



Technical Report 139

ADAS Enhanced by 5G Connectivity: Volumes 1 and 2

Research Supervisor: Todd E. Humphreys
Wireless Networking and Communications Group

September 2018

Data-Supported Transportation Operations & Planning Center (D-STOP)

A Tier 1 USDOT University Transportation Center at The University of Texas at Austin



**CENTER FOR
TRANSPORTATION
RESEARCH**



**Wireless Networking &
Communications Group**

D-STOP is a collaborative initiative by researchers at the Center for Transportation Research and the Wireless Networking and Communications Group at The University of Texas at Austin.

Technical Report Documentation Page

1. Report No. D-STOP/2018/139		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle ADAS Enhanced by 5G Connectivity: Volumes 1 and 2				5. Report Date September 2018	
				6. Performing Organization Code	
7. Author(s) Volume 1: Lakshay Narula, Michael J. Wooten, Matthew J. Murrian, Daniel M. LaChapelle, Todd E. Humphreys Volume 2: Lakshay Narula, Matthew J. Murrian, Todd E. Humphreys				8. Performing Organization Report No. Report 139	
9. Performing Organization Name and Address Data-Supported Transportation Operations & Planning Center (D-STOP) The University of Texas at Austin 3925 W. Braker Lane, 4th Floor Austin, Texas 78759				10. Work Unit No. (TRAIS)	
				11. Contract or Grant No. DTRT13-G-UTC58	
12. Sponsoring Agency Name and Address Data-Supported Transportation Operations & Planning Center (D-STOP) The University of Texas at Austin 3925 W. Braker Lane, 4th Floor Austin, Texas 78759				13. Type of Report and Period Covered	
				14. Sponsoring Agency Code	
15. Supplementary Notes Supported by a grant from the U.S. Department of Transportation, University Transportation Centers Program. Project Title: ADAS Enhanced by 5G Connectivity					
16. Abstract Volume 1: This paper explores the limit of digital maps' globally-referenced position accuracy when the mapping agents are equipped with low-cost Global Navigation Satellite System (GNSS) receivers performing standard code-phase-based navigation, and presents a globally-referenced electro-optical simultaneous localization and mapping pipeline, called GEOSLAM, designed to achieve this limit. The key accuracy-limiting factor is shown to be the asymptotic average of the error sources that impair standard GNSS positioning. Asymptotic statistics of each GNSS error source are analyzed through both simulation and empirical data to show that sub-50-cm accurate digital mapping is feasible in the horizontal plane after multiple mapping sessions with standard GNSS, but larger biases persist in the vertical direction. GEOSLAM achieves this accuracy by (i) incorporating standard GNSS position estimates in the visual SLAM framework, (ii) merging digital maps from multiple mapping sessions, and (iii) jointly optimizing structure and motion with respect to time-separated GNSS measurements. Volume 2: A system developed for low-cost precise urban vehicular positioning is demonstrated to achieve a probability of correct integer fixing greater than 96.5% for a probability of incorrect integer fixing surely less than 2.3% and likely less than 1%. The results are achieved without any aiding by inertial or electro-optical sensors. Development and evaluation of the unaided GNSS-based precise positioning system is a key milestone toward the overall goal of combining precise GNSS, vision, radar, and inertial sensing for all-weather high-integrity precise positioning for automated and connected vehicles. All components have been tailored in their design to yield competent sub-decimeter positioning in the mobile urban environment. A performance sensitivity analysis reveals that navigation data bit prediction on the GPS L1 C/A signals is key to high-performance urban real-time kinematic (RTK) positioning.					
17. Key Words urban vehicular positioning; CDGNSS; low-cost RTK positioning, vehicle localization; SLAM; sensor fusion			18. Distribution Statement No restrictions. This document is available to the public through NTIS (http://www.ntis.gov): National Technical Information Service 5285 Port Royal Road Springfield, Virginia 22161		
19. Security Classif.(of this report) Unclassified		20. Security Classif.(of this page) Unclassified		21. No. of Pages 58	22. Price

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 under the sponsorship of the U.S. Department of Transportation's University Transportation Centers Program, in the interest of information exchange. The U.S. Government assumes no liability for the contents or use thereof.

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. Mention of trade names or commercial products does not constitute endorsement or recommendation for use.

Acknowledgements

The authors recognize that support for this research was provided by a grant from the U.S. Department of Transportation, University Transportation Centers.



Preface

D-STOP project 139, *ADAS Enhanced by 5G Connectivity*, began in September 2017 and made great strides in foundational work before it ended in September 2018.

Advanced driver assistance systems (ADAS) are a key technology for improved traffic safety. Long before fully automated vehicles arrive in significant numbers, ADAS will see high penetration and substantially reduce accident rates. Toyota and Honda have both committed to focusing on “hands on the wheel, eyes on the road” ADAS long before (perhaps up to a decade) introducing higher levels of automation to consumers. This is a philosophy that resonates with the research team on this project.

Connectivity between vehicles, and between vehicles and infrastructure, makes ADAS more effective by enabling vehicles to “see” around corners and through other vehicles. But connectivity via DSRC, the 802.11-based standard that will likely be mandated by 2020, can become congested when a large number of vehicles, cyclists, and pedestrians congregate near intersections in urban areas. Moreover, DSRC does not offer the bandwidth for sharing of raw, or lightly-processed, sensor data between vehicles or from infrastructure to vehicles. In fact, in all likelihood, DSRC message traffic will be limited to the basic safety message, a low-rate, low-latency message that communicates a vehicle’s or cyclist’s or pedestrian’s current position and velocity to others in the vicinity. And even this message will become unreliable if too many DSRC transmitters find themselves fighting for slots in which to transmit, such as will occur in urban areas with heavy foot and vehicular traffic.

This project studied how emerging 5G technology can be used to “supercharge” ADAS by releasing it from the limitations of DSRC. We investigated how can ADAS benefit from the sub-10-ms latency, the 100 Mbps per-user download data rate, and the high connection density that 5G promises.


To provide our results, we present here two papers. These papers build a foundation for sharing of data, which underlies our unique approach to ADAS. The first, *Accurate Collaborative Globally-Referenced Digital Mapping with Standard GNSS*, explores our approach to ADAS in which we enable exchange of data between vehicles. The second, *Low-cost Precise Vehicular Positioning in Urban Environments*, describes a system for low-cost precise urban vehicular positioning we developed. The system is composed of a densely-spaced reference network, a software-defined GNSS receiver whose processing can be executed on general-purpose commodity hardware, and a real-time kinematic (RTK) positioning engine. A performance sensitivity analysis reveals that navigation data bit prediction on the GPS L1 C/A signals is key to high-performance urban RTK positioning that will further ADAS in the future.

*Volume 1: Accurate Collaborative Globally-Referenced
Digital Mapping with Standard GNSS*



Article

Accurate Collaborative Globally-Referenced Digital Mapping with Standard GNSS

Lakshay Narula ^{1,*} , J. Michael Wooten ², Matthew J. Murrian ², Daniel M. LaChapelle ² and Todd E. Humphreys ²

¹ Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712, USA

² Department of Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin, Austin, TX 78712, USA; jmwooten@utexas.edu (J.M.W.); matthew.murrian@utexas.edu (M.J.M.); daniel.m.lachapelle@utexas.edu (D.M.L.); todd.humphreys@mail.utexas.edu (T.E.H.)

* Correspondence: lakshay.narula@utexas.edu

Received: 16 June 2018; Accepted: 26 July 2018; Published: 28 July 2018



Abstract: Exchange of location and sensor data among connected and automated vehicles will demand accurate global referencing of the digital maps currently being developed to aid positioning for automated driving. This paper explores the limit of such maps' globally-referenced position accuracy when the mapping agents are equipped with low-cost Global Navigation Satellite System (GNSS) receivers performing standard code-phase-based navigation, and presents a globally-referenced electro-optical simultaneous localization and mapping pipeline, called GEOSLAM, designed to achieve this limit. The key accuracy-limiting factor is shown to be the asymptotic average of the error sources that impair standard GNSS positioning. Asymptotic statistics of each GNSS error source are analyzed through both simulation and empirical data to show that sub-50-cm accurate digital mapping is feasible in the horizontal plane after multiple mapping sessions with standard GNSS, but larger biases persist in the vertical direction. GEOSLAM achieves this accuracy by (i) incorporating standard GNSS position estimates in the visual SLAM framework, (ii) merging digital maps from multiple mapping sessions, and (iii) jointly optimizing structure and motion with respect to time-separated GNSS measurements.

Keywords: vehicle localization; SLAM; sensor fusion

1. Introduction

Localization is one of the primary operations that connected and automated vehicles must perform, both to navigate from one location to another and to interact with each other and with their surroundings within a mapped environment. Satellite-based navigation sensors have historically been the unrivalled sensor of choice for navigating from source to destination. However, the high-reliability sub-50-cm precision demanded by automated vehicles for lane-keeping and other applications, especially in urban areas, has significantly changed this landscape [1]. In most automated vehicles being developed, the Global Positioning System (GPS)/Global Navigation Satellite System (GNSS) is relegated to a secondary sensor whose role is to loosely constrain (within a few meters) the primary localization sensors, usually camera(s) and/or LiDAR, to a global reference frame when building a digital map. The vehicles then locate themselves to decimeter accuracy within this digital map.

Automated driving does not necessarily demand sub-50-cm agreement between the coordinates of a given point in the digital map and the coordinates of the same point in a well-defined global

reference frame. Rather, local self-consistency and accurate localization within the digital map is of greater importance. However, consistency of the digital map with a global coordinate frame is likely to become a pre-requisite for cooperative automated driving. Automated driving can be made more efficient (e.g., by platooning) and safe if basic safety information such as vehicle location, velocity, intent, etc. are shared among neighboring agents through vehicle-to-vehicle (V2V) and/or vehicle-to-everything (V2X) communication [2,3]. In many cases, such information can be perceived independently by each agent via its exteroceptive sensors. But in other situations—for example at a blind corner or during a left turn maneuver—full situational awareness will require external sensors—on other vehicles or on infrastructure—that wirelessly communicate data to the ego-vehicle. If all collaborating vehicles navigate within the same digital map, exchange of information can be performed with sub-50-cm precision [4,5], even if the map itself is only globally accurate to a few meters. However, it is unlikely that automated vehicles from different manufacturers will rely on a common digital map. Consequently, the precision of the exchanged vehicle position is lower-bounded by the disagreement on the coordinates of the same physical location between different maps. Thus, exchange of accurate vehicle pose among vehicles, as well as other associated high-level information such as sensor data in the vehicle's body frame, will demand consistency among, or translation between, different digital maps.

Standard code-phase-based GNSS position measurements, such as those provided by all mass-market GNSS receivers, may be biased by as much as 3–5 meters on any given mapping session. Maps anchored by these measurements may not exhibit lane-level consistency with each other. One possible solution is to create digital maps with decimeter-accurate carrier-phase differential GNSS (CDGNSS) systems [6]. However, at current prices, such systems can only be installed on a limited fleet of specialized mapping vehicles. Precise point positioning (PPP) techniques offer a low-cost alternative to CDGNSS, but the frequent cycle-slipping experienced in urban areas impedes the convergence of PPP techniques [7].

While mapping with a specialized fleet is feasible for urban areas, it is time-consuming and cumbersome to create and maintain maps of entire continents. Thus, a key enabler for large-scale up-to-date maps will be enlisting the help of the very consumer vehicles that need the map to build and update it. Consumer vehicles will likely be equipped with low-cost consumer-grade sensor suites. Accordingly, this paper explores the accuracy limit of globally-referenced mapping involving collaborating consumer vehicles whose sense of global position is based on standard code-phase-based GNSS receivers. Key parameters in this exploration are the asymptotic averages of the error sources that impair code-phase-based GNSS positioning: receiver thermal noise, satellite clock and orbit errors, ionospheric and tropospheric modeling errors, and multipath. One or more vehicles navigating through a digital map over time make multiple time-separated GNSS measurements of the same location. If these vehicles collaboratively update the map over multiple sessions, then the GNSS errors are averaged across all sessions with appropriate weighting.

Are the GNSS errors at every map location—including deep urban locations—asymptotically zero-mean, or, on the contrary, do location-dependent biases persist in averages of time-separated standard GNSS measurements? Such is the question this paper seeks to address. To this end, it describes and demonstrates a stereo-camera-based digital mapping pipeline called GEOSLAM (globally-referenced electro-optical simultaneous localization and mapping) that achieves the accuracy limit of digital mapping with standard GNSS. GEOSLAM combines standard GNSS position estimates with a visual SLAM system in a tightly-coupled architecture.

To achieve the accuracy limit, GEOSLAM merges and jointly optimizes maps over multiple sessions with time-separated GNSS position estimates. This paper details the techniques GEOSLAM invokes to smoothly transition between unmapped and previously-mapped regions, consistently fusing current and prior maps without the need for a six degrees-of-freedom (6-DoF) pose from an inertial navigation system (INS). GEOSLAM enables multi-agent collaborative mapping by storing and rendering its map in a global

frame of reference, such as the World Geodetic System 1984 or the International Terrestrial Reference Frame. Multi-session operation of GEOSLAM is demonstrated using camera and GNSS data collected in a moderate urban environment, and the accuracy of global localization with the multi-session GEOSLAM map is assessed with respect to CDGNSS-based ground truth.

2. Previous Work

Improving the accuracy of maps by averaging GPS/GNSS tracks has been explored previously using a variety of approaches. An early effort, detailed in [8], proposed the precise determination of lane centerlines by clustering and averaging the GNSS tracks of probe vehicles. The accuracy of the estimated centerline was assessed in terms of the spread of GNSS tracks, assuming, without analysis, that the error was zero-mean at every location. Likewise, [9] invoked the zero-mean assumption without examination. More recently, [10] proposed vehicle lane determination via PPP on a rural road under open-sky conditions. The current paper aims to perform localization at a similar accuracy level, but in urban and suburban areas and with the aid of a digital mapping sensor.

Minimizing the difference between GNSS measurements and the assigned map coordinates of locations visited multiple times by probe vehicles has been a common feature of the seminal works on map-based precise localization in urban environments for automated driving [4,5], but no analysis of the accuracy of the resulting map in the global coordinate system was provided.

The effect of multipath on measured pseudoranges was studied extensively for various signal types in [11]. However, this study was done under open-sky conditions with a static survey-grade antenna, hardly representative of a mass-market receiver in an urban environment. The effect of multipath from a single large reflector on a static receiver was studied in [12], where it was shown that the position solution of the receiver was significantly biased when tracking multipath-afflicted signals. However, no details were provided about the tracking strategy and navigation filter. Multipath characterization for a dynamic receiver in an urban area was performed in [13], but the study was carried out for a single run through the test area, and many of the important error sources (e.g., atmospheric modeling errors), were assumed to be negligible. A detailed study on the distribution of code-phase and Doppler offsets of the multipath components from individual satellites in a dynamic urban setting was carried out in [14]. However, the error was characterized as the combined distribution of code phase delays over the entire duration of the run, which marginalizes over the temporally- and physically-local biases. On the contrary, this paper explores the errors in the position domain for repeated sessions through a given realization of an urban corridor.

Other GNSS error sources such as errors in modeling of ionospheric [15] and tropospheric [16] delay have been studied extensively over many decades, and their long-term error characteristics have also been reported in the literature. However, the impact of these errors on the asymptotic statistics of code-phase-based GNSS position estimates has not been previously presented.

To the authors' best knowledge, despite the apparent simplicity of the problem, no prior work has studied the long-term statistics of GNSS errors in an urban environment representative of the conditions to be encountered by consumer vehicles creating digital maps anchored by code-phase-based GNSS positioning. One of this paper's primary contributions is to address this gap in the literature.

Sensor fusion of visible-light cameras and GNSS has been extensively studied [17–28]. Some of these works [26–28] have proposed visual odometry as a replacement for, or an augmentation of, the traditional GNSS/INS architecture. Visual data from cameras are exploited to perform dead reckoning in a visual odometry pipeline, wherein an important distinction from the current paper is that the 3D map points do not persist after a window of time has elapsed—that is, no map of feature points is maintained.

Clearly, such an approach does not allow improvement of the 3D map point positions over multiple mapping sessions.

In [25], the relative change in position between two image frames is first estimated based on time-differenced GNSS carrier phase measurements. The metric-accurate GNSS-derived change in position is exploited to initialize the otherwise unobservable scale in a monocular visual SLAM system. However, GNSS measurements are not incorporated any further once the absolute scale has been initialized. Unlike the current paper, the visual SLAM map is rendered in the arbitrary SLAM coordinate frame since only the relative change in position, and not the absolute position, was estimated based on GNSS measurements.

The vision-GNSS fusion in the current paper is closely aligned with the bi-objective bundle adjustment (BA) optimization techniques previously reported in [17,18,20,22–24]. In [17,18,20,22], the traditional visual SLAM reprojection cost function is jointly minimized along with a GNSS position error term. The methods proposed in [23,24] are also similar, but guarantee that the visual reprojection cost after incorporation of the GNSS term is not significantly greater than the visual-only case. However, none of these works showed significant empirical evidence of their efficacy on real-world vehicle data sets. Furthermore, collaborative mapping or multi-session improvement of the map was not discussed.

Collaborative multi-agent mapping, without GNSS aiding, has also been extensively discussed in the literature [29–35]. Some of these proposed solutions require significant overlap in the field-of-view of the agents, or require that the relative pose transformation between the agents be known a priori [30,32]. Other solutions, such as in [31,33–35], enable collaboration by performing data association between non-concurrent mapping sessions where the relative pose transformation between the agents is unknown. The multi-session strategy employed in this paper is similar, but with an important distinction: none of the previous works on collaborative mapping have incorporated GNSS measurements in the map-building process. Without global referencing, the problem of data association between non-concurrent sessions becomes intractable. With no estimate of the pose for the mapping platform in relation to the existing map, data association must be attempted against the entire map. It is easily observed that such data association will become infeasible when scaled to city- or country-wide maps. The current paper proposes rendering and storage of digital maps in a global coordinate frame, such that a new mapping session can readily estimate its approximate pose in relation to the prior map, and perform data association on a small segment of the prior map that is expected to be in view of the vision system.

The work presented in [36] is perhaps the most closely related to the current paper. In [36], a particular stretch of roadway is mapped 25 times with a low-cost sensor setup. However, [36] assumes, without detail, the availability of a lane-level accurate low-cost positioning module that provides the full 6-DoF pose for the mapping platform. This greatly simplifies the ensuing data association and mapping pipeline. No mention is made of the general setting of the roadway being mapped (open-sky highway, urban canyon, etc.), and while the accuracy of the mapped traffic signs is adequately reported, localization within the map is not discussed, presumably since lane-level accurate positioning is already available. Meanwhile, the current paper only assumes the availability of a meter-level accurate code-phase-based GNSS receiver that provides 3D position estimates. Global localization accuracy of a vehicle operating within the multi-session map is presented as the key performance indicator.

3. GNSS Error Analysis

3.1. Low-Cost GNSS in Urban Areas

Low-cost multi-GNSS receiver manufacturers have recently announced the development and release of low-cost multi-frequency multi-GNSS receivers [37]. Accordingly, the analysis in this section considers a vehicular platform equipped with a multi-frequency multi-GNSS receiver capable of tracking both code and carrier phase of GNSS signals.

Development of an extensive dense reference network in support of CDGNSS consumer vehicular positioning in urban areas, as suggested in [38], could be an expensive affair. PPP is a low-cost alternative to CDGNSS that requires only a sparse network of reference stations across the globe, but is not considered a viable option for urban GNSS positioning in this paper because the constant cycle slips and outages experienced in urban areas [6] make it difficult for PPP's float carrier phase ambiguity estimates to converge [7], in which case PPP degrades to code-phase positioning accuracy.

While convergence of PPP carrier-phase ambiguities may be infeasible in urban areas, a partial PPP solution that exploits precise satellite orbits and clocks, as well as ionospheric and tropospheric corrections, can certainly improve the accuracy of code-phase-based GNSS position estimates. Since connected and automated vehicles will perform enjoy network connectivity, this paper assumes the availability of such GNSS corrections. Thus, the kind of GNSS errors assessed in this section lie between those corresponding to the two extremes of standard standalone code-phase positioning and PPP. This type of GNSS positioning, hereafter referred to as enhanced code-phase positioning, exploits both code and carrier phase or frequency tracking, but, as opposed to PPP, does not attempt to estimate a quasi-constant float carrier phase ambiguity, making it suitable for urban applications.

3.2. Pseudorange Measurement

The pseudorange measurement at receiver R from satellite S_i is modeled as

$$\begin{aligned}\rho_i(t_R) &= h_i[\mathbf{x}(t_R), I_{\rho_i}(t_R), T_i(t_R), t_R] + w_{\rho_i}(t_R) \\ &= \Delta r_i + c[\delta t_R(t_R) - \delta t_{S_i}(t - \delta t_{\text{TOF}_i})] + I_{\rho_i}(t_R) + T_i(t_R) + w_{\rho_i}(t_R),\end{aligned}\quad (1)$$

where

$$\mathbf{x}(t_R) \triangleq \begin{bmatrix} \mathbf{r}_R(t_R) \\ \delta t_R(t_R) \end{bmatrix}$$

is the state of the receiver, comprising the receiver position, $\mathbf{r}_R(t_R)$, at the time of the signal receipt event, t_R , and the receiver clock bias, $\delta t_R(t_R) = t_R - t$, with respect to true time t . The nonlinear measurement model is denoted by h_i ; ρ_i denotes the measured pseudorange to S_i ; c denotes the speed of light in vacuum; $\delta t_{S_i}(t) = t_{S_i} - t$ denotes the satellite clock bias with respect to t ; δt_{TOF_i} denotes the time-of-flight of the signal from S_i , as an increment in true time; I_{ρ_i} and T_i denote the ionospheric and tropospheric delay experienced by the signal from S_i , respectively; $w_{\rho_i} \sim (\mu_{w_i}, \sigma_{w_i}^2)$ denotes the sum of measurement thermal noise, multipath interference, non-line-of-sight (NLOS) delay, and other unmodeled errors; and Δr_i denotes the true range between R and S_i , given as

$$\Delta r_i = \|\mathbf{r}_R(t_R) - \mathbf{r}_{S_i}(t_R - \delta t_R(t_R) - \delta t_{\text{TOF}_i})\|,$$

where \mathbf{r}_{S_i} is the satellite position at the signal transmit event. Note from (1) that the receiver clock bias component of the state contributes identically to all pseudorange measurements.

Taking n_z pseudorange measurements $\{\rho_i\}_{i=1}^{n_z}$ and predictions \bar{I}_{ρ_i} and \bar{T}_i for each measurement, R estimates its state by solving a nonlinear least squares problem based on (1). First, it linearizes the measurement model in (1) about an initial guess of its state $\bar{\mathbf{x}}(t_R) = [\bar{\mathbf{r}}_R^T(t_R) \bar{\delta t}_R(t_R)]^T$ and the modeled atmospheric delays:

$$\rho_i \approx h_i(\bar{\mathbf{x}}, \bar{I}_{\rho_i}, \bar{T}_i, t_R) + \underbrace{\left[\frac{\partial h_i}{\partial \mathbf{x}} \right]_{\mathbf{x}=\bar{\mathbf{x}}}}_{H_i} (\mathbf{x} - \bar{\mathbf{x}}) + \bar{I}_{\rho_i} + \bar{T}_i + w_{\rho_i},$$

with $\tilde{I}_{\rho_i} = I_{\rho_i} - \bar{I}_{\rho_i}$ and $\tilde{T}_i = T_i - \bar{T}_i$. Representing all n_z measurements in matrix form yields

$$\begin{bmatrix} \rho_1 \\ \vdots \\ \rho_{n_z} \end{bmatrix} = \begin{bmatrix} h_1(\bar{\mathbf{x}}, \bar{I}_{\rho_1}, \bar{T}_1, t_R) \\ \vdots \\ h_{n_z}(\bar{\mathbf{x}}, \bar{I}_{\rho_{n_z}}, \bar{T}_{n_z}, t_R) \end{bmatrix} + \begin{bmatrix} H_1 \\ \vdots \\ H_{n_z} \end{bmatrix} (\mathbf{x} - \bar{\mathbf{x}}) + \begin{bmatrix} \tilde{I}_{\rho_1} \\ \vdots \\ \tilde{I}_{\rho_{n_z}} \end{bmatrix} + \begin{bmatrix} \tilde{T}_1 \\ \vdots \\ \tilde{T}_{n_z} \end{bmatrix} + \begin{bmatrix} w_1 \\ \vdots \\ w_{n_z} \end{bmatrix}$$

or

$$\boldsymbol{\rho} = \mathbf{h}(\bar{\mathbf{x}}, \bar{\mathbf{I}}, \bar{\mathbf{T}}, t_R) + \mathbf{H}(\mathbf{x} - \bar{\mathbf{x}}) + \tilde{\mathbf{I}} + \tilde{\mathbf{T}} + \mathbf{w}. \quad (2)$$

Rearranging measured and modeled quantities on the left-hand side to get the standard form for a linearized measurement model yields

$$\mathbf{z} \triangleq \boldsymbol{\rho} - \mathbf{h}(\bar{\mathbf{x}}, \bar{\mathbf{I}}, \bar{\mathbf{T}}, t_R) + \mathbf{H}\bar{\mathbf{x}} \implies \mathbf{z} = \mathbf{H}\mathbf{x} + \tilde{\mathbf{I}} + \tilde{\mathbf{T}} + \mathbf{w}. \quad (3)$$

The i th row of the measurement sensitivity matrix \mathbf{H} is

$$H_i \approx \left[\frac{\bar{\mathbf{r}}_R^T(t_R) - \bar{\mathbf{r}}_{S_i}^T(t_R - \delta t_R - \delta t_{\text{TOF}_i})}{\|\bar{\mathbf{r}}_R(t_R) - \bar{\mathbf{r}}_{S_i}(t_R - \delta t_R - \delta t_{\text{TOF}_i})\|}, 1 \right].$$

By solving (3) for \mathbf{x} , updating $\bar{\mathbf{x}}$, and iterating until convergence, R obtains its state estimate $\hat{\mathbf{x}}(t_R)$:

$$\hat{\mathbf{x}}(t_R) \triangleq \begin{bmatrix} \hat{\mathbf{r}}_R(t_R) \\ \hat{\delta t}_R(t_R) \end{bmatrix}.$$

For dynamic applications such as vehicle tracking, the state $\mathbf{x}(t_R)$ is typically augmented to include the time derivatives of $\mathbf{r}_R(t_R)$ and $t_R(t_R)$, and the measurement model typically assumes direct measurement of apparent Doppler frequency.

3.3. Error Sources

The major sources of error in the estimates $\hat{\mathbf{r}}_R$ and $\hat{\delta t}_R$ are as follows:

3.3.1. Thermal Noise

Measurement thermal noise at the receiver is one of the components of w_{ρ_i} in (1). The effect of thermal noise can be accurately modeled as a white Gaussian random variable with zero mean and standard deviation σ_T . For the pseudorange measurement, σ_T is typically between 10–30 cm, depending on the signal carrier-to-noise ratio, signal bandwidth, and receiver tracking bandwidth [39]. Estimation of the receiver state from multiple appropriately-weighted measurements with independent thermal-noise errors, and processing such measurements over time through a filter based on the modeled dynamics of the receiver, renders negligible the position-domain effects of uncorrelated zero-mean thermal noise. As a result, thermal noise is not a major contributor to the asymptotic accuracy of a digital map.

