ROBOT NAVIGATION USING MULTIPLE SENSOR FUSION OF STOCHASTIC AND DETERMINISTIC ESTIMATION

I. A. Ashokaraj, A. Tsourdos, P. M. G. Silson and B. A. White

Department of Aerospace, Power and Sensors, Cranfield University,
Defence Academy of the United Kingdom,
Shrivenham, SN6 8LA, United Kingdom

Abstract. This paper describes a multiple sensor fusion approach in which a sensor based navigation scheme needs to fuse a stochastic robot position estimate from an extended Kalman Filter (EKF) with a deterministic robot position estimate from an Interval Analysis (IA) algorithm.

Keywords. Robot Navigation, Sensor Fusion, Interval Analysis, Box-Muller Transform

1. INTRODUCTION

Robot navigation is primarily about guiding a mobile robot to a desired destination or along a pre-specified path in which the robots environment consists of landmarks and obstacles. In order to achieve this objective the robot needs to be equipped with sensors suitable to localize the robot throughout the path it has to follow. Most of these sensors may give overlapping or complementary information and sometimes be redundant as well. There are many different architectures to fuse these information. Mobile robots generally carry dead reckoning sensors such as wheel encoders, inertial sensors (INS), such as accelerometers, gyroscopes, to measure acceleration and angle rate respectively, and landmark and obstacle detecting and map making sensors such as time of flight (TOF) ultrasonic sensors. All these sensors measurements can be fused to estimate the robots position by using a sensor fusion algorithm. Sensor fusion in this case is the method of integrating data from distinctly different sensors to estimate the robots position. Classical data fusion algorithms use stochastic filters such as Kalman Filters for robot position estimation (Smith et al. 1990). But one of the main disadvantages of using Kalman Filters with ultrasonic sensors for robot localisation problems is that the data association step in Kalman Filters is complex and also the fact that they are often affected by bias and drift from inertial sensors. Moreover an accurate model of the robot system and accurate statistics of the sensor noises are needed, which is not available accurately in many cases. Therefore in this case the EKF was used only for the inertial sensors and encoders, while in the case of the ultrasonic measurements, an Interval Analysis based deterministic algorithm was used to obtain an interval robot position estimate. Thus there is a need to fuse two independent robot position estimates which have fundamentally different representations of the uncertainty and this problem is addressed in this paper.

2. ROBOT LOCALISATION USING EKF WITH INERTIAL SENSORS AND ENCODERS

By fusing the measurement data from the sensors - wheel encoders, gyroscopes and accelerometers - in the mobile robot, a reliable estimation of the position and heading of the robot can be obtained. For the extended Kalman
Filter, the robot model equations can be rewritten as the state equation. All the process noises are assumed to be zero mean, uncorrelated white random noises only. The state vector of the discrete extended Kalman Filter in this case consists of

$$
x_k = \begin{bmatrix} p_k & \theta_k & U_k & i_k & X_k & Y_k \end{bmatrix}
$$

In this case the nonlinear state transition equations $F_k$ for the extended Kalman Filter is assumed to be

$$
p_{k+1} = (1 - aT_s)p_k + T_sV_2
$$

$$
\theta_{k+1} = \theta_k + \frac{T_sG_r}{M}i_k
$$

$$
U_{k+1} = \left(1 - \frac{B_uT_s}{M}\right)U_k + \left(\frac{T_sG_r}{M}\right)i_k
$$

$$
i_{k+1} = \left(1 - \frac{RT_s}{L}\right)i_k + \left(\frac{T_s}{L}\right)U_k
$$

$$
X_{k+1} = X_k + U_k \cos(\theta_k)T_s
$$

$$
Y_{k+1} = Y_k + U_k \sin(\theta_k)T_s
$$

In this case we know that the EKF receives measurements from the accelerometer, gyroscope and encoders and they are as follows:

1. velocity measurements from the wheel encoders,
2. acceleration from the accelerometers, which when integrated gives the velocity of the robot,
3. robot heading angle measurement from the encoders and
4. the angular velocity measurements from the rate gyroscope which when integrated once gives the heading angle of the robot.

Therefore the innovation equation for the extended Kalman Filter which receives the four measurements as observations is

$$
\nu_k = \begin{bmatrix} \hat{Z}_{k, \text{encoder}} & -\hat{\theta}_k \\
\hat{Z}_{k, \text{gyroscope}} & -\hat{\theta}_k \\
\hat{Z}_{k, \text{encoder}} & -\hat{U}_k \\
\hat{Z}_{k, \text{accelerometer}} & -\hat{U}_k \end{bmatrix}
$$

For both the velocity and heading angle of the robot there are two sets of measurements from two independent sensors as observations to the EKF and the EKF then estimates the best velocity and heading angle from which the robot's position is calculated. The calculated position of the robot and the heading angle estimate from the EKF are shown in figures 6 and 7.

