Cars on the Road!

Cooperative Vehicle Safety Applications Using an Integrated Positioning Solution

In order to achieve more reliable and practical autonomous vehicle navigation, user equipment must overcome the difficulty of urban canyons, where blocked and reflected GNSS signals are familiar challenge to ensuring continuous, precise positioning. The authors present a multi-sensor fusion position solution that support performance goals of cooperative vehicle safety applications in such places. This article examines two architectures that use configurations of integrated GPS/ inertial navigation, clocks, odometers, and lidar, evaluating their performance using data collected while navigating through downtown Detroit.

Cooperative vehicle safety applications should preferably have two-meter horizontal accuracy and six-meter vertical accuracy, all with a 95-percent availability. The solution must be developed to incorporate lower-cost sensor options, specifically, lower-cost inertial measurement units that can be generally characterized by the gyro drift of 100 degrees per hour and an accelerometer bias force of twice its mass times gravity (two milligals).

Our implementation of a cooperative vehicle safety system uses a low-latency 5.9 GHz communication link among vehicles and roadside infrastructure. This enables each vehicle to continually assess the chance of a collision. If the collision probability is high, the system may generate an in-vehicle warning for the driver, or even automatically initiate actions to help prevent the collision. A vehicle equipped with this system knows its own location and path, while it wirelessly monitors the locations and paths of surrounding vehicles.

These applications rely on two main technologies: (1) information exchange using dedicated short-range communications (DSRC), and (2) location using GNSS, although various other technologies are involved.

Although GNSS satisfies the desired accuracy level in open areas where unobstructed signals are available, it fails to support the desired performance in dense urban environments. In order to achieve the set performance goals, GNSS must be augmented with other sensors.

In this article we describe a multi-sensor architecture developed to enable precise positioning capabilities in difficult GNSS environments (such as urban canyons) for cooperative vehicle safety applications. Our overall goal is to enable meter-

ANDREY SOLOVIEV
UNIVERSITY OF FLORIDA

ZELJKO POPOVIC ANDYUTAKA MOCHIZUKI
HONDA R&D AMERICAS, INC.
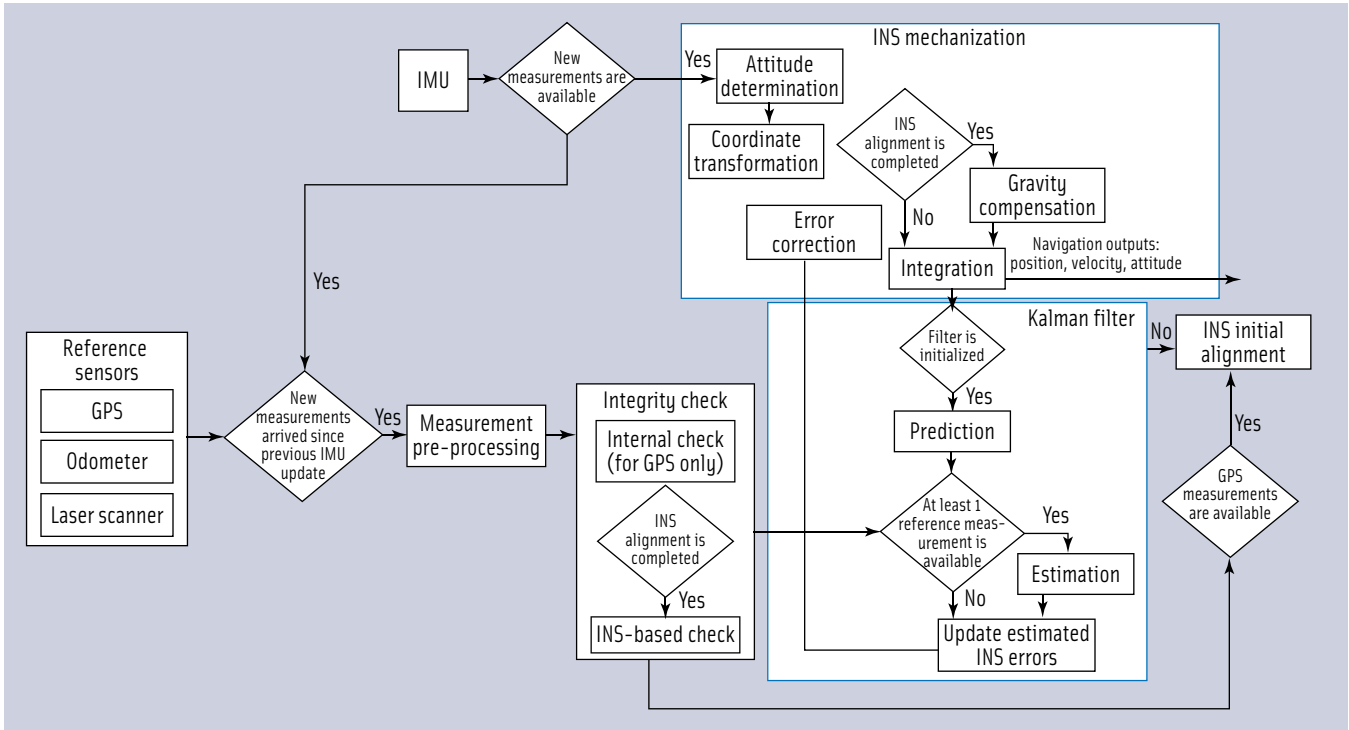
© iStockphoto.com/River North Photography

