing advances that enable lane-level vehicle position determination will provide benefits in CAV applications. The main goal is to a select an application and underlying technologies suitable for implementation demonstration within the Riverside Innovation Corridor (Task 5) and study in VISSIM (Task 6).
2. Lane-Level Mapping: This task considers the state of lane-level mapping with a focus on which tools are available to the academic researcher. One tool was selected and its mapping accuracy is characterized.
3. Lane-Level Map Matching: This task describes the lane determination algorithm that was used in Tasks 5 and 6 and analyzes its expected performance at various levels of position accuracy.
4. Mid-year Progress Report: This report documented the status of the project as of October 1, 2021.
5. Demonstration: Lane-level map matching is a technology that underlies various applications: intersection signal phase and timing, lane selection, lane-changing/lane-keeping assistance, lane-level routing/navigation, intersection collision avoidance (See Task 2). This task demonstrates and evaluates the probability of correct lane determination along the Riverside Innovation Corridor.
6. Application Simulation Evaluation: Lane-level queue prediction (LLQP) is needed for real-time traffic surveillance at signalized intersections. It enables effective lane-level signal phase and timing control and advanced driving assistance functions. An algorithm to

achieve real-time LLQP is presented in Appendix C. Accuracy of this LLQP algorithm is studied in VISSIM as a function of CAV penetration rate and GNSS position accuracy.

7. Data Management Compliance: This task describes the data products of this project and how that data is archived for public utility.
8. Final Report: This task documents the final report.

Each task is discussed in greater detail in a subsequent section of this final report.

This one-year effort has demonstrated the feasibility and assessed the reliability of real-time lane-level map matching and assessed its utility for lane-level queue determination for CAV applications. Lane-level mapping, lane-identification, lane-relative position determination, and lane-level queue prediction would benefit many sustainable transportation applications. Also, the technologies involved in this project are vehicle agnostic. The benefits of these technologies will accrue to road users and roadway vehicles of all types: gas or electric; commercial, public, or private; etc.

Task 1 – Literature Review

The literature review focused on technologies related to determining and using absolute position to benefit transportation applications [Du et al., 2008; Tian et al., 2019; Williams et al., 2021].

Types of Positioning

Absolute Position is determined relative to the origin of a coordinate system that is attached to the Earth. Knowledge of absolute position is required, for example, for the planning of a route from one location to another where both the origin and destination locations (e.g., hotel to the airport) are known in the coordinate system. It is also needed for sharing of information between vehicles (i.e., V2V) or between vehicles and infrastructure (i.e., V2I). Absolute position may be determined by use of Global Navigation Satellite Systems (GNSS) [Hofmann et al., 2007; Teunissen et al., 2017]. It can also be determined by feature-based detectors (e.g., Lidar, camera, Radar) when the feature locations are able to be determined in an absolute sense (e.g., from a digital map) [Cadena et al., 2016; Lowry et al., 2015; Bresson et al., 2017].

Relative Position is the location of one object relative to another (e.g., the vehicle), even when one or both of their absolute positions are unknown. Relative position of objects is important for tasks such as parallel parking, lane keeping, and collision avoidance. Relative positions can be determined directly using feature-based sensors (e.g., Lidar, camera, Radar, ultrasound) mounted on the vehicle [Cao et al., 2007; Milford et al., 2012].

These two types of positioning are complementary. Both have utility for transportation. Both can be achieved using data fusion methods from a multiplicity of sensors [Vu et al., 2012; Nedevschi et al., 2013; Rose et al., 2014; Rohani et al., 2016].

Advances in GNSS toward achieving Lane-level Accuracy

Multiple GNSS (e.g., GPS, GLONASS, Galileo, BeiDou) are now available, with each individually supplying 6-10 measurements per epoch, yielding a total of 24-40 single-frequency measurements, or 50-100 multi-frequency measurements [Verhagen et al., 2010, Camacho, 2017]. The increased signal availability enhances the speed and reliability of GNSS position determination.

The advent of Precise Point Positioning (PPP) enables significant enhancements in GNSS accuracy, on the global scale, without the need for local base station infrastructure [Bree et al., 2012; Elsheikh et al., 2019; Elsobeiey et al., 2015; Gao et al., 2006; Knoop et al., 2017; Kouba et al., 2001; Liu et al., 2015; Teunissen et al., 2014]. PPP is capable of achieving submeter-level accuracy using single-frequency pseudorange measurements or centimeter to decimeter accuracy using multifrequency carrier phase measurements. PPP is implemented using State Space Representation (SSR) correction messages. At present, very few GNSS receivers accept SSR format messages. Recently, commercial entities have investigated the potential to communicate PPP information through Observation Space Representation (OSR) format messages, because almost all existing receivers accept OSR format correction measurements [Rahman et al., 2019]. An open-source prototype implementation of this approach has recently been demonstrated [Hu et al., 2021] and will be used in Task 5 of this project.

The main challenges to GNSS positioning for transportation are dense urban areas characterized by many tall buildings, called urban canyons [Zhu et al., 2018], that may block satellite signals or reflect satellite signals causing either multipath or non-line of sight measurement errors. Urban areas that have these characteristics represent only a small percentage of roadway miles (The U.S. Department of Transportation classifies 3.9 million miles of roadway as rural and 1.2 million miles of roadway as urban [US DOE, 2015]. Not all urban roadways are in urban canyons.). In such urban areas, these issues can best be accommodated by incorporating alternative sensors [Vu et al., 2012; Wang et al., 2012; Groves et al., 2013; Nedevschi et al., 2013; Rose et al., 2014; Shunsuke et al., 2015; Rohani et al., 2016; Xu et al., 2018] and digital maps [Groves, 2011; Gu et al., 2015] to provide additional information and constraints to the position determination process.

CAV Applications that Benefit from Lane-level Accuracy

Vehicle-centric applications benefit significantly from real-time lane-level vehicle position accuracy [Tian et al., 2019; Williams et al.; 2021]. These applications are primarily driven by on-board sensors and/or communication technologies, intended to assess the ego-vehicle and/or the surrounding environment to adjust the ego vehicle’s operations. For instance, lane-level positioning accuracy is the minimum requirement for multiple CAVs to enable Cooperative Adaptive Cruise Control (CACC) to form a platoon or to perform Cooperative Ramp Merging (CRM) to determine the most efficient merging sequence [Wang et al., 2019]. Specific applications mentioned in [Toledo-Moreo et al., 2009] include *Enhanced Driver Awareness* and *Intelligent Speed Alert* that warn the driver of upcoming hazard along their trajectory. In addition, for many eco-ITS applications [Barth et al., 2012], such as eco-routing

[Boriboonsomsin et al., 2012; Zhou et al., 2016] and Eco-Approach and Departure (EAD) [Hao et al., 2019; Ye et al., 2019], lane-level positioning accuracy would help significantly improve their effectiveness, by better estimating the driving lane, road-grade (thus the energy consumption), and the lane-level traffic information (e.g., signal phase and timing, queue length).

Estimation of travel time has been stated as an FHWA priority due to its utility for evaluating the performance of traffic networks and is one of the most understood measures for road users due to its utility for optimizing route planning [Liu et al., 2009a; Lee et al., 2019]. Travel time estimation has been investigated both on freeways [Chen et al., 2001; Chen, 2002] and arterials [Liu et al., 2009a]. The use of probe vehicles is investigated in, for example, [Turner et al., 1998; Dailey et al., 2002; Liu et al., 2009a]. As any driver realizes, each lane can have different traffic conditions, thus different travel times [Lu et al., 2021]; therefore, this application is one that can benefit from the higher resolution enabled by probe vehicle (e.g., CAV) lane determination and lane-level positioning technologies.

Due to its criticality to the assessment and management of intersection traffic, vehicular queue length estimation has been investigated by many papers [Webster, 1958; Webster et al., 1966; Mirchandani et al., 2007; Skabardonis et al., 2008]. Considering the potential to achieve enhanced performance through lane-level position accuracy and more reliable lane-determination, lane-level queue prediction is the focus of Task 6 and is reviewed separately in a later subsection.

Road and Lane Determination

Road and lane perception are important for various CAV applications [Hillel et al., 2014]. The term *lane perception* has various possible meanings as well as contexts: On which lane of a road is the vehicle traveling? On which lane of which road is the vehicle traveling? Where is the vehicle within a given lane? The accuracy, reliability, and challenges of answering each question depends on the sensor suite that is available aboard the vehicle.

Road and Lane map matching have interesting similarities and differences.

- The goal of Road Level Map Matching (RLMM) is to determine the identity of the road on which the vehicle is traveling [Dmitriev, et al., 1999; Quddus et al., 2007]. In addition, the longitudinal position of the vehicle along the road (e.g., mile marker) may be of interest for route planning purposes. RLMM requires knowledge of absolute position. It can work with position measurements with accuracy at the 10 m level, which is possible by open service GNSS methods (i.e., non-differential). It is useful for non-safety applications such as vehicle routing, wherein it is reasonable to assume that the vehicle is on the road and traveling in the correct direction. Therefore, it is common to project positions that are off-road or on road segments that are traveling in the wrong direction back to the nearest road segment traveling in the correct direction [Scott, 1994; Fouque, 2008].
- The goal of Lane Level Map Matching (LLMM) is to determine the specific lane that the vehicle is occupying at the present time. Sometimes, going beyond lane identification, it

> is also of interest to determine the lane relative position of the vehicle. The longitudinal position along the lane is determined relative to a specific reference location (e.g., lane start point or intersection stop bar). The lateral position is defined as the shortest distance from the vehicle location to the lane centerline. LLMM requires absolute position accuracy that is a fraction of a lane width (i.e., meter level). This accuracy can be achieved by differential GNSS methods or by fusing information from different sensor modalities. Because LLMM is intended for critical safety applications, it is important that vehicle positions that may be off-road (e.g., accident) or traveling in the wrong direction (e.g., passing) should not be projected back to a lane traveling in the correct direction. To do so would cause crucial failures of applications such as Wrong-Way Driving Alert System. LLMM requires a digital map including centerlines for each lane accurate to the decimeter level.

Lane relative lateral position determination is commonly performed using vision [Dickmanns et al., 1987; Turk et al., 1988; DeSouza et al., 2002; Bonin-Font at al., 2008; Lundquist et al., 2011]; however, lane determination, especially relative to a map, is much more challenging using vision alone. LIDAR based road and lane detection methods are presented in [Ogawa et al., 2006; Kammal et al., 2008; Caltagirone et al., 2017]. Approaches combining GPS and vision are presented in [Rabe et al., 2016; Hansson et al., 2021].

Map matching methods fall into a few categories [Quddus et al., 2007; Du et al., 2008]: point-to-point, point-to-curve, and curve-to-curve. A map aided localization is introduced in [Toledo-Moreo et al., 2009], where the particle filter is used to consider navigation along each lane as discrete hypotheses. In this approach, the mapped lanes each act as a priori information for Bayesian estimation. Several other approaches also use the map to constrain the position estimate, e.g., [Scott, 1994; Fouque, 2008; Peyret et al., 2008]. Depending on the implementation, such approaches may project a vehicle that is off-road or in a passing lane back to a mapped lane traveling in the same direction of the vehicle, which can create safety issues. In addition, Hidden Markov Methods (HMM) have been used to combine information through time [Atia et al., 2017]. That approach relies on a priori assumed probabilities of lane changes. Enhanced performance might be achieved by computing these probabilities in real-time based on the vehicle state.

Lane-Level Queue Length Estimation and Prediction

The traditional means of gathering information about intersection approach queues is to embed loop detectors in the roadway [Muck, 2002; Liu et al., 2009b; Lee et al., 2015; Li et al., 2018]. Each loop detectors counts the number of vehicles passing a specific area along an intersection approach. The lane queue upstream of the furthest loop from the stop-bar cannot be directly detected, but might be inferred by combining with information from upstream and downstream intersections in the network. Multiple loop detectors along multiple lanes can be used to increase the accuracy of the queue estimate; however, each added loop detector increases both the installation and maintenance costs.

Alternatively, CAVs with high positioning accuracy can serve as reliable sensors, called probes, to enable dynamic lane-level queue inventories, with the goal of enabling much more effective traffic management. Therefore, there is interest in the use of CAV's as probes for intersection queue length estimation [Cheng et al., 2011; Zhao et al., 2019a; Zhao et al., 2019b; Comert et al., 2021]. The algorithms of these articles depend on estimates of the CAV penetration rate. Several articles present methods to estimate that parameter [Wong et al., 2019]. None of these articles study the accuracy as a function of GNSS position error. Those articles that study the problem in VISSIM assume that the GNSS error is zero, because the standard VISSIM does not include a GNSS position error model. This project develops a VISSIM GNSS error module as part of Task 6.

The intersection queue length estimation problem can be considered from multiple perspectives. Several articles consider the problem of maximum queue length estimation after all the vehicles in the cycle have passed the intersection [Liu et al., 2009; Cheng et al., 2011; Hao et al., 2013; Hao et al., 2014]. An alternative perspective that is of interest herein is to predict the instantaneous lane-level queue length during the cycle while some vehicles are still approaching the intersection (see Task 6 and Appendix C).

Application Selection

Tasks 5 and 6 will focus on intersection lane-level queue estimation and its underlying technology of vehicle lane identification.

For example, to enable Signal Phase and Timing (SPaT) to be scheduled based on real-time lane-level queues, each intersection would require (1) a lane-level map for approach lanes (see Task 2), algorithms to determine which lane corresponds to the measured vehicle position (see Task 3), and algorithms to use (probe) vehicle longitudinal position to determine the lane-level queue length (see Appendix C). Task 5 demonstrates the methods of Task 3 using the intersection maps developed in Task 2 for the Riverside Innovation Corridor. Task 6 uses VISSIM to study the accuracy of lane-level queue prediction as a function of vehicle position accuracy and vehicle penetration rate.

Task 2 – Lane-Level Mapping

High-definition (Hi-Def) digital maps are an indispensable automated driving technology that is developing rapidly [Liu et al., 2020]. The experimental demonstration in Task 5 and simulation evaluation of Task 6 will each require a lane-level map. Many other pilot projects also require lane-level maps.

Before creating that lane-level map, the team assessed the available tools for its creation. There are various commercial (e.g., HERE) or governmental map products in the market. Companies such as Amazon and Google that are interested in autonomous driving develop their own proprietary Hi-Def mapping systems only useable by entities with which they have contractual agreements. It is notable that, to the best of our knowledge, the U.S. Department of Transportation (USDOT) map tool (https://webapp2.connectedvcs.com/) is the only tool available with free access that allows the user to create MAP and Signal Phase and Timing (SPaT) messages. Tasks 5 and 6 of this project used a J2375 Map message of the demonstration area (i.e., Riverside Innovation Corridor) created by the USDOT map tool for their lane-level map. This map message was created under a previous project, but the roadway has not changed, so it did not require revision.

At the time of this study, a comprehensive analysis of the accuracy of this map tool was lacking in the literature. Therefore, the team acquired data to perform an assessment of the accuracy expected from the USDOT mapping tool.

This project completed the USDOT map tool accuracy assessment using the intersection of Iowa and University Avenue in Riverside, CA as a test site. The analysis used 39 feature points within about 200 meters of the verified point for that intersection and 55 feature points over longer distances from the verified point. All feature locations were mapped using the USDOT tool and compared with their location as surveyed using GNSS Real-Time Kinematic (RTK) methods [Hatch, 1983; Talbot, 1993; Hofmann-Wellenhof et al., 1997; Neumann et al., 1997]. Different error sources were evaluated to allow assessment of the USDOT map accuracy. In this investigation, the USDOT map tool was demonstrated to achieve 17-centimeter accuracy in the horizontal directions, which meets the lane-level map accuracy requirement.

The USDOT Mapping Tool loses the accuracy while converting to the J2735 output. The internal accuracy is better than 0.2 m. The conversion process that converts from geodetic decimal, to UPER HEX, to ASN.1 (J2735) can lose up to 1.0e-5 degrees of geodetic accuracy. For an Earth radius of 6.378e6 m, each 1e-5 error in geodetic degrees corresponds to 1.11 meters of position error. This conversion loses of accuracy is unfortunate, as the tools internal accuracy is much better. For the results that follow, we avoided this conversion loss of accuracy by copying the internal map results directly, so as to avoid the output conversion process.

The report describing the methodology and results of this accuracy assessment is included as Appendix A. The report has also been submitted for publication.

The location data to support the USDOT Mapping Tool accuracy assessment is included as one of the data products of this project.

Task 3 – Lane-Level Map Matching

The goal of this task is to implement and analyze algorithms to combine real-time, meter-level position measurements with lane-level maps (obtained from Task 2) to achieve lane determination and lane relative position. The algorithms developed in this task will be used in Tasks 5 and 6.

Let $\hat{\boldsymbol{p}}(t)$ denote the measurement of the actual vehicle location $\boldsymbol{p}(t)$. The position measurement is assumed to have lane-level accuracy (about 1 meter). The digital map is discussed under Task 2.

The road and lane determination literature is reviewed under Task 1. In this task, given that the position is assumed to be accurate to lane-level, we assume that the road determination problem is solved. When the vehicle is near an intersection with position known to the meter level, road determination is straightforward, except when the vehicle is actually in the intersection of the crossing streets. When the vehicle is not near an intersection, then lane-level position accuracy makes road identification straightforward. Therefore, the following subsections present and analyze the point-to-curve lane determination approach.

Point-to-Curve Lane Determination

Each lane in the digital map is described as a sequence of nodes or waypoints along the lane centerline. The centerline is defined to be the line connecting two adjacent nodes from the lane node list. In Figure 1, for a road with two lanes, separated by the wide yellow dashed line, the centerline waypoints for each lane are indicated by the orange circles. The lane centerline is indicated by the white line, for each lane, for the lane segment closest to the vehicle location (green dot). The vector from the two closest waypoints to vehicle location are shown as red arrows.

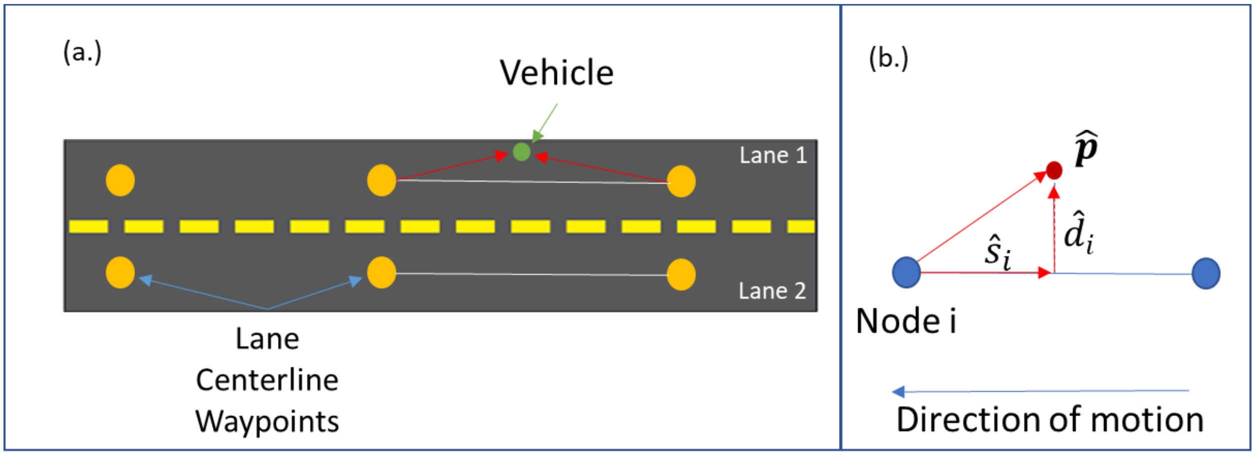


Figure 1. Depiction of lane determination for a vehicle operating on roadway with two lanes. (a) Left – Lane-level map. (b) Right – Lane relative position.

The lane relative coordinates are defined as $(\hat{s}(t),\ \hat{d}_i(t))$, where $\hat{s}(t)$ is the distance along the lane centerline from the stop bar for the lane to the projection of $\hat{\boldsymbol{p}}(t)$ onto the lane centerline. The variable $\hat{s}(t)$ is computed as the sum of the known distance of Node i from the stop-bar along the centerline plus $\hat{s}_i$. Based on the measurement $\hat{\boldsymbol{p}}(t)$ and the digital lane map, the signed distance $\hat{d}_i(t)$ is computed from the closest point on the lane centerline to $\hat{\boldsymbol{p}}(t)$. The measured vehicle position $\hat{\boldsymbol{p}}(t)$ is associated with the nearest lane that has $|\ \hat{d}_i(t)\ |\ \leq\ L_w/2$, where L_w is the lane width at $\hat{\boldsymbol{p}}(t)$. Once the lane has been determined, the lane relative coordinates are known.

Theoretical Assessment of Lane Determination Error

Figure 2 displays an image useful for analyzing the probability of correct lane determination. The lane stop bar is indicated by the solid black vertical line on the left side. The lane centerlines from the digital map are indicated by the narrow gray solid lines. The lane edges, which are computed from the lane width L_w and the centerlines, are drawn as wide dashed yellow lines.

There is uncertainty associated with the measurement $\hat{\boldsymbol{p}}(t)$. Assuming that the position error $(\hat{\boldsymbol{p}}(t) - \boldsymbol{p}(t))$ has a Gaussian distribution, this uncertainty is depicted by the blue ellipses in Figure 2, which represent contours of equal probability. The marginal probability density for $d_i(t)$ is

$$p(d_i) = (2\pi\sigma^2)^{-0.5}\ exp\left(-\left(d_i - \hat{d}_\iota\right)^2 \Big/ {}_{2\sigma^2}\right).$$

Assuming that the vehicle is in lane 1, this density is sketched along the right side of the lane in Figure 2. The integral over the blue cross-hatched region yields P_c the probability of a correct lane determination:

$$P_c = \int_{\underline{\tau}}^{\overline{\tau}} p(\tau)\, d\tau, \text{ where } \overline{\tau} = \frac{1}{\sigma}\left(\left({}^{L_w}/_{2}\right) - \hat{d}_\iota\right) \text{ is the upper limit, } \underline{\tau} = \frac{1}{\sigma}\left(-\left({}^{L_w}/_{2}\right) - \hat{d}_\iota\right)$$

is the lower limit, and τ is a dummy variable of integration. This is easily computed as $P_c = erf(\overline{\tau}) - \mathrm{erf}\left(\underline{\tau}\right)$, where $erf(x)$ is the error function for the standard Gaussian distribution. The probability of a lane determination error is $P_e = (1 - P_c)$. The probability of lane determination error is graphed as a function of the actual distance of the vehicle from the centerline for various values of σ^2. The graph clearly shows that as the position estimation accuracy increases, the probability of a lane determination error decreases for all distances from the lane centerline. Note that for vehicles driving along the lane boundary (i.e., $d_i = L_w/2$), the probability of lane determination error is always larger that 0.5. This is expected, as the vehicle could be in either lane.

