HCTL Open International Journal of Technology Innovations and Research (IJTIR)
http://ijtir.hctl.org
Volume 14, April 2015
e-ISSN: 2321-1814, ISBN (Print): 978-1-62951-946-3

Comparison of FastSLAM1.0 and FastSLAM2.0 for Relatively High Motion Noise Environment

Manjula Sahu[1], Chetan Kamble[2]

elexmanjula@gmail.com

Abstract

Simultaneous Localization and Mapping (SLAM) is receiving much attention in robotics and control community in an effort to build autonomous actuators, which can survive in uncertain and unexplored environment. FastSLAM, introduced by Montemerlo is an efficient and robust solution in the field of localization. The core idea of FastSLAM revolves around the Rao-blackwellized state, where the trajectory is represented by weighted samples and the map is computed analytically. This approach uses a particle filter in which each particle carries an individual map of the environment. Accordingly, a key question is how to reduce number of particles for accurately map the environment where the motion noise is very high. The paper illustrates two approaches FastSLAM1.0 and FastSLAM2.0, which differ by their sampling strategy. In general the performance of FastSLAM2.0 is similar to FastSLAM1.0 but it varies particularly in situations in which the motion noise is high in relation to the measurement noise. The comparison of their accuracy is shown by the simulation results for different number of particles and high motion noise.

Keywords

Robotics, SLAM, Particle Filter, RBPF, FastSLAM2.0

Introduction

Localization of a mobile robot on a previously unexplored environments where the GPS system can't be work, requires the robot to consistently generate a map of its surroundings by a process called simultaneous localization and mapping (SLAM) [1] [6] . SLAM originates from a seminal paper by Smith and Cheeseman, which proposed to use of the Extendedkalman filter (EKF) for solving SLAM problem [14] . It has been constructed and solved as a conceptual problem in a number of different forms like Kalman filter, Extended kalman kilter (EKF), Unscented kalman filter (UKF), Particle filter, Grid mapping [3] [4] etc. SLAM has also been implemented in a number of different areas like indoor to outdoor robots, underwater, and airborne systems [20] [21] . There are many results reported in the literature conveying theoretical and conceptual aspects of SLAM. However, some issues remains as it is in practically realizing and building rich maps as part of a SLAM algorithmdue to mathematical complexity.

Previously in the 90's the practical implementation of stochastic SLAM has built on the Eextendedkalman filter (EKF) [2] . The EKF-based SLAM algorithms experienced computational complexity, non-linearity and sensitivity to failures in data association that complicate their application to large and real-world environments. The covariance matrices in EKF are quadratic in the size of the map, and to update them requires time quadratic in the number of landmarks. This quadratic complexity has found to be a big obstacle in SLAMalgorithms to map when there is more number of features present [5] . Thus to make the SLAM more accurate for nonlinear, non-Gaussian motion and observation model, particle filter is used. The idea behind the particle filter is to represent the posterior density by a set of samples with associated weights. These samples are using to estimate the robot pose. The FastSLAM algorithm proposed in [6] is an efficient approach to SLAM based on particle filtering, which also reduce the mathematical complexity. This approach, factors the SLAM posterior exactly into a product of a robot path posterior and N landmark posteriors conditioned on the robot path estimate. The factored posterior, which is a non-linear model can be approximated using a particle filter. Here we present the accuracy between two algorithms, called FastSLAM1.0 and FastSLAM2.0 to map the real world environment.

This paper presents an advanced version of FastSLAM algorithm and compares it with the previous one. The proposed modification removes the uncertainty in the robot pose. Further the algorithm is made more accurate by making the proposal distribution to depend not only on estimated motion, but also on the recent measurement. Such an approach is very useful especially when the noise in motion is high relative to the measurement noise. To obtain a suitable proposal distribution, FastSLAM2.0 linearizes the motion and the measurement model in same way as EKF. The main contribution of this paper is to present the FastSLAM2.0 algorithm to track the path and robot's own position very accurately, even with less number of particles than the previous version and is suitable even when noise in motion is relatively high.

