Improvements in Terrain-Based Road Vehicle Localization By Initializing an Unscented Kalman Filter Using Particle Filters

Adam J. Dean, Jack W. Langelaan, and Sean N. Brennan*

Abstract— **This work develops an algorithm to initialize an Unscented Kalman Filter using a Particle Filter for applications with initial non-Gaussian probability density functions. The method is applied to estimating the position of a road vehicle along a one-mile test track and 7 kilometer span of a highway using terrain-based localization where the pitch response of the vehicle is compared to a pre-measured pitch map of each roadway. The results indicate that the method can be used to decrease the computational load of the algorithm while maintaining the accuracy of the Particle Filter, but that the challenge is to determine the appropriate moment to perform the switch between algorithms. A modified Chi-Squared test is used to determine a switchover point when the probability density function of the particle population can be approximated by a Gaussian for initializing the Unscented Kalman Filter. A normalized innovation squared test is also demonstrated to be useful for monitoring the health of the Unscented Kalman Filter.**

I. Introduction

With the goal of increasing the safety and efficiency of road vehicles, there is a great deal of interest in determining vehicle position. The primary means of localization is the Global Positioning System (GPS); however, multi-path errors, satellite obstruction, the ease of jamming, slow update rates, and other problems have increased the interest of developing an algorithm to localize a vehicle independently of GPS.

Previous work has shown that matching in-vehicle pitch measurements with a terrain map can be used to estimate a vehicle's longitudinal position with sub-meter accuracy and independently of GPS [1]. Experiments using a Particle Filter (PF) algorithm was used to localize a vehicle along the one-mile test track at the Thomas D. Larson Transportation Institute (LTI). Further experiments in [2] applied the PF algorithm to localize a vehicle along a 60 km highway with an accuracy of 5 meters.

The previous work in [1] and [2] used a PF algorithm to localize a vehicle because PFs are easily initialized, requiring no a priori position estimate; however, after the estimates are initialized, there is a severe computational cost of maintaining the vehicle position using a PF with thousands of position particles. The objective of this work is to reduce the computational cost of the previous terrain-based localization algorithm.

In order to localize a vehicle along a terrain map an estimator can be used that compares a map-predicted vehicle pitch measurement to the actual pitch measurement. The vehicle position is thereafter determined by propagating regions of highest map-sensor correlation using a motion model and wheel odometry. The motion model is one-dimensional and linear; however, the initial position probability is uniform along the map; hence a Kalman Filter (KF) cannot be used as it requires a Gaussian probability. Instead, a PF was used in previous work because the initial population of position estimates is scattered randomly across the terrain map. In application, and upon initialization, the algorithm will likely have a position estimate which will only accelerate convergence; however, this work assumes a uniform position probability in order to limit the number of assumptions.

The Extended Kalman Filter (EKF) could also be used instead of a KF, except it requires a model that is a continuously differentiable function of the states; thus, an EFK is not suitable because this work uses a map of discrete points and not a continuous function. To develop several continuous functions to represent a roadway network could be just as complex as using the PF alone.

An alternative approach is to use an Unscented Kalman Filter (UKF) [3]. The UKF is used in many estimation problems when a KF is unfit due to non-linear discrete-time dynamic equations, and has been shown to be more accurate than the EKF in vehicle positioning with non-linear models [4]. The UKF, which can be considered a special case of a PF, requires relatively little computational effort because it uses only a few particles called sigma points that are placed at pre-determined distances from the current position estimate, instead of thousands of particles required to use a PF. Pitch measurements at the sigma points are calculated corresponding to their position along the terrain map, thus linearization or continuous functions are not required. However, the UKF must be initialized using a Gaussian position estimate which in this work is assumed to be unavailable at algorithm start. The tradeoffs between computational and initialization complexity of the PF and UKF algorithms are summarized in Fig. 1.

The ideal case would be to somehow switch between each algorithm to use each when most appropriate. The intent of this study is to examine the implementation of such an algorithm; the terrain-based localization application

* Corresponding author

A. Dean is with the Department of Mechanical Engineering, Brigham Young University-Idaho, 525 S. Center St., Austin 164, Rexburg, ID 83460, deana@byui.edu, (208) 496-3833

S. Brennan is with the Department of Mechanical Engineering, Penn State University, 318 Leonhard Building, University Park, PA 16802, sbrennan@psu.edu, (814) 863-2430

J. Langelaan is with the Department of Aerospace Engineering, Penn State University, 229 Hammond Building, University Park, PA 16802, jlangelaan@psu.edu, (814) 863-6817

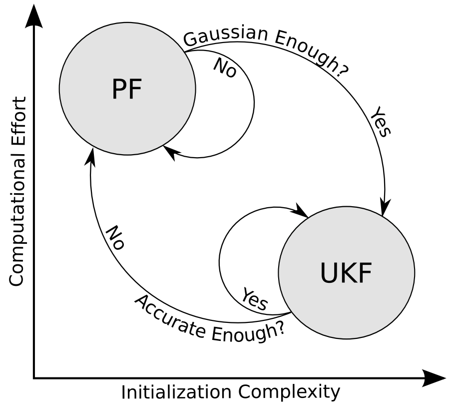

Fig. 1: Tradeoffs of the PF and UKF algorithms and criteria for switching between them.

will be used to demonstrate the algorithm and compare results. The goal is to use the PF when the estimate is being initialized or is highly uncertain, and use the UKF only after the PF algorithm has converged to essentially a Guassian distribution. A modified Chi-Squared (χ^2) test, which the authors call an Upsilon-Squared (Υ^2) test, is used to determine the “Gaussian-ness” of the PFs vehicle position estimate and use it to initialize the UKF. An innovation test is also used to monitor the UKF estimate and, when the position estimate is considered inaccurate, the algorithm reverts back to the PF by re-initializing the particle population across the terrain map.

