

EXHIBIT 1017

Sensitivity Analysis of a Tightly-Coupled GPS/INS System for Autonomous Navigation

ISAAC MILLER, Member, IEEE
MARK CAMPBELL, Member, IEEE
Cornell University

The work presented here empirically analyzes the design of the tightly-coupled position, velocity, and attitude estimator used as a feedback signal for autonomous navigation in a large scale robot driving in urban settings. The estimator fuses GNSS/INS signals in an extended square root information filter (ESRIF), a numerically-robust implementation of an extended Kalman filter (EKF), and was used as the basis for Cornell University's 2007 DARPA Urban Challenge robot, "Skynet." A statistical sensitivity analysis is conducted on Skynet's estimator by examining the changes in its behavior as critical design elements are removed. The effects of five design elements are considered: map aiding via computer vision algorithms, inclusion of differential corrections, filter integrity monitoring, Wide Area Augmentation System (WAAS) augmentation, and inclusion of carrier phases; the effects of extensive signal blackouts are also considered. Metrics of comparison include the statistical differences between the full solution and variant; the Kullback-Leibler divergence; and the average and standard deviation of the position errors, attitude errors, and filter update discontinuities.

Manuscript received January 8, 2010; revised October 10, 2010; released for publication March 24, 2011.

IEEE Log No. T-AES/48/2/943807.

Refereeing of this contribution was handled by M. Braasch.

This material is based upon work supported under a National Science Foundation Graduate Research Fellowship. This work is also supported by the DARPA Urban Challenge program, Contract HR0011-06-C-0147.

Authors' address: Sibley School of Mechanical and Aerospace Engineering, Cornell University, 208 Upson Hall, Ithaca, NY, 14850, E-mail: (mc288@cornell.edu).

0018-9251/12/\$26.00 © 2012 IEEE

I. INTRODUCTION

Global navigation satellite system (GNSS) signals such as GPS and dead reckoning inertial navigation systems (INS) are well known to be an ideal pair for sensor fusion and temporal filtering. The resulting fused position, velocity, and attitude (pose) solution has seen a staggering number of academic, commercial, and defense applications, including navigation, surveying, and guidance [1–6]. Such widespread and varied use of the GPS/INS sensing pair has resulted in a multitude of approaches for fusing the available information, depending on the type and characteristics required of the outputs. Three common integration strategies, loose coupling, tight coupling, and ultra tight coupling, fuse GPS with INS at various levels of GPS processing [3]. New emerging strategies seek to augment this basic level of coupling with application-based motion constraints and measurements of nearby landmarks, either from maps known a priori or generated in real time [7–11].

One area where GPS/INS integration is becoming increasingly important is robotics, where fused GPS/INS pose solutions are used as a feedback signal for path tracking and decision making. For example, Sukkarieh, et al. develop an integrated GPS/inertial measurement unit (IMU) system using Kalman filtering and fault detection intended for land vehicle applications [1]. Similarly, George and Sukkarieh develop an extended Kalman filter (EKF) to fuse GPS and IMU measurements, which are intended to improve performance in control of unmanned air vehicles [9]. More recently, Miller, et al. have developed a loosely-coupled GPS/INS integration using an extended square root information filter (ESRIF) to provide real-time position feedback in a full-size autonomous ground vehicle [12]. In a similar application, Thrun, et al. have also developed a loosely-coupled but problem-constrained sigma point filter (SPF) to guide a full-size autonomous ground vehicle over 130 mi of desert terrain without human intervention [13].

Development of these integrated GPS/INS pose solutions for robotics applications has uncovered an important yet subtle realization: requirements of pose solutions used in robotics applications differ substantially from requirements of pose solutions used in more traditional applications such as surveying and mapping. In more traditional applications, accuracy is often paramount, and position errors and magnitudes are important performance criteria. In robotics applications, pose solutions typically must strike a balance between several competing objectives: accuracy, integrity and robustness, and precision [1]. Accuracy, evaluated in terms of position errors, is necessary to ensure the robot follows a desired path. Integrity and robustness, defined in terms of resistance to large discontinuities caused by rapid changes in

information, are necessary to ensure the robot remains stable while tracking a desired path. Finally precision, defined in terms of an ability to produce similar pose solutions over time, is necessary to ensure the robot can travel the same path repeatedly. In no place was the balance between these objectives more apparent than the 2005 DARPA Grand Challenge, a United States Government-sponsored 130 mi desert race in which the only competitors were robots. There, failure to detect an increasingly-distorted GPS signal in Cornell University's pose estimator resulted in failure due to unstable driving after only 9 mi of travel [12].

Two years later, the United States Government sponsored a new robotic race, the 2007 DARPA Urban Challenge. This race was to take place in an urban setting, with each full size robot to complete a 60 mi set of mock supply missions without human intervention, in a city filled with traffic. Cornell's 2005 pose estimator was rewritten for the Urban Challenge, under full awareness of the lessons learned in the Grand Challenge and the new difficulties present in the urban environment. Ultimately the revised pose estimator proved successful, as Cornell's robot was one of only six to complete the entire 60 mi Urban Challenge [14].

This paper analyzes the design decisions made and the importance of each filter component in the revised pose estimator as a feedback signal for real-time field-proven autonomous navigation. The contribution of this work is a comprehensive, rigorous, and quantitative statistical sensitivity analysis of those design decisions and filter components. This sensitivity analysis is performed by evaluating statistical differences in pose estimates that arise as critical aspects of the pose estimator design are removed. Two types of sensitivity analysis are performed. The first is an ordered removal of design elements one by one, based roughly on their implementation and monetary cost: first map aiding, then differential corrections, followed by integrity monitoring, Wide Area Augmentation System (WAAS) corrections, and carrier phases. The second is an exhaustive study of all cases, in order to understand better which of the design elements contributed the most to the final solution. The effects of extensive signal blackouts are also considered. Metrics of comparison include the statistical differences between the full solution and filter variant; the Kullback-Leibler (K-L) divergence; and the average and standard deviation of the position errors, attitude errors, and filter update discontinuities. Note that this sensitivity analysis is performed strictly for autonomous applications in the spirit of Sukkarieh, et al., though evaluation is extended to quantitative statistical assessment of all critical design choices instead of qualitative assessment of a single fault detection system [1]. The statistical experiments are performed with experimental data collected by Cornell



Fig. 1. Cornell University's Skynet, an autonomous Chevrolet Tahoe that completed the 2007 DARPA Urban Challenge. Skynet uses a tightly-coupled GPS/INS pose estimator implemented as an ESRIF.

University's robot driving autonomously at the 2007 DARPA Urban Challenge. This data gives the study a unique ground truth, and deviations from this ground truth are evaluated in terms of potential impacts to autonomous driving. This work is an extension of [15] to include the exhaustive study and its associated conclusions.

The remainder of this paper is organized as follows. Section II introduces Cornell University's DARPA Urban Challenge robot, called "Skynet," with a brief discussion of its system architecture and hardware. Section III gives an overview of Cornell's pose estimator algorithm, a tightly-coupled ESRIF, and the important facets of its design. Section IV examines the pose estimator's performance through the ordered sensitivity analysis, whereby variants of the pose estimator are run on logged Urban Challenge sensor data to discern the effects of each critical design decision. Section V examines several cases of extended GPS blackouts. And Section VI presents the exhaustive sensitivity analysis. Section VII concludes with a statistical summary of the pose estimator's sensitivity to its inputs, with emphasis placed on requirements for successful autonomous navigation.

II. CORNELL UNIVERSITY'S "SKYNET"

Cornell University's Skynet, shown in Fig. 1, is a full size 2007 Chevrolet Tahoe built to compete as Team Cornell's entry in the 2007 DARPA Urban Challenge. Skynet is fully autonomous: completely actuated, filled with computers and a network architecture, equipped with sensors and probabilistic perception algorithms, and outfitted with navigation and traffic planning algorithms. It is capable of navigating and interacting with a populated urban environment, obeying California state driving laws without human intervention [14].

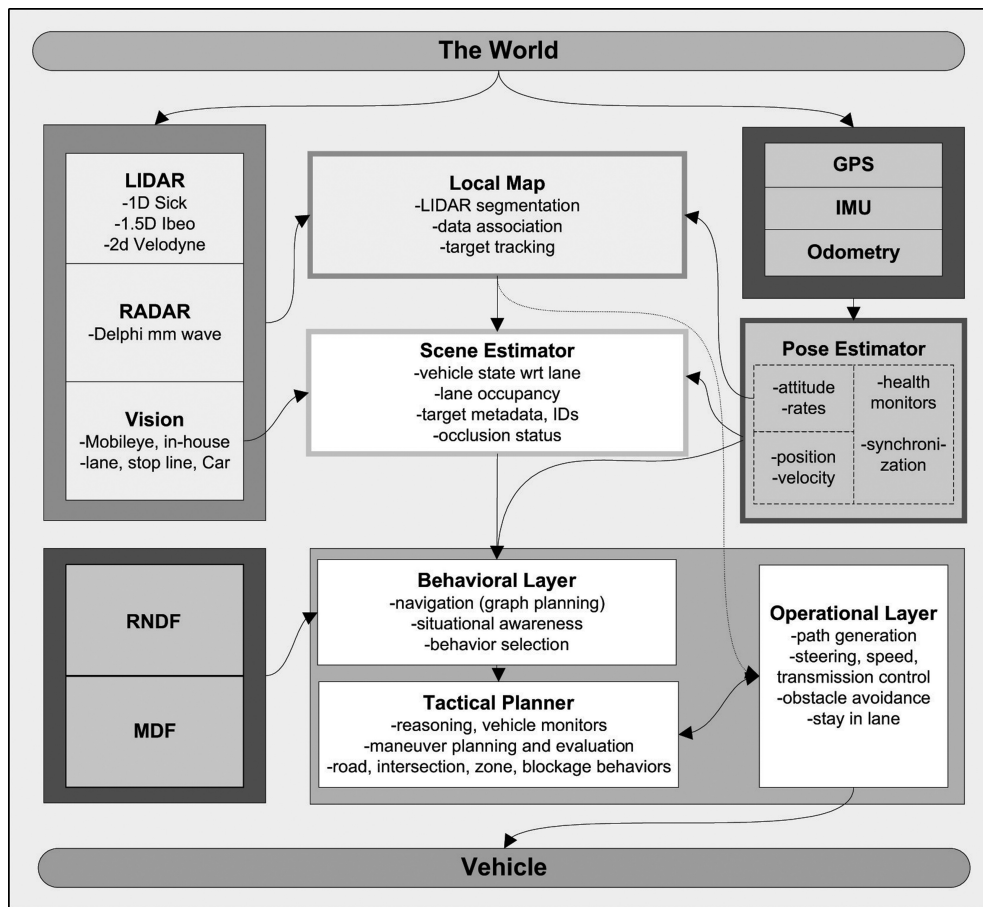


Fig. 2. Skynet's system architecture. Skynet includes pose estimator for position, velocity, and attitude estimation, perception algorithms for detecting and tracking moving and static obstacles, and planning algorithms for choosing driving path on road network (RNDF) to complete its specified mission (MDF). All Skynet's critical subsystems depend on output from pose estimator.

Between October 25, 2007 and October 31, 2007, Skynet participated in the Urban Challenge National Qualifying Event (NQE), a set of short 25-min autonomous urban missions designed by DARPA to evaluate candidates for selection into the Urban Challenge Event (UCE) itself. Completing these events safely and successfully, Skynet was selected as one of eleven robots allowed to compete in the 2007 DARPA Urban Challenge. On November 3, 2007, after approximately 6 hr and 60 mi of autonomous urban driving, Skynet successfully completed the DARPA Urban Challenge. It was one of only six robots to finish.