The paper is organized as follows; Section 2 presents the problem statement, in which we describe the general probability SLAM and the process and observation model used in our simulation, which is a 2-D vehicle with a rangebearing sensor mounted on its head. Section 3 describes the Rao-blackwellization particle filter (RBPF) algorithm for SLAM. Section 4 discusses the property of advance form of RBPF SLAM (FastSLAM2.0), and their sampling process along with prediction and update technique. Section 5 presents performance of the FastSLAM1.0 and FastSLAM2.0 algorithm and their comparison in different cases with the simulation results. The final section provides conclusion and discussion.

Problem Statement

Probabilistic SLAM

In the probability theory, the simultaneous localization and map building (SLAM) problem needs to present the distribution in the form of [19]

$$p(x_{v_t}, m \mid Z_{0:t}, U_{0:t}, x_{v_0})$$

For all times t. This probability distribution describes the joint posterior density of the vehicle state xvt (at time t)and the N landmark locations m = m1,m2,...,mN, with the given observations Z0:t = z1,z2,...,zt, history of control inputs up to and including time t, U0:t = u1,u2,...,utand initial state of the vehicle xv0. To calculate the posterior, the vehicle is given a probabilistic motion model, in the form of the conditional probability distribution (xvt|ut, xvt−1). This distribution describes how a control ut applied in the time interval (t −1, t), affects the current pose of the robot. Additionally, the vehicle is given a probabilistic measurement model, denoted as p(zt|xvt, m,nt), describing how measurements changes according to the recent state [6] . We can describe both the model by non-linear functions f and h, with the Gaussian noise ζ and δ:

$$x_{v_t} = f(x_{v_{t-1}}, u_t) + \zeta$$
$$z_{it} = h_i(x_{v_t}, m) + \delta$$

Where, ζ and δare Gaussian noise with zero meanand Q, Rcovariancerespectively.

The state transition is assumed to be Markov process inwhich the next state xtdepends only on the preceding statext-1and the applied control ut. The observation model definesby the probability of getting observation zt, when vehicle andlandmark poses are known [14] .

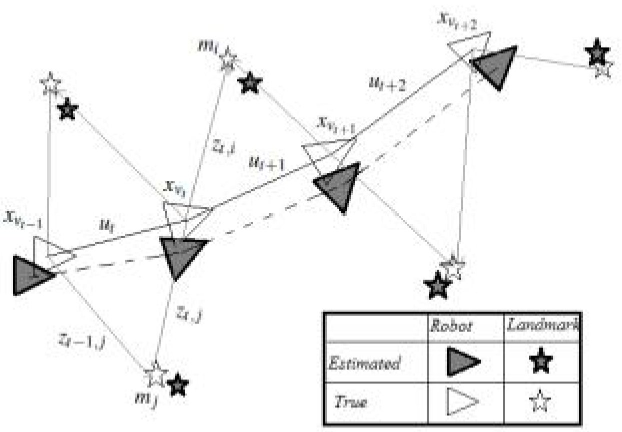


Fig.1: The essential SLAM problem

Fig(1) shows a typical SLAM problem, a mobile robot havingthree degree of freedom is moving through an unexploredenvironment and taking relative observation of a unknownlandmarks using a sensor located on the robot. At timeinstant t,xvt is a state vector describing the x, y position andorientation of the vehicle, and miis a vector describing thelocation of the ith landmark, which is assumed to be stationaryin this case.

The important characteristic of the SLAM (with knowndata association) which is used in FastSLAM is: estimationof landmarks are conditionally independent when the robot'spath is known. This algorithm uses a particle filter to sampleover robot paths. Each particle is having its own N low dimensionalEKFs, which is used to estimate map for eachparticle. This representation requires O(NM) memory, whereM is the number of particles in the particle filter and N is thenumber of landmarks. Updating this filter requires O(MlogN)time, with or without knowledge of the data associations [6] .