This paper is outlined as follows: Section II presents the experimental data used for this study. Section III discusses the UKF algorithm. Section IV demonstrates the challenges with using the χ^2 test for switching from the PF to the UKF and develops an alternative method used for switching from the PF to the UKF. Section V optimizes the switchover criterion in terms of accuracy and computational effort. Section VI applies and discusses the results of the localization algorithm. Section VII implements a method of monitoring the health of the UKF estimate and Conclusions summarize the main results of this study.

II. Data Acquisition

The algorithms used in this and prior work were implemented off-line using data previously recorded using an instrumented vehicle equipped with a NovAtel “SPAN” Differential Global Positioning System (DGPS) that is factory-integrated with a Honeywell HG1700 ring-laser gyro Inertial Measurement Unit (IMU) with positioning accuracy of 2 cm (one sigma) and attitude accuracies of 0.013 for pitch and roll and 0.04 degrees for yaw.

A terrain map was generated by recording position and attitude data over the entire Thomas D. Larson Transportation Institute (LTI) test track with the vehicle driving at 5 m/s. This resulted in a map resolution of 5 cm. A second set of data was recorded at about 10 m/s to be used as the in-vehicle pitch measurements in the off-line localization algorithm. Fig. 2 shows the overhead view of the mapped area with the vehicle’s starting position, and the pitch map and measured pitch data sets. Variations in pitch between the data sets are visible and are likely due to differences in speed and inexact path tracking. The pitch data were filtered using a low-pass filter at a cutoff frequency of 0.1 cycles/meter as discussed in [1].

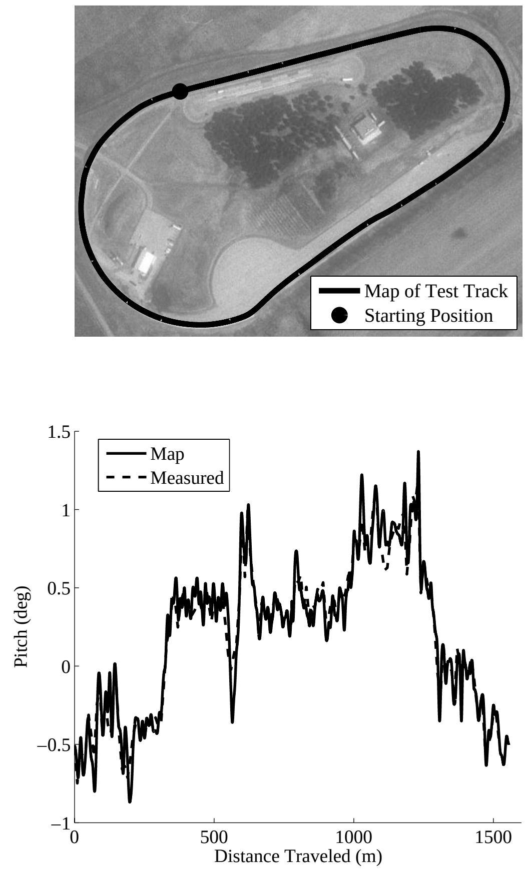

Fig. 2: Overhead view and in-vehicle pitch data collected around the LTI test track.

Highway data was also collected while traveling an average 27 m/s along highway 322 in State College, PA. The overhead map and corresponding pitch map are shown in Fig. 3.

III. Initializing the Unscented Kalman Filter

In order to decrease the computational load of the estimator, a UKF will be initialized from the PF estimate. Similar to the work done in [5] and [6] where PFs are used to initialize landmark position estimates, this work will initialize the vehicle position estimate using a PF until the probability distribution is “Gaussian enough” for use with an UKF. Once the pdf is considered to be Gaussian, the UKF is initialized using the PF results where the estimated vehicle position ($\hat{x}$) and variance (P) are calculated from the average position and variance of the PF particle population. The details of the PF algorithm are described in [1] but omitted here for brevity.

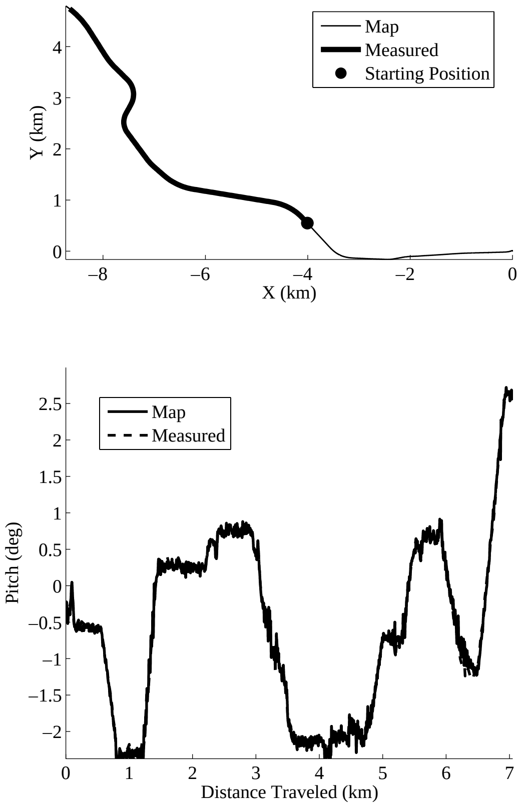

Fig. 3: Overhead view of the terrain map and in-vehicle pitch data collected along highway 322 in Pennsylvania.