FIGURE 1 **System architecture**

accurate absolute positioning in dense urban environments at a low cost.

System Design: The Technologies

Our solution develops a generic inertial-aided approach in which aiding sources can include GNSS, odometer, scanning lidar, and video cameras. We present the system architecture, describe specific system components, and evaluate its performance with experimental data collected in actual urban test environments.

The integrated solution is based on generic a multi-sensor fusion architecture described in the article by A. Soloviev and M. Miller (please refer to Additional Resources). The architecture utilizes a self-contained inertial navigation system (INS) as a core sensor. Other navigation aids generally extract their navigation-related measurements from external information which may or may not be available. When available, these externally dependent measurements are applied to estimate inertial error states and mitigate the INS output drift.

We initially derive the navigation solution from the inertial data, estimate inertial error states from the aiding measurements, and then adjust the INS outputs.

Data fusion is performed at a tightly coupled level, where observable measurements are fed to a complementary Kalman filter, and differences between aiding measurements and predictions are computed using the inertial data. This solution enables a generic formulation where adding/dropping of an aiding source is simply achieved by adding/dropping its corresponding measurement observables to/from the Kalman filter without the need to redesign the entire system architecture.

The aiding measurements evaluated as part of this effort, include a GPS receiver, an enhanced GPS receiver that is frequency-aided by an oven-controlled crystal oscillator (OCXO) for clock stability, an odometer (ODO), a video camera, and a scanning lidar.

To support a low-cost implementation, integration algorithms apply GPS carrier phase measurements for periodic estimation of inertial error states. These algorithms utilize temporal carrier phase differences, which enable efficient inertial measurement unit (IMU) drift estimation without needing to resolve integer ambiguities. As a result, relatively low-grade micro-electro-mechanical system (MEMS) inertial sensors can be used. This reduces the cost of the inertial sensor component by at least an order of magnitude, when compared with tactical and navigation grade sensors employed in existing commercial off-the-shelf solutions.

The multi-sensor fusion solution was developed and evaluated over the following configurations: GPS/INS, GPS/INS/clock, GPS/INS/clock/ODO, GPS/INS/clock/ODO/Video, and GPS/INS/clock/ODO/Lidar. All five configurations were implemented and initially evaluated in a high-fidelity simulation environment.

Based on the simulation analysis, the top two solution options were identified as GPS/INS/clock/ODO and GPS/INS/clock/ODO/Lidar. We then tested these two further by collecting experimental data in downtown Detroit, Michigan.

System Architecture

Figure 1 illustrates the overall architecture of the multi-sensor fusion algorithm. Computations are implemented as a recur-

sive procedure that is executed every time a new measurement is provided by the inertial measurement unit (IMU). When a new IMU measurement becomes available, the system updates INS navigation outputs by propagating this new measurement through the INS navigation mechanization. This includes attitude determination, coordinate transformation, gravity compensation (implemented only if INS alignment has been completed) and, finally, integration into velocity and position navigation outputs. INS position outputs serve as the overall output of the system.

After the inertial mechanization is completed, the system runs Kalman filter updates. The filter estimates errors in inertial navigation outputs by fusing INS and reference data (including GPS, odometer and laser scanner data). The Kalman filter updates are only executed if the filter is initialized (the filter initialization happens simultaneously with the INS alignment). First, prediction is implemented. Second, estimation is performed. The estimation step is only performed if at least one reference measurement is available. Otherwise, estimated states are assigned their predicted values. Finally, estimated INS error states are applied to correct the inertial navigation parameters.

Processing of reference measurements includes a pre-processing step and integrity check. Pre-processing converts raw sensor data into the measurement observation domain that is used by the Kalman filter. An integrity check performs quality control by removing measurements such as those GPS measurements corrupted by multipath or odometer measurements corrupted by wheel slips/spins.

Integrity checks include both an internal GPS integrity check and an INS-based integrity check for all reference sensors. In the case of GPS, some of the corrupted measurements can be removed by cross-checking multiple GPS measurements via the parity-based integrity monitoring algorithm.