Model for 2-D Range- Bearing SLAM

In this paper, we consider the SLAM state as vehicle pose(position and heading) xvtat time t and the location of thestationary landmarks m observed by the range and bearingsensor from the environment. The state at time t is represented

$$x_t = [x_{vt}, y_{vt}, \phi_{vt}, x_1, y_1, \ldots\ldots, x_N, y_N]^T = \begin{bmatrix} f(x_{v_{t-1}}, u_t \\ m \end{bmatrix}$$

Where, xvt,yvtis the x-y position of the vehicle, φvt is the headingangle at time t and landmark location m= [x1,y1,…….xN, yN] .For describing the vehicle motion, we use the kinematic modelfor the trajectory of the front wheel (assuming zero wheel slip).

$$x_{v_t} = f(x_{v_{t-1}}, u_t) = \begin{bmatrix} x_{vt-1} + V\Delta t cos(\phi_{vt-1} + \gamma_t) \\ y_{vt-1} + V\Delta t sin(\phi_{vt-1} + \gamma_t) \\ \phi_{vt-1} + \frac{V\Delta t}{B} sin(\gamma_t) \end{bmatrix}$$

Here ∆t= time from t-1 to t, during this period the velocity V and steering angle γtof the front wheel are assumed to be constant and they serves as a 'controls' ut = [V, γt] T. B is the wheel base between the front and rear axles, which is used tocalculate steering angle of the vehicle. For a range-bearing measurement from the vehicle to landmark mi= [xi, yi] , the observation model is given by

$$x_{v_t} = f(x_{v_{t-1}}, u_t) = \begin{bmatrix} x_{vt-1} + V\Delta t cos(\phi_{vt-1} + \gamma_t) \\ y_{vt-1} + V\Delta t sin(\phi_{vt-1} + \gamma_t) \\ \phi_{vt-1} + \frac{V\Delta t}{B} sin(\gamma_t) \end{bmatrix}$$

The motion model, observation model and the measured control parameters utare having noise in it, which leads touncertainty in the estimated state. Consequently we requirethe probabilistic filter to recursively estimate the distribution over the state with noisy information.

Rao-blackwellized particle filter

In general, Rao-blackwellised particle filtering sometimesdenoted as marginalized particle filter uses a sequential Monte-Carlo method, in which only some variables are sampled usingparticle filter and the others are handled through the EKFs [17] . It results a large decrease in the variance of a MonteCarlo estimate compared to a normal particle filter. FastSLAMis an algorithm which is used a Rao-blackwellized particle filter(RBPF) method to estimate state in Simultaneous localizationand mapping problem. This method is based on the importantcharacteristic that the posterior can be factored as:

$$p(x_v^t, m \mid z^t, u^t, n^t) = p(x_v^t \mid z^t, u^t, n^t) \prod p(m^n \mid x_v^t, z^t, u^t, n^t)$$

The factorization is assumed to be exact in SLAM problems.It says that if we theoretically knew the path of the vehicle,i.e. xtv = xv1, xv2,…..xvt, the estimation of the landmark positionsbecomes independent to each other. The in-dependency of thelandmark estimation makes it possible to factor the posteriorinto a term that estimates the probability of path and Nterms that estimate the position of the landmark [5] . Here,zt = z1, z2, …..zt is a sequence of measurements (i.e. range andbearing to nearby landmarks in this case), ut = u1, u2,…..utis a sequence of control inputs and nt = n1, n2,….nt are theidentification variables used in data association, where each ntspecifies the identity of the landmark observed by the sensor at time t.

FastSLAM samples the non-linear motion using particlefilter. Each particle has attached its own map, which is estimatedby N extended kalman filters [11] . The mthparticle xt [m] contains path xt, [m] along with N Gaussian landmark estimate with mean μn;t [m] and covariance Pcovn;t [m] :

$$x_t^{[m]} = x_v^{t,[m]}, \mu_{n,t}^{[m]}, Pcov_{n,t}^{[m]}, \ldots\ldots, \mu_{N,t}^{[m]}, Pcov_{N,t}^{[m]}$$

The key steps for FastSLAM are:

- Initially, sampled a new robot path based on the recentmotion control ut
- According to the samples and observations, update landmarksfilter corresponding to the new observation.
- After that, assign a weight to each particle according totheir probability.
- At last the most important step resampling is performed after crossing some threshold. Resampling, which is essential part of the particle filter, decreases diversity inthe samples by removing some less weighted particles.

