Proceedings of the 2011 IEEE
International Conference on Mechatronics
April 13-15, 2011, Istanbul, Turkey

The H_∞ FastSLAM Framework

Ramazan Havangi, Mohammad Ali Nekoui, Hamid Taghirad, Mohammad Teshnehlab

Control Department K. N. Toosi University of Technology
Seyed khandan bridge, Tehran, Iran
Havangi@ee.kntu.ac.ir

***Abstract*—FastSLAM is a framework using a Rao-Blackwellized particle filter. However, the performance of FastSLAM depends on correct a priori knowledge of the process and measurement noise covariance matrices (Q_t and R_t) that are in most applications unknown. On the other hand, an incorrect a priori knowledge of Q_t and R_t may seriously degrade the performance of FastSLAM. To solve these problems, this paper presents H_∞ FastSLAM. In this approach, H_∞ particle filter is used for the mobile robot position estimation and H_∞ filter is used for the feature location's estimation. The H_∞ FastSLAM can work in an unknown statistical noise behavior and thus it is more robust. Experimental results demonstrate the effectiveness of the proposed algorithm.**

***Keywords*— SLAM, Mobil robot, Particle filter, H_∞ Filter**

I. INTRODUCTION

The simultaneous localization and mapping (SLAM) is a fundamental problem of robots to perform autonomous takes such as exploration in an unknown environment. It represents an important role in the autonomy of a mobile robot. The two key computational solution to the SLAM are extended kalman filter (EKF-SLAM) and Rao-Blackwellized particle filter (FastSLAM). The EKF-SLAM approach is the most popular approach to solve the SLAM. Until now extensive research works have been reported employing EKF to the SLAM problem [5], [19-20]. Several applications of EKF-SLAM have been developed for indoor applications [3], [19], outdoor applications [1], underwater applications [24] and underground applications [16]. However, EKF-SLAM suffers from two major problems: the computational complexity and data association[18]. Recently, FastSLAM algorithm approach has been proposed as an alternative approach to solve the SLAM problem. FastSLAM is an instance of Rao-Blackwellized particle filter, which partitions the SLAM posterior into a localization problem and an independent landmark position estimation problem. There exist two versions of FastSLAM: FastSLAM1.0 and FastSLAM2.0. As FasSLAM2.0 is superior to FastSLAM1.0, this paper addresses FasSLAM2.0. In FastSLAM2.0, extended kalman particle filter is used for the mobile robot position estimation and EKF is used for the feature location's estimation. The key feature of FastSLAM, unlike EKF-SLAM, is the fact that data association decisions can be determined on a per-particle basis, and hence different particles can be associated with different landmarks. Each particle in FastSLAM may even have a different number of landmarks in its respective map. This characteristic gives the FastSLAM the possibility of dealing with multi-hypothesis association problem. The ability to simultaneously pursue multiple data associations makes FastSLAM significantly more robust to data association problems than algorithms based on incremental maximum likelihood data association such as EKF-SLAM. The other advantage of FastSLAM over EKF-SLAM arises from the fact that particle filters can cope with nonlinear and non-Gaussian robot motion models, whereas EKF approaches approximate such models via linear functions. There have been many investigations on FastSLAM [8], [9], [12], [13], [17], [21]. In references [2], [4], [10], [12], [13] it has been noted that FastSLAM degenerates over time. This degeneracy is due to the fact that a particle set estimating the pose of the robot loses its diversity. One of the main reasons for losing particle diversity in FastSLAM is sample impoverishment. It occurs when likelihood lies in the tail of the proposal distribution [7]. Researchers have been trying to solve those problems in [7], [8], [9], [21], [25-29]. In all previous research on FastSLAM, it is assumed that a priori knowledge of the process and measurement noise statistics is completely known. However, in most application these matrixes are unknown. On the other hand, an incorrect a prior knowledge of Q_t and R_t may seriously degrade the Kalman filter performance [6], [14. In this paper to solve these problems, H_∞ FastSLAM is proposed.

II. THE SLAM PROBLEM

The goal of SLAM is to simultaneously localize a robot and determine an accurate map of the environment. To describe SLAM, let us denote the map by Θ and the pose of the robot at time t by s_t. The map consists of a collection of features, each of which will be denoted by θ_n and the total number of stationary features will be denoted by N . In this situation, the SLAM problem can be formulized in a Bayesian probabilistic framework by representing each of the robot's position and map location as a probabilistic density function as:

$$p(s_t, \Theta \mid z^t, u^t, n^t) \quad (1)$$

978-1-61284-985-0/11/$26.00 ©2011 IEEE

In essence, it is necessary to estimate the posterior density of maps Θ and poses s_t given that we know the observation $z^t = \{z_1,...,z_t\}$, the control input $u^t = \{u_1,...,u_t\}$ and the data association n^t. FastSLAM is an efficient algorithm for the SLAM problem that is based on a straightforward factorization as follows [12-13]:

