JAUES

Journal of Al Azhar University Engineering Sector

Vol. 2, No. 2, January 2007, 21-39

UNMANNED GROUND VEHICLE NAVIGATION

Ayman El Badawi[1] and Haitham M. Rifai[2]

[1] Mechanical Engineering Dept. Faculty of Enfineering, Al Azhar University

[2] Egyptian Natural Gas Co. "GASCO"

الهدف من هذا البحث هو التنبؤ بدقة الملاحة لعربه ذاتية التوجيه بإستخدام مرشح كالمان. هذا المرشح يستخدم لتقدير موقع العربه بناء على مجموعة حساسات مشوشة الإشاره بمدخلات غير متزامنه. فى البدايه تم صياغة معادلات كامله للحركة العامه للعربه. ثم تم عمل إفتراضات مبسطه لصياغة المعادلات للإستخدام فى مستوى ثنائى الأبعاد. تم عمل محاكاه للحساسات و تم حساب الحالات بإستخدام مرشح كالمان الخطى. و عليه تم تقدير الخطأ فى موقع العربه. تم تطبيق مرشح كالمان الخطى بنجاح على الصياغه الملاحيه. تمكن مرشح كالمان من التعامل مع مجموعه من الحساسات المشوشة الإشاره و بمدخلات غير متزامنه.

ABSTRACT

The objective of this paper is to predict the navigation accuracies for an unmanned ground vehicle (UGV) that uses a Kalman filter. The Kalman filter is used to estimate the position of (UGV) based on noisy sensors with asynchronous inputs. First, a complete formulation is described for general UGV use. Next, simplifying assumptions are made to produce a formulation that is appropriate for a two-dimensional plane. The simulated UGV follows a course through a 2-D plane. The estimated states are determined through a linear Kalman filter with simulated sensor inputs. The estimated position error is evaluated. The linear Kalman filter approach was successfully applied to the simplified navigation formulation. The Kalman filter was able to handle multiple asynchronous inputs from noisy sensors.

KEYWORDS: Unmanned, Autonomous, Kalman Filter, Localization.

1. INTRODUCTION

In order to autonomously navigate and perform useful tasks, a mobile robot needs to know its exact position and orientation. Robot localization is thus a key problem in providing autonomous capabilities to a mobile robot. The different techniques that have been developed to tackle this problem can be classified into two main categories:

i. Relative (local) localization: evaluating the position and orientation using information provided by various on-board sensors (e.g. encoders, gyroscopes, accelerometers etc).

ii. Absolute (global) localization: obtaining the absolute position using beacons, landmarks or satellite-based signals (e.g. GPS).

A popular local technique, dead reckoning, employs simple geometric equations (a kinematic model of the robot) on using for example encoder data to compute the position of the robot relative to its start position. Dead reckoning cannot be used for long distances because it is prone to errors from external factors. For example, the kinematic model always has some inaccuracies, encoders have limited precision and there are external sources affecting the motion that are not observable by the sensors (e.g. wheel slippage). The localization error grows with time. Applying Kalman filter techniques can provide substantial improvement [1].

In case of absolute localization the error growth is neglected when measurements are available. The position of the robot is externally determined and its accuracy is usually time and location independent. In other words integration of noisy data is not required and thus there is no accumulation of error with time or distance traveled. The problem in absolute localization (e.g. using GPS) is that one cannot keep track of the robot for small distances (even with accurate GPS devices). Commercial GPS gives errors on the order of 3m at each measurement [2]. If the robot moves at 1 m/s one cannot use GPS at each second since the odometric estimate is less errorprone than the GPS measurement.

A variety of methods are used to solve the problem of estimating the exact position of the mobile robot. Combining data from multiple sources (sensors), often called sensor fusion, is a common approach.

Fusion is used, for example, to navigate mobile robots, for autonomous underwater vehicles, or for autonomous aircraft landing systems. The fusion of sensors enables the construction of a sensor system which is able to accurately estimate the position. The Kalman Filter (KF) is a technique of data sensor fusion.

1.1. KALMAN FILTER

Robots usually include a large number of heterogeneous sensors, each providing clues as to robot position and, critically, each suffering from its own failure modes. Optimal localization should take into account the information provided by all of these sensors. The Kalman filter technique incorporates all information, regardless of precision, to estimate the current value of the variable of interest (i.e., the robot's position). A general introduction to Kalman filters can be found in [3],[4].

Fig. 1 shows the general scheme of Kalman filter estimation, where a system has a control signal and system error sources as inputs. A measuring device enables measuring some system states with errors. The Kalman filter is a mathematical mechanism for producing an optimal estimate of the system state based on the knowledge of the system, the measuring devices, the description of the system noise and measurement errors and the uncertainty in the dynamic model.

Kalman filtering is a recursive estimation procedure that uses sequential sets of measurements. Prior knowledge of the state (expressed by the covariance matrix) is improved at each step by taking the prior state estimates and new data for the subsequent optimal state estimation [5].