A. Skynet's System Architecture

Skynet interacts with the world according to the action, sensing, and planning subsystems shown as general system blocks in Fig. 2. The vehicle block, denoted "vehicle" in Fig. 2, is the lowest level of these subsystems. It comprises both the hardware and software aspects of Skynet's steering, brake, throttle, and transmission actuation systems. These actuation systems, consisting of aerospace servomotors and

low level software tracking loops, allow Skynet's steering wheel, brake pedal, electronic throttle, and transmission cable to achieve commands issued by the onboard computers.

The localization algorithm, denoted "pose estimator" in Fig. 2, lies above the vehicle block. It fuses information from external satellite signals with onboard inertial navigation and vehicle motion sensors to produce an attitude and position solution. This solution is supplied to Skynet's intelligent planner, where it is used both as a feedback signal for basic path tracking, and as an absolute localization signal for dictating higher level vehicle behaviors in traffic. The localization algorithm also produces a relative attitude and position solution, which relates Skynet-fixed coordinate frames from previous times to the vehicle's current attitude and position.

The perception algorithms, denoted "local map" and "scene estimator" in Fig. 2, operate in parallel with the localization algorithm. The perception algorithms process information from Skynet's onboard sensors: laser rangefinders, millimeter-wave radar, and optical cameras to interpret Skynet's surroundings

for the intelligent planner. The local map fuses this incoming sensor information with relative attitude and position solutions from the localization algorithm to detect and track obstacles in a Skynet-fixed coordinate frame. The scene estimator fuses these tracked targets with the absolute attitude and position solution from the localization algorithm to assign absolute positions and lane occupancy probabilities to each tracked target. It also fuses the absolute solution with its own camera-based lane estimates to improve the final attitude and position solution that is passed to the planner.

The intelligent planning algorithm, denoted “behavioral layer,” “tactical layer,” and “operational layer” in Fig. 2, considers all low level feedback, localization, and obstacle information provided by the other subsystems to plan and follow routes to complete its assigned mission. This task consists of three levels of planning. First, the behavioral layer uses Skynet’s current position and a graph search algorithm to plan an initial path through the navigable streets defined by DARPA’s Route Network Definition File (RNDP) and Mission Data File (MDF). Second, the tactical layer guides Skynet along this high level route by selecting an efficient and law-abiding path over a short segment of the route. Finally, the path planned by the tactical layer is handed to the operational layer, which uses a constrained nonlinear optimization package to smooth the first 15 to 30 m of the path into one that is physically achievable. The path generated over this small planning horizon is then discretized into speed and curvature commands to be achieved by the low level actuation controllers. This small path is tracked over time in a Skynet-fixed coordinate frame using the relative attitude and position solution from the localization algorithm. The path is discarded at the next planning cycle, when a new high level planned path takes its place.

B. Skynet’s Localization Hardware

Skynet’s position, velocity, and attitude (pose) solution, generated by the pose estimator system block in Fig. 2, is used in all higher level reasoning and perception components to facilitate safe autonomous urban driving. As the only source of absolute position information on Skynet, the pose estimator’s accuracy, precision, robustness, and stability all directly influence how Skynet interprets and responds to its surroundings. Cornell University’s pose estimator has therefore been designed with consideration for these four abstract qualities while satisfying the more concrete requirements of the Urban Challenge: staying in designated lanes, coming to a stop within 1 m of each stop line, parking in waypoint-defined parking spaces, and obeying other location-dependent traffic laws. The pose estimator has also been designed with regard to the difficulties of the urban GPS signal environment, including heavy multipath and frequent signal obstruction.

Skynet’s pose estimator satisfies these basic design requirements by fusing information from reliable and high quality localization sensing hardware. Four sensors are fused: a Litton LN-200 IMU, Skynet’s antilock brake (ABS) wheel encoders, a Septentrio PolaRx2e@ three-antenna GPS receiver, and a Trimble Ag252 single-antenna high precision (HP) GPS receiver. The LN-200 is a combined three-axis fiber optic rate gyroscope and a three-axis silicon accelerometer. This tactical grade IMU is mounted on the floor of Skynet, along its centerline, just above the rear axle. The LN-200 integrates its rate gyros and accelerometers to produce estimates of differential vehicle rotation and motion at 400 Hz, and has superb 1 deg/hr gyro bias repeatability and 300 μ g accelerometer bias repeatability [16]. Intuitively, the IMU is used to integrate Skynet’s pose solution forward in time with a dead reckoning numerical integration scheme. The IMU is augmented with Skynet’s ABS wheel encoders, 64 count/rev encoders mounted to each of Skynet’s tires. Information from these encoders is obtained from Skynet’s stock GM CAN network at 30 Hz data transmission rates. The information is used to calculate an average speed at the center of Skynet’s rear axle, which is subsequently used to slow the rate of error growth in IMU integrations. Differential wheel speeds could also be used to measure yaw rate, though Skynet does not use these measurements due to their considerable noise.

The two GPS receivers, the Septentrio PolaRx2e@ and the Trimble Ag252, are both used as absolute position signals to correct errors that grow over time in dead reckoning. The Septentrio PolaRx2e@, a three-antenna single-clock 48-channel GPS receiver provides raw pseudo-range, Doppler shift, and carrier phase measurements of all visible GPS satellites on all its antennas at 5 Hz synchronized measurement intervals [17]. Its antennas are mounted in an “L” pattern on Skynet’s roof, with 1.118 m separating the front and rear driver’s side antennas, and 1.092 m separating the rear driver’s and passenger’s side antennas. The PolaRx2e@ decodes the WAAS signal at the same 5 Hz synchronized measurement rate to permit the use of more accurate signal models. The PolaRx2e@ also reports raw pseudo-ranges to WAAS satellites, though those are not used in Skynet’s pose estimator. In contrast, the Trimble Ag252 is a single-antenna GPS receiver, mounted on the centerline of Skynet’s roof, just above the rear axle. In Skynet’s pose estimator, the Ag252 is used solely to decode HP and virtual base station (VBS) OmniSTAR differential corrections. These corrections are decoded at a rate of 5 Hz, synchronized on the GPS second according to the Ag252’s clock [18]. With an advertised accuracy of ± 10 cm, these differential correction signals are the primary source

of satellite-based submeter positioning information in Skynet's pose estimator.

The sensors chosen for the pose estimator complement each others' weaknesses to improve overall robustness and reliability in Skynet's pose estimator. The LN-200 provides accurate measurements of differential vehicle motion at the high data rates necessary to produce a smooth pose solution, but the rate measurements must be numerically integrated. The resulting IMU dead reckoning scheme is therefore subject to integration errors that increase over time, with no absolute position or orientation information to correct the integration errors [5]. Skynet's ABS wheel encoders help slow the growth rate of these accumulating integration errors by providing speed measurements, which are used to adjust estimates of biases in the LN-200 accelerometer. Although these measurements improve the quality of the pose solution during dead reckoning, they are subject to measurement errors and the effects of wheel slip. The PolaraX2e@ provides data at a much slower rate than the IMU, but its GPS- and WAAS-based absolute attitude and position information can be used to prevent IMU integration errors from growing without bound. These three sensing modalities cannot consistently achieve submeter accuracy, even when fused, but they can when updated with infrequent measurements of the differential correction signal made by the Ag252. The differential correction signal tends to be brittle, as it is often biased and difficult to track for extended periods of time. However, statistical hypothesis tests can be used to determine which differential correction measurements are most accurate. These tests, which depend on historical data accumulated from all four sensing modalities, make the differential correction signal beneficial even in challenging urban signal environments.

III. CORNELL'S ATTITUDE AND POSITION ESTIMATION ALGORITHM

Skynet's four localization sensing modalities, described in Section IIB, are statistically fused via an ESRIF into Skynet's pose estimate. The information form of the filter is useful for GPS/INS integration because many disparate sensors can be easily fused, and initialization of the filter (for instance, with infinite covariance) is easier, such as for carrier phase ambiguity states. The square root formulation adds numerical robustness to the filter, which is important for cases where there is large scaling discrepancies between state variables, such as between sensor bias states and carrier phase ambiguity states.

This section reviews Cornell's pose estimator. First, Bierman's numerically robust Square Root Information Filter (SRIF) is reviewed in relation to the traditional Kalman Filter (KF) [19]. Next,

the general statistical formulation of the ESRIF is reviewed, as described by Psiaki, et al. in relation to the EKF [2]. Finally, Skynet's pose estimator is developed as a specific instance of an ESRIF designed for autonomous navigation.

A. A Brief Review of the Square Root Information Filter

The SRIF is a numerically stable implementation of the KF [19]. Like the KF, the SRIF is a recursive filter designed to estimate the a posteriori density of a state vector $x(k)$ at time index k conditioned on an observed time series of sensor measurements $Z(k) = [z(1), z(2), \dots, z(k)]$. Also like the KF, the SRIF operates under the assumption that $x(k)$ evolves in time according to the linear dynamics model:

$$x(k+1) = F(k)x(k) + G(k)u(k) + \Gamma(k)v(k) \quad (1)$$

where $F(k)$ is the state transition matrix, $G(k)$ is the control input matrix for known control input $u(k)$, and $\Gamma(k)$ is the noise input matrix for the unknown process noise vector $v(k)$. Similarly, the KF and SRIF assume that sensor measurements $z(k)$ are linear functions of the state:

$$z(k) = H(k)x(k) + w(k) \quad (2)$$

where $H(k)$ is the measurement matrix, and $w(k)$ is the unknown measurement noise vector. Most importantly, the KF and SRIF assume that the initial a priori state density $p(x(0))$, the process noise density $p(v(k))$, and the measurement noise density $p(w(k))$ are all Gaussian:

$$p(x(0)) \sim \mathcal{N}(\hat{x}(0), P(0)) \quad (3)$$

$$p(v(k)) \sim \mathcal{N}(0, Q(k)) \quad (4)$$

$$p(w(k)) \sim \mathcal{N}(0, R(k)) \quad (5)$$

where $\hat{x}(0)$ is the initial a priori state estimate, $P(0)$ is the initial a priori state covariance matrix, and $Q(k)$ and $R(k)$ are the possibly time-varying covariance matrices of the process and measurement noise vectors, respectively. Both the process noise and measurement noise are assumed to be white noise random processes. Here they are restricted to be zero mean, mutually uncorrelated, and uncorrelated with the initial state estimate, though those restrictions need not be imposed in general [6].

Under the aforementioned assumptions, both the a priori state probability density $p(x(k) | Z(k-1))$ and the a posteriori state probability density $p(x(k) | Z(k))$ are Gaussian:

$$p(x(k) | Z(k-1)) \sim \mathcal{N}(\bar{x}(k), \bar{P}(k)) \quad (6)$$

$$p(x(k) | Z(k)) \sim \mathcal{N}(\hat{x}(k), P(k)) \quad (7)$$

allowing the KF to maintain only the mean and covariance matrix, sufficient statistics of the state probability density [6]. This compact form leads to the KF's familiar prediction step, where the system

dynamics in (1), the current state estimate $\hat{x}(k)$, and the current state covariance matrix $P(k)$ are used to generate the predicted state estimate $\bar{x}(k+1)$ and covariance matrix $\bar{P}(k+1)$ conditioned on the measurement set $Z(k)$, and the KF's familiar update step, where the measurement function in (2), the predicted state estimate $\bar{x}(k+1)$, and the predicted state covariance $\bar{P}(k+1)$ are used to generate the updated state estimate $\hat{x}(k+1)$ and covariance matrix $P(k+1)$ conditioned on the measurement set $Z(k+1)$. KF estimation continues recursively in this fashion as time advances, and as new measurements become available [6].