---

**Algorithm for the measurement model**

<table>
<thead>
<tr>
<th>Algorithm $[d_{m}]$ (in: $[p]$; out: $[d_{m}]$)</th>
</tr>
</thead>
</table>
| 1. $[r_\ast] := \det(ab, \bar{a}, \bar{b}) \geq 0$
| 2. $[r_\ast] := \begin{cases} \det(ab, \bar{a}, \bar{b}) \geq 0 & \text{if } [r_\ast] = 0 \text{ then } [r_\ast] := -\infty; \\
| 3. $[r_\ast] := \det(u_1, \bar{a}, \bar{b}) \geq 0 \wedge (\det(u_2, \bar{a}, \bar{b}) \leq 0)$
| 4. $[r_\ast] := \det(u_1, \bar{a}, \bar{b}) \geq 0 \wedge (\det(u_2, \bar{a}, \bar{b}) \leq 0)$
| 5. $[r_\ast] := \det(u_1, \bar{a}, \bar{b}) \geq 0 \wedge (\det(u_2, \bar{a}, \bar{b}) \leq 0)$
| 6. $[r_\ast] := \det(u_1, \bar{a}, \bar{b}) \geq 0 \wedge (\det(u_2, \bar{a}, \bar{b}) \leq 0)$
| 7. $[r_\ast] := \det(u_1, \bar{a}, \bar{b}) \geq 0 \wedge (\det(u_2, \bar{a}, \bar{b}) \leq 0)$
| 8. $[r_\ast] := \det(u_1, \bar{a}, \bar{b}) \geq 0 \wedge (\det(u_2, \bar{a}, \bar{b}) \leq 0)$
| 9. $[r_\ast] := \det(u_1, \bar{a}, \bar{b}) \geq 0 \wedge (\det(u_2, \bar{a}, \bar{b}) \leq 0)$
| 10. $[r_\ast] := \det(u_1, \bar{a}, \bar{b}) \geq 0 \wedge (\det(u_2, \bar{a}, \bar{b}) \leq 0)$

---

**Algorithm for the remoteness of the cone from a segment**

**3. ROBOT LOCALISATION WITH INTERVAL ANALYSIS USING RANGE MEASUREMENTS.**

A brief overview of the algorithm for a single measurement process has been given in the figures 1 and 2. The figure 1 is basically an interval inclusion function (i.e.) a mathematical model which models all the possible distance measurements expected from all the ultrasonic sensors when the configuration is $[p]$, where $a_j,b_j$ are the two extreme points of a segment of the land marks as shown in figure 3. The figure 2 is the interval mathematical model of a single ultrasonic sensor distance measurement. Its a simple sensor model with interval values which models the distance measurement as the smallest distance between the sensor vertex “$s_i$” (as shown in figure 3) and the segment of a land mark $[a,b]$, in four simple scenarios which is described in detail in (Jaulin et al. 2001). In this case it was assumed that the range of the ultrasonic sensors is limited and this was used as given in (Ashokara et al. 2004).

This problem is solved using any of the two approaches namely SIVIA (Set Inversion Via Interval Analysis) (Jaulin et al. 2001) and ImageSP. A brief introduction to both
4. FUSION ARCHITECTURE FOR IA AND KF

Sensor fusion is a very important and keenly researched topic in the domain of mobile robotics. This is due to the fact that, instead of using bespoke expensive sensors for estimating the robots position, multiple low cost sensors can be used, thereby reducing the cost of developing the robot. Also the source of errors in one sensor may be different from another one and this fact can be exploited to eliminate the errors in the measurements.

In section 2, the fusion of inertial sensor information with odometry using a centralised architecture with extended Kalman Filter to obtain an estimation of the robots position is given and in section 3 an independent guaranteed estimation of the robots position using ultrasonic sensors with Interval Analysis is described. Thus we have two independent estimates of the robots position with their associated measure of uncertainty. The question now which needs to be answered is how to fuse these two fundamentally different representations of the robot position with their associated uncertainty and which architecture is the best suited towards this purpose. We know that the INS and odometry position estimation is corrupted by bias, drift etc even though the measurements are available more frequently than the ultrasonic sensor position estimate which is guaranteed to contain the true robot position. So the ultrasonic sensor robot position estimate when ever available is fused with INS position estimate using a nonlinear Kalman Filter to minimise and correct the errors in the INS position estimate. It can be termed as a hierarchical architecture as well because the two estimates are got from two groups of sensors which can then be fused together as well. Essentially two independent robot position estimates are obtained, one of which gives estimates in high frequency (the INS estimates at 100 measurements per second) and the other which is from the ultrasonic sensors gives the position estimates in low frequency (only one measurement every second). The ultrasonic sensors gives estimates only once every second because of the computation involved in processing these measurements.

