Adaptive Kalman Filtering Methods for Low-Cost GPS/INS Localization for Autonomous Vehicles

Adam Werries, John M. Dolan

***Abstract*— For autonomous vehicles, navigation systems must be accurate enough to provide lane-level localization. High-accuracy sensors are available but not cost-effective for production use. Although prone to significant error in poor circumstances, even low-cost GPS systems are able to correct Inertial Navigation Systems to limit the effects of dead reckoning error over short periods between sufficiently accurate GPS updates. Kalman filters are a standard approach for GPS/INS integration, but require careful tuning in order to achieve quality results. This creates a motivation for a Kalman filter which is able to adapt to different sensors and circumstances on its own. Typically for adaptive filters, either the process (Q) or measurement (R) noise covariance matrix of Kalman filters is adapted, and the other is fixed to values estimated *a priori*. We show that intelligently adapting both matrices in an intelligent manner can provide a more accurate navigation solution.**

I. INTRODUCTION

For autonomous vehicles, navigation systems must be sufficiently accurate to determine the current lane on the road. High-accuracy sensors have been available for many decades now, but even today they are prohibitively expensive for automotive production. It is desirable to obtain sufficient performance from the lowest-cost sensors available through careful calibration, modeling, and filtering. The Global Positioning System (GPS) and Inertial Navigation Systems (INS) are used extensively in mobile robotics applications. The INS often consists of one or more Inertial Measurement Units (IMU), containing a gyroscope, an accelerometer, and sometimes a magnetometer. GPS and INS are complementary in terms of their respective limitations and together are capable of providing a more accurate navigation solution under sensor fusion. The purpose of our research is to explore sufficiently accurate navigation solutions such that the hardware cost would be consistent with use for current production vehicles, and computational cost would be within reason for modern embedded systems.

II. RELATED WORK

The conventional Kalman Filter (CKF) is widely used for state estimation, but is highly dependent on accurate *a priori* knowledge of the process and measurement noise covariances (Q and R), which are assumed to be constant. An autonomous vehicle experiences a dynamic range of situations which will affect each sensor to a differing degree, inspiring the idea of adaptive Kalman filtering(AKF). AKF allows for the Q (process) and/or R (measurement) noise covariance matrices to be adjusted according to the environment and dynamics [1].

A. Multiple-Model Adaptive Estimation (MMAE)

One adaptive method that has been used for integrating low-cost sensors is Multiple-Model Adaptive Estimation (MMAE), which reduces the need for an accurate *a priori* knowledge of Q and R, by using a bank of Kalman filters. Each filter has its own set of parameters for Q and R, along with a normalized weight for each filter in the bank. Over time, the weights are adjusted, eventually settling on the "best" model [2]. However, running k simultaneous filters requires k-times more computational cost [2]. In addition, requiring multiple approximations of Q and R causes performance to depend greatly the quality of these approximations and their consistent applicability to different scenarios. In order to iterate at a sufficient rate for vehicle safety on embedded systems, the computational cost is not seen as an acceptable trade-off.

B. Innovation-based Adaptive Estimation (IAE)

In order to constantly adapt to new information, Q and R should reflect the noise characteristics of a recent fixed-window. Innovation-based adaptive estimation (IAE) uses the covariance of an N-length innovation sequence to adjust the Q and R matrices. The innovation is the difference between the expected measurement state and the actual measurement. This has shown up to a 50% improvement over the CKF under certain conditions [3] [4]. Similar to IAE, residual-based filters are shown to perform even more accurately for computing R for low-cost sensors[5], instead using the difference between the predicted state and the corrected state. Under steady-state conditions, Q may be computed using the innovation sequence as well, but under dynamic situations it is required to use the state correction sequence [3].