Although the SRIF makes exactly the same linear-Gaussian assumptions as the KF, it differs in its representation of the state probability density. Rather than representing the state probability density $p(x(k) | Z(k))$ by its mean $\hat{x}(k)$ and covariance matrix $P(k)$, the SRIF maintains a square root information form of the state $\hat{y}(k)$ and a square root information form of the covariance matrix $R_{xx}(k)$ [19]. At all times the SRIF variables $\hat{y}(k)$ and $R_{xx}(k)$ relate to their KF counterparts $\hat{x}(k)$ and $P(k)$ by the following definitions:

$$\hat{y}(k) = R_{xx}(k)\hat{x}(k) \quad (8)$$

$$R_{xx}^T(k) \cdot R_{xx}(k) = P^{-1}(k). \quad (9)$$

Analogous definitions extend to the prediction mean $\bar{x}(k)$ and covariance matrix $\bar{P}(k)$, and their square root information forms $\bar{y}(k)$ and $\bar{R}_{xx}(k)$.

The SRIF maintains the state probability density as square root information variables $\hat{y}(k)$ and $R_{xx}(k)$ primarily for numerical reasons [19]. In particular, the variable $\hat{y}(k)$ effectively stores a version of the state vector $x(k)$ normalized by the uncertainty in that state. This form is especially useful in estimation problems in which large scaling discrepancies exist between state variables, as it attenuates numerical errors that may arise when inverting badly-scaled matrices during the KF update step. In addition, representing the state covariance matrix in square root information form effectively doubles numerical precision, as all matrix inversions performed on full covariance matrices in the KF are performed on square root matrices in the SRIF. Furthermore, since (8) and (9) do not specify a particular matrix square root, stable and efficient factorization algorithms such as the Cholesky decomposition may be used in lieu of more expensive and potentially less stable eigenvector decompositions.

In addition to numerical stability, a second benefit to representing the state probability density as square root information variables $\hat{y}(k)$ and $R_{xx}(k)$ is the ability to initialize state estimates with infinite covariance. Because infinity cannot be properly represented on a digital computer, traditional KF implementations often approximate this naïve Bayes initialization by an initial covariance matrix $P(0)$ with “sufficiently

large” entries along the diagonal. This choice yields suboptimal state estimates while the KF converges, which may be particularly detrimental if one or more elements of the state vector are reset frequently. The SRIF does not suffer from this problem, as the naïve Bayes initialization is a situation in which one or more of the state variables have no a priori information. Such a situation is easily reflected in the square root information matrix by zeroing the appropriate elements of $R_{xx}(k)$.

Like the KF, the SRIF has analogous prediction and update steps. However, because these steps operate to maintain the information variables, they have a different form. Whereas the KF directly solves a weighted least squares problem by inverting a potentially ill-conditioned covariance matrix, the SRIF applies a QR-factorization to solve the same least squares problem with square root techniques [19].

B. The Extended Square Root Information Filter

Both the KF and the SRIF operate under the assumptions that the dynamics and measurement functions given in (1) and (2) are linear, and all random variables in the estimation problem are Gaussian. When either of those two assumptions are violated, an approximation to the KF, called the EKF, is often used to perform suboptimal estimation [6]. In the EKF, the dynamics describing the evolution of the state need not be linear:

$$x(k+1) = f(x(k), u(k), v(k)). \quad (10)$$

Similarly, the measurement equation describing the sensor model also does not need to be linear:

$$z(k) = h(x(k), w(k)) \quad (11)$$

although the measurement noise $w(k)$ is often assumed to enter additively into the measurement. The a priori state probability density $p(x(0))$, the process noise density $p(v(k))$, and the measurement noise $p(w(k))$ also do not need to be Gaussian, although in the EKF they are represented by their first and second moments as if they were.

One common variant of the EKF assumes the appropriate state estimation error, $\bar{e}(k) = x(k) - \bar{x}(k)$ or $\hat{e}(k) = x(k) - \hat{x}(k)$, is small. Under this assumption, the nonlinear dynamics of (10) and the nonlinear measurement of (11) are linearized about the state estimate $\bar{x}(k)$ or $\hat{x}(k)$ to yield linearized error dynamics:

$$\bar{e}(k+1) \approx \frac{\partial f}{\partial x} \bar{e}(k) + \frac{\partial f}{\partial v} v(k) \quad (12)$$

$$\nu(k+1) \approx \frac{\partial h}{\partial x} \bar{e}(k+1) + \frac{\partial h}{\partial w} w(k+1) \quad (13)$$

where $\nu(k) = z(k) - \bar{z}(k)$ is the measurement innovation [6]. Equations (12) and (13) restore linearity to the problem, allowing the EKF to utilize

the KF's familiar linear matrix equations during predictions and updates. The price paid for the ease of linearity is optimality: the EKF is not guaranteed to produce minimum mean squared error (MMSE) estimates, unlike the KF.

The approximations made in (12) and (13) can also be used to create an ESRIF equivalent to the EKF [2]. Like the EKF, the ESRIF largely resembles its linear counterpart, with partial derivative Jacobian matrices replacing the state transition matrix $F(k)$, the noise input matrix $\Gamma(k)$, and the measurement matrix $H(k)$. The only notable differences are two nonhomogeneous correction terms that arise in the information state prediction and update equations. These terms correct for the fact that (12) and (13) are linear in the error variables $\hat{e}(k)$ and $\bar{e}(k)$, not $\hat{x}(k)$ and $\bar{x}(k)$.

The ESRIF retains all the numerical advantages over the EKF that the linear SRIF possesses over the KF. The only additional downside to the ESRIF is the fact that any nonlinear elements of the dynamics function $f(\cdot)$ and the measurement function $h(\cdot)$ must be linearized about the estimated state $\hat{x}(k)$ or $\bar{x}(k)$. As a consequence, any states evolving according to the nonlinear dynamics or entering into the nonlinear measurement equations cannot be initialized with infinite covariance (zero information).

C. The Attitude and Position Estimation Algorithm

Skynet's pose estimator applies an ESRIF to statistically fuse its IMU, wheel encoders, and GPS receivers into a real-time feedback signal for autonomous driving in an urban environment. At a high level, the pose estimator consists of 17 "core" states estimated at all times, plus numerous "adjunct" states depending on the current GPS satellite constellation. The 17 core states, always present in the pose estimator, estimate attitude, position, velocity, bias, and time. Attitude is stored as the $ZY'X''$ Euler angles yaw, pitch, and roll required to orient the local geocentric East-North-Up (ENU) coordinate frame with respect to a coordinate frame aligned with IMU axes. Position is stored as the three-element vector position of the IMU in an Earth-centered, Earth-fixed (ECEF) coordinate frame. Velocity is stored as the three-element vector velocity of the IMU in the ECEF frame. The bias state is stored as a six-element vector of slowly-drifting rate gyroscope and accelerometer biases; separate biases are stored for each axis of the rate gyro and each axis of the accelerometer. The time state is stored as a two-element vector of PolARx2e@ GPS clock bias and clock drift rate.

The pose estimator's adjunct states estimate quantities related to individual and paired GPS satellites. Two types of adjunct states are estimated: residual satellite biases, and carrier phase biases. Residual satellite biases (RSBs) estimate

time-correlated biases in the calculated line-of-sight distance from Skynet's GPS antennas to each visible GPS satellite. One bias is estimated for each visible satellite. Carrier phase biases (CPBs) estimate the unknown but constant carrier phase ambiguity obtained when carrier phase measurements from two satellites are double differenced across two PolARx2e@ antennas [20]. These CPBs are estimated as floating (decimal) numbers rather than fixed as integers. Although integer approximations of CPBs are often quite accurate, discontinuities arising in the pose solution as integers are fixed and can be detrimental to autonomous driving [20, 21]. In Skynet's pose estimator, one satellite is selected as the reference satellite against which all double differences are made. Two double differences are created for each remaining satellite: one using the PolARx2e@'s main and first auxiliary antennas, and one using the PolARx2e@'s main and second auxiliary antennas. If the same set of N satellites is tracked across all three PolARx2e@ antennas, this scheme of RSBs and CPBs results in a total of $N + 2(N - 1)$ adjunct states estimated in the pose estimator. As old satellites drop below the horizon and new satellites appear, the number of adjunct states changes to accommodate.

The prediction step of Skynet's ESRIF pose estimator performs a discrete numerical integration using one IMU measurement. The method used is a dual-rate integration method similar to that proposed by Savage [22–24]. In Skynet's pose estimator, the faster of the dual-rate integrations is performed on the IMU itself, which reports integrated incremental changes in orientation and velocity at 400 Hz. The slower of the dual-rate integrations is performed in software. At a high level, the incremental changes in orientation and velocity measured by the IMU are first corrected by the biases, similar to the model discussed in Ohlmeyer [5]. After correction, the corrected measurements are used to integrate the state of the pose estimator. During integration, additional corrections are made for incremental rotation of the Earth, and for apparent centripetal and coriolis accelerations arising from the rotating ECEF coordinate frame. Gravity, as measured by the accelerometers, is also subtracted out using the EGM-96 Earth gravity potential model with a numerically stable formulation of Legendre polynomial recursions [25, 26]. Other elements of the pose estimator state vector: rate gyro and accelerometer biases, clock drift rate, and all CPBs, are modeled as random walk processes. The RSBs are modeled as autocorrelated random processes with characteristic correlation times of 600 s to approximate the time correlation effects of biases introduced by multipath [6].

Measurement updates in Skynet's ESRIF pose estimator are performed asynchronously, that is, when they become available from the sensors. All

localization sensors except the IMU, which is used solely for predictions, are used to update the pose estimate. A scalar speed measurement is generated from the average of the ABS wheel encoders on the rear wheels. This measurement is compared with the velocity magnitude calculated from the current state estimate $\hat{x}(k)$ to generate an innovation for the linearization described in (13). The Ag252 differential correction measurements are incorporated in a manner similar to the ABS wheel speeds: the position of the Ag252 antenna as measured by the differential correction signal is compared with the position of the Ag252 antenna as predicted by the current pose solution to generate an innovation. Note that a scalar yaw rate measurement could also be generated from differential wheel speeds, but, as mentioned in Section IIB, these measurements are not used in Skynet's pose estimator due to significant noise.

Because the PolaRx2e@ provides raw GPS measurements from three antennas simultaneously, its measurement updates provide richer information than the wheel encoders or the Ag252. Rather than extract an innovation from the GPS solution reported by the PolaRx2e@ directly, individual innovations are instead extracted for all pseudo-ranges, Doppler shifts, and carrier phase double differences measured on the PolaRx2e@'s three antennas. The result is a large set of PolaRx2e@ measurements: one pseudo-range for each tracked satellite on each antenna tracking it, one Doppler shift for each tracked satellite on each antenna tracking it, and two double differenced carrier phases for each satellite except the reference satellite across the PolaRx2e@ antenna pairs. Thus, for the same N satellites tracked by each PolaRx2e@ antenna, a total of $3N + 3N + 2(N - 1)$ measurements are formed at each PolaRx2e@ update. For each of these measurements, the ESRIF implements the full GPS software interface specified in the GPS interface control document [27]. In other words, the ESRIF computes pseudo-range, Doppler shift, and carrier phase innovations by comparing raw GPS measurements made at the PolaRx2e@ with the expected raw measurements implied by the current pose solution. The broadcast GPS satellite orbit and signal propagation models are also augmented with WAAS signals, as specified in the WAAS interface control document [28]. The signal propagation models are further refined with tropospheric corrections based on current atmospheric conditions [29–32].

Utilization of the raw GPS measurements rather than the filtered PolaRx2e@ GPS solution within the ESRIF creates a tightly-coupled pose estimator. Such a configuration is particularly suited to urban environments, as it allows GPS information to be incorporated as it is acquired, even when fewer than four satellites are visible [4]. In addition, direct utilization of raw GPS observables permits errors to be modeled on a satellite-by-satellite basis [33].