Once initialized, the UKF algorithm is implemented following Algorithm 3.1 in [3] with $\alpha = 1$, $\beta = 2$, $\kappa = 0$, and $N_s = 3$ sigma points by repeating the following:

First, the location of the sigma points, X, are calculated about the estimated vehicle position $\hat{x}$:

$$\begin{aligned} X_{k-1} &= [\hat{x}_{k-1},\ \hat{x}_{k-1} + C,\ \hat{x}_{k-1} - C] &(1)\\ C &= \alpha\sqrt{N_s \cdot P_{k-1}} &(2) \end{aligned}$$

Second, the time update uses the motion model to advance the sigma points, measure their corresponding pitch, Y, according to their position along the terrain map, and estimate the expected pitch measurement, $\hat{y}$, using:

$$\begin{aligned} X_k^- &= X_{k-1} + dx_{k-1} &(3)\\ \hat{x}_k^- &= X_{k-1}^- \cdot W_m &(4)\\ P_k^- &= \left(X_{k-1}^- - \left[\hat{x}_{k-1}^-, \hat{x}_{k-1}^-, \hat{x}_{k-1}^-\right]\right) \cdot W_c \\ &\quad \cdot \left(X_{k-1}^- - \left[\hat{x}_{k-1}^-, \hat{x}_{k-1}^-, \hat{x}_{k-1}^-\right]\right)^T + Q &(5)\\ Y_k^- &= f_{NL}\left(X_{k-1}^-\right) &(6)\\ \hat{y}_k^- &= Y_{k-1}^- \cdot W_m &(7) \end{aligned}$$

where the superscript "-" (e.g. X^-) denotes an a priori value or value calculated prior to the measurement update, k is the sample time, dx is the distance the vehicle travels between time steps as inferred from odometry, Q is the variance in the odometry measurement, f_{NL} is the look-up table used to determine the pitch of each sigma point corresponding to their position along the terrain map, and

$$W_m = \left[\frac{\alpha^2 - 1}{\alpha},\ \frac{1}{2\alpha N_s},\ \frac{1}{2\alpha N_s}\right] \tag{8}$$

$$W_c = \begin{bmatrix} \frac{\alpha^2-1}{\alpha} + 1 - \alpha^2 + \beta & 0 & 0 \\ 0 & \frac{1}{2\alpha N_s} & 0 \\ 0 & 0 & \frac{1}{2\alpha N_s} \end{bmatrix} \tag{9}$$

Third, the measurement model is implemented in the measurement update:

$$\begin{aligned} P_{yy,k} &= \left(Y_{k-1}^- - \left[\hat{y}_{k-1}^-, \hat{y}_{k-1}^-, \hat{y}_{k-1}^-\right]\right) \cdot W_c \\ &\quad \cdot \left(Y_{k-1}^- - \left[\hat{y}_{k-1}^-, \hat{y}_{k-1}^-, \hat{y}_{k-1}^-\right]\right)^T + R &(10)\\ P_{xy,k} &= \left(X_{k-1}^- - \left[\hat{x}_{k-1}^-, \hat{x}_{k-1}^-, \hat{x}_{k-1}^-\right]\right) \cdot W_c \\ &\quad \cdot \left(Y_{k-1}^- - \left[\hat{y}_{k-1}^-, \hat{y}_{k-1}^-, \hat{y}_{k-1}^-\right]\right)^T &(11)\\ K &= P_{xy,k} \cdot P_{yy,k}^{-1} &(12)\\ \hat{x}_k &= \hat{x}_k^- + K \cdot \left(\theta_{a,k} - \hat{y}_k^-\right) &(13)\\ P_k &= P_k^- - K \cdot P_{yy,k} \cdot K^T &(14) \end{aligned}$$

where $\theta_{a,k}$ is the measured in-vehicle pitch and R is the measurement noise variance on pitch.

Before implementing the UKF, a method is needed to determine when the PF estimate is "Gaussian enough" to be used to initialize the UKF.

IV. Modified Chi-Squared Test

In order to determine the "Gaussian-ness" of the distribution, this study will first follow the work presented in [5] which used a χ^2 test to determine how well the distribution of particles fit a Gaussian distribution of the same mean and variance. Once the χ^2 value was reduced below a threshold, then the distribution was assumed to be Gaussian. An alternate approach is to calculate the Kullback-Leibler distance, or relative entropy, but the method is fairly expensive [7] with similar results to using the χ^2 test. Modifications to the χ^2 test will be needed for implementation, as shown below.

The Chi-Squared (χ^2) test is a goodness-of-fit test that is executed every time step and is used to determine how well the distribution of particles follows an assumed distribution; in this work the desired distribution is a Gaussian.

In order to calculate the value of χ^2, a histogram is made of the position estimates with $n_b = 13$ evenly distributed bins located between ± 3 standard deviations of the mean such that the bin width is $dx_{bin} = \sigma_x/2$ and the bin centers are located at

$$b = [\mu_x - 3\sigma_x, \mu_x - 2.5\sigma_x, \ldots, \mu_x + 3\sigma_x] \tag{15}$$

where μ_x is the mean of the position particles and σ_x is the standard deviation of the position particles from the mean. The bin locations are chosen to fully capture the shape of the desired Gaussian distribution for comparison with the particle population. It should be noted that the time index "k" was removed from the above and following equations in order to simplify the notation; however, all parameters in the

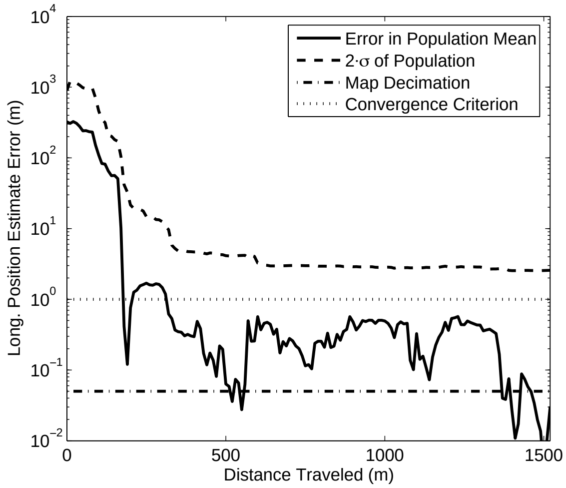

Fig. 4: Position estimate error as a function of the distance traveled.