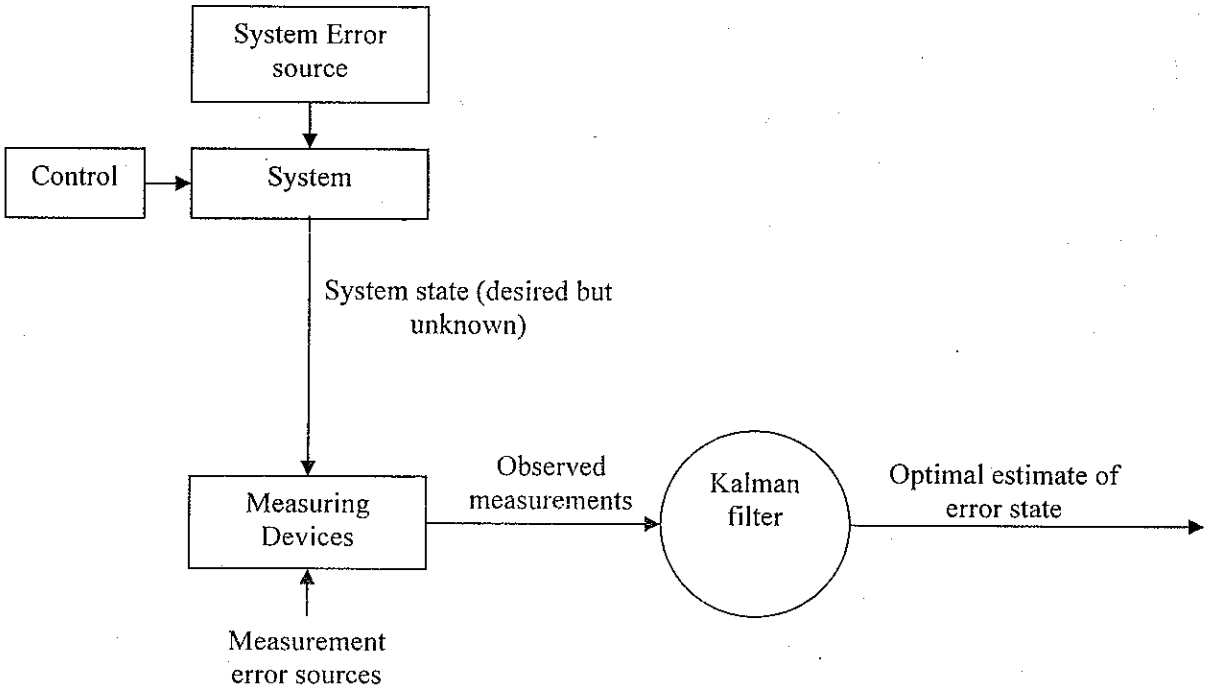

Fig. 1 General theme of Kalman filter estimation.

Optimality depends on the criteria chosen to evaluate the performance and on the assumptions. Within the Kalman filter theory the system is assumed to be linear and white with Gaussian noise. The assumption of Gaussian error is invalid for mobile robot applications but, nevertheless, the results are extremely useful. In other engineering disciplines, the Gaussian error assumption has in some cases been shown to be quite accurate [3]

2. KALMAN FILTER FORMULATION

The linear Kalman filter implements five equations Eq. 1 to. Eq. 5. The use of the five equations depends on the presence or absence of sensor measurements. When measurements are present, the Kalman filter uses Eq. 1 to Eq. 3, on the other hand Eq.4 and Eq. 5 operates for every time step, regardless of the presence or absence of the measurements. The first step used when measurements are present was to compute the Kalman gain:

$$K_k = P_k^- H_k^T \left[H_k P_k^- H_k^T + R_k \right]^{-1} \qquad \text{Eq. 1}$$

Next, the estimated state was updated:

$$\hat{x}_k = \hat{x}_k^- + K_k \left[z_k - H_k \hat{x}_k^- \right] \qquad \text{Eq. 2}$$

The final step when measurements are present was to update the covariance matrix:

$$P_k = P_k^- - K_k \left(H_k P_k^- \right) \qquad \text{Eq. 3}$$

For every time step, regardless of the absence or presence of measurements, the state was projected ahead:

$$\hat{x}_{k+1}^{-} = \Phi_k \hat{x}_k \qquad \text{Eq. 4}$$

Finally, the covariance matrix for next measurement was updated at every time step:

$$P_{k+1}^{-} = \Phi_k P_k \Phi_k^T + \Gamma_k Q_k \Gamma_k^T \qquad \text{Eq. 5}$$

Where $P_1(k)$ is the predicted error covariance matrix at time k, $K(k)$ is the Kalman gain matrix at time k, and $P(k)$ is the updated error covariance matrix at time k. R is the measurement noise matrix, which contains the expected covariance of the measurement data. Q is the state noise matrix, which contains the expected covariance of the state matrix values. Essentially, R is a measure of confidence in the measurements and Q is a measure of confidence in the state estimate. That is, if an extremely precise instrument were being used to quantify a particular measurement variable, the corresponding value in the measurement noise matrix, R, would be very low.

A general formulation for a vehicle with x, y, z, θ, ϕ, ψ, and β states was described. This was a general case that is suitable for an actual UGV navigation algorithm implementation. Next the problem was simplified for this research. The simplified approach may be extended to the general UGV navigation algorithm in future efforts.

3. GENERAL UGV NAVIGATION FORMULATION

There are numerous sources for navigation using Kalman filters. Several of those Kelly [6], Murphy [7], Parkinson [8], Chroust [9], Cremean [10], Hong [11], Basir [12]. Kelly [6] appears to have the most comprehensive and relevant treatment of UGV navigation Kalman filter algorithms and this general approach is applied here. Kelly is often referenced in UGV literature.

The UGV coordinate system used here is shown in Fig.2. and Fig.3. The navigation frame x-axis points East, the y-axis points North, while the z-axis points directly upwards. The navigation frame is local. The vehicle body frame and the navigation frame are aligned when the vehicle is level and facing North. A flat Earth assumption was made for this effort. If the UGV travels any significant distance, a model that accounts for the Earth's curvature must be applied.

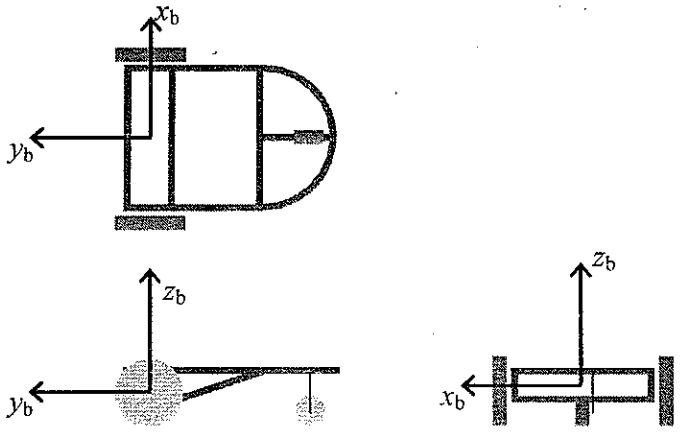

Fig. 2. UGV Coordinate System.