These errors are modeled as independent, zero mean, Gaussian, and white for each pseudo-range and Doppler shift. Errors are similarly modeled as zero mean, Gaussian, and white for double differenced carrier phase measurements, though proper cross correlations are computed between double differences that have either a satellite or antenna in common. Correlations between Doppler shifts and double differenced carrier phases are not modeled; this suboptimal decision is made to avoid additional filter states needed to prewhiten short-term errors in the PolaRx2e@ tracking loops. Skynet's pose estimator takes advantage of individually-modeled satellite errors to perform statistical hypothesis tests validating each measurement before it is fused into the pose estimate. These tests are more powerful than those that could be done on the GPS receiver itself, as they incorporate historical information from Skynet's other three localization sensors. Similar hypothesis tests are performed on ABS wheel encoder and Ag252 differential correction measurements to ensure robustness and consistency in the resulting pose solution.

In addition to the pose estimator, Skynet runs a map aiding system, called the PosteriorPose algorithm, which augments pose estimates with information from vision-based road cues such as lane lines and stop lines [11]. The map aiding algorithm is implemented as a particle filter, matching computer vision measurements of road cues to their surveyed locations in the road network. The algorithm makes feed-forward corrections to output from the pose estimator; for computational reasons, map aided pose estimates are not fed back into the pose estimator itself.

IV. ORDERED SENSITIVITY ANALYSIS IN THE DESIGN OF CORNELL'S POSE ESTIMATOR

Cornell's pose estimator algorithm was field proven in the 2007 DARPA Urban Challenge, where it served as the positioning signal Skynet used to complete 60 mi of autonomous urban driving [14]. This success validates the design decisions described in Section III as a whole, but gives little insight into the effects of each individual decision on the performance of the final algorithm. In order to gain that insight, the pose estimator must be deconstructed. Each critical design decision aspect is therefore systematically removed, to determine the pose estimator's sensitivity to its design in a practical setting.

While the UCE provided an excellent system level endurance test for Skynet, its sheer magnitude makes it unwieldy for individual unit tests against the pose estimator. In particular, the UCE's 6-hr duration makes it computationally undesirable to use for algorithm evaluation in its entirety, and it



Fig. 3. Bird's eye view of DARPA Urban Challenge NQE Area C course. The 20 min compact course includes tree cover, low buildings, and power lines, making it ideal for unit tests against pose estimator. Rectification and registration errors have not been entirely removed from image.

is unclear which, if any, portion of the UCE might be better or worse to use for an evaluation. More importantly, exogenous environmental stimuli in the UCE frequently resulted in complex interactions between the competing robots. What appeared *prima facie* to be algorithmic faults were often mere manifestations of differing interpretations of the UCE rules, and are left to other papers [14, 34].

Rather than contend with such complex and unpredictable interactions, this study uses an excerpt of data taken from Skynet's logs of the DARPA Urban Challenge NQE. Unlike the UCE, which consisted of 6 hr of largely uninterrupted autonomous driving, the NQE was a set of 3 short courses designed to test individual components of urban driving behavior. NQE Area C, shown from a bird's eye view in Fig. 3, provides the ideal setting for unit tests against the pose estimator. The 20 min course is compact, requiring robots to loop back over its three city blocks and two intersections several times during the prescribed mission. The course also contains features that often challenge satellite navigation: trees, low buildings, and power lines, but these features are not so dense that GPS is unavailable. Finally, Skynet completed NQE Area C flawlessly, yielding a log of synchronized raw sensor data, processed pose data, and video records for use as an externally-validated ground truth solution.

In the sections that follow, the key design elements comprising Skynet's pose estimator are systematically removed. The pose estimator's design elements

are removed sequentially and cumulatively; each section removes one additional design element from the remainder of the pose estimator. This approach is taken primarily for the sake of brevity, to avoid an exponential walk through the pose estimator's design space. Design elements are roughly removed in order of implementation and monetary cost: first map aiding, then differential corrections, followed by integrity monitoring, WAAS corrections, and carrier phases. Finally, pose estimator performance is studied in extended signal blackouts, where there are no external navigation signals at all. This ordering is selected in an attempt to magnify the differences between successive variations of the pose estimator algorithm, though it is possible that some combinations of design features not tested in this study may also be insightful.

To statistically evaluate the variants of the pose estimator algorithm generated below, each pose estimator variant is run on the NQE Area C data log. The output of each pose estimator variant is then compared with the baseline output, recorded on the day Skynet completed NQE Area C. For this comparison, the baseline pose estimator output is treated as a field-verified ground truth, where the reported estimation mean $\hat{x}(k) = R_{xx}^{-1}(k)\hat{y}(k)$ and covariance $P(k) = R_{xx}^{-1}(k)R_{xx}^{-T}(k)$ describe the probability density function of Skynet's true pose:

$$x(k) \sim \mathcal{N}(\hat{x}(k), P(k)). \quad (14)$$

In other words, Skynet's true pose $x(k)$ at time index k is henceforth taken as a Gaussian random variable with mean $\hat{x}(k)$ and covariance $P(k)$ as calculated when Skynet completed NQE Area C on October 29, 2007. This baseline data is then used to cast the sensitivity analysis as a statistical hypothesis test, by questioning whether each variant of the pose estimator algorithm produces output that is likely to have been drawn from this true pose probability density. If so, the variant will have produced a solution that is statistically indistinguishable from the logged truth data, revealing the algorithm's insensitivity to the changes made in that variant.

The output of each pose estimator variant $x^\ddagger(k)$ is first tested to see whether it is likely to have been drawn from the true pose density given in (14). This comparison is made by noting that if $x^\ddagger(k) \sim \mathcal{N}(\hat{x}(k), P(k))$ at time index k , then the test statistic

$$\tau(k) = [x^\ddagger(k) - \hat{x}(k)]^T P^{-1}(k) [x^\ddagger(k) - \hat{x}(k)] \quad (15)$$

will be distributed as a chi-squared random variable with n degrees of freedom, where n is the length of the vectors $x^\ddagger(k)$ and $\hat{x}(k)$ [35]. To evaluate specifically the pose estimator's final position solution, the test is conducted only on the in-plane East and North components of the pose estimators' position estimates. These in-plane components are

TABLE I
Statistical Similarity between Pose Estimator Variants, Removing One Component at a Time

Pose Variant	Diff %	K-L	Pos Error (m)		Att Error (°)		Discontinuity (m)	
			Avg	Std	Avg	Std	Avg	Std
No Map Aiding	4.43	278766	0.087	0.049	-0.21	3.27	0.0046	0.0318
+ No Differential Corrections	100.0	3389763	1.188	0.326	-0.19	3.27	0.0149	0.0332
+ No Integrity Monitoring	98.64	2711164	0.831	0.522	-0.18	3.27	0.0223	0.0446
+ No WAAS	89.36	2063906	0.485	0.373	-0.15	3.26	0.0220	0.0553
+ No Carrier Phases	98.58	2399391	0.691	0.268	-1.08	0.50	0.0095	0.0390
Differential Corrections Blackout	85.9	1093692	0.535	0.391	-0.19	3.27	0.0125	0.0329
Full Blackout	91.6	59060233	4.977	3.095	-1.87	0.45	N/A	N/A
Full Blackout, No Speeds	92.6	4.7×10^{12}	557.625	674.091	-1.91	0.41	N/A	N/A

generated by a common projection into an East-North plane tangent to the WGS-84 ellipsoid Earth model at a point near the center of the Urban Challenge course. The test statistic $\tau(k)$ is therefore distributed as a chi-squared random variable with two degrees of freedom, for each evaluated time k . In the following sensitivity analysis, thresholds placed on $\tau(k)$ will be used to judge whether $x^{\ddagger}(k)$ is likely to have come from the same probability density function as the true $\hat{x}(k)$ at each particular evaluation time (iteration) k . The number of iterations where $x^{\ddagger}(k)$ reasonably matches $\hat{x}(k)$ will then be used as a measure of the proximity of the pose estimator variant to the true pose density.

Seven other statistical quantities are also computed to aid in the comparison of pose estimator variants. To quantitatively evaluate the proximity of each variant to the baseline solution, a symmetric K-L divergence is computed between each variant and the baseline. The K-L divergence is a scalar measure of the distance between the two probability density functions in terms of the statistical information available to discriminate between them [36]; thus, the measure, which is always positive, includes errors in mean, covariance, and even higher order moments. Given two normal distributions $x_1 \sim \mathcal{N}(\hat{x}_1, P_1)$ and $x_2 \sim \mathcal{N}(\hat{x}_2, P_2)$, the K-L divergence is written as

$$D_{\text{KL}}(x_1, x_2) = \frac{1}{2} \left[\ln \left(\frac{|P_1|}{|P_2|} \right) + \text{tr}(P_1^{-1} P_2) + (\hat{x}_1 - \hat{x}_2)^T P_1^{-1} (\hat{x}_1 - \hat{x}_2) - n \right] \quad (16)$$

where n is the dimension of the variables. The K-L divergence is not symmetric, unfortunately; typically, x_1 is the true distribution. Alternatively, a symmetric form can be used,

$$D_{\text{KL}} = D_{\text{KL}}(x_1, x_2) + D_{\text{KL}}(x_2, x_1). \quad (17)$$

In the subsequent sensitivity studies, the symmetric divergence is used, which is calculated as the average of the K-L divergence between the variant and baseline. For computational feasibility across the

entire data set, this divergence is computed under the approximation that the pose solution at each time step is independent of the pose solution at every other time step.

Finally, to aid with physical insight, the position and attitude errors (mean and standard deviations) are also computed for each variant. And, to evaluate each variant's practical utility as a feedback signal for closed-loop autonomous driving, the mean and standard deviation of the "discontinuity" in the ESRIF position update are also calculated. The discontinuity is formally defined as the norm of the change in car position, from before the measurement update to after. This metric is calculated at each time step, and the mean and standard deviation are computed.

The statistical results calculated for each pose estimator variant in the ordered analysis are summarized in Table I, and are interpreted in detail in Sections IVA–IVE which follow. Table I lists, for each pose estimator variant, the percent of iterations at which the variant is significantly different from the baseline (% Diff), the symmetric K-L divergence between the variant and the baseline (K-L), the mean and standard deviation of the positioning error between the variant and the baseline (Pos Error), the mean and standard deviation of the heading error between the variant and the baseline (Att Error), and the mean and standard deviation of the ESRIF update discontinuities in each variant (Discontinuity), where all variants have been compared with the baseline pose solution recorded when Skynet competed in the DARPA Urban Challenge.

A. Sensitivity To Map Aiding

The sensitivity analysis of the pose estimator begins by removing its map aiding algorithm. Cornell's map aiding algorithm utilizes the DARPA-provided RNDF and onboard computer vision algorithms to search for surveyed landmarks such as lane lines and stop lines to aid in localization. These landmark-based onboard sensing cues are then fused with the output of the pose estimator in a bootstrap particle filter to create a posterior estimate of Skynet's position and heading [11]. This estimate is sent to

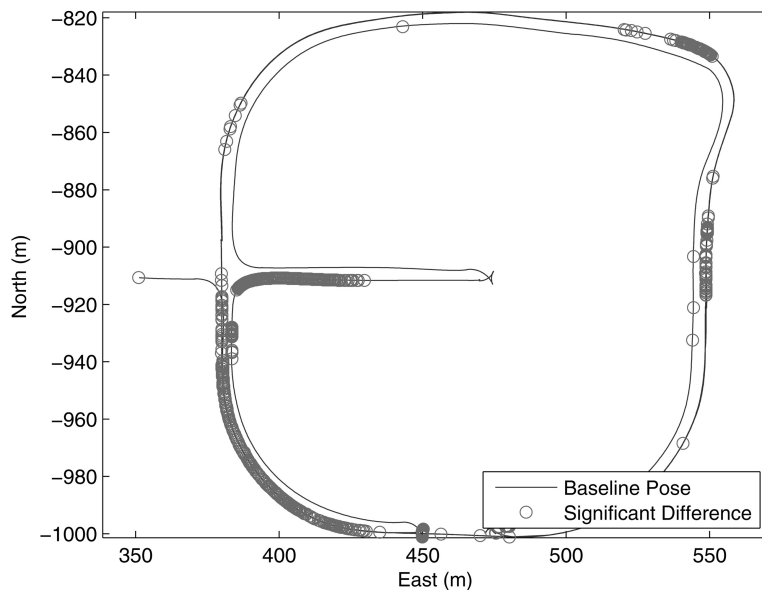


Fig. 4. Skynet's position when baseline pose estimator with map aiding produces statistically different pose estimate than pose estimator without. Significant differences tend to occur after sharp turns, where map aiding provides positioning cues along new directions.

the intelligent planner at a 10 Hz rate as the preferred localization signal for planning and path tracking.