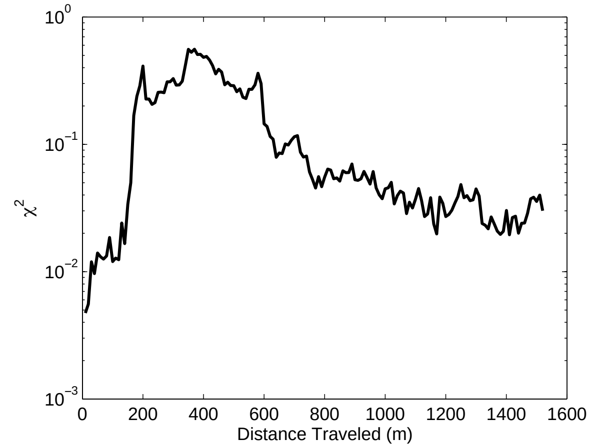

Fig. 5: χ^2 test values as a function of the distance traveled.

χ^2 test and those discussed hereafter are updated every time step.

The value of the histogram at each bin (h_i) is the number of particles in each bin (n_i), normalized by the total number of particles (N) and the bin width (dx_{bin})

$$h_i = \frac{n_i}{N \cdot dx_{bin}} \tag{16}$$

such that h_i represents the probability density of the particles.

The desired value at each bin location is calculated from a Gaussian pdf

$$G_i = \frac{1}{\sigma_x \cdot \sqrt{2\pi}} \cdot exp\left(-\frac{z_i^2}{2 \cdot \sigma_x^2}\right) \tag{17}$$

where $z_i = b_i - \mu_x$. Then χ^2 is calculated as

$$\chi^2 = \sum_{i=1}^{n_b} \frac{(h_i - G_i)^2}{G_i} \tag{18}$$

The χ^2 value can then used to determine when the particle population is Gaussian enough to initialize an UKF algorithm by selecting a threshold value (χ^2_{min}) such that when $\chi^2 < \chi^2_{min}$ the algorithm switches from using the PF to using the UKF. The UKF is initialized using μ_x and σ_x of the particle estimates at the switchover point.

To demonstrate the use of the χ^2 parameter, the PF was implemented using map and in-vehicle measurement data collected at the LTI test track with $N = 1,000$ particles, $dx = 10$ meters, $R = 0.1$ deg^2, and $Q = (0.01 \cdot dx)^2$ m^2.

The results of the PF are shown in Fig. 4 where the position estimate error and two times the standard deviation of the particle population $(2\sigma_x)$ are plotted as a function of the distance of vehicle travel.

It can be seen that the position estimate converges to sub-meter accuracy, or when the error in the position estimate is less than one meter, after about 300 meters of travel. The standard deviation $(2\sigma_x)$ is also shown to converge to about 5 meters at that location. With this level of accuracy and precision, we can justly conclude that the PF has converged to correctly estimate the position of the vehicle. It should also be noted that the PF maintains the accurate estimate of the vehicle's position as it completes the loop around the track.

The value of χ^2 was also calculated throughout the simulation and is shown in Fig. 5 as a function of the distance travelled.

It can be seen in Fig. 5 that the initial χ^2 value (about 0.005) is lower than when the PF has converged to the correct vehicle position (about 0.3 at D=300 meters). This is because the Gaussian approximation to the particle population is calculated from σ_x, which is initially very large to represent the initial uniform pdf of the particle population. Thus, the initial χ^2 value is low because the Gaussian approximation with the large initial σ_x is a good fit to the initial, uniform particle pdf, despite the fact that the PF has not converged to the correct estimate. This continues to be the case until the particle population becomes multi-modal and the assumed Gaussian pdf is no longer a good fit.

One can observe from these results that use of the χ^2 test as a switchover metric yields poor results. Due to the initial values of χ^2 being lower than when the PF is converged, choosing a PF to UKF switchover value of χ^2_{min} lower than the initial value is not possible. Thus, the UKF will never be activated. It is therefore necessary to modify the χ^2 test to develop a metric that decreases as the particle distribution becomes more Gaussian and more converged.

One possible modification to the χ^2 test becomes evident when examining the standard deviation σ_x along with the χ^2 value. Although the initial χ^2 values are very low, the initial σ_x is very high, while at the end of the algorithm σ_x is low, as shown in Fig. 4, and χ^2 is also low. Thus, the test can be modified by scaling χ^2 with σ_x such that Eq. (18) is modified to

$$\Upsilon^2 = \chi^2 \cdot \sigma_x^2 = \sum_{i=1}^{n_b} \sigma_x^2 \cdot \frac{(h_i - G_i)^2}{G_i} \tag{19}$$

This new "Upsilon-squared" test (named because Upsilon

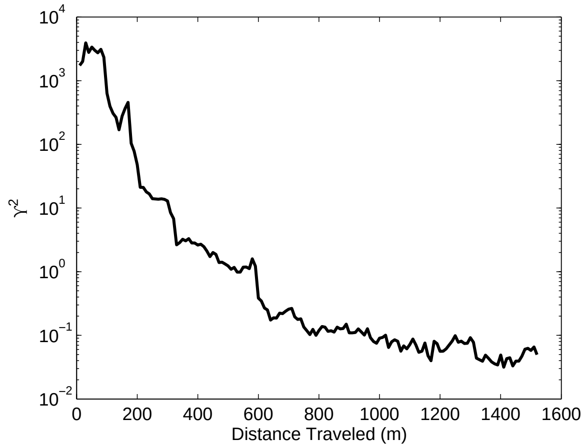

Fig. 6: Υ^2 test values as a function of the distance traveled.

preceeds Chi in the greek alphabet) is used on the data from the previous results and is shown in Fig. 6.