3.3.2. Satellite Orbit and Clock Errors

Satellite orbit and clock errors manifest in the modeled satellite position $\bar{\mathbf{r}}_{S_i}$ and the modeled satellite clock bias $\bar{\delta t}_{S_i}$. The International GNSS Service (IGS) provides orbit and clock models for GNSS satellites. The predicted *ultra rapid* orbits and satellite clocks have an accuracy of ~ 5 cm and ~ 3 ns, respectively [40]. These may add up to ~ 1 m of combined pseudorange model error for a given satellite. The 17-h retroactively-available *rapid* orbits and satellite clock models are accurate to ~ 2.5 cm and ~ 75 ps

RMS errors, respectively [40], adding up to less than 5 cm of RMS error in the modeled pseudorange for a given signal. Since the orbit and clock parameters are fit to measurements made at IGS analysis centers, the errors in the estimated parameters must be asymptotically zero-mean by design of the estimator. For post-processing applications such as mapping, it is reasonable to assume the availability of *rapid* orbit and satellite clock products, and thus the asymptotic average position errors due to errors in modeled satellite position and clock bias can be reduced to a sub-5-cm level.

3.3.3. Ionospheric Modeling Errors

The code-modulated GNSS signal propagates slower through the ionosphere as compared to vacuum due to the slightly-greater-than-unity *group* index of refraction for this atmospheric layer. The excess group delay is given as

$$\Delta\tau_g = \frac{40.3 \cdot TEC}{cf^2},$$

where *TEC* is the total electron content in electrons/m² and *f* is the frequency of the propagating signal. At the GPS L1 frequency centered at 1575.42 MHz, the excess ionospheric group delay is roughly 16.24 cm per TECU (1 TECU \triangleq 10¹⁶ electrons/m²). If not modeled, the ionospheric delay can lead to ranging errors greater than 15 m.

The ionospheric delay can be estimated via an ionosphere model or, in case of a multi-frequency receiver, eliminated via a combination of multiple-frequency pseudorange measurements. The latter technique does not require any external aiding, but the formation of the ionosphere-free combination exacerbates pseudorange noise, including any biases due to tracking of multipath signals. Compensating for ionospheric delay with the aid of an ionosphere model is applicable to both single- and multi-frequency receivers. It relies on accurate delay modeling based on ionospheric measurements at permanent GNSS reference stations, such as those that form the IGS network. While both methods have their merits, the analysis in this section considers corrections from an ionospheric model, and thus will not be relevant to applications where the ionosphere-free combination is applied. Note that those applications would likely experience worse multipath errors than the ones presented later, requiring a separate multipath analysis along the lines of Section 3.3.5.

Ionospheric model accuracy was studied comprehensively in [15]. The method in [15] generates unambiguous carrier-phase measurements from a global distribution of permanent receivers to compute the true slant total electronic content (STEC) for each satellite, and compares the model prediction for a number of models with the ground truth. In [41], the same authors compared PPP convergence times when applying different ionospheric correction models. This section extends the analysis in [15,41] to examine whether there exist long-term position-domain biases in enhanced code-phase positioning.

The post-fit residuals for multiple regional and global ionospheric models, computed as described in [15], were graciously made available by the same authors for the year 2014. These residuals were computed for GPS signals as observed at about 150 reference stations around the globe at 5 min intervals.

To observe the position-domain effect of the ionospheric modeling errors in isolation, this section neglects all other error sources, reducing the linearized measurement model in (3) to

$$z = Hx + \tilde{I}.$$

Historical GPS satellite almanacs can be combined with the timestamps from the residuals data to obtain the measurement sensitivity matrix *H* at each epoch for each station. With an elevation-dependent measurement covariance matrix *R*, the error in the weighted least-squares solution due to errors in ionospheric modeling is

$$\hat{x} - x = \left(H^T R^{-1} H \right)^{-1} H^T R^{-1} \bar{I}.$$

Figure 1 summarizes the results for ionospheric corrections obtained from the IGS global ionospheric map (GIM). Each of the arrows in Figure 1 points in the direction of the position bias in the east-north plane, as estimated over 12 months of data from 2014 (more than 800,000 samples per station). The magnitude of the horizontal position bias is depicted by the color of the arrow according to the scale shown on the right. Interestingly, there is a clear trend of southward bias in the position error for most stations in the northern hemisphere, and a mild trend of northward bias in the position error for stations in the southern hemisphere. A numerical summary of the IGS GIM position bias is presented in Table 1, along with a similar analysis for the Wide Area Augmentation System (WAAS) ionospheric corrections available for the contiguous United States (CONUS) region. As reported in [15], the WAAS model was found to exhibit a significantly smaller RMS error in ionosphere TEC estimates when compared to the IGS GIM; however the long-term position bias due to WAAS corrections is similar to or worse than those for the IGS model.

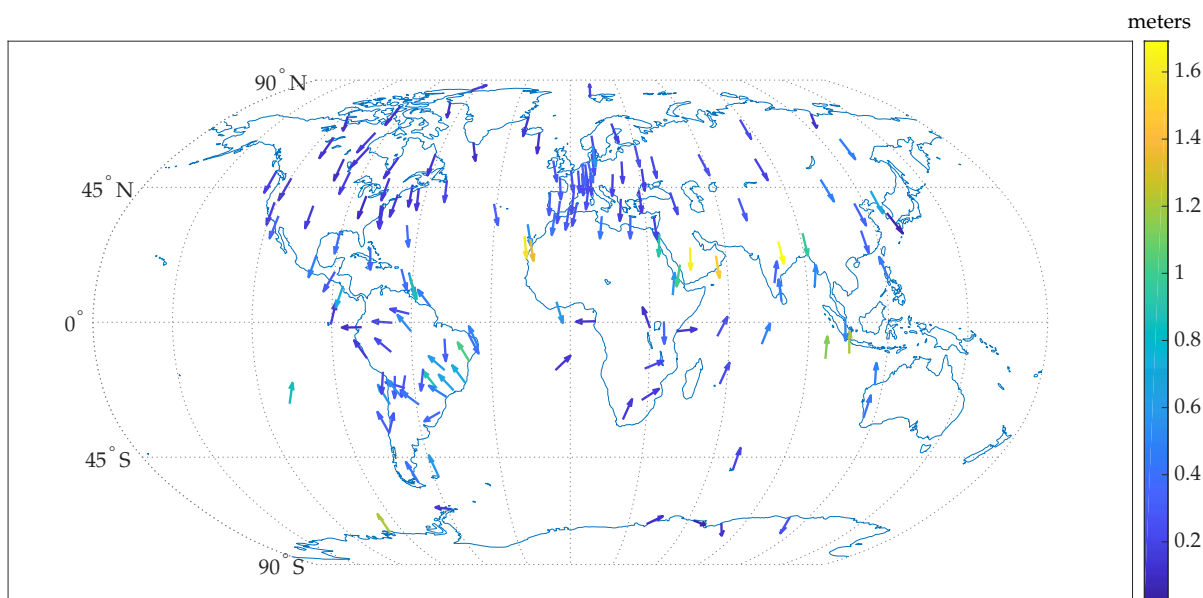


Figure 1. Direction and magnitude (the latter represented by color, in meters) of the long-term average horizontal position error due to errors in the delay estimates provided by the IGS GIM. Note that the meridians are curved outwards due to projection of the spherical map, and that arrows parallel to the curved meridians point directly south or north.

Another global ionospheric model, the Fast PPP model [41], was also studied as above. Fast PPP natively models the ionosphere as a two-layered shell, but is also made available in the standard one-layer IONEX (ionosphere-map exchange) format [15] for dissemination. The results presented in Table 1 represent the IONEX version of Fast PPP. In comparison with the IGS corrections, it is clear that the Fast PPP IONEX GIM corrections result in substantially unbiased long-term position errors at the global test locations. However, it must be conceded that the results in Table 1 are best-case results, as they are based on data from the same permanent reference stations used to constrain the model.

Table 1. Long-term average position error due to ionospheric model errors (ϕ denotes station latitude). IGS: International Global Navigation Satellite System (GNSS) Service; PPP: precise point positioning; WAAS: Wide Area Augmentation System; CONUS: contiguous United States; IONEX: ionosphere-map exchange format.

Ionosphere Model	Region	East (m)	North (m)	Up (m)
IGS	$\phi \geq 30^\circ$	0.0107	-0.2129	0.6733
	$30^\circ > \phi > -30^\circ$	-0.0651	-0.0692	1.5467
	$\phi \leq -30^\circ$	0.0237	0.2450	0.3355
WAAS	CONUS	-0.0048	-0.2916	-0.1248
Fast PPP IONEX	$\phi \geq 30^\circ$	-0.0042	-0.0099	-0.0122
	$30^\circ > \phi > -30^\circ$	-0.0390	0.0013	-0.3053
	$\phi \leq -30^\circ$	-0.0325	-0.0087	0.0309

To understand the reason behind the systematic biases with IGS corrections, note that any ionospheric modeling bias that identically affects all satellites does not have any impact on the accuracy of the GNSS position solution, as this common error is absorbed in $\hat{\delta}t_R$. Rather, position-domain biases arise from the azimuthal- and elevation-dependence of ionosphere model errors. From analysis of the spatial distribution of post-fit residuals, it was found that appreciable azimuthal and elevation residual gradients persist in the IGS ionospheric corrections. These gradients are represented graphically in Figure 2 for one representative station from the northern hemisphere (station code: EUSK, latitude: $50^\circ 40' 26.87''$, longitude: $6^\circ 45' 48.72''$) and one representative station from the southern hemisphere (station code: VACS, latitude: $-20^\circ 17' 48.47''$, longitude: $57^\circ 29' 13.79''$). The post-fit residuals are binned in azimuth and elevation and the average value in each bin is denoted by the color of the representing disc. The size of the disc denotes the number of samples of post-fit residuals available in each bin. Due to the inclination angle of the GPS satellite orbits, the angular distribution of satellites at any given latitude is non-uniform.

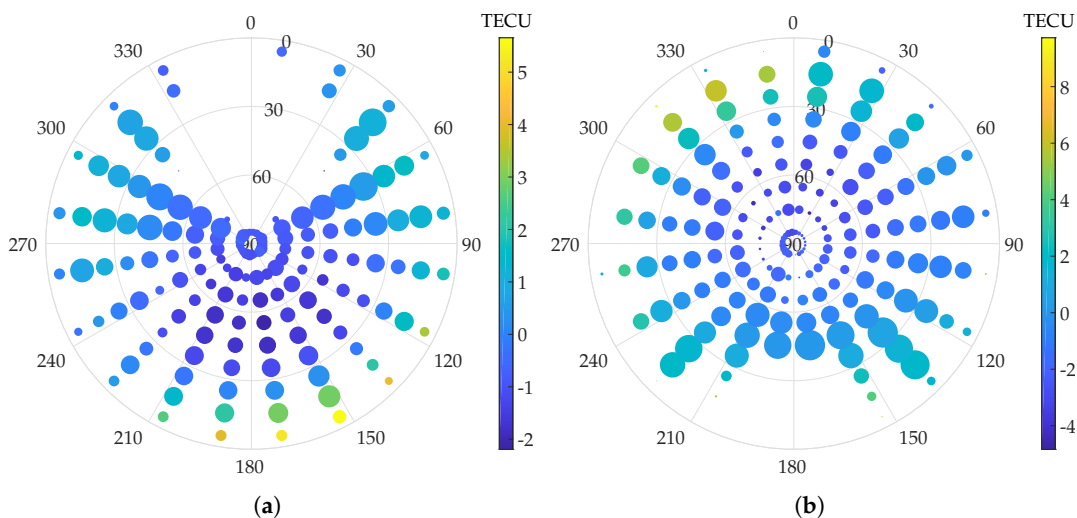


Figure 2. Azimuth and elevation dependence of post-fit IGS global ionospheric map (GIM) residuals. (a) A representative station from the northern hemisphere. (b) A representative station from the southern hemisphere. The average residual error (in TECU) is denoted by the color of the disc. The size of the disc indicates the number of samples of post-fit residuals available in each bin.

From Figure 2, it is clear that the elevation gradients in the ionospheric residuals are pronounced. A subtle azimuthal gradient also exists, mainly along the north-south direction. Such spatial non-uniformity, coupled with the non-uniform satellite angular distribution, may be the reason for the observed persistent position biases. While the elevation gradients are consistent for stations at all locations, the azimuthal gradients appear to invert along the north-south direction between the northern and southern hemisphere. This is likely the reason for the opposite direction of the average horizontal position bias in the northern and southern hemispheres.

Such persistent position-domain biases due to inaccurate ionospheric modeling have not been previously reported in the literature, and are a rather remarkable result. While some single-frequency PPP (SF-PPP) techniques eliminate the ionospheric delays based on the GRAPHIC combination [42], many other techniques that rely solely on ionospheric corrections from GIMs have been shown to achieve 30 cm 95% accuracy in the east-north plane after convergence, with sub-10-cm bias [43], seemingly contradicting the results here. The key difference is that the SF-PPP methods involve estimation of a float carrier ambiguity term for each satellite arc. A portion of the systematic biases in the GIM estimates is likely absorbed in these states of the estimator, thereby attenuating the position biases in the east-north plane. For instance, the SF-PPP technique in [43] is based on the phase-adjusted pseudorange algorithm [44], wherein the ambiguity term for each satellite, physically an unknown constant, is in fact iteratively estimated with small but non-zero process noise. In such an estimator, the ambiguity term can absorb slowly time-varying systematic biases. In other SF-PPP techniques, the ionospheric correction term is explicitly included as a state to be estimated, and the estimates from GIM are applied as pseudo-observations [45,46]. Once again, decimeter-level biases in the GIM estimates of the ionospheric delay may not necessarily appear in the final reported position accuracy of the SF-PPP method. Of course, such absorption of biases in augmented states is not undesirable. However, for the case of urban vehicular positioning, convergence of SF-PPP is a concern due to carrier phase cycle slipping, as discussed earlier. In an enhanced code-phase-based receiver, the high variance of the code noise leads to poor observability of the decimeter-level horizontal position bias due to ionospheric modeling errors. Thus, ionospheric biases are not often estimated in a code-phase-based GNSS estimator.

Another factor of note is that 2014 was a maximum in the 11-year solar activity cycle, and thus the IGS GIM accuracy may have been worse than usual over this period of time.

In conclusion, persistent decimeter-level biases in the east-north plane and meter-level biases in the vertical direction can arise when ionospheric delay corrections are sourced from the IGS GIM, or similar, even under ideal open-sky conditions. More advanced models of the ionosphere with more accurate slant TEC measurements may achieve better results. Elimination of the ionospheric delay based on the ionosphere-free combination is another option, but tends to worsen multipath-induced position errors. If corrections from some ionosphere model lead to unbiased position errors, then for globally-referencing digital maps by averaging GNSS measurements over many sessions it is advisable to avoid the combination of multi-frequency signals.

3.3.4. Tropospheric Modeling Errors

In the troposphere, or more generally the neutral atmosphere, the index of refraction departs from unity much less than in ionosphere at GNSS frequencies, causing a delay of ~ 2.4 m at zenith. The index of refraction in the troposphere is non-dispersive, and thus cannot be estimated using multiple-frequency signals. The tropospheric delay is obtained from models of the climatological parameters (temperature, pressure, and water vapor pressure) along the propagation path.

State-of-the-art tropospheric models [16] fit a small number of location- and day-of-year-dependent coefficients to climatological data from numerical weather models (NWMs) to model the zenith

tropospheric delay. The zenith delay is mapped to any elevation angle using mapping functions [47]. Similar to the ionospheric models, the tropospheric mapping functions may introduce a differential azimuth- and elevation-dependent error. For empirically-derived mapping functions such as VMF1 [47] and GMF [48], the mean error at lowest elevation of 5° has been shown to be under 50 mm (this value is typically reported as 10 mm station height error, which is approximately one-fifth of the delay error at lowest elevation [47]). As a result, this paper assumes that time-averaged tropospheric model errors would introduce sub-5-cm errors in the position domain, and would thus not impede asymptotically accurate collaborative mapping in both horizontal and vertical components at the several-decimeters level.

3.3.5. Multipath Error

In ideal circumstances, each signal received from an overhead satellite arrives only along the least-time path. In practice, however, this so-called line-of-sight (LOS) component is accompanied by other components due to signal diffraction and single- or multiple-signal reflections off surrounding surfaces and obstacles (e.g., the glass facade of a nearby building, poles, trees, etc.). The complex baseband representation of the N signal components received from a particular satellite at a particular frequency and code is

$$r(t) = \sum_{i=0}^{N-1} A_i(t)C[t - \tau_i(t)] \exp[j\theta_i(t)],$$

where A_i is the amplitude of the i th component, $C(t)$ is the GNSS code modulation, $\tau_i(t)$ is the delay of the i th signal component relative to an unobstructed LOS signal, and $\theta_i(t)$ is the beat carrier phase of the i th component. The combination of multiple components distorts the received signal and causes errors in the pseudorange and phase measurements.

Unlike the study of ionospheric modeling errors, for application in urban mapping, multipath errors cannot be characterized with data from survey stations with a clear view of the sky. This section considers a simulation approach for scalable analysis of multipath tracking errors in an urban environment. The objective of this study was to inspect the presence of persistent biases caused by multipath due to the surrounding structure in the navigation solution averaged over multiple sessions

Scenario Setup

The present simulation study was based on the open-access Land Mobile Satellite Channel Model (LMSCM) [49], itself based on extensive experimentation with a wideband airborne transmitter at GNSS frequencies in urban and suburban environments. First, an urban corridor was simulated stochastically following the procedure described in [50]. The corridor was composed of buildings, trees, and poles. Some of the important parameters for the generation of the scene are summarized in Table 2, and a part of the scene realization is shown in Figure 3. Multi-GNSS satellite trajectories were generated at randomly-selected times based on GPS and Galileo satellite almanac data. An average of 25 satellites were available above an elevation mask of 5° , consistent with modern multi-GNSS receivers. The satellites were assumed to be stationary over the simulation period of 60 s. Navigation solution errors were computed over 1000 60-s sessions.

Table 2. Some urban scenario parameters.

Distance from road center to buildings	24 m	Distance from road center to vehicle	5 m
Mean distance between road center and trees	20 m	Antenna height	2 m
Mean building width	30 m	Building width standard deviation	25 m
Mean building height	40 m	Building height standard deviation	20 m
Probability of gap between buildings	0.5	Mean gap width	30 m
Mean distance between trees	60 m	Mean distance between poles	25 m

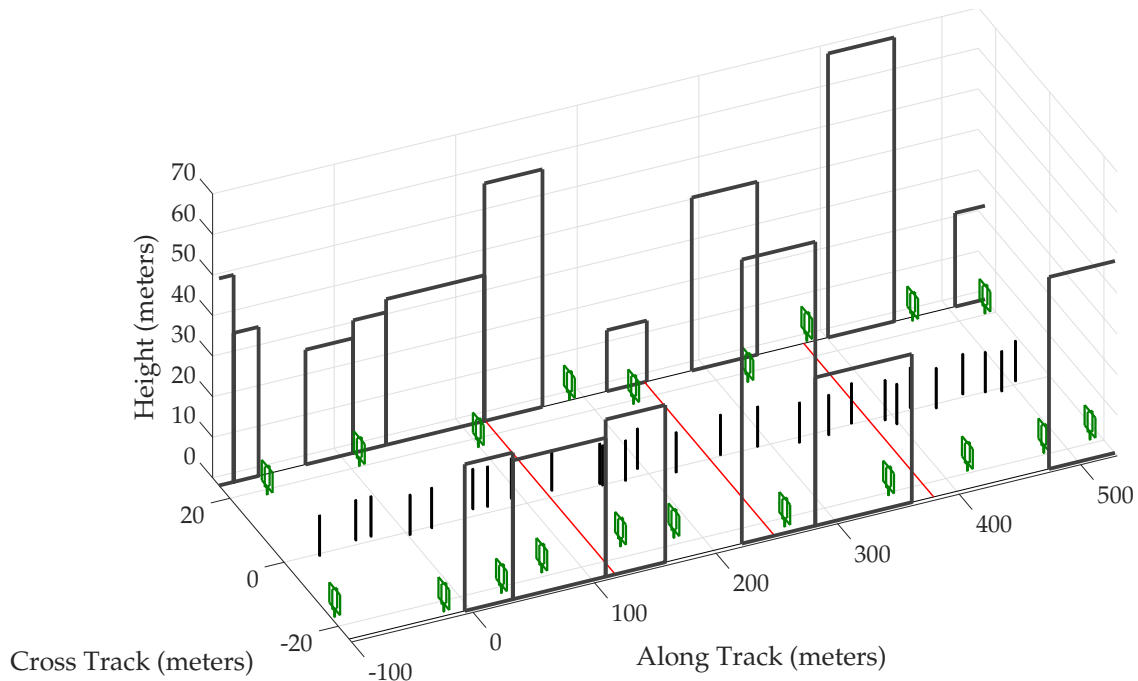


Figure 3. Initial segment of the simulated urban corridor. Red lines across the road denote the positions where the vehicle is momentarily stopped.

The vehicle trajectory was kept consistent across all 1000 driving sessions to avoid decorrelation of multipath error due to variable receiver motion. The trajectory was parametrized by its speed and heading, as described in [50]. The vehicle started at the zero coordinate on the along-track axis, and traveled in the positive direction, which was assumed to be aligned with the local north. The simulated trajectory was 60 s long and simulated a vehicle in stop-and-go traffic executing one 90° right turn, as shown in Figure 4. The vehicle traveled roughly 430 m and faced eastwards at the end of the trajectory. The three low-speed intervals, marked with red line segments in Figure 3, are expected to present severe multipath effects since multipath errors decorrelate slowly, and thus tend to reinforce one another within the navigation filter, when the vehicle moves slowly.

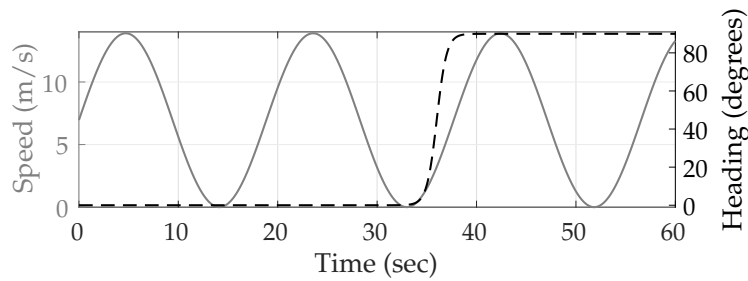


Figure 4. Vehicle speed (solid line) and heading (dashed line) simulating stop-and-go motion with a 90° right turn.

Multipath Simulation

The LMSCM generates power, delay, and carrier phase for N LOS and echo signals. The interaction of the LOS with the simulated obstacles is governed by deterministic models for attenuation, diffraction, and delay. The LOS components of the combined signal, denoted $r_{\text{LOS}}(t)$, may be composed of multiple components due to signal diffraction. These components are modeled as

$$r_{\text{LOS}}(t) = \sum_{i=0}^{N_{\text{LOS}}-1} A_i(t) C[t - \tau_i(t)] \exp[j\theta_i(t)].$$

In the special case of an unobstructed LOS signal, $N_{\text{LOS}} = 1$, $A_0(t) = 1$, $\tau_0(t) = 0$, and

$$\theta_0(t) = \frac{\|\mathbf{r}_R(t) - \mathbf{r}_S(t)\| \cdot 2\pi}{\lambda} + \gamma_0,$$

where λ denotes the wavelength and γ_0 is a constant due to phase initialization in the satellite and receiver [51].

The LMSCM generates the $N - N_{\text{LOS}}$ NLOS echoes stochastically based on satellite azimuth and elevation, receiver dynamics, and general characteristics of the scene (e.g., an *urban car* scenario). This stochastic procedure might not be representative of multipath over multiple sessions through the same urban corridor, where certain echoes might persist over different sessions. To address this limitation, the LMSCM was augmented by the present authors to generate one- and two-bounce deterministic reflective NLOS echoes off the simulated buildings, and a one-bounce NLOS echo off the ground surface. These three additional reflective NLOS echoes, denoted $r_{\text{DET}}(t)$, were added to $r(t)$ and are modeled as

$$r_{\text{DET}}(t) = \sum_{i=N}^{N+2} b_i(t) A_i(t) C[t - \tau_i(t)] \exp[j(\theta_i(t) + \theta'_i(t))],$$

where $b_i(t) \in \{0, 1\}$ denotes whether the surrounding geometry supports the reflective echo. Since these reflections are expected to be the stronger than other diffracted and multiple-bounce NLOS echoes, the amplitudes $A_i(t)$, $i \in \{N, N+2\}$ for reflective echoes were drawn from the distribution of the strongest echo generated stochastically by the LMSCM at each epoch. By experiment, this distribution was found to be log-normal with $20 \log_{10}(A_i) \sim \mathcal{N}(-22, 5)$, $i \in \{N, N+2\}$. The delays for the reflective echoes are given as

$$\tau_i(t) = \frac{\|\mathbf{r}'_R(t) - \mathbf{r}_S(t)\| - \|\mathbf{r}_R(t) - \mathbf{r}_S(t)\|}{c}, \quad i \in \{N, N+2\},$$

where $\mathbf{r}'_R(t)$ is the position of the imaginary *image* antenna [52] about the reflecting plane (building or ground). Similarly, the carrier-phase of the reflective echoes is computed geometrically as

$$\theta_i(t) = \frac{\|\mathbf{r}'_R(t) - \mathbf{r}_S(t)\| \cdot 2\pi}{\lambda} + \gamma_0, \quad i \in \{N, N+2\}.$$

A random carrier-phase offset $\theta'_i(t) \in [0, 2\pi)$ was added at the reflection point every time a new reflective echo was spawned to simulate the material-specific phase offset introduced by the reflection process.

Receiver

A receiver simulator was developed to account for the mediating effects that a receiver's tracking loops and navigation filter have on multipath-induced position errors in a receiver's reported position solution. The simulated receiver tracks the combination of all N_{LOS} line-of-sight signals and $N+2 - N_{\text{LOS}}$ multipath echoes for a given signal. If $R(\tau)$ denotes the correlation function of the GNSS signal's spreading code, then the multipath delay error in the tracked code phase, relative to unobstructed LOS, is given as the solution to [52]

$$0 = S_{\text{coh}}(\tau) = \sum_{i=0}^{N+2} A_i \cos(\theta_i - \theta_c) \times \left[R\left(\tau - \tau_i + \frac{d}{2}\right) - R\left(\tau - \tau_i - \frac{d}{2}\right) \right],$$

where θ_c is the tracked carrier-phase of the combined signal:

$$\theta_c = \text{atan2}\left(\sum_{i=0}^{N+2} A_i R(\tau_c - \tau_i) \sin(\theta_i), \sum_{i=0}^{N+2} A_i R(\tau_c - \tau_i) \cos(\theta_i)\right).$$

The parameter d is the early-to-late correlator spacing in the receiver. It is well-known that a wide-bandwidth receiver with narrow correlator spacing mitigates the effect of multipath [52]. To this end, the receiver considered in this simulation implements $d = 0.1$. It must be mentioned that $R(\tau)$ was implemented as the correlation function for GPS L1 C/A identically for all the simulated signals. Modernized GNSS signals have better multipath mitigation characteristics [11], but this behavior was not included in the simulation.