5. APPROXIMATING INTERVALS TO A NUMBER WITH GAUSSIAN DISTRIBUTION

In the robot position estimate obtained using the INS and odometric sensors with nonlinear Kalman Filters, the uncertainty in the robot position is represented in terms of covariance by a real number and it is assumed to have a Gaussian distribution. Therefore when this INS position estimate is given as a measurement to the nonlinear Kalman Filter, it will be easy to implement. However in the case of the position estimate using ultrasonic sensors with Interval Analysis algorithm, the robot position is in terms of an interval, which is guaranteed to contain the true robot position. But the exact position of the robot is not known and so it has equal probability to be anywhere within that interval. This implies that the interval robot position estimate basically has an uniform distribution. So the research focus of the work described in this paper now is the study of the best approach to fuse a robot position estimate with Gaussian distribution with another robot position estimate with uniform distribution. Since fusing this as such is not possible, there is a need to approximate either one of them to the other one so that both the measurements have the same form of representation of uncertainty which will make the task of fusing them easier.

In this case it was concluded that converting the uniform distribution to Gaussian distribution is better than the other way around because of the following reasons.

- Firstly as shown in the figure 4, the Gaussian distribution extends to infinity and therefore approximating a Gaussian distribution to uniform distribution may lead to errors. However when the uniform distribution is approximated as Gaussian distribution it will just become conservative as shown in figure 4.
- Secondly even in the case that the measurements are converted to intervals, i.e. both the measurements have uniform distribution, when both the measurements need to fused, it needs to have a simple intersection. Incase if there is an empty intersection, the addition of both intervals will give rise to another interval which has an uncertainty

Fig. 3. Robot and Sensor model

SIVIA and IMAGESP is given in the next two subsections.
greater than either one of them. This is not desired because the purpose of fusing measurements is to have an accurate measurement.

- Also when both the measurements have Gaussian distribution then Kalman Filters can be used for fusing the measurements.

Therefore the intervals are approximated into a Gaussian distribution with a mean and variance. This approximation can also be done in a few ways.

One way of approximating the intervals to compute the covariance values is to choose the mid point of the interval as the mean and then compute the variance using the extremes of the intervals. This may look very straightforward way of computing the variance values and they can be diagrammatically represented as in the Figure 5. However since the intervals have uniform distribution it is not possible to take the mean and variance of intervals and use them in the Kalman Filters which need to have a Gaussian distribution. Therefore the uniform distribution should be approximated into a Gaussian distribution. There are a few ways to do this in the random number generation theory, where the uniformly distributed random numbers are converted to random numbers with a particular distribution. In the case of Gaussian also called the normal distribution, which have a mean of 0 and variance of 1, there are a few techniques such as Box-Muller transform, Von Neumann’s rejection method to convert random numbers with uniform distribution to random numbers with normal distribution. A Box-Muller transform (Box and Muller 1958) is a method of generating pairs of independent normally distributed random numbers, given a source of uniformly distributed random numbers. Since in this case only the extremes of the intervals are known, i.e. a pair of uniformly distributed numbers the Box-Muller transform was used.

The Box-Muller algorithm generates a pair of independent exactly $N(0,1)$ distributed random numbers $(z_1, z_2)$ using a pair $(u_1, u_2)$ of uniformly distributed random numbers $\in [0,1]^2$ according to:

$$z_1 = \sqrt{-2 \log u_1 \cos(2\pi u_2)} \quad (4)$$
$$z_2 = \sqrt{-2 \log u_1 \sin(2\pi u_2)} \quad (5)$$

This is the polar form of the Box-Muller algorithm. An efficient version of this transformation is given by the following acceptance rejection algorithm:

Set

$$V_1 = 2u_1 - 1, \quad V_2 = 2u_2 - 1, \quad W = V_1^2 + V_2^2. \quad (7)$$

If $(W < 1)$,

Set

$$Y = \frac{\sqrt{-2 \log W}}{W} \quad (8)$$
$$z_1 = \mu + \sigma V_1 Y, \quad (9)$$
$$z_2 = \mu + \sigma V_2 Y. \quad (10)$$

else goto equation 7.

where, $\mu$ and $\sigma$ are the mean and the standard distribution.