The probability for the mthparticle to be sampled is given by Wt [m] referred to as importance factor:

$$W_t^{[m]} = \eta \int p(z_t \mid m_{nt}, n_t) p(m_{nt} \mid x_{t-1}^{[m]}, z_{t-1}, n_{t-1})$$

Where, η = constant

$$p(z_t \mid m_{nt}, n_t) \sim N(z_t; h(m_{n,t}), x_{v_t}^{[m]}, R)$$

$$p(m_{nt} \mid x_{v_{t-1}}^{[m]}, z_{t-1}, n_{t-1}) \sim N(m_{nt}; \mu_{nt,t-1}^{[m]}, Pcov_{nt,t-1}^{[m]})$$

After generating particles from the motion model, the newset of sample is obtained by sampling from the proposaldistribution. In case of FastSLAM1.0, motion modelitself taken as proposal distribution. Each particle xt, [m] iscalculated with the probability proportional to the importancefactor Wt [m] . Linearization of measurement model makes theresult Gaussian and closed, therefore FastSLAM employsEKF trick, i.e., model is approximated by linear function bycalculating Jacobian. The key factor of FastSLAM is that, eachparticle is having individual data association [6] , which makesit more flexible to estimate the state.

FastSLAM2.0

The new FastSLAM2.0 is employed to remove the inefficiency of the proposal distribution in regular FastSLAM.In Fastslam1.0, the position xvt [m] is sampled in according tothe prediction arising from the motion model. It does not consider the measurement zt generated at time t; instead, the measurement is incorporated through re-sampling [5] . This approach is not good enough when the noise in the vehicle motion is large compared to measurement noise. Because of that, the sampled poses can fall into re-sampling phase even having high probability. Unlikely to FastSLAM1.0, the Fast-SLAM2.0 comprises the current observation into the proposal distribution, not just for calculating importance weight but also in order to batter match the posterior.

Sampling the pose

The modified version of proposal distribution needs twobasic changes in the FastSLAM1.0 algorithm. First, to generatesamples from the new proposal distribution and second,generate a formula for the weights of the particles.

- FastSLAM2.0 implements a new idea i.e. vehicle statesare sampled, considering the motion control ut and themeasurement zt. This is denoted as

$$x_t^{[m]} = p(x_{v_t} \mid x_{v_{t-1}}^{[m]}, u_t, z_t, n_t)$$

- Incorporating the measurement makes sense if we considerthe current estimate of the observed landmark [3] .
- Taking samples from the proposal distribution can takeas a process of generating new trajectories. Therefore,the proposal distribution, that is used to find importancefactor can be calculated as :

$$p(x_{v_t} \mid x_v^{t-1,[m]}, u_t, z_t, n_t) = \eta^{[m]} \int p(z_t \mid m_{n_t}, x_{v_t}, n_t)$$
$$p(m_{n_t} \mid x_v^{t-1,[m]}, z^{t-1}, n^{t-1}) dm_{n_t} p(x_{v_t} \mid x_{v_{t-1}}, u_t)$$

That means proposal distribution is a factor of state distribution p(xvt j xvt-1, ut) and the probability of measurement zt.

Therefore to obtain a suitable proposal distribution, theFastSLAM2.0 algorithm linearizes the motion model,same as EKF-based SLAM [2] . As a result, modified version of proposal distribution is calculated in closedform.

Prediction and update

Prediction and update are the two major process in filteringtheory. In general, sampling directly from the proposal distributionis not possible. So to make the proposal distributionGaussian, we approximate robot motion model by a linearfunction.