We performed INS-based integrity checking for all the reference sensors, by first computing what the measurement should be, measuring what the actual sensor reports, and then comparing these with the predicted measurements, while removing outliers. If at least one reference measurement is available after the integrity check, the Kalman filter uses it to compute the estimates of inertial error states.

System Components

Let's take a closer look at the key components used in our candidate systems.

Inertial Navigation. As shown in Figure 1, inertial navigation implements a standard strapdown INS mechanization routine that includes attitude determination, coordinate transformation, gravity compensation, and attitude determination steps.

Prior to INS navigation mechanization, inertial navigation states must be initialized with initial position, velocity, and attitude. Position and velocity are initialized using GPS receiver outputs estimated from temporal carrier phase changes, as summarized in the next subsection. For a more detailed description of this technique, see the article by F. van Graas and A. Soloviev cited in Additional Resources.

We determined the initial attitude by comparing components of two non-collinear vectors at the navigation and body frames, using a lower-grade IMU with the gyroscope drift rate of about 100 degrees per hour. Standard gyro-compassing procedures using gravity vector and Earth rate vector are not applicable since the Earth rate cannot be adequately measured. Instead, we aligned the IMU "in-flight" using gravity measurements and a unit vector collinear with the velocity vector.

We derived navigation frame components of the unit velocity vector from GPS carrier phase-based velocity estimates. For body-frame components, we assumed that velocity is aligned with the forward-looking axis of the vehicle's body frame.

GPS. The GPS position solution is provided directly by the GPS receiver. Additional processing of GPS measurements includes velocity estimation to initiate the INS's alignment, and an internal integrity check. GPS velocity is derived from carrier phase changes over time wherein the carrier phase from current and previous measurement epochs are used.

Carrier phase changes are first computed then adjusted for satellite Doppler terms (changes in carrier phase due to the motion of the satellite along the receiver-to-satellite line-of-sight, LOS) and geometry terms (changes in the carrier phase due to changes in the orientation of LOS vectors).

These adjusted carrier phase changes $\Delta\varphi^{adj}$ are applied for the estimation of average velocity, following the method described by in the previously mentioned article by van Graas and Soloviev:

$$\mathbf{H} \cdot \begin{bmatrix} \langle \mathbf{V} \rangle \\ \langle \delta \dot{t}_{rcvr} \rangle \end{bmatrix} \cdot \Delta t_{GPS} = \begin{bmatrix} \Delta\varphi_1^{adj} \\ ... \\ \Delta\varphi_{N_{SV}}^{adj} \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ ... \\ \varepsilon_{N_{SV}} \end{bmatrix} \quad (1)$$

$$\mathbf{H} = \begin{bmatrix} -\mathbf{e}_1 & 1 \\ ... & ... \\ -\mathbf{e}_{N_{SV}} & 1 \end{bmatrix}, \mathbf{e}_n = \frac{\mathbf{R}_{SV}^{(n)} - \mathbf{R}}{\left\| \mathbf{R}_{SV}^{(n)} - \mathbf{R} \right\|}, n = 1,..., N_{SV}$$

where

$\langle \mathbf{V} \rangle$ is the average velocity;

$\langle \delta \dot{t}_{rcvr} \rangle$ is the average clock drift;

Δt_{GPS} is the GPS update interval;

N_{SV} is the number of visible space vehicles (satellites);

ε is the carrier phase measurement error;

$\mathbf{R}_{SV}$ is the satellite position vector;

$\mathbf{R}$ is the receiver position vector;

$\|\|$ is the Euclidian vector norm; and,

$\mathbf{E}$ is the LOS unit vector.

Based on equation (1), we used a least mean square (LMS) estimation procedure to compute the average velocity, such that

$$\begin{bmatrix} \langle \mathbf{V} \rangle \\ \langle \delta \dot{t}_{rcvr} \rangle \end{bmatrix} = \frac{1}{\Delta t_{GPS}} \left(\mathbf{H}^T \cdot \mathbf{H} \right)^{-1} \cdot \mathbf{H}^T \cdot \begin{bmatrix} \Delta\varphi_1^{adj} \\ ... \\ \Delta\varphi_{N_{SV}}^{adj} \end{bmatrix} \quad (2)$$
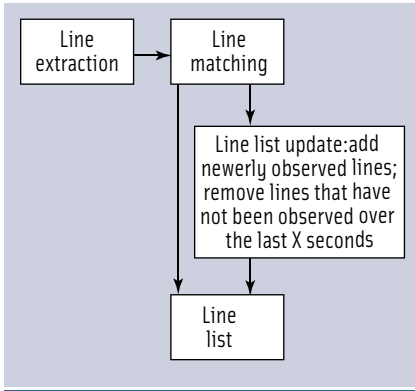
FIGURE 2 **Pre-processing of laser measurements**

(Note that average velocity and average clock drift are obtained.)