C. Improvements

Using a CKF and INS provided with Groves' textbook [6], we have modified the filter to improve estimates for our vehicle. We modified the INS to process IMU measurements in batch, averaged over the INS integration period. We carefully measured the lever arm from the IMU to the GPS, adding its effects to the model in order to reduce errors resulting from the rotating reference frame. We then modified the filter to add adaptive estimation. Typically for adaptive filters, either R or Q is fixed, and the other is adapted [7]. In this paper, we will show that accuracy can be improved from

This work was supported by the Department of Transportation University Transportation Center at Carnegie Mellon University (CMU)

Adam Werries and John Dolan are with the Robotics Institute, CMU, Pittsburgh, PA 15213, USA awerries@andrew.cmu.edu; jdolan@andrew.cmu.edu

intelligently adapting both R and Q in a stable manner. We adapted R online in a known manner by computing the R matrix from the standard deviations (or dilution of precision) reported by the GPS receiver, while clamping the result with separate minimum and maximum values for position and velocity. This resulted in an improvement of 15% in root-mean-squared (RMS) error over raw low-cost GPS measurements in comparison to a high-accuracy reference Applanix GPS with Real Time Kinematics (RTK) mode enabled. We then show results from adapting Q online using the state correction sequence and scaling by the filter time interval, resulting in an improvement of 23% in RMS error over raw GPS measurements, a further 10% improvement over the Kalman filter with only R-adaptation. Ultimately we reduced the navigation error from 2.9m RMS to 2.3m RMS, and from 6.0m max error to 4.4m. Methods for low-cost GPS/INS integrated localization that achieve similar results require additional filter banks or tuning parameters [2] [8] [9], complex neural networks [10] [11] and fuzzy logic [12][13], or violate the requirement for real-time performance for vehicle safety by pre-filtering IMU data with wavelet decomposition methods [10].

III. APPROACH

A. Architecture

In our system, we use a Kalman filter for a loosely-coupled integration of GPS and INS. The INS is taken from Groves' textbook [6], along with the base Kalman filter which was heavily modified in order to support our timing, modeling, and adaptive requirements. In loosely-coupled integration, the GPS receiver's position and velocity solution is utilized to apply corrections to the INS. This is opposed to the more complex tightly-coupled integration, which uses the GPS's pseudo-range and pseudo-range-rate measurements in order to compute a solution [6]. Shown in figure 1, the INS maintains a running navigation solution which is used as the output of our navigation system. The Kalman filter states are the position, velocity, and attitude errors of the INS, along with estimates of the accelerometer and gyroscope biases. As each GPS solution is received, the filter iterates, using the difference between the GPS and INS solutions as a measurement to update the filter states. A closed-loop correction of the INS is then applied, and the new estimates of the biases are applied to incoming IMU measurements. The R matrix for applying GPS corrections is computed from the standard deviation reported by the GPS receiver. Q is then adapted online using a state-correction covariance matrix, as discussed in section III-F.

B. Kalman filtering

As shown below in algorithm 1, the standard Kalman filter maintains a Gaussian belief state with mean $\hat{x}$ and covariance matrix P. At each iteration, the state is propagated using a system model represented by the state transition matrix Φ and assumed-Gaussian process noise covariance matrix Q. In our case, Q describes the variation noise over a time interval of the INS error states, which are described in section III-D. The

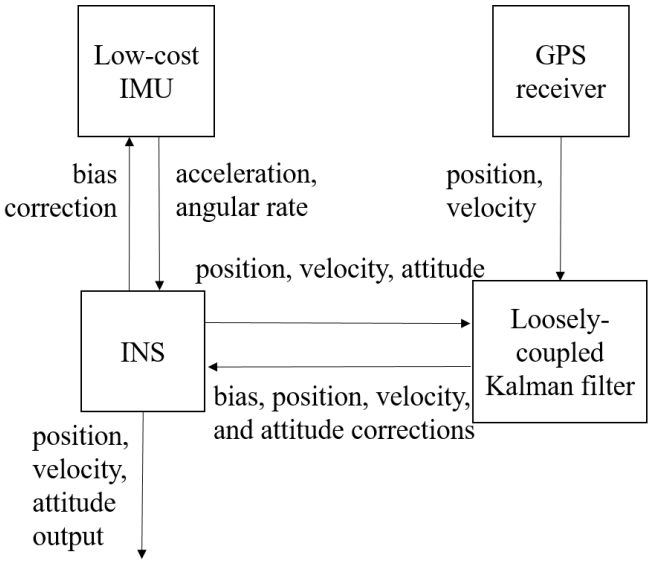

Fig. 1. Loosely-coupled integration using a GPS to apply closed-loop corrections to an INS

measurement model relating the $\hat{x}$ state to its measurements is described by H, with assumed-Gaussian measurement noise covariance matrix R. In our case, R describes the variation on the GPS measurements for position and velocity, described in section III-E. The Kalman gain K is then computed in order to optimally apply the difference between the measurement and the expected state after propagation by the process model. After the measurement vector z is received, x and P are corrected.