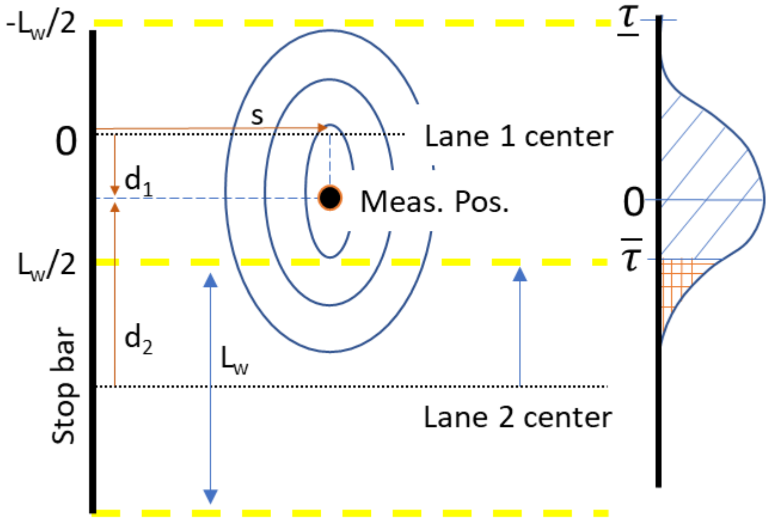


Figure 2. Depiction of method and variables used for point-to-curve lane determination with probability of correct lane determination.

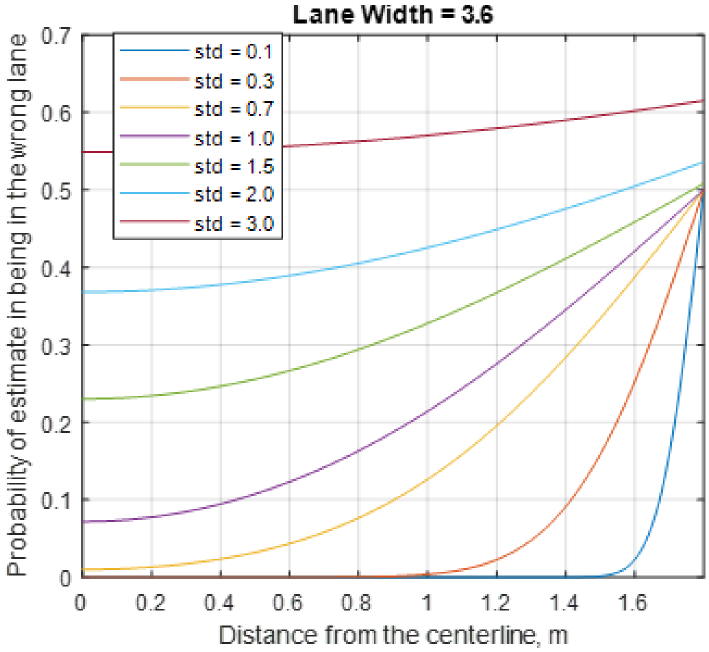


Figure 3. Probability of lane determination error as a function of the distance from the lane centerline drawn for various values of σ^2.

Task 4 – Progress Report

The six-month project report was submitted in October 2021. It covered the period from April 1, 2021 through October 1, 2021.

Task 5 – Demonstration

Lane-level map matching underlies various CAV applications. This task demonstrated and evaluated lane-level map matching in the context of those CAV applications relevant to intersections and intersection signal management. The task was implemented on the Riverside Innovation Corridor. The task included: (1) developing lane-level maps for all lanes of three intersections; (2) implementing an on-vehicle software system for lane determination and lane-relative positioning using a low-cost GNSS receiver; and, (3) demonstrating lane determination and lane-level positioning using real-time experimental data.

Intersection Maps for the Riverside Innovation Corridor

The task was implemented using the USDOT Map Tools (see Task 2) to develop lane-level J2735 maps for three intersections on the Riverside Innovation Corridor (University Avenue intersections with Iowa, Cranford and Chicago Avenues) [Oswald et al., 2021].

Figure 4 shows the lane maps for the three intersections. Each lane center is plotted as green for ingress and red for egress. The single blue lane in each image shows the lane that was determined for the vehicle position marked by the blue star. One of these images is shown by the real-time GUI during the experiment, when the vehicle is within that intersections map region.

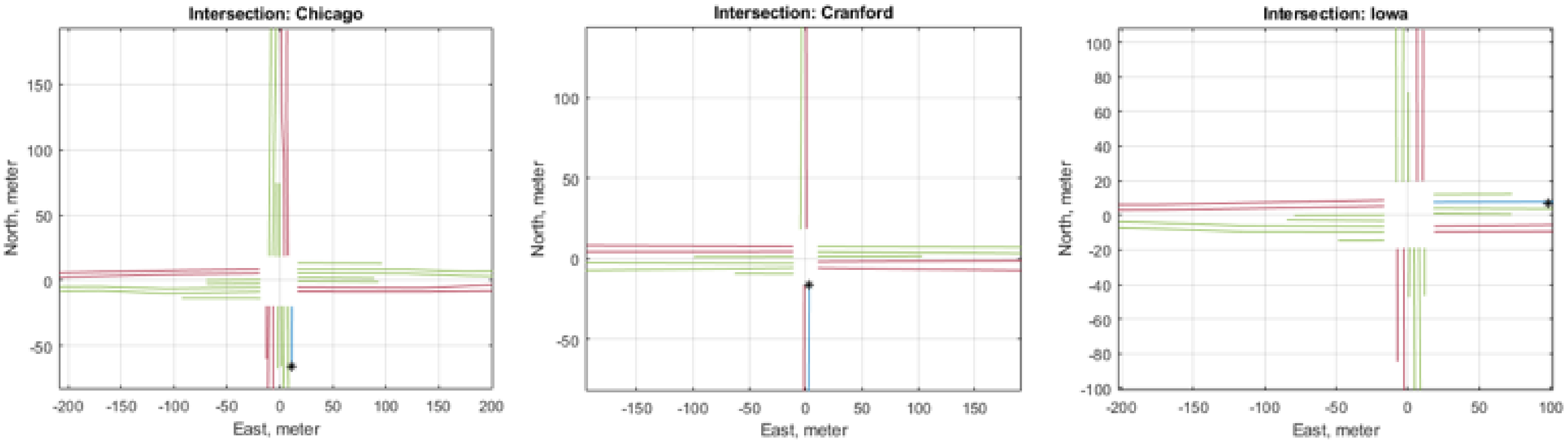


Figure 4. Lane maps for the UC Riverside Innovation Corridor.

The lane-level map for the three intersections along the UCR Innovation Corridor is included as one of the data products of this project.

On-Vehicle Demonstration Software

Figure 5 shows the hardware and software structure of the lane determination system used in the experiments of this task.

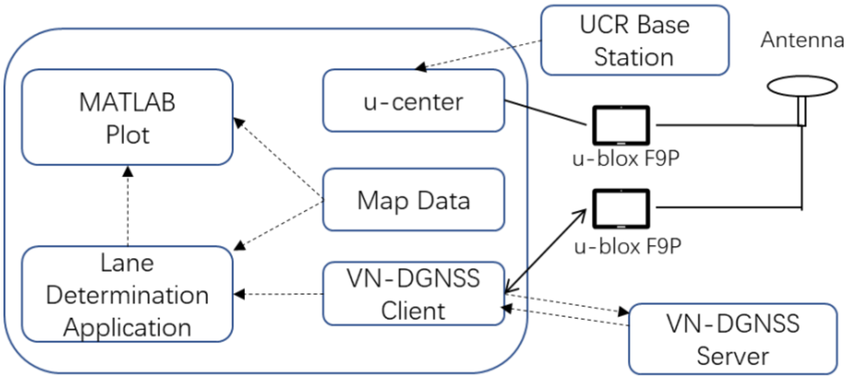


Figure 5. The structure of lane determination experiment system.

A single u-blox dual-band antenna was connected to two u-blox F9P receivers through a signal splitter:

1. One F9P u-blox receiver was connected through a VN-DGNSS client to receive RTCM OSR corrections from a VN-DGNSS server [Hu et al., 2021]. This VN-GNSS approach is expected to achieve submeter positioning accuracy. The VN-GNSS approach does not require any nearby differential GNSS base station, so it is appropriate as an on-vehicle commercial positioning system on continental or local scales.
2. Another F9P u-blox receiver was operating in Real-time kinematic (RTK) mode to produce *ground truth positioning data* by using the receiver manufacturer's u-center software to connect to a UCR base station. RTK GNSS requires a local base station, so it is not usable as an on-vehicle commercial positioning system on continental or global scales. RTK GNSS processing achieves centimeter accuracy [Hatch 1983; Farrell et al., 2000] position which makes it appropriate for use as a ground truth reference.

The VN-DGNSS client communicates GNSS correction information from the VN-DGNSS server and delivers it to the F9P u-blox receiver, which uses it to perform differential GNSS to achieve positioning accuracy at the meter level. The VN-DGNSS client collects the position data from F9P u-blox receiver and sends it to the Lane Determination Application (LD-APP). The LD-APP implements a Point-to-Curve Lane Determination algorithm (see Task 3). The results were plotted as a Matlab figure in real-time (see Figure 4).

Demonstration Results

This section presents the lane determination experimental results. Two experiments were performed. In each experiment, the vehicle was driven along the Riverside Innovation Corridor.

During the driving, the vehicle was in various lanes and at various distances from its lane centerline.

- The experiment that took place on 11/10/2021 lasted for about 42 minutes, providing 1486 data points. The receiver was configured to use GPS.
- The experiment that took place on 3/12/2022 lasted for about 30 minutes, providing 985 data points. The receiver was configured to use GPS and Galileo.
- The experiment that took place on 3/30/2022 lasted for about 30 minutes, providing 1115 data points. This receiver was configured to use GPS and Galileo.

In each experiment, at each measurement time, the lane ID and lane relative location was determined both for the VN-DGNSS receiver and for the RTK receiver. The answer from the RTK receiver was treated as ground truth. The answer from the VN-DGNSS receiver was compared to ground truth to compute the probability of lane determination error as a function of the distance from the lane centerline.

Figure 6 displays the results of the experiments. Each experiment is in a separate row. In each figure in the left column, the black curve shows the probability of lane determination error from one experiment overlaid on the theoretic results discussed relative to Figure 3. The graph was created as follows: (1) for the current location, compute the lane width (Lw), distance from the center line (d), and percentage lane width from the centerline (q = 2 d/Lw); (2) compute the number of samples and the number of lane determination errors in each of five bins, for q in: 0-0.1, 0.1-0.2, 0.2-0.3, 0.3-0.4, and 0.4-0.5. Each point on the graph also shows error bars, where the magnitude of the error bar is computed using $\sqrt{\frac{P_c\,P_e}{N}}$ where (for a Bernoulli random variable) $P_e = \left(1 - {}^{n}/_{N}\right)$ is the probability of error, $P_c = 1 - \ P_e$ is the probability of a correct answer, n is the number of wrong answers in the bin, and N is the total number of samples (right or wrong) used to estimate P_e for that bin. The bins nearer to the lane center have more points because the vehicle cannot be safely driven near the lane edge for long periods of time.

The right column shows the cumulative probability of the horizontal position error during each experiment, where the position error is the norm of the difference between the VN-DGNSS and RTK positions.

From the graphs in the left column, we can conclude that for vehicles driven within 25% of the 3.6 m lane width (i.e., within 0.9 meters) of the lane centerline, the correct lane is determined for over 90% of the samples.

During the processing of these experimental results, we found that lane width is not constant and not equal to the left and right. This is because the digital lane centerlines of adjacent lanes are not always parallel and the lane center nodes may not be perfectly selected in the map tool. This causes the centerlines of the lanes to the left and right to move toward or away from each other at different rates. The unequal left and right distances to the lane boundaries must be accounted for correctly when determining the lane ID decisions, especially when the vehicle operates near the lane boundary.

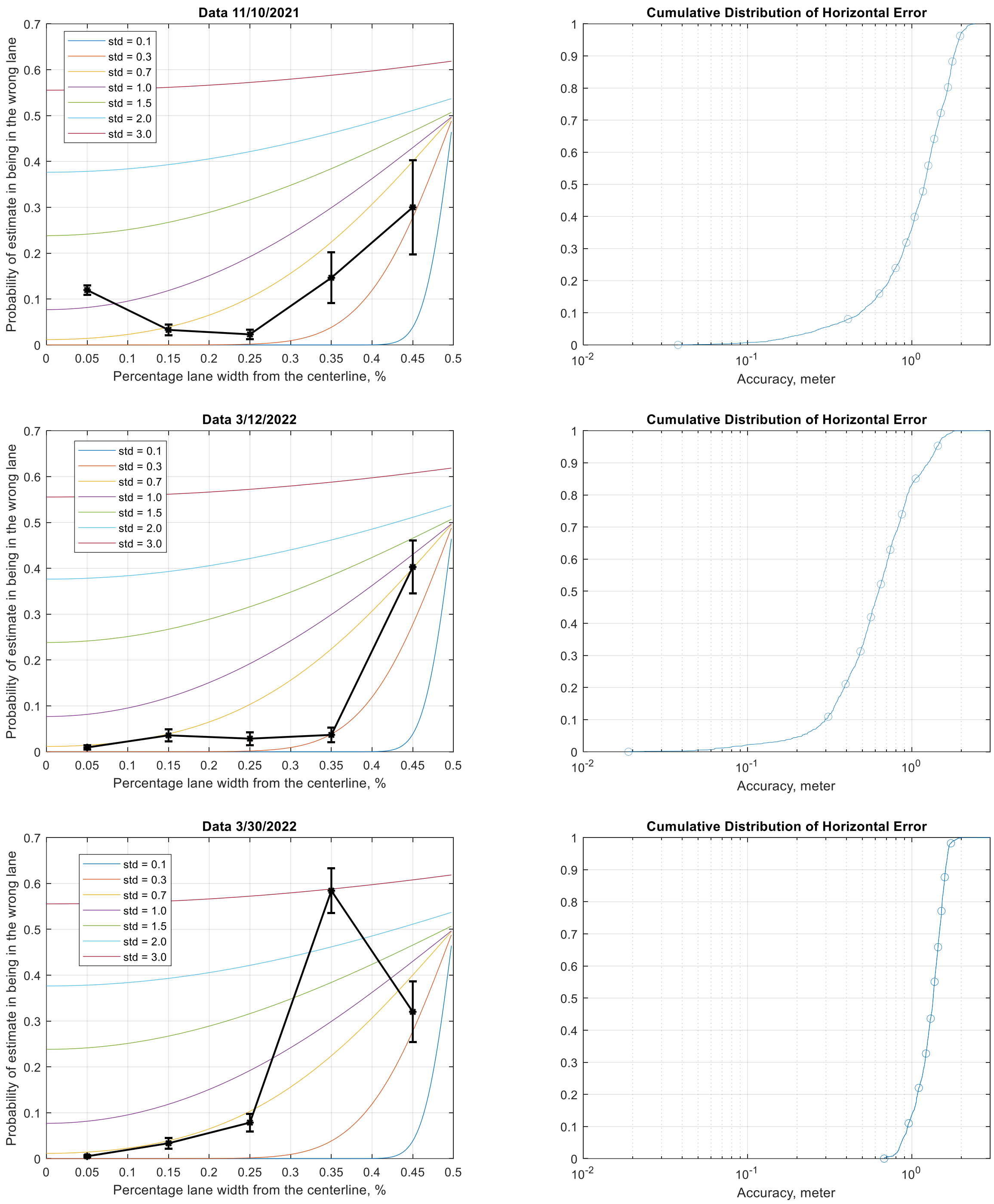


Figure 6. Lane detection accuracy from three experiments. Each row shows data from one experiment. The left column compares the wrong lane probability as a function of the distance from the lane centerline. The right column shows the cumulative distribution function for the position error during each experiment.

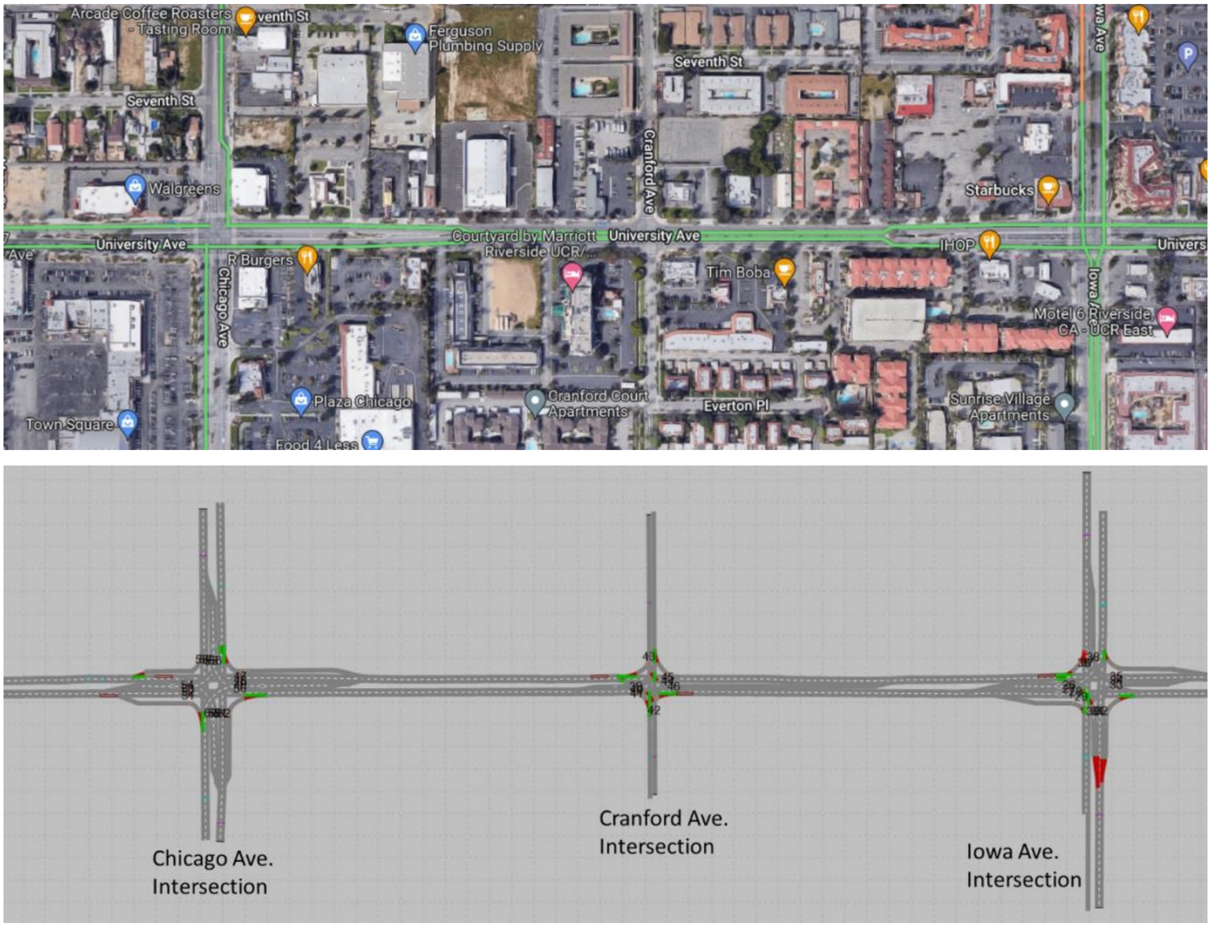


Figure 7. Portion of the Innovation Corridor in Riverside, CA that is studied in the simulation and demonstration. (a) Top – Google map view. (b) Bottom – VISSIM View.

Task 6 – Simulation Study

This simulation study focuses on the accuracy of lane-level intersection queue estimation as a function of two variables: probe vehicle penetration density and GNSS position accuracy.

Simulation Network

The simulations are done using the PTV VISSIM [PTV Group, 2022] microscopic traffic simulation software. The network in VISSIM is modelled after the Innovation Corridor in Riverside, California [Oswald et al., 2021]. The simulated network is about 1.3 km of the *Riverside Innovation Corridor* and includes 3 intersections along University Ave.: Chicago Ave., Cranford Ave., and Iowa Ave. The majority of this corridor consists of 2 lanes in the Westbound direction and 2 lanes in the Eastbound direction, and a speed limit of 50 km/h. Fixed-time signal control is coded for all three intersections. The cycle lengths of the Chicago, Cranford and Iowa avenue intersections are 96, 78 and 96 seconds, respectively. Green and yellow intervals of all three intersections' East-West through phase are 30 and 5 seconds. Traffic demands have been calibrated based on the turning movement count survey on June 2nd, 2016 by the City. Figure 7 shows the study corridor in both Google Map and VISSIM.

Simulation Scenario

The simulation time is 1 hour, representing a typical morning peak hour (8 am – 9 am) traffic conditions along the corridor. There are 30 different scenarios simulated with each being run 10 times. These scenarios cover 5 different penetration rates of 20%, 40%, 60%, 80% and 100%; and, 6 different GNSS error levels with position error standard deviations of 0, 0.5, 1, 1.5, 2, and 2.5 m.

Key Modules

At signalized intersections, lane-level queue length is a critical traffic state that is monitored for efficient traffic signal control. To evaluate the impact of GNSS errors and probe vehicles density on real-time lane-level queue length estimation accuracy, we develop three key modules: *GNSS error model*, *lane-level map-matching*, and *lane-level queue length estimation.* All three are implemented in the simulation environment via application programming interfaces (APIs).

PTV VISSIM allows for an external dynamic link library (DLL) to interface with its vehicle models. The DLL is written in C/C++, and is called for each vehicle at each time step of the simulation run. VISSIM passes the current state of the target vehicle to the DLL. The standard VISSIM computation engine provides the perfect location (i.e., the ground truth location) of the target vehicle. The simulation structure is shown in Figure 8. Using the API, the GNSS error model outputs a measured position by perturbing the ground truth position with a time-correlated random error process computed based on a user-specified position error standard deviation. The measured position is then fed into the lane-level map-matching algorithm to determine which lanes on which the target vehicles are traveling.

Based on the lane indices and other state information (e.g., velocity), the queue length along each approaching lane of each intersection can be estimated online and compared with ground truth to determine the queue estimation error. The statistics of lane index errors and lane-level queue length errors are output and processed to evaluate the system performance.