We estimated the instantaneous velocity vector that is used to initially align the INS by fitting a polynomial over the average velocities from several consecutive measurement epochs.

After computing the velocity solution, an internal integrity check is performed on the GPS measurements. The integrity check procedure identifies whether abnormally large outliers are present in carrier phase measurements. (This integrity check only detects if failures (i.e., error outliers) may be present without attempting to identify the failed satellite channels.)

If the previous step identifies the presence of failures, the GPS carrier phase measurements are further checked based on INS data (as described in a later section). Failure detection is based on a QR-factorization parity check of the LMS velocity estimation.

Odometer. An internal wheel speed sensor provides the odometer measurements. The wheel-speed counter value is periodically read and used to compute ODO-based navigation solution. Because the counter is read directly from the sensor, the odometer measurements are precisely time-stamped, which is critical for their efficient use in the sensor fusion algorithm.

Pre-processing of the odometer data converts the odometer counter increment into the units of distance used to update the odometer-based position estimate, which provides the input to the Kalman filter. The odometer position estimate is initialized with the INS position after completion of the INS alignment.

The following description outlines our odometer position solution method. First, the incremental distance S is calculated,

$$S = \Delta S \cdot (\text{Count} - \text{Count}_p) \qquad (3)$$

where ΔS is the distance increment per ODO count, and Count and Count_p are counter values for the current and previous update epochs. The distance increment is then converted into the body-frame displacement vector. For relatively straight motion, when no significant angular turns are present, a straight path approximation is applied, where

$$\Delta \widetilde{\mathbf{R}} = \begin{bmatrix} 0 & S & 0 \end{bmatrix} \qquad (4)$$

The odometer-sensitive axis is aligned with the y-axis of the body-frame. We define the coordinate system as: *x*-axis to the right of the vehicle; *y*-axis in the direction of motion, and z-axis is upward. In cases of angular turns, we use a circular motion approximation and calculate the incremental position change as follows,

$$\Delta \widetilde{\mathbf{R}} = \begin{bmatrix} \text{sign}(w_z) \cdot (R_{\text{curve}} \cos \alpha - R_{\text{curve}}) \\ R_{\text{curve}} \sin \alpha \\ 0 \end{bmatrix} \qquad (5)$$

$$R_{\text{curve}} = \frac{\|\mathbf{V}\|}{|w_z|}, \ \alpha = \frac{S}{R_{\text{curve}}}$$

where, the INS provides both the velocity vector, **V**, and angular rate vector, **w**.

Next, we transform the position change from the odometer body-frame into the navigation frame using the inertial direction cosine matrix $\widetilde{\mathbf{C}}_b^N$,

$$\Delta \widetilde{\mathbf{R}}_N = \widetilde{\mathbf{C}}_b^N \cdot \Delta \widetilde{\mathbf{R}} \qquad (6)$$

Finally, the odometer navigation solution is updated,

$$\widetilde{\mathbf{R}}_{ODO} = \widetilde{\mathbf{R}}_{ODO_p} + \Delta \widetilde{\mathbf{R}}_N \qquad (7)$$

Note that INS attitude is used to update the odometer navigation solution. As a result, inertial attitude errors become correlated with odometer error states. This is taken into account by updating corresponding cross-elements of the Kalman filter covariance matrix after every ODO navigation update.

Laser Scanner. Figure 2 illustrates the pre-processing of the laser data, which is executed if a new laser scan arrives between the current and previous IMU updates. The laser scanner sends ranging pulses and measures a distance to reflecting object(s) by measuring the time of flight.

Ranging pulses are sent in the scanner measurement angular range. Hence, each scan image is represented by the angular array and a corresponding ranging measurement for each angle.

The current system implementation uses a laser scanner with an angular range from 0 to 180 degrees and angular resolution of 1 degree. The laser is mounted on the front edge of the roof of the test vehicle with a forward orientation and scans an approximately horizontal plane.

Pre-processing of scan images includes line extraction, line matching and line list update. The procedure first extracts lines from the scan image. The extraction algorithm implements a sample-by-sample line fit. In short, this algorithm can be explained as follows. The algorithm starts by taking the first two samples in the laser scan and fitting a line though them. Next, a third sample is added and a distance from this sample to the line is computed. If this distance is below a certain threshold (which is defined by the laser measurement errors and assumed characteristics of texture of scanned objects, such as building walls), then the sample is added to the line and the line parameters are updated.

The algorithm continues till it finds a sample that does not fit into the line (that is, the distance from this sample to the line is above the threshold). At this point, a new line is started using this sample and the scan sample next to it. The line extraction algorithm continues in this manner until all the samples in the scan are processed.