Another important observation is that when the LOS signal is strong as compared to the echo signals, the time derivative of the tracked carrier-phase is equal to the Doppler frequency of the LOS signal, which changes smoothly in accordance with the motion between the satellite and the receiver. However, when the LOS signal is comparable to or weaker than other rapidly-decorrelating echoes, the combined carrier-phase is uniformly random. In a GNSS receiver, the phase lock loop's phase-lock indicator indicates whether a sufficiently strong LOS signal is available, enabling carrier lock [6]. The simulator's phase-lock indicator is asserted only if (1) the tracked Doppler frequency does not deviate significantly from a second-order polynomial, and (2) the strongest received echo is attenuated by more than 25 dB with respect to an unattenuated signal.

Navigation Filter

At each epoch, n_z multipath-free, ionosphere-free, and troposphere-free simulated pseudorange measurements were combined with corresponding simulated multipath tracking delay errors and fed to a navigation filter that estimates the receiver state. The navigation filter implemented in this paper is an extended Kalman filter (EKF) with a nearly constant velocity motion model following [53]. The standard details of the EKF are omitted for brevity.

The effect of multipath tracking on the navigation solution is strongly dependent on the receiver's multipath rejection scheme. Two schemes are explored here. The first is a hypothetical ideal multipath rejection scheme that excludes all signals for which the LOS signal has a smaller-than-10-dB advantage over its multipath echoes. The second scheme implements a normalized innovation squared (NIS) test to reject multipath signals based on measurement innovations [53]. At the $(k + 1)$ th measurement update step, the difference between the predicted and observed measurement vector, called the innovation and denoted $\nu(k + 1)$, is squared and normalized by its covariance, which is the sum of the measurement covariance matrix, $R(k + 1)$, and the propagated state covariance transformed through the measurement sensitivity matrix, $H(k + 1)P(k + 1|k)H(k + 1)^T$. In the absence of multipath tracking errors, the resulting NIS statistic is chi-squared distributed with n_z degrees of freedom. If the NIS statistic exceeds a chosen threshold, then the signal with the largest normalized innovation is dropped. This continues until the NIS statistic falls below the threshold or the number of remaining signals drops to a preset minimum number of required signals.

Simulation Results

Figure 5 shows the mean position error in the east, north, and up directions over 1000 sessions for the two multipath rejection schemes mentioned previously. From Figure 5a, it can be seen that sub-20 cm average error is achievable with hypothetical ideal multipath exclusion. A closer look at Figures 4 and 5a reveals that the decimeter-level sinusoidal position error trend, initially in the north direction and later in the east direction, in fact corresponds with the along-track accelerations of the vehicle that were not adequately tracked by the nearly-constant-velocity-model-based navigation filter.

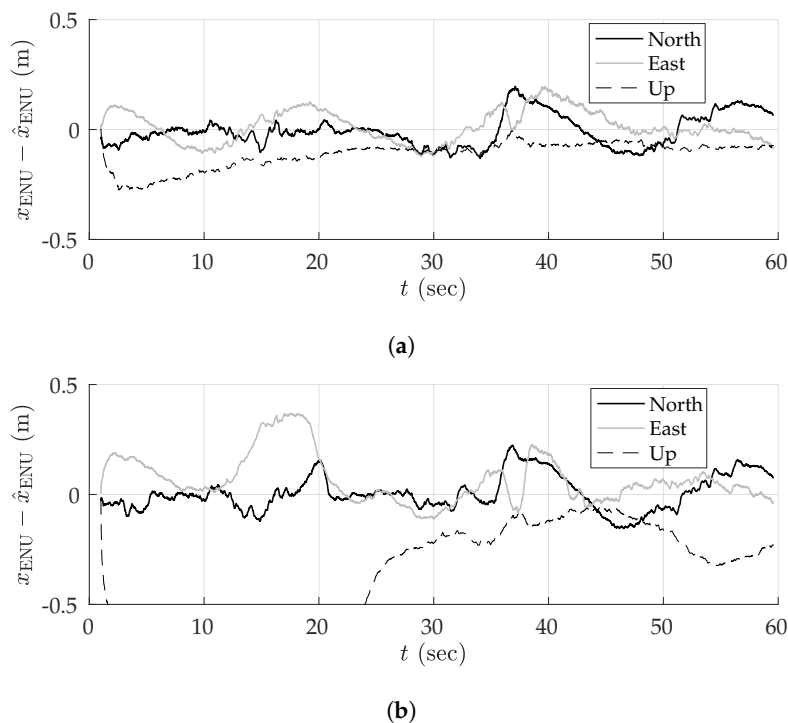


Figure 5. Mean position error in the east-north-up (ENU) frame over 1000 sessions due to multipath. (a) Ideal multipath exclusion. (b) Normalized innovation statistic (NIS)-based multipath exclusion. The black, gray, and dashed-black lines represent the error in the east, north, and up directions, respectively. The up error in the bottom panel reached a maximum magnitude of 1.75 m.

Figure 5b shows that the NIS test based exclusion of signals was able to approach the performance of ideal exclusion in the horizontal plane, save for the first stationary period where the vehicle was moving at low speed between buildings on both sides. The average vertical position error was much worse, growing as large as 1.75 m in magnitude.

To determine whether the average errors shown in Figure 5 are in fact persistent biases, a study of the standard deviation of position errors was conducted. The standard deviation of the average errors in east, north, and up directions was computed for disjoint averaging ensembles of size 1, 2, 4, 8, 16, 32, 50, and 100 sessions taken from the total of 1000 simulated sessions. For instance, 125 disjoint ensembles of eight sessions were selected, and the position errors were averaged over the eight sessions in each set. The standard deviation of the eight-session-averaged errors was then computed across the 125 ensembles. In the case of an averaging ensemble with only a single session (i.e., no averaging), the computed standard deviation is simply the measured standard deviation of the position error across all 1000 simulated runs. In the case of averaging over 100 sessions, the standard deviation is computed based on 10 disjoint averaging ensembles of 100 sessions each.

Note that because the simulation study was based on the same 1000 simulations for all averaging ensembles, the east, north, and up means taken across all averaging ensembles are equivalent to those shown in Figure 5. The more interesting trend is the decreasing standard deviation with increasing size of the averaging ensemble, as shown in Figure 6 for the case of NIS-based multipath rejection and for the east and north error components. As expected, the standard deviation of errors was higher at locations where the vehicle moved at low speed and multipath decorrelated slowly. Additionally, the standard deviation was larger at the beginning of the trajectory where the street was lined with tall buildings on both sides.

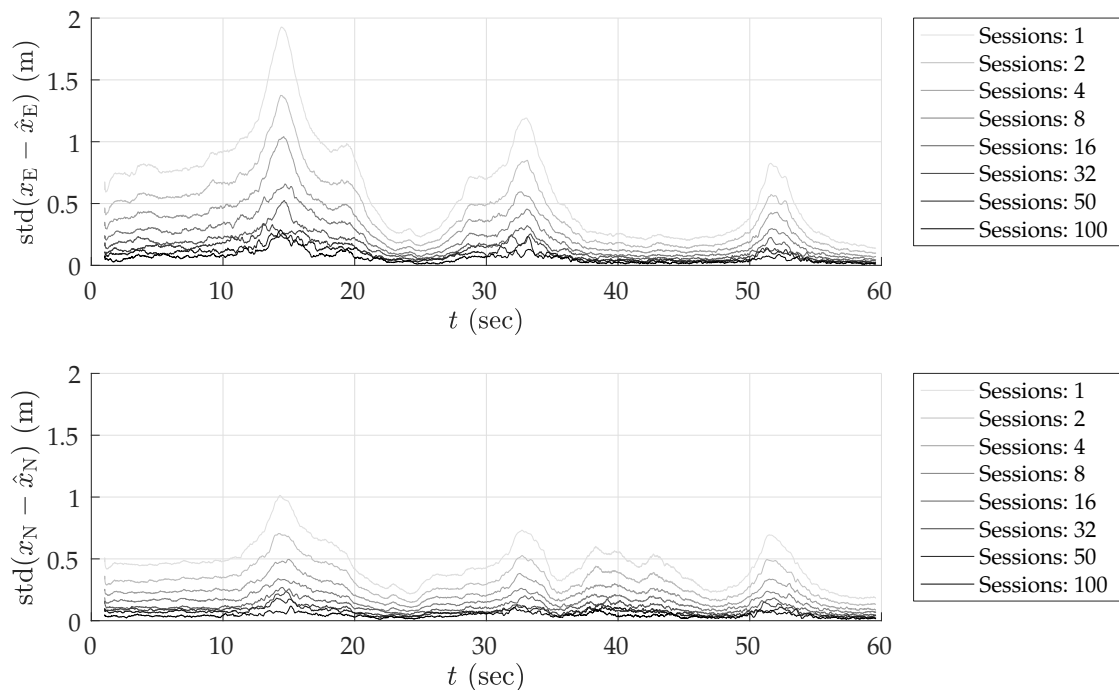


Figure 6. Standard deviation of average position error in east and north directions for NIS-based multipath exclusion as a function of the number of sessions over which the errors are averaged. **Top panel:** standard deviation in the east direction. **Bottom panel:** standard deviation in the north direction.

The standard deviation of the average east and north position error over 100 sessions was bounded below 15–20 cm. Thus, it is highly likely that the ~40-cm error in the north direction between 15–20 s in Figure 5b is in fact a persistent non-zero bias.

Table 3 summarizes the results of the multipath simulation study. It shows the 95-percentile horizontal error magnitude for increasing averaging ensemble sizes and for both ideal and NIS-based multipath exclusion. The 0–60 s average case lists the 95-percentile error over the entire trajectory, whereas the 13–19 s average case lists the 95-percentile error in the worst-case segment of the trajectory in terms of horizontal position bias and standard deviation. This challenging segment is illustrative of persistent problem spots that will arise in urban areas, within which multipath-induced biases will be larger than average. As expected, the 95-percentile error in Table 3 shrank as the averaging ensemble size became larger. For the urban corridor and vehicle dynamics considered in this simulation, NIS-based exclusion achieved 35 cm 95-percentile horizontal error with averaging over 100 sessions. Even in the worst-case region of the trajectory, the 95-percentile horizontal error remained below 50 cm. As multipath exclusion approaches the ideal case, with aid from other sensors or a 3D model of the surroundings, for example, the 95-percentile horizontal error could be reduced to as low as 25 cm for the simulated corridor.

Table 3. 95-percentile horizontal errors.

Averaging Ensemble Size:		1	2	4	8	16	32	50	100
Ideal	0–60 s average (m)	1.5910	1.1262	0.7902	0.5488	0.4078	0.3090	0.2696	0.2147
	13–19 s average (m)	2.5925	1.7809	1.2136	0.8927	0.6416	0.4145	0.3544	0.2609
NIS	0–60 s average (m)	1.7851	1.2795	0.9245	0.6588	0.5169	0.4175	0.3920	0.3526
	13–19 s average (m)	3.1217	2.1953	1.5467	1.1720	0.8456	0.6470	0.5950	0.4702

From the Section 3.3.3’s analysis of asymptotic ionospheric errors, and from this section’s multipath simulation study, one can draw the following conclusion: so long as the asymptotic horizontal position errors of the ionosphere corrections are below 5 cm, as is true for the Fast-PPP model, and assuming statistical independence of ionospheric and multipath errors, it appears feasible to achieve 50-cm horizontal positioning accuracy at approximately 95% by averaging over 100 mapping sessions.

4. Globally-Referenced Electro-Optical SLAM (GEOSLAM)

This section describes a simultaneous localization and mapping (SLAM) pipeline capable of globally-referenced collaborative multi-session digital mapping. The pipeline combines visual measurements from a stereo visible-light camera system with position measurements from GNSS signals. The objective of this pipeline is to demonstrate the development of an accurate digital map based on multiple mapping sessions with standard GNSS position estimates. The following sections detail the visual SLAM pipeline, the integration of GNSS measurements, and the techniques for multi-session mapping developed in GEOSLAM.

4.1. Visual SLAM

The visual SLAM component of GEOSLAM is similar to existing high-performance SLAM pipelines developed in the robotics community [54–56]. Visual SLAM algorithms may be categorized as either sparse or dense. Sparse visual SLAM algorithms [54,55] create a map of distinctive features such as corners or edges in the scene, while dense SLAM algorithms [56] map the depth for each pixel in the captured frames. The point cloud generated by sparse SLAM algorithms is sufficient for the purpose of localization. Dense reconstruction is appealing to the human eye, but does not provide any tangible

benefit to localization, while consuming much more computational resources. As a result, GEOSLAM implements sparse feature-point-based SLAM.

In [57] it was shown that for the visual SLAM problem, structure-from-motion BA (batch non-linear optimization) outperforms filtering techniques such as the extended Kalman filter, yielding higher accuracy per unit of computing time. It was also noted that having a high number of features points per image frame provides better accuracy than having a large number of frames with fewer feature points per frame. Thus, in typical practice, only a select subset of frames among those captured is retained for processing; frames in this subset are called keyframes. Most recent state-of-the-art visual SLAM algorithms use a keyframe-based BA approach instead of sequential filtering. Likewise, GEOSLAM performs BA-based non-linear optimization to refine both structure and motion.

Figure 7 shows a block diagram representation of the system architecture proposed in this paper. The yellow-colored blocks in this figure are components of the GEOSLAM pipeline, detailed next.

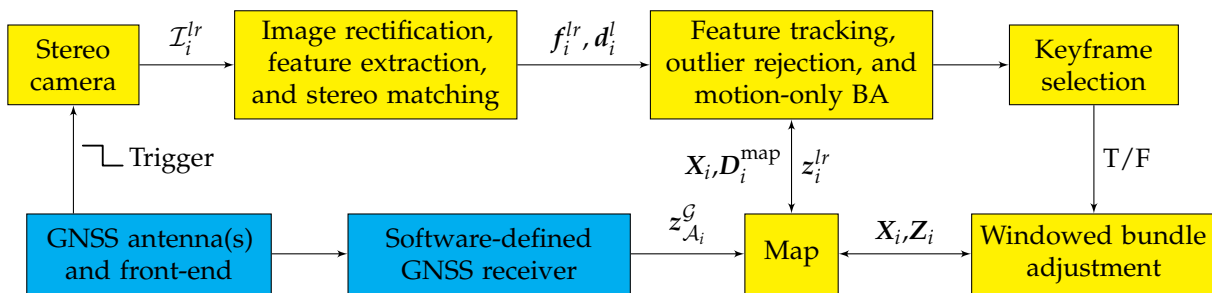


Figure 7. Globally-referenced electro-optical simultaneous localization and mapping (GEOSLAM) block diagram. BA: bundle adjustment.

By way of notation, let

$$z_{nm}^{lr} \triangleq (u_{nm}^l, v_{nm}^l, u_{nm}^r)$$

denote the image plane location of the stereo-matched feature matched to the m th map point, p_m , in the n th stereo keyframe, K_n . The horizontal and vertical coordinates are denoted u and v , respectively, while the superscripts l and r denote the left and right camera frames, respectively. Note that the feature location is specified by only three coordinates. The vertical feature coordinate in the right camera frame, v_{nm}^r , is omitted because for an undistorted and rectified camera model it must hold that $v_{nm}^l = v_{nm}^r$, making one of the coordinates redundant. If p_m is not matched to any feature in K_n , then let $z_{nm}^{lr} = \emptyset$. Furthermore, let

$$\mathcal{M}_n = \{m : z_{nm}^{lr} \neq \emptyset\}$$

denote the set of indices of all map points matched to some feature in K_n . In the visual SLAM literature, the covisibility window of keyframe K_i is defined as the set of keyframes that share at least T map points with K_i . Mathematically, the covisibility window of keyframe K_i is the set of keyframes with indices

$$\text{cov}(i) \triangleq \{n : |\mathcal{M}_n \cap \mathcal{M}_i| > T\},$$

where $|\mathcal{A}|$ denotes the cardinality of the set \mathcal{A} . The covisibility window determines the keyframes to be optimized in a windowed BA. The visibility of common points is regarded as a proxy for correlation between the structure-from-motion states. However, in a sensor fusion architecture, the states for other sensors (e.g., GNSS) may be spatially correlated beyond the covisibility window. Furthermore, other sensors may experience outages that extend beyond the covisibility window. In such a scenario, it would

be desirable to optimize over a batch of keyframes that span the availability gap. Accordingly, GEOSLAM extends the concept of covisibility to N levels as

$$\text{cov}(i, N) \triangleq \begin{cases} \{n : |\mathcal{M}_n \cap (\cup_{k \in \text{cov}(i, N-1)} \mathcal{M}_k)| > T\} & N > 1, \\ \{n : |\mathcal{M}_n \cap \mathcal{M}_i| > T\} & N = 1. \end{cases}$$

When processing K_i , GEOSLAM's objective is to estimate the map point 3D locations $\mathbf{x}_{p_m}^S \in \mathbb{R}^3$ and keyframe poses $(\mathbf{x}_{\mathcal{C}_n}^S, \boldsymbol{\theta}_{\mathcal{C}_n}^S) \in (\mathbb{R}^3, \mathbb{R}^3)$ in the N -level covisibility window for the i th keyframe, where \mathcal{S} stands for the local SLAM frame, \mathcal{C}_n is the left camera coordinate frame associated with K_n , and $\boldsymbol{\theta}_{\mathcal{C}_n}^S$ is the angle-axis representation of the keyframe orientation. The state vector to be estimated is represented as

$$\mathbf{X}_i \triangleq \left[\left\{ (\mathbf{x}_{\mathcal{C}_n}^S, \boldsymbol{\theta}_{\mathcal{C}_n}^S) : n \in \text{cov}(i, N) \right\}, \left\{ \mathbf{x}_{p_m}^S : m \in \cup_{k \in \text{cov}(i, N)} \mathcal{M}_k \right\} \right]^T, \quad (4)$$

where the two sets on the right-hand side are arranged as a concatenation of row vectors so that \mathbf{X}_i becomes a column vector.

When triggered by the GNSS front end, the camera setup captures a pair, denoted \mathcal{I}_i^{lr} , of concurrent images from the left and right cameras, where the subscript denotes that the current pair is a candidate to be the i th stereo keyframe K_i . The intrinsic and extrinsic parameters of the stereo camera setup are assumed to have been calibrated a priori. The stereo image pair is then undistorted and rectified according to the given calibration, and SIFT features are detected and computed separately for each image [58]. SIFT feature matching is performed between the left and right image with the additional constraint that matching features must have approximately the same vertical coordinate to within a few pixels. The set of stereo feature measurements for \mathcal{I}_i^{lr} , f_i^{lr} , and the set of feature descriptors as computed in the left image, d_i^l , are passed on to the tracking module.

The tracking module has access to the 3D map point positions within \mathbf{X}_i and to the set of SIFT descriptors, D_i^{map} , corresponding to the map points expected to be seen in the candidate keyframe \mathcal{I}_i^{lr} . The tracker performs directed matching of the features between the stereo image and the map. First, a quick feature matching is performed using the Fast Approximate Nearest Neighbor Search Library (FLANN) [59]. With sufficient matches, an initial approximation of the current camera pose is obtained using the five-point algorithm wrapped in random sample consensus (RANSAC) iterations. With this approximate pose, an iteration of exhaustive nearest neighbor search is performed for each map point potentially in view of the camera, but only within a small window of its projected position on the image plane. Subsequently, RANSAC iterations are performed on the full set of feature matches to remove any remaining outliers, and a motion-only BA is performed wherein the current camera pose is optimized based on the feature matches to a fixed set of 3D map points.

After tracking the stereo image pair as described above, GEOSLAM decides whether or not the candidate keyframe \mathcal{I}_i^{lr} must be selected as a keyframe. This decision is made based on the number of map points that were matched to the image features, and the distance traveled by the platform since the last keyframe was chosen. New keyframes are not spawned if the platform is nearly stationary. If the platform is in motion, and the number of feature matches to the map drops below a threshold, then the candidate keyframe \mathcal{I}_i^{lr} is chosen as keyframe K_i , windowed BA is performed over N levels of covisibility, and the unmatched stereo features in K_i are spawned as new map points. Additionally, if \mathcal{I}_i^{lr} is selected to be K_i , then the set of measurements from the features matched to the map, denoted $\mathbf{z}_i^{lr} \triangleq \{z_{im}^{lr} : m \in \mathcal{M}_i\}$, along with their SIFT descriptors, are passed on to the map module for storage, future feature matching, and processing in the windowed BA routine.

In a visual-only SLAM system, the state vector \mathbf{X}_i is optimized with respect to the measurement vector \mathbf{Z}_i , defined as

$$\mathbf{Z}_i \triangleq \left[\left\{ \mathbf{z}_n^{lr} : n \in \text{cov}(i, N) \right\} \right]^T.$$

The windowed BA routine in GEOSLAM minimizes the 3D-to-2D reprojection error. The error term e_{nm} for observation of map point p_m in the stereo keyframe K_n is given as

$$e_{nm} = z_{nm}^{lr} - \Pi(\mathbf{x}_{p_m}^S, \mathbf{x}_{C_n}^S, \boldsymbol{\theta}_{C_n}^S),$$

where Π is the projection function for an undistorted and rectified stereo camera

$$\Pi(\mathbf{x}_{p_m}^S, \mathbf{x}_{C_n}^S, \boldsymbol{\theta}_{C_n}^S) = \begin{bmatrix} f \frac{x_{nm}^S}{z_{nm}^S} + c_u \\ f \frac{y_{nm}^S}{z_{nm}^S} + c_v \\ f \frac{x_{nm}^S}{z_{nm}^S} + c_u - bf \end{bmatrix},$$

$$[x_{nm}, y_{nm}, z_{nm}]^T = R(\boldsymbol{\theta}_{C_n}^S)^T (\mathbf{x}_{p_m}^S - \mathbf{x}_{C_n}^S),$$

in which $R(\cdot)$ denotes the rotation matrix corresponding to the argument angle-axis vector, and f , (c_u, c_v) , and b are the focal length, the principal point, and the baseline distance between the left and right cameras of the rectified stereo camera model, respectively. The cost function to be minimized for visual SLAM is given as

$$C_i = \sum_{n \in \text{cov}(i, N)} \sum_{m \in \mathcal{M}_n} \rho(e_{nm}^T \Omega_{nm}^{-1} e_{nm}),$$

where ρ may be the standard least squares cost function $\rho(\cdot) = (\cdot)$ or a more robust cost function such as the Huber or Tukey cost functions, and where $\Omega_{nm} = \sigma_{nm}^2 I_{3 \times 3}$ is the covariance of the feature measurements.

GEOSLAM performs BA minimization via Google's *ceres-solver*. The automatic differentiation feature of *ceres-solver* is used to compute the Jacobian for the measurement model.

An important feature of the GEOSLAM visual pipeline is the ability to merge maps from multiple mapping sessions. This is embodied in an algorithm similar to the loop closure technique from the visual SLAM literature [60]. Map merging is described in detail in Section 4.3.2.

4.2. GNSS Aiding

Conventional visual SLAM algorithms are known to drift from the true platform trajectory as a function of the distance traveled by the platform. Furthermore, the map of the structure is created in the arbitrary \mathcal{S} frame. Such a map cannot be intelligibly shared with another mapping agent having a different \mathcal{S} frame. Meanwhile, GNSS position estimates are obtained in the global \mathcal{G} frame and do not exhibit any distance-dependent drift. Accordingly, GEOSLAM ingests standard GNSS position estimates from a software-defined GNSS receiver, called GRID/pprx [6,61], in a tightly-coupled architecture to create a globally-referenced map, enable cooperative multi-session mapping, and constrain the drift of visual SLAM. Since the stereo camera setup is triggered by the same clock that drives digitization of the GNSS samples (see Figure 7), it is possible to produce GNSS measurements synchronized with the camera image epochs. This section details the various coordinate frames in GNSS-aided visual SLAM, the updated BA cost function, and an initialization routine required to enable GNSS aiding in SLAM.

4.2.1. Coordinate Frames

The GNSS-aided visual SLAM system has three coordinate frames of interest: the K_i camera frame C_i , the local SLAM frame \mathcal{S} , and the global frame \mathcal{G} . The SLAM frame \mathcal{S} adopts the position and orientation of the first keyframe prior to optimization as its origin and orientation. Thus, \mathcal{S} is fixed relative to \mathcal{G} , but each C_i changes relative to \mathcal{G} as the platform moves.

Note that the structure-from-motion states in Equation (4) are represented in the \mathcal{S} frame, whereas the K_n th keyframe's corresponding GNSS measurement, denoted $z_{\mathcal{A}_n}^{\mathcal{G}} \in \mathbb{R}^3$, is natively represented in the \mathcal{G} frame. The latter is transformed to the \mathcal{S} frame through an unknown but fixed rotation, $R_{\mathcal{G}}^{\mathcal{S}} \in SO(3)$, and translation, $t_{\mathcal{G}}^{\mathcal{S}} \in \mathbb{R}^3$. This transformation is estimated at initialization as explained in Section 4.2.2. After initialization, $z_{\mathcal{A}_n}^{\mathcal{G}}$ is rendered in \mathcal{S} as

$$z_{\mathcal{A}_n}^{\mathcal{S}} = R_{\mathcal{G}}^{\mathcal{S}} z_{\mathcal{A}_n}^{\mathcal{G}} + t_{\mathcal{G}}^{\mathcal{S}}.$$

GEOSLAM estimates the 6-DoF pose of the left camera, but the GNSS antenna phase center is not co-located with the camera center; rather, it is offset from the camera center by a fixed vector (same for all n) in C_n denoted, $t_{\mathcal{A}_n}^{\mathcal{C}_n} \in \mathbb{R}^3$. Thus, the error term associated with the GNSS position estimate for K_n is given as

$$e_{\mathcal{A}_n} = z_{\mathcal{A}_n}^{\mathcal{S}} - \left(x_{\mathcal{C}_n}^{\mathcal{S}} + R(\theta_{\mathcal{C}_n}^{\mathcal{S}}) t_{\mathcal{A}_n}^{\mathcal{C}_n} \right).$$

Under the assumption of temporally-uncorrelated GNSS errors, the updated BA cost function to be minimized is

$$C_i = \sum_{n \in \text{cov}(i, N)} \left[\sum_{m \in \mathcal{M}_n} \rho \left(e_{nm}^T \Omega_{nm}^{-1} e_{nm} \right) + e_{\mathcal{A}_n}^T \Gamma_n^{-1} e_{\mathcal{A}_n} \right],$$

where $\Gamma_n = R_{\mathcal{G}}^{\mathcal{S}} \Gamma'_n (R_{\mathcal{G}}^{\mathcal{S}})^T$ and Γ'_n is the covariance matrix of the GNSS position estimate associated with K_n , expressed in \mathcal{G} .