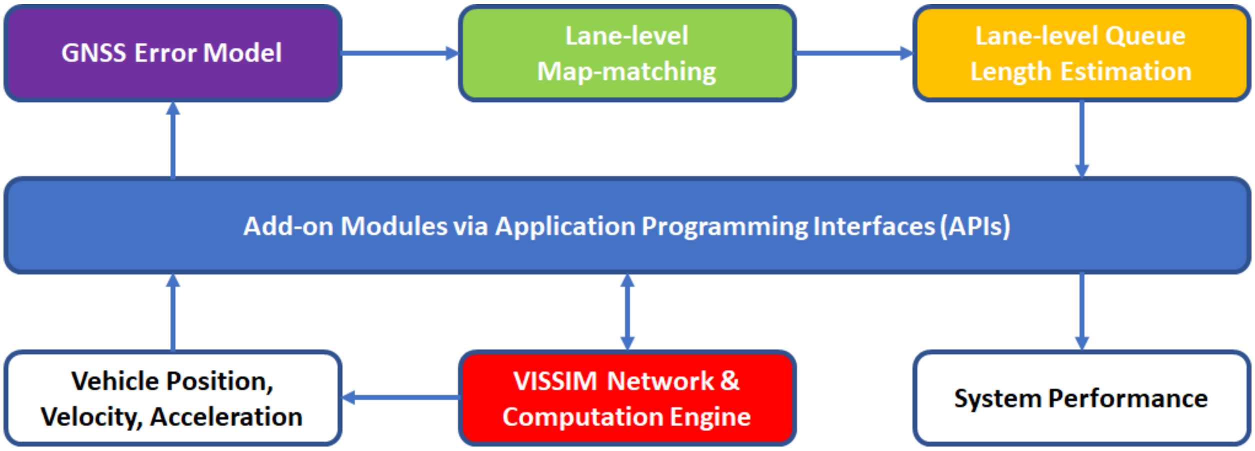

GNSS Error Model
Lane-level Map-matching
Lane-level Queue Length Estimation
Add-on Modules via Application Programming Interfaces (APIs)
Vehicle Position, Velocity, Acceleration
VISSIM Network & Computation Engine
System Performance

Figure 8. Key modules and flow diagram for the VISSIM lane-level queue length estimation.

GNSS Error Module

The standard PTV VISSIM implementation supplies user algorithms with the ground truth vehicle location at each time step. To study the effects of GNSS position estimation error on lane queue estimation accuracy, this project required a module that would additively corrupt the ground truth location with measurement error to produce a measured position.

This project designed, implemented, and used a GNSS error module. The theory underlying the implementation is described in Appendix B. That appendix also points to the Dryad URL for the software that implements the approach.

Lane-level Map-matching Module

Before incorporating the lane-level queue length estimation algorithm, a lane-level map-matching algorithm is required to determine the lane index (i.e., lane ID) of each probe vehicle at each simulation time step.

This lane determination algorithm used herein is a point-to-curve algorithm discussed under Task 3. Note that the default driver behavior in VISSIM moves vehicles along the centerline of its lane. The GNSS measurement of the vehicle position will not be on the lane center.

The lane determination accuracy A will be quantified as

$$P_c = \frac{S_C}{S_T}, A = 100 * P_c \text{ , and } P_e = 1 - P_c \tag{6-1}$$

where S_c is the number of correct samples, S_T is the total number of samples, P_c is the probability of a correct determination, A is the percentage of determinations, and $P_e$is the probability of an incorrect determination.

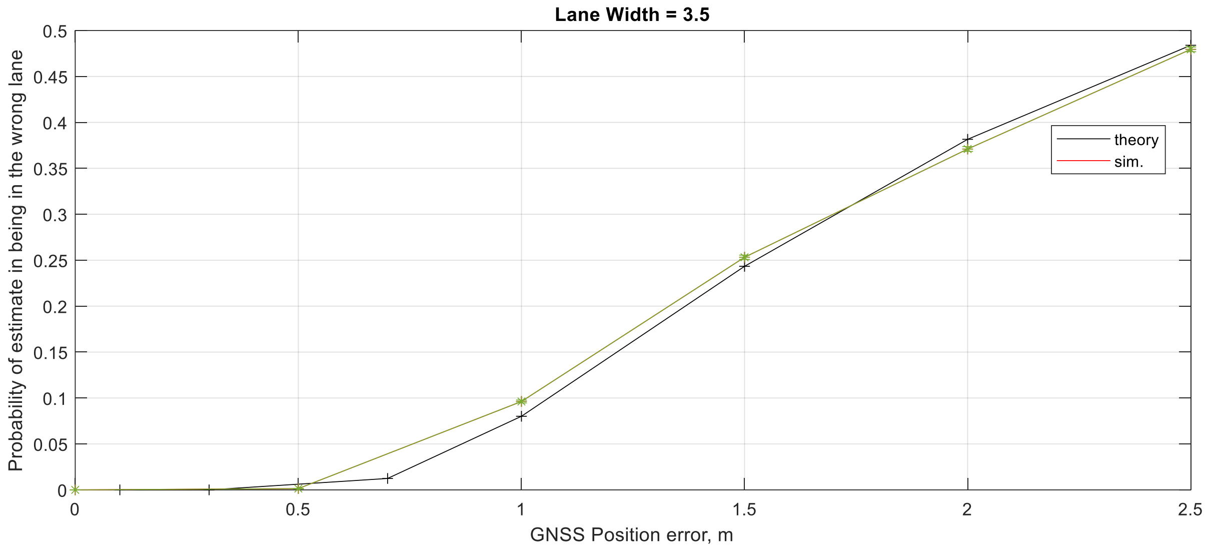


Figure 9. Lane determination accuracy as a function of position error standard deviation.

Figure 9 shows curves comparing the theoretical and simulated lane determination accuracy as a function of the GNSS position error. The GNSS position error is the measurement standard deviation parameter that is input to the GNSS error module. The lane width in the VISSIM network is 3.5 meters.

Figure 9 also shows error bars representing the standard deviation of the lane determination error at each GNSS error. Treating each decision as a Bernoulli random variable the standard deviation of the estimate either P_c or P_e is computed as $\sqrt{\frac{P_c\,P_e}{S_T}}$. The standard deviation computation is repeated for each of the six points on the graph using the values of P_c and S_T relevant for that data point.

Figure 10 presents the grouped histograms of the magnitude of the lane determination error for different GNSS errors. In all cases, the lane determination error is less than 1 lane.

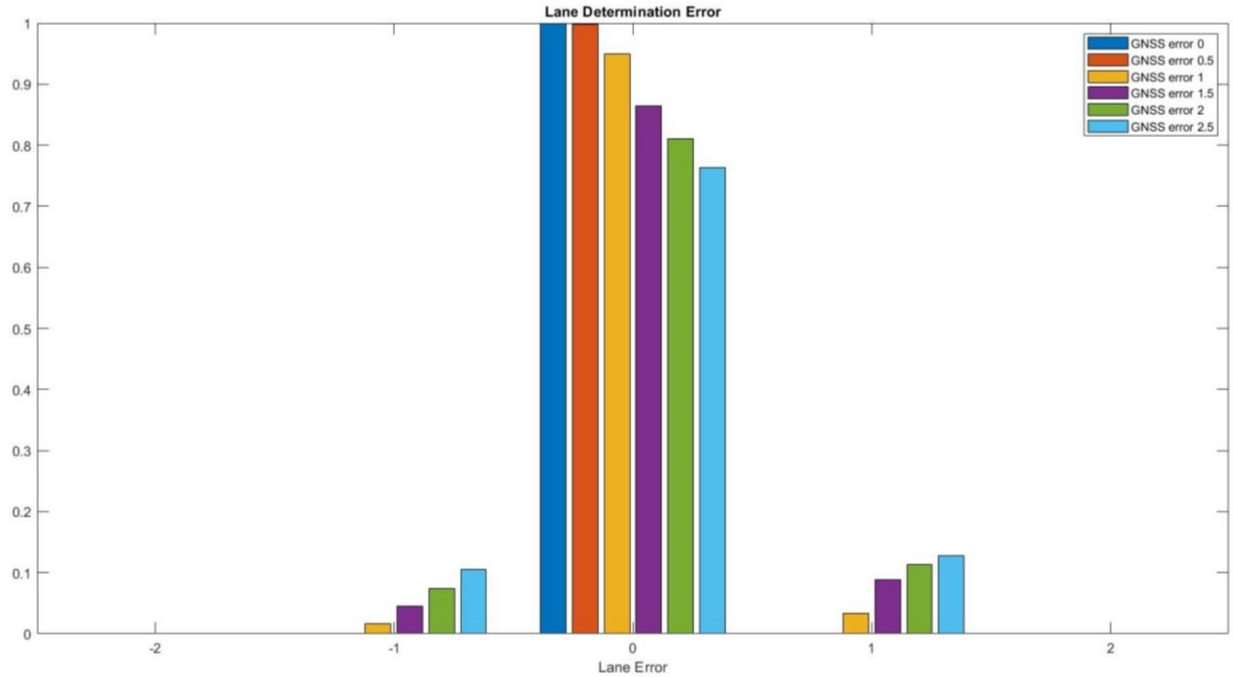


Figure 10. Histograms of lane determination errors across different levels of GNSS errors.

Lane-level Queue Length Estimation Module

The lane-level queue length estimation method is presented in the Appendix C. Within the VISSIM evaluation code, the values of the algorithm parameters were: Ld = 100 (meters), Vf = 13.89 (meters/second); λ_0= 0.05555 for Westbound traffic (average number of arriving vehicles/ second), λ_0= 0.076389 for Eastbound traffic (average number of arriving vehicles/ second), h = 0.4 (seconds/vehicle), and k = 0.1429 (vehicles/ meter).

Lane-level Queue Length Estimation Simulation Results

This section assesses the impact of GNSS position error and CAV penetration rate on the accuracy of queue length estimation.

The queue length estimation error is defined as

$$E = Q_a - Q_e, \tag{6-2}$$

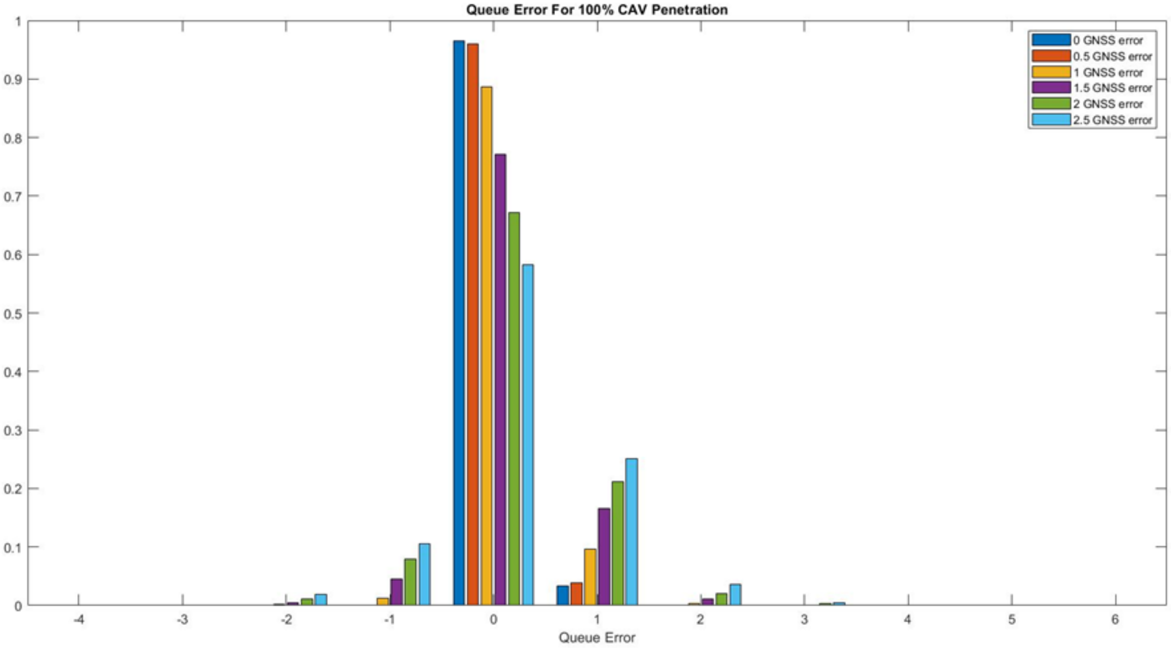
where Q_a represents the actual queue length computed by VISSIM using ground truth and Q_e represents the queue length estimated by the intersection controller using the measured position data from the CAV's.

Figure 11 (a. – e.) show the normalized grouped histograms of lane-level queue length estimation errors for each penetration rate as a function of the GNSS error standard deviation in meters. In each figure, the GNSS error standard deviations take the values: 0 m, 0.5m, 1m, 1.5m, 2m, and 2.5m. Figure 11 (a.) shows the error statistics for a CAV penetration rate of 100%; Figure 11 (b.) shows the error statistics for a CAV penetration rate of 80% CAV; and so on for Figure 11 (c. – e.). The penetration rate for each figure is stated in the figure title. The horizontal axis in each figure represents the number of vehicles by which the estimated queue length differs from the actual queue length. For instance, "0" means that the queue estimation is correct; an error of "1" means that the queue estimation is under-estimated by 1 vehicle; an error of "-1" means that the queue estimation is over-estimated by 1.

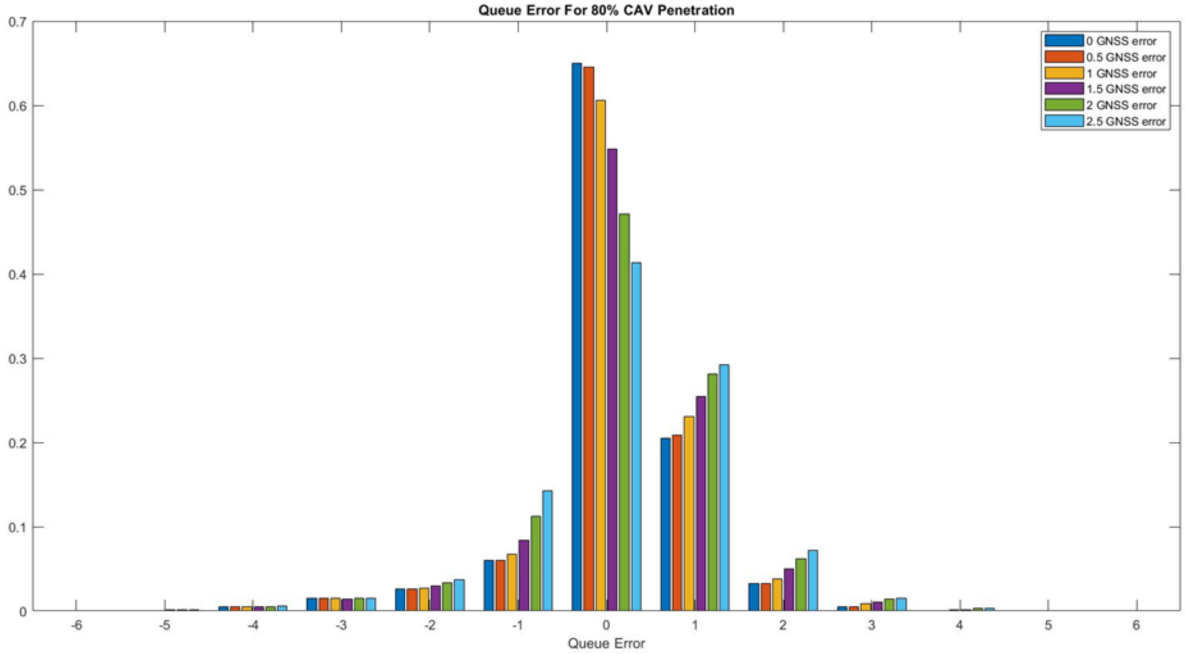
Figure 12 (a. – f.) show the normalized grouped histograms of lane-level queue length estimation error for each GNSS error level as a function of penetration rate. Figure 12 (a.) shows the queue estimation error statistics for a GNSS error of 0 for each CAV penetration rate, i.e., 20%, 40%, 60%, 80% and 100%, respectively. Figure 12 (b.) shows the queue estimation error statistics for a GNSS error of 0.5m for each penetration rate, etc.

Figure 11 and Figure 12 show that queue length estimation errors increase as the GNSS error increases and as the penetration rate decreases from 100%. This decline in performance is especially strong with respect to the decreasing penetration rate. This is because the algorithms performance is heavily dependent on the probability that a CAV is sufficiently near the end of the queue.

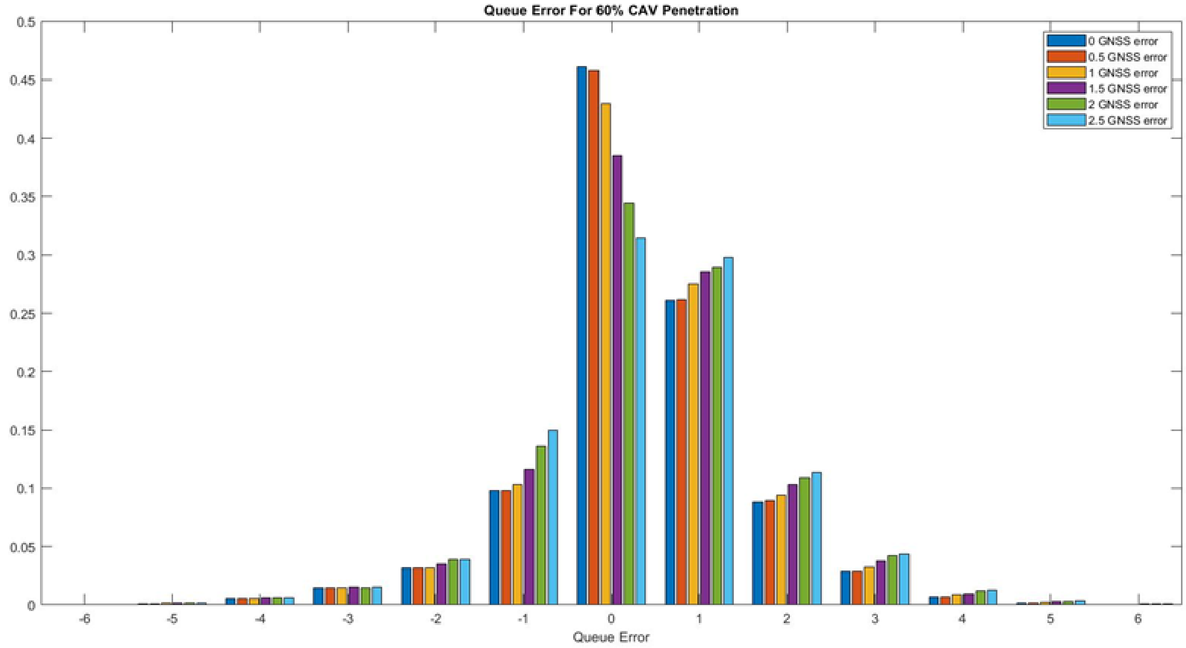
It is clear that the lane-level queue estimation accuracy using only the position measurements from the CAV's is strongly affected by the penetration rate. CAV's do have additional information from the safety sensors that can be useful, such as distance to adjacent vehicles in front, behind, and in adjacent lanes. New queue determination methods that includes this additional information could significantly improve performance at lower penetration rates.



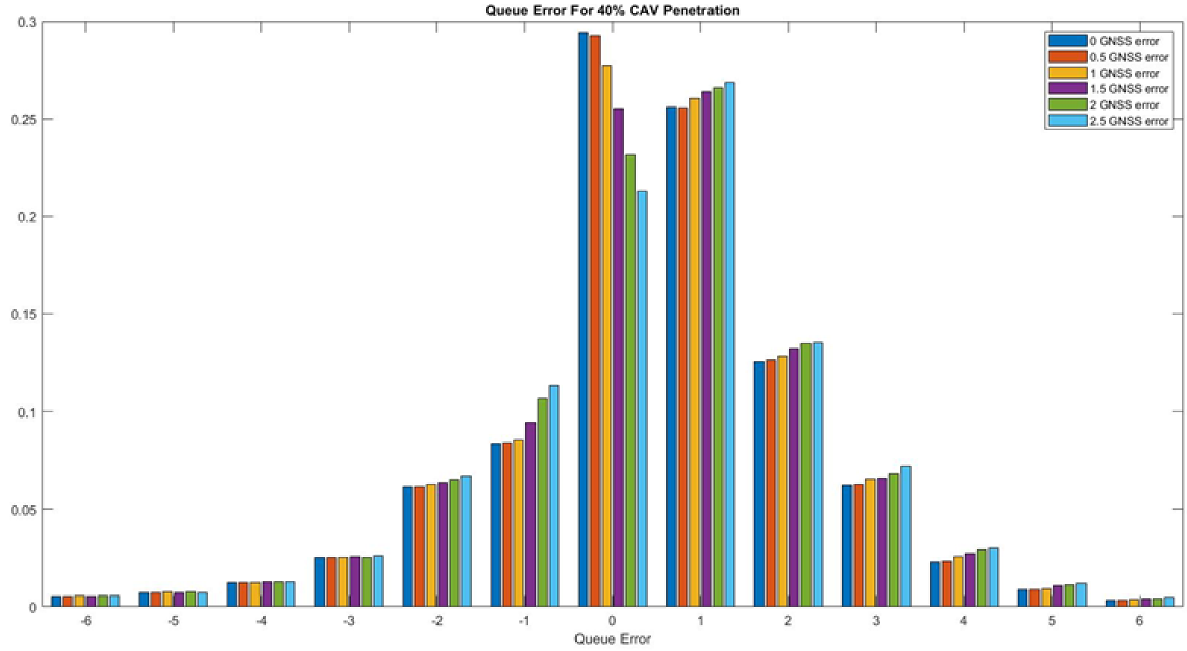
(a.)


Queue Error For 80% CAV Penetration
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
-6
-5
-4
-3
-2
-1
0
1
2
3
4
5
6
Queue Error
0 GNSS error
0.5 GNSS error
1 GNSS error
1.5 GNSS error
2 GNSS error
2.5 GNSS error

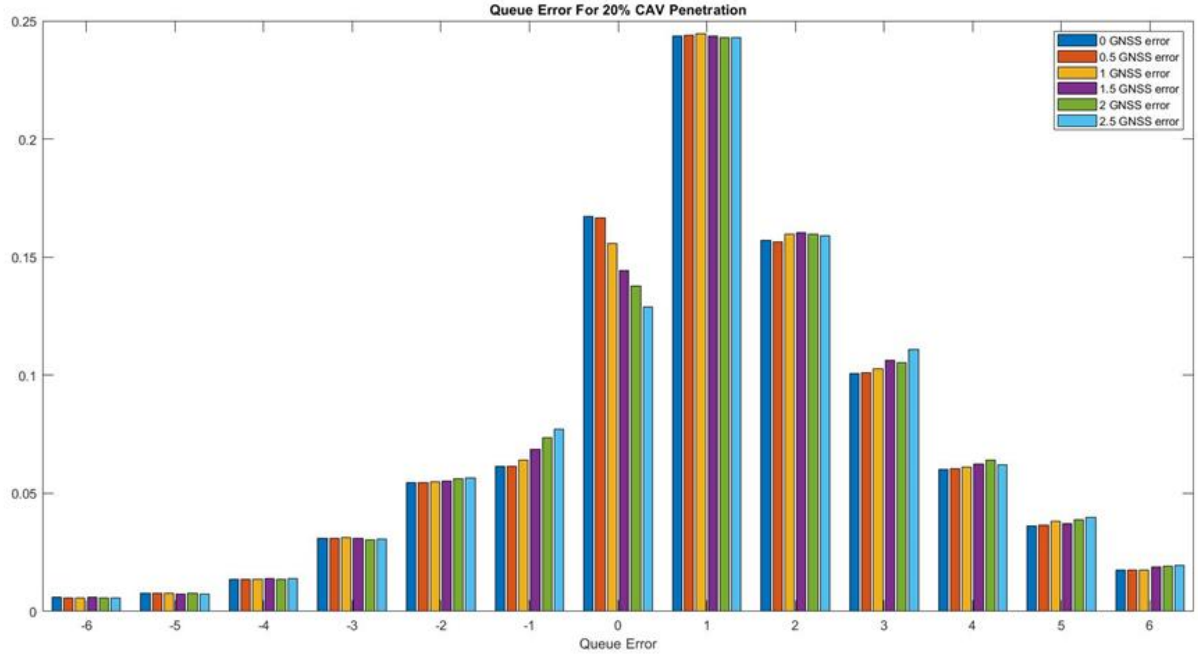
(b.)