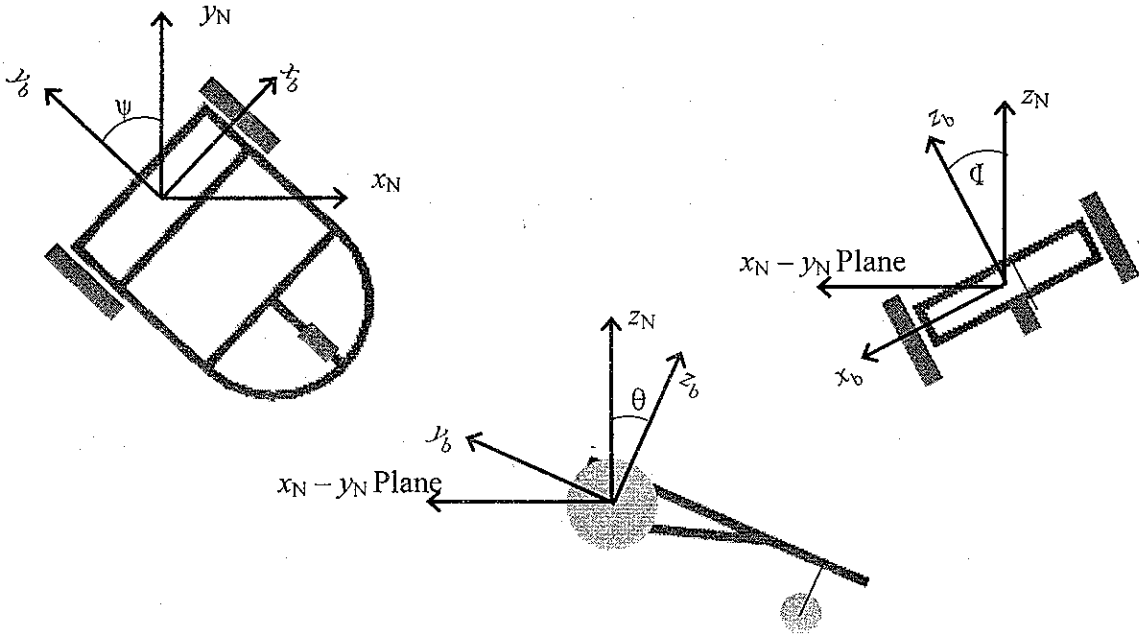

Fig. 3. UGV Rotation Angle Definition.

For a UGV, the state variables are:

$$\bar{x} = \begin{bmatrix} x & y & z & V & \theta & \phi & \psi & \dot{\beta} \end{bmatrix}^T \qquad \text{Eq. 6}$$

Where x, y, and z is the position in the navigation frame, and θ, ϕ, and ψ are the angles between the body frame and the navigation frame. The velocity, V, is constrained to occur only along the y_b axis. The turn rate $\dot{\beta}$ is the angular rate around the z_b axis.

The state transition matrix was used to update the state at each time step based on the previous state estimate. The state transition matrix is:

$$\Phi = \begin{bmatrix} 1 & 0 & 0 & -s\psi \cdot c\theta \cdot dt & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & c\psi \cdot c\theta \cdot dt & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & s\theta \cdot dt & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & s\phi \cdot dt \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & -t\theta \cdot c\phi \cdot dt \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & \frac{c\phi}{c\theta} \cdot dt \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Eq. 7}$$

The Q matrix contains the maximum expected state errors. The Q matrix is assumed to be:

$$Q = \begin{bmatrix} \sigma_x^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_y^2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_z^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_V^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_\theta^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_\phi^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \sigma_\psi^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \sigma_{\dot{\beta}}^2 \end{bmatrix} \quad \text{Eq. 8}$$

The sensors considered here included GPS, velocity encoder, Doppler velocity sensor, compass, Attitude Heading Reference System (AHRS), and a steering sensor. The complete measurement considered here is:

$$\bar{z} = \begin{bmatrix} x_{GPS} & y_{GPS} & z_{GPS} & V_{enc} & V_{od} & \psi_{com} & \theta_{AHRS} & \phi_{AHRS} & \psi_{AHRS} & \alpha \end{bmatrix}^T \quad \text{Eq. 9}$$

It is also common to replace the AHRS with gyros to measure angular accelerations, integrate the angular accelerations over time to get angular velocities, and integrate the angular velocities to get angular position. The angular drift may be corrected by means of a compass, a bubble level, or both. This approach essentially performs the same function as an AHRS. Bruch [2000] describes the implementation of a UGV navigation Kalman filter that uses GPS, a 3-axes compass, velocity encoders, and a gyro. A number of other sensor combinations are possible.

The H matrix, called the measurement matrix or observation matrix, relates sensor readings to the vehicle state. The complete measurement matrix is:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \frac{\partial \alpha}{\partial V} & 0 & 0 & 0 & \frac{\partial \alpha}{\partial \dot{\beta}} \end{bmatrix} \quad \text{Eq. 10}$$

[6]

Kelly [6] provides the derivatives of α as:

$$\frac{\partial \alpha}{\partial V} = \frac{-L \cdot \dot{\beta}}{1 + \left(L \cdot \dot{\beta}\right)^2} \quad \text{and} \quad \frac{\partial \alpha}{\partial \dot{\beta}} = \frac{L \cdot V}{1 + \left(L \cdot \dot{\beta}\right)^2} \quad \text{Eq. 11}$$

However, a simple dimensional analysis of the denominator indicates an error. The velocity derivative of steering angle is dropped from further consideration here, since the Doppler and encoder are the primary velocity sensors. The turn rate derivative of steering angle is re-derived to produce:

$$\frac{\partial \alpha}{\partial \dot{\beta}} = \frac{L \cdot V}{V^2 + \left(L \cdot \dot{\beta}\right)^2} \quad \text{Eq. 12}$$

When no sensor reading is present, the non-zero value in the H matrix corresponding to the idle sensor is set to zero. This prevents the Kalman filter from incorporating outdated, null, or erroneous measurements.

The sequence covariance matrix, or R matrix, measures the expected sensor errors. These errors are set to the sensor error standard deviation for the purposes of this effort. The R matrix is assumed to be:

$$R = \begin{bmatrix} \sigma^2_{GPS,xy} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma^2_{GPS,xy} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma^2_{GPS,z} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma^2_{enc} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma^2_{dop} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma^2_{comp} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \sigma^2_{pitch} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \sigma^2_{roll} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \sigma^2_{yaw} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \sigma^2_{steer} \end{bmatrix} \quad \text{Eq. 13}$$

4. SIMPLIFIED UGV NAVIGATION FORMULATION

The formulation described above is now simplified for the purposes of this research. The key assumption is that the vehicle traverses perfectly smooth, level terrain. No elevation changes occur, so there is no need to measure or track the vehicle altitude (z-coordinate). The smooth terrain assumption means that the vehicle is not perturbed in pitch, roll, or yaw by high-frequency local or large-scale terrain features. The vehicle travels in the x-y plane and rotates only about the z-axis in the vehicle body frame.