The pose estimator is thus run on the Area C log data with map aiding removed. Hypothesis testing this variant at the one-sided 5% significance level (the 95% confidence level) using the test statistic $\tau(k)$ from (15) reveals statistically significant pose estimates in only 532 of the 12020 iterations in the NQE Area C log data. The average difference between the two pose variants across the entire data set is 8.7 cm, with a standard deviation of 4.9 cm. Although the two variants produce slightly different pose solutions, it is evident that the pose estimator is relatively insensitive to map aiding. This result is consistent with the findings of Miller and Campbell, who show small (but statistically significant) improvements with map aiding in a full GPS environment, and large improvements with map aiding in GPS blackouts [11].

Although the pose estimator is relatively insensitive to map aiding augmentations in a full GPS environment, it is instructive to examine Skynet's position when map aiding produces a significantly different pose solution. These locations are plotted in Fig. 4. Most significant differences occur during shallow turns, or immediately after sharp turns. Some of these differences are likely due to disagreement between the surveyed waypoints defining the course and the painted road lines. Differences occurring immediately after sharp turns most likely represent the information gained when Skynet acquires new lane lines perpendicular to its former direction of travel. In these cases, the newly-acquired lane lines

provide information along a new axis, allowing further refinement to Skynet's pose estimate.

B. Sensitivity to Differential Corrections

With map aiding removed from the pose estimator, the next design element to be removed is the differential correction signal. As mentioned in Section IIB, the HP and VBS differential correction signal decoded from the Ag252 are the primary source of submeter absolute positioning information available to Skynet's pose estimator. With map aiding already removed in Section IVA, the resulting pose estimator variant relies on raw GPS observables and WAAS corrections as its remaining sources of absolute positioning information.

Without its primary source of submeter absolute positioning information, the pose estimator suffers considerably. Not surprisingly, this pose estimator variant yields statistically significant pose estimator errors at the 5% significance level in all 12020 time steps contained within the NQE Area C data log. As shown in Table I, these differences yield a substantial factor of 10 increase in the K-L divergence between this pose estimator variant and the baseline recorded at the NQE. In physical terms, this corresponds to a mean difference of 1.19 m between this variant and the baseline, with a standard deviation of 0.33 m. Fig. 5 plots the time history of the variant's errors in the East and North directions throughout the log data.

Although the pose estimator's absolute accuracy suffers greatly when denied its most important source of submeter positioning information, it does not suffer from large discontinuities. Fig. 6 plots

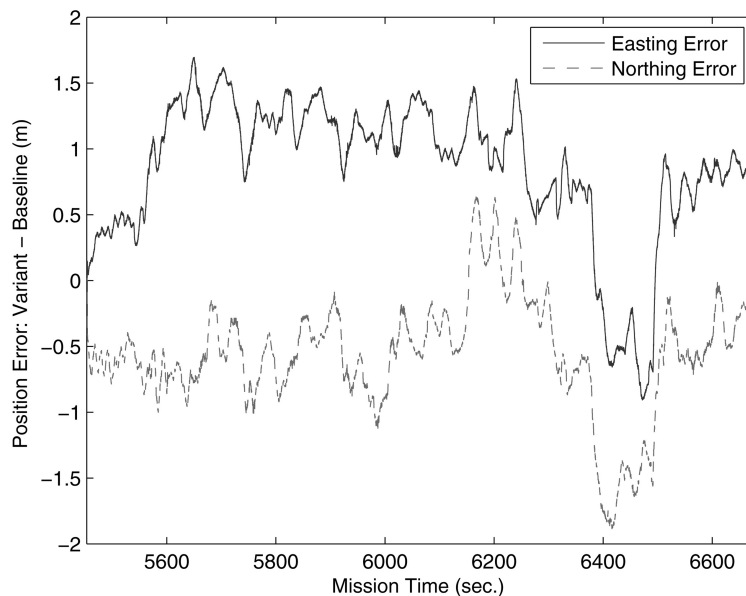


Fig. 5. When denied differential corrections, pose estimator produces ground track with mean error of 1.19 m. Although the mean error is approximately one vehicle width, pose solution has no sudden discontinuities that would be devastating to autonomous driving.

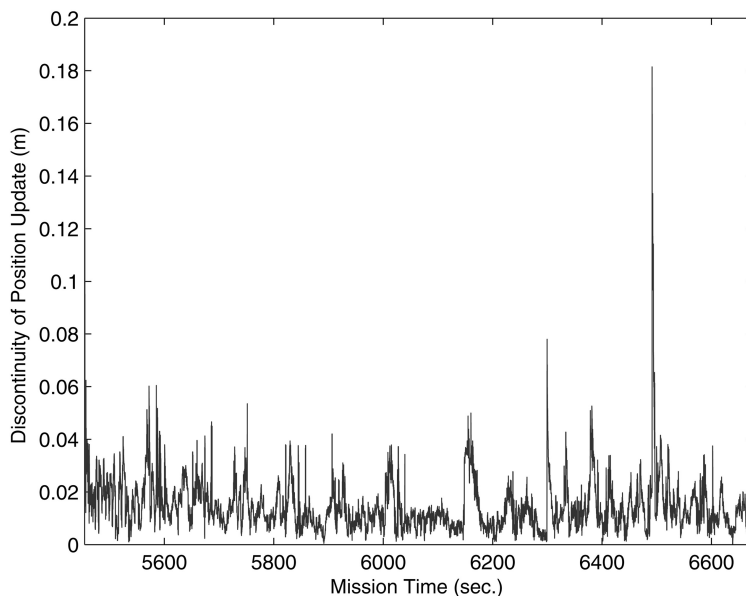


Fig. 6. Although denied differential corrections, the magnitude of the pose estimator's update discontinuities are still very small. These small discontinuities are critical for autonomous driving, as large discontinuities create instability when fed back in autonomous path following loops.

the magnitudes of the discontinuities in the pose estimator's updates when it is denied differential corrections. In this variant of the pose estimator, these updates have a mean discontinuity of 1.5 cm with a standard deviation of 3.3 cm. The fact that these discontinuities are small is not guaranteed by the EKF or the ESRIF, which place no constraints on the continuity of the pose solution during measurement updates. Rather, it is the result of a deliberate effort

to design the system for integrity, robustness, and precision even before accuracy, as these qualities are essential for autonomous navigation. In particular, large discontinuous jumps in the pose solution translate directly into large discontinuous steps in path tracking error. At worst, an aggressive path tracking controller will respond to such errors with unsafe and possibly catastrophic swerving. At best, the path tracking controller's bandwidth must be lowered

considerably if these discontinuous errors cannot be removed from the positioning algorithm itself.

In addition to precision, the pose estimator's attitude solution retains its accuracy even without differential corrections. In particular, the pose estimator variant's heading (yaw) solution, used for Skynet's heading measurement, differs from the baseline solution by an average of only -0.19° with a standard deviation of 3.27° . It is thus apparent that Skynet's pose estimator receives very little attitude information from differential corrections, despite the fact that they are its most important source of absolute positioning information.

C. Sensitivity to Filter Integrity Monitoring

After differential corrections, the next design element removed from the pose estimator is its suite of filter integrity monitoring (FIM) hypothesis tests. In general Bayesian estimation, these FIM tests are meant to detect when the filter has become inconsistent, that is, when one or more of the linear, Gaussian, or correlation assumptions of the EKF or ESRIF have been violated [6]. In the context of pose estimation for autonomous navigation, however, such *ex post facto* FIM tests are not helpful. For the same reasons discussed in Section IVB, it is far better to maintain filter consistency by identifying and rejecting faulty measurements before they corrupt the fused information in the pose estimator [1]. Skynet's pose estimator therefore performs two online FIM tests. The first test, meant to detect (but not identify) faulty measurements, is based on a global chi-squared statistic formed over the entirety of each measurement vector. This FIM test is performed on each measurement vector, whether generated from the PolRx2e@, the Ag252, or the ABS wheel encoders. Measurement vectors from the Ag252 or ABS wheel encoder found to be faulty in this hypothesis test at the 1% significance level (the 99% confidence level) are discarded. If a measurement vector from the PolRx2e@ is found to be faulty, a second test is performed to attempt to identify offending elements of the measurement vector. The second test is similar to the cycle slip detection technique used by Mohiuddin and Psiaki, which tests each satellite for faults individually by evaluating the integrity of all measurements except those from the satellite in question [37]. Skynet's pose estimator performs a substantially cheaper variant of this test by pairing each satellite with the most consistent satellite in an attempt to identify and reject, at the 1% significance level (the 99% confidence level), carrier phase cycle slips and large multipath distortion without discarding the entire measurement vector.

With the FIM tests turned off, the pose estimator blindly accepts all measurements without question. Since the filter still has no map aiding or differential corrections, it fails the 5% significance test in

11856 of 12020 time steps. Interestingly, these results suggest that this variant of the pose estimator is actually statistically closer to the baseline pose estimator than the differential corrections variant tested in Section IVB. This result is further corroborated with K-L divergences in Table I, and with the position output itself, which differs from the baseline algorithm by an average of 83.1 cm with a standard deviation of 52.2 cm. From these results it is evident that FIM tests do not reject only faulty measurements, and some valid information may also be discarded. This extra information does not appear to fit the ESRIF's linear-Gaussian modeling assumptions.

Despite some gains in accuracy, the practical price of including all measurements is paid in the pose estimator's robustness and precision. Aside from the larger standard deviation between this pose variant and the baseline, it also has average update discontinuities of 2.2 cm with a standard deviation of 4.5 cm. More importantly, the variant is less stable, with 11 separate instances of discontinuities exceeding 20 cm and one exceeding 70 cm. These discontinuities are most commonly due to the appearance of new GPS satellites at low elevations, whose measurements tend to suffer more from ionospheric, tropospheric, and multipath distortion. Paradoxically, these error-prone satellite measurements also tend to offer position information along new axes, thereby affecting the pose estimate more than measurements from old satellites. A time history of the largest of these discontinuities is shown in Fig. 7. If it were used as a feedback signal, the path tracking controller would likely make an unsafe maneuver.

D. Sensitivity to WAAS Corrections

The fourth design element removed from the pose estimator is the WAAS signal. The WAAS signal nominally includes improved positioning, pseudo-range, and ionospheric corrections for each individual satellite [28]. More importantly for Skynet's pose estimator, the WAAS corrections include uncertainty estimates and models of their degradation over time. These uncertainty parameters allow the satellites to be validated more accurately within the pose estimator, ensuring each satellite's information is incorporated with a fair weighting.