Queue Error For 60% CAV Penetration
0.5
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
-6
-5
-4
-3
-2
-1
0
1
2
3
4
5
6
Queue Error
0 GNSS error
0.5 GNSS error
1 GNSS error
1.5 GNSS error
2 GNSS error
2.5 GNSS error

(c.)

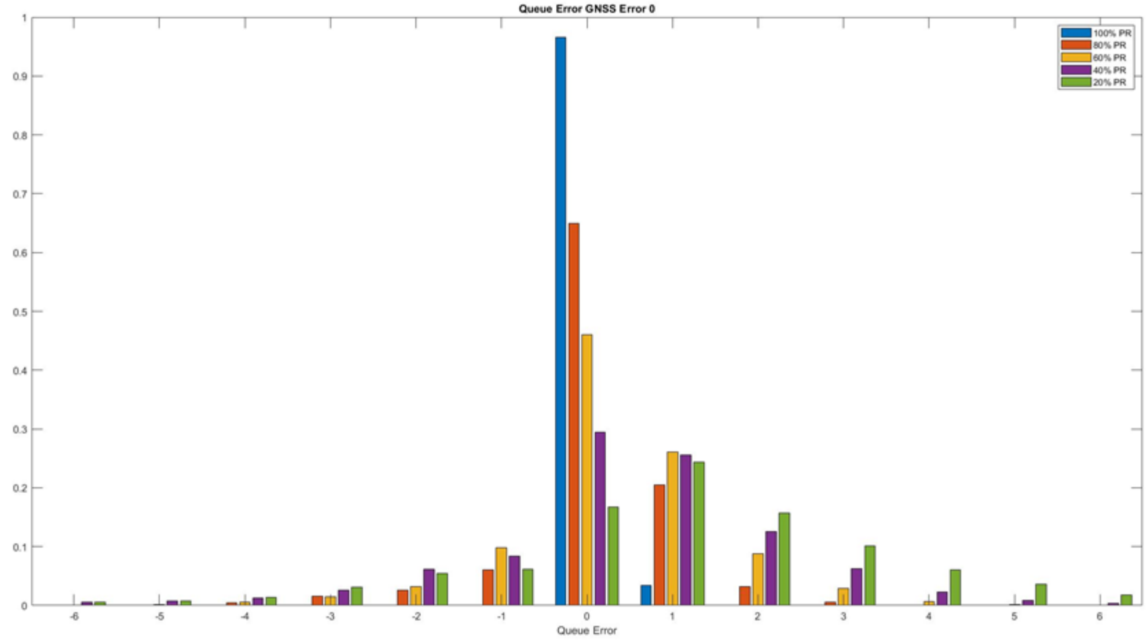


(d.)

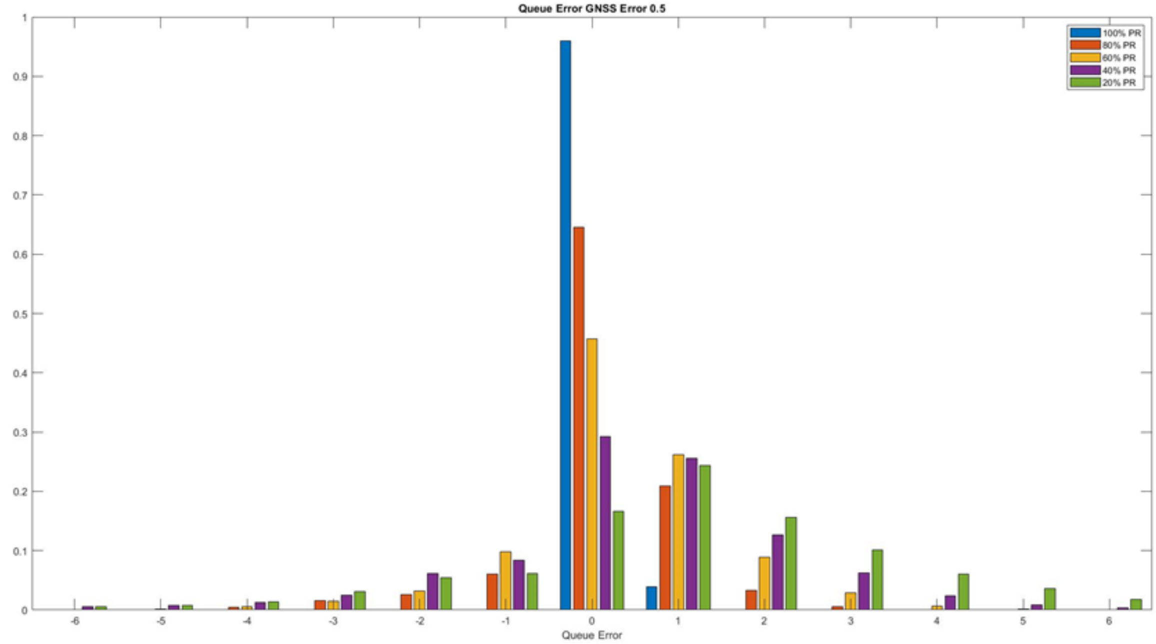


(e.)

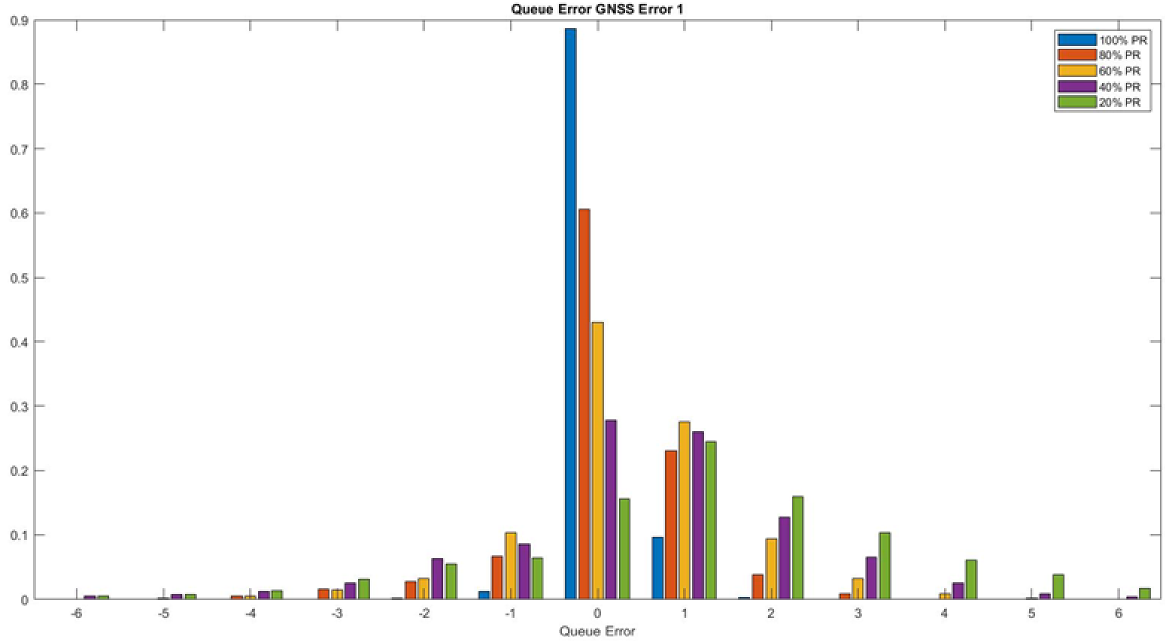
Figure 11 (a.-e.). Grouped histograms of lane-level queue length estimation error for different GNSS error levels at each individual penetration rate.


Queue Error GNSS Error 0
100% PR
80% PR
60% PR
40% PR
20% PR
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
-6
-5
-4
-3
-2
-1
0
1
2
3
4
5
6
Queue Error

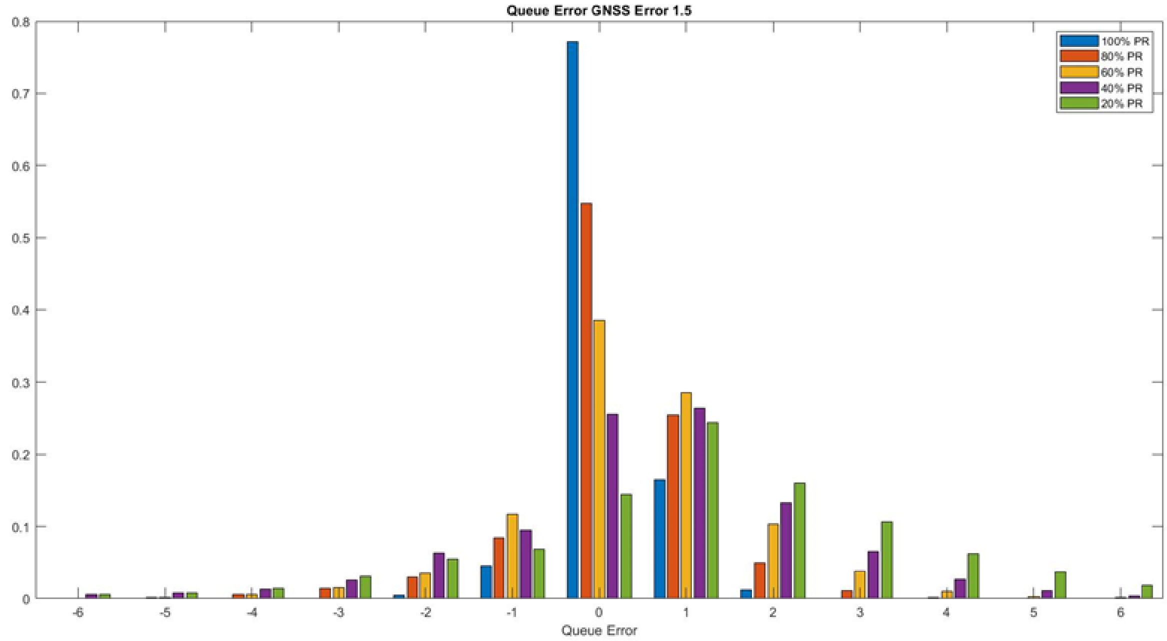
(a.)


Queue Error GNSS Error 0.5
100% PR
80% PR
60% PR
40% PR
20% PR
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
-6
-5
-4
-3
-2
-1
0
1
2
3
4
5
6
Queue Error

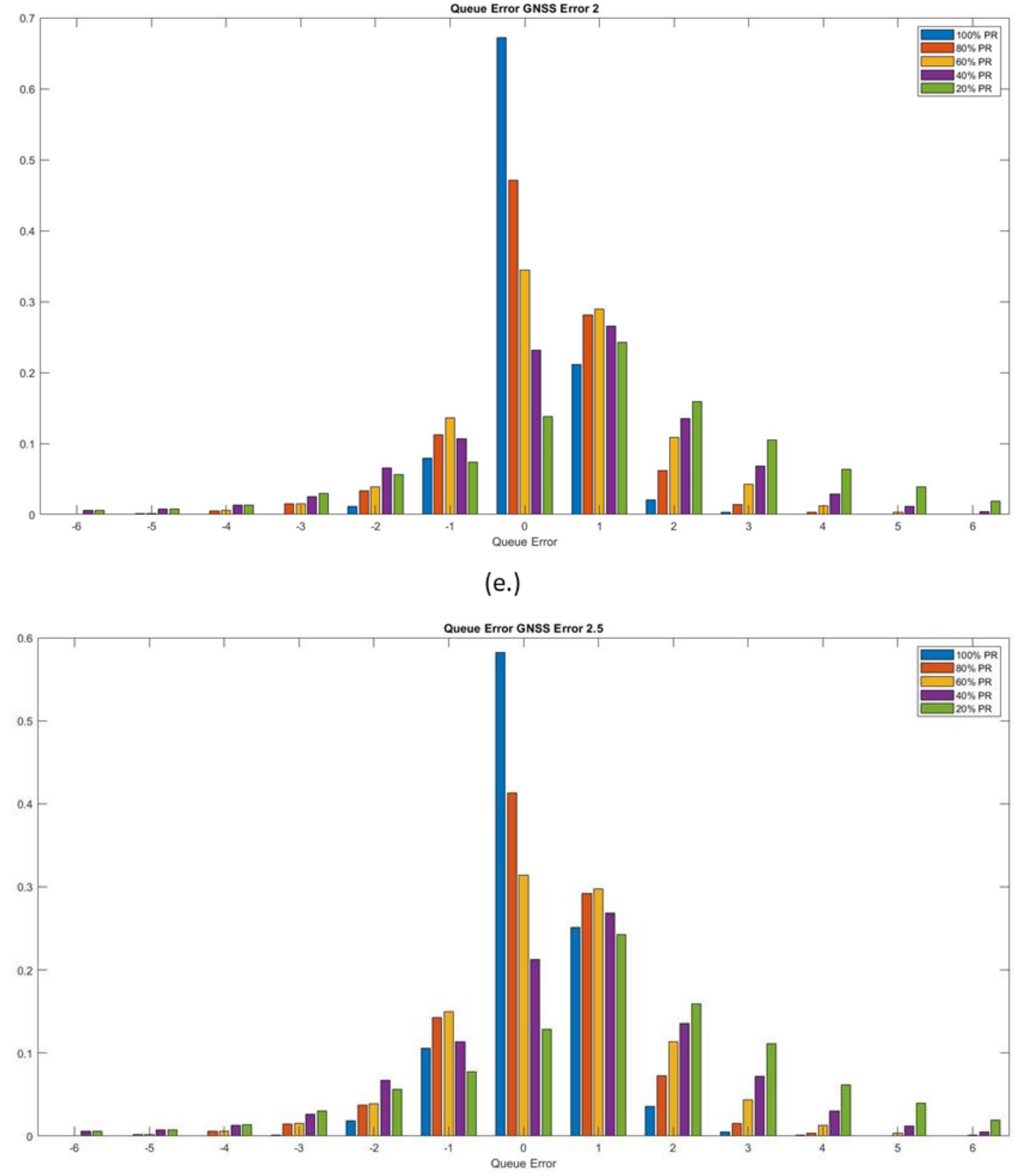
(b.)


Queue Error GNSS Error 1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
-6
-5
-4
-3
-2
-1
0
1
2
3
4
5
6
Queue Error
100% PR
80% PR
60% PR
40% PR
20% PR

(c.)


Queue Error GNSS Error 1.5
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
-6
-5
-4
-3
-2
-1
0
1
2
3
4
5
6
Queue Error
100% PR
80% PR
60% PR
40% PR
20% PR

(d.)



(e.)

(f.)

Figure 12 (a.-f.). Grouped histograms of lane-level queue length estimation error for different penetration rates of technology at each individual GNSS error level.

Task 7 – Data Management Plan Compliance

This task focused on data compliance. The specific data products of this project are defined in the Data Summary section of this report.

Task 8 – Final Report

This document is the final report.

This project funded two students: Wang Hu and David Oswald. Both are electrical engineering PhD candidates at the University of California Riverside.

In addition to the mid-project report and this final report, the project has produced:

(1) USDOT Mapping Tool accuracy assessment – This assessment is included as Appendix A and has also been submitted for publication.

(2) GNSS error model – The software implementation is being distributed through GitHub as open-source software. The theory is described in Appendix B.

(3) Lane-queue estimation algorithm – The algorithm is described theoretically in Appendix C. The software implementation is released as described under Task 7. That algorithm and its simulation results as describe under Task 6 may be organized into another article for publication.

The data products of this project are described in the Data Summary section of this report.

References

M. M. Atia et al., "A Low-Cost Lane-Determination System Using GNSS/IMU Fusion and HMM-Based Multistage Map Matching," IEEE Transactions on Intelligent Transportation Systems, vol. 18, no. 11, pp. 3027-3037, 2017.

A. Bar Hillel, R. Lerner, D. Levi, G. Raz, “Recent progress in road and lane detection: a survey,” Machine Vision and Applications 25, 727–745, 2014.

M. Barth, K. Boriboonsomsin, “ECO-ITS: Intelligent transportation system applications to improve environmental performance. FHWA-JPO-12-042, US DOT, RITA. Joint Program Office for Intelligent Transportation Systems; May 1, 2012 (https://rosap.ntl.bts.gov/view/dot/3367).

F. Bonin-Font, A. Ortiz, G. Oliver, “Visual navigation for mobile robots: A survey,” Journal of Intelligent and Robotic Systems,”53(3):263-96, 2008.

K. Boriboonsomsin, M. J. Barth, W. Zhu, A. Vu, "Eco-Routing Navigation System Based on Multisource Historical and Real-Time Traffic Information," IEEE Transactions on Intelligent Transportation Systems, 13, 4, pp. 1694-1704, 2012.

R. Bree, C. C. Tiberius, “Real-time single-frequency precise point positioning: Accuracy assessment,” GPS Solutions, 16 (2), pp. 259-266. 2012.

G. Bresson, Z. Alsayed, L. Yu, S. Glaser, “Simultaneous localization and mapping: A survey of current trends in autonomous driving,” IEEE Transactions on Intelligent Vehicles, 2(3), pp. 194-220, 2017.

C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, J. J. Leonard, “Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age,” IEEE Transactions on Robotics, 32(6), pp. 1309-1332, 2016.

S. Camacho-Lara, “Current and future GNSS and their augmentation systems,” Handbook of Satellite Applications, 2017.

M. Cao, A. Vu and M. Barth, "A Novel Omni-Directional Vision Sensing Technique for Traffic Surveillance," 2007 IEEE Intelligent Transportation Systems Conference, 2007, pp. 678-683, 2007.

L. Caltagirone, S. Scheidegger, L. Svensson, and M. Wahde, “Fast lidar-based road detection using fully convolutional neural networks,” in 2017 IEEE Intelligent Vehicles Symposium, pp. 1019–1024, 2017.

C. Chen, K. Petty, A. Skabardonis, P. Varaiya, Z. Jia, “Freeway performance measurement system: mining loop detector data,” Transportation Research Board, 80th Annual Meeting, pp. 96–102, 2001.

C. Chen, “Freeway Performance Measurement System (PeMS),” Ph.D. Dissertation, University of California, Berkeley, 2002.

Y. Cheng, X. Qin, J. Jin, B. Ran, and J. Anderson, “Cycle-by-cycle queue length estimation for signalized intersections using sampled trajectory data,” Transportation Research Record, vol. 2257, no. 1, pp. 87–94, 2011.

G. Comert, N. Begashaw, “Cycle-to-cycle queue length estimation from connected vehicles with filtering on primary parameters,” International Journal of Transportation Science and Technology, pp. 1–15, 2021.

D. J. Dailey, F. W. Cathey, “AVL-Equipped Vehicles as Traffic Probe Sensors,” Publication WA-RD 534.1. Washington State Transportation Center, University of Washington, 2002.

G. N. DeSouza, A. C. Kak, “Vision for mobile robot navigation: A survey,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(2): pp. 237-67, 2002.

E. D. Dickmanns, A. Zapp “A curvature-based scheme for improving road vehicle guidance by computer vision,” Mobile Robots I, 727, pp. 161-168, 1987.

S. P. Dmitriev, O. A. Stepanov, B. S. Rivkin, D. A. Koshaev, D. Chung, “Optimal map-matching for car navigation systems,” Proc. of 6th International Conference on Integrated Navigation Systems, 1999.

J. Du and M. J. Barth, “Next-Generation Automated Vehicle Location Systems: Positioning at the Lane Level,” IEEE Transactions on Intelligent Transportation Systems, 9(1), pp. 48 – 57, 2008.

M. Elsheikh, W. Abdelfatah, A. Noureldin, U. Iqbal, M. Korenberg, “Low-Cost Real-Time PPP/INS Integration for Automated Land Vehicles,” Sensors 2019, *19*, 4896.

M. Elsobeiey, S. Al-Harbi, “Performance of real-time Precise Point Positioning using IGS real-time service”. GPS Solutions. 2015.

J. A. Farrell, “Aided Navigation Systems: GPS and High Rate Sensors,” New York, NY, McGraw-Hill, W. Rinaldi (Ed.), 552 pages, 2008.

J. A. Farrell, M. Barth, “The Global Positioning System and Inertial Navigation: Theory and Practice,” New York: McGraw-Hill Publishing, 370 pp, 1999.

J. A. Farrell, T. D. Givargis and M. J. Barth, "Real-time differential carrier phase GPS-aided INS," in IEEE Transactions on Control Systems Technology, vol. 8, no. 4, pp. 709-721, 2000.

C. Fouque, Ph. Bonnifait and D. Btaille, "Enhancement of global vehicle localization using navigable road maps and dead-reckoning", Proc. IEEE Pos. Loc. Nav. Syst. Symp., pp. 1286-1291, 2008.

Y. Gao, Y. Zhang, K. Chen, “Development of a Real-Time Single-Frequency Precise Point Positioning System and Test Results,” *Int. Tech. Meeting of the Satellite Division of The Institute of Navigation*, pp. 2297-2303, 2006.

P. Groves, Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems, Second Edition, 2013.

P. D. Groves, “Shadow Matching: A New GNSS Positioning Technique for Urban Canyons,” The Journal of Navigation, 64(3), pp. 417 – 430, 2011.

P. D. Groves, Z. Jiang, “Height aiding, C/N0 weighting and consistency checking for GNSS NLOS and multipath mitigation in urban areas,” The Journal of Navigation, 66(5), pp. 653-669, 2013.

Y. Gu, L.T. Hsu, S. Kamijo, “GNSS/onboard inertial sensor integration with the aid of 3-D building map for lane-level vehicle self-localization in urban canyon,” IEEE Transactions on Vehicular Technology, 65(6), pp.4274-4287, 2015.

A. Hansson, E. Korsberg, R. Maghsood, E. Norden, Selpi, “Lane-level map matching based on HMM,” IEEE Transactions on Intelligent Vehicles, vol. 6, no. 3, pp. 430–439, 2021.

Hao, P., Ban, X., Guo, D., & Ji, Q., (2014). Cycle-by-cycle queue length distribution estimation using sample travel times. *Transportation Research Part B*, 68(1) 185-204.