4.2.2. Initialization in GNSS-Aided SLAM

When initializing, GEOSLAM performs visual-only SLAM for the first N_i keyframes in the \mathcal{S} frame, and stores the GNSS position measurements of the antenna provided in the \mathcal{G} frame. Subsequently, GEOSLAM finds the least-squares Euclidean transformation to obtain the optimal rotation matrix $R_{\mathcal{G}}^{\mathcal{S}}$ and translation vector $t_{\mathcal{G}}^{\mathcal{S}}$ between the two coordinate systems from the set of vector observations. Note that a full similarity transformation is not required since the known stereo baseline renders the \mathcal{S} frame with correct scaling. The estimated Euclidean transformation minimizes the squared difference between the transformed GNSS measurements in \mathcal{S} and the visual SLAM predicted trajectory of the GNSS antenna, also in \mathcal{S} . The specific method used to estimate the transformation is based on SVD decomposition as discussed in [62].

$$\left(R_{\mathcal{G}}^{\mathcal{S}}, t_{\mathcal{G}}^{\mathcal{S}} \right) = \arg \min_{R \in SO(3), t \in \mathbb{R}^3} \sum_{n=1}^{N_i} \left\| \left(R z_{\mathcal{A}_n}^{\mathcal{G}} + t \right) - \left(x_{\mathcal{C}_n}^{\mathcal{S}} + R(\theta_{\mathcal{C}_n}^{\mathcal{S}}) t_{\mathcal{A}_n}^{\mathcal{C}_n} \right) \right\|^2 \quad (5)$$

It must be noted that this transformation need only be approximately correct such that the GNSS estimates, used as measurements, will not diverge with respect to the visually-derived trajectory. Because the jointly estimated trajectory in \mathcal{S} gets transformed back to \mathcal{G} using the same approximate transformation, any errors in the transformation are cancelled.

4.3. Multi-Session Mapping

Refinement of the visual feature map over multiple sessions with time-separated GNSS measurements is central to the idea of approaching the accuracy limit of mapping with standard GNSS. Consider a vehicle revisiting an area mapped previously in one or more sessions. When GEOSLAM matches greater than T features in the current keyframe to the features already present in the prior map, the keyframes from the previous sessions in that section of the map are included in the covisibility window of the current keyframe. After such a merge is detected and verified, a BA may be performed on the covisible keyframes from multiple sessions to average time-separated standard GNSS errors. It is important to note that multi-session mapping can only be realized when sufficient feature matches are found between multiple sessions. This is not a straightforward task, as evidenced by recent efforts on lifelong feature mapping efforts [63]. This issue is further discussed in Section 5. In the current section, multi-session map database management and map merging are discussed.

4.3.1. Map Database

Storage and reuse of maps is a pre-requisite for multi-session mapping. For a given session, the SLAM map is created in the \mathcal{S} frame. However, such a map is not readily usable in successive mapping sessions since the \mathcal{S} frame is distinct for each session. Fortunately, the integration of visual SLAM with GNSS enables transformation of the SLAM map in to the \mathcal{G} frame.

At the end of the p th mapping session in the local frame \mathcal{S}_p , GEOSLAM stores the data to a map database after applying the $(R_{\mathcal{G}}^{\mathcal{S}_p}, \mathbf{t}_{\mathcal{G}}^{\mathcal{S}_p})$ transformation, as estimated during initialization for the p th session, to all map point positions, all keyframe poses, and all GNSS measurements associated with each keyframe:

$$\begin{aligned}\mathbf{x}_{\mathcal{C}_n}^{\mathcal{G}} &= (R_{\mathcal{G}}^{\mathcal{S}_p})^T (\mathbf{x}_{\mathcal{C}_n}^{\mathcal{S}_p} - \mathbf{t}_{\mathcal{G}}^{\mathcal{S}_p}); & R(\boldsymbol{\theta}_{\mathcal{C}_n}^{\mathcal{G}}) &= (R_{\mathcal{G}}^{\mathcal{S}_p})^T R(\boldsymbol{\theta}_{\mathcal{C}_n}^{\mathcal{S}_p}), \\ \mathbf{x}_{p_m}^{\mathcal{G}} &= (R_{\mathcal{G}}^{\mathcal{S}_p})^T (\mathbf{x}_{p_m}^{\mathcal{S}_p} - \mathbf{t}_{\mathcal{G}}^{\mathcal{S}_p}), \\ \mathbf{z}_{\mathcal{C}_n}^{\mathcal{G}} &= (R_{\mathcal{G}}^{\mathcal{S}_p})^T (\mathbf{z}_{\mathcal{C}_n}^{\mathcal{S}_p} - \mathbf{t}_{\mathcal{G}}^{\mathcal{S}_p}).\end{aligned}$$

At the beginning of the $(p + 1)$ th session, GEOSLAM again estimates the $(R_{\mathcal{G}}^{\mathcal{S}_{p+1}}, \mathbf{t}_{\mathcal{G}}^{\mathcal{S}_{p+1}})$ transformation during initialization. The map database from previous session(s) is then loaded after applying the transformation for the $p + 1$ session, such that the prior map points, keyframes, and measurements are rendered in the \mathcal{S}_{p+1} frame:

$$\begin{aligned}\mathbf{x}_{\mathcal{C}_n}^{\mathcal{S}_{p+1}} &= R_{\mathcal{G}}^{\mathcal{S}_{p+1}} \mathbf{x}_{\mathcal{C}_n}^{\mathcal{G}} + \mathbf{t}_{\mathcal{G}}^{\mathcal{S}_{p+1}}; & R(\boldsymbol{\theta}_{\mathcal{C}_n}^{\mathcal{S}_{p+1}}) &= (R_{\mathcal{G}}^{\mathcal{S}_{p+1}}) R(\boldsymbol{\theta}_{\mathcal{C}_n}^{\mathcal{G}}), \\ \mathbf{x}_{p_m}^{\mathcal{S}_{p+1}} &= R_{\mathcal{G}}^{\mathcal{S}_{p+1}} \mathbf{x}_{p_m}^{\mathcal{G}} + \mathbf{t}_{\mathcal{G}}^{\mathcal{S}_{p+1}}, \\ \mathbf{z}_{\mathcal{C}_n}^{\mathcal{S}_{p+1}} &= R_{\mathcal{G}}^{\mathcal{S}_{p+1}} \mathbf{z}_{\mathcal{C}_n}^{\mathcal{G}} + \mathbf{t}_{\mathcal{G}}^{\mathcal{S}_{p+1}}.\end{aligned}$$

After loading the prior map, the standard GEOSLAM pipeline is executed for each stereo image pair in the $(p + 1)$ th session. In addition, GEOSLAM attempts to detect if the vehicle is currently passing through a previously-mapped region. If so, a map merge is declared and the current and prior keyframes are jointly optimized, as detailed in Section 4.3.2. Finally, at the end of the mapping session, the combined map is stored back in the database as described before.

4.3.2. Map Merging

As mentioned before, the matching of feature points across multiple sessions is central to the idea of averaging standard GNSS errors. Once sufficiently many features are matched between the current stereo keyframes and prior map points, GEOSLAM declares a map merging event. This is akin to the well-known problem of detecting loop closure in the visual SLAM literature [60]. This section details GEOSLAM's map merging and loop closing routine. Hereafter, the terms map merging and loop closure are used interchangeably since GEOSLAM treats them identically.

First, note that when detecting a map merge event, feature matching must be attempted against map points that have not been matched in the most recent keyframes. Thus, after processing the i th keyframe, a possible merge is checked for against the set of map points $\{m : m \notin \cup_{n \in \text{cov}(i, N)} \mathcal{M}_n\}$. If this bag-of-words-style feature matching succeeds, then RANSAC iterations are performed to determine whether the matches are geometrically consistent, as well as to robustly estimate the camera pose $(\check{x}_{C_i}^S, \check{\theta}_{C_i}^S)$ implied by the merge event. If enough inliers are found, the map merge routine is executed.

The map merging process is depicted visually in Figures 8–10. A typical merge situation is shown in Figure 8, where the platform pose at the i th keyframe is inconsistent with the prior map at the merge location. To avoid such discontinuity in the ensuing joint BA, as an initial guess GEOSLAM enforces that the i th keyframe pose be consistent with the pose implied by the visual merge matches $(\check{x}_{C_i}^S, \check{\theta}_{C_i}^S)$, and that the keyframes and map points from the prior session(s) be unchanged. To this end, a pose-graph optimization [64] is performed over a large N_m -level covisibility window for K_i , where the relative translations and rotations between covisible keyframes, as estimated in the current session, are provided as delta-pose measurements, while the 6-DoF poses for the terminal nodes in the covisibility window, as well as for K_i , are held constant. In particular, let \mathcal{K}_0 denote the set of terminal keyframes in the covisibility graph $\text{cov}(i, N_m)$. Furthermore, define the delta-pose pseudo-measurements

$$\begin{aligned} \delta x_{nk}^S &\triangleq \hat{x}_{C_n}^S - \hat{x}_{C_k}^S, \\ \delta \theta_{nk}^S &\triangleq \theta \left(R \left(\hat{\theta}_{C_n}^S \right) R \left(\hat{\theta}_{C_k}^S \right)^T \right), \end{aligned}$$

where the superscript (\cdot) denotes GEOSLAM's estimate of the state before the merge event, and $\theta(\cdot)$ denotes the angle-axis representation of the input rotation matrix. The pose-graph optimization minimizes the following cost function with respect to $(x_{C_n}^S, \theta_{C_n}^S) \forall n \in \text{cov}(i, N_m)$:

$$C = \sum_{n \in \text{cov}(i, N_m)} \sum_{k \in \text{cov}(n, 1)} \left[\left\| \left(x_{C_n}^S - x_{C_k}^S \right) - \delta x_{nk}^S \right\|_{P_{\delta x}^{-1}}^2 + \left\| \theta \left(R \left(\theta_{C_n}^S \right) R \left(\theta_{C_k}^S \right)^T \right) - \delta \theta_{nk}^S \right\|_{P_{\delta \theta}^{-1}}^2 \right],$$

where $\|q\|_P^2 = q^T P q$, with the following constraints:

$$\begin{aligned} (x_{C_i}^S, \theta_{C_i}^S) &= (\check{x}_{C_i}^S, \check{\theta}_{C_i}^S), \\ (x_{C_n}^S, \theta_{C_n}^S) &= (\hat{x}_{C_n}^S, \hat{\theta}_{C_n}^S) \forall n \in \mathcal{K}_0. \end{aligned}$$

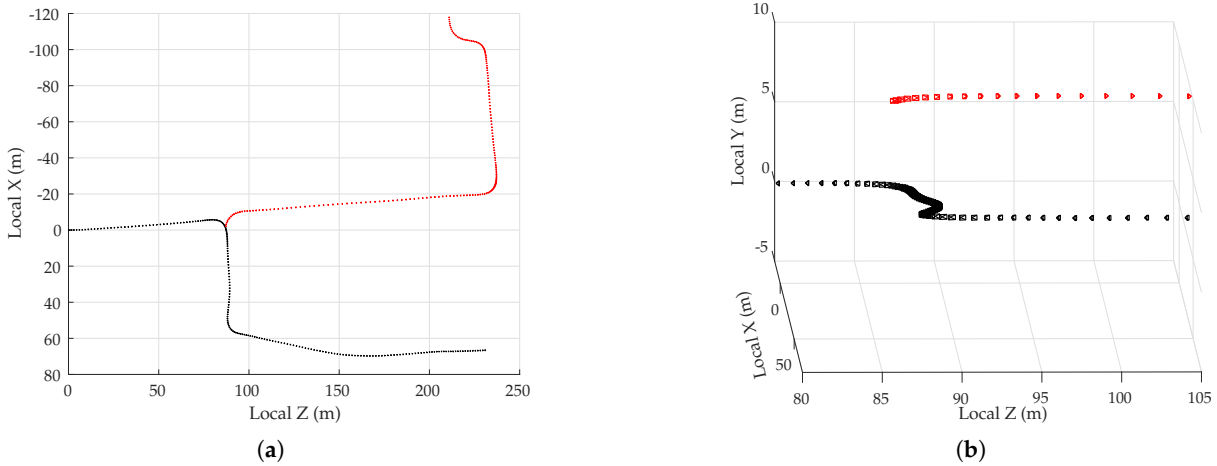


Figure 8. GEOSLAM trajectories at the instant when a merge has been detected and verified. The cameras colored black are keyframes from a prior map, and those colored red are from the current session. (a) Top view of the trajectories. (b) View from 5° elevation showing a discontinuity in the vertical component.

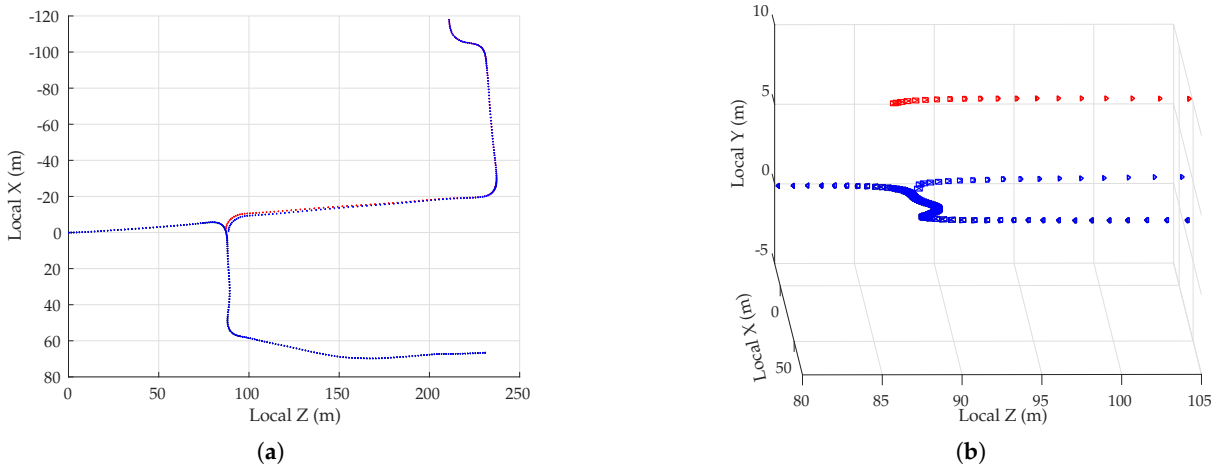


Figure 9. GEOSLAM trajectories post-pose-graph optimization (in blue), overlaid on the corresponding (black and red) trajectories from Figure 8. All keyframes are colored blue at this stage since prior and current keyframes are now connected. (a) Top view of the trajectories. Note that the discontinuity at the merge location is smoothly distributed across N_m levels of covisibility in the current session, and that the keyframe poses from the prior map are unchanged at this stage. (b) View from 5° elevation. Keyframes from the current trajectory have been adjusted to remove the discontinuity, blue and black keyframes exactly overlap. Not shown: the corresponding map points in the current session are also adjusted to match the updated keyframe poses.

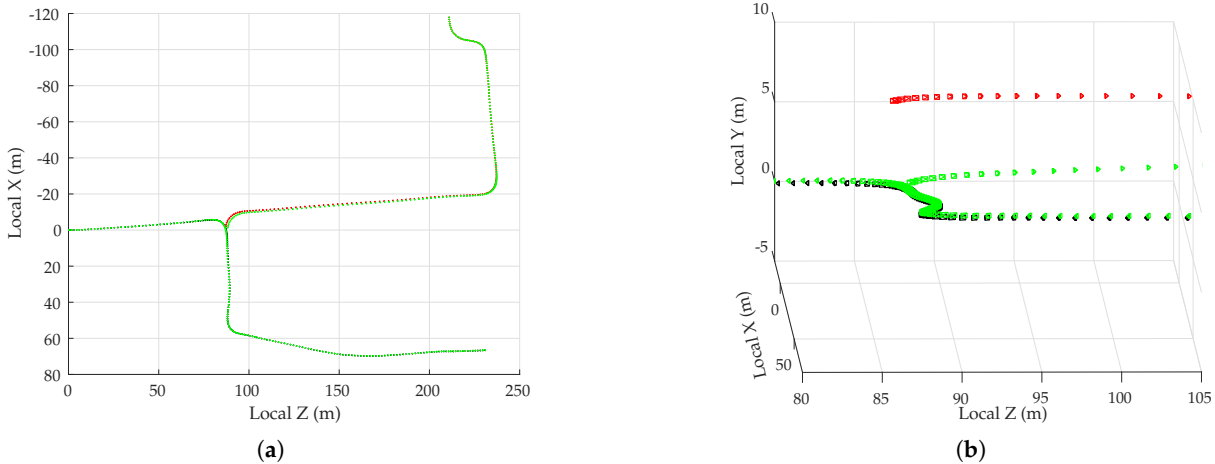


Figure 10. GEOSLAM trajectories after joint BA of current and prior keyframes (in green), overlaid on the corresponding (black and red) trajectories from Figure 8. (a) Top view of the trajectories. Note that both the current and prior keyframes (and map points, not shown) have been adjusted to optimally minimize the BA cost function over N_m levels of covisibility. (b) View from 5° elevation.

As a result of this pose-graph optimization, any discontinuity at the merge location between the prior and current keyframes is smoothed out, as shown in Figure 9. Subsequently, the map points as seen in the current keyframes are also adjusted in accordance to the pose-graph optimization. Finally, the duplicated map points near the merge location are fused together. As a result of shared feature matches between the current and prior keyframes, the updated covisibility window for K_i includes keyframes from prior session(s).

The merged map is then optimized in a windowed BA, this time with feature point coordinates and GNSS positions as measurements. Note that this is a joint windowed BA with both current and prior keyframes and map points. As a result, both the current and prior states are appropriately adjusted based on the number and covariance of the feature point and GNSS measurements. The result of the map merging routine is shown in Figure 10.

5. Empirical Results

To validate the results obtained in the above analyses, GNSS and visual data were collected in a moderate urban area north of the University of Texas at Austin campus in Austin, TX. This section presents the data collection setup, error statistics of various flavors of code-phase GNSS positioning, and results from GEOSLAM's multi-session GNSS-aided-visual mapping.

5.1. Rover and Reference Platforms

The rover GNSS receiver is one among several sensors housed in an integrated perception platform called the University of Texas Sensorium [6]. Designed for connected and automated vehicle research, the Sensorium is a self-contained sensor housing that can be mounted atop any standard passenger vehicle. Two Antcom G8Ant-3A4TNB1 triple-frequency patch antennas are flush-mounted in the cross-track direction on the Sensorium's upper plate, separated by just over one meter. The antennas' signals are routed to a unified radio frequency (RF) front end whose output intermediate frequency (IF) samples are processed in real-time (to within less than 10 ms latency) by the Sensorium's onboard computer.

The samples are also stored to disk for post-processing. The experimental setup also includes a surveyed GNSS reference station that aids in the generation of the ground truth trajectory.

The GNSS data were processed by a software-defined GNSS receiver tracking signals from GPS L1 C/A, GPS L2CLM, Galileo E1, and SBAS. Data from both the primary (passenger's side) and secondary (driver's side) antennas were used to reconstruct a sub-dm-accurate CDGNSS-based ground truth trajectory, as described in [6]. Enhanced code-phase positioning was performed on the data from the primary antenna, incorporating precise orbit and clock products from IGS, ionospheric corrections from WAAS satellites, and the Saastamoinen model for tropospheric corrections, in addition to NIS-based exclusion of multipath signals. Double-differenced pseudorange-based positioning was also performed with the data from the primary antenna, as discussed later in this section. The code-phase-based position estimates were (i) compared against the ground truth from the primary antenna to study the code-phase positioning error statistics, and (ii) fed to GEOSLAM for vision-GNSS sensor fusion. The primary antenna feed was also input to a ublox M8T receiver for comparison against the enhanced code-phase software receiver.

The Sensorium features a front-facing stereo camera rig composed of two Basler acA2040-35gm cameras that capture synchronous stereo image pairs when triggered by a signal tied to the GNSS front-end's sampling clock. The images are captured in grayscale at 10 frames per second and timestamped by the Sensorium's computer. The cameras are configured to automatically adjust the exposure time based on lighting, while the focal length, focus, and aperture are held fixed, having been adjusted physically prior to capture.

5.2. Test Route

The test route was a 1-km loop north of the University of Texas at Austin campus in Austin, TX. The route included a variety of light-to-moderate urban conditions, from open-sky to overhanging trees to built-up areas. The Dean Keeton corridor, toward the left in Figure 11, was the most challenging stretch along the test route for GNSS positioning. It passes below a pedestrian bridge and is flanked on both sides by buildings ranging from 30 to 65 meters tall set back 28 meters from the center of the roadway.



Figure 11. An overview of the 1-km test route. The Dean Keeton corridor, toward the left, is spanned by a pedestrian bridge and flanked by buildings on both sides. A total of 75 laps of the test route were driven over six separate campaigns.

To study the code-phase-based positioning error characteristics over time-separated sessions in the same area, and to perform multi-session mapping with GEOSLAM, multiple laps of the test route were driven over six separate campaigns. The first two campaigns were conducted on 21 December 2017 and

15 January 2018, while the other four campaigns were conducted in pairs of two on 3 June 2018 and 4 June 2018. The GNSS error charts are presented for a total of 75 laps of the test route, while multi-session mapping with GEOSLAM was performed over eight laps/sessions of data from the four latest campaigns.

Imagery collected over the four June 2018 campaigns exhibits appreciable visual diversity, offering a real-world challenge to multi-session GEOSLAM operation. Figure 12a,b show the variation in lighting and visual features between the data collected on 3 June 2018 and 4 June 2018.



Figure 12. Different visual conditions on two days of data collection. (a) An image captured on the first day of data collection. Note the sharp shadows and absence of parked cars. (b) An image captured on the second day of data collection. Note the absence of sharp shadows and complete blockage of curb due to parked cars.

5.3. Empirical GNSS Error Analysis

Figure 13 shows the error in the enhanced code-phase GNSS position solutions with respect to the ground truth. The error is plotted versus the distance along the 1-km loop. The beginning of this loop was taken to be immediately after the overhead pedestrian bridge along the Dean Keeton corridor. It is observed that the enhanced code-phase GNSS errors are clustered separately for each of the campaigns, and that each cluster is offset from zero by as much as 1 m in the horizontal plane. Such error characteristics are representative of ionospheric modeling errors, which have a long decorrelation time. It is also evident that the error variance was larger as the receiver exits the challenging portion of the loop at which point the tracking loops were recovering from signal loss under the bridge. The effect was especially pronounced in the vertical direction. Figure 14 shows similar errors for the commercial ublox M8T receiver. The error traces from the ublox receiver show a wider spread than the enhanced code-phase receiver, likely due to lack of precise orbit and clock corrections.

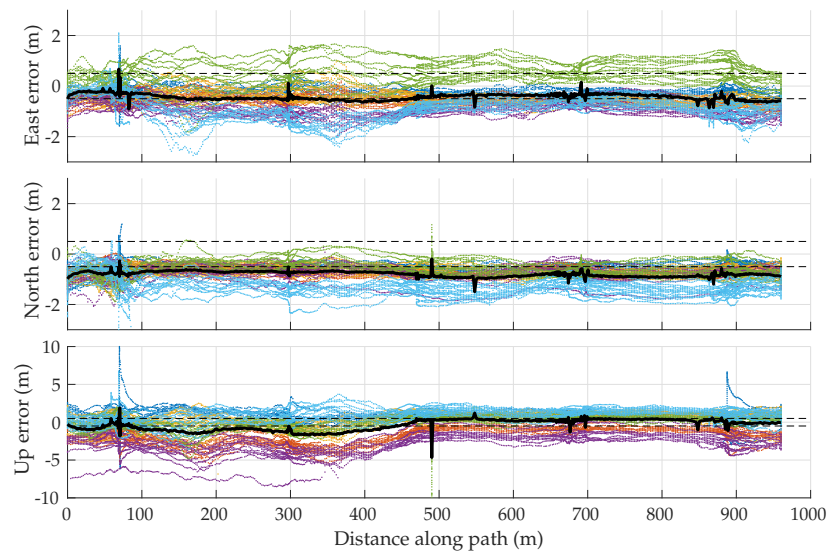


Figure 13. Errors in enhanced code-phase position estimates with respect to ground truth in the east, north, and up directions. Different colors distinguish data from six different campaigns. The dashed reference lines are drawn at ± 50 cm. The solid black lines show the mean positioning error over the six campaigns. The error standard deviation is nearly constant along the path in the horizontal plane at ~ 0.6 m in the east and ≈ 0.4 m in the north direction. In the up direction, the standard deviation is ~ 2.1 m for the first 400 m along the path, and ≈ 1.3 m for the rest.

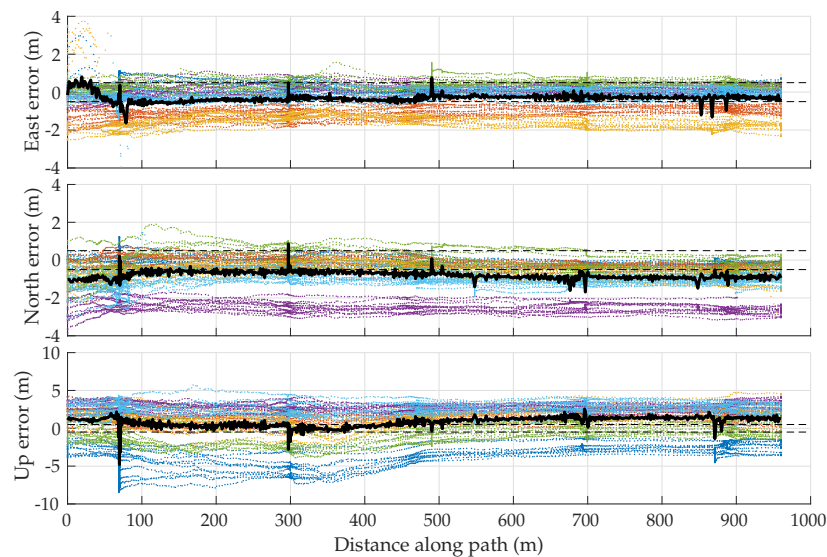


Figure 14. Errors in ublox M8T position estimates with respect to ground truth in the east, north, and up directions. Different colors distinguish data from six different campaigns. Dashed reference lines are drawn at ± 50 cm. The solid black lines show the mean positioning error over the six campaigns. The error standard deviation in the east is ~ 1.5 m over the first 100 m along the path and ~ 0.7 m over the rest; ~ 0.9 m in the north; and ~ 2.7 m over the first 400 m and ~ 2 m over the rest in the up direction.

On the basis of Figures 13 and 14, one might be tempted to conclude that errors in enhanced code-phase and stand-alone GNSS navigation solutions are substantially non-zero-mean, especially in the

north and up directions, despite the overhead GNSS constellation changing substantially between sessions. It certainly appears that the permanent structures (buildings, bridge) along the test loop left a bias in the vertical direction during the first 400 m along the loop. However, the bias in the north direction, and to a lesser extent in the east, may only be an artifact of the small sample size: ionospheric modeling errors were not yet averaged down to nearly zero in the east and ~ 30 cm in the north, as one would expect from the WAAS ionospheric model (see Table 1).

Given that the asymptotic properties of ionospheric modeling errors are better understood than those of multipath errors, it is instructive to eliminate, insofar as possible, all ionospheric modeling errors from the along-track error histories. To this end, a differential code phase GNSS technique was applied whereby the navigation solution was based on double-difference pseudorange measurements using data from a nearby reference station at a precisely known location. Such double differencing over a short 1-km baseline eliminates virtually all ionospheric and tropospheric errors, but does nothing to reduce vehicle-side multipath. Thus, one can empirically examine multipath effects in isolation from ionospheric effects.