In comparison to Fig. 5, the values of Υ^2 in Fig. 6 are clearly reduced when the PF algorithm converges to the correct vehicle position and the pdf is uni-modal. Thus, a threshold value (Υ^2_{min}) can be chosen such that when $\Upsilon^2 < \Upsilon^2_{min}$ the particle population can be considered "Gaussian enough" and the algorithm switches from the PF to the UKF.

V. Optimal Threshold

In order to minimize computational effort and maximize estimate accuracy, this section presents a methodology to determine the optimal threshold value of Υ^2_{min} for switching from the PF to the UKF estimation method. A Monte Carlo simulation is used to repeat the initialization algorithm 25 times for each of the varying values of Υ^2_{min}. Each simulation is initialized with a different random initial population for the PF and iterated until $\Upsilon^2 < \Upsilon^2_{min}$, when the UKF is initialized, and iterated until the vehicle completes one loop around the test track.

The final estimate error (E_x) calculated using the true vehicle position measured using DGPS, and standard deviation (σ_x) for each simulation around the track are averaged over the final 300 meters of travel. Additionally, the number of floating point operations (FLOPS) is also calculated and normalized by the number of FLOPS that would have been required for a pure PF implementation ($5.63\text{x}10^6$). This process is repeated over 25 runs, and the average values are recorded. The resulting average estimate error and normalized FLOPS are shown in Fig. 7.

The process is also repeated using data collected along highway 322 in State College, PA with a map decimation of 0.5 m, and with $N = 7,115$ particles, $dx = 25$ meters, $R = 0.1$ deg^2, and $Q = (0.01 \cdot dx)^2$ m^2. The results are also shown in Fig. 7.

From Fig. 7, it can be seen that the average estimate error is relatively constant for low values of Υ^2_{min}; hence, a low value of Υ^2_{min} is desirable. It can also be seen that the LTI

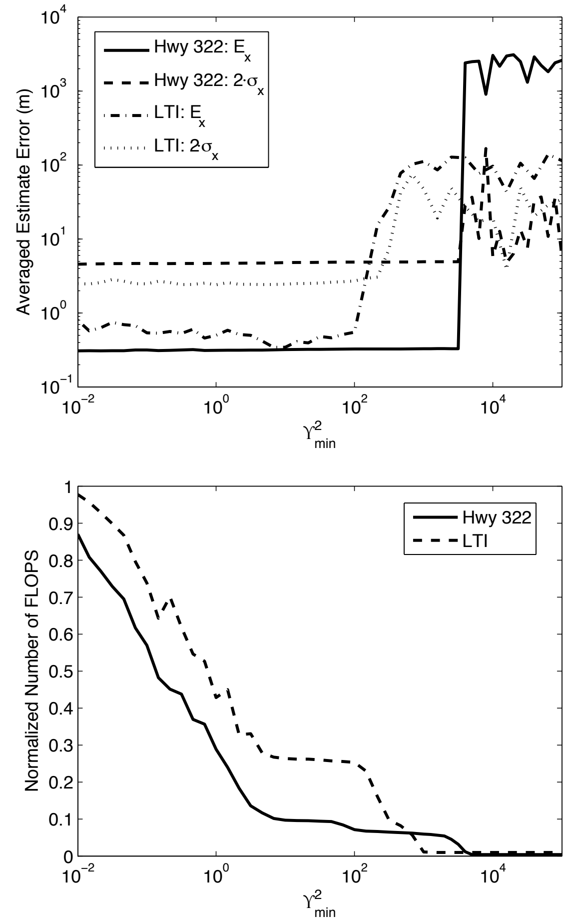

Fig. 7: Tradeoff in the average estimate error and normalized floating point operations as a function of Υ^2_{min}.

estimate error for $\Upsilon^2_{min} > 100$ is larger than $2\sigma_x$, indicating that either the PF converged to an erroneous position or the PF converged correctly but the UKF failed to maintain the true vehicle position within the expected range of its pdf, hence the algorithm "lost" the vehicle. This also occurred with the highway 322 data for $\Upsilon^2_{min} > 3,000$.

Fig. 7 also demonstrates the decrease in computational effort as Υ^2_{min} increases, as evidenced by the decrease in the number of FLOPS; hence, it is desirable to have a large Υ^2_{min} such that the UKF is initialized quickly. Thus, a tradeoff in choosing Υ^2_{min} exists between reducing the estimate error and the computational cost.

After comparing the results in Fig. 7, it is evident that suitable threshold values are $\Upsilon^2_{min} \in [0.01, 100]$ for estimation accuracy and $\Upsilon^2_{min} \in [10, 10,000]$ for computational effort. However, because the Gaussian approximation is better for smaller values of Υ^2_{min}, a threshold of $\Upsilon^2_{min} = 10$ is chosen as a good tradeoff between the algorithm accuracy and computational load.