When WAAS corrections are removed, the pose estimator defaults to a very conservative weighting of satellite information. Satellites near the horizon are assumed to have large variances, and thus contribute almost no information to the final pose solution. Even without differential corrections or integrity monitoring, this conservative approach produces some useful results. At the 5% significance level, this variant is statistically different from the baseline pose estimator in 10741 of 12020 time steps, and

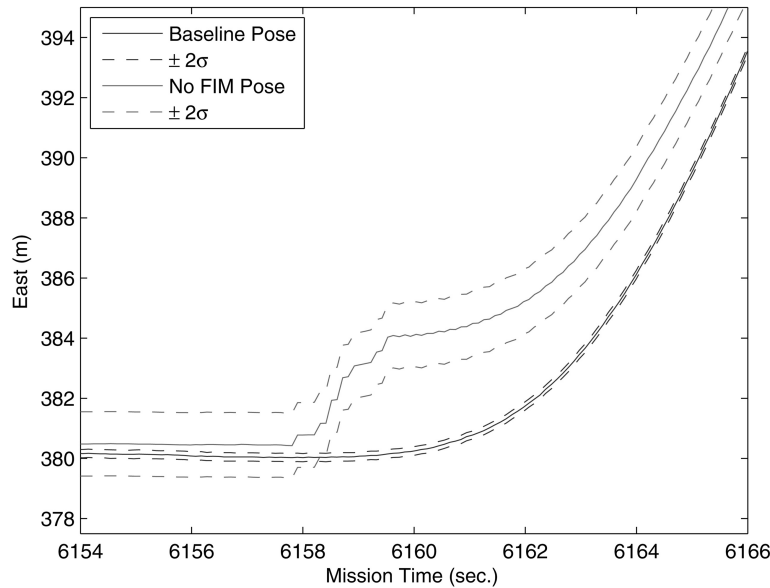


Fig. 7. When FIM is turned off, pose estimator suffers from large discontinuities.

Table I indicates that this conservative approach is actually closer to the baseline pose estimator than many other variants. Intuitively, the result makes sense: by adopting a skeptical view of the information coming from low-elevation satellites, this variant is on average less sensitive to the most common signal errors that caused difficulties for the no-FIM variant examined in Section IVC. This variant has an average error of only 48.5 cm with a standard deviation of 37.3 cm compared with the baseline.

This result should not be taken as evidence against the usefulness of WAAS. Although this variant is statistically closer to the baseline truth on average, it still suffers from the same acute problems as the FIM variant tested in Section IVC. The discontinuities in this variant's ESRIF updates have an average of 2.2 cm with a standard deviation of 5.5 cm, indicating that this variant experiences more extreme discontinuities than the one using WAAS corrections. The result is corroborated by examining the individual update discontinuities: there are 23 separate instances of discontinuities exceeding 20 cm, and there are 2 instances where the variant differs from the baseline solution by more than 3 m. These results suggest that the pose estimator's accuracy is relatively unaffected by WAAS corrections, though WAAS corrections do aid in the algorithm's overall robustness and precision by reducing discontinuities.

To make the analysis more physical, this variant can be compared statistically with the no-FIM variant examined in Section IVC. Hypothesis tests at the 5% significance level indicate that this variant differs from the no-FIM variant in Section IVC in only 1089 of 12020 time steps, and the symmetric K-L divergence between the variants is only 25673. Comparing this

value to other much larger divergence values in Table I supports the previous conclusion: WAAS primarily aids in the robustness and precision of the pose estimator, not in its absolute accuracy.

E. Sensitivity to GPS Carrier Phases

The final external satellite measurement to be removed from the pose estimator is measured carrier phases of GPS satellites. These carrier phases are used in two places in Skynet's pose estimator: as measurements of satellite Doppler shifts with respect to the receiver, and for double differencing signals between Skynet's antennas. These carrier phase signals are an important source of attitude information. Intuitively, each Doppler shift measurement yields velocity information along the line of sight between Skynet and the satellite to which the measurement corresponds. Similarly, each carrier phase double difference yields relative positioning information of one of Skynet's antennas with respect to another, a measurement made extremely accurate due to the noise canceling properties of the double difference operator applied to antennas in close proximity.

The pose estimator's performance is further degraded from the no-WAAS variant examined in Section IVD when both these measurements are removed. At the 5% significance level, this variant produces significantly different output from the baseline in 11847 of 12020 time steps. In addition to a larger K-L divergence, shown in Table I, algorithmic errors are larger, averaging at 69.1 cm with a standard deviation of 26.8 cm. Interestingly, the variant achieves surprisingly low attitude errors, averaging only -1.1° with a standard deviation of 0.5° . These

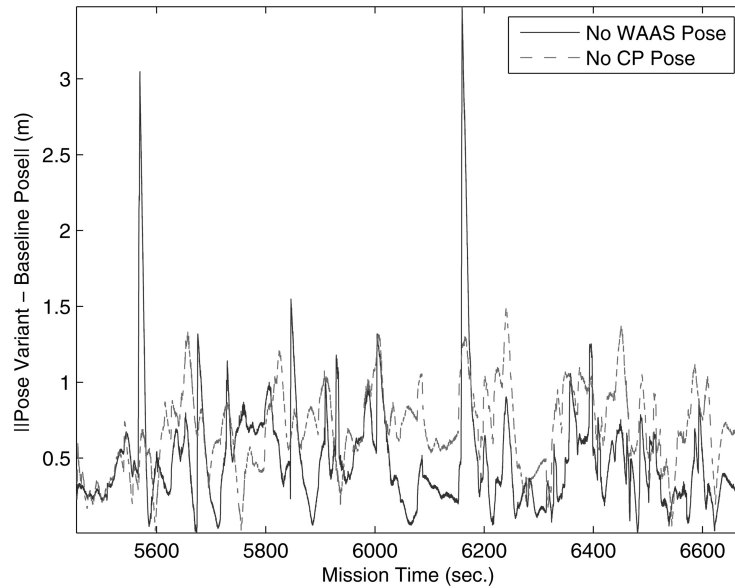


Fig. 8. Incorporating Doppler shifts and carrier phases (CP) without online validation yields a less robust pose solution than ignoring those measurements entirely.

errors may be deceptively low, however, due to the short mission duration relative to the stability of Skynet's rate gyros.

Although the pose estimator still retains the conservative assumptions discussed in the no-WAAS variant examined in Section IVD, it is evident that the benefits of incorporating carrier phase measurements trade off heavily with algorithm performance. In particular, since these measurements are often modeled as being much more accurate than pseudo-ranges, the pose estimator is very sensitive to common non-Gaussian errors such as cycle slips. In Section IVD, this resulted in a number of large discontinuities in the pose solution. In the carrier phase variant discussed here, those measurements are ignored entirely, resulting in a maximum discontinuity of less than 5 cm over the entire data set.

Fig. 8 compares baseline errors between the no-carrier-phases pose variant and the variant including carrier phases studied in Section IVD. On average the two variants are quite similar: at the 5% significance level they differ from each other in only 159 of 12020 time steps, and their symmetric K-L divergence is only 17551.5. Yet despite the similarities, the conservative pose variant using only pseudo-ranges is more precise and robust, making it more suitable for autonomous navigation in the absence of statistical measurement validation. In light of the slow drift rates in Skynet's tactical grade IMU, the price paid in degraded attitude accuracy is outweighed by the more conservative filter's precision and robustness, assuming statistical measurement validation is not performed. These results do not consider measurement integrity monitoring, however;

it is clear that carrier phase measurements are subject to non-Gaussian errors such as cycle slips. Thus, if measurement validation is to be performed, variants including carrier phase measurements are preferred, since biases in vehicle yaw translate directly into steady state errors in path tracking.

V. SENSITIVITY TO SIGNAL BLACKOUTS

Sections IVA–IVE have deconstructed the pose estimator, removing all sources of absolute positioning information except the most basic treatment of GPS pseudo-ranges themselves; the baseline, therefore, is IMU+GPS L1 pseudo-ranges (i.e., code tracking) with all other elements turned off. In essence, these sections have considered the ramifications of pose estimator design decisions on autonomous robotics in an ideal GPS signal environment. All satellite signals have been treated as readily available, leaving the user to decide which information to use and which to ignore. Such an experimental assumption disregards two events with significant impact on the practical deployment of autonomous field robots: complete signal blackouts, and (less intuitively) signal reacquisitions.

From a strict Bayesian perspective, the EKF and ESRIF naturally handle blackouts and reacquisitions: the pose solution is predicted when no new measurements are available, and it is updated when they are. Yet this statistically rigorous solution carries the same practical risks first discussed in Section IVB. Not only does the pose solution degrade over time as numerical integration errors mount, but signal reacquisition after a long measurement drought tends to produce and even magnify discontinuous jumps in

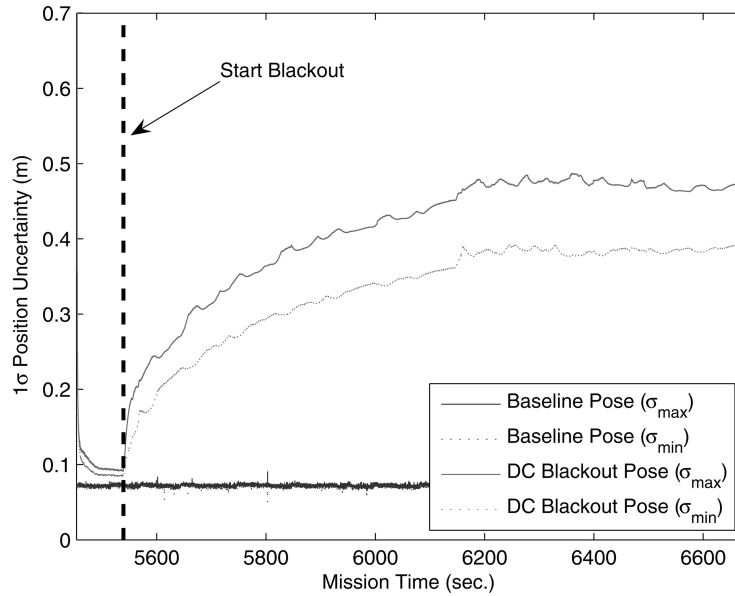


Fig. 9. Square roots of singular values of covariance matrices for baseline pose estimator and one in which differential corrections (DC) have been blacked out at mission time 5539.0. Although pose solution degrades rapidly when differential corrections are first lost, degradation tapers to slow pace.

the pose solution. The authors addressed these issues as part of Cornell's 2005 DARPA Grand Challenge entry [12], and more advanced solutions are not pursued here.

In keeping with the sensitivity analysis conducted so far, the baseline pose estimator is instead examined in a statistical sense in two types of signal blackouts: loss of differential corrections, and full loss of signal. Both of these events are tested by denying the baseline pose estimator access to the appropriate signals in the data log, as well as external map aiding information. Aside from these signal omissions, other design elements are retained: FIM, WAAS, and carrier phases. The mission time of the signal loss will be the same in each experiment: at 5539.0 s Skynet remains stationary in the data log up until this time, giving the pose estimator approximately 86 s of full signal visibility for initialization purposes. Once dropped, the signals remain unavailable to the end of the data log, approximately 19 min later.

The first experiment drops only differential corrections. This variant's K-L divergence in Table I reveals a surprising proximity to the baseline solution with full information. The result suggests that full and continuous coverage from a differential corrections signal may not be necessary, due to the temporal "memory" of the ESRIF and the quality of Skynet's IMU. Fig. 9 quantifies this claim by plotting the minimum and maximum 1σ position uncertainties, $\sigma_{\min}(k)$ and $\sigma_{\max}(k)$, as defined by the pose estimator's covariance matrix. These values are obtained via singular value decomposition of the pose estimator's

covariance matrix:

$$P(k) = U(k) \cdot \begin{bmatrix} \sigma_{\max}^2(k) & 0 \\ 0 & \sigma_{\min}^2(k) \end{bmatrix} \cdot V^T(k) \quad (18)$$

where $U(k)$ and $V^T(k)$ are unitary matrices. Intuitively, $\sigma_{\min}(k)$ and $\sigma_{\max}(k)$ correspond to 1σ error bounds (in meters) in the most and least uncertain directions of the pose solution. Although the pose solution degrades rapidly when differential corrections are first lost, the degradation slows over time. More importantly, if submeter accuracy is a design goal as it was in the Urban Challenge, then these results indicate that the pose estimator can tolerate long stretches of time without differential corrections.