$$p(s^t,\Theta \mid z^t,u^t,n^t) = p(s^t \mid z^t,u^t,n^t)\prod_{n=1}^{N} p(\theta_n \mid s^t,z^t,u^t,n^t) \quad (2)$$

where $s^t = \{s_1,...,s_t\}$ is a robot path. This factorization states that the SLAM problem can be decomposed into estimating the product of a posterior over robot path and N landmark posteriors given the knowledge of the robot path. The FastSLAM algorithm implements the path estimator $p(s^t \mid z^t,u^t,n^t)$ using a particle filter and the landmarks pose $p(\theta_n \mid s^t,z^t,u^t,n^t)$ are realized by EKF, using separate filters for different landmarks. The Structure of the M particles is as follow [12], 17]:

$$s_t^{[m]} = < s^{t,[m]}, \mu_{1,t}^{[m]}, \Sigma_{1,t}^{[m]}, ..., \mu_{N,t}^{[m]}, \Sigma_{N,t}^{[m]} > \quad (3)$$

Where $[m]$ indicates the index of the particle, and $s^{t,[m]}$ is the m th particle's path estimate, and $\mu_{N,t}^{[m]}, \Sigma_{N,t}^{[m]}$ are the mean and the covariance of the Gaussian distribution representing the n th feature location conditioned on the path $s^{t,[m]}$. In general, it is not possible to draw samples directly from the SLAM posterior. Instead, the samples are drawn from a simpler distribution called the proposal distribution $q(s^{t,[m]} \mid z^t,u^t,n^t)$. The choice of the proposal distribution $q(s_t^{[m]} \mid s^{t-1,[m]},z^t,u^t,n^t)$ is one of the most critical issues in the design of a FastSLAM. In FastSLAM1.0, new poses are sampled using the transitional prior [11]:

$$q(s^{t,[m]} \mid z^{t-1},u^t,n^{t-1}) = p(s_t \mid u_t, s_{t-1}^{[m]}) \quad (4)$$

However, FastSLAM1.0 approach is particularly troublesome if the observation is too accurate relative to the vehicle' motion noise. To solve this problem, Montemerlo proposed an improved version called FastSLAM2.0 [13]. In FastSLAM2.0, vehicle poses are sampled under consideration of both the control u_t and measurement z_t, which is denoted as follow:

$$q(s^{t,[m]} \mid z^{t-1},u^t,n^{t-1}) = p(s^{t,[m]} \mid z^{t-1},u^t,n^{t-1}) \quad (5)$$

As a result, the fastSLAM2.0 is superior to FastSLAM1.0 in all aspects. In FastSLAM2.0, the importance weight is given by following equation:

$$w_k^{[m]} = \eta w_{k-1}^{[m]} \frac{p(z_t \mid s^{t,[m]},z^{t-1},u^t,n^t)\,p(s_t^{[m]} \mid z^{t-1},u^t,n^t)}{p(s_t^{[m]} \mid s^{t-1,[m]},z^t,u^t,n^t)} \quad (6)$$

III. THE H_∞ FILTER

The kalman filter minimizes the variance of estimation error. The optimality of kalman filter depends on the knowledge of the state space model noise. For providing optimal performance in minimum mean square error (MMSE) sense, this filter requires both process model noise and measurement model noise process to be Gaussian. If the noise statistics are unknown, then kalman filter is no longer optimal. Unlike the kalman filter algorithm which gives the MMSE estimate of e_k, the H_∞ filtering algorithm gives the optimal estimate of e_k such that the effect of the worst disturbance on the estimation is minimized. To describe H_∞ filter, consider the following time variant state space model:

$$\begin{aligned} x_{k+1} &= F_k x_k + G_k w_k \\ y_k &= H_k x_k + v_k \end{aligned} \quad (7)$$

where F_k, G_k, and H_k are know matrix with appropriate dimensions. The process noise w_k and the measurement noise v_k are assumed to be energy bonded l_2 signals whose statistical properties are unknown, i.e.,

$$\|w_k\|_2^2 = \sum_{k=0}^{\infty} w_k^T w_k < \infty \quad \|v_k\|_2^2 = \sum_{k=0}^{\infty} v_k^T v_k < \infty \quad (8)$$

The optimum H_∞ coast is given as follow [30-34]:

$$J = \frac{\sum_{k=0}^{N} \|y_k - \hat{y}_k\|_2^2}{\|x_0 - \hat{x}_0\|_{P_0^{-1}}^2 + \sum_{k=0}^{N} \|w_k\|_{Q_k^{-1}}^2 + \sum_{k=0}^{N} \|v_k\|_{R_k^{-1}}^2} < \gamma \quad (9)$$