Likely to EKF-SLAM approximation, the proposal distributionis Gaussian with mean and covariance:

$$x_v = x_v^{[m]} + Pcov^{[m]} H_v^T Sf^{T\,[m]} v$$

$$Pcov_{x_t}^{[m]} = [H_v^T Sf^{T\,[m]} H_v + Pcov^{T\,[m]}]^{-1}$$

Where,$v = (z_{est} - z_t^{[m]})$ is the innovation matrix.

Sf, the innovation covariance of the feature observation is present as,

$$Sf = H_f^T Pcov_{n_t,t-1}^{[m]} H_f + R$$

The matrices HvandHfare the Jacobian of measurement function h with respect to xvand m respectively as:

$$H_v = \Delta_{x_{v_t}} h(x_{v_t}, m_{n_t}) \mid_{x_{vt} = \hat{x}_{v_t}^{[m]}, m_{n_t} = \hat{m}_t^{[m]}}$$
$$H_f = \Delta_{m_{n_t}} h(x_{v_t}, m_{n_t}) \mid_{x_{v_t} = \hat{x}_{v_t}^{[m]}, m_{n_t} = \hat{m}_t^{[m]}}$$

The updating step remains same as the FastSLAM1.0; thereforethe update equations are:

$$K_t^{[m]} = Pcov_{n_t,t-1}^{[m]} H_f^T Sf^{T\,[m]}$$
$$x_v = x_v^{[m]} + K_t^{[m]}(z_{est} - z_t^{[m]})$$
$$Pcov_{x_t}^{[m]} = [I - K_t^{[m]} H_v] Pcov^{T\,[m]}$$

Here, K is the kalman gain and $z_{est}$is the estimatedobservation.

Importance Weight

Importance weight are having a significant role in theaccurate convergence of the particles. Taking samples fromthe proposal distribution are distributed according to p(xtv|zt-1,ut-1, nt-1), and therefore do not match with the desiredposterior p(xtv|zt, ut, nt). The importance factor is used tocorrect this difference. Therefore, it is calculate as a ratio oftarget distribution over proposal distribution as

$$W_t^{[m]} = \frac{Target distribution}{Proposal distribution}$$
$$= \frac{p(x_v^{t,[m]} \mid z^t, u^t, n^t)}{p(x_v^{t-1,[m]} \mid z^{t-1}, u^{t-1}, n^{t-1}) p(x_v^{t,[m]} \mid x_v^{t-1,[m]}, u^t, n^t)}$$

It is shown from the above equation that, when the target distribution islarger than the proposal distribution, the samples receives highweights and similarly where the target distribution is smallerthan the proposal distribution, the samples will be given lowerweights.

Simulation result

In this paper, we consider the vehicle model with wheelbase of 10.5cm, control noise (σv = 0.3m/ σγ = 3°), withvariance σvandσγ in velocity and steering angle respectivelyand observation noise is (σr = 0.1m, σθ = 1°), with varianceσr,σθin range and bearing angle respectively with the zeromean. Controls are updated at every 40Hz(25 millisecond)and observation scans are obtained at every 5Hz(200 millisecond).Each scan consist of range-bearing measurement to alllandmarks within a 3 meter radius. The area covering by therange sensor (mounted on the robot's head) in each scan canbe vary with different sensors. Data association is assumed tobe known.

Simulations were performed between two different algorithm,i.e. FastSLAM1.0 and FastSLAM2.0. Initially ineach algorithm, simulation were run with 100 particles. Resamplingis performed after the effective sample size fallsbelow 75 % of the total number of particles. We can accuratelycompare the two results shown in Fig. 2(a)and 2(c), thatthe FastSLAM2.0 tracks the path more accurately than theFastSLAM1.0.

FastSLAM2.0 also has the special property that it canconverge with a single particle. The proof of the convergence isgiven in [6] . When the measurement error is significantly smallthen the motion error, FastSLAM2.0 will perform excellentover the regular FastSLAM. The results of this case are shownin Fig.3 (a) and 3(b). In the simulation, the performance of thetwo algorithms is compared on simulated data with very highlevel of motion noise.