Hao, P., Sun, Z., Ban, X., Guo, D., & Ji, Q. (2013). Vehicle index estimation for signalized intersections using sample travel times. *Transportation Research Part C*, 36(1), 513-529.

P. Hao, G. Wu, K. Boriboonsomsin, M. J. Barth, "Eco-Approach and Departure (EAD) Application for Actuated Signals in Real-World Traffic," IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 1, pp. 30-40, 2019.

R. Hatch, “The synergism of GPS code and carrier measurements,” International geodetic symposium on satellite Doppler positioning, Vol. 2, pp. 1213-1231, 1983.

B. Hofmann-Wellenhof, H. I. M. Lichtenegger and J. P. Collins, Global Positioning System: Theory and Practice, 1997.

B. Hofmann-Wellenhof, H. Lichtenegger, E. Wasle, “GNSS–Global Navigation Satellite Systems: GPS, GLONASS, Galileo, and More,” Springer, 2007.

W. Hu, A. Neupane, and J. A. Farrell, “Using PPP Information to Implement a Global Real-Time Virtual Network DGNSS Approach”, Under review, Submitted November 17, 2021. (https://arxiv.org/abs/2110.14763).

W. Hu, D. Oswald, G. Wu, J. A. Farrell, “Assessment of U.S. Department of Transportation Lane-Level Map for Connected Vehicle Applications,” submitted to 33rd IEEE Intelligent Vehicles Symposium, 2022.

S. Kammel and B. Pitzer, “Lidar-based lane marker detection and mapping,” in 2008 IEEE Intelligent Vehicles Symposium, pp. 1137–1142, 2008.

V. L. Knoop, P. F. de Bakker, C. C. J. M. Tiberius, B. van Arem, "Lane Determination with GPS Precise Point Positioning," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 9, pp. 2503-2513, Sept. 2017.

J. Kouba, P. Héroux, “Precise Point Positioning Using IGS Orbit and Clock Products”. *GPS Solutions* 5, 12–28, 2001.

H. Lee, S. Choi, H. Jung, B. B. Park, S. H. Son, “A route guidance system considering travel time unreliability,” Journal of Intelligent Transportation Systems, 23(3), 282–299, 2019.

S. Lee, S. C. Wong, Y. C. Li, “Real-time estimation of lane-based queue lengths at isolated signalized junctions,” Transportation Research Part C: Emerging Technologies, 56, pp. 1-17, 2015.

B. Li, W. Cheng, L. Li, "Real-Time Prediction of Lane-Based Queue Lengths for Signalized Intersections", Journal of Advanced Transportation, vol. 2018, pp. 1-18, 2018.

S. Liu, S. Fuping, Z. Lundong, W. Li, X. Zhu, “Tight integration of ambiguity-fixed PPP and INS: model description and initial results,” GPS Solutions, 2015.

H. X. Liu, W. Ma, “A virtual vehicle probe model for time-dependent travel time estimation on signalized arterials,” Transportation Research Part C: Emerging Technologies, 17, 1, pp. 11-26, 2009a.

H. X. Liu, X. Wu, W. Ma, H. Hu, “Real-time queue length estimation for congested signalized intersections,” Transportation Research Part C: Emerging Technologies 17 (4), pp. 412-427, 2009b.

R. Liu, J. Wang, B. Zhang, “High Definition Map for Automated Driving: Overview and Analysis,” Journal of Navigation, 73(2), 324-341, 2020.

L. Lu, Z. He, J. Wang, J. Chen, W. Wang, “Estimation of lane-level travel time distributions under a connected environment,” Journal of Intelligent Transportation Systems, 25:5, 501-512, 2021.

S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, M. J. Milford, “Visual place recognition: A survey,” IEEE Transactions on Robotics, 32(1), pp. 1-9. 2015.

C. Lundquist, T. B. Schön, “Joint ego-motion and road geometry estimation,” Information Fusion, 12, 4, pp. 253-263, 2011.

M.J. Milford, G. F. Wyeth, “SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights,” IEEE International Conference on Robotics and Automation, pp. 1643-1649, 2012.

P. B. Mirchandani, N. Zou, “Queuing models for analysis of traffic adaptive signal control,” IEEE Transactions on Intelligent Transportation Systems 8 (1), 2007.

J. Muck, “Using detectors near the stop-line to estimate traffic flows,” Traffic Engineering and Control 43, 429–434, 2002.

S. Nedevschi, V. Popescu, R. Danescu, T. Marita, F. Oniga, "Accurate Ego-Vehicle Global Localization at Intersections Through Alignment of Visual Data With Digital Map," IEEE Transactions on Intelligent Transportation Systems, 14, 2, pp. 673-687, 2013.

J. B. Neumann, K. J. VanDierendonck, A. Manz, T. J. Ford, "Real-Time Carrier Phase Positioning Using the RTCM Standard Message Types 20/21 and 18/19," 10th International Technical Meeting of the Satellite Division of The Institute of Navigation, pp. 857-866, 1997.

T. Ogawa and K. Takagi, “Lane recognition using on-vehicle lidar,” in 2006 IEEE Intelligent Vehicles Symposium, pp. 540–545, 2006.

D. Oswald, P. Hao, N. Williams, M. Barth, “Development of an Innovation Corridor Testbed for Shared Electric Connected and Automated Transportation”. UC Davis: National Center for Sustainable Transportation, 2021. http://dx.doi.org/10.7922/G21C1V6T Retrieved from https://escholarship.org/uc/item/99q6w075.

F. Peyret, D. Btaille, J. Laneurit and R. Toledo-Moreo, "Lane-level positioning for cooperative systems using EGNOS and enhanced digital maps", Proc. ENC GNSS Congr., 2008.

PTV Group PTV VISSIM, 2022. (https://www.ptvgroup.com/)

M. A. Quddus, W. Y. Ochien and R. B. Noland, “Current Map-Matching Algorithms for Transport Applications: State-of-the-Art and Future Research Directions,” New York: Elsevier Transportation Research Part C, vol. 15, pp. 312-328, 2007.

J. Rabe, M. Meinke, M. Necker, and C. Stiller, “Lane-level map-matching based on optimization,” in 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC). IEEE Press, p. 1155–1160, 2016.

F. Rahman, F. Silva, Z. Jiang, J. A. Farrell, “ECEF Position Accuracy and Reliability: Continent Scale Differential GNSS Approaches,” Phase C Report, UC Riverside: Bourns College of Engineering, 2019. https://escholarship.org/uc/item/05p9p3c9

M. Rohani, D. Gingras, D. Gruyer, "A Novel Approach for Improved Vehicular Positioning Using Cooperative Map Matching and Dynamic Base Station DGPS Concept," in IEEE Transactions on Intelligent Transportation Systems, 17, 1, pp. 230-239, 2016.

C. Rose, J. Britt, J. Allen, D. Bevly, "An Integrated Vehicle Navigation System Utilizing Lane-Detection and Lateral Position Estimation Systems in Difficult Environments for GPS," in IEEE Transactions on Intelligent Transportation Systems, 15, 6, pp. 2615-2629, Dec. 2014.

C. A. Scott, "Improved GPS position for motor vehicles through map-matching", Proc. ION GPS Conf., 1994.

K. Shunsuke, Y. Gu, H. Li-Ta, "GNSS/INS/on-board camera integration for vehicle self-localization in urban canyon," IEEE 18th Int. Conf. on Intelligent Transportation Systems, pp. 2533-2538, 2015.

A. Skabardonis, N. Geroliminis, “Real-time Monitoring and Control on Signalized Arterials,” Journal of Intelligent Transportation Systems. 12 (2), 64–74, 2008.

N. C. Talbot, "Centimeters in the Field, A users Perspective of Real-Time Kinematic Positioning in a Production Environment," 6th International Technical Meeting of the Satellite Division of The Institute of Navigation, pp. 1049-1057, 1993.

P. Teunissen, A. Khodabandeh, “Review and principles of PPP-RTK methods,” Journal of Geodesy. 1-24, 2014.

D. Tian, G. Wu, K. Boriboonsomsin, M. Barth, “Performance Measurement Evaluation Framework and Co-Benefit\/Tradeoff Analysis for Connected and Automated Vehicles (CAV) Applications: A Survey”, IEEE ITS Magazine, Vol. 10, No. 3, pp. 110 – 122, 2018.

P. J. Teunissen, O. Montenbruck, editors. “Springer Handbook of Global Navigation Satellite Systems,” New York, NY, USA, 2017.

R. Toledo-Moreo, D. Betaille, F. Peyret and J. Laneurit, “Fusing GNSS, Dead-Reckoning, and Enhanced Maps for Road Vehicle Lane-Level Navigation,” IEEE Journal of Selected Topics in Signal Processing, vol. 3, no. 5, pp. 798-809, 2009.

M. A. Turk, D. G. Morgenthaler, K. D. Gremban, M. Marra, “VITS-A vision system for autonomous land vehicle navigation,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 10(3), pp. 342-61, 1988.

S. Turner, W.L. Eisele, R.J. Benz, D.J. Holdener, “Travel Time Data Collection Handbook,” Publication FHWA-PL-98-035, Federal Highway Administration, US Department of Transportation, 1998.

US Dept. of Energy, Office of Energy Efficiency & Renewable Energy, Vehicle Technologies Office, “Fact #902: Rural versus Urban Vehicle Miles of Travel by State,” December 7, 2015.

S. Verhagen, D. Odijk, P. J. Teunissen, and L. Huisman, “Performance improvement with low-cost multi-GNSS receivers,'' IEEE NAVITECH, pp. 1--8, 2010.

A. Vu, A. Ramanandan, A. Chen, J. A. Farrell, M. Barth, “Real-Time Computer Vision/DGPS-Aided Inertial Navigation System for Lane-Level Vehicle Navigation,” IEEE Transactions on Intelligent Transportation Systems, 13, 2, pp. 899-913, 2012.

L. Wang, P. D. Groves, M. K. Ziebart, “Multi-constellation GNSS performance evaluation for urban canyons using large virtual reality city models,” The Journal of Navigation, Volume 65, Issue 3, pp. 459 – 476, 2012.

Z. Wang, Y. Bian, S. E. Shladover, G. Wu, S. Li, M. Barth, “A survey on cooperative longitudinal motion control of multiple connected and automated vehicles”, IEEE Intelligent Transportation Systems Magazine, 12(1), pp. 4 – 24, 2019.

F. Webster, “Traffic signal settings,” Road Research Technical Paper 39. Road Research Laboratory, Her Majesty’s Stationery Office, London. 1958.

F. V. Webster, B. M. Cobbe, “Traffic signals,” Road Research Technical Paper 56, Road Research Laboratory, Her Majesty’s Stationery Office, London. 1966.

N. Williams, and M. Barth, “A Qualitative Analysis of Vehicle Positioning Requirements for Connected Vehicle Applications,” IEEE Intelligent Transportation Systems Magazine, 13(1), pp. 225 – 242, 2021.

W. Wong, S. Shen, Y. Zhao, and H. X. Liu, “On the estimation of connected vehicle penetration rate based on single-source connected vehicle data,” Transportation Research Part B, vol. 126, pp. 169–191, 2019.

Q. Xu, X. Li, C.-Y. Chan, “Enhancing localization accuracy of MEMS-INS/GPS/in-vehicle sensors integration during GPS outages,” IEEE Transactions on Instrumentation and Measurement 67(8), pp. 1966-1978, 2018.

F. Ye, P. Hao, X. Qi, G. Wu, K. Boriboonsomsin, M. J. Barth, “Prediction-Based Eco-Approach and Departure at Signalized Intersections with Speed Forecasting on Preceding Vehicles,” IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 4, pp. 1378-1389, 2019.

Y. Zhao, J. Zheng, W. Wong, X. Wang, Y. Meng, and H. X. Liu, “Various methods for queue length and traffic volume estimation using probe vehicle trajectories,” Transportation Research Part C, pp. 70–91, 2019a.

Y. Zhao, J. Zheng, W. Wong, X. Wang, Y. Meng, and H. X. Liu, “Estimation of queue lengths, probe vehicle penetration rates, and traffic volumes at signalized intersections using probe vehicle trajectories,” Transportation Research Record, vol. 2673, no. 11, pp. 660–670, 2019b.

M. Zhou, H. Jin, W. Wang, “A review of vehicle fuel consumption models to evaluate eco-driving and eco-routing,” Transportation Research Part D: Transport and Environment, 49, pp. 203-218, 2016.

N. Zhu, J. Marais, D. Bétaille and M. Berbineau, “GNSS Position Integrity in Urban Environments: A Review of Literature,” IEEE Transactions on Intelligent Transportation Systems, vol. 19, no. 9, pp. 2762-2778, 2018.

Data Summary

Products of Research

In this project, data were collected in support of Tasks 2, 5, and 6. The list of data sets with descriptions are organized in Table 1. Each of these data products has been uploaded to the NCST Dryad repository. The URL to the data is described in the Data Access and Sharing Section.

The items in Table 1, are organize each into their own subdirectory within the Dryad repository for this project. In addition, there is a subdirectory named MapData that contains the J2735 map data for the Riverside Innovation Corridor. This data is necessary for the analysis of the Task 5 results, but was not produced during this project.

Table 1. Data products that resulted from the project.

Task	Data description	Data Files	File Format
2	Lane-Level Mapping. Experimental data were acquired to assess the accuracy of the USDOT Mapping Tool. A report describing the analysis and results of this task is contained in Appendix A [USDOT2022]. This data was used to create Figures 3-6 of that report. The data analysis used 39 feature points within about 200 meters of the intersection verified point and 55 feature points distributed over longer distances from the verified point (94 points total). Along with the data files, the repository includes a README file and two Matlab scripts that process the data.	click_test.m dot_data_cecert.xlsx dot_data_expand.xlsx	Matlab Excel Excel
5	Demonstration. Experimental data was acquired to assess the probability of correct lane	Contents of each data subdirectory: Test 1- RTK.csv Test 1- LD_App_log.txt Test 2- RTK.csv	 csv txt

Task	Data description	Data Files	File Format
	determination. Three road tests were performed. Their data was used to create Figure 6. The data for each test is organized into its own subdirectory. The main directory contains a README file that discusses the file contents and how to process them using the included Matlab scripts.	Test 2- VN.csv Test 3 - RTK.xlsx Test 3 - VN_GPS.xlsx	csv csv Excel Excel
6	Simulation Study. The simulation produced the data used to create Figure 9, Figure 10, Figure 11 (a-e), and Figure 12 (a-f). Each simulation run created 4 csv files: Chicago Intersection Queue information, Cranford Intersection Queue information, Iowa Intersection Queue information, and general vehicle information. Queue information consisted of the estimated queue information and actual queue information for each lane versus time. General vehicle information consisted of simulation time, vehicle id, vehicle speed, vehicle position, perturbed vehicle position, and vehicle direction. Each csv file has column headers for distinction. In total there were 1200 csv files: 4 csv files for each simulation	ChicQueueData_GNSSerror_x_ypercent_date CranQueueData_GNSSerror_x_ypercent_date IowaQueueData_GNSSerror_x_ypercent_date CAV_DLL_GNSSerror_x_ypercent_date *where x is the GNSS errors of 0, 0.5, 1.0, 1.5, 2.0, 2.5; y is CAV penetration percentages o 20, 40, 60, 80, 100; date is year-month-day_hour_minute_second. ChicQueueData… files are Chicago intersection queue data CranQueueData… files are Cranford intersection queue data IowaQueueData… files are Iowa intersection queue data CAV_DLL_… files are general vehicle information	All files are csv format

Task	Data description	Data Files	File Format
	run, 10 simulation runs for each scenario, and 30 scenarios as described in the Simulation Scenario Section.		

Data Format and Content

The file types and formats are described in Table 1.

Data Access and Sharing

The data are made available publicly via the UC Riverside instance of Dryad: https://datadryad.org/stash, which is licensed under a CC0 1.0 Universal (CC0 1.0) Public Domain Dedication license. The DOI for the dataset is https://doi.org/10.6086/D11M43.

Reuse and Redistribution

The data should be restricted for research use only. If the data are used, our work should be properly cited as:

> Jay A. Farrell, Guoyuan Wu, Wang Hu, David Oswald, Peng Hao (2022), Lane-Level Localization and Map Matching for Advanced Connected and Automated Vehicle (CAV) Applications, Project Funded by NCST 2021-2022, UC Riverside, Dataset, https://doi.org/10.6086/D11M43.

Appendix A – USDOT Lane Mapping Assessment

The following six pages are the report of the lane-level mapping accuracy of the USDOT mapping tools described in the section on Task 2. That report has also been submitted for publication [USDOT2022].

Assessment of U.S. Department of Transportation Lane-Level Map for Connected Vehicle Applications

Wang Hu, David Oswald, Guoyuan Wu, *Senior Member, IEEE* and Jay A. Farrell, *Fellow, IEEE*

***Abstract*— High-definition (Hi-Def) digital maps are an indispensable automated driving technology that is developing rapidly. There are various commercial or governmental map products in the market. It is notable that the U.S. Department of Transportation (USDOT) map tool allows the user to create MAP and Signal Phase and Timing (SPaT) messages with free access. However, a comprehensive accuracy analysis of this map tool is currently lacking in the literature. This paper provides such an analysis. The analysis uses 39 feature points within about 200 meters of the verified point and 55 feature points over longer distances from the verified point. All feature locations are surveyed using GNSS and mapped using the USDOT tool. Different error sources are evaluated to allow assessment of the USDOT map accuracy. In this investigation, the USDOT map tool is demonstrated to achieve 17-centimeter accuracy, which meets the lane-level map accuracy requirement.**

I. INTRODUCTION

Digital roadway maps have a long history, especially at the roadway-level, providing information about road interconnectivity with positions accurate to the decimeter level [1]. Using sensors such as LiDAR, radar, and digital cameras, along with techniques such as Geographic Information System (GIS) and Machine Learning, the precision of roadway maps can reach centimeter level [2], [3].

The advent of automated vehicles has motivated interest in Hi-Def digital maps which may include different capabilities: road-level, lane-level, and road features [4]. Hi-Def digital maps are a fundamental technology for connected and cooperative vehicles enabling applications such as: GNSS-based lane recognition [5], [6], per lane queue determination [7], per lane over-speed warning [8], etc. General Motors Co. had every mile of interstate in the United States and Canada mapped using LiDAR for the Cadillac's Super Cruise, which is a hands-off semi-autonomous system [9]. Waymo, Uber Technologies, and Ford Motor Co. also have fleets of vehicles out to create Hi-Def maps for use in autonomous vehicles [10]. Mobileye collects data from the millions of customers' vehicles using their front-facing cameras, and combines the data to create Hi-Def maps that will have road features, e.g., lane markers, traffic signals, and road boundaries [11], [12]. Mobileye's approach allows for maps to be constantly updated for temporary features such as constructions and roadblocks [12].

Other companies have adopted Mobileye's crowd-sourcing approach to create Hi-Def maps: TomTom (tomtom.com) and HELLA Aglaia (hella-aglaia.com) partnered to create crowd-sourced Hi-Def maps; NVIDIA [1] uses camera and radar data from their autonomous vehicle platform; Bosch and Volkswagen[2] partnered to create Hi-Def maps by leveraging data from sensors and a digital twin model; and Mitsubishi teamed up with Woven Planet (woven-planet.global) to use their Automated Mapping Platform that uses vehicle sensor data and satellite imagery to create Hi-Def maps.

While companies are working on automated technologies for high-definition roadway maps with global extent and commercial tools are available for manual construction of such maps for smaller regions, currently there are few free tools available to researchers for projects and demonstrations.

The notable exception is the USDOT J2735 MAP tool, discussed in Section II. It provides a web application user interface that uses satellite imagery to enable users to manually select and map lanes and features to create J2735 MAP messages. J2735 MAP messages describe an intersection's physical layout, such as lanes, stop bars, and allowed maneuvers, in a digital form standardized by the Society of Automotive Engineers (SAE) [13]. The USDOT MAP tool can generate binary outputs as specified for Dedicated Short-Range Communications (DSRC) roadside units [14] or usable through cellular Infrastructure-to-Vehicle (I2V) communications. At present, the literature does not include any assessment of the accuracy of the maps produced by the USDOT map tool. In addition, the establishment of MAP or SPaT message in this map tool requires a verified point for each intersection. The assessment of the effective range of one verified point is conducive to decrease the demand of the amount of verified points for dense MAP messages.

This paper investigates the accuracy of feature points extracted from the USDOT map tool in comparison with Global Navigation Satellite Systems (GNSS) survey data. Section III presents the data acquisition methods using both GNSS survey and the USDOT map tool. It also defines and quantifies the various error sources involved in the analysis. Section IV assesses the overall map accuracy and discusses the bias induced by any inaccuracy of the verified point. The assessment shows that the USDOT map tool attains 17 centimeter accuracy, which satisfies the lane-level Hi-Def map requirement (10-20 cm) [4]. The map tool maintains the same level of accuracy within at least 10 km of the USDOT

This research was conducted as part of the "Lane-Level Localization and Map Matching for Advanced Connected and Automated Vehicle (CAV) Applications" project funded by the National Center for Sustainable Transportation (NCST) during the period of performance from April 1, 2021 through March 31, 2022.

[1] URL:`https://www.nvidia.com/en-us/self-driving-cars/hd-mapping/`

[2] URL: `https://www.bosch-presse.de/pressportal/de/en/swarm-intelligence-for-automated-driving-231431.html`

map tool verified point.

II. USDOT Connected Vehicles Tool

The USDOT Connected Vehicles Tool (available at `https://webapp2.connectedvcs.com/`) offers free on-line access to tools for creating maps to support various Connected and Automated Vehicle (CAV) message types. The *ISD Message Creator* constructs lane-level intersection maps to support MAP and SPaT messages. The detailed instructions under the "Help" button make the site self-explanatory. Our interest herein is assessing the position accuracy of the J2735 MapData message output by the tool.

III. Accuracy Assessment Method

Accuracy will be assessed by comparing the coordinates of feature points determined by two different methods: the US-DOT Map Tool and GNSS Real-time Kinematic Positioning (RTK) survey. These feature points are selected to satisfy the following specifications:

- Each point should be easily and uniquely identifiable both to the surveyor and within the US-DOT tool. This is typically achieved by defining the points to be at the intersection of two nearly orthogonal lines.
- Each point should have a clear view of the sky.
- Each point should be near, but not on the road. This constraint is added to ensure the safety of the person performing the survey without needing to interrupt normal traffic operation.
- The features in the US DOT imagery and the real environment should be at the same locations. The US DOT imagery is based on georectification of historic photos that may have been taken months in the past; therefore, recent changes in the real environment may not be accurately represented in that imagery.