Thus the Box-Muller algorithm is actually a transformation which maps a pair of random numbers with uniform distribution to a pair of random numbers with normal distribution. So using this algorithm the intervals are converted to a number with normal distribution. One of this pair of numbers is then given as the measurement with the appropriate variance calculated using the uniform distribution to the extended Kalman Filter. Since in this case the $x, y, \theta$ are the outputs from the Interval algorithm, they are given as measurements to the EKF and the implementation details of this extended Kalman Filters are given in section 6.

6. IMPLEMENTATION DETAILS OF THE EKF FOR POSITION FUSION

In this section the implementation details in the extended Kalman Filter is described. The same extended Kalman Filter state equations given in equation 2 were used here, as the variables obtained from the ultrasonic sensor measurements are already present as states in the state equations. Now the only requirement for the
implementation of the EKF for the fusion of the position and heading angle measurements are to change the measurement update equations by changing the $H$ matrix and the innovation vector in the nonlinear Kalman Filter equations.

The innovation vector in equation 3 when the measurements from the ultrasonic sensor become available will become as given in equation 11.

\[ \nu_k = \begin{bmatrix} \hat{\theta}_{\text{encoder}} \\ \hat{\theta}_{\text{gyroscope}} \\ \frac{\hat{p}_k U_k}{\text{Len}} \\ \hat{U}_{\text{encoder}} \\ \hat{U}_{\text{accelerometer}} \\ \hat{X}_k \hat{\text{intY}} \\ \hat{Y}_k \hat{\text{intX}} \end{bmatrix} \]  \tag{11}

Also for the EKF, the jacobian matrix $\nabla H_{k_x}$ will be as given in equation 12.

\[ \nabla H_{k_x} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{p}{\text{Len}} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -U \sin(\theta) \cos(\theta) & 0 & 1 & 0 \\ 0 & U \cos(\theta) \sin(\theta) & 0 & 0 & 1 \end{bmatrix} \]  \tag{12}

Then the EKF fusion will need to proceed recursively with the position and heading angle obtained using IA and then transformed using the Box-Muller transform updated once every second (due to computation time), where as the position from inertial sensors and encoders are updated 100 times per second. However if the measurements from IA are given only once every second, they may not influence the EKF estimates and so the transformed estimates are propagated using the EKF model and given as measurements to the EKF until a new measurement is obtained from the IA algorithm.

Figure 6 shows the robot position estimate using EKF only, which is affected by sensor bias, drift etc. Similarly in figure 7 the heading angle of the robot using the INS and encoders which is affected by bias and drift is shown. The figure 6 and figure 6 shows the robot position and heading angle respectively after using the transformed interval position and heading angle estimate from the SIVIA interval robot position. Similarly the figures 6 and 6 shows the EKF position and heading angle estimates after using the transformed IMAGESP interval position and heading angle estimates as measurements in the EKF. In all the cases it can be observed that the EKF estimate with the transformed interval estimates as measurements is much better than the EKF position and heading angle estimate from the INS and encoders. It should be noted however that the SIVIA interval position uncertainty is greater than that of the IMAGESP algorithm. The reduction in uncertainty of the estimated interval robot position using the IMAGESP algorithm is attained at the cost of more computation time when compared with the SIVIA algorithm. This is because, in the IMAGESP algorithm, the whole initial subpaving is divided into many boxes of identical width less than or equal to $\epsilon$ (0.01 in our case) and the global test $t(p)$ is performed on all of these boxes.

7. CONCLUSION

A method to fuse a deterministic robot position estimate obtained using an Interval Analysis algorithm with a stochastic robot position estimate obtained using EKF is described. The EKF uses accelerometers, gyroscopes and encoders to measure the robot speed and heading angle, so that the robot position can be estimated. But the EKF robot position estimate is affected by er-
Fig. 8. Fused robot position estimate using the interval robot position from SIVIA with limited range sensors, transformed to a real number with Gaussian distribution by the BOX-Muller transform, as measurement to the EKF.

Fig. 9. Fused robot heading angle estimate using the interval heading angle from SIVIA with limited range sensors, transformed to a real number with Gaussian distribution by the BOX-Muller transform, as measurement to the EKF.

errors in robot model, sensor bias, drift etc. The Interval Analysis is a deterministic approach to estimating the robots position without using a model of the robot system, thereby minimizing errors due to inaccurate robot model. The interval robot position estimate which has a uniform distribution is then transformed using the Box-Muller transform to a number with a mean and variance which has a Gaussian distribution. This is then given as a measurement to the EKF. It can be seen from the results that the transformed interval robot position and heading angle estimates, limits and bounds the errors in the INS and encoder robot position estimates.

8. REFERENCES