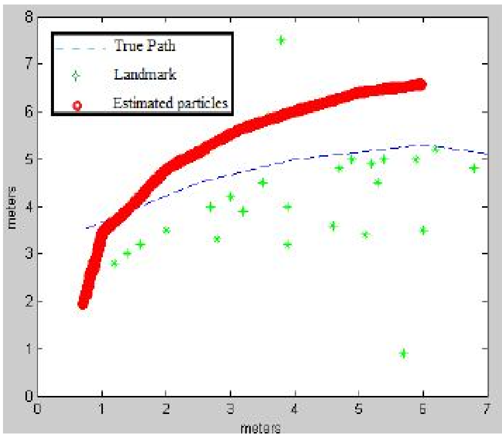
The performance of the two algorithms can also be comparedby keeping the measurement error constant for both thealgorithms and varying the number of particles. As comparingFig. 2(b) and 2(c), we can say that FastSLAM2.0 requires aless number of particles than regular SLAM in order to achievea given level of accuracy.

Conclusion and Discussion

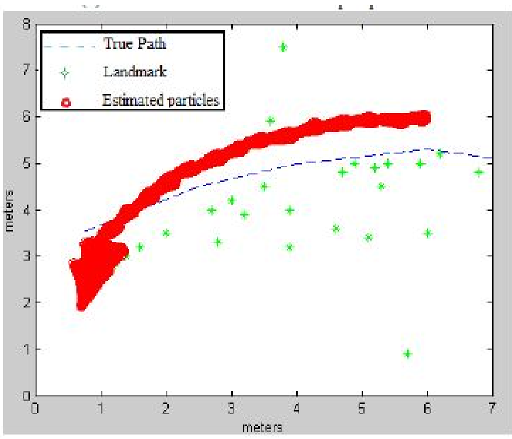
The paper discussed two different approaches, called FastSLAM1.0 and FastSLAM2.0 and their performance comparisonhaving different cases. In General the two algorithmgives almost same performance until the motion noise is notrelatively high. In the first case we compared accuracy ofthe algorithms, varying the number of particles. As we canconclude from the simulation result that the FastSLAM2.0decreases the number of particles necessary to achieve agiven level of accuracy. In the second case we comparedperformance of the two algorithm when the motion noise ishigh relative to the measurement noise. For very high value ofmotion error, FastSLAM1.0 begins to diverge, while the errorin FastSLAM2.0 continues to decrease.

FastSLAM takes samples over the robot paths, insteadof maintaining a parameterized distribution of solution likethe EKF. This feature enables FastSLAM to apply differenthypotheses to different circumstance. Therefore it can performvery well, even with consistently high data association ambiguity.Hence we conclude that the FastSLAM2.0 is well suitablein every aspects over FastSLAM1.0.

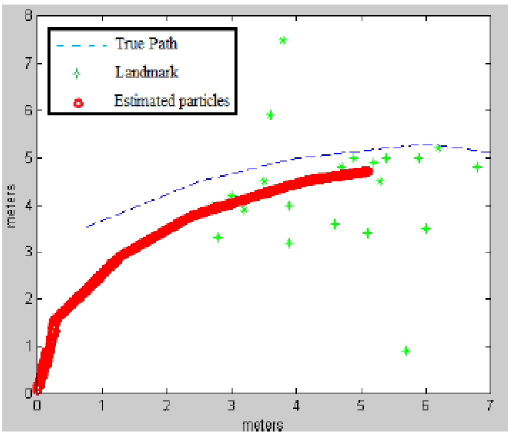
Taking samples over the robot pose enables a special case ofthe multi-robot slam problem to be implemented in future [2] .Other than that we can also generate more accurate map foroutdoor and dynamic environment with optimal control policy [23] .