VI. UKF Initialization Results

A. LTI Test Track

With the threshold value of $\Upsilon^2_{min} = 10$, the combined PF and UKF algorithm is used used to estimate the vehicle

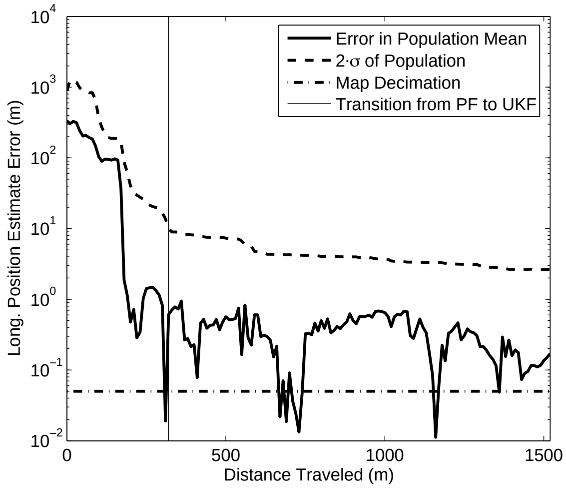

Fig. 8: The particle position estimate error and standard deviation as a function of the distance traveled. The solid vertical line indicates the point of transition from the PF to the UKF.

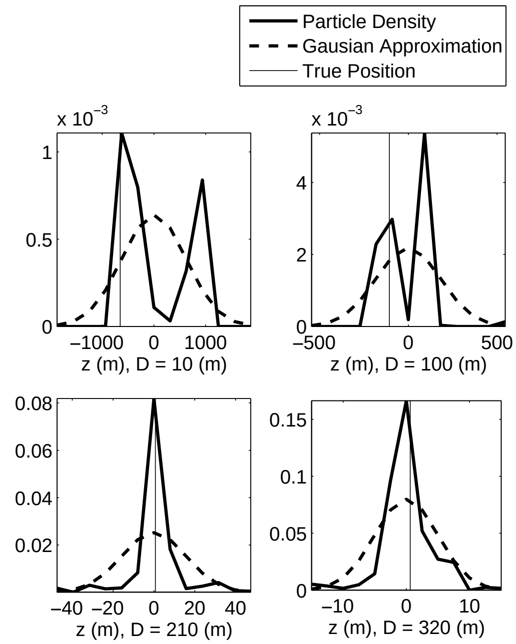

Fig. 9: Particle population and Gaussian probability density functions at progressive travel distances.

position along the LTI test track. The resulting position estimate error is demonstrated in Fig. 8 where it can be seen that once the PF converged and the UKF was initialized, the UKF continued to estimate the vehicle position with sub-meter accuracy. The distance at which the algorithm switched from a PF to UKF is indicated by the solid vertical line.

The pdf of the particle population and the Gaussian distribution at several intervals up to the time of transition is shown in Fig. 9. The advantage of the PF is seen in the first frame at $D = 10$ meters when the pdf is bimodal as a result of maintaining two possible vehicle locations; the true vehicle position relative to the estimated vehicle position is indicated by the solid vertical line. After 320 meters of travel the bimodal pdf has converged to a Gaussian distribution that has satisfied the Υ^2_{min} criterion.

From these results, it is evident that the Υ^2 test is an accurate means of determining when the particle population is “Gaussian enough” for initializing a PF. Also, it is evident that the UKF was capable of maintaining a sub-meter vehicle position estimate while reducing the computational cost. In fact, the PF required $37 \cdot N + 9$ FLOPS per iteration, or 37,009 FLOPS per iteration using $N = 1,000$ particles; the UKF, however, required $38 \cdot N_s + 6$ FLOPS per iteration, or 120 FLOPS per iteration using $N_s = 3$ sigma points. Thus, using the UKF resulted in a substantial 99.7% reduction in computations per iteration.

B. Highway Implementation

Now that the UKF algorithm has been shown to localize a vehicle accurately along the LTI test track, the algorithm is further tested using the data collected along highway 322 in State College, PA. The advantages of using a UKF is even more evident when in use over a long period of time, or a longer travel distance. The resulting estimate error as a function of the distance traveled is shown in Fig. 10.

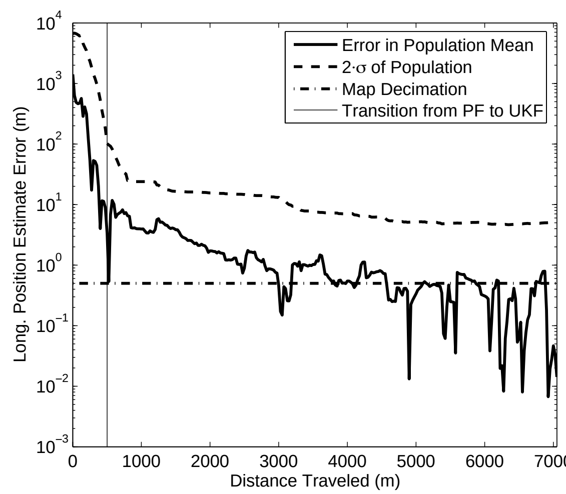

Fig. 10: The particle position estimate error and standard deviation as a function of the distance traveled along the highway. The solid vertical line indicates the point of transition from the PF to the UKF.