Figure 15 shows the results of this study based on all six data capture campaigns. Note that biases for all components are much smaller. It appears that for the test route chosen, non-zero-mean horizontal errors in the enhanced code phase positions were almost entirely driven by ionospheric modeling errors, and not by persistent effects of multipath due to the permanent structures along the test route. This is broadly consistent with the analyses presented earlier in this paper on position-domain biases due to ionospheric and multipath errors. However, it does appear that a bias due to multipath remained in the vertical direction over the first 400 m, even when ionospheric errors were removed. Apparently, the arrangement of buildings over this segment caused non-line-of-sight effects that did not average away. Mercifully, horizontal errors, which appear to be close to zero-mean over the six campaigns, matter most for high-accuracy digital mapping, since obstacle avoidance and vehicle coordination are largely 2-D problems, and since multiple vehicles can straightforwardly agree on a particular feature's relative vertical position from an inferred road surface.

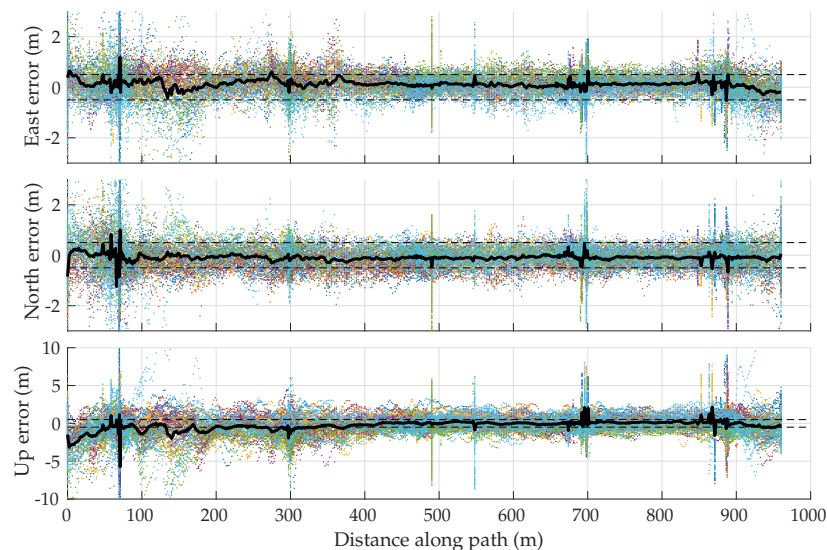


Figure 15. Errors in double-differenced pseudorange-based position estimates with respect to ground truth in the east, north, and up directions. Different colors distinguish data from six different campaigns. Dashed reference lines are drawn at ± 50 cm. The solid black lines show the mean positioning error over the six campaigns. The error standard deviation in the east and north directions is ~ 0.9 m over the first 200 m along the path and ~ 0.4 m over the rest. In the up direction, the standard deviation is ~ 1.9 m over the first 400 m and ~ 1 m over the rest.

Based on Figure 15, one can conclude that multi-session averaging with a sufficiently accurate ionospheric model, such as the Fast PPP model, yields sub-50-cm global referencing accuracy for digital maps in the horizontal plane with code-phase-based GNSS, even in the presence of persistent multipath.

5.4. Multi-Session Mapping Results

GEOSLAM processed two laps/sessions of data from each of the four campaigns conducted on 4 and 5 June, fusing the visual data from the captured images with the double-differenced pseudorange-based position estimates of the primary antenna. Figure 16 summarizes the result from GEOSLAM's multi-session GNSS-aided-visual SLAM. The black data points denote the difference between the ground truth trajectory of the primary GNSS antenna and GEOSLAM's estimate of the same in local east, north, and up directions for all eight sessions. The gray data points denote the difference between the ground truth trajectory of the primary GNSS antenna and the coincident double-differenced pseudorange-based estimate of the same for all eight sessions.

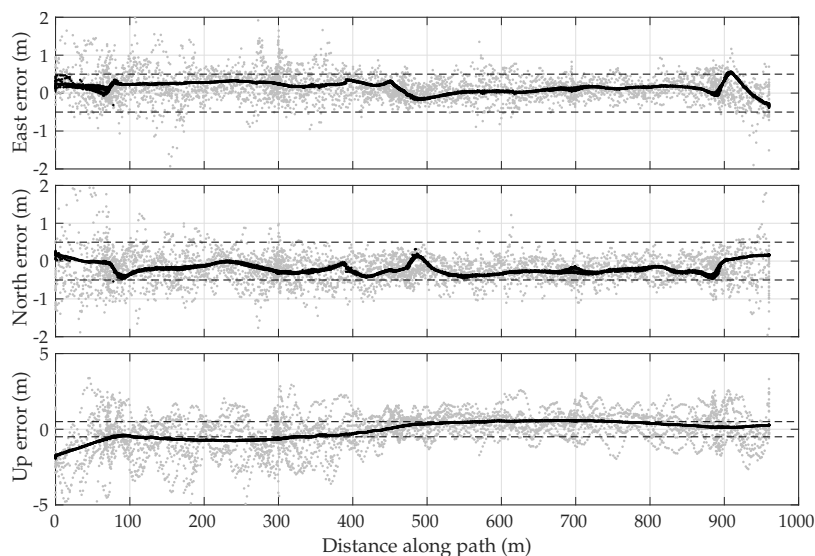


Figure 16. Errors in GEOSLAM's estimate of the primary antenna position (in black) with respect to ground truth in the east, north, and up directions for eight mapping sessions from four different data collection campaigns. The errors in double-differenced pseudorange-based primary antenna position estimates for each of the eight sessions, fed as measurements to GEOSLAM, are plotted in gray for reference. Dashed reference lines are drawn at ± 50 cm.

As one might expect, the error in GEOSLAM's estimate of the antenna position was approximately the same as the average double-differenced pseudorange-based error over eight sessions. Furthermore, due to the approximately zero-mean nature of the double-differenced pseudorange-based estimates, the GEOSLAM estimate of the trajectory was within 50 cm of the truth trajectory in the horizontal plane. Note that the error in GEOSLAM's position estimate was highly repeatable over eight different sessions, so much so that it appears there is a single black trace in Figure 16, while in truth eight independent traces were plotted. This indicates that (i) the localization of the vehicle within the visual map was highly precise: GEOSLAM made the same errors with respect to ground truth over eight different sessions; and (ii) the visual map was merged across eight sessions from four different campaigns: if the maps from any two campaigns were not merged through visual matching of features, then the GNSS position estimate for a

keyframe from one campaign would not affect another keyframe from a different campaign since they would not be covisible, and thus the eight black traces in Figure 16 would not overlap.

6. Conclusions

The accuracy limits of collaborative global referencing of digital maps with standard GNSS were explored through simulation and real data. The asymptotic average of position errors due to thermal noise, satellite orbit and clock errors, and tropospheric modeling errors were assumed to be negligible. It has been shown that the position error due to inaccurate ionospheric modeling may lead to persistent dm-level biases in the horizontal position if the corrections are sourced from the IGS GIM, but other recent models such as the Fast PPP IONEX GIM perform better in this regard. Multipath errors persist with multiple mapping sessions through the same urban corridor and may not be zero mean. With adequate multipath exclusion, persistent multipath biases may be reduced below 50 cm on average. In conclusion, sub-50-cm accurate digital mapping has been shown to be feasible in the horizontal plane after multiple mapping sessions with code-phase-based GNSS, but larger biases persist in the vertical direction. A globally-referenced electro-optical SLAM pipeline, termed GEOSLAM, has been detailed and demonstrated to achieve sub-50-cm horizontal localization accuracy in a moderate urban environment by incorporating code-phase-based GNSS position estimates in the visual SLAM framework and jointly optimizing maps merged across time-separated sessions.

Author Contributions: Conceptualization, L.N. and T.E.H.; Data curation, L.N., J.M.W., M.J.M. and D.M.L.; Funding acquisition, T.E.H.; Methodology, L.N.; Project administration, T.E.H.; Software, L.N., J.M.W. and M.J.M.; Supervision, T.E.H.; Writing—original draft, L.N.; Writing—review & editing, L.N. and T.E.H.

Funding: This work has been supported by the Data-supported Transportation Operations and Planning Center (DSTOP), a Tier 1 USDOT University Transportation Center, and by funding from Toyota, Honda, and Qualcomm through the University of Texas Situation-Aware Vehicular Engineering Systems (SAVES) Center (<http://utsaves.org/>), an initiative of the Wireless Networking and Communications Group.

Conflicts of Interest: The authors declare no conflict of interest. The funding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

References

1. Wymeersch, H.; Seco-Granados, G.; Destino, G.; Dardari, D.; Tufvesson, F. 5G mmWave Positioning for Vehicular Networks. *IEEE Wirel. Commun.* **2017**, *24*, 80–86. [[CrossRef](#)]
2. Kenney, J.B. Dedicated short-range communications (DSRC) standards in the United States. *Proc. IEEE* **2011**, *99*, 1162–1182. [[CrossRef](#)]
3. Choi, J.; Va, V.; Gonzalez-Prelcic, N.; Daniels, R.; Bhat, C.R.; Heath, R.W. Millimeter-wave vehicular communication to support massive automotive sensing. *IEEE Commun. Mag.* **2016**, *54*, 160–167. [[CrossRef](#)]
4. Levinson, J.; Montemerlo, M.; Thrun, S. *Map-Based Precision Vehicle Localization in Urban Environments*; Robotics: Science and Systems; MIT Press: Cambridge, MA, USA, 2007; Volume 4, p. 1.
5. Levinson, J.; Thrun, S. Robust vehicle localization in urban environments using probabilistic maps. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation (ICRA), Anchorage, Alaska, 3–8 May 2010; pp. 4372–4378.
6. Humphreys, T.E.; Murrian, M.; Narula, L. Low-cost Precise Vehicular Positioning in Urban Environments. In Proceedings of the IEEE/ION PLANS Meeting, Monterey, CA, USA, 23–26 April 2018.
7. Hutton, J.J.; Gopaul, N.; Zhang, X.; Wang, J.; Menon, V.; Rieck, D.; Kipka, A.; Pastor, F. Centimeter-Level Robust GnsS-Aided Inertial Post-Processing for Mobile Mapping without Local Reference Stations. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2016**, *XLI-B3*, 819–826. [[CrossRef](#)]

8. Rogers, S. Creating and evaluating highly accurate maps with probe vehicles. In Proceedings of the 2000 IEEE Intelligent Transportation Systems, Dearborn, MI, USA, 1–3 October 2000; pp. 125–130.
9. Guo, T.; Iwamura, K.; Koga, M. Towards high accuracy road maps generation from massive GPS traces data. In Proceedings of the 2007 IEEE International Geoscience and Remote Sensing Symposium, Barcelona, Spain, 23–28 July 2007; pp. 667–670.
10. Knoop, V.L.; de Bakker, P.F.; Tiberius, C.C.; van Arem, B. Lane determination with GPS precise point positioning. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 2503–2513. [[CrossRef](#)]
11. Circiu, M.S.; Meurer, M.; Felux, M.; Gerbeth, D.; Thöler, S.; Vergara, M.; Enneking, C.; Sgammini, M.; Pullen, S.; Antreich, F. Evaluation of GPS L5 and Galileo E1 and E5a Performance for Future Multifrequency and Multiconstellation GBAS. *Navigation* **2017**, *64*, 149–163. [[CrossRef](#)]
12. Kos, T.; Markezic, I.; Pokrajacic, J. Effects of multipath reception on GPS positioning performance. In Proceedings of the ELMAR-2010, Zadar, Croatia, 15–17 September 2010; pp. 399–402.
13. Le Marchand, O.; Bonnifait, P.; Ibañez-Guzmán, J.; Betaille, D.; Peyret, F. Characterization of GPS multipath for passenger vehicles across urban environments. *ATTI dell'Istituto Italiano di Navigazione* **2009**, *189*, 77–88.
14. Xie, P.; Petovello, M.G. Measuring GNSS multipath distributions in urban canyon environments. *IEEE Trans. Instrum. Meas.* **2015**, *64*, 366–377.
15. Rovira-Garcia, A.; Juan, J.; Sanz, J.; González-Casado, G.; Ibañez, D. Accuracy of ionospheric models used in GNSS and SBAS: Methodology and analysis. *J. Geod.* **2016**, *90*, 229–240. [[CrossRef](#)]
16. Böhm, J.; Möller, G.; Schindelegger, M.; Pain, G.; Weber, R. Development of an improved empirical model for slant delays in the troposphere (GPT2w). *GPS Solut.* **2015**, *19*, 433–441. [[CrossRef](#)]
17. Shepard, D.P.; Humphreys, T.E. Scalable Sub-Decimeter-Accurate 3D Reconstruction. 2017. Available online: <https://ctr.utexas.edu/wp-content/uploads/138.pdf> (accessed on 28 July 2018).
18. Shepard, D.P.; Humphreys, T.E. High-Precision Globally-Referenced Position and Attitude via a Fusion of Visual SLAM, Carrier-Phase-Based GPS, and Inertial Measurements. In Proceedings of the IEEE/ION PLANS Meeting, Monterey, CA, USA, 5–8 May 2014.
19. Pesyna, K.M., Jr. Advanced Techniques for Centimeter-Accurate GNSS Positioning on Low-Cost Mobile Platforms. Ph.D. Thesis, The University of Texas at Austin, Austin, TX, USA, 2015.
20. Bryson, M.; Sukkarieh, S. A Comparison of Feature and Pose-Based Mapping using Vision, Inertial and GPS on a UAV. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, San Francisco, CA, USA, 25–30 September 2011.
21. Ellum, C. Integration of raw GPS measurements into a bundle adjustment. *IAPRS Ser.* **2006**, *35*, 3025.
22. Kume, H.; Taketomi, T.; Sato, T.; Yokoya, N. Extrinsic Camera Parameter Estimation Using Video Images and GPS Considering GPS Positioning Accuracy. In Proceedings of the 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 23–26 August 2010; pp. 3923–3926. [[CrossRef](#)]
23. Lhuillier, M. Incremental Fusion of Structure-from-Motion and GPS Using Constrained Bundle Adjustments. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 2489–2495. [[CrossRef](#)] [[PubMed](#)]
24. Lhuillier, M. Fusion of GPS and structure-from-motion using constrained bundle adjustments. In Proceedings of the CVPR 2011, Colorado Springs, CO, USA, 20–25 June 2011; pp. 3025–3032. [[CrossRef](#)]
25. Soloviev, A.; Venable, D. Integration of GPS and Vision Measurements for Navigation in GPS Challenged Environments. In Proceedings of the IEEE/ION PLANS Meeting, IEEE/Institute of Navigation, Indian Wells, CA, USA, 4–6 May 2010; pp. 826–833.
26. Aumayer, B.M. Ultra-tightly Coupled Vision/GNSS for Automotive Applications. Ph.D. Thesis, University of Calgary, Calgary, AB, Canada, 2016.
27. Chu, T.; Guo, N.; Backén, S.; Akos, D. Monocular camera/IMU/GNSS integration for ground vehicle navigation in challenging GNSS environments. *Sensors* **2012**, *12*, 3162–3185. [[CrossRef](#)] [[PubMed](#)]
28. Wang, J.J.; Kodagoda, S.; Dissanayake, G. Vision Aided GPS/INS System for Robust Land Vehicle Navigation. In Proceedings of the ION International Technical Meeting, Anaheim, CA, USA, 26–28 January 2009; pp. 600–609.
29. Sajad, S.; Michael, T.; Mae, S.; Howard, L. Multiple-robot Simultaneous Localization and Mapping: A Review. *J. Field Robot.* **2016**, *33*, 3–46. [[CrossRef](#)]

30. Zou, D.; Tan, P. CoSLAM: Collaborative Visual SLAM in Dynamic Environments. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 354–366. [[CrossRef](#)] [[PubMed](#)]
31. Forster, C.; Lynen, S.; Kneip, L.; Scaramuzza, D. Collaborative monocular SLAM with multiple Micro Aerial Vehicles. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 3962–3970. [[CrossRef](#)]
32. Piasco, N.; Marzat, J.; Sanfourche, M. Collaborative localization and formation flying using distributed stereo-vision. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 1202–1207. [[CrossRef](#)]
33. Indelman, V.; Nelson, E.; Michael, N.; Dellaert, F. Multi-robot pose graph localization and data association from unknown initial relative poses via expectation maximization. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014; pp. 593–600. [[CrossRef](#)]
34. Cunningham, A.; Wurm, K.M.; Burgard, W.; Dellaert, F. Fully distributed scalable smoothing and mapping with robust multi-robot data association. In Proceedings of the 2012 IEEE International Conference on Robotics and Automation, Saint Paul, MN, USA, 14–18 May 2012; pp. 1093–1100. [[CrossRef](#)]
35. Andersson, L.A.A.; Nygard, J. C-SAM: Multi-Robot SLAM using square root information smoothing. In Proceedings of the 2008 IEEE International Conference on Robotics and Automation, Pasadena, CA, USA, 19–23 May 2008; pp. 2798–2805. [[CrossRef](#)]
36. Dabeer, O.; Ding, W.; Gowaiker, R.; Grzechnik, S.K.; Lakshman, M.J.; Lee, S.; Reitmayr, G.; Sharma, A.; Somasundaram, K.; Sukhvasi, R.T.; Wu, X. An end-to-end system for crowdsourced 3D maps for autonomous vehicles: The mapping component. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 634–641. [[CrossRef](#)]
37. u-blox. u-blox Announces U-Blox F9 Robust and Versatile High Precision Positioning Technology for Industrial and Automotive Applications. 2018. Available online: <https://bit.ly/2GoXaOm> (accessed on 27 July 2018).
38. Murrian, M.J.; Gonzalez, C.W.; Humphreys, T.E.; Pesyna, K.M., Jr.; Shepard, D.P.; Kerns, A.J. Low-cost precise positioning for automated vehicles. *GPS World* **2016**, *27*, 32–39.
39. Misra, P.; Enge, P. *Global Positioning System: Signals, Measurements, and Performance*, 2nd rev. ed.; Ganga-Jumana Press: Lincoln, MA, USA, 2012.
40. GPS Satellite Ephemerides/Satellite & Station Clocks. Available online: <http://www.igs.org/products> (accessed on 30 March 2018).
41. Rovira-Garcia, A.; Juan, J.M.; Sanz, J.; González-Casado, G. A Worldwide Ionospheric Model for Fast Precise Point Positioning. *IEEE Trans. Geosci. Remote Sens.* **2015**, *53*, 4596–4604. [[CrossRef](#)]
42. Yunck, T.P. Chapter 21: Orbit Determination. In *Global Positioning System: Theory and Applications*; American Institute of Aeronautics and Astronautics: Washington, DC, USA, 1996; Volume 2, pp. 559–592.
43. van Bree, R.J.P.; Tiberius, C.C.J.M. Real-time single-frequency precise point positioning: Accuracy assessment. *GPS Solut.* **2012**, *16*, 259–266. [[CrossRef](#)]
44. Le, A.Q.; Tiberius, C. Single-frequency precise point positioning with optimal filtering. *GPS Solut.* **2007**, *11*, 61–69. [[CrossRef](#)]
45. Odijk, D. *Fast Precise GPS Positioning in the Presence of Ionospheric Delays*; No. 52 in Fast Precise GPS Positioning in the Presence of Ionospheric Delays, NCG, Nederlandse Commissie voor Geodesie; Delft University of Technology: Delft, The Netherlands, 2002.
46. Shi, C.; Gu, S.; Lou, Y.; Ge, M. An improved approach to model ionospheric delays for single-frequency Precise Point Positioning. *Adv. Space Res.* **2012**, *49*, 1698–1708. [[CrossRef](#)]
47. Boehm, J.; Werl, B.; Schuh, H. Troposphere mapping functions for GPS and very long baseline interferometry from European Centre for Medium-Range Weather Forecasts operational analysis data. *J. Geophys. Res. Solid Earth* **2006**, *111*. [[CrossRef](#)]
48. Böhm, J.; Niell, A.; Tregoning, P.; Schuh, H. Global Mapping Function (GMF): A new empirical mapping function based on numerical weather model data. *Geophys. Res. Lett.* **2006**, *33*. [[CrossRef](#)]
49. Lehner, A.; Steingass, A. A novel channel model for land mobile satellite navigation. In Proceedings of the ION GNSS Meeting, Long Beach, CA, USA, 13–16 September 2005; pp. 13–16.

50. Lehner, A.; Steingass, A. *Technical Note on the Land Mobile Satellite Channel Model—Interface Control Document*; Association of Radio Industries and Businesses: Tokyo, Japan, 2008.
51. Psiaki, M.; Mohiuddin, S. Modeling, analysis, and simulation of GPS carrier phase for spacecraft relative navigation. *J. Guid. Control Dyn.* **2007**, *30*, 1628. [[CrossRef](#)]
52. Braasch, M.S. *Springer Handbook of Global Navigation Satellite Systems*; Chapter Multipath; Springer: Cham, Switzerland, 2017; pp. 443–468.
53. Bar-Shalom, Y.; Li, X.R.; Kirubarajan, T. *Estimation with Applications to Tracking and Navigation*; John Wiley and Sons: New York, NY, USA, 2001.
54. Mur-Artal, R.; Tardos, J.D. ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras. *arXiv* **2016**, arXiv:1610.06475.
55. Leutenegger, S.; Lynen, S.; Bosse, M.; Siegwart, R.; Furgale, P. Keyframe-based visual-inertial odometry using nonlinear optimization. *Int. J. Robot. Res.* **2015**, *34*, 314–334. [[CrossRef](#)]
56. Engel, J.; Schöps, T.; Cremers, D. LSD-SLAM: Large-scale direct monocular SLAM. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2014; pp. 834–849.
57. Strasdat, H.; Montiel, J.; Davison, A.J. Visual SLAM: Why filter? *Image Vis. Comput.* **2012**, *30*, 65–77. [[CrossRef](#)]
58. Lowe, D.G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [[CrossRef](#)]
59. Muja, M.; Lowe, D.G. Fast approximate nearest neighbors with automatic algorithm configuration. In *Proceedings of the VISAPP (1)*, Lisboa, Portugal, 5–8 February 2009; Volume 2, p. 2.
60. Mur-Artal, R.; Montiel, J.M.M.; Tardos, J.D. ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Trans. Robot.* **2015**, *31*, 1147–1163. [[CrossRef](#)]
61. Lightsey, E.G.; Humphreys, T.E.; Bhatti, J.A.; Joplin, A.J.; O’Hanlon, B.W.; Powell, S.P. Demonstration of a Space Capable Miniature Dual Frequency GNSS Receiver. *Navig. J. Inst. Navig.* **2014**, *61*, 53–64. [[CrossRef](#)]
62. Sorkine-Hornung, O.; Rabinovich, M. Least-squares rigid motion using SVD. *Computing* **2017**, *1*, 1.
63. Mühlheller, P.; Bürki, M.; Bosse, M.; Derendarz, W.; Philippsen, R.; Furgale, P. Summary maps for lifelong visual localization. *J. Field Robot.* **2016**, *33*, 561–590. [[CrossRef](#)]
64. Strasdat, H.; Davison, A.J.; Montiel, J.M.; Konolige, K. Double window optimisation for constant time visual SLAM. In *Proceedings of the 2011 IEEE International Conference on Computer Vision (ICCV)*, Barcelona, Spain, 6–13 November 2011; pp. 2352–2359.



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Volume 2: Low-cost Precise Vehicular Positioning in Urban Environments



Low-cost Precise Vehicular Positioning in Urban Environments

Todd E. Humphreys
Aerospace Engineering
The University of Texas at Austin
todd.humphreys@mail.utexas.edu

Matthew Murrian
Aerospace Engineering
The University of Texas at Austin
murrian@utexas.edu

Lakshay Narula
Electrical and Computer Engineering
The University of Texas at Austin
narula@utexas.edu

Abstract—A system developed at The University of Texas for low-cost precise urban vehicular positioning is demonstrated to achieve a probability of correct integer fixing greater than 96.5% for a probability of incorrect integer fixing surely less than 2.3% and likely less than 1%. This is demonstrated using data captured during 3.4 hours of driving on a repeating urban test route over three separate days. The results are achieved without any aiding by inertial or electro-optical sensors. Development and evaluation of the unaided GNSS-based precise positioning system is a key milestone toward the overall goal of combining precise GNSS, vision, radar, and inertial sensing for all-weather high-integrity precise positioning for automated and connected vehicles. The system described and evaluated herein is composed of a densely-spaced reference network, a software-defined GNSS receiver whose processing can be executed on general-purpose commodity hardware, and a real-time kinematic (RTK) positioning engine. All components have been tailored in their design to yield competent sub-decimeter positioning in the mobile urban environment. A performance sensitivity analysis reveals that navigation data bit prediction on the GPS L1 C/A signals is key to high-performance urban RTK positioning.

Keywords—urban vehicular positioning; CDGNSS; low-cost RTK positioning.

I. INTRODUCTION

Few of the leading self-driving car projects exploit carrier-phase differential GNSS (CDGNSS) positioning for vehicle localization. Waymo's GNSS needs are reportedly satisfied by a pair of ublox M8 receivers performing a standard code- and Doppler-based navigation solution. Likewise, a recent teardown of Tesla's Hardware 2.0 Autopilot module revealed nothing more than a single ublox M8L receiver [1], which is not capable of precise CDGNSS-type positioning. But as automated vehicles become increasingly connected, and as they enter markets beyond the sunny confines of Silicon Valley and Chandler, Arizona, they will need some way of determining their globally-referenced position to better than 30 cm in all weather conditions.

Future Vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) connectivity will permit vehicles to relay their positions and velocities to each other with millisecond latency, enabling tight coordinated platooning and efficient intersection management. More ambitiously, broadband V2V and V2I enabled by 5G wireless networks will permit vehicles to share unprocessed or lightly-processed sensor data. *Ad hoc* networks of vehicles and infrastructure

will then function as a single sensing organism. The risk of collisions, especially with pedestrians and cyclists— notoriously unpredictable and much harder to sense reliably than vehicles—will be significantly reduced as vehicles and infrastructure contribute sensor data from multiple vantage points to build a blind-spot-free model of their surroundings.

Such collaborative sensing and traffic coordination requires vehicles to know and share their own position. How accurately? The proposed DSRC basic safety message, a first step in V2V coordination, does not yet define a position accuracy requirement, effectively accepting whatever accuracy a standard GNSS receiver provides [2]. But automated intersection management [3], tight-formation platooning, and unified processing of sensor data—all involving vehicles of different makes who may not share a common map—will be greatly facilitated by globally-referenced positioning with sub-30-cm accuracy.

Poor weather also motivates high-accuracy absolute positioning. Every automated vehicle initiative of which the present authors are aware depends crucially on lidar or cameras for fine-grained positioning within their local environment. But these sensing modalities perform poorly in low-visibility conditions such as a snowy whiteout, dense fog, or heavy rain. Moreover, high-definition 3D maps created with lidar and camera data, maps that have proven crucial to recent progress in reliable vehicle automation, can be rendered dangerously obsolete by a single snowstorm, leaving vehicles who rely on such maps for positioning no option but to fall back on GNSS and radar to navigate a snow-covered roadway in low-visibility conditions. When, as is often the case on rural roads, such snowy surroundings offer but few radar-reflective landmarks, radar too becomes useless. GNSS receivers operate well in all weather conditions, but only a GNSS receiver whose errors remain under 30 cm 95% of the time could avoid drifting onto a snow-covered road's soft shoulder. Code- and Doppler-based GNSS solutions will find it challenging to meet this requirement, even with modernized GNSS offering wideband signals at multiple frequencies.