The lines thus extracted are characterized by their polar parameters: *line distance*, which is the distance to the line from

the scanner, and, *line angle*, which is the angle between the x-axis of the scanner body-frame and the line's perpendicular (i.e., a line that originates from the scanner location perpendicular to the extracted line). Extracted lines are matched to the lines that were previously observed, wherein the information about the previously observed lines is stored in the line list.

Laser-based navigation uses the changes in line parameters between scans. To compute changes in line parameters between scans, we must first establish a correspondence between lines extracted from the current scan lines and lines that were previously observed and stored in the line list. The line list is initially populated at the first scan. Every time a new line is observed, it is added to the list.

This line-matching procedure applies an INS-based navigation solution to predict what ranges and angles should be expected in the current scan for all the lines from the list. (The article by A. Soloviev *et alia* listed in Additional Resources provides a detailed description of the line-extraction algorithm and line-matching procedure.)

If these predicted values for a particular line from the list can be closely matched to parameters of a line extracted from the current scan, then a match is declared and the matched line pair is applied to compute laser-based navigation observables in the Kalman filter. Newly observed lines (i.e., lines that cannot be matched to any of the previously observed lines) are added

to the line list. If a line from the list has not been observed for a certain period of time (which is currently set to 20 seconds), it is removed from the list to reduce the computational load.

When the line is added to the line list it is transformed from the laser body frame to the ENU frame using INS data:

$$\widetilde{\rho}_0^{(k+1)} = \widetilde{\rho}_n^{(m)} + \widetilde{R}_x \cdot \cos\left(\widetilde{\alpha}_n^{(m)}\right) + \widetilde{R}_y \cdot \sin\left(\widetilde{\alpha}_n^{(m)}\right) \qquad \textbf{(8)}$$

$$\widetilde{\alpha}_0^{(k+1)} = \widetilde{\alpha}_n^{(m)} + \widetilde{\psi}$$

where

ρ and α are line range and angle (subscript 0 indicates the line list and subscript n indicates the current scan);

m is the index of the line in the current scan that cannot be matched to any lines in the line list;

k is the number of lines in the list before the new line is added;

$\widetilde{R}_x$ and $\widetilde{R}_y$ are x and y components of the INS-based position solution; and,

$\widetilde{\psi}$ is the INS-based heading angle.

Note that the use of the INS solution for initializing the line introduces correlation between inertial errors and line errors. This correlation is taken into account by updating the Kalman filter covariance matrix, each time a new line is added to the list.

Integrity Check

As mentioned earlier, we apply an integrity check to remove

measurement outliers. The GPS integrity check is applied to the GPS data, and an INS-based integrity checks odometer and laser scanner data.

If the internal GPS integrity check detects a failure or an insufficient number of satellites (fewer than five), then an INS-based integrity check is also implemented. The latter integrity check predicts measurements using inertial data, compares predicted measurements with actual sensor measurements, and removes those measurements for which large discrepancies exist between predicted and measured values.

Integrity is checked for GPS position and adjusted carrier phase changes, the odometer navigation solution, and changes in ranges and angles of lines extracted from scan images. If these measurements pass integrity checks, we enter them as measurement observables in the Kalman filter used to estimate the INS drift terms.

The main principle of the INS-based integrity check is formulated by equation (9):

$$\delta u = \tilde{u}(\mathbf{R}, \mathbf{C}_b^N) - \hat{u}(\tilde{\mathbf{R}}, \tilde{\mathbf{C}}_b^N)$$

$$\begin{cases} |\delta u| < \gamma \sigma_{\delta u} \rightarrow \text{measurement is valid} \\ |\delta u| \geq \gamma \sigma_{\delta u} \rightarrow \text{measurement is faulty} \end{cases} \quad \textbf{(9)}$$

where

$\tilde{u}$ is an actual sensor measurement that is generally a function of position vector ($\mathbf{R}$) and angular orientation ($\mathbf{C}_b^N$);

$\hat{u}$ is a predicted measurement based on inertial data ($\tilde{\mathbf{R}}$ and $\tilde{\mathbf{C}}_b^N$);

$\sigma_{\delta u}$ is the standard deviation of the measurement/prediction discrepancy that is computed based on measurement error covariance (for the fault-free case) and covariances of INS prediction errors; and

γ is the integrity check scaling factor: its choice is a trade-off between the probability of false alarm and probability of missed detection; currently $\gamma=5$ is chosen.