With these assumptions, states z, θ, and ϕ are no longer necessary.

The vehicle state variables become:

$$\bar{x} = \begin{bmatrix} x & y & V & \psi & \dot{\beta} \end{bmatrix}^T \quad \text{Eq. 14}$$

With the smooth, level terrain assumption the state transition matrix is greatly simplified. Note that the sine of all pitch and roll angles becomes 0, and the cosine of all pitch and roll angles becomes 1. The vehicle state transition matrix becomes:

$$\Phi = \begin{bmatrix} 1 & 0 & -s\psi \cdot dt & 0 & 0 \\ 0 & 1 & c\psi \cdot dt & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & dt \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Eq. 15}$$

The disturbance covariance matrix becomes:

$$Q = \begin{bmatrix} \sigma^2_x & 0 & 0 & 0 & 0 \\ 0 & \sigma^2_y & 0 & 0 & 0 \\ 0 & 0 & \sigma^2_V & 0 & 0 \\ 0 & 0 & 0 & \sigma^2_\psi & 0 \\ 0 & 0 & 0 & 0 & \sigma^2_{\dot{\beta}} \end{bmatrix} \quad \text{Eq. 16}$$

The GPS z-position measurement is no longer needed. Also, the AHRS pitch and roll measurements are not necessary. The measurement list becomes:

$$\bar{z} = \begin{bmatrix} x_{GPS} & y_{GPS} & V_{enc} & V_{od} & \psi_{com} & \psi_{AHRS} & \alpha \end{bmatrix}^T \quad \text{Eq. 17}$$

The fully-populated measurement matrix becomes:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & \frac{\partial \alpha}{\partial \dot{\beta}} \end{bmatrix} \quad \text{Eq .18 [6]}$$

The sequence covariance matrix becomes:

$$R = \begin{bmatrix} \sigma^2_{GPS,xy} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma^2_{GPS,xy} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma^2_{enc} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma^2_{od} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma^2_{comp} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma^2_{yaw} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \sigma^2_{steer} \end{bmatrix} \quad \text{Eq.19}$$

4.1. State Disturbance Errors

Kelly [6] recommends that the position errors for the disturbance covariance matrix be estimated as:

$$\sigma_x = \frac{a_{\max,x} \cdot \Delta t^2}{2} \text{ and } \sigma_y = \frac{a_{\max,y} \cdot \Delta t^2}{2} \quad \text{Eq. 20}$$

Where Δt is inverse of the sampling rate. The recommended accelerations are:

$$a_{\max,x} = 0.1 \cdot a_{\max} \text{ and } a_{\max,y} = 1.0 \cdot a_{\max} \quad \text{Eq. 21}$$

Kelly [6] recommends that the maximum acceleration for a vehicle operating on rough terrain as 1g.

The angular position state disturbance errors are estimated as:

$$\sigma_{\psi} = 1.0 \cdot \Omega_{\max} \qquad \text{Eq. 22}$$

Where Ω_{max} is the maximum angular velocity. The maximum angular velocity can be related to the vehicle natural frequency, f, by:

$$\Omega_{\max} = 2 \cdot \pi \cdot f \qquad \text{Eq. 23}$$

A vehicle dynamics analysis must be performed to determine the vehicle natural frequency. The vehicle natural frequency is simply an input to this code.
The velocity disturbance error is estimated by:

$$\sigma_{V} = a_{\max} \cdot \Delta t \qquad \text{Eq. 24}$$

where a_{max} is once again assumed to be 1g for rough terrain.

The angular velocity state disturbance error is estimated by:

$$\sigma_{\dot{\beta}} = \alpha_{\max} \cdot \Delta t \qquad \text{Eq. 25}$$

$$\alpha_{\max} = \frac{\Omega_{\max}}{\tau} \qquad \text{Eq. 26}$$

where τ is a time constant that is similar to the time required to get from zero to the maximum angular velocity state. A vehicle dynamics analysis must be performed to estimate the value of τ, and it is simply input to the code for this effort.

Velocity Sensors
The encoder and odometer velocity errors are estimated by:

$$\sigma_{enc} = SFE_{enc} \cdot V_{enc} \qquad \text{Eq. 27}$$

$$\sigma_{dop} = SFE_{dop} \cdot V_{od} \qquad \text{Eq. 28}$$

respectively, where the odometer and encoder scale factor errors are dependent upon the sensors.

Heading Sensors
Kelly [6] recommends conservatively using the vendor's estimates for the AHRS and compass attitude errors.

Steering Sensor
The steering position error is modeled as an absolute error. This error must be determined from tests.

5. SIMULATION

The simulation environment models the motion of the UGV and sensor measurements. The Kalman filter acts upon the measurement information provided by the sensor models. The output of the navigation algorithm is compared against the simulated vehicle parameters over time.

Circle and 8-shape paths are provided in the simulation environment. The user inputs the radius and velocity for either path option. The input to the state model is time, and the states are output by various state object accessor functions. The true states are sent to the sensor models. The sensors models apply errors using the methods described earlier to generate simulated measurements. These simulated measurements are input to the Kalman filter. The state model incorporates the states of the simplified UGV formulation.

5. 1. PATH 1: CIRCLE

This path takes the form of a circle, where the vehicle starts at the origin on a tangent to the circle. The body axis and the navigation frame are aligned at the start time. Initially, the vehicle is heading due North and then starts to turn Westward. The two inputs are the circle radius and a constant velocity.
The circular path in the navigation frame is shown in the following figure:

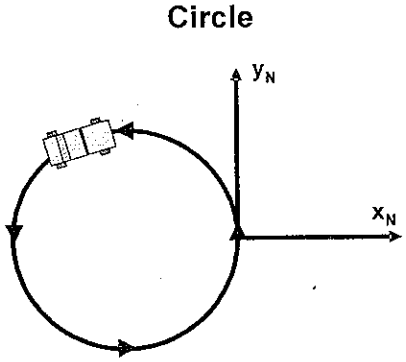

Fig. 4 Circular Path.