At the end of each iteration, we have the choice of adapting the noise matrices or leaving them as they are.

Algorithm 1 Algorithm for Kalman Filter

1: compute Φ_{k-1} and Q_{k-1}
2: $\hat{x}_k^- = \Phi_{k-1}\hat{x}_{k-1}^+$
3: $P_k^- = \Phi_{k-1}P_{k-1}^+\Phi_{k-1}^T + Q_{k-1}$
4: compute H_k and R_k with measurement model
5: $K_k = P_k^- H_k^T(H_k P_k^- H_k^T + R_k)^{-1}$
6: formulate z_k
7: $\hat{x}_k^+ = \hat{x}_k^- + K_k(z_k - H_k\hat{x}_k^-)$
8: $P_k^+ = (I - K_k H_k)P_k^-(I - K_k H_k)^T + K_k R_k K_k^T$
9: Adapt Q_k according to system performance

where:

$\hat{x}^-$	= *a priori* state vector
$\hat{x}^+$	= *a posteriori* state vector
P^-	= *a priori* state error covariance matrix
P^+	= *a posteriori* state error covariance matrix
z	= measurement vector
Φ	= process model matrix, state transition
H	= measurement model matrix
Q	= process noise covariance matrix
R	= measurement noise covariance matrix
K	= Kalman gain

The process model, measurement model, and adaptive algorithm are described in detail in the following sections.

C. INS mechanization

Following position/velocity initialization from the first GPS fix and attitude initialization from stationary leveling/gyrocompassing [6], the INS begins processing gyroscope (ω_{ib}^b) and accelerometer (f_{ib}^b) measurements received from the IMU. As shown in figure 2, each measurement increments the solution as an integral over the time interval, correcting for gravity, the rotating reference frame, and the current estimate of the biases in the sensors.

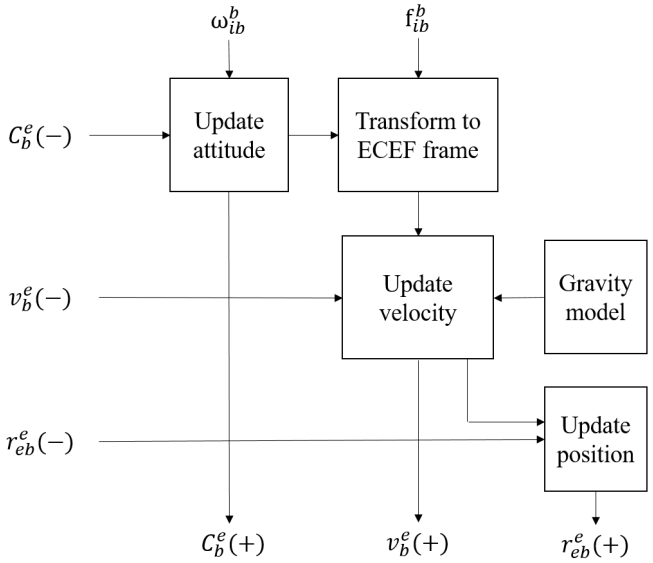

Fig. 2. Earth-centered Earth-fixed frame Inertial Navigation System [6]

The INS internal state is as follows, where each state is post-corrected by the KF,

- ω_{ib}^b = angular rate measurement from gyroscope
- f_{ib}^b = specific force measurement from accelerometer
- C_b^e = INS attitude solution, frame transformation matrix from body frame to Earth-centered Earth-fixed (ECEF) frame
- v_b^e = INS velocity solution of the body in the ECEF frame
- p_b^e = INS position solution of the body in the ECEF frame

D. Process model

The process model is responsible for propagation according to the expected value for each state. The state vector for our system consists of monitoring the attitude ($\delta\psi$), velocity (δv), and position (δr) errors shown in figure 3, along with the accelerometer and gyroscope biases (b_a and b_g),

$$x = \begin{pmatrix} \delta\psi_{eb}^e \\ \delta v_{eb}^e \\ \delta r_{eb}^e \\ b_a \\ b_g \end{pmatrix} \tag{1}$$