Functions u are defined as follows:

For GPS position measurement

$$u_{GPS\_POS} = \mathbf{R} \quad \textbf{(10)}$$

In this case, the sensor measurement function is computed based on the GPS receiver position solution that is converted into the local East-North-Up (ENU) frame. INS-based estimates are calculated as

$$\hat{u}_{GPS\_POS} = \tilde{\mathbf{R}} + \mathbf{C}_b^N \cdot \mathbf{L}_{GPS} \quad \textbf{(11)}$$

where $\mathbf{L}_{GPS}$ is the lever arm vector from the IMU to GPS receiver with vector components being resolved in the IMU body frame. For *GPS adjusted carrier phase time difference*

$$u_{GPS\_CP} = \Delta\varphi^{adj} \quad \textbf{(12)}$$

Sensor measurement is derived by differencing GPS carrier phase measurements over time and then compensating them for SV Doppler and geometry changes as described by van Graas and Soloviev:

$$\tilde{u}_{GPS\_CP} = \Delta\varphi^{adj}_{GPS} \quad \textbf{(13)}$$

Note that adjusted carrier phase values are computed for all the satellites in view regardless of the fact if the number of satellite is sufficient for the GPS-based navigation or not: i.e., even if fewer than four satellites are available. The INS-based measurement estimate is calculated as follows:

$$\hat{u}_{GPS\_CP} = -\left(\mathbf{e}, \Delta\tilde{\mathbf{R}} + \left(\tilde{\mathbf{C}}_b^N(t_n) - \tilde{\mathbf{C}}_b^N(t_n)\right) \cdot \mathbf{L}_{GPS}\right) + \Delta\delta\hat{t}_{rcvr} \quad \textbf{(14)}$$

where:

(,) is the vector dot product;

$\mathbf{e}$ is the receiver-to-satellite LOS unit vector;

$\Delta\tilde{\mathbf{R}}$ is the INS-based position change between GPS updates;

t_n and t_{n-1} are the timing epochs of the current and previous GPS updates, accordingly, and,

$\Delta\delta t_{rcvr}$ is the change in the receiver clock bias between GPS updates. This change is estimated by the Kalman filter.

For the *odometer-based solution*

$$u_{ODO} = \mathbf{R}$$
$$\tilde{u}_{ODO} = \tilde{\mathbf{R}}_{ODO} \quad \textbf{(15)}$$
$$\hat{u}_{ODO} = \tilde{\mathbf{R}} + \tilde{\mathbf{C}}_b^N \cdot \mathbf{L}_{ODO}$$

where L_{ODO} is the IMU-to-odometer lever arm.

For *laser scanner line ranges*

$$u_\rho = \rho_0 - \rho_n = \cos\alpha_0 \cdot R_x + \sin\alpha_0 \cdot R_y$$
$$\tilde{u}_\rho = \tilde{\rho}_0^{(k)} - \tilde{\rho}_n^{(p)} \quad \textbf{(16)}$$
$$\hat{u}_\rho = \left([1\ 0\ 0]\cos(\tilde{\alpha}_n^{(p)}) + [0\ 1\ 0]\sin(\tilde{\alpha}_n^{(p)})\right)\left(\tilde{\mathbf{R}} + \tilde{\mathbf{C}}_b^N \cdot \mathbf{L}_{Laser}\right)$$

where $\mathbf{L}_{Laser}$ is the IMU-to-laser lever arm vector. In (16), line k from the line list is assumed to be matched to line p from the current scan.

For *laser line angles*:

$$u_\alpha = \alpha_0 - \alpha_n = \psi$$
$$\tilde{u}_\alpha = \tilde{\alpha}_0^{(k)} - \tilde{\alpha}_n^{(p)} \quad \textbf{(17)}$$
$$\hat{u}_\alpha = \hat{\psi}(\mathbf{C}_b^N)$$

where ψ is the vehicle heading angle that is estimated based on the INS direction cosine matrix.

Kalman Filter

The Kalman filter fuses reference data and inertial data to estimate inertial drift terms or, equivalently, inertial error states. At each IMU update (see Figure 1), the filter first implements a prediction step where the IMU error states are predicted based on their previous values and a time-propagation model, also referred to as the system dynamic-state model.

Next, the estimation step is carried out. If at least one reference measurement is available after the integrity check, reference measurements are fused with predicted system states (that is, predicted INS error states) to compute updated estimates of the system states.

If no reference measurements are available, state predictions simply become state estimations. Generally speaking, the estimation step uses new information that is provided by the reference measurements to update predicted system states.

The estimation step can use multiple reference source combinations, such as GPS + odometer or GPS + laser data, by weighting measurements from each reference source according to their quality, which is formally characterized by covariances of measurement errors.

Reference measurements are formulated as complementary pairs, or differences between the actual measurement and measurement value that is predicted based on inertial data, and notated as $\delta u = \tilde{u} - \hat{u}$. Formulation of the measured and estimated u functions for the Kalman filter is exactly the same as their formulation for the integrity check. Specifically, equations (10) through (17) define the complementary filter measurements for GPS (position and carrier phase), odometer, and laser scanner, respectively.