The second experiment drops all external satellite signals. Two variants of the pose estimator are run in the total signal blackout: one just integrating the IMU, and one updating IMU integrations with wheel speed measurements. The experiment results in significantly degraded pose solutions: with no absolute position information available after the signal blackout, the effects of steadily-growing numerical integration errors quickly overwhelm information obtained when the external navigation signals were present. Fig. 10 plots the Euclidean distances between the two pose estimator variants and the baseline pose solution, to quantify the rate at which the estimator loses information during the blackout. The rate is rapid: both variants reach 1 m error levels approximately 25 s after the beginning of the blackout.

The K-L divergences in Table I highlight the importance of using wheel speed sensors to slow the growth of numerical integration and dead

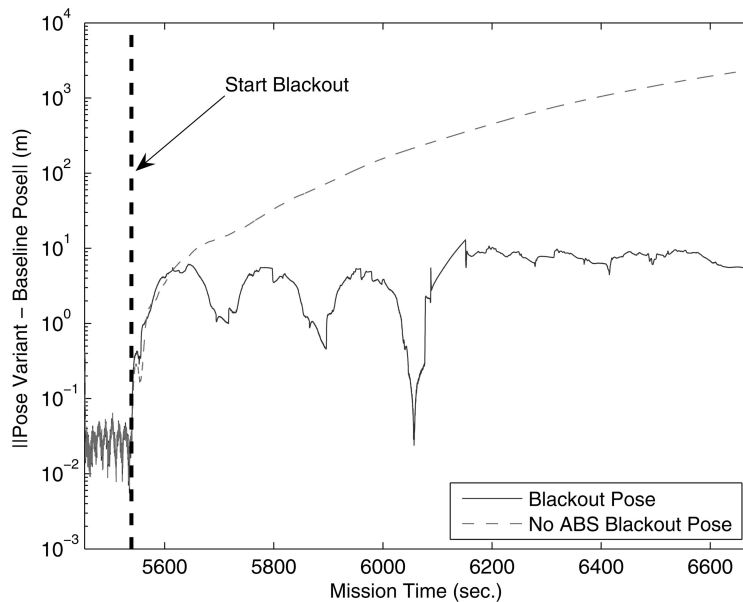


Fig. 10. Euclidean error from baseline for two pose variants in GPS blackout: one using ABS wheel speed measurements (Blackout Pose) and one not using ABS wheel speeds (No ABS Blackout Pose). All external satellite positioning signals have been blacked out at mission time 5539.0.

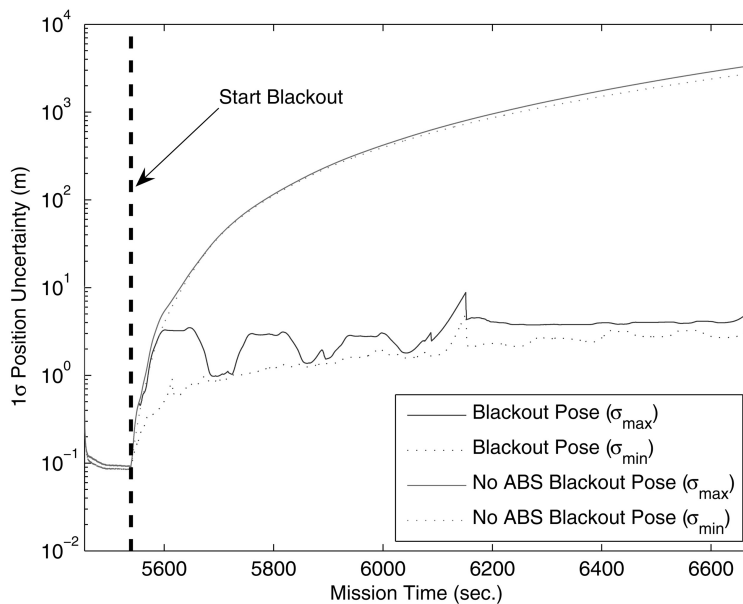


Fig. 11. Square roots of singular values of covariance matrices for pose estimators with and without wheel speed measurements. All external satellite positioning signals have been blacked out at mission time 5539.0.

reckoning errors. Although both pose estimator variants experience similar errors for the first 25 s of signal blackout, the variant utilizing wheel speeds experiences final errors more than two orders of magnitude lower than the variant without wheel speeds. This result is due to the ESRIF aggregating information to estimate mutual correlations between all elements of the pose estimator state. These correlations, stored in the square root information

matrix, are estimated from external navigation signals and other sensor measurements, when they are present. When navigation signals are blacked out, the ESRIF uses the correlation model to update its entire state estimate from wheel speed measurements. This allows the variant utilizing wheel speeds to occasionally decrease its position uncertainty, as shown in Fig. 11. In contrast, the variant ignoring wheel speeds makes no measurement updates,

TABLE II
Summary of Statistical Similarity between Pose Estimator Variants (all components)

Pose Variant	Diff %	K-L		Pos Error (m)			Att Error (°)			Discontinuity (m)		
		#	Rank	Avg	Std	Rank	Avg	Std	Rank	Avg	Std	Rank
011110	0.73	270040	1	0.065	0.038	1	-0.21	3.27	15	0.0054	0.0318	12
010110	1.27	272311	2	0.071	0.040	2	-0.17	3.26	8	0.0047	0.0386	11
111110IV.A	4.43	278766	3	0.087	0.049	3	-0.21	3.27	16	0.0046	0.0318	10
110110	8.51	282268	4	0.094	0.053	4	-0.17	3.26	10	0.0039	0.0386	9
111010	18.4	349037	5	0.135	0.169	6	-0.15	0.66	5	0.0070	0.0351	14
011010	19.9	354119	6	0.135	0.178	5	-0.18	3.27	12	0.0081	0.0357	18
110010	20.6	353854	7	0.142	0.172	8	-0.12	0.61	1	0.0064	0.0417	13
010010	21.4	360150	8	0.141	0.184	7	-0.15	3.26	3	0.0075	0.0423	15
000110	46.0	332243	9	0.168	0.103	9	-1.14	0.61	19	0.0011	0.0385	3
000010	46.0	332243	10	0.168	0.103	10	-1.14	0.61	20	0.0011	0.0385	4
001110	48.3	337309	11	0.178	0.100	11	-1.64	1.53	25	0.0030	0.0318	7
001010	48.3	337309	12	0.178	0.100	12	-1.64	1.53	26	0.0030	0.0318	8
100110	50.4	353608	13	0.189	0.115	13	-1.15	0.62	21	0.0011	0.0385	1
100010	50.4	353608	14	0.189	0.115	14	-1.15	0.62	22	0.0011	0.0385	2
101110	52.6	355994	15	0.196	0.110	15	-1.64	1.55	27	0.0028	0.0318	5
101010	52.6	355994	16	0.196	0.110	16	-1.64	1.55	28	0.0028	0.0318	6
110000IV.D	89.4	2063906	17	0.485	0.373	20	-0.15	3.26	4	0.0220	0.0553	29
010000	91.7	2536538	18	0.724	0.471	23	-0.14	3.26	2	0.0283	0.0598	32
000100	92.5	2048353	19	0.392	0.160	17	-1.17	0.53	23	0.0079	0.0390	16
000000	92.5	2048353	20	0.392	0.160	18	-1.17	0.53	24	0.0079	0.0390	17
010100	97.1	1933871	21	0.464	0.149	19	-0.16	3.26	6	0.0116	0.0389	22
110100	98.4	2427362	22	0.770	0.283	24	-0.16	3.26	7	0.0098	0.0389	21
100100	98.6	2399391	23	0.691	0.268	21	-1.08	0.50	17	0.0095	0.0390	19
100000IV.E	98.6	2399391	24	0.691	0.268	22	-1.08	0.50	18	0.0095	0.0390	20
111000IV.C	98.6	2711164	25	0.831	0.522	25	-0.18	3.27	11	0.0223	0.0446	30
011000	99.3	3421845	26	1.058	0.649	29	-0.17	3.27	9	0.0271	0.0480	31
001100	99.7	2957894	28	0.911	0.472	27	-1.71	1.66	32	0.0166	0.0350	27
001000	99.7	2957894	27	0.911	0.472	26	-1.71	1.66	31	0.0166	0.0350	26
101100	99.9	3207263	29	1.076	0.381	30	-1.65	1.67	29	0.0152	0.0345	24
101000	99.9	3207263	30	1.076	0.381	31	-1.65	1.67	30	0.0152	0.0345	25
111100IV.B	100.0	3389763	32	1.188	0.326	32	-0.19	3.27	14	0.0149	0.0332	23
011100	100.0	3091051	31	1.050	0.389	28	-0.19	3.27	13	0.0168	0.0332	28

Note: Pose Variant is a 6-digit binary number ABCDEF, with “1” indicating the feature is turned on (“0” means turned off), and: A—wheel speeds; B—Carrier phase double differences and Doppler shifts; C—WAAS corrections; D—FIM (hypothesis tests); E—Differential Corrections (OmniStar); F—Map aiding. Cases from the ordered sensitivity study are denoted by their subsections.

so its estimated errors grow much faster. Strictly speaking, both variants are unusable for autonomous navigation due to their large positioning errors and discontinuities. Nevertheless, the relatively slow accumulation of error in the pose variant utilizing wheel speeds makes it a promising candidate for applying map aiding and rate limiting techniques to improve its accuracy and precision [11, 12].

VI. EXHAUSTIVE SENSITIVITY ANALYSIS

An exhaustive study of the performance metrics was then undertaken, where all ordering of the removal of components was completed. Table II shows these results, ranked by the order of the % Diff metric; other rankings are also given in the table, along with the correlation to the cases in Table I. Note that the baseline case, “000000,” is IMU+GPS L1 pseudo-ranges (i.e., code tracking) with all other elements turned off.

There were several conclusions. First, the differential corrections were again clearly the most

important factor in the % Diff metric, as the top 16 cases of this metric used the corrections. There is then a large drop in the % Diff metric, from 16 to 17. The next most important component was the carrier phase (top 8) followed by the filter integrity (top 4). Analysis of the K-L and Position metrics gives similar results, as shown by their rankings. Analysis of the Attitude metrics shows that the carrier phase is the most important factor, followed by wheel speeds and then differential corrections. Analysis of the Discontinuity metric is very important for the robustness of the solution. In this case, while the differential corrections are still the most important, FIM and wheel speeds are also important.

One other interesting item is that in certain filter variants, FIM was not influential at all, while it was important to others. Cross correlations between measurements in a single epoch vary a lot depending on the variant under study. For cases with less cross correlation, especially those with no carrier phase double differences, the hypothesis tests are not as

powerful. In these cases, it is less likely that any measurement will be rejected.

VII. CONCLUSION

A statistical analysis of the design elements of a tightly-coupled pose estimator for autonomous navigation by robots in an urban environment has been performed. The estimator, designed and implemented in Cornell University's 2007 DARPA Urban Challenge robot, Skynet, fuses information from a Litton LN-200 IMU, Skynet's ABS wheel encoders, a Septentrio PolaRx2e@ three-antenna GPS receiver, and a Trimble Ag252 differential corrections GPS receiver. Information is fused in an ESRIF, a numerically robust implementation of an EKF. The pose estimator also fuses map-based information from onboard vision sensors, which are incorporated separately via particle filter [11].