Figures 1 and 2 use imagery from the US-DOT tool to show the geographic distribution of the feature points. Each orange dot in Fig. 1 shows the location of each of $N_1 = 39$ feature points near University of California-Riverside (UCR) College of Engineering Center for Environmental Research and Technology (CE-CERT). One of these points is defined as the verified point (denoted $\mathbf{P}^e_v$) for the US-DOT tool. The feature points in Fig. 1 are each within about 200 meters of the verified point. The solid red box displays the region within the dashed red line at the maximum zoom level allowed by the tool. Fig. 2 uses orange dots to indicate the location of 11 survey areas. Each survey area, labeled from S_1 to S_{11}, includes 5 feature points. These $N_2 = 55$ feature points allow accuracy to be assessed over longer distances from the verified point. The red box in Fig. 2 indicates the region portrayed in Fig. 1.

To assess accuracy, we compare GNSS survey and US-DOT mapping tool locations for each feature point. The symbol $\mathbf{P}^e$ denotes the feature position determined by GNSS survey. The symbol $\hat{\mathbf{P}}^e$ denotes the position of the feature point determined by the US DOT mapping tool. The superscript on the vector $\mathbf{P}$ denotes the frame-of-reference, such as e for Earth-Centered Earth-Fixed (ECEF) and g for the North, East and Down (NED) frame. The NED frame feature location of a point $\mathbf{P}^g$ is computed by

$$\mathbf{P}^g = \mathbf{R}^g_e \left(\mathbf{P}^e - \mathbf{P}^e_v\right) \tag{1}$$

where $\mathbf{P}^e_v$ is the origin of the NED frame and $\mathbf{R}^g_e$ is the rotation matrix from the ECEF frame to the NED frame [15]. Eqn. (1) is valid both for GNSS survey and USDOT mapping tool locations.

Section III-A discusses the GNSS survey and assesses its sources of error when determining the coordinates of each point. Section III-B discusses the US-DOT tool and assesses its sources of error when determining the coordinates of each point. Section IV combines the US-DOT and GNSS data to assess the overall map accuracy.

A. Data Acquisition: GNSS Survey

This section presents the procedure for determining the real-world position of the verified point and of each feature point (denoted $\mathbf{P}^e_k$ for $k = 1, \ldots, N$) by use of GNSS RTK survey, using a dual-frequency ublox ZED-F9P receiver connected to a dual-band ublox antenna. The antenna is placed on the ground above the corresponding feature point. The receiver communicates with the UCR base station to obtain Radio Technical Commission for Maritime Services (RTCM) corrections and reports RTK fixed position solution in WGS84 ECEF frame.

During the survey process for each point, the ZED-F9P in RTK fixed mode was used to record the position for at least 20 seconds. The mean of these measurements is used in Sec. IV as the surveyed position. The standard deviation of each coordinate in each surveyed position is less than 0.005 m. The RTK GNSS surveyed position Mean Square Error (MSE), denoted herein as σ_G at the centimeter level (see e.g., Table 21.7 in [16]).

In addition to the RTK GNSS survey position error characterized by σ_G, there is also antenna placement error due to the fact that the human operator cannot perfectly place the antenna over the feature and account for the antenna phase offset. This error is accounted for by the symbol σ_S with MSE $\sigma_S = 0.01\,m$.

B. Data Acquisition: USDOT Map Tool

The goal of this section is two-fold: (1) to describe the process by which this tool was used to obtain the geodetic coordinates for the selected locations; and (2) to define and assess the related sources of error.

Process. Starting from the URL for the US-DOT tool given in Section II, the steps are as follows:

1) In the ISD Message Creator,
   a) Click 'View Tool', then under 'File' button click 'New Parent Map'.
   b) Center the map imagery over the region of interest at the 'Zoom Level 21', which is the highest resolution, as shown in the inset of Fig. 1.
   c) Click 'Builder' from the left bottom corner.
   d) Drag the 'Verified Point Marker' to the feature point defined in Section III and shown in Fig. 1. A 'Verified

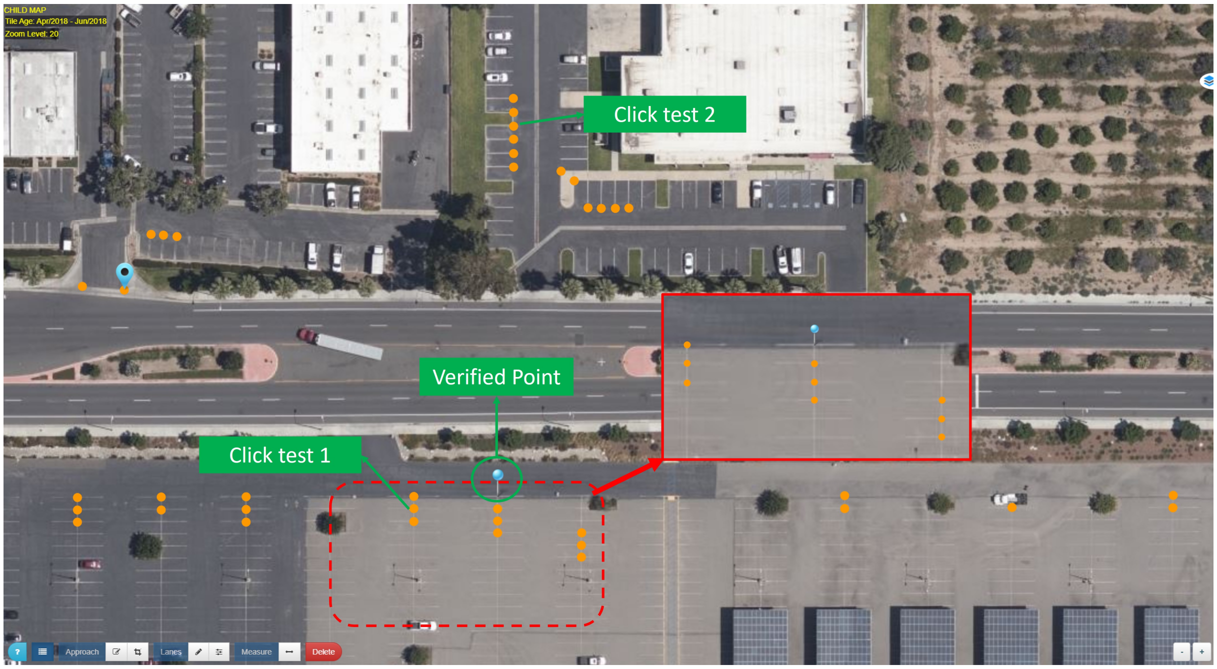


Fig. 1: USDOT map accuracy test points near UCR CE-CERT

Point Configuration' window will automatically open. Input the GNSS survey coordinates for the verified Latitude/Longitude/Elevation.

e) Drag the 'Reference Point Marker' near the verified point in the map. The reference point is required for the tool. It determines the relative position of all feature locations in the J2735 map message, but does not affect the results of the experiments.

2) Under the 'File' button from the top menu, select 'New Child Map'. Click 'Cancel' for the popup questions. Use the pencil in the 'Lanes' button located near the left bottom corner. Double-click each desired feature location. An orange dot will be displayed as shown Fig. 1.
3) Click the pencil in the 'Lanes' button to turn it off. Then, select (i.e., mouse click) each feature point in the tool imagery (e.g., orange points in Fig. 1) and note their coordinates as $\hat{\mathbf{P}}^e_k$.

Note that all positions acquired from the US-DOT tool are WGS84 ECEF geodetic coordinates.

Error Sources. The above process allows measurement error to occur in at least two ways. First, the user will have error in the clicking of points. For example, Steps 1d and 2 involve mouse clicks to select points. At best, the accuracy of such mouse clicks will be the size of the pixel in meters; however, the screen resolution may result in lower accuracy. The click error will be denoted by σ_C. Second, the geodetic coordinates assigned to the clicked points will be imperfect due to georectification errors. This mapping error will be denoted by σ_M.

Error Assessment. The goal of this subsection is to characterize the click accuracy σ_C in meters. Point-click experiments are performed for two feature points, which are marked as 'Click test' in Fig. 1. For each experiment, using 'Zoom level 21', the targeted feature point is manually clicked 15 times (moving the cursor away and returning it between clicks) and their position $\hat{\mathbf{P}}^e_{C_i}$ is recorded for $i = 1, \ldots, 15$.

TABLE I: Standard deviation for click test

	σ_N	σ_E	σ_C	σ_V
Click test 1	0.053 m	0.028 m	0.060 m	0.0 m
Click test 2	0.042 m	0.032 m	0.053 m	0.0 m

The accuracy analysis is performed in a locally-level NED tangent frame with its origin point at the verified position $\hat{\mathbf{P}}^e_v$. The NED feature location $\hat{\mathbf{P}}^g_{C_i}$ is computed from $\hat{\mathbf{P}}^e_{C_i}$ using Eqn. (1).

Herein, click test error is characterized by the Standard Deviation (STD) of each component of $\mathbf{P}^g_{C_i} = [P^g_{N_i}, P^g_{E_i}, P^g_{D_i}]^T$. The STD of North σ_N and East σ_E are listed in Table I. The vertical STD σ_V is 0 since there are no changes in the Down coordinates in each click of each experiment. The horizontal STD, which defines the click accuracy σ_C, is calculated by

$$\sigma_C = \sqrt{\sigma_N^2 + \sigma_E^2}. \tag{2}$$

The values of σ_C summarized in Table I, will be used in Section IV-C to estimate a value for σ_M.

IV. Accuracy Assessment

This section uses the USDOT data in comparison with the GNSS survey data to assess the accuracy of the feature locations provided by the USDOT mapping tool.

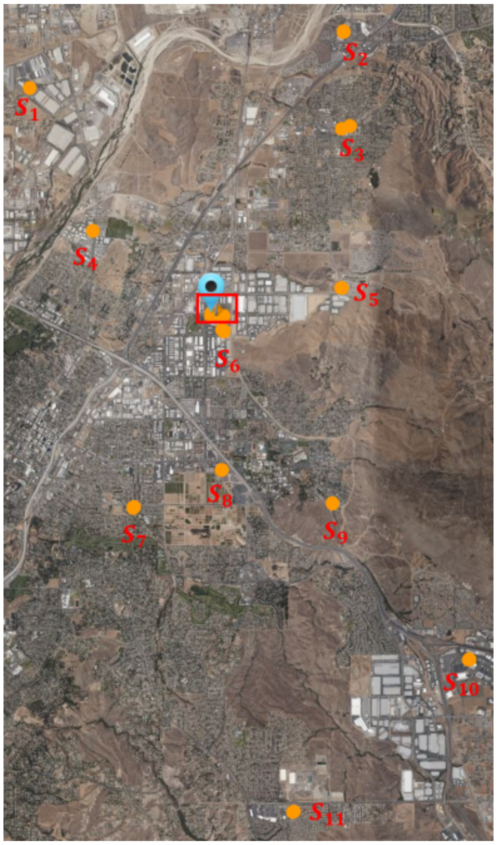


Fig. 2: Expanded test area for accuracy analysis.

A. *Bias Analysis on USDOT Tool: Verified Point*

Fig. 3 displays the north and east components of the error between the USDOT feature points from each click test $\hat{\mathbf{P}}^g_{C_i}$ and the GNSS surveyed positions $\mathbf{P}^g_C$ for the same features. For each click test, the NED frame positions $\hat{\mathbf{P}}^g_{C_i}$ and $\mathbf{P}^g_C$ are computed using Eqn. (1). The position error is calculated by

$$\delta \mathbf{P}^g_{C_i} = \mathbf{P}^g_{C_i} - \mathbf{P}^g_C \tag{3}$$

where $\delta \mathbf{P}^g_{C_i} = [\delta P^g_{N_i}, \delta P^g_{E_i}, \delta P^g_{D_i}]^T$ are the NED components of mapping error for click test C_i.

Note that both the north and east components of the position error vector are biased by -0.13 m and 0.21 m, respectively. Due to the fact that the bias is statistically the same for both feature points (i.e., click tests) and all clicks, this bias is attributed to the error in the placement on the verified point within the USDOT tool. See also the discussion of Figs. 4b and 5b.

B. *Feature Mapping Accuracy Analysis*

The USDOT map tool provides geographic coordintes (i.e., Longitude, Latitude, and Altitude) for feature points. The geographic coordinates are transferred to ECEF coordinates using the method described in Eqns. (2.9-2.11) of [17], then to local tangent plane using Eqn. (1). The USDOT location of feature k is denoted as $\hat{\mathbf{P}}^e_k$ and $\hat{\mathbf{P}}^g_k$. The GNSS surveyed location of feature k is denoted as $\mathbf{P}^e_k$ and $\mathbf{P}^g_k$. The position error for feature k is computed as

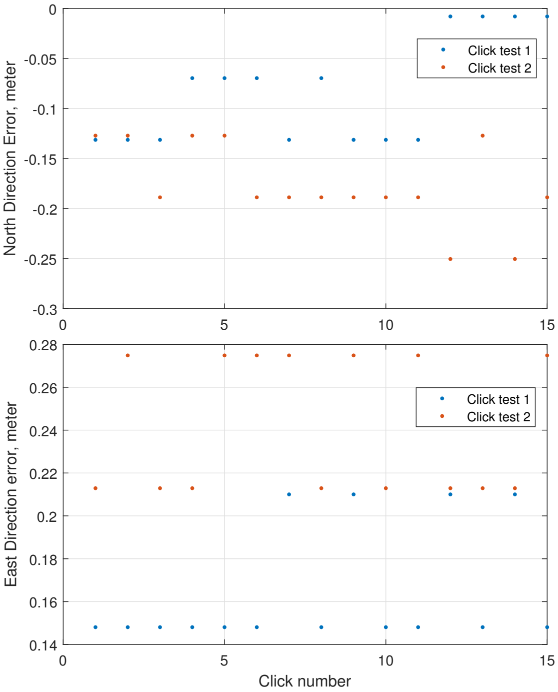


Fig. 3: North and east errors between USDOT tool and GNSS survey (i.e, $\delta P^g_{N_i}$ and $\delta P^g_{E_i}$ from eqn. (3)) using the two feature points from the click test.

$$\delta \mathbf{P}^g_k = \hat{\mathbf{P}}^g_k - \mathbf{P}^g_k = \mathbf{R}^g_e(\hat{\mathbf{P}}^e_k - \mathbf{P}^e_k)$$

where $\delta \mathbf{P}^g_k = [\delta P^g_{N_k}, \delta P^g_{E_k}, \delta P^g_{D_k}]^T$ defines the north, east, and down components of the error vector. The metrics for analyzing the accuracy of the k-th feature are the *horizontal error norm*:

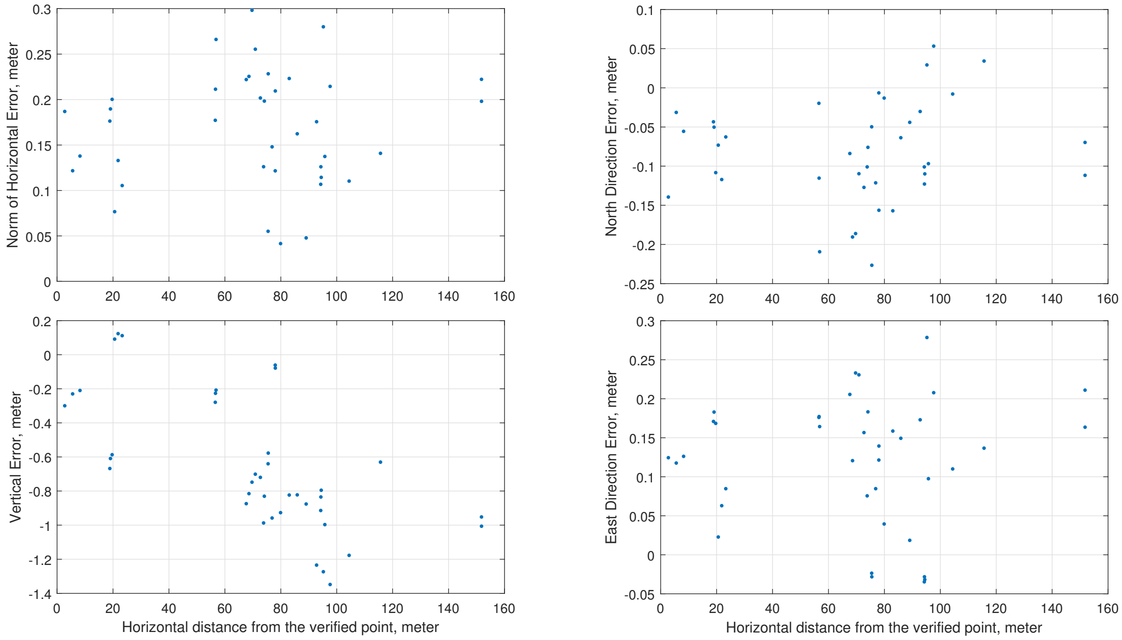
$$\delta P^g_{H_k} = \sqrt{(\delta P^g_{N_k})^2 + (\delta P^g_{E_k})^2};$$

and, the *vertical error*: $\delta P^g_{D_k}$. The Horizontal Distance (HD) between the k-th test point and verified point ($\mathbf{P}^g_v = \mathbf{0}$) is

$$D_{H_k} = \sqrt{(\mathbf{P}^g_{N_k})^2 + (\mathbf{P}^g_{E_k})^2}.$$

Fig. 4 displays data for assessing accuracy for the features shown in Fig. 1 that are near CE-CERT. Fig. 4a displays the horizontal error norm and vertical error for the feature points near the UCR CE-CERT. Fig. 5 presents data for the expanded area shown in Fig. 2. The expanded area includes 11 clusters. Data for each cluster is depicted in a different color in Fig. 5. In each figure the x-axis is the horizontal distance D_{H_k} from the verified point. Fig. 4a shows 0.17 m mean and 0.30 m maximum horizontal error. Fig. 5a shows the horizontal error norm and vertical errors over longer horizontal distances from the verified point. Fig. 5a shows 0.18 m mean and 0.31 m maximum horizontal error. There are no discernible trends in the horizontal error as a function of the distance from the verified point.

Fig. 5a also shows that the vertical error does change as a function of the distance from the verified point. The



(a) Horizontal error norm $\delta P^g_{H_k}$ and vertical error $\delta P^g_{D_k}$ versus HD to the verified point (i.e., D_{H_k}).

(b) North and East error components versus HD to the verified point (i.e., D_{H_k}).

Fig. 4: USDOT map accuracy assessment near UCR CE-CERT.

tool georectifies remote sensing satellite imagery to achieve its accuracy in the horizontal directions. Satellite imagery does not provide depth information; therefore, the underlying vertical accuracy is limited.

Figs. 4b and 5b show the individual components of the horizontal error. In Fig. 4b the mean north and east errors are -0.08 m and 0.12 m, respectively. In Fig. 5b the mean north and east errors are -0.08 m and 0.15 m, respectively. These biases are consistent with each other and with those in Fig. 3. This verifies the conclusion that the verified point selected within the USDOT tool is biased by this amount relative to the desired feature point, due to the limited resolution of the imagery in that tool.

The symbol σ_H represents the MSE of the experimental horizontal position error $\delta P^g_{H_k}$. The MSE of $\delta P^g_{H_k}$ is 0.18 m for N_1 points and 0.20 m for N_2 points. The MSE σ_H over all 94 feature points is 0.19 m.

Fig. 6 plots the horizontal and vertical errors versus vertical difference relative to the verified point. The horizontal accuracy remains constant as elevation changes. The vertical error is an order of magnitude larger than the horizontal error and does change with both the horizontal and vertical separation from the reference point.

C. Map Accuracy Assessment

The experimental horizontal position error σ_H is the result of the four specific errors discussed in Sections III-A and III-B, specifically:

$$\sigma_H^2 = 2\,\sigma_C^2 + \sigma_M^2 + \sigma_G^2 + \sigma_S^2.$$

The US-DOT click accuracy σ_C is multiplied by 2 since it is applied to the clicks for both the feature point and the verified point. Since we have experimentally determined values for σ_H, σ_C, σ_G, and σ_S, we can compute $\sigma_M = \sqrt{\sigma_H^2 - \left(2\,\sigma_C^2 + \sigma_G^2 + \sigma_S^2\right)}$. Using either value of σ_C from the two click tests, the resulting value of σ_M is 0.17 m.

V. Conclusions

Hi-Def digital maps are an indispensable automated driving technology for CAV applications. The USDOT map tool allows users to create MAP and SPaT messages with free access, but an assessment of its accuracy does not exist in the current literature. This document assessed the accuracy of the US-DOT map tool using a set of 94 feature points with an 10 km area. The assessed accuracy is 17 centimeters. The assessment also demonstrated that this horizontal map accuracy was maintained within the 10 km distance of the USDOT map tool verified point that was used in this study.

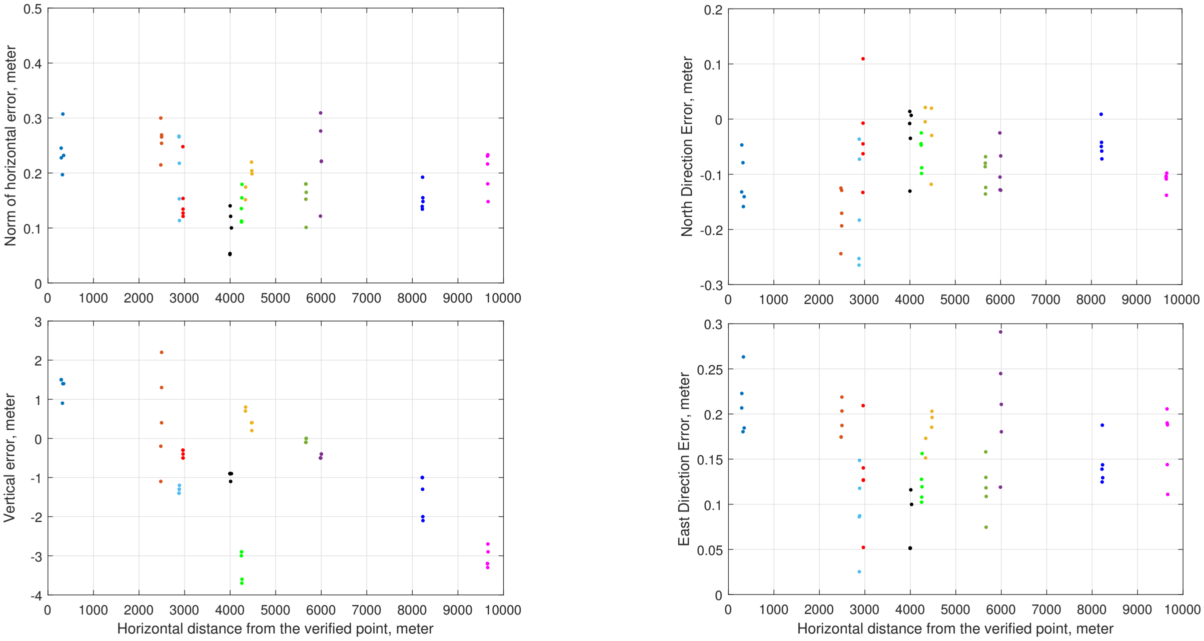
References

[1] M. White, "Emerging Requirements for Digital Maps for In-Vehicle Pathfinding and Other Traveller Assistance," in *Vehicle Navigation and Information Systems Conference*, vol. 2. IEEE, 1991, pp. 179–184.