The period of the circular path is:

$$T = \frac{2 \cdot \pi \cdot R}{V} \qquad \text{Eq. 29}$$

The circle path simulation state equations are:

$$\psi = 2 \cdot \pi \cdot \frac{(t - (Lap - 1) \cdot T)}{T} \qquad \text{Eq. 30}$$

$$x = -R \cdot (1 - \cos(\psi)) \qquad \text{Eq. 31}$$

$$y = R \cdot \sin(\psi) \qquad \text{Eq. 32}$$

$$\dot{\beta} = \frac{V}{R} \qquad \text{Eq. 33}$$

$$\alpha = \tan^{-1}\left(\frac{L \cdot \dot{\beta}}{V}\right) \qquad \text{Eq. 34}$$

The heading angle formula automatically ensures that the heading angle is between 0 and 360 degrees.

5. 3. PATH 2. (8-SHAPE)

In the Figure-8 path, the vehicle alternates between counter clockwise and clockwise circles. Each circular path is considered a lap. The following figure shows the Figure-8 path:

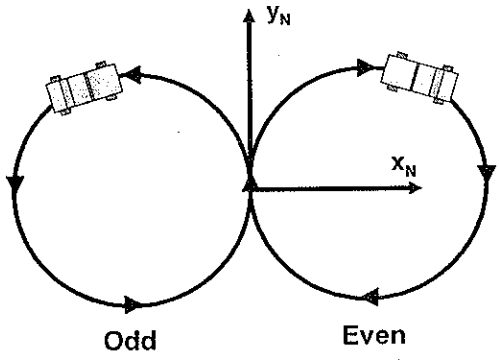

Fig. 5. ,8-shape Path.

The equations for the counter-clockwise circle are identical to the circular path described above. The states that differ for the clock-wise lap are:

$$\psi = -2 \cdot \pi \cdot \frac{(t - (Lap - 1) \cdot T)}{T} \qquad \text{Eq. 35}$$

$$x = R \cdot (1 - \cos(\psi)) \qquad \text{Eq. 36}$$

$$y = -R \cdot \sin(\psi) \qquad \text{Eq. 37}$$

$$\dot{\beta} = -\frac{V}{R} \qquad \text{Eq. 38}$$

The only difference between the clockwise and counter-clockwise state equations listed are sign changes. The steering angle equation remains unmodified.

5. 4. SIMULATION RESULTS

The results shown here were generated using the sensor parameters in Table 1.

Table 1. Sensor Parameters.

Sensor	Standard Deviation	Absolute/ Scale Factor	Bias	Measurement Frequency
GPS - Radius	5.1 m	Absolute	0	1 Hz
GPS bias	-	-	0 m	-
Encoder	10%	Scale Factor	0	10 Hz
Doppler	5%	Scale Factor	0	5 Hz
Compass	3 degrees	Absolute	0	20 Hz
AHRS Yaw	1 degree	Absolute	0	10 Hz
Steering Angle	0.5 degrees	Absolute	0	5 Hz

The UGV estimated path is shown in Fig. 6. The UGV travels along a figure-8 course with a radius of 100 m at 20 m/s for 2 minutes. Only slight variations in the position are observed.

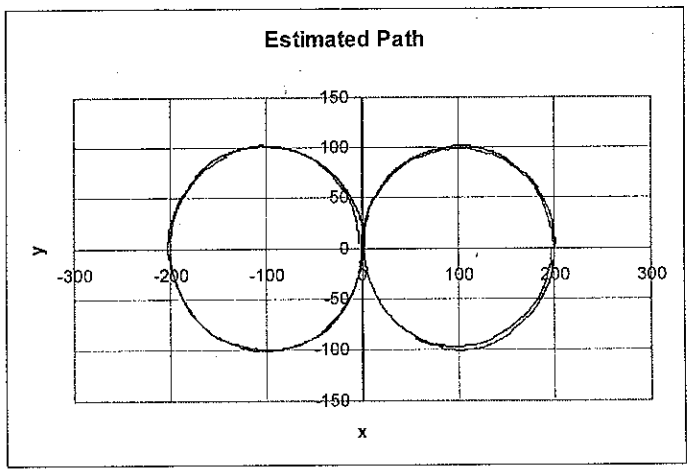

Fig. 6. Vehicle True and Estimated Positions.

The x and y position errors are shown in Fig. 7. The small errors are largely due to the small GPS errors shown by Schriever [2004]. Path following would be possible if these GPS accuracies are routinely available, suitable control laws are applied, and the coordinates of the path are known to suitable accuracy. Otherwise, different path following approaches are necessary.

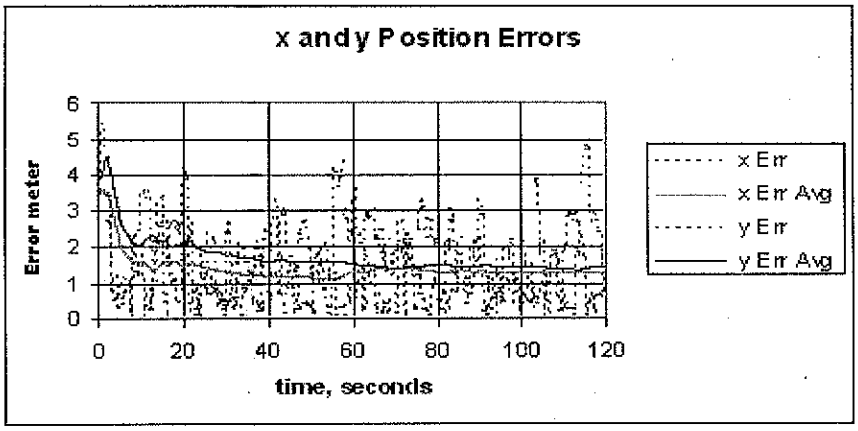

Fig. 7. Vehicle *x* and *y* Position Errors.

The vehicle velocity and velocity errors are shown in Fig. 8.