The filter states include the following:

- INS error states (21 states total)
  - Position errors (3 states)
  - Delta position errors for the error in position change between consecutive GPS update epochs (3 states)
  - Velocity error states (3 states)
  - Attitude error states (3 states)
  - Sensor bias states (3 states for gyros and 3 states for accelerometers)
  - Delta attitude error states for the error in the attitude change between consecutive GPS update epochs (3 states)
- GPS receiver clock states (3 states)
  - Receiver clock bias
  - Receiver clock drift
  - Receiver clock drift accumulated between consecutive GPS update epochs
- Odometer states (10 states)
  - Odometer position errors (3 states)
  - Odometer scale-factor (1 state)
  - Misorientation between odometer and IMU body-frames (3 states)
  - Delta position error propagation between the last update of the odometer counter and the current IMU update (3 states)
- Laser states (3+2N_{lines} states, where N_{lines} is the number of lines in the line list)
  - Misorientation between laser and IMU body-frames (3 states)
  - Line errors including range error states and angular error state (2 states per line)

Test Results

We evaluated the multi-sensor fusion solution developed with actual data collected in test drives through downtown Detroit, where streets were lined with multi-story buildings that obstructed GPS signals, often reducing visibility to less than four satellites (see **Figure 3**).

Much of the testing was performed in two-day blocks with attempts to maximize the cover of the changing satellite constellation by 10-hour test sessions shifted by 4 hours between
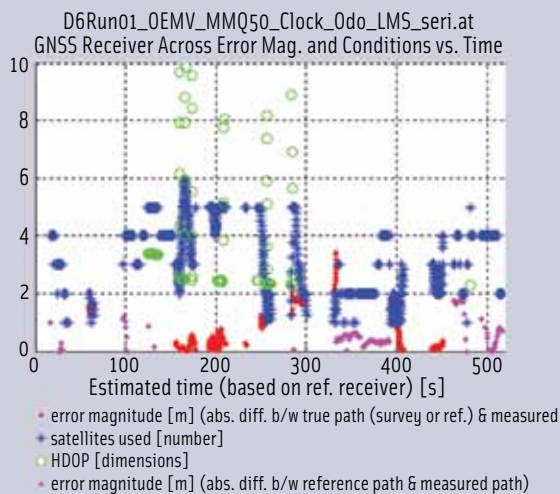

FIGURE 3 The test route (above) and an example of test drive results based on satellite visibility conditions and corresponding rrors, Day 6, Run 1. The number of satellites takes on fractional values due to interpolation.

neighboring days. The data was processed off-line by the position estimation algorithm running on commercial computational software on a PC.

The vehicle/equipment setup includes:
- A high-precision dual-frequency GPS receiver capable of receiving corrections from the Wide Area Augmentation System (WAAS)
- High-sensitivity GPS receiver
- External oven controlled crystal oscillator (OCXO) clock connected to the high-precision receiver for clock aiding
- A lower-cost IMU characterized by the gyro drift of 100 degrees per hour and accelerometer bias of two milligals
- Internal wheel speed sensor, with resolution of five pulses per rotation
- Lidar
- An integrated GNSS/inertial system coupled with a ring laser gyro for position reference when conditions allowed. **Figure 4** shows the vehicle's equipment configuration.

Measurements of all non-GPS sensors (IMUs, wheel speed, and lidar) were time-stamped using a 1 PPS (pulse per second) signal from the high-precision GPS receiver and the receiver timing message.

Figures 5 and **6** show trajectories of the GPS-only solution for the high-precision GPS receiver with WAAS corrections and the high-sensitivity GPS receiver.

The presence of large outliers for both receivers is clearly visible, demonstrating that the GNSS technology by itself is not capable of supporting the accuracy performance goals in dense urban environments.

Figures 7 and **8** show test trajectories reconstructed by the GPS/INS/clock/ODO and GPS/INS/clock/ODO/Lidar solutions.

Augmentation of GPS positioning with other sensor modalities clearly improves the solution accuracy. Both GPS/clock/ODO and GPS/INS/clock/ODO/lidar configurations

demonstrate the reconstruction of smooth trajectories. Trajectory estimation is consistent for all seven test trajectories shown with the position solution staying within the actual lane of vehicle motion for the majority of the test duration.

Figure 9 provides a quantitative characterization of the solution accuracy derived by comparing the multi-sensor fusion solution with the reference trajectory, constructed as a combination of surveyed reference points and the GNSS/inertial solution (for those portions of the trajectory where the surveyed reference was not available).

These results indicate that the GPS/INS/clock/ODO solution satisfy the accuracy goals — two meters horizontal, six meters vertical, 95 percent of the time — for a majority of the test trials. Addition of lidar data to the sensor fusion further improves the positioning accuracy while reducing the errors to the acceptable level for the limited cases of error outliers.