[2] K. Massow, B. Kwella, N. Pfeifer, F. Häusler, J. Pontow, I. Radusch, J. Hipp, F. Dölitzscher, and M. Haueis, "Deriving HD maps for highly automated driving from vehicular probe data," in *19th Int. Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2016, pp. 1745–1752.

[3] J. Jiao, "Machine Learning Assisted High-Definition Map Creation," in *42nd Annual Computer Software and Applications Conference (COMPSAC)*, vol. 1. IEEE, 2018, pp. 367–373.

[4] R. Liu, J. Wang, and B. Zhang, "High definition map for automated driving: Overview and Analysis," *The J. of Navigation*, vol. 73, no. 2, pp. 324–341, 2020.



(a) Horizontal error norms and Vertical error

(b) North and East error components

Fig. 5: Graphs of USDOT map accuracy assessment for expanded area. S_1 in green with HD near 5.7 km, S_2 in dark purple with HD near 6 km, S_3 in amber with HD near 4.4 km, S_4 in light blue with HD near 2.9 km, S_5 in orange with HD near 2.4 km, S_6 in blue with HD near 0.3 km, S_7 in black with HD near 4 km, S_8 in red with HD near 3 km, S_9 in green with HD near 4.2 km, S_{10} in deep blue with HD near 8.2 km, and S_{11} in magenta with HD near 9.7 km.

[5] S. Rogers, P. Langley, and C. Wilson, "Learning to predict lane occupancy using GPS and digital maps," in *Proc. of the 5th Int. Conference on Knowledge Discovery and Data Mining*, 1999, pp. 104–113.

[6] C. Yan, C. Zheng, C. Gao, W. Yu, Y. Cai, and C. Ma, "Lane Information Perception Network for HD Maps," in *23rd Int. Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2020, pp. 1–6.

[7] V. Potó, Á. Somogyi, T. Lovas, Á. Barsi, V. Tihanyi, and Z. Szalay, "Creating HD map for autonomous vehicles-a pilot study," in *34th Int. Colloquium on Advanced Manufacturing and Repairing Technologies in Vehicle Industry*, 2017.

[8] J. Lógó, N. Krausz, V. Potó, and A. Barsi, "Quality Aspects of High-Definition Maps," *The Int. Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 43, pp. 389–394, 2021.

[9] B. Pal, S. Khaiyum, Y. Kumaraswamy *et al.*, "Recent advances in Software, Sensors and Computation Platforms Used in Autonomous Vehicles, A Survey," *Int. J. Res. Anal. Rev*, vol. 6, no. 1, 2019.

[10] M. Szántó and L. Vajta, "Introducing crowdmapping: A novel system for generating autonomous driving aiding traffic network databases," in *2019 Int. Conference on Control, Artificial Intelligence, Robotics & Optimization (ICCAIRO)*. IEEE, 2019, pp. 7–12.

[11] M. Abrams and T. Romer, "Eyes on the Road," *Mechanical Engineering*, vol. 139, no. 12, pp. 33–33, 2017.

[12] K. Wong, Y. Gu, and S. Kamijo, "Mapping for Autonomous Driving: Opportunities and Challenges," *IEEE Intell. Transp. Syst. Mag.*, vol. 13, no. 1, pp. 91–106, 2020.

[13] C. Hedges and F. Perry, "Overview and use of SAE J2735 message sets for commercial vehicles," SAE Technical Paper, Tech. Rep., 2008.

[14] J. B. Kenney, "Dedicated Short-Range Communications (DSRC) Standards in the United States," *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1162–1182, 2011.

[15] W. Hu, A. Neupane, and J. A. Farrell, "Using PPP Information to Implement a Global Real-Time Virtual Network DGNSS Approach," *arXiv preprint arXiv:2110.14763*, 2021.

[16] P. Teunissen and O. Montenbruck, *Springer Handbook of Global Navigation Satellite Systems*. Springer, 2017.

[17] J. A. Farrell, *Aided Navigation: GPS with High Rate Sensors*. McGraw-Hill, Inc., 2008.

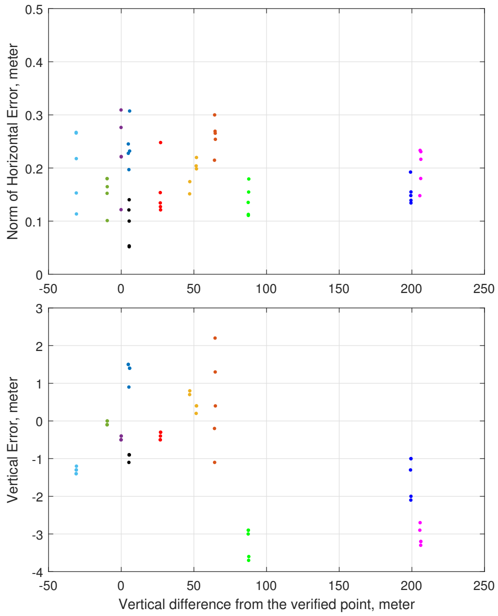


Fig. 6: Horizontal error norms and Vertical error of expanded area.

Appendix B – GNSS Position Error Model for PTV VisSim

The following six pages describe the theory that enabled development of a GNSS position error model for PTV VisSim.

PTV Vissim GNSS Position Error Module: User Manual and Technical Description

Wang Hu, David Oswald, Guoyuan Wu, Jay A. Farrell
Department of Electrical and Computer Engineering
University of California, Riverside

April 1, 2022

Abstract

The default vehicle position reported by PTV Vissim is free from measurement error. As CAV applications continue to advance and to utilize GNSS to provide position estimates, it is becoming increasingly important to study the effects of position measurement error on the performance of CAV algorithms.

The purpose of this document is to serve both as a user manual for and technical description of a Global Navigation Satellite Systems (GNSS) position error model developed at UCR for use within Vissim.

1 PTV Vissim: GNSS Position Error Module

For each vehicle, the error module computes a three-dimensional GNSS position error vector that is added to the three-dimensional ground-truth position vector that is supplied by the PTV Vissim simulation for that vehicle. The position error vector for each vehicle is independent of those for all other vehicles.

1.1 Software Repository

The GNSS error model is added to the External Driver Model DLL for PTV Vissim. The PTV PTV Vissim DriverModel DLL – Interface Documentation says, "The External Driver Model DLL Interface of PTV Vissim provides the option to replace the internal driving behavior by a fully user-defined behavior for some or all vehicles in a simulation run."

A PTV Vissim External Driver Model that includes the GNSS Position Error Module function is named 'gnss_error'. The C++ source code, dll executable and readme file are available on the Dryad project repository[1] within the "Task6 VISSIM/VISSIM_PositionErrorSource Code" subdirectory.

To use the 'gnss_error' model, download the executable for this dll. To change the error model functionality or to add it into existing driver error models, download the dll's source code, edit it, compile, then use as described below.

1.2 Functionality in PTV Vissim

The DriverModel.dll is used by checking "Use external driver model" in the "Vehicle Type" dialog box. This allows the user to choose specific type(s) of vehicles to use the DriverModel.dll. All vehicles of the type(s) chosen will have the driving behavior modeled by the selected DLL. There is also an option to browse for the DriverModel.dll to choose the correct file.

The GNSS error model for PTV Vissim takes the vehicle location provided by PTV Vissim (i.e., $\mathbf{p}(t)$) and adds an error $\delta\mathbf{p}(t)$. See eqn. (1). This allows for testing applications such as lane-determination in PTV Vissim simulation as a function of the GNSS position error.

PTV Vissim performs simulations with a step-size denoted by T. The current simulation has 10 time steps per simulation second. It can be changed under "Simulation/Parameters" in the main menu. Within PTV Vissim the value of T, which also serves as the position measurement period, is a global variable that is available to be used by the GNSS position error module.

1.3 Parameters that Users Can Adjust

The module allows the user to set two parameters:

$\bar{P}$: This is the steady-state covariance of the per component position error in m^2.

T_c: This is the correlation time of the position error in seconds.

These parameters have default values of $\bar{P} = 0.25m^2$ and $T_c = 60s$.

Reasonable ranges for these values are $\bar{P} \in [0.1, 100]m^2$ and $T_c \in [30, 300]s$. The value of $\bar{P} = 0.1m^2$ corresponds to a position accuracy of $0.32m$, which could result from Differential GNSS (DGNSS) operation in an environment with small multipath. The value of $\bar{P} = 100m^2$ corresponds to a position accuracy of $10.00m$, which is cause predominantly by atmospheric delay in Single Frequency (SF) GNSS Open Service (OS). Smaller values of T_c correspond to moving vehicles using DGNSS, because the common-mode errors are removed and the reflective surfaces that cause multipath are changing frequently. Stationary vehicles or those using OS DGNSS should use larger values of T_c.

The parameters $\bar{P}_N\, m^2$, $\bar{P}_E\, m^2$ and the correlation times for each axis are set once, before running the PTV Vissim simulation. Their values are set in the 'gnss_error_config.txt' file. They are then used within the software module to determine model parameters and a time-correlated position error sequence.

[1] The URL is https://datadryad.org/stash/share/bEuqWGxhCZYEkUV-XHI6OxbLNRLrcJMRRNg9daPqy5E.

2 Error model: Technical Description

For each vehicle, PTV Vissim supplies the *ground truth* position vector $\mathbf{p}(t) \in \Re^3$ at each simulation time step $t = kT$, where T is the simulation time-step (see Section 1.2). This module computes an position error vector $\delta\mathbf{p}(t) \in \Re^3$ such that each component has steaty-state variance $\bar{P}$ and correlation time T_c. The three components of the position error vector are uncorrelated with each other. The position error vector for each vehicle is independent from that of other vehicles. The measurement of the position that is output by the module is computed as

$$\tilde{\mathbf{p}}(t) = \mathbf{p}(t) + \delta\mathbf{p}(t). \tag{1}$$

Section 2.1 defines notation that will be used in the model. Section 2.2 describes the implementation of the error model that is implemented for each component of $\delta\mathbf{p}(t)$.

2.1 Parameters and Notation

The position error will contain three types of error

$$[\delta\mathbf{p}(t)]_i = x_i(t) + \mu_i + \eta_i(t) \tag{2}$$

for $i = 1, \ldots, 3$, where $[\delta\mathbf{p}(t)]_i$ represents the i-th component of the vector $\delta\mathbf{p}(t)$ at time t. The symbol $x_i(t)$ represents the time correlated portion of the error. It is assumed to be wide-sense stationary, with zero mean, steady-state covariance $\bar{P}$, and correlation time T_c. The parameters $\bar{P}$ and T_c are user-adjustable, see Section 1.3. The symbol μ_i represents the mean value of the error. This term is included to enable discussion of its effect in the analysis of Section 4.3. This is necessary, because when analyzing finite duration experimental data, the mean of that data is removed as a first step. In actual GNSS performance, over long durations, the mean of the position error is zero. The symbol $\eta_i(t)$ represents white measurement noise with $var(\eta_i(t)) = \sigma^2_{\eta_i}$.

The simulation model is implemented in discrete-time with $t = kT$ for $k = 0, 1, 2, \ldots$ where T is the simulation time step. The time-step T is defined by PTV Vissim (see Section 1.2).

The following section presents an example first-order Gauss-Markov model for the error state $x_i(t)$. It can be expected to reproduce the time-correlated nature of the GNSS errors for values of T_c up to 60-100 seconds. Higher-order models may yield better fidelity for larger values of T_c, but will have higher computational cost.

2.2 Time-Correlated Error Model: $x_i(t)$ at $t = kT$

As discussed relative to eqn. (1), for each vehicle and each time, the position error is a three-dimensional vector. This section discussed the model for $x_k = x_i(t)$ at $t = kT$. In this notation, the subscript i is dropped. The same model is used for each components of $\delta\mathbf{p}$ and all vehicles. Each component for each vehicle is driven by its own independent white noise process.

The first-order Gauss-Markov discrete-time state transition model is

$$x_{k+1} = \alpha x_k + \gamma \omega_k, \tag{3}$$

where $\alpha = exp(-T/T_c)$ and $\gamma = \sqrt{\bar{P}(1-\alpha^2)}$ are model parameters, and ω_k is the white Gaussian noise with $Q = var(\omega_k) = 1$. The formula for α is discussed in Section 4.7 of [1]. The formula for γ and $\bar{P}$ are discussed in the next paragraph.

Define the symbol $P_k = cov(x_k)$. The state transition model for P_k can be derived using eqn. (3):

$$\begin{aligned} P_{k+1} &= E\langle x_{k+1}, x_{k+1}\rangle, &(4)\\ &= E\langle(\alpha x_k + \gamma\omega_k)(\alpha x_k + \gamma\omega_k)\rangle &(5)\\ &= E\langle \alpha^2 x_k^2 + 2\gamma\alpha\,\omega_k x_k + \gamma^2\omega_k^2\rangle &(6)\\ &= \alpha^2 P_k + \gamma^2 Q. &(7)\end{aligned}$$

In steady-state, $\bar{P} = P_k = P_{k+1}$. Therefore,

$$\begin{aligned} \bar{P} &= \alpha^2 \bar{P} + \gamma^2 Q &(8)\\ \bar{P}(1-\alpha^2) &= \gamma^2 Q &(9)\\ \gamma^2 &= \bar{P}(1-\alpha^2)/Q. &(10)\end{aligned}$$

The result after eqn. (3) uses the fact that $Q = 1$.

The user defines the parameters T_c and $\bar{P}$. PTV Vissim determines T. The software then computes α and γ and implements the model of eqn. (3) for each component of the position error for each vehicle at each measurement epoch.

3 Correlation Analysis

The experimental data will be analyzed using correlation methods. This section derives analytic results that will be used in Section 4. The correlation sequence for a discrete-time signal $y_k \in \Re$ is defined as $R_y(n) = E\langle y_k, y_{k+n}\rangle$.

3.1 Simulation Model Correlation Analysis: x_k

For the model of eqn. (3), the correlation sequence satisfies:

$$\begin{aligned} R_{x_i}(n) &= E\langle x_i(k), x_i(k+n)\rangle\\ &= E\langle x_i(k), \alpha x_i(k+n-1) + \gamma\omega_i(k+n-1)\rangle\\ &= \alpha E\langle x_i(k), x_i(k+n-1)\rangle\\ &\ldots\ldots\\ &= \alpha^n E\langle x_i(k), x_i(k)\rangle\\ R_x(n) &= \alpha^n \bar{P}_i, &(11)\end{aligned}$$

with $R_{x_i}(0) = \bar{P}_i$ because $x_i(k)$ is zero mean.

3.2 Correlation Analysis: $\delta\mathbf{p}_i$

In the following analysis, the symbol $\delta\mathbf{p}_i(k)$ is used as a short-hand notation for $[\delta\mathbf{p}(kT)]_i$.

For the model of eqn. (2), the correlation sequence satisfies:

$$\begin{aligned} R_{\delta\mathbf{p}_i}(n) &= E\langle \delta\mathbf{p}_i(k), \delta\mathbf{p}_i(k+n)\rangle\\ &= E\langle x_i(k) + \mu_i + \eta_i(k), x_i(k+n) + \mu_i + \eta_i(k+n)\rangle\\ &= E\langle x_i(k), x_i(k+n)\rangle + \mu_i^2 + E\langle \eta_i(k), \eta_i(k+n)\rangle\\ &= R_{x_i}(n) + \mu_i^2 + \sigma^2_{\eta_i}\delta_n &(12)\end{aligned}$$

where δ_n represents the Kronecker impulse function which has a value of one for $n = 0$ and is otherwise zero.

Substituting the result from eqn. (11) into eqn. (12), yields:

$$R_{\delta\mathbf{p}_i}(n) = \alpha^n \bar{P}_i + \mu_i^2 + \sigma^2_{\eta_i}\delta_n. \tag{13}$$

4 Experimental Data Analysis

This section assesses the period of validity of the model in Section 2.2 using GNSS data described in Sec. 4.1.

4.1 GNSS Positioning Error Data Description

The GNSS positioning data[2] was collected on a moving platform experiment on July 2, 2021. The duration of the data accumulation experiment was about 1 hr. The data was collected with a sampling period of $T_s = 1s$. During the experiment, two antennae were mounted on the roof of and two receivers were mounted in the a car that was driven in an urban area near UCR that had a clear view of the sky. One antenna-receiver pair was used to determine the time-varying *ground truth* vehicle position with centimeter accuracy during the experiment. This antenna was a dual-band u-blox antenna that was connected to a u-blox M8P receiver that was communicating with the UCR base station and performing carrier-phase, integer-fixed real-time kinematic positioning (RTK) [2–5]. The second antenna-receiver pair was used for analysis, of the accuracy of the time-varying vehicle position as measured by a standard commercial-off-the-shelf receiver that did not have access to carrier differential measurements. This antenna was a dual-band ublox antenna that was connected to an u-blox ZED-F9P receiver that was connected to a VN-DGNSS server to obtain RTCM code measurement corrections [6]. Only single-frequency L1/E1/B1 pseudo range measurements are used in the following analysis. For both receivers, GPS, GALILEO and BeiDou are enabled in both receivers.

4.2 GNSS Position Error Computation Method

At time $t = kT$, referred to as the k-th measurement epoch, the symbol $\mathbf{p}_k$ denotes the ground truth position reported by receiver 1; and, $\hat{\mathbf{p}}_k$ denotes the position estimated by receiver 2. Both ublox receivers report the position measurements in the ECEF frame. A post superscript will be used to indicate the frame-of-reference: $\mathbf{p}_k^e$ for ECEF and $\mathbf{p}_k^g$ for local tangent frame. In both frames, the GNSS positioning error vector is computed as

$$\delta\mathbf{p} = \mathbf{p} - \hat{\mathbf{p}}. \tag{14}$$

Vectors are transformed between frames using the equation: $\delta\mathbf{p}^g = \mathbf{R}_e^g\,\delta\mathbf{p}^e$, where the symbol $\mathbf{R}_e^g$ represents the rotation matrix from the ECEF frame to the local NED frame (see eqn. (2.34) in [1]). In the local tangent frame, the coordinates of $\mathbf{p}^g$, $\hat{\mathbf{p}}^g$, and $\delta\mathbf{p}^g$ are indicated by N, E, and D subscripts.

4.3 Experimental Correlation Graphs

Before computing the correlation sequence for the experimental data, the experimental mean ($\delta\bar{\mathbf{p}}^g = \frac{1}{N}\sum_{k=1}^{N}\delta\bar{\mathbf{p}}_k^g$) of the data was subtracted from the position error vector at each time. The mean value pf the postion error vector for this data set is

$$\delta\bar{\mathbf{p}}^g = \begin{bmatrix}0.074 & -0.106 & 1.172\end{bmatrix}^\top.$$

The correlation sequences for the (zero mean) north, east, and vertical GNSS position error components are shown in Figs. 1a-1c as the blue dots.

4.4 Correlation Model Parameter Fitting

For each component of the (zero mean) position error vector, the following text describes an approach to determine the model parameters $\bar{P}$ and α:

- Step 1: Compute the zero mean positioning error data:

  $$\delta\tilde{\mathbf{p}}_i = \delta\mathbf{p}_i - \mu_i,\ \text{where}\ \mu_i = mean(\delta\mathbf{p})_i.$$

  The subscript i denotes the i-th component. The mean is recorded in the first column of Table 1.

- Step 2: Use the method described in the Appendix entitled "First-order Markov Process: System Identification" to determine the parameters α, $\bar{\mathbf{P}}$, γ, and σ_η (called μ, $\mathbf{P}_0$, σ_ω, and σ_η in the appendix) such that the correlation model of eqn. (13) optimally fits the correlation sequence for each component $\delta\tilde{\mathbf{p}}_i$ of the experimental data for lags $n \in [0, 60]$.

From these and the definition of α following eqn. (3), $T_c = -T/\ln(\alpha)$. For the results that follow, $T = 1$ sec.

Table 1: Estimated model parameter for the data set described in Section 4.

	μ, m	α	σ_η, m	$\bar{\mathbf{P}}, m^2$	T_c, s
North	0.41	0.9849	0.00	0.08	65.60
East	-0.09	0.9922	0.03	0.08	128.51
Down	1.49	0.9896	0.02	0.18	95.31

The red dots in Figs. 1a-1c show the graph of the correlation model for the parameters contained in Table 1 for $n \in [0, 60]$. The curve for the correlation model of eqn. (13) using the parameter values summarized in Table 1 fit the experimental correlation data reasonably well for about the first 60 seconds.

Longer durations of model validity could be achieved, but would require higher order models (i.e., additional simulation computation). Since one instantiation of eqn. (3) is required for each of the three position error components for each simulated vehicle, the additional computations are not deemed necessary.

4.5 Conclusions, Discussion, and Caveats

The experimental results should be interpreted with care. The main conclusions are that the position errors are correlated through time, with the correlation decreasing as a function of time over the first several tens of seconds. Over this time period, these characteristics can be well modeled using Gauss-Markov discrete-time state-space models. First-order models perform well over several tens of seconds.

The specific model parameters in Table 1 result from one experiment on a moving car in a specific environment using DGNSS. They should not be interpreted as correct or best for any situation. They may be interpreted as reasonable for similar situations. For

[2] The data is available in the *Lane-Level-Localization/GNSS_Basic_Error_Model/simulation/* folder at `https://github.com/Azurehappen/Lane-Level-Localization`.

stationary vehicles or GNSS OS, both T_c and $\bar{P}$ should be significantly increased.

The values of the parameter μ in Table 1 are the mean of the experimental data. Such a mean is not accounted for in the correlation model of eqn. (11) corresponding to the GNSS position error model of eqn. (3) as implemented in PTV Vissim. To adjust for this, the α and $\bar{P}$ parameters should be adjusted to increase the correlation over the time period of interest. For N and E, the adjustment is small, because μ^2 is small relative to $\bar{P}$. For D, $\mu^2 \gg \bar{P}$; therefore, if the user is interested in the vertical error component, the α and $\bar{P}$ parameters may need to be significantly increased.

5 Ideas for Future Work

The first-order Gauss-Markov models described herein and implemented in PTV Vissim achieve the goal of accounting for GNSS time-correlated position errors. They do not account for various other factors:

- GNSS position errors at any given time will have correlation between the position error components. This means that the position error covariance matrix

$$\mathbf{P} = cov(\delta \mathbf{p}) = E\langle \delta \mathbf{p}\, \delta \mathbf{p}^\top \rangle$$

  is a positive definite and non-diagonal matrix. The models described in this document will always have **P** being positive definite, but diagonal. The Gauss-Markov state-space model could be extended to include this inter-component correlation if desired.

- The error covariance matrix **P** for GNSS position errors will be time-varying with its specific structure determined by the recent history of the constellation of satellites used to compute the position. The error covariance matrix produced by the simulation model described herein will be diagonal with each diagonal element being the constant $\bar{P}$. The model could be improved to include a constellation of satellites that determine the position error sequence in a manner that results in a non-diagonal covariance matrix.