Where P_0, Q_k, R_k are the weighting matrices for the initial condition, the process noise and the measurement noise. Moreover, $P_0 > 0$, $Q_k > 0$ and $R_k > 0$. The notation $\|x_k\|_{Q_k}^2$ is defined as $\|x_k\|_{Q_k}^2 = x_k^T Q_k x_k$ .The denominator of J can be considered as the energy of the unknown disturbance, and the numerator is the energy the of the estimation error. The objective of H_∞ estimation is to minimize the maximum value of J as following

$$\min_{\hat{x}_k} \max_{v_k, w_k, x_0} J < \gamma \quad (10)$$

The attenuation factor $\gamma > 0$ is the maximum attention value specified by the user. Therefore, the H_∞ filter aims to provide a uniformly small estimation error $e_k = y_k - \hat{y}_k$ for any $w_k, v_k \in l_2$ and $x_0 \in R^n$, such that the energy gain J is bonded by a prescribed value. The solution for H_∞ optimization can be obtained by augmenting the state constraint to cost function in (9) using a set of lagrange multipliers and performing the min and max operations with respect to the state space model variables. For a given $\lambda > 0$, one possible level γ, H_∞ filter as following equation[30-34]:

$$\begin{aligned} \hat{x}_{k+1}^- &= F_k x_k \\ P_{k+1}^- &= F_k P_k F_k^T + G_k Q_k G_k^T \end{aligned} \quad (11)$$

Where

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1} \quad (12)$$

$$\hat{x}_k = \hat{x}_k^- + K_k (y_k - H_k \hat{x}_k^-) \quad (13)$$

$$P_k = P_k^- - P_k^- \begin{bmatrix} H_k^T & I \end{bmatrix} R_{e,k}^{-1} \begin{bmatrix} H_k \\ I \end{bmatrix} P_k^- \quad (14)$$

$$R_{e,k} = \begin{bmatrix} R_k & 0 \\ 0 & -\gamma I \end{bmatrix} + \begin{bmatrix} H_k \\ I \end{bmatrix} P_k^- \begin{bmatrix} H_k^T & I \end{bmatrix} \quad (15)$$

IV. THE H_∞ FASTSLAM

In this section, H_∞ FastSLAM describe in detail. The algorithm update of posterior of H_∞ FastSLAM can be described as:

1. Sampling Strategy in H_∞ FastSLAM
2. Landmark estimation based on H_∞ filter
4. Feature Initialization
3. Calculate importance weight
5. Resampling

A. *Sampling Strategy in H_∞ FastSLAM*

The FastSLAM relies on importance sampling, so it requires the design of proposal distributions that can approximate the true posterior reasonably well. The most common strategy is to sample from the transition motion. However, this strategy can fail if the most measurements appear in the tail of the proposal distribution, or if the likelihood is too sharp in comparison to the proposal distribution. In this case, most of weights of participles are insignificant and samples impoverishment occurs. Several researchers have introduced the most current observations into the proposal distribution and have used some heuristic techniques to improve the accuracy of the proposal distribution [8], [9], [11], [13], [16], [17], [27]. However, the performance of the methods and the quality of the estimation depends on the correct a priori knowledge of process Q_t and measurement noise covariance matrices R_t . In proposed method, to solve this problem extended H_∞ filter instead of EKF is used (H_∞ FastSLAM). In H_∞ FastSLAM, such as FastSLAM 2.0, poses are sampled under consideration of both the motion u_t and the measurement z_t .This is formally denoted by the following sampling distribution, which now takes the measurement z_t into consideration:

$$q(s_t^{[m]} \mid s^{t-1,[m]}, z^t, u^t, n^t) \quad (16)$$

An effective approach to accomplish this is to use H_∞ filter generated Gaussian approximation:

$$q(s_t^{[m]} \mid s^{t-1,[m]}, z^t, u^t, n^t) \sim N(s_t, s_t^{[m]}, P_t^{[m]}) \quad (17)$$