Carrier-phase-based GNSS positioning can meet the most demanding accuracy requirements envisioned for automated and connected vehicles, but has historically been either too expensive or too fragile, except in open areas with a clear view of the overhead satellites, for widespread adoption. Coupling

a CDGNSS receiver with a tactical grade inertial sensor, as in [4]–[7] enables robust precise positioning even during the extended signal outages common in dense urban areas. Apollo 2.0, Baidu’s open self-driving platform, includes such a coupled positioning system, in particular, the NovAtel SPAN system, among the core sensors of its reference hardware ¹.

But GNSS-inertial systems with tactical-grade inertial measurement units (IMUs) cost tens of thousands of dollars and have proven stubbornly resistant to commoditization. Coupling a GNSS receiver with automotive- or industrial-grade IMUs, is much more economical, and significantly improves performance, as shown in [8]. But such coupling only allows approximately 5 seconds of complete GNSS signal blockage before the IMU no longer offers a useful constraint for so-called integer ambiguity resolution, which underpins the fastest, most accurate, and most robust CDGNSS techniques, namely, single-baseline RTK, network RTK, and PPP-RTK [9]–[11].

University of Texas researchers have recently proposed a novel and inexpensive technique for robustifying RTK positioning: tightly coupling carrier-phase-based GNSS positioning with inertial sensing and vision [12], [13]. Such coupling takes advantage of the remarkable progress in high-resolution, low-cost cameras within the intensely competitive smartphone markets. The current authors are engaged in developing a high-integrity RTK-vision system for precise vehicular positioning in rural and urban environments. Further coupling with radar will make the system robust to low-visibility conditions.

As a step toward this goal, it is of interest to evaluate the performance of stand-alone RTK techniques—those unaided by IMUs or vision—in urban environments. Such a study will reveal why and when aiding is necessary, and how an RTK positioning system might behave if aiding were somehow impaired or unavailable, whether due to sensor faults or, in the case of exclusive visual aiding, poor visibility conditions.

Little prior work has explored stand-alone RTK performance in urban environments, likely because the performance results tend to be somewhat poor. Short-baseline RTK experiments between two vehicles in [14] revealed that multi-frequency (L1-L2) GPS and Glonass RTK yielded poor results in residential and urban environments. Only along a mountain highway with a relatively clear view of the sky was availability greater than 90% and accuracy better than 30 centimeters. RTK positioning in downtown Calgary was disastrous, with less than 60% solution availability and RMS errors exceeding 9 meters.

More recently, Li et al. [8] show that, with the benefit of greater signal availability, stand-alone dual-frequency GPS + BDS + GLONASS RTK can achieve correct integer fixing rates of 76.7% on a 1-hour drive along an urban route in Wuhan, China. But Li et al. do not provide data on the incorrect fixing rate, nor do they assess the accuracy of their ground truth trajectory, so the significance of their correct fixing rate is difficult to assess.

Recent urban RTK testing by Jackson et al. [15] indicates that no low-to-mid-range consumer RTK solution offers greater than 35% fixed solution availability in urban areas, despite a dense reference network and dual-frequency capability. A key failing of existing receivers is their slow recovery after passing under bridges or overpasses. Jackson et al. show that the Piksi Multi and the Eclipse P307 dual-frequency receivers require from 25 (Eclipse) to 40 (Piksi) seconds to recover a fixed position after passing under an overpass.

Professional-grade receivers appear to handle momentary obstructions better, but the primary markets of professional-grade receivers have traditionally been surveying, machine control, and precision agriculture—applications that typically enjoy a relatively unobstructed view of overhead satellites compared to urban vehicular positioning. Consider that in 2018 a professional-grade multi-constellation multi-frequency Trimble receiver requires 4 seconds to produce an RTK fix on a short baseline with a clear sky view (see Fig. 15 in [16]). This may be an acceptably short time-to-fix for traditional RTK applications, but not for urban vehicular positioning.

This paper describes and evaluates an unaided RTK positioning system developed at The University of Texas that has been designed for vehicular operation in both rural and urban environments. Solution availability and accuracy have been markedly improved since publication of preliminary results in [17]. This paper’s primary contributions are (1) a demonstration of the remarkably good performance that can be achieved with a low-cost software-defined stand-alone RTK GNSS platform in an urban environment, and (2) a thorough evaluation of the system’s sensitivity to various impairments.

II. CHALLENGES OF MOBILE PRECISE POSITIONING IN URBAN ENVIRONMENTS

The mobile urban satellite-to-user channel is distinguished by rapid channel evolution. As the vehicle travels along streets closely lined with tall buildings, only glimpses of power are available from signals arriving from directions roughly perpendicular to the roadway. A GNSS receiver designed to provide phase-locked carrier measurements for RTK positioning in such environments must simultaneously (1) prevent frequency unlock during the deep fades caused by building occlusions, and (2) exploit momentary signal availability by immediately acquiring full-cycle phase lock and indicating this to downstream processing.

Tracking in the mobile urban channel is unlike indoor or weak-signal tracking, such as explored in [18], [19], in that the urban fading environment is substantially binary: either the line-of-sight signal is present at a fairly healthy carrier-to-noise ratio C/N_0 , or it is hopelessly attenuated after passing through entire buildings constructed of concrete, steel, and glass. The traditional weak-signal-tracking technique of extending the signal integration time and lowering the tracking loop bandwidths can be useful to slow the rate of frequency unlock during such fading, but not for actually recovering a weak signal from the noise. There is simply no signal to recover.

¹<http://apollo.auto/platform/hardware.html>

Fig. 1 illustrates this fact. The initial disturbance at 950 seconds is due to an overhead traffic light. This is followed in rapid succession by a complete signal blockage due to a tall building on the south side of the east-west street, a brief (four-second) interval of clear satellite availability as the receiver catches a glimpse of the signal between two buildings, and another signal eclipse by a second building.

A GNSS receiver designed for urban tracking will make full use of such between-building glimpses. This requires immediate (within approximately 100 ms) recovery of full-cycle phase lock, which is only possible on suppressed-carrier signals like GPS L1 C/A if the receiver can accurately predict the modulating data symbols. Downstream RTK processing must also be poised to exploit signal glimpses by (1) identifying and rejecting observables from blocked or otherwise compromised signals, and (2) immediately re-evaluating the corresponding integer ambiguities when signals reappear. A multi-stage cycle slip detection and recovery technique, such as proposed in [20], is too slow for urban positioning.

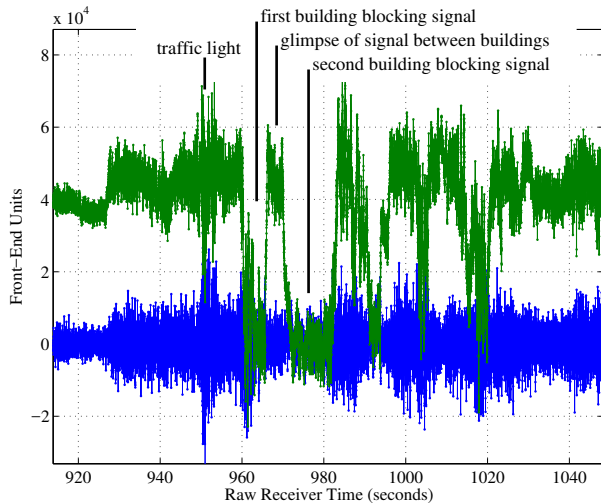


Fig. 1. In-phase (green, top) and quadrature (blue, bottom) 10-ms complex correlation products for a GPS L1 C/A signal at 35 degrees elevation arriving from the south to a vehicle traveling west on an urban roadway. The 20-ms LNAV navigation data bits have been wiped off to allow full carrier cycle recovery. Rapid fading—and rapid recovery—occur as buildings intermittently block the signal.

A related hallmark of the urban mobile channel is the wide and rapid variation of the number of signals available for RTK positioning. Fig. 2 shows the number N_{DD} of double-difference (DD) signals (each one providing a DD pseudorange and a DD carrier phase observable) over a 600-second segment of urban driving. N_{DD} remains constant only when the vehicle is stopped. The implication for RTK processing is that integer ambiguity continuity will often be lost, requiring rapid and continuous re-estimation of ambiguities.

The histogram shown in 3 indicates that, although volatile, N_{DD} remains above 16 for more than half the measurement epochs, which implies that this particular urban environment is not an especially challenging one. Even still, because

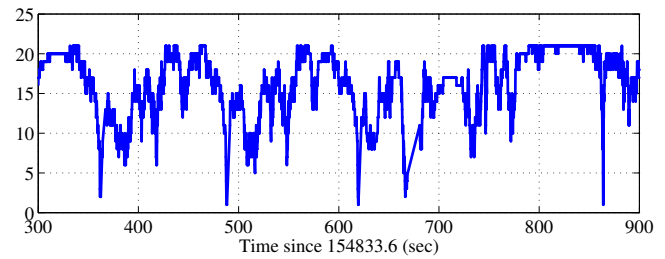


Fig. 2. The number of double difference signals, N_{DD} , available for RTK positioning over a 600-second interval of urban driving.

single-epoch integer ambiguity resolution becomes unlikely for $N_{DD} < 14$ [21], one should expect unaided RTK to struggle.

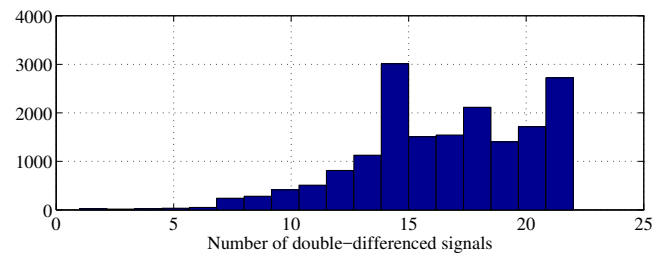


Fig. 3. Histogram of N_{DD} over a 1-hour interval of urban driving.

III. SYSTEM DESCRIPTION

A. Overview

GNSS components of the University of Texas precise positioning system are shown in Fig. 4. The sub-components enclosed in the gray box are the target of the present work's optimization efforts for good performance in urban environments.

Two rover antennas feed analog signals to a radio frequency (RF) front end, which down-mixes and digitizes the signals, producing a stream of intermediate frequency (IF) samples. The RF front end used in the present work produces samples 9.8 Msps for two antennas and two frequencies: a band centered at GPS L1 and one centered at GPS L2. The (single-sided) analog bandwidth of each band is 4 MHz—wide enough to capture over 90% of the power in the GPS L1 C/A, Galileo E1 BOC(1,1), and GPS L2C signals.

Four IF sample streams, one for each antenna and band, are fed to PpRx, an embeddable multi-frequency software-defined GNSS receiver developed primarily at the University of Texas [22]–[24]. PpRx draws ephemeris data and GPS LNAV data bit estimates from the Longhorn Dense Reference Network (LDRN), a set of 8 GNSS reference stations deployed in Austin, TX. Each reference station in the LDRN runs a strict-real-time variant of PpRx and sends its data to a central network server from which any compatible receiver in Austin can draw assistance data and network observables. The master and alternate master reference stations produce observables at 5 Hz. The other stations, which produce data at a slower rate,

are used primarily to estimate ionospheric and tropospheric corrections according to simple linear model, as described in [17], [25].

PpRx feeds code and carrier observables, and other useful signal information, to PpEngine, an RTK engine developed by members of the UT Radionavigation Laboratory. For the results presented in this paper, PpEngine draws observables and ephemeris data from a single LDRN reference station at a time—the traditional RTK topology. The precise solution produced by PpEngine is a fixed (integer-resolved) or float solution depending on the results of an integer aperture test [26].

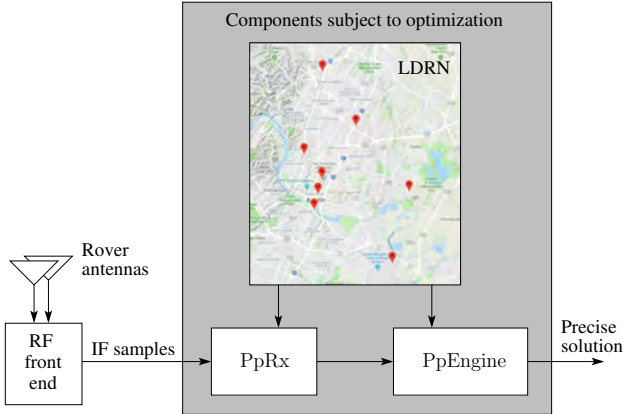


Fig. 4. The University of Texas precise positioning system.

B. Performance Metrics

The performance of precise positioning systems in safety-of-life applications is assessed in terms of integrity, accuracy, and availability [27], [28]. For several emerging applications of practical interest, such as automated and connected vehicles, no regulatory body has set clear positioning performance requirements. An industry consensus appears to be emerging which calls for a 95% accuracy requirement of 30 cm, but it is not clear what the associated integrity risk or continuity requirements should be. It is likely that the U.S. National Highway Traffic Safety Administration, and other regulatory bodies worldwide, will eventually issue positioning performance requirements for connected and automated vehicles.

Assuming that a CDGNSS solution whose integer ambiguities are correctly fixed will be more than sufficiently accurate for all urban RTK applications of practical interest, this paper focuses on two performance metrics: P_S , the probability (or rate in repeated trials) of correctly (successfully) resolving the full integer set, and P_F , the probability that one or more integer ambiguities failed to resolve correctly [27]. A third probability, $P_U = 1 - P_F - P_S$, that of the undecided event, is the probability that a float solution, or no solution at all, is produced, due to an aperture test failure or failure of some other validation test.

An unavoidable tradeoff between P_S and P_F exists such that any increase in P_S comes at the expense of an increase

in P_F (not necessarily of the same amount) [29]. Therefore, an optimization problem can be stated in terms of P_S and P_F as follows: maximize P_S for $P_F \leq \bar{P}_F$, where \bar{P}_F is a fixed tolerable probability of failed fixing. Integer aperture bootstrapping techniques such as [29] and its generalization to partial ambiguity resolution in [27] analytically determine thresholds for the integer aperture test to ensure $P_F \leq \bar{P}_F$. For the optimal integer least squares (ILS) approach adopted in this paper, it is not possible to calculate an analytical aperture threshold, but an approximate one can be obtained via simulation such that $P_F \leq \bar{P}_F$ is satisfied almost surely [30]. A value of $\bar{P}_F = 0.001$ was adopted for the present paper, meaning that a fixing failure rate less than 1 in 1000 epochs was deemed acceptable. However, as will be illustrated later on, multipath, GNSS signal passage through foliage, and other signals impairments common in urban areas cause the empirical P_F to significantly exceed \bar{P}_F when the aperture threshold is chosen according to the Gaussian error assumptions ubiquitous in the integer aperture literature. Thus, a looser *empirical* upper bound \bar{P}_F must be chosen. The optimization problem is then to maximize P_S subject to the empirical P_F respecting the bound $P_F \leq \bar{P}_F$.

C. Design Philosophy

With origins in scintillation-resistant carrier tracking [31], [32], PpRx was designed from the beginning for robust carrier recovery. Likewise, from its inception PpEngine was targeted for the harsh urban environment. Over the past few years, development of PpRx, PpEngine, and the LDRN has proceeded as a parallel evolution, with each subsystem benefiting from improvements in the others.

The overriding design philosophy of this development has been to adapt, rebuild, and reconfigure all three subsystems, separately and in parallel, with a singular goal, namely, to maximize P_S subject to $P_F \leq \bar{P}_F$. This approach benefits greatly from a purely software-based approach to GNSS signal processing (as opposed to processing that exploits dedicated silicon or FPGAs), for two reasons. First, a software-defined approach is almost infinitely flexible: all processing downstream from the RF front end can be reconsidered, rebuilt, and re-evaluated in a rapid iterative process using an efficient and common high-level programming language. Second, software-defined receivers can exploit multiple cores to run *faster* than real time on recorded IF samples [23]. The PpRx-PpEngine pipeline runs at 10x real time on a 6-core Intel Xeon 2.27GHz processor, enabling rapid iteration cycles for quickly probing the optimization landscape.

Rapid iteration is especially important when the GNSS signal environment is difficult to characterize and highly variable, as is the case for urban GNSS. Signal tracking and integer fixing strategies optimized for traditional RTK applications such as surveying and precision agriculture tend to perform poorly when confronted with bridges, tall reflective buildings, and overhead foliage in an urban environment [15]. In fact, it is doubtful whether robust urban RTK can be grounded so strongly on fundamental signal processing

theory as weak signal GNSS tracking [18], [19] and benign-environment mixed real/integer filtering [33]. To be sure, optimal approaches such as those in [33] offer an excellent starting point. But the current authors have been surprised to see some reasonable adaptations of these strategies fail, and counterintuitive strategies succeed, in urban RTK (examples will be offered in subsequent sections). This has engendered a certain epistemological humility and renewed respect for rapid-cycle experimentation.

D. Carrier and code tracking

GNSS carrier and code tracking in an urban environment must be opportunistic, taking advantage of short clear glimpses to overhead satellites as they present themselves. PpRx's code and carrier tracking architecture, illustrated in Fig. 5 has been designed for immediate (within approximately 100 ms) recovery of full-cycle phase lock after a blockage, and, importantly, for prompt lock indication. The following subsections describe the essential elements of PpRx's tracking strategy, calling out parameters whose values significantly affect urban RTK performance.

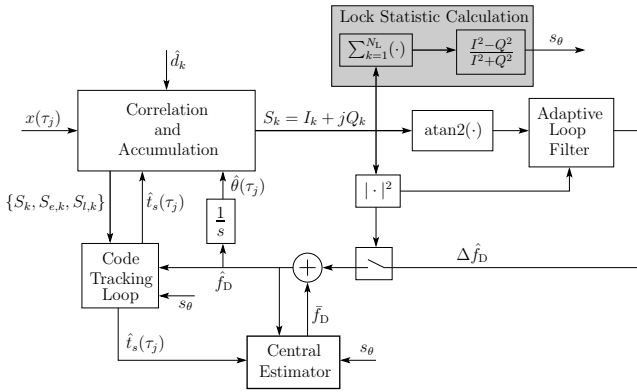


Fig. 5. PpRx's carrier and code tracking architecture.

1) *Correlation and accumulation*: Correlation and accumulation is performed on a sequence of noisy IF samples $x(\tau_j)$, $j = 0, 1, \dots$, where τ_j denotes the time of the j th sample according to the receiver's clock. Within the correlation and accumulation block, a complex local replica signal is formed with code and carrier phase estimates, $\hat{i}_s(\tau_j)$ and $\hat{\theta}(\tau_j)$, provided by the code and carrier phase tracking loops. The outputs of the correlation and accumulation block are prompt, early, and late complex correlation products S_k , $S_{e,k}$, and $S_{l,k}$ of the form $S_k = I_k + jQ_k$, where I_k and Q_k are the in-phase and quadrature accumulations. (The green and blue traces in Fig. 1 correspond to I_k and Q_k , respectively.) The accumulation interval, T_a , is an important configuration parameter for urban RTK.

2) *LNAV data bit prediction*: For the GPS L1 C/A signal, which has no dedicated pilot component, estimates \hat{d}_k of the LNAV data bits, whose structure is detailed in [34], are provided to the local signal replica generator by a prediction engine within PpRx. Extensive testing has shown that

subframes 1-3, which contain ephemeris data, can be nearly-perfectly predicted after the first 18 seconds following a 2-hour GPS time boundary. Almanac data in subframes 4 and 5 are also highly predictable, as they remain constant between approximately daily update events. However, it is worth noting that almanac data can be updated asynchronously—at times not aligned with a 2-hour boundary. For all subframes, reserved bits of the TLM word, and the TLM word's parity bits, must be treated with caution as they can be updated asynchronously with unpredictable content.

The PpRx LNAV prediction engine labels each data bit with one of three certainty levels. The 30-bit HOW word, non-reserved and non-parity bits of the TLM word, and all ephemeris words that have passed parity are marked *certain*. Parity-checked almanac data and reserved and parity bits of the TLM word are marked *almost certain*. Incomplete words, or those that fail parity checking, are marked *uncertain*. The source of each data bit in the local signal replica depends on the prediction engine's corresponding certainty. Prediction engine bits marked *uncertain* are never applied; instead, the observed bit from the received signal is used in the signal replica. This amounts to decision-directed suppressed carrier recovery [35]. Predicted bits marked *almost certain* are used if the signal to noise ratio (SNR) of the prompt complex correlation product S_k falls below a threshold SNR_{pull} ; otherwise the observed bit is used. Bits marked *certain* are used unconditionally in the local replica.

It will be shown in a later section that LNAV data bit prediction is critically important for robust urban RTK.

3) *Lock statistic calculation*: Also key to robust urban RTK is the ability to exclude corrupt or otherwise inaccurate carrier phase measurements. But, due to poor signal availability, an urban RTK engine cannot afford to be overly conservative: it must minimize the number of adequate-quality measurements that get falsely labeled as corrupt. An important indicator for this wheat-from-tares separation process is the lock statistic s_θ . Let I and Q be coherent sums of I_k and Q_k over N_L accumulation intervals. Then s_θ is calculated as [36]

$$s_\theta = \frac{I^2 - Q^2}{I^2 + Q^2}$$

The goal of the carrier tracking loop is to adjust its phase estimate $\hat{\theta}_k$ to shift signal power from Q_k to I_k . Thus, for a loop in lock, $I^2 \gg Q^2$ and s_θ is near unity.

A new lock statistic is produced every N_L accumulations. N_L , must be chosen large enough to suppress thermal noise in I_k and Q_k , but small enough to provide a prompt indicator of phase lock to all dependent processing. PpEngine relies crucially on s_θ to screen out bad measurements. Note from Fig. 5 that s_θ is also fed to the code tracking loop and to PpRx's central state estimator; each one adapts its behavior to rely less on Doppler measurements when s_θ is low.

4) *Carrier tracking*: As illustrated in Fig. 5, PpRx employs a vector signal tracking architecture wherein a central estimator, implemented as a Kalman filter with a nearly-constant-velocity polynomial-type dynamics model, receives

observables from all tracking channels and drives local replica generation for each channel [37]. More particularly, PpRx employs a hybrid strategy in which, for each channel, a local phase tracking loop is closed around a modeled Doppler value \hat{f}_D provided by the central estimator. The local loop's residual Doppler frequency estimate $\Delta\hat{f}_D$ is added to \hat{f}_D to produce the full estimate \hat{f}_D used in replica generation.

A four-quadrant arctangent phase discriminator, $\text{atan2}(Q_k, I_k)$, which is nearly optimal for decision-directed carrier recovery, and optimal for data-free signals, or when data bit wipeoff is error-free, feeds a phase error measurement at every accumulation interval to the carrier tracking loop filter. PpRx's carrier loop filter is designed according to the controlled-root formulation of [38]. The filter adapts its bandwidth B_θ at every accumulation interval according to the value of $|S_k|$. The adaptation schedule has a significant effect on RTK performance.

One might expect that adapting B_θ so to maintain a constant loop SNR as $|S_k|$ varies would yield the best results. This is effectively the adaptation schedule that gets applied in Kalman-filter-based weak signal tracking [18]. However, this reasonable approach was found to yield reduced urban RTK performance. More effective is a three-tiered schedule that reduces B_θ when $|S_k|$ falls below a fairly low threshold γ_1 , and sets B_θ to zero if $|S_k|$ falls below another threshold $\gamma_0 < \gamma_1$. Within this lowest tier, $\Delta\hat{f}_D$ is also driven to zero over a few accumulation intervals, thereby breaking the local feedback loop. In this open-loop mode, the local replica's phase estimate is driven entirely by the model Doppler \hat{f}_D . The lock statistic s_θ continues to be calculated. If s_θ is sufficiently close to unity, the central estimator, the code tracking loop, and the RTK engine continue to treat $\hat{\theta}(\tau_j)$ as a valid measurement. But this is a rare occurrence; s_θ is typically far from unity in open-loop mode.

Open loop tracking has been found to be useful for preventing frequency unlock during intervals when signals are entirely blocked, e.g., by buildings or bridges, and for enabling fast re-acquisition of carrier lock immediately following the blockage.

5) *Code tracking*: PpRx's code tracking loop, which is aided by the Doppler estimate \hat{f}_D , is implemented as a 1st-order loop that toggles between a non-coherent (dot product) discriminator and a coherent discriminator. The coherent discriminator is applied when the channel is phase locked and no recent phase trauma (indicated by s_θ) has been detected; otherwise, the non-coherent discriminator is applied. A flag attached to each code phase measurement $\hat{t}_s(\tau_j)$ indicates whether it was produced under coherent or non-coherent tracking.

As with carrier tracking, the code tracking loop filter's bandwidth, B_{t_s} , is adaptive. But rather than responding to $|S_k|$ as the carrier loop's bandwidth does, B_{t_s} takes on a different value for each of four code tracking modes: (1) pre-phase lock, (2) first post-lock transient, (3) second post-lock transient, and (4) steady-state. These modes are designed to ensure rapid convergence of the code phase estimate $\hat{t}_s(\tau_j)$ after initial signal acquisition, or in the aftermath of phase

unlock.

6) *Discussion on Optimization*: The tracking architecture diagrammed in Fig. 5 and described in the foregoing subsections yields remarkably good urban RTK performance, but it is not claimed to be optimal for this application. It is likely that further vectorization, in which the code tracking loop's code phase estimate is also driven by a central-estimator-provided model value $\bar{t}_s(\tau_j)$, would yield better performance. Or perhaps entirely replacing the local code and carrier loops with a local batch estimator that takes in an interval of $\{S_k, S_{e,k}, S_{l,k}$ values generated using central-estimator-provided $\bar{t}_s(\tau_j)$ and $\hat{f}_D(\tau_j)$ time histories over the interval, and computes from this batch of correlation products residual code phase, carrier phase, and Doppler estimates, would prove to be even more robust to the frequent signal outages and severe multipath in urban tracking.

The larger point is that the usefulness of reasoning *a priori* about optimal architectures for urban RTK is somewhat limited. Likewise, it is hard to say *a priori* what values of the various parameters in the current tracking architecture are optimal in the sense of maximizing P_S for $P_F \leq \bar{P}_F$. Consider the parameters, listed below with default values for the GPS L1 C/A channels, that have been shown to yield good urban RTK tracking for a land vehicle. Default parameter values for the other PpRx channel types are similar. Note too that different parameter values are adopted for the PpRx instances embedded in the LDRN network, as the LDRN stations are static and positioned with a clear view of the sky.

T_a	accumulation interval, 10 ms
N_L	phase lock calculation averaging interval, 2
\bar{s}_θ	phase lock statistic threshold for the code tracking loop and for the central estimator, 0.4
B_θ	default carrier tracking loop bandwidth, 25 Hz
γ_0	SNR of S_k below which carrier tracking operates in open-loop mode, 14.8 dB
γ_1	SNR of S_k above which carrier tracking operates closed-loop mode with bandwidth B_θ , 20 dB
B_θ^1	carrier tracking loop bandwidth that applies when S_k 's SNR lies between γ_0 and γ_1 , 5 Hz
$B_{t_s}^0$	pre-phase-lock code tracking bandwidth, 3 Hz
$B_{t_s}^1$	first transient code tracking bandwidth, 3 Hz
$B_{t_s}^2$	second transient code tracking bandwidth, 0.5 Hz
B_{t_s}	steady-state code tracking bandwidth, 0.5 Hz

Several other tracking parameters, such as the timing for the staged code phase tracking modes and the interval over which $s_\theta \geq \bar{s}_\theta$ must hold before a transition to coherent code phase tracking, have been omitted for brevity. Moreover, the many parameters that govern the behavior of the central estimator, which, due to the vector tracking architecture shown in Fig. 6, influences signal tracking, have not been mentioned at all. Clearly, there are many degrees of freedom over which signal tracking may be optimized for urban RTK.