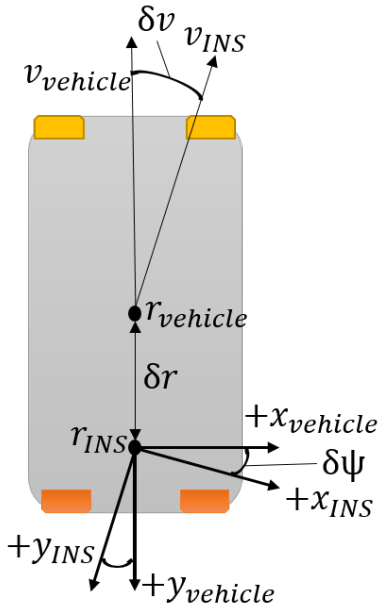

Fig. 3. Diagram of vehicle showing coordinate axes and pose error variables used in process model

where

- $\delta\psi_{eb}^e$ = INS solution attitude error
- δv_{eb}^e = INS solution velocity error
- δr_{eb}^e = INS solution position error
- b_a = Accelerometer bias
- b_g = Gyroscope bias

Thus the state transition matrix is [6],

$$\Phi = \begin{bmatrix} F_{11}^e & 0_3 & 0_3 & 0_3 & \hat{C}_b^e\tau_s \\ F_{21}^e & F_{22}^e & F_{23}^e & \hat{C}_b^e\tau_s & 0_3 \\ 0_3 & I_3\tau_s & I_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & I_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & I_3 \end{bmatrix} \tag{2}$$

where,

$$F_{11}^e = I_3 - \Omega_{ie}^e\tau_s \tag{3a}$$

$$F_{21}^e = \left[-\left(\hat{C}_b^e\hat{f}_{ib}^b\right)\wedge\right]\tau_s \tag{3b}$$

$$F_{22}^e = I_3 - 2\Omega_{ie}^e\tau_s \tag{3c}$$

$$F_{23}^e = -\frac{2\hat{\gamma}_{ib}^e}{r_{eS}^e(\hat{L}_b)}\frac{(\hat{r}_{eb}^e)^T}{|\hat{r}_{eb}^e|}\tau_s \tag{3d}$$

and

- Ω_{ie}^e = Earth rotation rate
- $\hat{\gamma}_{ib}^e$ = Gravity model value at current location
- $\hat{L}_b$ = Current latitude
- τ_s = Filter iteration time difference (epoch)
- $\wedge$ = Indicates skew-symmetric matrix of previous vector

The process noise covariance matrix Q is first estimated according to *a priori* testing values, and then adapted as described in section III-F.

E. Measurement model

The measurement model is responsible for relating measurements to states. The measurement for our system consists of the difference between the GPS and INS navigation solutions, as in the measurement vector in (4), [6].

$$\delta z_k^{e-} = \begin{pmatrix} \hat{r}^e_{eaG} - \hat{r}^e_{eb} - \hat{C}^e_b l^b_{ba} \\ \hat{v}^e_{eaG} - \hat{v}^e_{eb} - \hat{C}^e_b(\hat{\omega}^b_{ib} \wedge l^b_{ba}) + \Omega^e_{ie}\hat{C}^e_b l^b_{ba} \end{pmatrix} \quad (4)$$

To apply this measurement, we need to formulate H_k as follows [6],

$$H^e_{G,k} = \begin{bmatrix} H^e_{r1} & 0_3 & -I_3 & 0_3 & 0_3 \\ H^e_{v1} & -I_3 & 0_3 & 0_3 & H^e_{v5} \end{bmatrix} \quad (5a)$$

$$H^e_{r1} = \left[(\hat{C}^e_b l^b_{ba})\wedge\right] \quad (5b)$$

$$H^e_{v1} = \left[\{\hat{C}^e_b(\hat{\omega}^b_{ib} \wedge l^b_{ba}) - \Omega^e_{ie}\hat{C}^e_b l^b_{ba}\}\wedge\right] \quad (5c)$$

$$H^e_{v5} = \hat{C}^e_b \left[l^b_{ba}\wedge\right] \quad (5d)$$

where,

- z_k^{e-} = Measurement vector, difference between GPS and INS solutions, accounting for the lever arm difference, ECEF frame
- l^b_{ba} = Lever arm from IMU to GPS in body frame
- $\hat{r}^e_{eaG}$ = GPS-reported position vector, ECEF frame
- $\hat{v}^e_{eaG}$ = GPS-reported velocity vector, ECEF frame