Two sensitivity analyses were performed, one which focused on removing one design element at a time (Table II), and the second which was an exhaustive study of each design element (Table I). The approach for each was to empirically investigate the sensitivity of Skynet's pose estimator to each of its design elements by statistically quantifying the differences between filter variants with design elements removed, and the baseline case where all components were used. The empirical analysis used logged data taken at the DARPA Urban Challenge NQE, with the logged output of the pose estimator used as a baseline solution. The effects of five design elements were considered: map aiding via computer vision algorithms, inclusion of differential corrections, filter integrity monitoring, WAAS augmentation, and inclusion of carrier phases; the effects of extensive signal blackouts are also considered. Metrics of comparison include the statistical differences between the full solution and variant; the K-L divergence; and the average and standard deviation of the position errors, attitude errors, and filter update discontinuities.

The sensitivity results showed that the pose filter performance was most sensitive to its differential corrections for the % Diff, K-L and Position metrics; carrier phase and filter integrity were also important elements in the top designs based on these three metrics. Note that each of these metrics—% Diff, K-L, and Position—yielded similar results in terms of rankings of the filter variants (i.e., which filter component was most important). Analysis of the Attitude metrics shows that the carrier phase is the most important factor, followed by wheel speeds and then differential corrections.

Although the pose estimator is statistically most sensitive to differential corrections, it can be argued that integrity, robustness, and precision are more important than absolute accuracy for autonomous urban navigation because this signal is used to

drive the robot. With this in mind, many tested pose estimator variants yielded intolerably large discontinuities (Discontinuity) in their pose solutions. In fact, only variants including FIM were found suitable in this respect; wheel speeds also improved the Discontinuity metric.

Similar results were found in variants experiencing signal blackouts. A 19 min differential corrections blackout revealed that although the pose estimator is rather sensitive to differential corrections, it does not need to receive them continuously. This result makes 60 mi autonomous drives like the DARPA Urban Challenge possible, as differential corrections are often only received intermittently. Two pose estimator variants were further tested in a 19 min total signal blackout: one variant incorporating Skynet's wheel speed measurements, and one only integrating the IMU. Both variants were found unsuitable for autonomous navigation, though related research has suggested that the variant incorporating wheel speeds is potentially robust enough to be repaired with map aiding techniques [11].

ACKNOWLEDGMENT

The authors thank Aaron Nathan and other members of the Cornell DARPA Urban Challenge Team, for providing sensors, sensor interfaces, and other testing support. The authors also thank Peter Moran for his photography work. The authors would also like to thank Septentrio Satellite Navigation for sponsoring Team Cornell and for hardware support.

The authors would also like to thank Mark Psiaki of Cornell University's Department of Mechanical and Aerospace Engineering for helpful discussion and comments.

This material is based upon work supported under a National Science Foundation Graduate Research Fellowship. This work is also supported by the DARPA Urban Challenge program (contract HR0011-06-C-0147), with Dr. Norman Whitaker as Program Manager.

REFERENCES

- [1] Sukkarieh, S., Nebot, E., and Durrant-Whyte, H. A high integrity IMU/GPS navigation loop for autonomous land vehicle applications. *IEEE Transactions on Robotics and Automation*, **15**, 3 (June 1999), 572–578.
- [2] Psiaki, M., et al. ALEXIS spacecraft attitude reconstruction with thermal/flexible motions due to launch damage. *Journal of Guidance, Control, and Dynamics*, **20**, 5 (Sept.–Oct. 1997), 1033–1041.
- [3] Alban, S., et al. Performance analysis and architectures for INS-aided GPS tracking loops. In *Proceedings of ION National Technical Meeting* (ION-NTM), 2003, 611–622.

- [4] Artes, F. and Nastro, L.
Applanix POS LV 200: Generating continuous position accuracy in the absence of GPS reception.
Applanix Corporation, Aug. 2005, http://www.applanix.com/media/downloads/products/articles/papers/POSLV_2005_06_POSLV200.pdf.
- [5] Ohlmeyer, E., et al.
GPS-aided navigation system requirements for smart munitions and guided missiles.
In *AIAA Guidance, Navigation, and Control Conference Collection of Technical Papers*, AIAA, Reston, VA, Aug. 1997, 954–968.
- [6] Bar-Shalom, Y., Rong Li, X., and Kirubarajan, T.
Estimation with Applications to Tracking and Navigation: Theory, Algorithms and Software.
Hoboken, NJ: Wiley, 2001.
- [7] Cui, Y. and Ge, S. S.
Autonomous vehicle positioning with GPS in urban canyon environments.
IEEE Transactions on Robotics and Automation, **19**, 1 (Feb. 2003), 15–25.
- [8] El Najjar, M. and Bonnifait, P.
A road-matching method for precise vehicle localization using belief theory and Kalman filtering.
Autonomous Robots, **19**, 2 (Sept. 2005), 173–191.
- [9] George, M. and Sukkarieh, S.
Tightly coupled INS/GPS with bias estimation for UAV applications.
In *Proceedings of the 2005 Australasian Conferences on Robotics and Automation*, Dec. 2005.
- [10] Syed, S. and Cannon, E.
Map-aided GPS navigation: Linking vehicles and maps to support location-based services.
In *GPS World*, Questex Media, Inc., Duluth, MN, Nov. 2005, http://www.gpsworld.com/gpsworld/article/articleDetail.jsp?id=221180_&pageID=1&sk=&date=.
- [11] Miller, I. and Campbell, M.
Particle filtering for map-aided localization in sparse GPS environments.
In *Proceedings of the 2008 International Conference on Robotics and Automation*, May 2008, 1834–1841.
- [12] Miller, I., et al.
Cornell University's 2005 DARPA grand challenge entry.
Journal of Field Robotics, **23**, 8 (2006), 625–652.
- [13] Thrun, S., et al.
Stanley: The robot that won the DARPA grand challenge.
Journal of Field Robotics, **23**, 9 (2006), 661–692.
- [14] Miller, I., et al.
Team Cornell's Skynet: Robust perception and planning in an urban environment.
Journal of Field Robotics, **25**, 8 (Aug. 2008), 493–527.
- [15] Miller, I., et al.
Tightly-coupled GPS/INS system design for autonomous urban navigation.
In *2008 IEEE/ION Position, Location and Navigation Symposium*, 2008, 1297–1310.
- [16] Northrop Grumman Navigation Systems Division
LN-200 fiber optic inertial measurement unit, 2000.
<http://www.nsd.es.northropgrumman.com/Html/LN-200/brochures/ln200.pdf>.
- [17] Septentrio Satellite Navigation
PolarX2 User Manual, Version 2.6, June 2005.
- [18] Trimble
AgGPS 252 Receiver User Guide, Version 1.00, Revision A, Feb. 2004, http://trl.trimble.com/docshare/dsweb/Get/Document-159152/AgGPS252_100A_UserGuide_55510-00-ENG.pdf.
- [19] Bierman, G.
Factorization Methods for Discrete Sequential Estimation.
New York: Academic Press, 1977.
- [20] Psiaki, M. and Mohiuddin, S.
Modeling, analysis and simulation of GPS carrier phase for spacecraft relative navigation.
Journal of Guidance, Control, and Dynamics, **30**, 6 (Nov.–Dec. 2007), 1628–1639.
- [21] de Jonge, P. and Tiberius, C.
The LAMBDA method for integer ambiguity estimation: Implementation aspects.
Delft University of Technology, Technical Report Delft Geodetic Computing Centre LGR-Series 12, Aug. 1996.
- [22] Savage, P.
Strapdown inertial navigation integration algorithm design Part 1: Attitude algorithms.
Journal of Guidance, Control, and Dynamics, **21**, 1 (Jan.–Feb. 1998), 19–28.
- [23] Savage, P.
Strapdown inertial navigation integration algorithm design Part 2: Velocity and position algorithms.
Journal of Guidance, Control, and Dynamics, **21**, 2 (Mar.–Apr. 1998), 208–221.
- [24] Savage, P.
Errata: Strapdown inertial navigation integration algorithm design Part 2: Velocity and position algorithms.
Journal of Guidance, Control, and Dynamics, **21**, 2 (Mar.–Apr. 1998), 13.
- [25] Lemoine, F., et al.
The development of the joint NASA GSFC and NIMA geopotential model EGM96.
NASA/TP-1998-206861, July 1998.
- [26] Lundberg, J. and Schutz, B.
Recursion formulas of Legendre functions for use with nonsingular geopotential models.
Journal of Guidance, Control, and Dynamics, **11**, 21 (Jan.–Feb. 1988), 31–38.
- [27] GPS Joint Program Office
Navstar GPS space segment/navigation user interfaces, Mar. 2006, IS-GPS-200D.
- [28] U.S. Department of Transportation Federal Aviation Administration
FAA specification for the wide area augmentation system (WAAS), Aug. 2001, FAA-E-2892b.
- [29] Niell, A.
Global mapping functions for the atmosphere delay at radio wavelengths.
Journal of Geophysical Research, **101**, B2 (1996), 3227–3246.
- [30] Saastamoinen, J.
Atmospheric correction for the troposphere and stratosphere in radio ranging of satellites.
In S. Henriksen, A. Mancini, and B. Chovitz (Eds.), *Geophysical Monograph Series*, vol. 15, 1972, 247–251.
- [31] Davis, J., et al.
Geodesy by radio interferometry: Effects of atmospheric modeling errors on estimates of baseline length.
Radio Science, **20**, 6 (Nov.–Dec. 1985), 1593–1607.
- [32] Bosen, J.
A formula for approximation of the saturation vapor pressure of water.
Monthly Weather Review, **88**, 8 (Aug. 1960), 275–276.
- [33] Scherzinger, B.
Precise Robust Positioning with Inertial/GPS RTK.
Applanix Corporation, 2000, http://www.applanix.com/media/downloads/products/articles/papers/POSLV_2000_09_PreciseRobustPositioning.pdf.

- [34] Fletcher, L., et al.
The MIT-Cornell collision and why it happened.
Journal of Field Robotics, **25**, 10 (Oct. 2008).
- [35] Anderson, T.
An Introduction to Multivariate Statistical Analysis
(2nd ed.).
Hoboken, NJ: Wiley, 1984.
- [36] Kullback, S. and Leibler, R. A.
On information and sufficiency.
The Annals of Mathematical Statistics, **22**, 1 (Mar. 1951),
79–86.
- [37] Mohiuddin, S. and Psiaki, M.
Satellite relative navigation using carrier-phase differential
GPS with integer ambiguities.
In *AIAA Guidance, Navigation, and Control Conference
and Exhibit*, San Francisco, CA, Aug. 2005, Paper
AIAA-2005-6054.



Isaac Miller is working toward his Ph.D. in mechanical engineering at Cornell University. He received his M.S. in mechanical engineering from Cornell University in 2006 and his B.S. in mechanical engineering from Caltech in 2003.

His research interests include robotic perception and planning in the context of Bayesian estimation and probabilistic representation, sensor fusion and Bayesian estimation, attitude and position estimation, real-time algorithmic implementation for robotic field deployment, controls, and dynamic modeling and simulation. His past work includes design and implementation of localization and perception algorithms for Cornell University's 2007 DARPA Urban Challenge robot and 2005 DARPA Grand Challenge robot.



Mark Campbell received his B.S. in mechanical engineering from Carnegie Mellon in 1990, and M.S. and Ph.D. in control and estimation from MIT in 1993, 1996.

He is an associate professor in Mechanical and Aerospace Engineering at Cornell University. His current research interests are in estimation and control of autonomous vehicles.

Dr. Campbell received a NASA award for the MACE experiment flown on STS-67, best paper awards from the AIAA and FIE, and teaching awards from Cornell, Washington, and ASEE. He is an Associate Fellow of the AIAA, Australian Research Council International Fellow, and Associate Director of the AACC board.