Are the values listed above optimal for urban RTK? Surely not. Could better values be selected *a priori* based on signal processing theory? Not likely. Is it reasonable to adjust parameter values one at a time to maximize P_S for $P_F \leq \bar{P}_F$? Yes;

such was the approach taken for the current paper. But it is quite possible that a combinatorial approach such as a genetic algorithm would yield better results.

E. Precise positioning

PpRx and the LDRN send carrier and code phase observables, together with signal quality indicators s_θ and C/N_0 , and various other meta-data, to PpEngine for processing. PpEngine is capable of processing observables from both rover antennas simultaneously, exploiting the known distance between these. But for the results presented in this paper, PpEngine was invoked only in its simplest single-antenna mode, producing a precise 3-dimensional baseline between the primary rover antenna and a selected reference station antenna in the LDRN. This simple single-baseline RTK mode was chosen so that the precise positioning system's performance could be evaluated in a familiar configuration and easily compared with other single-baseline RTK evaluations such as [8].

1) *Treatment of real- and integer-valued states:* The current embodiment of PpEngine adopts a straightforward approach to RTK. It first forms code and carrier measurement double differences (DDs) from the rover and reference data, then sends these to a mixed real/integer extended Kalman filter for processing. The filter is implemented as a square-root information filter, as in [33], but limits growth of the number of integer states by either (1) marginalizing at each epoch over float-valued integer ambiguity states modeled as Gaussian-distributed, or (2) conditioning on the estimated integer values. Thus, PpEngine's current approach is to discard all integer states, by marginalization or by conditioning, after each measurement epoch. The marginalization option, which yields the so-called float solution, can be thought of as a special case of the sub-optimal filter in [33] with a window length $i = 1$. The conditioning option, which yields the so-called fixed solution, is invoked only if the integer estimates, found by integer least squares (ILS) [39], are validated by an aperture test.

Conditioning the real-valued states on the lowest-cost integer estimates yields a maximum *a posteriori* 3D baseline estimate. After each measurement update, the real-valued states are propagated to the next measurement epoch, whereupon a new set of integer estimates are formed and conditioning or marginalization occurs yet again. Importantly, if the integer states are validated at the l th measurement epoch, it is the integer-conditioned real-valued states that are propagated to the $(l + 1)$ th measurement epoch. Thus, although all integer states are discarded between measurement updates, correct integer resolution is highly likely at the $(l + 1)$ th epoch if integer ambiguities were correctly resolved at the l th epoch because the real-valued states carry forward a decimeter-accurate position estimate.

Carrying forward integer-conditioned real-valued states is perilous because eventually an erroneous integer estimate passes the aperture test, whereupon the integer-conditioned real-valued states are corrupted by conditioning on the incorrect fix. What is more, the associated square-root information matrices indicate high confidence in the erroneous real-valued

state, raising the chances that the next integer estimates, which are constrained by the prior real-valued states, will also be incorrectly fixed. This vicious cycle, which can persist for several seconds, is eventually broken by an aperture test failure prompted by signal loss, large measurement errors, or the persistent lack of consistency between the incoming observables and the current state.

How often is this cycle entered? For an aperture test configured for a fixed failure rate of \bar{P}_F , it occurs with probability as high as \bar{P}_F even in the ideal case where code and carrier measurement errors are mutually independent, independent in time, and Gaussian-distributed. In fact, all three of these conditions are violated in significant measure for urban RTK, leading to false fixing rates several times larger than \bar{P}_F .

Of course, if the P_F experienced in practice could somehow be bounded below an extremely small value, the risk associated with carrying forward integer-conditioned real-valued states could be made tolerably low. To this end, one might expect that, for good geometry and an adequate number of signals (a strong *a priori* model in the language of [28]), and for some arbitrarily small bound \bar{P}_F , there ought to exist an aperture test that, when configured to respect a sufficiently small \bar{P}_F , would yield an empirical $P_F \leq \bar{P}_F$ while still maintaining a reasonably high P_S . But this does not appear to be the case. Urban multipath and blockage can at times conspire to generate an extremely self-consistent yet erroneous measurement set whose associated integer solution can pass all but the strictest of aperture tests. To prevent such erroneous integer estimates from slipping through and corrupting the real-valued states requires an aperture test so strict that it also excludes almost all correctly-fixed integer estimates, despite a strong underlying model.

Given the above considerations, it would seem folly to pursue a strategy of carrying forward integer-conditioned real-valued states. Recognizing this, the authors are developing a generalization of PpEngine that can manage growth in the number of integer state elements using a variant of the suboptimal approach of [33]. Nonetheless, as will be shown, the current approach is remarkably effective in a moderate urban environment. It also has the virtue of simplicity, making it a good choice for initial development, and of computational efficiency, permitting rapid experimental iteration.

2) *Dynamics Model:* Because this paper's focus is on RTK unaided by any non-GNSS sensors, the mixed real- and integer-valued state estimator within PpEngine was configured to ignore all available inertial measurements and instead rely on a simple nearly-constant-velocity dynamics model for state propagation between measurements [40]. The dynamics model assumes roughly equivalent process noise in the along-track and cross-track directions, but much smaller process noise (by a factor of 10) in the vertical direction, in keeping with a land vehicle operating in a relatively flat urban environment.

3) *Robust measurement update:* Urban multipath and diffraction cause code and carrier observables to exhibit large errors with a much higher probability than even a conservative Gaussian model would predict. Dealing with measurement

error processes such as these, which have thick-tailed distributions, requires robust estimation techniques; that is, techniques with reduced sensitivity to measurement outliers [41].

Outliers are especially problematic for integer fixing in RTK positioning. By action of the decorrelation adjustment preceding ILS, a single bad measurement can contaminate multiple measurements in the decorrelated domain, rendering resolution of the associated integers impossible. Partial ambiguity resolution, as in [42], [43], offers little relief in such cases because contamination caused by outliers is not necessarily limited to an identifiable subset of integers. It is more effective to exclude questionable measurements before the decorrelation adjustment.

PpEngine implements a multi-level exclusion process, depicted in Fig. 6, to mitigate the effects of measurement outliers. At each measurement epoch, measurements are first screened on the basis of three quality indicators: carrier-to-noise ratio C/N_0 , phase lock statistic s_θ , and elevation angle θ_{el} . Signals whose values fall below user-selected thresholds for these quantities are excluded from all DD combinations.

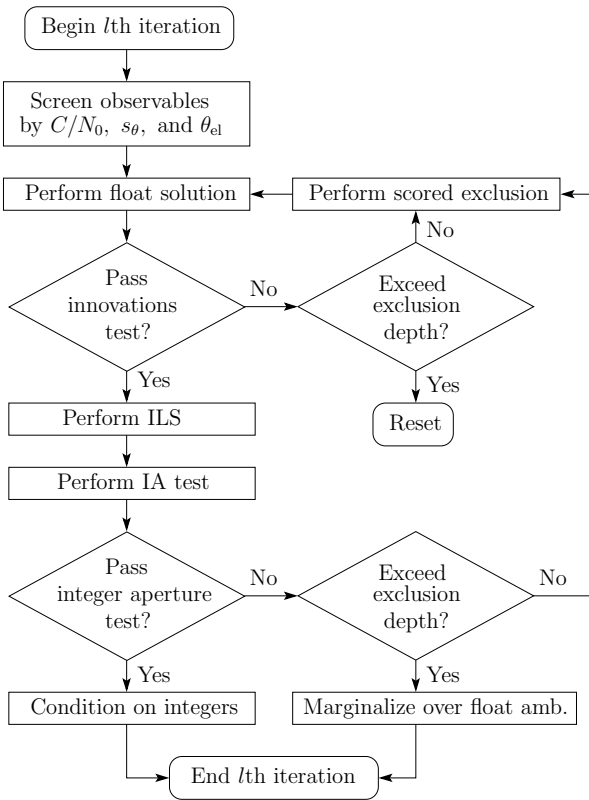


Fig. 6. Flow diagram for the PpEngine exclusion and fixing logic.

A second level of exclusion occurs as part of the float solution. A χ^2 -type test is applied to all DD measurement innovations [40], with exclusion triggered if the normalized innovations squared statistic exceeds a chosen threshold. For the current implementation of PpEngine, this test is only effective at excluding anomalous DD code phase (pseudorange) measurements, since the float states are discarded, and thus

unconstrained, from epoch to epoch. Note that innovations testing benefits strongly from a correctly integer-constrained state because the exclusion threshold can be made tighter. However, with an incorrectly-integer-constrained state, innovations testing may end up excluding the very measurements necessary to correct the state.

If a set of innovations fails the innovations test, DD measurements (both code and carrier for a particular DD combination) are excluded one at a time (with replacement). Exclusion is ordered such that the next DD combination removed is the one with the next-lowest quality score that has not yet been removed. A quality score is formed for each DD combination via a linear combination of scores based on C/N_0 , s_θ , and θ_{el} . If such N -choose-1 elimination fails to create a subset of DD measurements that passes the innovations test, exclusion can proceed to N -choose- m elimination, with $m > 1$. If a user-configurable exclusion depth is exceeded, the estimator state is reset.

The third level of exclusion is based on the integer aperture test following integer estimation via ILS. This is the standard data-driven integer fixing process whereby the integer-fixed solution is selected only on successful validation by some type of aperture test; otherwise, the float solution is accepted [27]. The aperture test is configured for a fixed failure rate (under independent Gaussian errors) of \bar{P}_F . If the integer aperture test fails, N -choose-1 exclusion (with replacement) is attempted, starting with the lowest-scoring DD combinations and working up through higher-scoring combinations. N -choose- m exclusion, with $m > 1$, is currently not attempted at this layer of exclusion because testing a large number of subsets is eventually “doomed to succeed” at passing the aperture test, causing P_F to significantly exceed \bar{P}_F even under benign conditions [43].

If the aperture test is passed before the permissible exclusion depth is exceeded, the solution is conditioned on the integers and the integer states are dropped. Otherwise, the integer state elements are marginalized out as float values. In either case, the state is propagated to the next measurement epoch via the dynamics model and the process repeats.

IV. EXPERIMENTAL SETUP

The precise positioning system was evaluated experimentally using data collected on December 18 and 21, 2017, and January 15, 2018 in a moderate urban environment north of the University of Texas campus in Austin, TX. A total of 3.4 hours of data were collected in 4 sessions over the three days.

A. Rover platform

The rover GNSS receiver is one among several sensors housed in an integrated perception platform called the Sensorium, pictured in Fig. 7. Designed for connected and automated vehicle research, the Sensorium is a self-contained sensor housing that can be mounted atop any standard passenger vehicle. Although hardly visible in Fig. 7, two Antcom 53G1215A-XT dual-frequency patch antennas are flush-mounted in the cross-track direction on the Sensorium’s

upper plate, separated by just over 1 meter. The antennas' signals are routed to a unified RF front end whose output IF samples are processed in real time (to within less than 10 ms latency) by the Sensorium's onboard computer. The samples are also stored to disk for post-processing.

Data from both the primary (driver's side) and secondary (passenger's side) antennas were used to reconstruct the ground truth trajectory, as described below. But only data from the primary antenna were used in the urban RTK performance evaluation. No other Sensorium sensors were involved in the current paper's results.



Fig. 7. The University of Texas Sensorium is a platform for automated and connected vehicle perception research. It includes stereo visible light cameras, an industrial-grade IMU, an automotive radar unit, a dual-antenna, dual-frequency software-defined GNSS receiver, 4G cellular connectivity, and a powerful internal computer.

B. Test route

The test route was primarily composed of two nested loops, as shown in Fig. 8, although other roadways near this area were also explored. The route includes a variety of light-to-moderate urban conditions, from open-sky to overhanging trees to built-up areas. The Dean Keeton corridor, pictured in Figs. 9 and 10, is the most challenging stretch along the test route. It passes below two pedestrian bridges and is flanked on both sides by buildings ranging from 30 to 65 meters tall set back 28 meters from the center of the roadway.

A repeating route is not ideal for evaluating urban RTK performance; a route with greater variety of urban locations would be better. The route in Fig. 8 was chosen to support other research in visual 3D mapping for which repeated sessions through the same corridors was necessary to explore the map's accuracy convergence. Future work in urban RTK will explore routes with greater variety. Meanwhile, the current route remains useful for urban RTK performance evaluation because satellite movement causes multipath and signal blockage conditions to differ significantly from lap to lap and day to day.

V. GROUND TRUTH TRAJECTORY

A primary challenge in urban RTK performance evaluation is obtaining a ground truth trajectory against which to compare the reported trajectory of the system under test. Ideally, the ground truth should be complete and provably accurate to better than 5 cm. Under such conditions, the incorrect fixing rate



Fig. 8. Overview of the test route. The smaller triangular loop was driven 30 times, the larger loop 19 times, both clockwise. The area shown lies on the north side of The University of Texas campus in Austin, TX.



Fig. 9. A 3D overview of the Dean Keeton corridor, spanned by two pedestrian bridges and flanked by buildings on both sides.

P_F can be measured by declaring an incorrect fix whenever the reported trajectory deviates by more than 15 cm from the ground truth. A more straightforward comparison of resolved integers between the reference and test systems, as in [4], is not generally possible in urban areas because the systems may not track the same set of satellites at each epoch throughout the test.

But what system is capable of provably determining the location of a mobile GNSS antenna in an urban area to within 5 cm of its true location? Prior work in urban positioning has relied on forward-backward smoothed trajectories from coupled RTK-inertial systems with a tactical-grade IMU [8], [14]. But the estimated 95% accuracy of the truth trajectory in [14] was only 55 cm in residential areas, and 59 cm in urban canyons, which is far too loose to allow confident measurement of P_F . And the authors of [8] make no attempt to assess the accuracy of their reference system.

A. Examination of DD carrier phase residuals

A defensible claim of better-than-5-cm ground truth accuracy is made in [4], where a tactical-grade IMU was tightly integrated with an RTK system on a short baseline in nearly-perfect open-sky rural conditions. The reference system did experience one brief GNSS outage as the vehicle passed under a roadway sign, but Petovello et al. assert that "an analysis of the measurement residuals and the static position at the end



Fig. 10. A street-level view of the Dean Keeton corridor just before passing below the east pedestrian bridge.

of the run confirmed that the ambiguities had indeed been resolved correctly.”

In urban areas, DD code and carrier measurement residuals likewise carry valuable information about whether the reference system’s integer ambiguities have been correctly resolved. In particular, the absence of a pronounced secular trend in the DD carrier residuals over an interval spanning 30 seconds or more during which the integer estimate remains constant is a strong indicator that the associated ambiguity has been resolved correctly over that interval. Unfortunately, these conditions do not always hold in urban testing. The DD carrier residuals shown in Fig. 11 for a particular GPS signal are small (all less than 50 mm) and appear to be free from secular trends. Moreover, the vast majority of intervals over which the associated estimated ambiguities remain constant span more than 30 seconds. For these intervals, a confident declaration of correct integer resolution can be made. But over the 1-hour interval shown, there are several constant-integer intervals that are too short to allow confident declaration of correctly-resolved integers.

B. Dual-antenna ground truth generation

Given that residuals analysis alone appears insufficient to confidently discern a truth trajectory for urban RTK, an alternative approach is adopted for this paper.

1) *Independent RTK solutions*: Independent RTK solutions are obtained for both the primary and secondary rover antennas. Because these are separated by several GNSS wavelengths on the Sensorium’s top plate, and given the wavelength-scale sensitivity of multipath phase to path length, they experience significantly uncorrelated carrier multipath except when the multipath source is to the front or rear of the vehicle, which is rare (multipath reflections come primarily from buildings to the left or right of a vehicle [44]).

2) *Strict trajectory*: From the independent RTK solutions, a *strict* trajectory is constructed. Let S_P be the set of primary antenna positions that have passed aperture test validation, and let T_P be the set of associated time points. Let S_S and T_S be equivalent sets, respectively, for the secondary antenna. The intersection set $T_\cap = T_P \cap T_S$ contains time points at which both the primary and secondary fixed solutions are available.

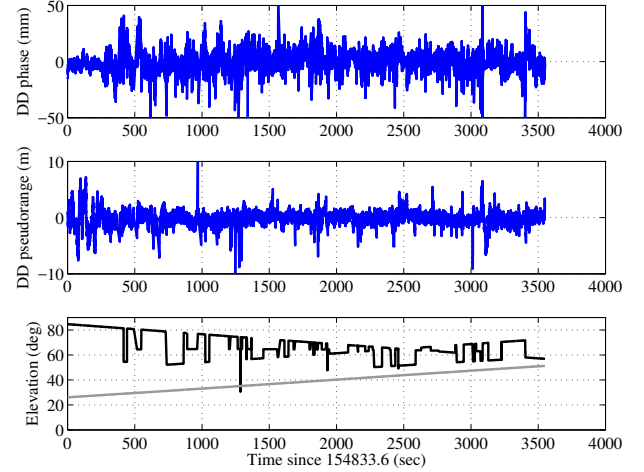


Fig. 11. DD carrier phase (top) and pseudorange (center) residuals for GPS PRN 26’s L1 C/A signal for the 1-hour Jan. 15, 2018 test session. The lower plot shows the satellite’s elevation angle (gray) and the pivot satellite’s elevation angle (black).

Let S_{P_\cap} and S_{S_\cap} be the sets of positions for the primary and secondary antennas associated with T_\cap . These are ordered such that the k th primary position $s_{P,k} \in S_{P_\cap}$ corresponds to the k th secondary position $s_{S,k} \in S_{S_\cap}$. The strict trajectory $S_{S_\cap}^{\text{strict}} \subseteq S_{S_\cap}$ is defined as the set of positions $s_{P,k}$ for which $|b_k - b| < \epsilon$, where

$$b_k = \|s_{P,k} - s_{S,k}\|$$

is the measured baseline length between the two antennas at the k th epoch, b is the known baseline length, and ϵ is an acceptable error threshold, taken to be 15 cm for this paper. Thus, the strict trajectory consists of all fixed primary antenna positions that can be checked against, and are consistent with, a corresponding fixed secondary antenna position. The strict trajectory for an inner loop in the test route is shown as the blue trace in Figs. 12 and 13.

3) *Filling gaps in the strict trajectory*: Approximately 95.5% of epochs in the 3.4 hours of collected data find correspondence in the strict trajectory. The remaining 4.5% of epochs are those for which either the primary or secondary antenna did not produce a fixed solution, or the magnitude of the solution difference disagreed with the known baseline b by more than $\epsilon = 15$ cm. Perhaps unsurprisingly, almost all of these missing or outlier epochs occur near the bridges along the Dean Keeton corridor. Two techniques are applied to attempt fill-in of the strict trajectory gaps.

Transfer via the baseline-constrained solution: A fixed local RTK solution between the primary and secondary rover antennas is obtained at every epoch possible. This local solution’s availability is increased by applying the known baseline constraint b , as in [45]. A missing epoch in the strict trajectory is populated via transfer from the secondary antenna under the following conditions: (i) a fixed secondary antenna position is available, (ii) a fixed baseline-constrained primary-

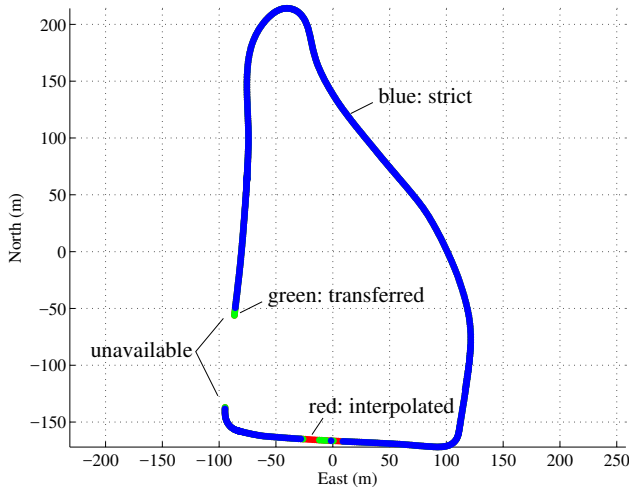


Fig. 12. Horizontal component of a 130-second segment of the ground truth trajectory showing strict (blue), transferred (green), interpolated (red), and unavailable epochs. The red and green points on the lower east-west segment correspond to epochs when the rover vehicle passed under the east Dean Keeton pedestrian bridge.

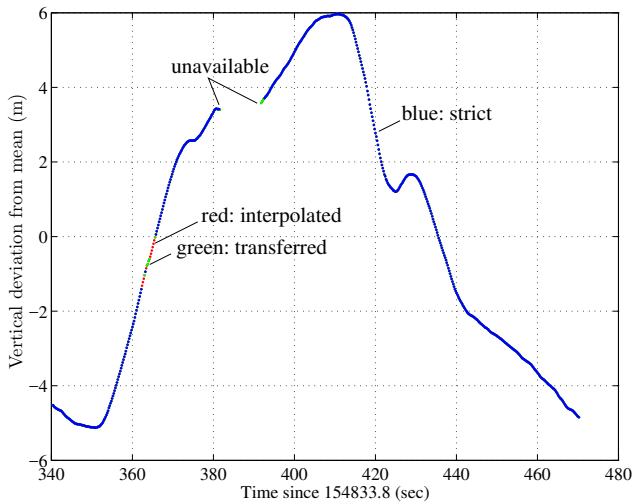


Fig. 13. Vertical component time history associated with the trace in Fig. 12.

to-secondary solution is available, and (iii) when the secondary and local solution vectors are added, the resulting “transferred” primary antenna location is within ϵ of the position predicted by a shape-preserving piecewise cubic interpolation across the strict trajectory gap. The interpolation is carried out via Matlab’s `interp1` function with `METHOD = 'cubic'`. Transferred epochs are shown as green points in Figs. 12 and 13.

Short-hop interpolation: After all permissible secondary-to-primary solutions have been transferred, an attempt is made to bridge the remaining gaps via cubic interpolation (again using the `interp1` function). Only spans shorter than 5 seconds are allowed to be interpolated. Interpolated epochs are shown as red points in Figs. 12 and 13.

C. Ground truth completeness and accuracy

The ground truth trajectory constructed as described above from the strict trajectory, with gaps filled where possible, encompasses approximately 97% of all epochs in the 3.4-hour data set. The missing 3% of epochs almost all coincide with areas near the bridges passing over the Dean Keeton corridor. The west bridge is especially problematic, as the rover vehicle is often stopped for an extended time near or under the bridge while waiting for the stoplight at Dean Keeton and Guadalupe.

Visual inspection of the ground truth trajectory reveals no obvious errors. However, the authors concede that better-than-5-cm accuracy of the available ground truth cannot be guaranteed. Future work will explore additional approaches for completing and verifying ground truth for urban RTK.

VI. BASELINE SYSTEM PERFORMANCE

The baseline urban RTK system is the configuration of the University of Texas precise positioning system that maximizes P_S while respecting $P_F \leq \bar{P}_F$ for some chosen empirical incorrect fixing probability bound \bar{P}_F . This section discusses the baseline system’s performance. The following section will compare the baseline system against several alternatively-configured systems.

A. Baseline configuration

1) *PpRx:* The baseline system’s carrier tracking loops were configured as detailed in Section III-D, with minor variations for the different signal types. The code tracking loops’ default bandwidth was 0.5 Hz. PpRx was configured to track the following signal types: GPS L1 C/A, GPS L2C (combined M + L tracking), Galileo E1 BOC(1,1) (combined B + C tracking), and SBAS (WAAS) on L1. PpRx was configured to output observables at 5 Hz.

2) *PpEngine:* The baseline system’s RTK engine was configured as follows. The master LDRN reference station, located within 1.2 km of all points on the test route, was taken as the reference receiver, producing reference observables at 5 Hz. The master station’s antenna is a Trimble Zephyr II geodetic antenna. A single-baseline RTK solution with a near-zero age of data was performed between the rover’s primary antenna and the reference station at a 5-Hz cadence. The following thresholds were applied in the first-level screening processing within PpEngine: $C/N_0 \geq 37.5$ dB-Hz, $s_\theta \geq 0.55$, and $\theta_{el} \geq 15$ deg. Signals whose values fell below any one of these thresholds were excluded from all DD combinations. Elevation-dependent weighting was applied in the float solution. The threshold above which float innovation statistics failed the normalized innovation squared test was chosen to be 2. Scored N -choose-1 exclusion was applied for both failed float innovations tests and failed aperture tests. A depth of 5 signals was allowed for the N -choose-1 exclusion, after which the estimator was either reset or integers marginalized, according to the flow diagram in Fig. 6. The difference test of [30], which was found to work as well in urban environments, was chosen as the integer aperture test. The test was configured for a fixed failure rate of $\bar{P}_F = 0.001$. The undifferenced

pseudorange and phase measurement error were taken to be $\sigma_\rho = 1$ m and $\sigma_\phi = 3.5$ mm, respectively. The nearly-constant-velocity dynamics model was configured for a 0.2 m/s deviation in horizontal velocity, and a 0.02 m/s deviation in vertical velocity over a 1-second interval.

A rough calibration of the vertical L1-L2 phase center offset was attempted for the primary rover antenna as mounted on the Sensorium.

B. P_S and P_F for the baseline system

Because the ground truth trajectory is incomplete, empirical P_S and P_F can only be determined to within upper and lower bounds. Let P_V denote the empirical probability of successful integer validation at any given epoch, whether the estimated integers are correct or not. In other words, P_V is the availability of a fixed solution. Let P_T denote the fraction of validated epochs that are testable; that is, the fraction having a corresponding populated epoch in the ground truth trajectory. Let P_E denote the fraction of testable epochs that are assumed to have incorrect integer estimates because their 3-dimensional position differs by more than $\epsilon = 15$ cm from the ground truth position.

For the baseline system over the 3.4-hour data set taken over three days, the average values of P_V , P_T , and P_E were

$$P_V = 0.988, P_T = 0.976, P_E = 0.0048$$

With these values one can bound the probability of incorrect fix P_F as

$$P_V P_T P_E = 0.0046 \leq P_F \leq 0.0234 = P_V(1 - P_T)$$

The lower bound optimistically assumes that none of the non-testable validated epochs were in error, whereas the upper bound pessimistically assumes that all non-testable validated epochs were in error. Given that the probability of successful fix $P_S = P_V - P_F$, one can similarly bound P_S :

$$P_V P_T = 0.965 \leq P_S \leq 0.984 = P_V(1 - P_T P_E)$$

Whether the baseline system's performance is impressive depends on the actual value of P_F . Visual inspection seems to indicate that $P_F < 0.01$, which is a factor of 10 larger than \bar{P}_F but may be tolerable for a larger system that combines stand-alone RTK with inertial and electro-optical sensing, as the Sensorium of Fig. 7 is intended to do.

The current system's lower bound for P_S , 0.965, is significantly higher than the $P_S = 0.767$ reported by Li et al. [8] for dual-frequency GPS + BDS + GLONASS stand-alone RTK, but a fair comparison is complicated by the facts that (1) Li et al. do not report P_F , and (2) the Wuhan and Austin urban testing environments are different.