It can be seen in Fig. 10 that the algorithm transitioned from the PF to the UKF with only 10 meters of position accuracy; however, once converged, the UKF algorithm was able to improve to and maintain sub-meter accuracy through the remainder of the simulation. The particle population is shown to converge to a Gaussian distribution in Fig. 11 where it is evident that the PF particle population satisfied

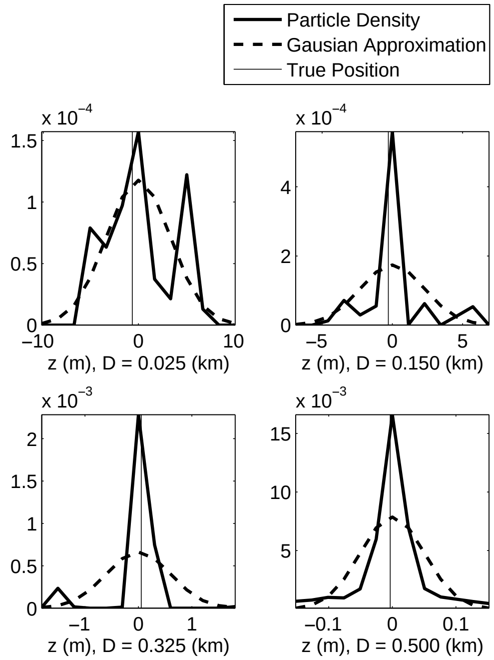

Fig. 11: Particle population and Gaussian probability density functions at progressive travel distances along the highway.

the convergence criterion after only 500 meters of travel.

The computational savings of using the UKF can be seen in comparing the number of FLOPS per iteration. With $37 \cdot N + 9$ FLOPS per iteration and using $N = 7,115$ particles, the PF required 263,264 FLOPS per iteration. This a substantial increase in computational effort from the LTI simulation due to the large number of particles. The UKF, however, required only 120 FLOPS per iteration, as calculated above. Thus, initializing and switching from a PF to an UKF resulted in a 99.95% reduction in computational effort per iteration.

VII. Switchover Control with Innovation Monotoring

The Υ^2 test has shown the capability of determining when the pdf of a particle population is Gaussian enough for initializing a UKF, and the algorithm has demonstrated the ability to maintain an accurate vehicle position estimate. It is a concern, however, that the algorithm be capable of maintaining an accurate position estimate in the presence of unexpected vehicle behavior, accelerations, lane deviations, or changes in the terrain. For example, if a vehicle diverts from the mapped roadway, the vehicle position will be lost; a successful algorithm needs to be capable of re-initializing the vehicle position estimate when an inaccurate estimate is suspected.

If the UKF algorithm loses the vehicle position, the UKF needs to be aware that the position estimate is no longer accurate. This can be accomplished by means of a Normalized Innovation Squared (NIS) test [8] which can be used to monitor the health of the UKF estimate. The NIS

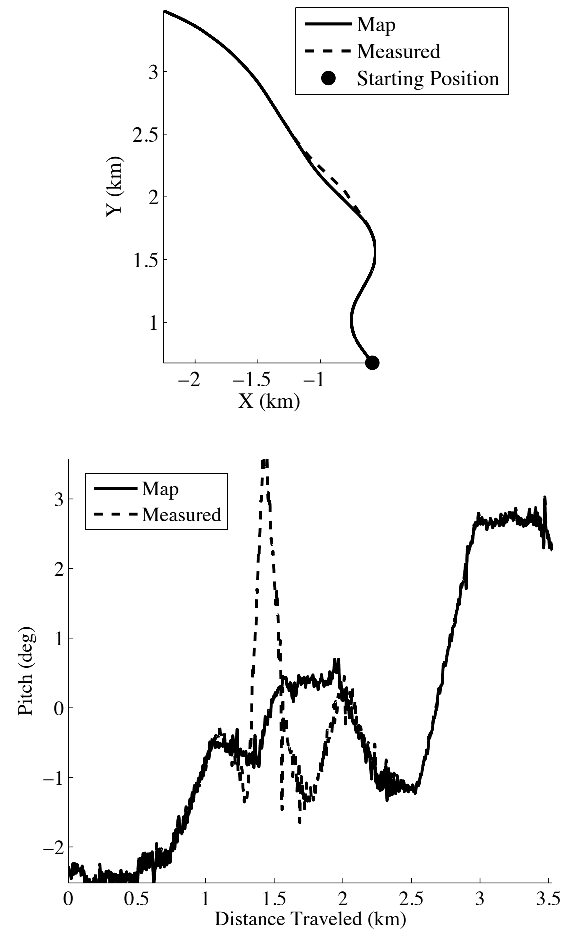

Fig. 12: In order to simulate a departure and re-entry from the highway, false terrain data is inserted into the pitch map collected along highway 322 in State College, PA.

error (ϵ) is calculated as

$$\epsilon_k = \nu_k \cdot P_{yy,k}^{-1} \cdot \nu_k \tag{20}$$

where ν is the innovations of the UKF, calculated as:

$$\nu_k = \theta_a - \hat{y}_k^- \tag{21}$$

Once the NIS error reaches a specified maximum limit ($\epsilon > \epsilon_{max}$) then the UKF is stopped and a PF is re-initialized with N particles scattered randomly across the entire map.

In order to demonstrate this method, vehicle data is collected again along small portion of highway 322, as shown in Fig. 12, where the vehicle exits the highway at an off-ramp, then quickly re-enters the highway, simulating a path deviation. Thus, the measured in-vehicle pitch is offset slightly from the pitch map and large deviations in the pitch measurements are introduced.