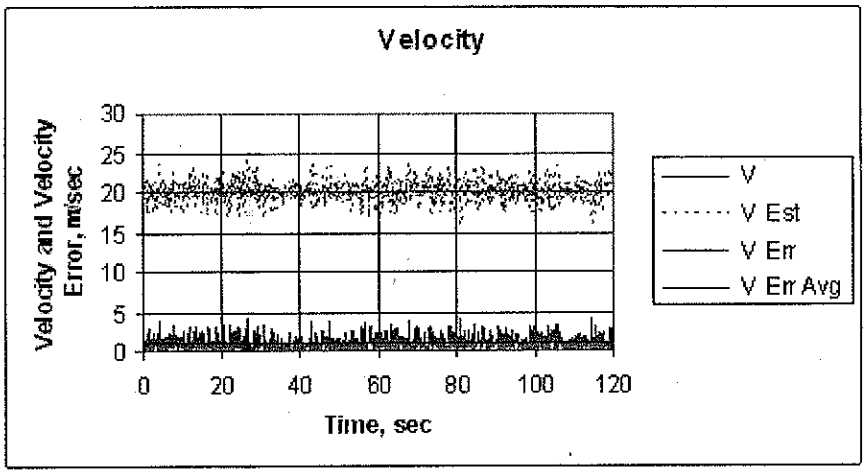

Fig. 8. Vehicle Velocity.

The vehicle yaw and yaw estimates are shown in Fig.9. The yaw errors are shown in Fig.10.

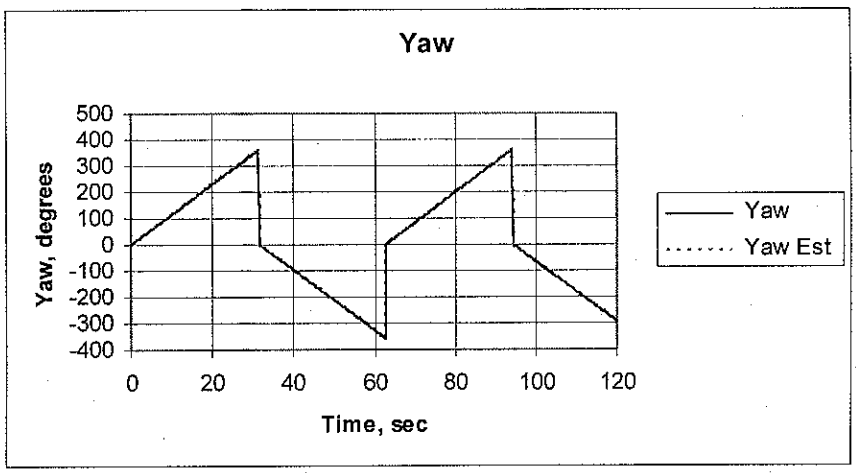

Fig. 9. Vehicle Yaw.

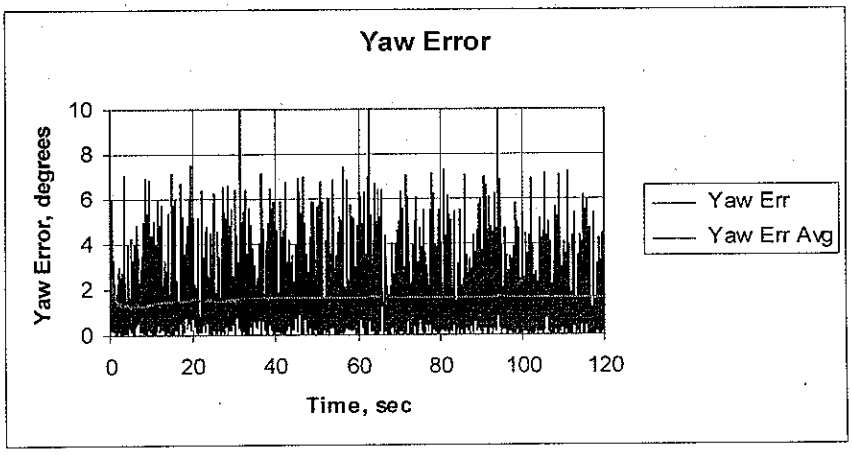

Fig. 10. Vehicle Yaw Error.

The vehicle turn rate and turn rate errors are shown in Fig. 11.

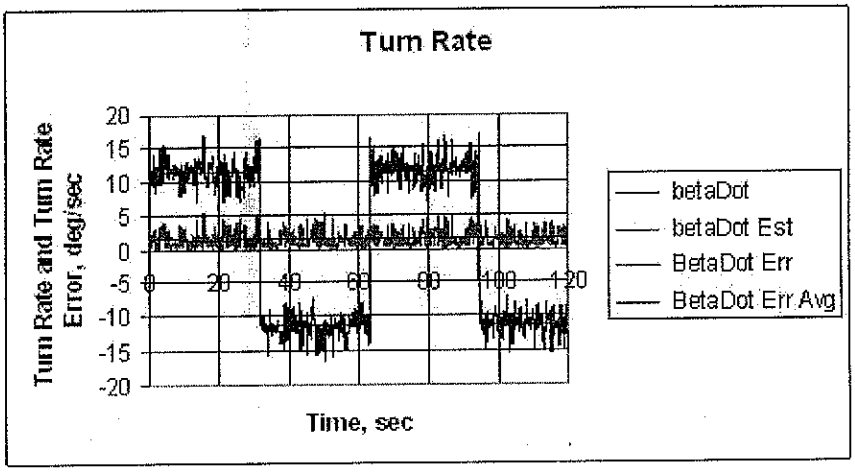

Fig. 11. Vehicle Turn Rate.

The GPS measurements are shown in Fig 12.

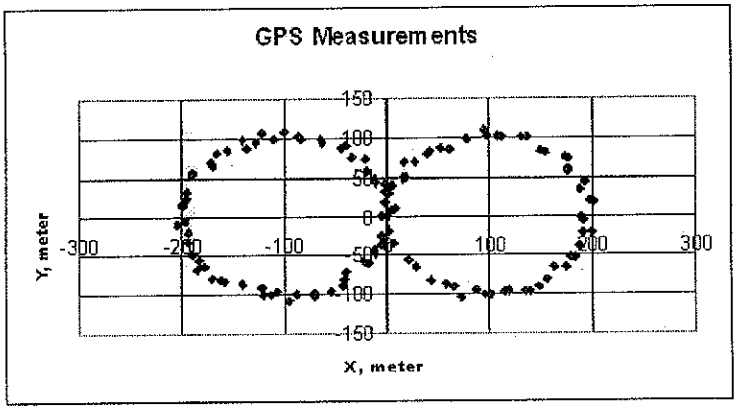

Fig. 12. GPS Measurements.