The measurement noise covariance matrix R is computed according to the per-axis ECEF-frame position and velocity standard deviations reported by the GPS user-equipment. This is then scaled by the Kalman filter iteration interval. Minimum and maximum variance values were enforced in order to ensure stability.

F. State-correction sequence adaptation of process matrix Q

The process noise matrix adaptation is formulated as follows [3], where N is the number of recent state-correction sequences used in the computation,

$$\Delta x_k = \hat{x}^+_k - \hat{x}^-_k \quad (6a)$$

$$\hat{Q}_k = \frac{1}{N}\sum_{j=k-N}^{k} \Delta x_j \Delta x_j^T + P^+_k - \Phi P^+_{k-1}\Phi^T \quad (6b)$$

IV. TESTING PROCEDURE

Testing was performed on public roads through a large park on a hill near the Carnegie Mellon University campus using our Cadillac SRX vehicle, where the path is shown in figure 4. A high-accuracy RTK-enabled Applanix navigation module was used to provide an accurate-as-possible ground-truth. This specific location allows for high-quality GPS results, and consistently provides the Applanix module with Fixed or Float RTK, supplying a solution accurate to within 5 (fixed) to 30 (float) centimeters. Time was synced using the Network Time Protocol between the vehicle's Linux machines and a Raspberry Pi 2 (RPi-2) used for low-cost data collection. Time difference between the two is estimated at less than 10ms.

A low-cost IMU (InvenSense MPU-6050) was affixed to the RPi-2, which was then mounted in the vehicle. This specific IMU is ideal because the accelerometer and gyroscope are within the same QFN package, minimizing cross-coupling errors and relaxing the calibration requirements. This sensor was about $5 on Digikey at the time of writing.

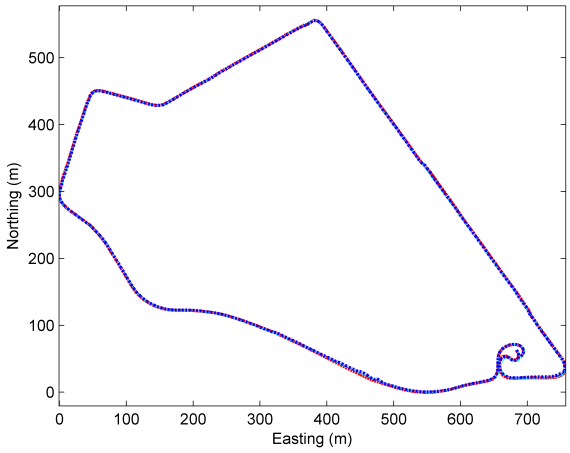

Fig. 4. Top-down view of vehicle path

A low-cost GPS, the Novatel Flex6, was also connected to the same RPi-2. The GPS antenna was then mounted on the roof of the vehicle. This GPS is ideal for our testing because it provides a full ECEF-frame navigation solution with position, velocity, and standard deviations. This GPS is available by quoted price from Novatel, but the price is consistent with use in a production vehicle. The manufacturer claims on the product page that with L1/L2 signals, error can reach as low as 1.2 meters, and 0.6m with SBAS [14].

The lever arm between the IMU and GPS mounting locations was carefully measured, although some error is unavoidable without a full 3D model of the vehicle. It is estimated that the error is less than 5 centimeters for each axis, which has less than a 1% effect on the performance.

The data were then run through the filter offline in MATLAB simulation, although the intention is ultimately to transition to a filter in C++ running online on the RPi-2.

V. RESULTS

The U.S. Department of Transportation has a recommendation of at least 2.7 meters for local urban roads [15]. This places a bound on the required performance: error and uncertainty must be no greater than half of that width, 1.35m, in order to achieve lane-level localization. The results of testing with the IMU and GPS described in the previous section are seen in table (I). Results are shown for both the conventional Kalman filter, where we only adapt R based on the GPS-reported standard deviation, and our fully adaptive Kalman filter, which also uses the state-correction sequence to estimate Q online. The window size chosen for Q was 60, as it allowed the system to quickly respond to changing noise characteristics.