The tradeoff in any of these changes is that increased model fidelity will cost additional computation during simulation. The authors are interested in collaborative efforts to enhance the model. Please feel free to contact the last author.

6 Acknowledgment

This model was implemented as part of the "Lane-Level Localization and Map Matching for Advanced Connected and Automated Vehicle (CAV) Applications" project funded by the National Center for Sustainable Transportation (NCST) during the period of performance from April 1, 2021 through March 31, 2022.

References

[1] J. A. Farrell, *Aided Navigation: GPS with High Rate Sensors*. McGraw-Hill, Inc., 2008.

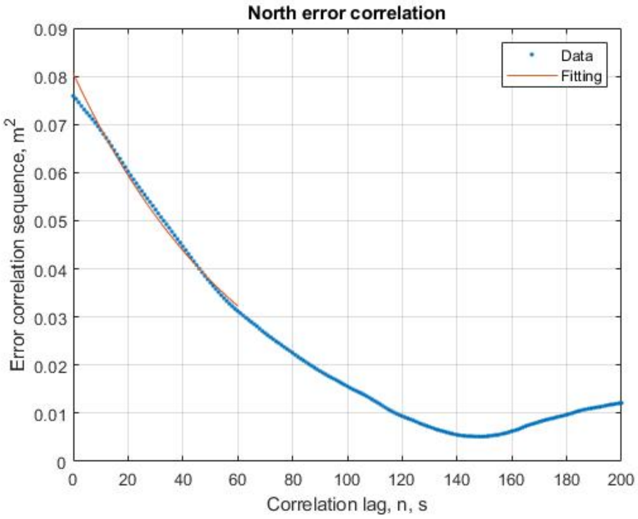
[2] R. Hatch, "The synergism of GPS code and carrier measurements," *International geodetic sym-posium on satellite Doppler positioning*, vol. 2, pp. 1213–1231, 1983.

[3] P. Hwang, "Kinematic GPS for differential postioning: Resolving integer ambiguities on the fly," *Navigation: J. Inst. Nav.*, vol. 38, 1991.

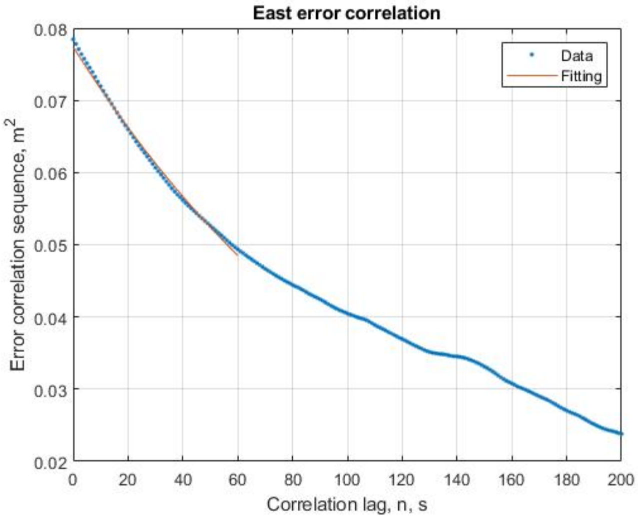
[4] P. Teunissen, "A new method for fast carrier phase ambiguity estimation," *IEEE PLANS*, p. 562–573, 1994.

[5] J. A. Farrell, T. D. Givargis, and M. J. Barth, "Real-time differential carrier phase GPS-aided INS," *IEEE Transactions on Control Systems Technology*, vol. 8, no. 4, pp. 709–721, 2000.

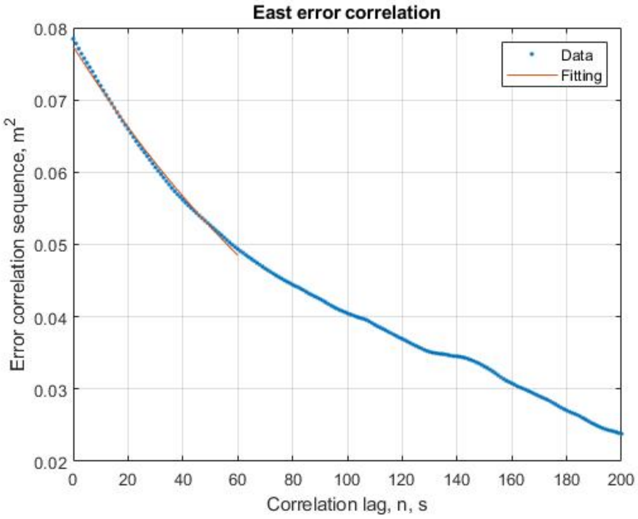
[6] W. Hu, A. Neupane, and J. A. Farrell, "Using PPP Information to Implement a Global Real-Time Virtual Network DGNSS Approach," *Under review. See https://arxiv.org/abs/2110.14763*, 2021.



(a) Correlation sequence for North GNSS position error.



(b) Correlation sequence for East GNSS position error.



(c) Correlation sequence for Down GNSS position error.

Appendix – First-order Markov Process: System Identification

We have available data $\mathbf{U} = \{u_i\}_{i=1}^N$ where each u_i is modeled as

$$\begin{aligned} u_i &= b_i + \eta_i, \text{ where } \eta_i \sim N(0, \sigma_\eta^2) \text{ is a white sequence and} && (15)\\ b_{i+1} &= \mu\, b_i + \omega_i, \text{ where } \omega_i \sim N(0, \sigma_\omega^2) \text{ is a white sequence.} && (16) \end{aligned}$$

The process u_i is assumed to be wide sense stationary (i.e., all its moments are time invariant) and η_i, ω_j, and $b(0)$ are assumed to be independent Gaussian random variables. The parameter μ has magnitude less than one. The goal is to estimate μ, $C_b(0)$, σ_η and σ_ω.

First, analyze the covariance/correlation of $\mathbf{U}$:

$$\begin{aligned} C_u(j) &= E\langle u_i u_{i+j}\rangle = E\langle (b_i + \eta_i)(b_{i+j} + \eta_{i+j})\rangle; \text{ therefore,} \\ C_u(0) &= E\langle (b_i^2 + 2\eta_i b_i + \eta_i^2)\rangle = C_b(0) + \sigma_\eta^2 && (17)\\ C_u(1) &= E\langle (b_i b_{i+1} + \eta_i b_{i+1} + \eta_{i+1} b_i + \eta_i \eta_{i+1})\rangle = C_b(1) && (18)\\ &\vdots \quad \vdots \\ C_u(j) &= C_b(j). && (19) \end{aligned}$$

Next, analyze the steady state covariance b_i. Let $P_0 = C_b(0) = E\langle b_i b_i\rangle$. Because the system is in steady state (i.e., WSS),

$$P_0 = E\langle b_i b_i\rangle = E\langle b_{i+1} b_{i+1}\rangle, \forall i.$$

Based on eqn. (16),

$$\begin{aligned} E\langle b_{i+1} b_{i+1}\rangle &= E\langle (\mu\, b_i + \omega_i)(\mu\, b_i + \omega_i)\rangle \\ &= \mu^2 E\langle b_i^2\rangle + 2\mu E\langle b_i \omega_i\rangle + E\langle \omega_i^2\rangle \\ P_0 &= \mu^2 P_0 + \sigma_\omega^2. && (20) \end{aligned}$$

Finally, analyze the covariance/correlation sequence for b_i:

$$\begin{aligned} C_b(1) &= E\langle b_i b_{i+1}\rangle = E\langle b_i(\mu\, b_i + \omega_i)\rangle = \mu P_0 && (21)\\ C_b(2) &= E\langle b_i b_{i+2}\rangle = E\langle b_i(\mu\, b_{i+1} + \omega_{i+1})\rangle = \mu C_b(1) = \mu^2 P_0 \\ &\vdots \quad \vdots \\ C_b(j) &= \mu^j P_0. && (22) \end{aligned}$$

Substituting eqns (21-22) into (17-19), and replacing $C_u(j)$ with their values computed from $\mathbf{U}$ yields:

$$\begin{aligned} \hat{C}_u(0) &= P_0 + \sigma_\eta^2 && (23)\\ \hat{C}_u(1) &= \mu P_0 && (24)\\ \hat{C}_u(2) &= \mu^2 P_0 && (25)\\ &\vdots \quad \vdots \\ \hat{C}_u(j) &= \mu^j P_0. && (26) \end{aligned}$$

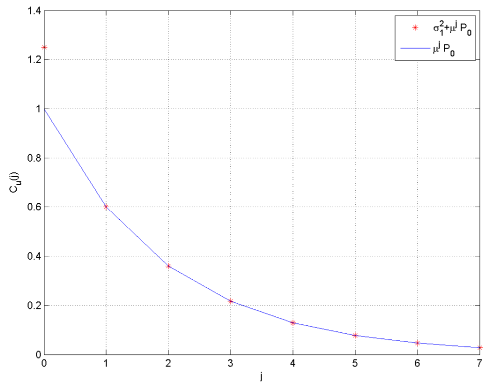


Figure 2: Plot of C_u for Section 6.

These equations can be solved as follows:

1. For $j \geq 1$, using eqns. (24-26), estimate $a = \ln(\mu)$ and $b = \ln(P_0)$ from the linear relationship: $\ln(\hat{C}_u(j)) = ja + b$.
2. Given $\hat{a}$ and $\hat{b}$ from Step 1, solve for $\hat{\mu} = \exp(\hat{a})$ and $\hat{P}_0 = \exp(\hat{b})$.
3. Compute $\hat{\sigma}_\omega^2 = (1 - \hat{\mu}^2)\hat{P}_0$ from eqn. (20).
4. If $\hat{C}_u(0) \geq \hat{P}_0$, compute $\hat{\sigma}_\eta^2 = \hat{C}_u(0) - \hat{P}_0$ from eqn. (23); otherwise, $\hat{\sigma}_\eta^2 = 0$.

Appendix C – Queue Length Estimation

The following describes the lane queue estimation algorithm. The lead designer of this algorithm was Dr. Peng Hao, who worked in collaboration with Guoyuan Wu and J. A. Farrell to write the following algorithm description.

Peng Hao, Guoyuan Wu, J. A. Farrell
University of California, Riverside

Problem Statement

This document is concerned with lane-level maximum queue length estimation at some future time T >= t, where t is the current time. For the problem of estimating the expected queue length at the present time t, only the two terms Q1 and Q2 (defined later in this document) are used.

Comparison with the Literature

Several articles in the literature consider the problem of maximum queue length estimation after all the vehicles in the cycle have passed the intersection [Hao et al., 2013; Hao et al., 2014]. The problem considered herein is different. This document aims to predict the maximum queue length in a cycle when some vehicles are still on their ways to approach the intersection.

Definitions:

The following is a list of parameters and variables used in this document. Various of the parameters and variables are illustrated in Figure B1.

- Length of the detection zone (Ld): The distance between an upstream location (defined as the boundary of the detection zone) and the stop-bar along an ingress approach.
- The set of vehicles in the intersection detection zone has two subsets, those that are *detected* and those that are *not detected*.
  - The detected vehicles are also referred to as *probe vehicles* or *probes.* Probe vehicles have an estimate of their position and velocity, which they communicate to the infrastructure and to other vehicles. All CV's are assumed to be detected.
  - Undetected vehicles do not communicate with the infrastructure or with other vehicles.
- The set of vehicles will be divided into the following three categories.
  - Queued Vehicles: Detected or not detected vehicles that are stopped in front of the last detected queued vehicle at the current time. The last detected queued vehicle is included.
  - Arrival Flow Vehicles: Vehicles that have arrived in the detection zone following the last detected queued vehicle and before the current time t.

  - Future Flow Vehicles: Vehicles that may arrive after the current time.

  Arrival and future flow vehicles are currently in free flow, but may transition to queued in the future.

- Future Deceleration rate (a): rate at which detected vehicles decelerate in the detection zone.
- Average departure headway (h): The time gap between each vehicle passing the stop line (seconds per vehicle).
- Jam density (k): The average number of vehicles per unit of distance per lane in the queue.
- Penetration Rate (p): Expected ratio of probe vehicles to the total number of vehicles.
- Stop location (L_s): The stop position relative to the stop-bar for the last detected vehicle in the queue. This is positive for vehicles approaching the stop-bar and negative for vehicles that have passed the stop-bar.
- M: The maximum number of vehicles that can fit in a lane of length Ld at the assumed jam density k (i.e., M = Ld k).
- Free-flow speed (Vf): Assumed velocity of vehicles that are not decelerating.
- Current time (t): The time at which the queue length is computed.
- T' is the time at which the free flow line of the last detected queued vehicle would intersect the stop line. The free flow line is extended from before the time that the vehicle became queued (see Vehicle A in Figure B1). If there is no detected arriving or queued vehicle, then T' is the time that the signal becomes red T_R.
- Time at which a free-flow vehicle will pass the stop bar if it enters the detection zone at time t (T=t+Ld/Vf)
- Estimated discharge time of the last queued vehicle (T_{Q_1}).
- Time the lane signal turns green (T_G).
- Time the lane signal turns red (T_R).
- Vehicle arrival rate (λ_0): The average number of arriving vehicles per unit of time per lane.
- Queue index: The queue position (in number) of a queued vehicle of a specific lane
- Q_1: Total number of queued vehicles at time t, i.e., the index of the last queued vehicle (see Part 1).
- $E[Q_2]$: The expected number of arrival flow vehicles that will become queued vehicles (see Part 2).
- $E[Q_3]$: The expected number of vehicles outside Part 1 and Part 2 that will be queued within the current cycle.

The parameters that affect the performance of the algorithm are: h, p, k, and λ_0. The variables Q_1, $E[Q_2]$, and $E[Q_3]$ are computed by the algorithm.

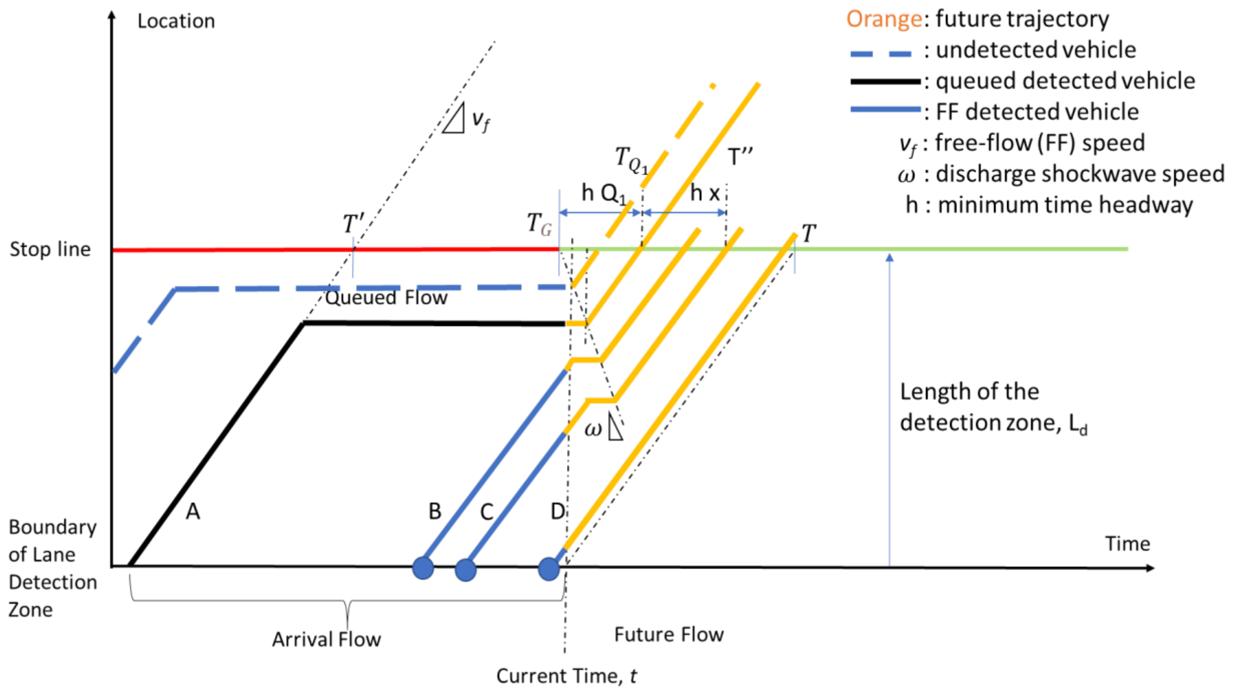


Figure B1. Illustration of variables and parameters for one lane of an intersection approach. Vehicle A is stopped at time t. Vehicles B, C, D are detected, but not queued and not decelerating. Q1 is computed only using only using the position of A and is at least 1. For the value of x assumed in this figure, n(x) = 2.

Assumptions

This document used the following assumptions:

1. All lanes in an intersection approach have the same value of Ld.
2. All vehicles have the same type, size and shape, which is known.
3. Vehicle position means the center of the vehicle.
4. The position of the GNSS antenna from the vehicle center is known, so that the vehicle position can be computed from the GNSS position and heading.
5. The average headway (h), jam density (k), penetration rate (p), free-flow speed (Vf), and deceleration (a) are known and the same for all vehicles.
6. Vehicle arrivals at the boundary of the detection zone are described by a Poisson process with average rate λ_0 (# of vehicles per unit time). The arrival rate parameter (λ_0) is known.
7. No lane changes occur within the observation zone.

Later these assumptions can be relaxed.

Part 1: Queued Detected Vehicles Flow (Vehicle A in Figure)

This section focuses on detected vehicles that are either already stopped or that are decelerating with the expectation that they will be stopped. In Figure B1, vehicle A is the last detected vehicle in the lane queue. In Figure B1, the black solid line shows the stop location as a function of time for the last detected vehicle (vehicle A).

The goal of this section is to compute $Q_1(t)$. The formula is

$$Q_1(t) = \lfloor k \cdot \ L_s \rfloor + 1, \qquad \text{(A1)}$$

where k is the jam density, L_s is the stop position relative to the stop bar for the last detected vehicle in the queue, and $\lfloor \cdot \rfloor$ represents the floor function. Note that Q1 is an integer. The computation needs to consider two possible situations.

Stopped Vehicles (Vehicle A at time t in Figure): This case only applies if there are no detected decelerating probe vehicles and there is at least one queued probe (i.e., stopped). In this case, the calculation applies only to the last detected vehicle. (i.e., vehicle A in Figure B1). From the stop location of the last detected vehicle, $Q_1(t)$ can be directly computed:

$$L_s = L_c, \qquad \text{(A2)}$$

where L_c is the current distance of the last stopped vehicle in the lane queue to the stop-bar. Therefore, Q1 can be computed using eqn. (A1).

Decelerated vehicles: If there is at least one Decelerating Probe vehicle (a<0), then the one with the most positive value of L_s will be the last one (i.e., furthest from the stop bar), coming to a stop behind vehicle A. The predicted stop location L_s of the decelerating vehicle is

$$L_s = L_c - \frac{v^2}{2a} \qquad \text{(A3)}$$

which can be used in eqn. (A1). In this equation, L_c is the current location, v is the current speed, and a is the deceleration rate.

Part 2: Arriving Flow (Vehicles B, C, D)

This section accounts for the contribution to the queue from arrival flow vehicles, whether detected or undetected. In Figure B1, at the current time t vehicles A, B, C, D are detected, and B, C, D are in the arrival flow. There may be additional undetected arrival flow vehicles.

The additional queue length, denoted as $E[Q_2]$, that is computed in this section is the number of vehicles expected to arrive at the stop line between max(T', T_R) and min(T_{Q_1}, T). All symbols are defined in the Definitions section. This includes detected free flow vehicles and undetected vehicles.

If $Q_1(t) = 0$ (there is no Decelerating/Queued Probe), then T'(t) = T_R and $T_{Q_1}(t) = T_G$.

If $Q_1(t)$ is greater than zero, then T' is as defined in the Definitions Section. The last detected queued vehicle's departure time (i.e., the time to clear the known queue) is predicted as

$$T_{Q_1}(t) = T_G + h\ Q_1(t) \tag{B1}$$

where the queue discharges at the stop line, h is the average headway, and T_G is the time of the start of the green.

If the penetration rate was 100%, then all vehicles would be detected. When the penetration rate is less than 100%, then there may be undetected vehicles in the queue and additional undetected vehicles may arrive. The theory to accommodate vehicles that are expected to arrive is as follows. Let x denote the number of undetected vehicles in the queue after the last detected vehicle in the queue (i.e., vehicle A). The time to clear the queue is

$$T(x)'' = \min\ (T_G + h(Q_1 + x), T) \tag{1}$$

In this expression, $T_G + h(Q_1 + x)$ is the expected departure time of the last queued vehicle in the Arrival Flow, and T is defined in the Definitions Section.

For each value of *x* from 0, …, M:

1. Compute *T(x)''* using eqn. (1).
2. Compute the number of known probe vehicles (e.g., B, C, D) predicted to pass the stop bar between T' and T''. This number is n for this x (i.e., n(x)).

   The probability mass function for $Q_2 = x$ is

   $$P(Q_2 = x) = \beta\ P_{Po}(x; \lambda)\ P_B(n|x; p)$$

   where β is a normalization factor, $P_{Po}(x; \lambda)$ is the Poisson distribution for *x* given rate $\lambda = \lambda_0(T'' - T')$

   $$P_{Po}(x, \lambda) = \frac{\lambda^x e^{-\lambda}}{k!};$$

   $P_B(x, n)$ is the probability of n probes out of x given penetration rate *p* (Binomial distribution),

   $$P_B(n|x; p) = \begin{cases} \binom{x}{n} p^n (1-p)^{x-n}, for\ x \geq n \\ 0\ , otherwise \end{cases}.$$

   Therefore, the expectation of Q_2 is

   $$E[Q_2] = \frac{\sum_{x=0}^{M} x\ P(Q_2=x)}{\sum_{x=0}^{M} P(Q_2=x)}\ . \tag{2}$$

In this expression, M is the maximum number of vehicles that can fit in the lane of length Ld at the assumed jam density k (i.e., M = Ld k).

Part 3: Future Flow

Based on the results from Part 2, if $T_G + h(Q_1 + Q_2)$ is later than the T defined in the Definitions Section, then traffic may arrive after the current time and may also contribute to the queue. As there is no detected vehicle information here from which to infer the queue length, the algorithm uses the expected value of the vehicles in this time interval to approximate the additional expected queue length Q_3 in this part:

$$E[Q_3] = \max\left(0, \lambda_0[T_G + h(Q_1 + Q_2) - T]\right).$$

The Complete Answer

The queue length is then $Q_1 + E[Q_2] + E[Q_3]$

References

Hao, P., Ban, X., Guo, D., & Ji, Q., (2014). Cycle-by-cycle queue length distribution estimation using sample travel times. *Transportation Research Part B*, 68(1) 185-204.

Hao, P., Sun, Z., Ban, X., Guo, D., & Ji, Q. (2013). Vehicle index estimation for signalized intersections using sample travel times. *Transportation Research Part C*, 36(1), 513-529.