Approximates the distribution using H_∞ filter as following equations:

$$\begin{aligned} \hat{s}_{t+1}^{[m]} &= f(s_t^{[m]}, u_t) \\ P_{t+1}^{[m]-} &= \nabla f_t P_t^{[m]} \nabla f_t^T + \nabla G_u Q \nabla G_u^T \end{aligned} \quad (18)$$

where

$$\nabla f_t = \left. \frac{\partial f}{\partial s_t} \right|_{s_t = s_t^{[m]}} \qquad \nabla G_u = \frac{\partial f}{\partial u} \quad (19)$$

$$K_t^{[m]} = P_t^{[m]-} H_t^T (H_t P_t^{[m]-} H_t^T + R_t)^{-1} \quad (20)$$

$$s_t^{[m]} = \hat{s}_t^{[m]} + K_t^{[m]} (z_k - h(\hat{s}_t^{[m]})) \quad (21)$$

$$P_t^{[m]} = P_t^{[m]-} - P_t^{[m]-} \begin{bmatrix} G_{\theta_{n_t}}^T & I \end{bmatrix} R_{e,t-1}^{-1} \begin{bmatrix} G_{\theta_{n_t}} \\ I \end{bmatrix} P_t^{[m]-} \quad (22)$$

$$R_{e,t} = \begin{bmatrix} R & 0 \\ 0 & -\gamma I \end{bmatrix} + \begin{bmatrix} G_{\theta_{n_t}} \\ I \end{bmatrix} P_t^{[m]-} \begin{bmatrix} G_{\theta_{n_t}}^T & I \end{bmatrix} \quad (23)$$

From the Gaussian distribution generated by the estimated mean and covariance of the vehicle, the state of each particle is sampled:

$$s_t^{[m]} \sim N(\hat{s}_t^{[m]}, P_t^{[m]}) \quad (24)$$

When there is no observation, the vehicle state is predicted without the measurement update using (18). If many landmarks are observed at the same time, (20) and (21) are repeated for each observed landmark, and the mean and the covariance of the vehicle are updated based on the previously updated one.

B. *LANDMARK ESTIMATE BASED H_∞ FILTER*

The H_∞ FastSLAM represents the posterior landmark estimates $p(\theta_n \mid s^t, z^t, u^t, n^t)$ using low-dimensional H_∞ filter. In fact FastSLAM2.0 updates the posterior over the landmark estimates, respected by the mean $\mu_{n,t-1}^{[m]}$ and the covariance $\Sigma_{n,t-1}^{[m]}$ .the updated values $\mu_{n,t}^{[m]}$ and $\Sigma_{n,t}^{[m]}$ are then added to the temporary particle set S_t , along with the new pose. The update depends on whether or not a landmark n was observed at time t . For $n \neq n_t$, the posterior over the landmark remains unchanged as following [24]:

$$\begin{aligned} \mu_{n,t}^{[m]} &= \mu_{n,t-1}^{[m]} \\ \Sigma_{n,t}^{[m]} &= \Sigma_{n,t-1}^{[m]} \end{aligned} \quad (25)$$

For the observed feature $n = n_t$, the update is specified through the following equation [24]:

$$\begin{aligned} & p(\theta_{n_t} \mid s^{t,[m]}, n^t, z^t) = \\ & = \eta \underbrace{p(z_t \mid \theta_{n_t}, s^{t,[m]}, n^t, z^{t-1})}_{\sim N(z_t, g(\theta_{n_t}, s_t^{[m]}, R_t))} \underbrace{p(\theta_{n_t} \mid s^{t,[m]}, n^t, z^{t-1})}_{\sim N(\theta_{n_t}, \mu_{n,t-1}^{[m]}, \Sigma_{n,t-1}^{[m]})} \end{aligned} \quad (26)$$

The probability $p(\theta_{n_t} \mid s^{t,[m]}, n^t, z^{t-1})$ at time $t-1$ is represented by a Gaussian distribution with mean $\mu_{n,t-1}^{[m]}$ and covariance $\Sigma_{n,t-1}^{[m]}$. For the new estimate at time t to also be Gaussian, FastSLAM linearizes the perceptual model $p(z_t \mid \theta_{n_t}, s^{t,[m]}, n^t, z^{t-1})$ by EKF. Especially, FastSLAM approximates the measurement function g by the following first degree Taylor expansion [24]:

$$g(\theta_{n_t}, s_t^{[m]}) = \underbrace{g(\mu_{n,t-1}^{[m]}, s_t^{[m]})}_{\hat{z}_t^{[m]}} + \underbrace{g'(s_t^{[m]}, \mu_{n,t-1}^{[m]})}_{G_t^{[m]}}(\theta_{n_t} - \mu_{n,t-1}^{[m]}) \tag{27}$$
$$= \hat{z}_t^{[m]} + G_t^{[m]}(\theta_{n_t} - \mu_{n,t-1}^{[m]})$$