The filter error compared to the high-quality Applanix solution is shown in figure 5, split into Northing, Easting,

Source	RMS	RMS Red.	Max	Max Red.
2D GPS	1.9345	-	4.3471	-
2D CKF	1.8012	6.89%	4.2962	1.17%
2D AKF	1.7544	9.31%	4.0770	6.21%
3D GPS	2.9424	-	6.0439	-
3D CKF	2.5014	14.99%	4.8136	20.36%
3D AKF	2.2700	22.85%	4.4184	26.89%

TABLE I

FILTER ERROR RESULTS IN METERS

and Altitude (Universal Tranverse Mercator coordinates for clarity). Of special note is the region around 200 seconds. The altitude graph clearly shows that the GPS has provided a poor solution, and the filter is able to continue utilizing the information provided by the GPS, while not being too aggressive with corrections due to the increase in the R matrix.

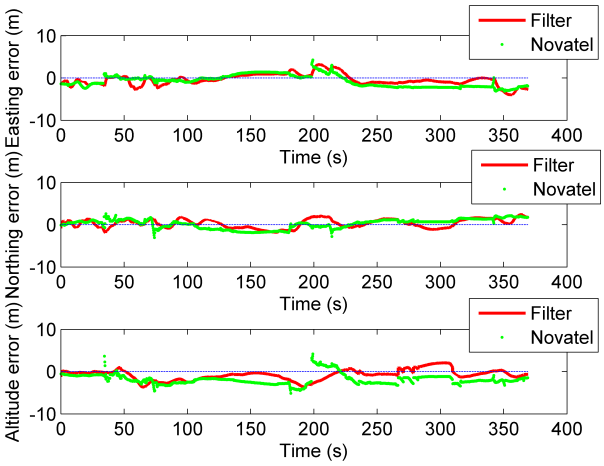

Fig. 5. Error between ground-truth vs low-cost GPS and ground-truth vs filter solution

R is shown over time in figure 6, showing how it changes based on the quality. Also seen is the minimum/maximum value clamping that was applied in order to ensure filter stability. Q is shown over time in figure 7, showing extremely high noise in the beginning of the filter sequence, settling down after about 100 seconds after the accelerometer and gyroscope properties have converged. Most important, the adaptation of Q is shown to have a large effect on stabilizing the uncertainty in the INS state error and bias, as shown in figures 8 and 9.

One possible issue shown is that the uncertainty later in the dataset appears to reduce below the value of the errors in the system! This shows that the filter is too confident in its estimate, and that further tuning of initial parameters, minimum/maxium covariance values, and adaptation methods are required. Again, the GPS's expected error is 1.2 meters, and it is unlikely that a GPS/INS system will be able to reduce its RMS error below that without the assistance of additional sensors. The largest improvements can be made in reducing