Using this data, it is assumed that the algorithm has already converged, thus the UKF algorithm is initialized to the correct vehicle position; the algorithm is then implemented without using the NIS test to monitor the innovations. The resulting estimate error is shown in Fig. 13 where it can be seen that the UKF, without using the NIS test, was unable

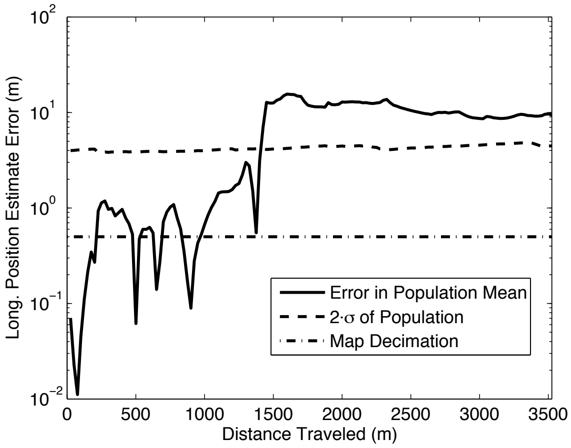

Fig. 13: Vehicle position estimate error without using the NIS test.

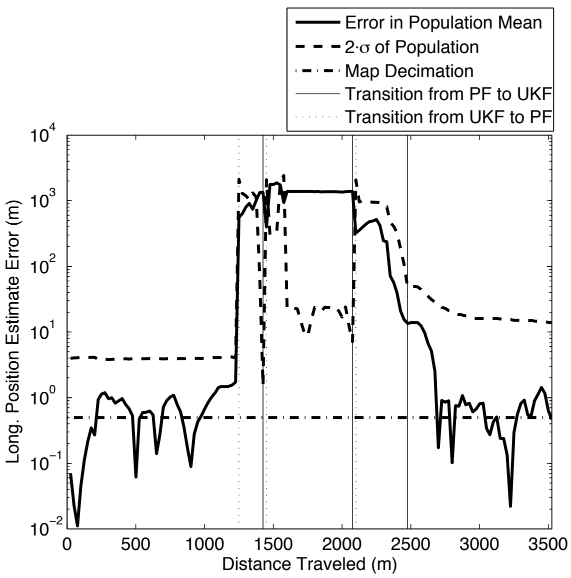

Fig. 14: Vehicle position estimate error with the NIS test and multiple switchover points; the solid vertical line represents the point of transition from the PF to the UKF algorithm and the dashed vertical line represents the transition back from the UKF to the PF algorithm.

to maintain an accurate position estimate during highway departure or recover to an accurate estimate once the vehicle returned to the highway, as evidenced by the fact that the estimate error is greater than $2\sigma_x$.

To demonstrate the switchover control using the NIS test, a value of $\epsilon_{max} = 1$ is chosen (by trial and error) and the UKF algorithm is implemented again with the position estimate initialized to the correct vehicle position. The resulting estimate error is shown in Fig. 14.

It can be seen in Fig. 14 that when the vehicle departed from the mapped lane around 1.1 km of travel, the NIS test indicated an increase in estimate error and stopped the UKF to initialize the PF, as indicated by the dashed vertical line. Then, for nearly a kilometer, while the vehicle was traveling off the mapped lane, the algorithm converged twice to an erroneous position estimate and continued to switch between the PF and UKF. Around 2.1 km of travel, once the vehicle was again traveling along the mapped lane, the PF converged to an accurate position estimate until the algorithm transitioned to the UKF at 2.5 km. The UKF was then able to converge and maintain sub-meter accuracy for the remainder of the experiment.

Thus, it can be seen that the NIS test is capable of monitoring the health of the UKF. Also, the combination of using the Υ^2 test for determining when to initialize the UKF and the NIS test for monitoring the health of the UKF has demonstrated an increase in the robustness of the terrain-based localization algorithm. Similar innovation monitoring can also be implemented on the PF algorithm to increase the robustness even further, although that was not performed here in order to demonstrate the accuracy of using the Υ^2 and NIS tests.

VIII. Conclusions

A UKF vehicle positioning algorithm was shown to be capable of localizing a vehicle's longitudinal position with sub-meter accuracy and independently of GPS. Use of the UKF was shown to reduce the computational effort of the PF by over 99%. The UKF was initialized using a variance estimate from a PF algorithm. The switchover from the PF to the UKF algorithm was found to be unsuccessful when using a chi-squared test. However, a modification to the chi-squared test was developed and shown to be effective. The UKF was shown to lose the correct estimate when the vehicle departs the roadway even temporarily, but by monitoring the innovation error of the UKF, it was shown that the PF could be re-initialized and could re-capture the correct position estimate in circumstances where the UKF fails.

References

[1] A. Dean, R. Martini, and S. Brennan, "Terrain-based road vehicle localization using particle filters," in *Proceedings. 2008 American Control Conference*, Jun. 11–13, 2008.

[2] A. Dean, P. Vemulapalli, and S. Brennan, "Highway evaluation of terrain-aided localization using particle filters," in *Proceedings. 2008 Dynamic Systems Control Conference*, Oct. 20–22, 2008.

[3] E. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000*, pp. 153–158, 2000.

[4] H. Xu, C. Wang, M. Yang, and R. Yang, "Position estimation for intelligent vehicles using an unscented Kalman filter," *International Journal of Vehicle Autonomous Systems*, vol. 6, no. 1, pp. 186–194, 2008.

[5] T. Fitzgibbons and E. Nebot, "Bearing Only SLAM using Colour-based Feature Tracking," *2002 Australasian Conference on Robotics and Automation, Auckland, New Zealand*, 2002.

[6] L. Montesano, J. Gaspar, J. Santos-Victor, and L. Montano, "Fusing vision-based bearing measurements and motion to localize pairs of robots," *International Conference on Robotics and Automation*, 2005.

[7] T. Bailey, "Constrained Initialization for Bearing-only SLAM," *IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1966–1971, 2003.

[8] Y. Bar-Shalom and X. R. Li, *Estimation and Tracking: Principles, Techniques, and Software*. Artech House, Norwood, MA, 1993.