Conclusion

We have presented a multi-sensor fusion solution for satisfying the accuracy goals of cooperative vehicle safety applications in dense urban environments, with the system designed to operate with a lower-cost IMU. Two configurations of the system architecture — an integrated GPS/INS/clock/ODO and a GPS/INS/clock/ODO/lidar — were tested in urban environments in downtown Detroit. Results demonstrate that the fused multi-sensor solution drastically improves positioning accuracy compared to GNSS-only position estimation and met accuracy goals for the majority of test trials. Future work should focus on decreasing the cost of the system while maintaining and/or enhancing the performance goals via a trade-off calculation of the optimal sensor set based on performance-per-cost.

Acknowledgment

Manufacturers

The GPS receiver with WAAS corrections was an OEMV
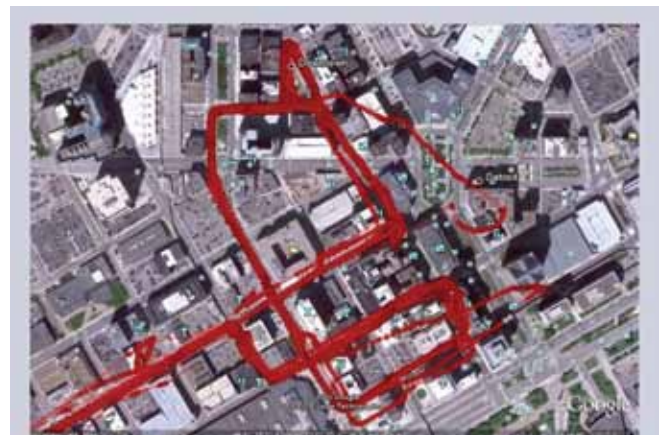
FIGURE 7 Test results for GPS/INS/clock/ODO configuration for seven test trajectories

FIGURE 8 Test results for GPS/INS/clock/ODO/Lidar configuration, for seven test trajectories

FIGURE 9 Selected performance analysis results for multi-sensor position estimation

Propak-V3-L1 from **NovAtel Inc.**, Calgary, Alberta, Canada. The integrated GNSS/inertial unit was NovAtel's SPAN-SE with an HG1700 AG58 ring laser gyro from **Honeywell Aerospace**, Phoenix, Arizona. The high-sensitivity receiver was an LEA-5T from **u-blox AG**, Thalwil, Switzerland. The lower-cost IMU was an MMQ-50, from **Systron Donner Inertial**, Concord, California, USA. The laser scanner was the LMS-200 Lidar, from **SICK AG**, Waldkirch, Germany. Matlab software from **The Mathworks, Inc.**, Natick, Massachusetts, USA, was used to process the results.

Additional Resources

[1] Farrell, J. L., "GPS/INS–Streamlined," *NAVIGATION, Journal of the Institute of Navigation*, Vol. 49, No. 4, Summer 2002

[2] Soloviev, A., and D. Bates and F. van Graas, "Tight Coupling of Laser Scanner and Inertial Measurements for a Fully Autonomous Relative Navigation Solution," *NAVIGATION, Journal of the Institute of Navigation*, Vol. 53, No. 3, 2007

[3] Soloviev, A. and M. Miller "Navigation in Difficult Environments: Multi-Sensor Fusion Techniques," NATO RTO Lecture series, Spring 2010

[4] van Graas, F, and A. Soloviev, "Precise Velocity Estimation Using a Stand-Alone GPS Receiver," *NAVIGATION, Journal of the Institute of Navigation*, Vol. 51 No. 4, 2004

Authors

Andrey Soloviev currently serves as a part-time research faculty at the University of Florida, Research and Engineering Education Facility and as president of Qunav, a small R&D business entity that he recently founded. Previously he served as a senior research engineer at the Ohio University Avionics Engineering Center. He holds B.S. and M.S. degrees in applied mathematics and physics from Moscow University of Physics and Technology and a Ph.D. in electrical engineering from Ohio University. His research interests focus on all aspects of multi-sensor integration for navigation applications. He is a recipient of the ION Early Achievement Award and the RTCA William Jackson Award.

Zeljko Popovic is a research engineer at Honda R&D Americas, Inc. in Southfield, Michigan, investigating GPS positioning performance in obstructed environments for the needs of cooperative vehicle safety. He has previously developed automotive control algorithms and software for Siemens VDO and Visteon. He earned a B.S. in mechanical engineering (mechatronics option) from the University of Waterloo and an M.S. in computer engineering from the University of Michigan-Dearborn. He has been awarded multiple patents.

Yutaka Mochizuki is a chief engineer at Honda R&D Americas, Inc. in Southfield, Michigan, investigating cooperative vehicle safety applications using dedicated short range communications (DSRC). He was the technical lead for the Crash Avoidance Metrics Partnership (CAMP) Cooperative Intersection Collision Avoidance System — Violation (CICAS-V) project. He earned a B.S. in mechanical engineering from Tohoku University in Japan. **IG**