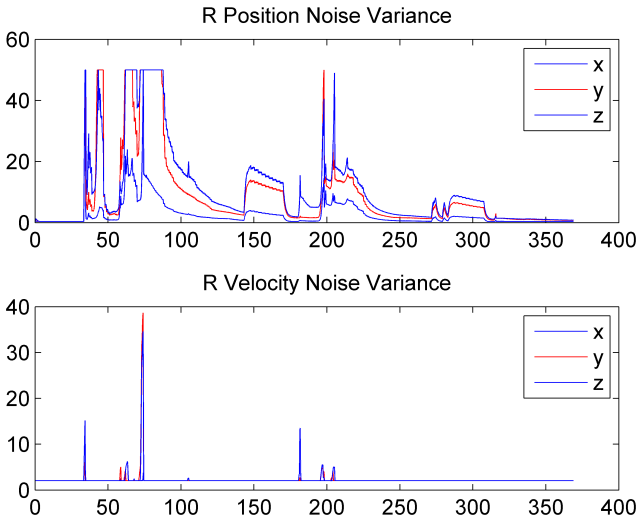

Fig. 6. R measurement noise covariance matrix (GPS) over the course of a dataset

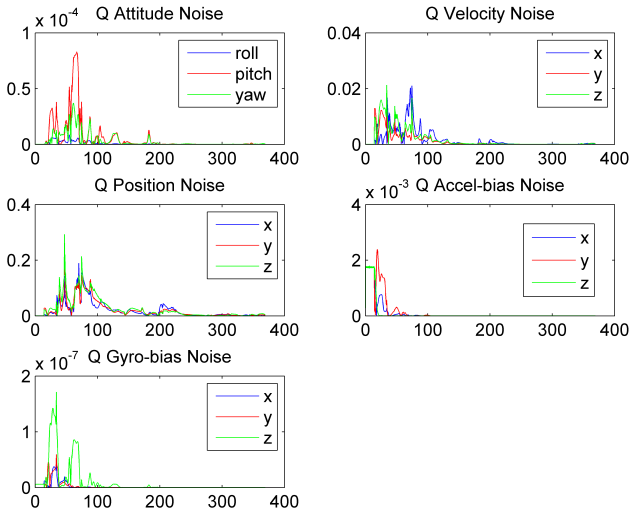

Fig. 7. Q process noise covariance matrix (INS error) adapting over the course of a dataset

maximum errors, where even the GPS receiver itself is aware that the solution accuracy is poor, which can be represented in R, allowing the INS to take over. Thus, it is apparent that improving GPS accuracy will improve RMS-error accuracy, while improving INS accuracy will improve maximum-error accuracy.

In the case of low-cost sensors, the accuracy of the gyroscope is far too poor to achieve gyrocompassing in order to reduce the yaw error for more than initialization efforts. One mistake made is that this dataset does not include a proper dynamic initialization period, where a vehicle should induce rotations in alternating opposite directions at sufficient speed to determine gyroscope and accelerometer errors before attempting to use the localization solution. In

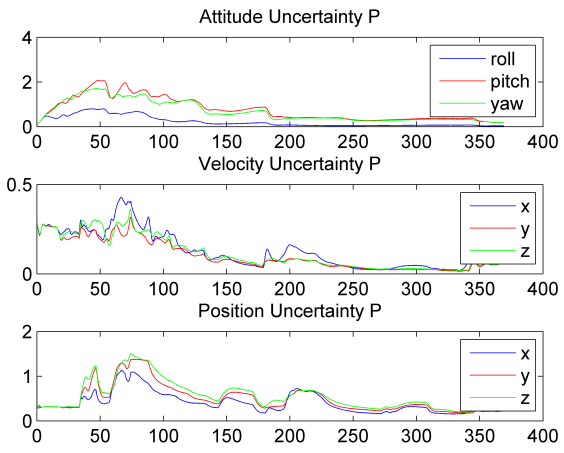

Fig. 8. Standard deviation of the INS navigation solution error, part of covariance matrix Ps

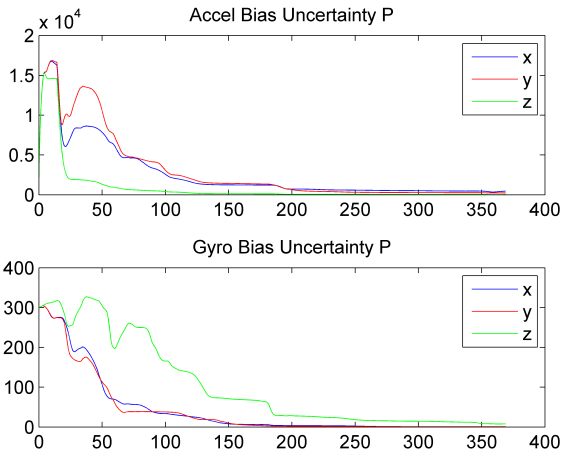

Fig. 9. Standard deviation of the accelerometer and gyro biases, part of covariance matrix P

the case of large angle errors, the small-angle approximations included in the model are violated, and it is likely that we will have to account for this in future research [6].

Conclusion

Adapting R and Q according the system noise characteristics has shown a marked improvement in localization accuracy for integrating low-cost GPS/INS systems. However, the quality has not improved to the point where lane-level localization can be achieved, and further research is required if we wish to use such inexpensive sensors. As always, calibration can be done more carefully, the filter can be tuned more accurately, and vibration effects can be analyzed and reduced through mechanical means. Adaptive methods such as process noise scaling [7] and reinforcement learning for parameter estimation [16] may further improve results.