Under this approximation, the posterior of landmark n_t is indeed Gaussian. The mean and covariance are obtained using the following measurement update:

$$\hat{z}_t = g(s_t^{[m]}, \mu_{n_t,t-1}) \tag{28}$$

$$G_{\theta_{n_t}} = \nabla_{\theta_{n_t}} g(s_t, \theta_{n_t})|_{s_t = s_t^{[m]}; \theta_{n_t} = \mu_{n_t,t-1}^{[m]}} \tag{29}$$

$$Z_{n,t} = G_{\theta_{n_t}} \Sigma_{n_t,t-1}^{[m]} G_{\theta_{n_t}}^T + R_t \tag{30}$$

$$K_t = \Sigma_{n_t,t-1}^{[m]} G_{\theta_{n_t}}^T Z_{n,t}^{-1} \tag{31}$$

$$\mu_{n_t,t}^{[m]} = \mu_{n_t,t-1}^{[m]} + K_t(z_t - \hat{z}_t) \tag{32}$$

$$\Sigma_{n_t,t}^{[m]} = \Sigma_{n_t,t-1}^{[m]} - \Sigma_{n_t,t-1}^{[m]} \begin{bmatrix} G_{\theta_{n_t}}^T & I \end{bmatrix} R_{e,t}^{-1} \begin{bmatrix} G_{\theta_{n_t}} \\ I \end{bmatrix} \Sigma_{n_t,t-1}^{[m]} \tag{33}$$

$$R_{e,t} = \begin{bmatrix} R & 0 \\ 0 & -\gamma I \end{bmatrix} + \begin{bmatrix} G_{\theta_{n_t}} \\ I \end{bmatrix} \Sigma_{n_t,t-1}^{[m]} \begin{bmatrix} G_{\theta_{n_t}}^T & I \end{bmatrix} \tag{34}$$

C. Feature Initialization

A new feature is initialized as a function of the robot pose $s_t^{[m]}$ and measurement z_t .The feature mean $\mu_{n,t}^{[m]}$ and the feature covariance $\Sigma_{n,t}^{[m]}$ in the feature initialization are calculated as follows[12],[13]:

$$\mu_{n,t}^{[m]} = g^{-1}(z_t, s_t^{[m]})$$
$$\Sigma_{n,t}^{[m]} = (G_{\hat{n}}^{[m]} R_t^{-1} G_{\hat{n}}^{[m]T})^{-1} \tag{35}$$

D. Calculating Importance weight

Like FastSLAM2.0, the importance weight H_∞ FastSLAM should be computed by considering the most recent observation, and it is given by.

$$w_t^{[m]} = w_{t-1}^{[m]} \frac{p(z_t \mid s^{t,[m]}, z^{t-1}, u^t, n^t) p(s_t^{[m]} \mid u^t, n^t)}{p(s_t^{[m]} \mid s^{t-1,[m]}, z^t, u^t, n^t)} \tag{36}$$

E. Resampling

Sine the variance of the importance weights increase over time [22], [23], resampling plays a vital role in FastSLAM. In the resampling process, particles with low importance weight are eliminated and particles with high weights are multiplied. After, the resampling, all particle weights are then reset to

$$w_t^{[m]} = \frac{1}{N} \tag{37}$$

This enables the FastSLAM to estimate increasing environmental sates defiantly without growing a number of particles. However, resampling can delete good samples from the sample set, in the worst case, the filter diverges. The decision on how to determine the point of time of the resampling is a fundamental issue. Liu introduced the so-called effective number of particles N_{eff} to estimate how well the current particle set represents the true posterior [25]. This quality is computed as

$$N_{eff} = \frac{1}{\sum_{i=1}^{N} w_t^{[i]}} \tag{38}$$

Where $w^{(i)}$ refers to the normalized weight of particle i .The resampling process is operated whenever N_{eff} is bellow a pre-defined threshold, N_{tf} . Here N_{tf} is usually a constant value as following

$$N_{tf} = \frac{3}{4} M \tag{39}$$

Where M is number of particles.

V. Simulation Results

Simulation experiments have been carried out to evaluate the performance of the proposed approach in comparison with the classical method. The proposed solution for the SLAM problem has been tested for the benchmark environment, with varied number and position of the landmarks, available in [15].

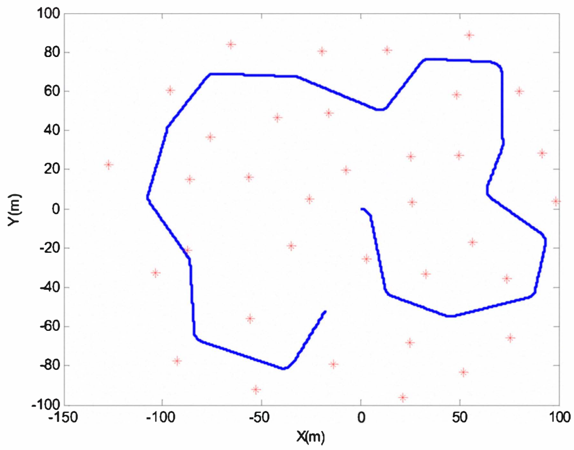


Fig. 1 The experiment environment: The star point "*" denote the landmark positions and blue line is the path of robot