The velocity measurement simulations are shown in Fig. 13. As would be expected based upon the inputs, the encoder has noisier measurements than the Doppler.

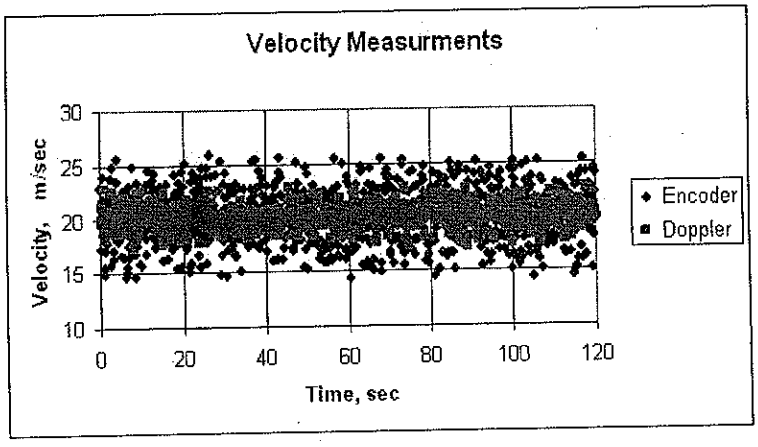

Fig. 13. Velocity Measurements.

The yaw measurements are shown in Fig 14. Whenever the yaw measurement exceeds ±360 degrees, the measurement is brought back within bounds. The majority of the zero measurement values represent no sensor output, as is the case for the steering angle measurement. Note that the vehicle makes alternating turns in the figure-8 course.

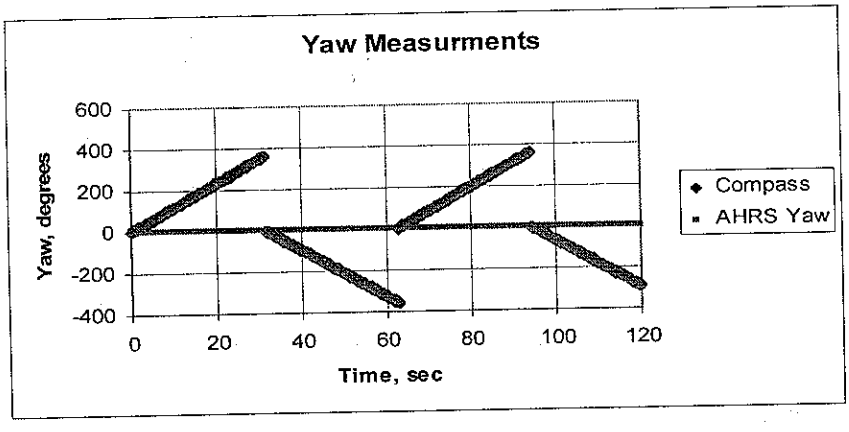

Fig. 14. Yaw Measurements.

The steering angle measurement simulation is shown in Fig. 15.

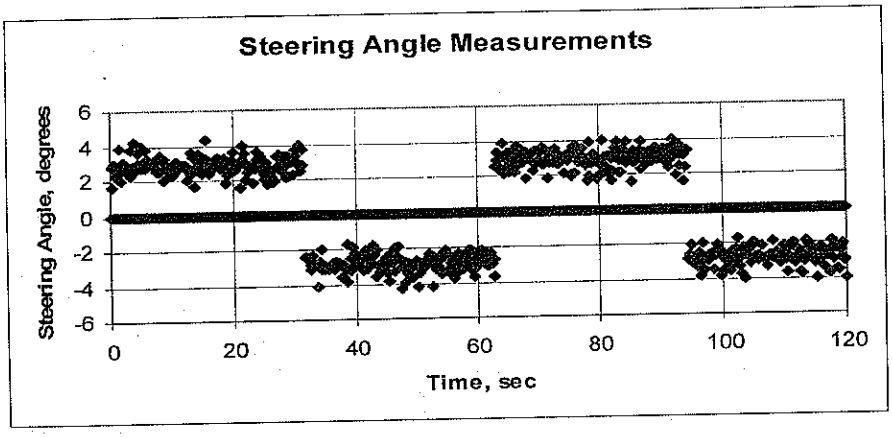

Fig. 15. Steering Angle Measurements.

The GPS accuracy plays a key role in the overall navigation accuracy. The discussion of the simulation thus far has focused on nominal GPS performance. Now the same simulation is run with degraded GPS performance. The GPS horizontal position standard deviation is modified from the previous 5.1 m to 20 m and 40 m. All other sensor parameters remain the same.

The UGV estimated x-y position, position errors, and GPS position measurements for the 40-m GPS standard deviation are shown in Fig.16, Fig. 17 and Fig.18 respectively.The navigation errors continue to degrade over the previous cases. Despite the significant errors in the GPS measurements, the integrated navigation solution shows impressive performance.

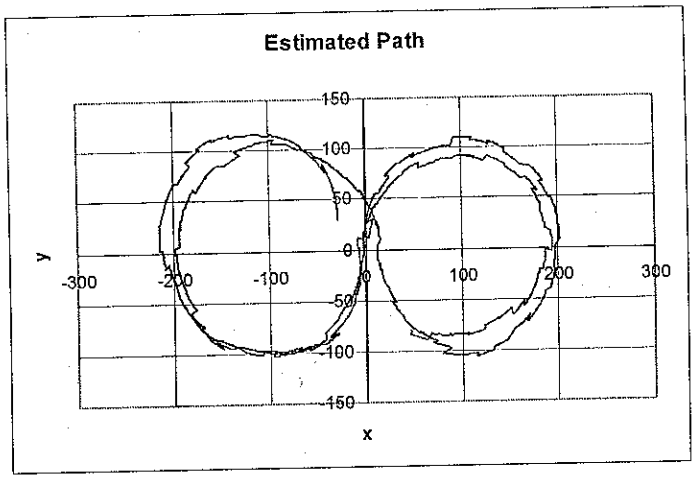

Fig. 16. Estimated Path for GPS Horizontal Error Standard Deviation of 40 m.