Acknowledgment

A special acknowledgment to Paul D. Groves, whose clear and thorough textbook made this work possible [6].

References

[1] R. Mehra, "Approaches to adaptive filtering," vol. 17, no. 5, Oct 1972, pp. 693–698.

[2] C. Hide, T. Moore, and M. Smith, "Adaptive kalman filtering for low-cost ins/gps," *The Journal of Navigation*, vol. 56, no. 1, pp. 143–152, 01 2003. [Online]. Available: http://search.proquest.com/docview/229556669?accountid=9902

[3] A. H. Mohamed and K. P. Schwarz, "Adaptive kalman filtering for ins/gps," *Journal of Geodesy*, vol. 73, no. 4, pp. 193–203, 1999. [Online]. Available: http://dx.doi.org/10.1007/s001900050236

[4] A. Fakharian, T. Gustafsson, and M. Mehrfam, "Adaptive kalman filtering based navigation: An imu/gps integration approach," in *Networking, Sensing and Control (ICNSC), 2011 IEEE International Conference on*, April 2011, pp. 181–185.

[5] C. Hide, T. Moore, and M. Smith, "Adaptive kalman filtering algorithms for integrating gps and low cost ins," in *Position Location and Navigation Symposium, 2004. PLANS 2004*, April 2004, pp. 227–233.

[6] P. D. Groves, *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*, 2nd ed. Artech House, Inc., 2013.

[7] W. Ding, J. Wang, C. Rizos, and D. Kinlyside, "Improving adaptive kalman estimation in gps/ins integration," *The Journal of Navigation*, vol. 60, no. 3, pp. 517–529, 09 2007. [Online]. Available: http://search.proquest.com/docview/229564577?accountid=9902

[8] S. Y. Cho, "Im-filter for ins/gps-integrated navigation system containing low-cost gyros," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 4, pp. 2619–2629, October 2014.

[9] R. Toledo-Moreo, M. A. Zamora-Izquierdo, B. Ubeda-Minarro, and A. F. Gomez-Skarmeta, "High-integrity imm-ekf-based road vehicle navigation with low-cost gps/sbas/ins," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 3, pp. 491–511, Sept 2007.

[10] A. Noureldin, T. B. Karamat, M. D. Eberts, and A. El-Shafie, "Performance enhancement of mems-based ins/gps integration for low-cost navigation applications," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 3, pp. 1077–1096, March 2009.

[11] K. Saadeddin, M. F. Abdel-Hafez, M. A. Jaradat, and M. A. Jarrah, "Optimization of intelligent approach for low-cost ins/gps navigation system," *Journal of Intelligent & Robotic Systems*, vol. 73, no. 1, pp. 325–348, 2013. [Online]. Available: http://dx.doi.org/10.1007/s10846-013-9943-2

[12] X. Zhao, Y. Qian, M. Zhang, J. Niu, and Y. Kou, "An improved adaptive kalman filtering algorithm for advanced robot navigation system based on gps/ins," in *Mechatronics and Automation (ICMA), 2011 International Conference on*, Aug 2011, pp. 1039–1044.

[13] E. Shi, "An improved real-time adaptive kalman filter for low-cost integrated gps/ins navigation," in *Measurement, Information and Control (MIC), 2012 International Conference on*, vol. 2, May 2012, pp. 1093–1098.

[14] Flexpak6 triple-frequency + l band gnss receiver. Accessed: 2016-02-29. [Online]. Available: http://www.novatel.com/products/gnss-receivers/enclosures/flexpak6/

[15] Mitigation strategies for design exceptions: Lane width. Accessed: 2016-02-29. [Online]. Available: http://safety.fhwa.dot.gov/geometric/pubs/mitigationstrategies/chapter3/3_lanewidth.cfm

[16] C. Goodall, X. Niu, and N. El-Sheimy, "Intelligent tuning of a kalman filter for ins/gps navigation applications," 09 2007, pp. 2121–2128. [Online]. Available: https://www.ion.org/publications/abstract.cfm?articleID=7473