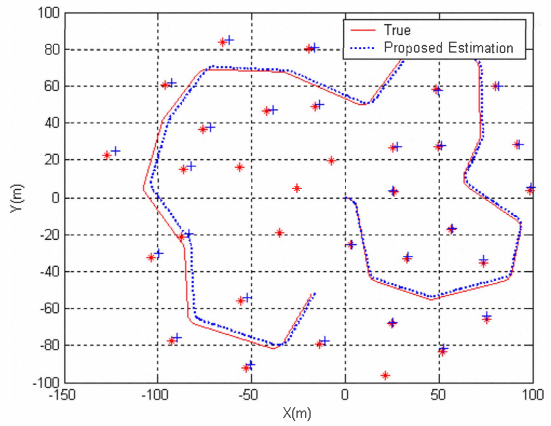
Fig.1 shows the robot trajectory and landmark location. The star points (*) depict location of the landmarks that are known and stationary in the environment. The state of the robot can be modeled as (x, y, ϕ) that (x, y) are the Cartesian coordinates and ϕ is the orientation respectively to the global environment. The kinematics equations for the mobile robot are in the following form

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} (V + v_v)\cos(\phi + [\gamma + v_\gamma]) \\ (V + v_v)\sin(\phi + [\gamma + v_\gamma]) \\ \frac{(V + v_v)}{B}\sin(\gamma + v_\gamma) \end{bmatrix} \tag{40}$$

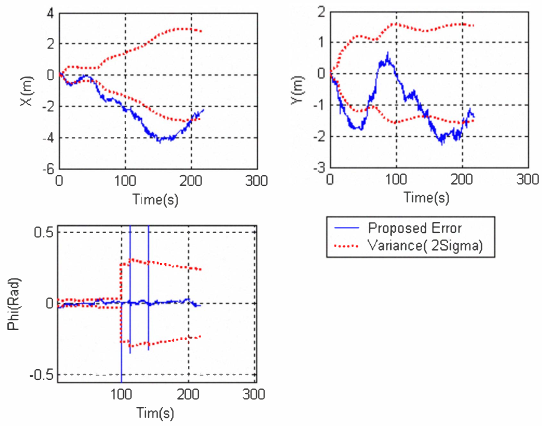
Where B is the base line of the vehicle and $u = [V \quad \gamma]^T$ is the control input at time t consisting of a velocity input V and a steer input γ. The process noise $v = \begin{bmatrix} v_v & v_\gamma \end{bmatrix}^T$ is assumed to be Gaussian. The vehicle is assumed to be equipped with a range-bearing sensor that provides a measurement of range r_i and bearing θ_i to an observed feature ρ_i relative to the vehicle. The observation z of feature ρ_i in the map can be expressed as:

$$\begin{bmatrix} r_i \\ \theta_i \end{bmatrix} = \begin{bmatrix} \sqrt{(x-x_i)^2+(y-y_i)^2}+\omega_r \\ \tan^{-1}\dfrac{y-y_i}{x-x_i}-\phi+\omega_\theta \end{bmatrix} \qquad (41)$$

Where (x_i, y_i) is the landmark position in map and $W = \begin{bmatrix} \omega_r & \omega_\theta \end{bmatrix}^T$ related to observation noise. The initial position of the robot is assumed to be $x_0 = 0$. The robot moves at a speed 3m/s and with a maximum steering angle 30 deg. Also, the robot has 4 meters wheel base and is equipped with a range-bearing sensor with a maximum range of 20 meters and a 180 degrees frontal field-of-view. The control noise is $\sigma_v = 0.3$ m/s and $\sigma_\gamma = 3^o$. A control frequency is 40 HZ and observation scans are obtained at 5 HZ. The measurement noise is 0.1 m in range and 0.1^o in bearing. Data association is assumed known. To evaluate the proposed method the performance, it is compared with FastSLAM2.0 for benchmark environment. First, we consider the situation where measurement noise is wrongly considered as $\sigma_r = 0.7$, $\sigma_\theta = 1.0$. The performance of the proposed method is compared with FastSLAM2.0. Fig.2 and Fig.3 show the comparison between the proposed algorithm and the FastSLAM2.0. It can be clearly seen that the results of the proposed algorithm are better than that of FastSLAM2.0. In other words, in the proposed algorithm, estimated vehicle path and estimated landmark with the actual path and the actual positions landmarks, coincide as closely as possible. This is because the proposed method does not require a priori knowledge of the system (the process and measurement noise covariance matrices Q_t and R_t, respectively) and deepened on only an assumption that the noises are bounded in certain energy level.