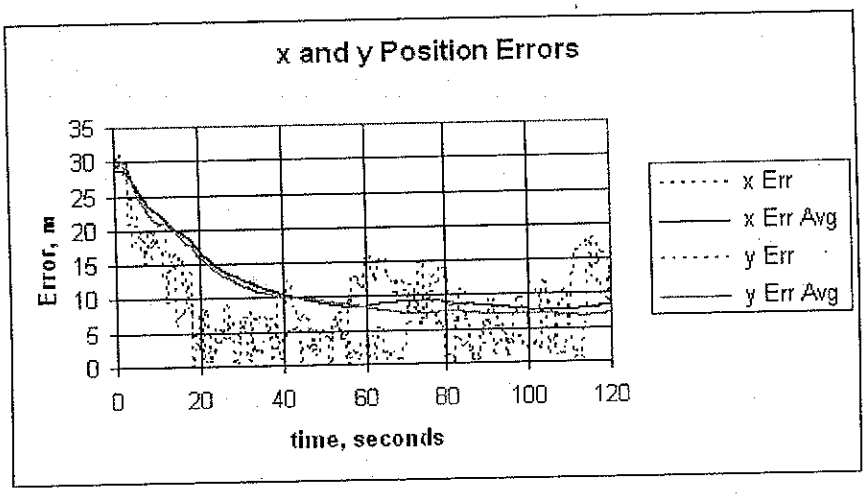

Fig. 17. Position Errors for GPS Horizontal Error Standard Deviation of 40 m.

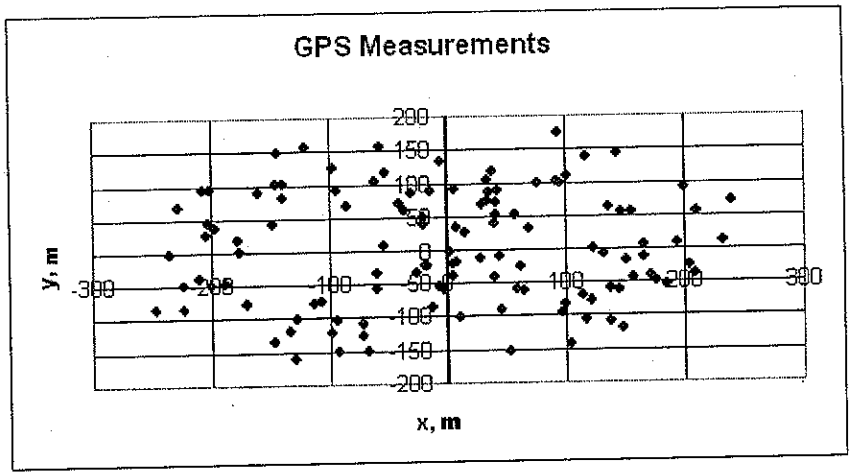

Fig. 18. GPS Position Measurements for GPS Horizontal Error Standard Deviation of 40 m.

6. CONCLUSIONS

The linear Kalman filter approach was successfully applied to the simplified navigation formulation. The Kalman filter was able to handle multiple, asynchronous inputs from noisy sensors.

The simulation results show that the navigation accuracies provided by the selected sensor suite is inappropriate for path following if the GPS accuracy is degraded. The sensor suite evaluated here must be supplemented by other sensors. Vision-based path following sensors and algorithms or other path-following approaches are necessary.

The navigation algorithm demonstrated the ability to estimate the position with degraded GPS performance. The Kalman filter effectively used the GPS to bound the position error drift.

REFERENCES

1. S.I. Roumeliotis, G.S. Sukhatme, and G.A. Bekey. "Smoother based 3-d attitude estimation for mobile robot localization". In Proceedings of the 1999 IEEE International Conference in Robotics and Automation, May 1999.
2. Magellan Meridian Gold datasheet, http://www.magellangps.com
3. P.S. Maybeck, "The Kalman filter: an introduction to concepts, Autonomous robot vehicles", Springer-Verlag New York, Inc., New York, 1990
4. [1] G. Welch, G. Bishop "An Introduction to the Kalman Filter" UNC-Chapel Hill, TR 95-041, July, 2006
5. S.I. Roumeliotis, G.S. Sukhatme, and G.A. Bekey, "Circumventing Dynamic Modeling: Evaluation of the Error-State Kalman Filter applied to Mobile Robot Localization". In Proc. 1999 IEEE International Conference on Robotics and Automation, Detroit, Michigan, 1999
6. A.J. Kelly, "A 3D State Space Formulation of a Navigation Kalman Filter for Autonomous Vehicles", Carnegie Mellon University, The Robotics Institute, Technical report CMU-RI-TR-94-19, 1994
7. K. Murphy, S. Legowik "GPS Aided Retro-traverse For Unmanned Ground Vehicles", Presented at SPIE 10th Annual Aero-Sense Symposium. Conference 2738, Navigation and Control Technologies for Unmanned Systems. Orlando, FL. April 1996.
8. M. O'Connor, T. Bell, G. Elkaim, and Dr. Bradford Parkinson, "Automatic Steering of Farm Vehicles Using GPS" In Proceedings of the International Conference on Precision Agriculture, 1996

9. S. G. Chroust, M. Vincze "Fusion of Vision and Inertial Data for Motion and Structure Estimation", Journal of Robotic Systems, 2004.
10. L.B. Cremean "An Information-Rich Autonomous Vehicle for High-Speed Desert Navigation" Journal of field robotics, June 2006. http://gc.caltech.edu/media/papers/teamcaltech-jfr05.pdf
11. T. Hong, M. Abrams, T. Chang, and M.O. Shneier, "An Intelligent World Model for Autonomous Off-Road Driving," Computer Vision and Image Understanding, 2000. http://citeseer.ist.psu.edu/hong00intelligent.html
12. O.A. Basir, H.C. Shen, "Interdependence and Information Loss in Multi-Sensor Systems ", Journal of Robotic Systems, 1999