(a) FastSLAM1.0 with 100 sample particles

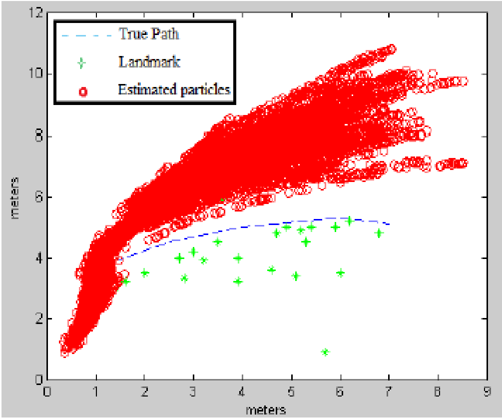


(b) FastSLAM1.0 with 1000 sample particles

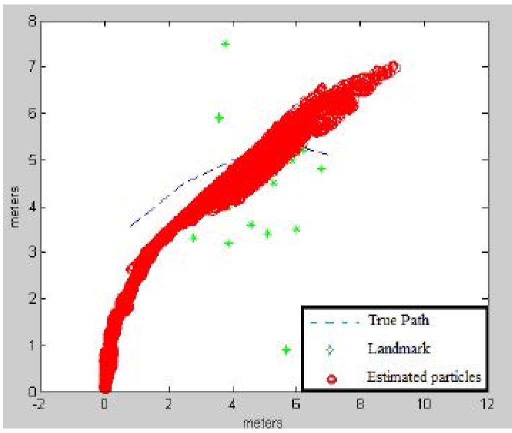


(c) FastSLAM2.0 with 100 sample particles

Fig. 2: Simulation Result for two different algorithm



(a)FastSLAM1.0



(b) FastSLAM2.0

Fig. 3: (a) and (b) shows the result of FastSLAM1.0 and

FastSLAM2.0 for very high motion noise

References

[1] Hugh Durrant Whyte And Tim Bailey."Simultaneous Localization andMapping: Part I" IEEE Robotics and Automation Magazine vol. 13, issue2,page-99-110, 05 June 2006.

[2] Tim Bailey, Juan Nieto and Eduardo Nebot "Consistency of the FastSLAMAlgorithm", International Conference on Robotics and Automation -ICRA, pp. 424-429, 2006.

[3] Hugh Durrant Whyte and Tim Bailey, "Simultaneous Localization andMapping: Part II., IEEE Robotics and Automation Magazine, Vol.13,Issue-3, page: 108-117, September2006.

[4] Randall C. Smith,PeterCheeseman , "On the Representation and Estimation of Spatial Uncertainty ", International Journal of RoboticsResearch ,Vol.5, Issue 4, Winter, 1987 Pages 56-68

[5] Sebastian Thrun,WolframBurgard,Dieter Fox. "PROBABILISTICROBOTICS",vol. 31, MIT Press, Cambridge, MA, 2005.

[6] Michael Montemerlo "FastSLAM: A Factored Solution to the SimultaneousLocalization and Mapping Problem With Unknown Data Association", PhD thesis, CMU-RI-TR-03-28 The Robotics Institute CarnegieMellon University,july2003 Pittsburgh.

[7] Michael Montemerlo, Sebastian Thrun,DaphneKoller and Ben Wegbreit"FastSLAM 2.0: An Improved Particle Filtering Algorithm for SimultaneousLocalization and Mapping that Provably Converges ", Proceedings ofthe 18th international joint conference on Artificial Intelligence , IJCAI,page 1151-1156. Morgan Kaufmann, 2003.

[8] Lasse Klingbeil ,TimWark "A Wireless Sensor Network for Real-timeIndoor Localisation and Motion Monitoring", IPSN Proceedings of the7th international conference on Information processing in sensor networks2008,IEEE Computer Society Washington, DC, USA Pages 39-50.

[9] Po Yang, Wenyan Wu "Efficient Particle Filter Localization Algorithm in Dense Passive RFID Tag Environment", IEEE Transactions on Industrial Electronics, Vol. 61, No. 10, October 2014

[10] Juan Nieto, Jose Guivant, Eduardo Nebot,SebastianThrun, "Real TimeData Association for FastSLAM", Robotics and Automation, 2003. Proceedings.ICRA '03. IEEE International Conference on Vol.1, page 412-418 September 14-19, 2003.