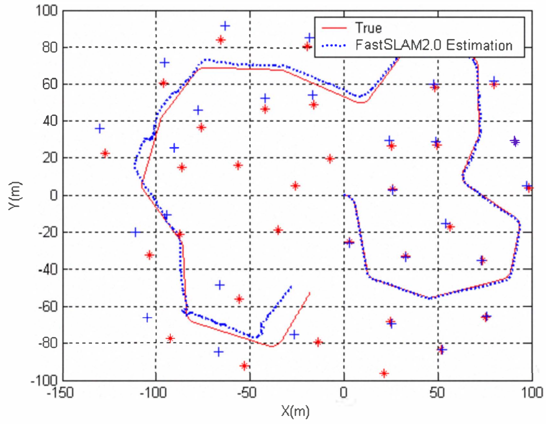


(a)

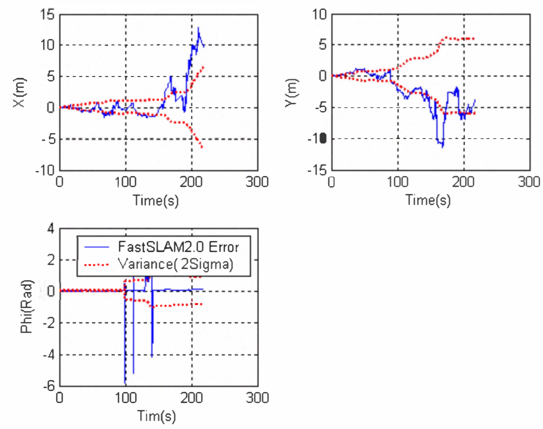


(b)

Fig. 2 The proposed method: a) Estimated robot path and estimated landmark with true robot path and true landmark. The "…" is the estimated path, the "+" are the estimated landmark positions. b) Estimated pose error with $2-\sigma$ bound.



(a)



(b)

Fig.3. FastSLAM2.0: a) Estimated robot path and estimated landmark with true robot path and true landmark. The "…" is the estimated path, the "+" are the estimated landmark positions. b) Estimated pose error with $2-\sigma$ bound.

VI. CONCLUSION

This paper presents a H_∞ FastSLAM. In the proposed method, H_∞ particle filter for robot pose estimation, and a H_∞ filter for landmark feature estimation is developed. The performance of proposed method is compared with classical FastSLAM2.0 for benchmark environment. The results show the effectiveness of the proposed method. This is because in our proposed method, depend on not to a priori knowledge of the process covariance matrix Q_t and the measurement noise covariance matrix R_t .

REFERENCES

[1] T.Bailey, "Mobile robot localization and mapping in extensive outdoor environments," Ph.D, dissertation, Univ. Sydney, Sydney, NSW, Australia, 2002.

[2] T.Bailey, J. Nieto and E.Nebot, "Consistency of the FastSLAM algorithm," IEEE Int. Conf, on Robotics and Automation.pp.424-429, 2006.

[3] M. Bosse, J. Leonard and S. Teller, "Large-scale CML using a network of multiple local maps," in workshop Notes of the ICRA Workshop on Concurrent Mapping and Localization for Autonomous Mobile Robots (W4), Washington, D.C, 2002.

[4] M. Boli´c, P. M. Djuri´c and S.Hong, "Resampling algorithms for particle filters: a computational complexity perspective," Journal on Applied Signal Proc., No. 15, pp. 2267-2277, 2004.

[5] M. W. M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, "A Solution to the Simultaneous Localization and Map Building (SLAM) Problem," IEEE Trans. Robot. Automat. Vol. 17, No. 3, pp.229–241, 2001.

[6] R. J. Fitzgerald. "Divergence of the Kalman filter," IEEE Trans. Autom. Control, Vol. AC-16, No. 6, pp. 736–747, Dec. 1971.

[7] N. Kwak, I. K. Kim, H. C. Lee, B. H. Lee, "Adaptive Prior Boosting Technique for the Efficeint Sample Size in FastSLAM," in the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (San Diego, CA) pp. 630–635, 2007.

[8] C. Kim, R. Sakthivel, W. K. Chung, "Unscented FastSLAM: A Robust algorithm for the simultaneous localization and mapping problem," in IEEE Int. Conf. Robot. Autom., pp.2439–2445, 2007.

[9] C. Kim, R.Sakthivel, W. K. Chung, "Unscented FastSLAM: A Robust and Efficient Solution to the SLAM Problem," IEEE Trans. Robot., Vol. 24, No. 4, 2008.

[10] L. Zhang, X. Meng, Y. Chen, "Convergence and Consistency analysis for FastSLAM," Intelligent Vehicles Symposium, 2009 IEEE.