C. Error distribution

Fig. 14 shows the cumulative distribution function (CDF) and a time history of errors for the baseline system's testable aperture-test-validated positions for the Jan. 15, 2018 data set. Positioning performance appears excellent, with over 99.7% of testable epochs having errors smaller than 10 cm. But

one should bear in mind that errors in aperture-test-validated but non-testable solutions are not shown in Fig. 14. Also, among the few errors that do appear, two exceed 1 meter over the 1-hour interval, which would make the baseline system unacceptable as the sole positioning sensor for connected or automated vehicles.

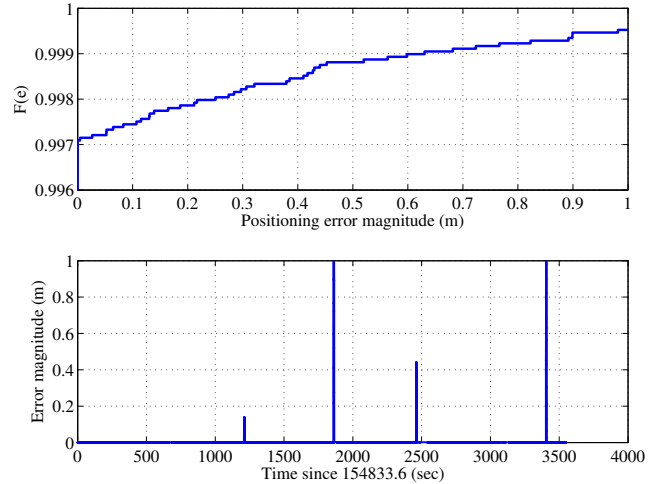


Fig. 14. Cumulative distribution function (top) and time history (bottom) of positioning errors with respect to the ground truth for the baseline system.

D. Residuals Time Histories

It is instructive to examine the DD carrier and code residuals for urban RTK. Fig. 11, discussed previously, shows residuals for GPS PRN 26 during the 1-hour Jan. 15, 2018 session. The RMS values of the time histories correspond to undifferenced phase and pseudorange deviations of $\sigma_\phi = 4.1$ mm and $\sigma_\rho = 0.65$ m. Note that the pseudorange errors are large during the first 250 seconds, over which the vehicle was stationary. This is because PpRx was configured with a large code tracking bandwidth (0.5 Hz) and operates on fairly narrow-band signals (4 MHz) without any dedicated multipath mitigation. When the vehicle begins to move, code multipath gets averaged out non-coherently due to rapid changes in the multipath phase.

Fig. 15 is identical to Fig. 11 but for the L2C signal from GPS PRN 26. For this signal, the undifferenced deviations are slightly larger than for its L1 C/A counterpart: $\sigma_\phi = 4.6$ mm and $\sigma_\rho = 0.78$ m. This appears to be generally the case for L2C signals despite their having the same ranging code bandwidth, and nearly the same power, as L1 C/A signals.

Fig. 16 shows DD residuals for Galileo PRN 4, whose elevation time history is similar to GPS PRN 26 over the same interval. For this signal, $\sigma_\phi = 4.2$ mm and $\sigma_\rho = 0.78$ m. Finally, Fig. 17 shows DD residuals for WAAS PRN 131, for which $\sigma_\phi = 4.4$ mm and $\sigma_\rho = 0.65$ m. The WAAS signal benefits from a high (49-deg. from Austin) and constant elevation angle, but its DD pseudorange residual has a 1.5-meter bias, likely due to asymmetry in the WAAS signal's autocorrelation function. Similarly-constant biases, albeit with different values, are observed for WAAS PRNs 135 and 138.

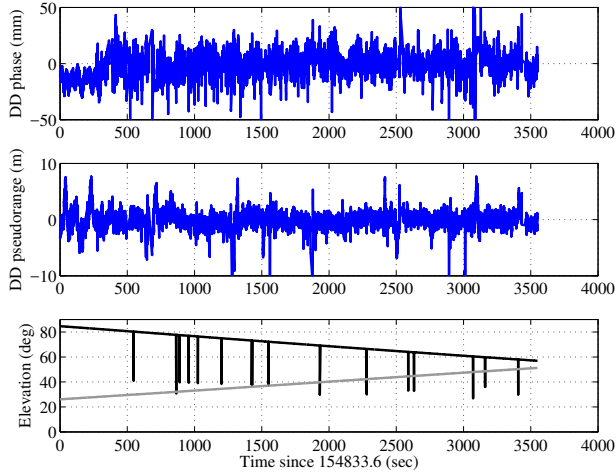


Fig. 15. As Fig. 11 but for PRN 26's L2C signal.

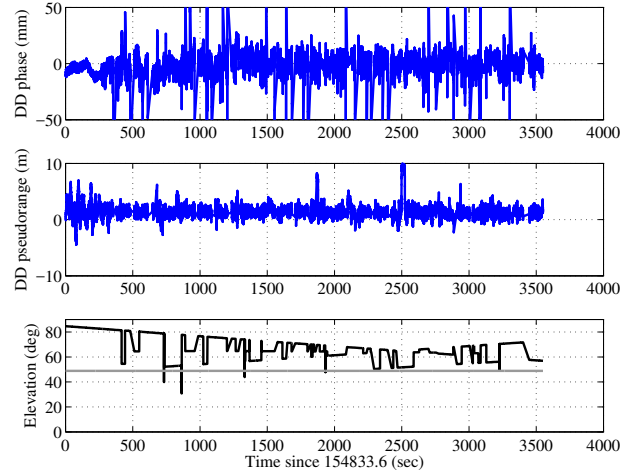


Fig. 17. As Fig. 11 but for WAAS PRN 131 on L1.

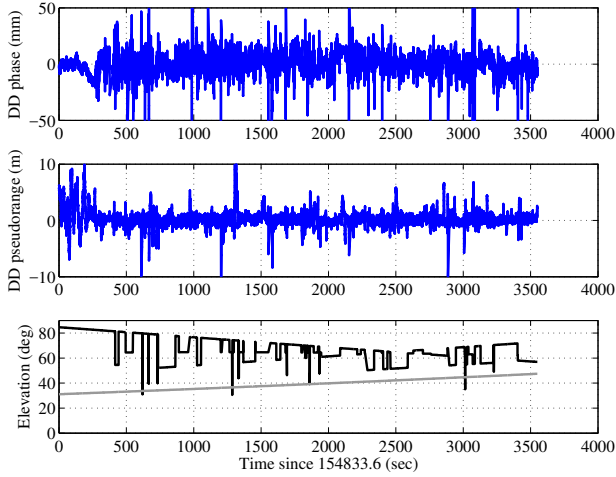


Fig. 16. As Fig. 11 but for Galileo PRN 4's E1 (B+C) signal.

VII. PERFORMANCE DEGRADATION ANALYSIS

This section reports on one of this paper's most significant contributions, a performance degradation analysis in which features of the baseline system are removed or altered one at a time to assess their relative contribution to baseline system performance. The analysis was limited to the 1-hour data set from Jan. 15, 2018. Table I, where P_V , P_T , and P_E are as defined previously, summarizes the results of the analysis. Starting with Scenario 2, subsections below will discuss each scenario in turn.

A. LNAV data bit prediction disabled

Eliminating the baseline's system's LNAV data bit prediction capability, which was described in Section III-D2, has a devastating effect on performance. The availability of validated epochs is scarcely reduced, but P_E rises tremendously, from 0.23% to 36.4%. Fig. 18 shows that large errors persist over

many tens of seconds. It is possible that the PpRx carrier tracking strategy, or the PpEngine integer fixing strategy, could be better tailored for the case where LNAV data bit prediction is disabled, thus reducing P_E , but this would likely cause a significant drop in P_V . Clearly, LNAV data bit prediction is a key capability for urban RTK.

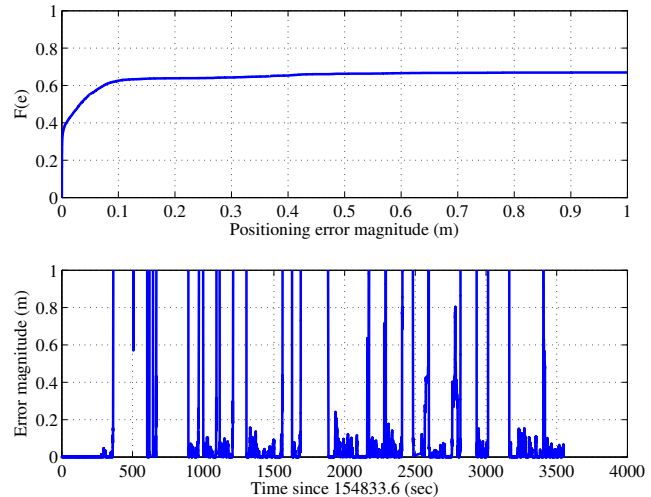


Fig. 18. As Fig. 14 but for Scenario 2.

B. Scalar tracking with adaptive B_θ

Eliminating vector tracking, as described in Section III-D4, in favor of scalar tracking, but retaining carrier tracking loop bandwidth adaptation, has no significant effect on P_V but P_E increases from from 0.23% to 5.1%. Fig. 19 shows that the increase in P_E is primarily due to a single 150-second interval with a persistent large error. Further inspection reveals that the error interval begins at the west bridge. Thus, vector tracking appears helpful, but not critically so.

TABLE I
SUMMARY OF PRECISE POSITIONING RESULTS

Scenario	Description	P_V : Validated Epochs (%)	P_T : Testable (%)	P_E : Errors (%)
1	Baseline system	98.9	96.0	0.23
2	LNAV data bit prediction disabled	97.6	96.2	36.4
3	Scalar tracking with adaptive B_θ	98.7	96.2	5.1
4	Scalar tracking with fixed B_θ	98.5	96.2	4.3
5	GPS L2CL tracking	94.3	96.2	1.8
6	Age of data = 200 ms	98.8	96.0	0.27
7	Age of data = 400 ms	98.4	96.4	0.27
8	Age of data = 600 ms	98.3	96.4	0.35
9	Age of data = 800 ms	98.3	96.4	0.35
10	10 km baseline	97.9	96.4	2.0
11	Sans WAAS	98.7	96.0	4.2
12	Sans GPS L2C (L+M)	97.0	96.4	2.7
13	Sans Galileo E1 (B+C)	95.9	96.5	6.8
14	No scored exclusion	96.6	96.8	4.9
15	35 deg. el. mask	92.3	96.1	6.6
16	35 deg. el. mask, 10 km baseline	81.1	95.9	16.8

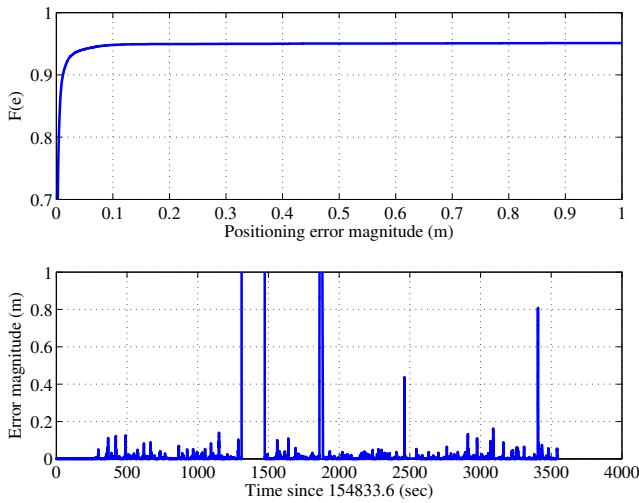


Fig. 19. As Fig. 14 but for Scenario 3.

C. Scalar tracking with fixed B_θ

Eliminating both vector tracking and carrier tracking loop bandwidth adaptation does not appear to cause significant further degradation beyond eliminating vector tracking.

D. GPS L2CL tracking

For GPS L2C tracking, PpRx jointly tracks the pilot (CL) and data-bearing medium-length (CM) codes, wiping off the INAV data symbols modulating the CM code with symbol value estimates based not on prediction, as with LNAV, but merely on observation. The rationale for this strategy is that the CL pilot renders prediction less necessary than for the GPS L1 C/A signal, which does not enjoy a pilot. Eliminating joint L2C L+M tracking in favor of pure L2CL tracking might be thought a more reliable strategy given that no symbol wipeoff mistakes are ever made when tracking only the pilot. But the results of Table I indicate that joint L+M is indeed valuable, as it increases P_V and decreases P_E .

E. Age of data

Scenarios 6-9 explore the effect of increased age of reference data, from the baseline age (near zero latency relative to the rover stream) to 800 ms. Very little reduction occurs in P_V , and little increase in P_E , indicating that the baseline system is not particularly sensitive to increased age of data. However, other experimentation has shown that an age of data beyond 800 ms begins to affect the WAAS carrier DDs. The cause of this degradation is the relatively poor stability of the WAAS clocks, which degrades the accuracy of carrier phase extrapolation to the rover epoch. No such effect appears for GPS or Galileo signals until an age of data beyond 10 seconds.

F. 10-km baseline

The baseline system's distance to the reference receiver, commonly referred to as the reference-rover baseline, is no greater than 1.2 km. For Scenario 10, the LDRN alternate master station, which sits 10 km from the test route, was instead taken as reference. The alternate master station has a Trimble Zephyr II antenna identical to the master station's. Note that a 10-km baseline is still considered to be within the short-baseline regime for standard RTK [46]. Nonetheless, a slight decrease in P_V and increase in P_E is observed. The CDF in the top panel of Fig. 20 also shows that the main drawback of the longer baseline is the increase in large errors, presumably due to incorrect integer fixing, rather than the slight decrease in accuracy of correctly-fixed solutions caused by the longer baseline.

G. Value of additional signals

Scenarios 11-13 explore the degradation in the baseline system that occurs when all signals of a particular type are eliminated from the RTK solution. Over the 1-hour interval, DDs based on 3 WAAS, 3 Galileo, and 4 GPS L2C signals were originally available. Table I indicates that loss of any one of the signal types degrades performance, with Galileo being the most important for the 1-hour data interval studied.

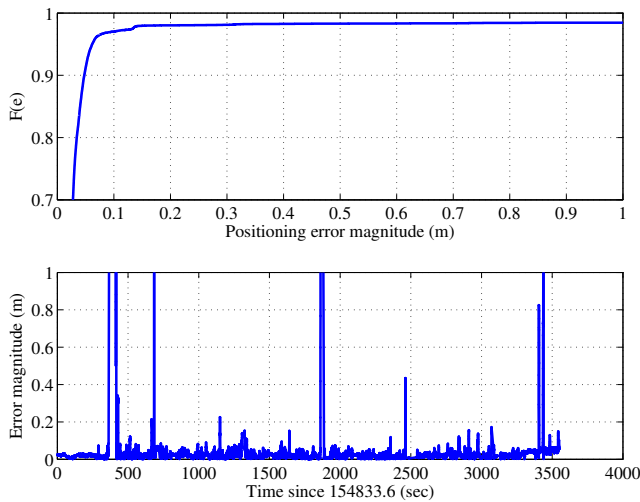


Fig. 20. As Fig. 14 but for Scenario 10.

H. No Scored Exclusion

Scenario 14 removes the scored exclusion strategy described in Section III-E3 by setting the exclusion depth to 0 caused a noticeable reduction in P_V and increase in P_E . Increasing the exclusion depth beyond 5, the baseline system's value, had no discernible effect.

I. 35 degree elevation mask angle

Scenarios 15 and 16 explore the effect on system performance of increasing the elevation mask angle from 15 to 35 degrees, simulating a denser urban environment. For these scenarios, the average number of DDs drops from above 16 (for the baseline system) to less than 11. Scenario 15 takes the usual ~ 1 -km baseline whereas Scenario 16 takes the 10-km baseline. Both scenarios exhibit significant degradation in both P_V and P_E , but the degradation is especially pronounced for the 10-km baseline, with P_V dropping to 81.1% and P_E rising to 16.8%. This is consistent with the argument in [17] that a dense reference network is especially important in urban settings with reduced signal availability.

VIII. CONCLUSIONS

A low-cost urban real-time kinematic (RTK) positioning system developed at The University of Texas for precise vehicular location has been described and evaluated. The system is unaided by inertial or electro-optical sensors. Over 3.4 hours of urban testing, the system achieves a probability of correct integer fixing greater than 96.5% for a probability of incorrect integer fixing surely less than 2.3% and likely less than 1%. Fixed integer solutions are available for 99% of measurement epochs. Of these, 96% are testable against a ground truth trajectory and are shown to be accurate to within 10 cm over 99.7% of the time. A performance sensitivity analysis revealed that navigation data bit prediction on the GPS L1 C/A signals is key to high-performance urban RTK positioning, and that various other features of the positioning system contribute in minor but cumulatively significant ways.

ACKNOWLEDGMENTS

This work has been supported by the National Science Foundation under Grant No. 1454474 (CAREER), by the Data-supported Transportation Operations and Planning Center (DSTOP), a Tier 1 USDOT University Transportation Center, and by the University of Texas Situation-Aware Vehicular Engineering Systems (SAVES) Center (<http://utsaves.org/>), an initiative of the Wireless Networking and Communications Group.

REFERENCES

- [1] kdday, "Inside the NVIDIA PX2 board on my HW2 AP2.0 Model S," May 2017, <https://goo.gl/wNmzh2>.
- [2] 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.
- [3] D. Fajardo, T.-C. Au, S. Waller, P. Stone, and D. Yang, "Automated intersection control: Performance of future innovation versus current traffic signal control," *Transportation Research Record: Journal of the Transportation Research Board*, no. 2259, pp. 223–232, 2011.
- [4] M. Petovello, M. Cannon, and G. Lachapelle, "Benefits of using a tactical-grade IMU for high-accuracy positioning," *Navigation, Journal of the Institute of Navigation*, vol. 51, no. 1, pp. 1–12, 2004.
- [5] B. M. Scherzinger, "Precise robust positioning with inertially aided RTK," *Navigation*, vol. 53, no. 2, pp. 73–83, 2006.
- [6] H. T. Zhang, "Performance comparison on kinematic GPS integrated with different tactical-grade IMUs," Master's thesis, The University of Calgary, Jan. 2006.
- [7] S. Kennedy, J. Hamilton, and H. Martell, "Architecture and system performance of SPAN—NovAtel's GPS/INS solution," in *Position, Location, And Navigation Symposium, 2006 IEEE/ION*. IEEE, 2006, p. 266.
- [8] T. Li, H. Zhang, Z. Gao, Q. Chen, and X. Niu, "High-accuracy positioning in urban environments using single-frequency multi-GNSS RTK/MEMS-IMU integration," *Remote Sensing*, vol. 10, no. 2, p. 205, 2018.
- [9] S. Godha, "Performance evaluation of low cost MEMS-based IMU integrated with GPS for land vehicle navigation application," Master's thesis, The University of Calgary, 2006.
- [10] P. Teunissen and A. Khodabandeh, "Review and principles of PPP-RTK methods," *Journal of Geodesy*, vol. 89, no. 3, pp. 217–240, 2015.
- [11] Y. Cui, X. Meng, Q. Chen, Y. Gao, C. Xu, S. Roberts, and Y. Wang, "Feasibility analysis of low-cost GNSS receivers for achieving required positioning performance in CAV applications," in *2017 Forum on Cooperative Positioning and Service*, May 2017, pp. 355–361.
- [12] D. P. Shepard and T. E. Humphreys, "High-precision globally-referenced position and attitude via a fusion of visual SLAM, carrier-phase-based GPS, and inertial measurements," in *Proceedings of the IEEE/ION PLANS Meeting*, May 2014.
- [13] K. M. Pesyna, Jr., "Advanced techniques for centimeter-accurate GNSS positioning on low-cost mobile platforms," Ph.D. dissertation, The University of Texas at Austin, Dec. 2015.
- [14] R. B. Ong, M. G. Petovello, and G. Lachapelle, "Assessment of GPS/GLONASS RTK under various operational conditions," in *Proc. ION GNSS*, 2009, pp. 3297–3308.
- [15] J. Jackson, B. Davis, and D. Gebre-Egziabher, "An assessment of low-cost RTK GNSS receivers," in *Proceedings of the IEEE/ION PLANS Meeting*, Monterey, CA, 2018.
- [16] S. Riley, H. Landau, V. Gomez, N. Mishukova, W. Lentz, and A. Clare, "Positioning with Android GNSS observables," *GPS World*, vol. 29, no. 1, pp. 18–34, Jan. 2018.
- [17] M. J. Murrian, C. W. Gonzalez, T. E. Humphreys, K. M. P. Jr., D. P. Shepard, and A. J. Kerns, "Low-cost precise positioning for automated vehicles," *GPS World*, vol. 27, no. 9, pp. 32–39, September 2016.
- [18] M. L. Psiaki and H. Jung, "Extended Kalman filter methods for tracking weak GPS signals," in *Proceedings of the ION GPS Meeting*. Portland, Oregon: Institute of Navigation, 2002, pp. 2539–2553.

- [19] H. Niedermeier, B. Eissfeller, J. Winkel, T. Pany, B. Riedl, T. Worz, R. Schweikert, S. Lagrasta, G. Lopez-Risueno, and D. Jimenez-Banos, "Dingpos: High sensitivity GNSS platform for deep indoor scenarios," in *Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on*. IEEE, 2010, pp. 1–10.
- [20] S. Mohiuddin and M. L. Psiaki, "High-altitude satellite relative navigation using carrier-phase differential global positioning system techniques," *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 5, pp. 1628–1639, Sept.–Oct. 2007.
- [21] S. Verhagen, P. J. Teunissen, and D. Odijk, "The future of single-frequency integer ambiguity resolution," in *VII Hotine-Marussi Symposium on Mathematical Geodesy*. Springer, 2012, pp. 33–38.
- [22] T. E. Humphreys, B. M. Ledvina, M. L. Psiaki, and P. M. Kintner, Jr., "GNSS receiver implementation on a DSP: Status, challenges, and prospects," in *Proceedings of the ION GNSS Meeting*. Fort Worth, TX: Institute of Navigation, 2006, pp. 2370–2382.
- [23] T. E. Humphreys, J. Bhatti, T. Pany, B. Ledvina, and B. O'Hanlon, "Exploiting multicore technology in software-defined GNSS receivers," in *Proceedings of the ION GNSS Meeting*. Savannah, GA: Institute of Navigation, 2009, pp. 326–338.
- [24] E. G. Lightsey, T. E. Humphreys, J. A. Bhatti, A. J. Joplin, B. W. O'Hanlon, and S. P. Powell, "Demonstration of a space capable miniature dual frequency GNSS receiver," *Navigation, Journal of the Institute of Navigation*, vol. 61, no. 1, pp. 53–64, 2014.
- [25] M. Murrian, C. Gonzalez, T. E. Humphreys, and T. D. Novlan, "A dense reference network for mass-market centimeter-accurate positioning," in *Proceedings of the IEEE/ION PLANS Meeting*, Savannah, GA, 2016.
- [26] P. Teunissen, "Integer aperture gnss ambiguity resolution," *Artificial Satellites*, vol. 38, no. 3, pp. 79–88, 2003.
- [27] G. N. Green and T. E. Humphreys, "Data-driven generalized integer aperture bootstrapping for high-integrity positioning," *IEEE Transactions on Aerospace and Electronic Systems*, 2017, submitted for review. [Online]. Available: <http://goo.gl/pfq9EU>
- [28] —, "Position domain integrity analysis for generalized integer aperture bootstrapping," *IEEE Transactions on Aerospace and Electronic Systems*, 2017, in preparation. [Online]. Available: <http://goo.gl/qgKu9W>
- [29] P. Teunissen, "Integer aperture bootstrapping: a new GNSS ambiguity estimator with controllable fail-rate," *Journal of Geodesy*, vol. 79, no. 6-7, pp. 389–397, 2005. [Online]. Available: <http://dx.doi.org/10.1007/s00190-005-0481-y>
- [30] L. Wang and S. Verhagen, "A new ambiguity acceptance test threshold determination method with controllable failure rate," *Journal of Geodesy*, vol. 89, no. 4, pp. 361–375, 2015. [Online]. Available: <http://dx.doi.org/10.1007/s00190-014-0780-2>
- [31] T. E. Humphreys, M. L. Psiaki, J. C. Hinks, B. O'Hanlon, and P. M. Kintner, Jr., "Simulating ionosphere-induced scintillation for testing GPS receiver phase tracking loops," *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, no. 4, pp. 707–715, Aug. 2009.
- [32] T. E. Humphreys, M. L. Psiaki, B. M. Ledvina, A. P. Cerruti, and P. M. Kintner, Jr., "A data-driven testbed for evaluating GPS carrier tracking loops in ionospheric scintillation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 46, no. 4, pp. 1609–1623, Oct. 2010.
- [33] M. Psiaki, "Kalman filtering and smoothing to estimate real-valued states and integer constants," *Journal of Guidance, Control, and Dynamics*, vol. 33, no. 5, pp. 1404–1417, Sept.–Oct. 2010.
- [34] GPS Directorate, "Systems engineering and integration Interface Specification IS-GPS-200G," 2012, <http://www.gps.gov/technical/icwg/>.
- [35] T. E. Humphreys, M. L. Psiaki, and P. M. Kintner, Jr., "Modeling the effects of ionospheric scintillation on GPS carrier phase tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 46, no. 4, pp. 1624–1637, Oct. 2010.
- [36] A. J. Van Dierendonck, *Global Positioning System: Theory and Applications*. Washington, D.C.: American Institute of Aeronautics and Astronautics, 1996, ch. 8: GPS Receivers, pp. 329–407.
- [37] M. Lashley and D. Bevely, "What are vector tracking loops, and what are their benefits and drawbacks?" *GNSS Solutions Column, Inside GNSS*, vol. 4, no. 3, pp. 16–21, 2009.
- [38] S. A. Stephens and J. B. Thomas, "Controlled-root formulation for digital phase-locked loops," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 31, no. 1, pp. 78–95, Jan. 1995.
- [39] P. J. Teunissen, "The least-squares ambiguity decorrelation adjustment: a method for fast GPS integer ambiguity estimation," *Journal of Geodesy*, vol. 70, no. 1-2, pp. 65–82, 1995.
- [40] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. New York: John Wiley and Sons, 2001.
- [41] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge Univ Press, 2000, vol. 2.
- [42] A. Brack, "On reliable data-driven partial GNSS ambiguity resolution," *GPS Solutions*, vol. 19, no. 3, pp. 411–422, 2015. [Online]. Available: <http://dx.doi.org/10.1007/s10291-014-0401-9>
- [43] A. Parkins, "Increasing gnss rtk availability with a new single-epoch batch partial ambiguity resolution algorithm," *GPS solutions*, vol. 15, no. 4, pp. 391–402, 2011.
- [44] E. Steingass and A. L. German, "Measuring the navigation multipath channel—a statistical analysis," in *Proceedings of the ION GNSS Meeting*, 2004.
- [45] K. Chiang, M. Psiaki, S. Powell, R. Miceli, and B. O'Hanlon, "GPS-based attitude determination for a spinning rocket," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 4, pp. 2654–2663, 2014.
- [46] R. Odolinski, P. Teunissen, and D. Odijk, "Combined GPS+BDS for short to long baseline RTK positioning," *Measurement Science and Technology*, vol. 26, no. 4, p. 045801, 2015.