[11] Liang Zhang, Xu-jiongMeng, Yao-wu Chen, "Convergence and Consistencyanalysis for FastSLAM. Intelligent Vehicles Symposium, 2009IEEE,page- 447-452.

[12] K. Chong and L. Kleeman, "Feature based mapping in real, large scaleenvironments using an ultrasonic", International Journal of RoboticsResearch, vol.18 (1): page- 319, 1999.

[13] M. Deans. "Bearings-Only Localization and Mapping"PhD thesis,Carnegie Mellon University, 2002.

[14] Y. Bar-Shalom, X.R. Li, and T. Kirubarajan "Estimation with Applicationsto Tracking and Navigation", John Wiley and Sons, 2001.

[15] M. Bosse, P. Newman, J. Leonard, and S. Teller,"An Atlas framework forscalable mapping", Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on Vol.2,Page:1899 - 1906.

[16] Mohinder S. Grewal, Angus P. Andrews, "KalmanFiltering:Theory andPractice Using Matlab", Second Edition , John Wiley And Sons.

[17] A Doucet, N. de Freitas, K. Murphy, and S. Russell. "Rao-Blackwellisedparticle filtering for dynamic Bayesian networks". Sixteenth conferenceon Uncertainty in artificial intelligence (UAI), page- 176-183, June 2000.

[18] Guoqiangmao, BarisFidan "Localization Algorithms and Strategies forWireless Sensor Networks",Information Science Reference,Hershey,Newyork2009.

[19] Andreas StrksenStordal,"Sequential Monte Carlo Methods for BayesianFiltering",PhD Thesis, Mathematical Statistics University of Bergen,Norway 29th May 2008.

[20] Robertson, P.; Angermann, M.;Krach B. (2009)"Simultaneous Localizationand Mapping for Pedestrians using only Foot-MountedInertial Sensors",Ubicomp2009. Orlando, Florida, USA: ACM.

doi:10.1145/1620545.1620560.

[21] J. Mullane, B.-N. Vo, M. D. Adams, and B.-T. Vo,"Arandomfinite-set approach to Bayesian SLAM" IEEE Transactions on RoboticsVol.27,Page:268-282, April 2011.

[22] Michael Angermann,Patrick Robertson, "Simultaneous Localization andMapping Without Exteroceptive Sensors-Hitchhiking on Human Perceptionand Cognition",Vol. 100,Proceedings of the IEEE, Page:1840-1848,May 13th, 2012

[23] Iizuka, S.,Nakamura, T. ; Suzuki, S., "Robot navigation in dynamic environment using Navigation function APF with SLAM",IEEE proceeding Mecatronics (MECATRONICS), 2014 10th France-Japan/ 8thEurope-Asia Congress onNov. 2014, Page(s): 89 - 92

[24] Tim bailey ,Hugh Durrant Whyte , "Mobile Robot Localisation andMapping in Extensive Outdoor Environments PhD thesis, AustralianCentre for Field Robotics Department of Aerospace, Mechanical andMechatronic Engineering The University of Sydney,August2002.

Authors

Manjula Sahu

She received the B.E. degree in Electronics & telecommunication engineering from R.C.E.T. Bhilai, in 2010 and M.Tech degree in electrical engineering from V.J.T.I.Mumbai, in 2013. In 2013, she joined the Department of electrical engineering, V.J.T.I. Mumbai, as an assistant professor. Her research areas are particle filter, SLAM, optimal control, micro-electronics.

Chetan Kamble

He received the B.E. degree in Electronics engineering from Nagpur University, in 2010, and the M.Tech. degreein electronics engineering from VJTI Mumbai, in 2013. Currently he is pursuing PhD from V.J.T.I. Mumbai and his research areas are biomedical engineeringand Nano electronics.