[11] M. Montemerlo, S. Thrun, D. Koller, B.Wegbreit, "FastSLAM: a factored solution to the simultaneous localization and mapping problem," Proc. of AAAI national conf. artif. intell., Edmonton, AB, Canada, 2002.

[12] M.Montemerlo, "FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem with Unknown Data Association," PhD thesis, Carnegie Mellon University, 2003.

[13] M.Montemerlo, S.Thrun, D.Koller, B.Wegbreit, "Fastslam 2.0: an improved particle filtering algorithm for simultaneous localization and mapping that provably converges," in the Int. Joint Conf. on Artificial Intelligence, 2003.

[14] R. K. Mehra, "On the identification of variances and adaptive Kalman filtering," IEEE Trans Autom Control, Vol. AC-15, No.2, pp. 175–184, Apr, 1970.

[15] Online Avaiable:http://www.personal.acfr.usyd.edu.au/tbailey/ software / index .html, 2008.

[16] S. Thrun, D. H¨ ahnel, D. Ferguson, M. Montemerlo, R. Triebel, "A system for volumetric robotic mapping of abandoned mines," in IEEE Int. Conf. Robot. Automat. Taipei, Taiwan, May 2003, pp.4270–4275.

[17] S. Thrun, M. Montemerlo, D. Koller, B.Wegbreit, J.Nieto, E. Nebot, "FastSLAM: An efficient solution to the simultaneous Localization and mapping problem with unknown data association," Journal of Machine Learning Research, 2004.

[18] S. Thrun, W. Burgard, D. Fox, "Probabilistic robotics," MIT Press, Cambridge, 2005.

[19] S. Thrun, D. Fox, W. Burgard, "A probabilistic approach to concurrent mapping and localization for mobile robots," Mach. Learn., Vol. 31, pp. 29–53, 1999.

[20] R. Smith, P. Cheeseman, "On the representation and estimation of spatial uncertainty," Int. J. Robot. Res., Vol. 5, No. 4, pp. 56–68, 1986.

[21] X. Wang, H. Zhang, "A UPF-UKF framework for SLAM," IEEE Intl. Conf. on Robotics and Automation, pp 1664-1669, 2007.

[22] G. Grisetti,C. Stachniss, W. Burgard, "Improving Grid-based SLAM with Rao-Blackwellized Particle Filters by Adaptive Proposal and Selective Resampling," in IEEE Intl. Conf. on Robotics and Automation, pp 2443-2448,2005.

[23] G. Grisetti, C. Stachniss, W. Burgard. "Improved Techniques for Grid Mapping with Rao-Blackwellized Particle Filters," IEEE Trans. on Robotics, Vol. 23(1), pp 34-46, 2007.

[24] S. Williams, G. Dissanayake, H. F.Durrant-Whyte, "towards terrain-aided navigation for underwater robotics," Adv. Robot., Vol.15, No. 5, pp. 533–550.

[25] A.Kong, J.S.Liu, W.H.Wong, "Sequential and Bayesian missing data problems," journal of the American Statistical Association, Vol.89, No.425, PP.278-288, 1994.

[26] M.Li, H.Bing-rong, L.Rong-hua, W.Zhen-Hua, "A novel method for mobile robot simultaneous localization and mapping," Journal of Zhejiang university Scince A, P937-944, 2006.

[27] X.Chen, "An Adaptive UKF-Based Particle Filter for Mobil Robot SLAM," International Joint Conference on Artificial Inteligence, 2009.

[28] N.Kwak, G.Kim,B.Lee, "A new compensation technique based on analysis of resampling process in FastSLAM," Journal of Robotica, Vol.26, pp. 205–217, 2007.

[29] I.Kim, N.Kwak, H.Lee, B.Lee, "Improved particle fusing geometric relation between particles in FastSLAM", Journal of Robotica, Vol. 27, pp. 853–859, 2009.

[30] B. Hassibi and T. Kailath, " H_∞ Adaptive Filtering," Proc. IEEE ICASSP'95, Detroit, 949-952, 1995.

[31] X.Shen and L.Deng, "Discrete H1 Filter Design with Application to Speech Enhancement," Proc. IEEE ICASSP'95, Detroit, 1504-1507, 1995.

[32] K.Nishiyama, "Computational Improvement of the Fast H_∞ Filter Based on Information of Input Predictor," IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL. 55, NO. 8, August, 2007.

[33] K.Nishiyama, "An H_∞ Optimization and Its Fast Algorithm for Time-Variant System Identification," IEEE Trans on Signal Processing, Vol. 52, No. 5, May, 2004.

[34] D.Simon, "Optimal State Estimation: Kalman H_∞ and Nonlinear Approaches," Wiley, New York, 2006.