Novel Adaptive Particle Filters in Robot Localization[1)]

JIANG Zheng-Wei GU Yuan-Tao
(*Department of Electronic Engineering, Tsinghua University, Beijing* 100084)
(E-mail: marrjiang@yahoo.com.cn)

Abstract The research of robot localization aims at accuracy, simplicity and robustness. This article improves the performance of particle filters in robot localization *via* the utilization of novel adaptive technique. The proposed algorithm introduces probability retracing to initialize particle sets, uses consecutive window filtering to update particle sets, and refreshes the size of particle set according to the estimation state. Extensive simulations show that the proposed algorithm is much more effective than the traditional particle filters. The proposed algorithm successfully solves the nonlinear, non-Gaussian state estimation problem of robot localization.

Key words Robot localization, particle filters, K-L distance, probability retrieval

1 Introduction

The research of mobile robot navigation is mainly divided into four interrelated areas: map building, self-localization, path planning, and obstacle avoidance. Robot self-localization is one of the key problems[1]. The robot collects noisy sensor measurements during the movement to localize itself in a known environment. Robot self-localization is a procedure of fusion of inaccurate information generated from multi-sensors. It is also an on-line state estimation problem of nonlinear, non-Gaussian processes. Most successful methods of robot self-localization are the variants of Bayesian filters[2,3], where the robot state (including its location and motion) is represented by the posterior probability distribution over the map space.

Based on Markov hypothesis, particle filters are sequential Monte Carlo methods[4∼6] that represent the posterior probability distribution by a set of random samples which are updated by the method of sequential importance sampling with resampling (SISR). Particle filters are the efficient tools to estimate the various unknowns for dynamic systems, even for the nonlinear, non-Gaussian time-varying systems. So particle filters suit for the problem of robot localization – an nonlinear, non-Gaussian Bayesian state estimation problem, where the traditional methods such as Kalman filters[7∼9] are not effective.

Thrun *et al.* presented Monte Carlo localization, the simple particle filters in robot localization[10]. Fox used the Kullback-Leibler distance (KLD) to measure the approximation error introduced by the sample-based representation of the particle filter, and reduces the computational complexity[11]. Kwok *et al.* introduced real-time particle filters to make full use of observations by distributing the samples among the different observations during a filter update[12], and then enhanced real-time particle filters by adapting the size of the mixture using KLD sampling[13]. In order to make more efficient trade-off between accuracy, simplification and robustness, in this paper we introduce probability retracing to initialize particle sets, use consecutive window filtering to update particle sets, and refresh the particle set size according to the estimation efficiency.

The rest of the paper is organized as follows: in section 2 we first outline the traditional particle filters and the available technologies to improve its performance; then in section 3, the details about the proposed method are described and investigated. The simulation experiments are depicted in section 4 and the conclusion is drawn in the last section.

2 The particle filtering algorithms

In robot localization, the posterior probability density function (PDF) of robot state (including the robot location and its motion velocity) is:

$$B(x_t) = p(x_t|z_t, u_{t-1}, z_{t-1}, u_{t-2}, z_{t-2}, \cdots, u_0, z_0) \tag{1}$$

1) Supported by National Natural Science Foundation of P. R. China (60402030)
Received December 21, 2004; in revised form July 28, 2005

where x_t is the robot state at time t, u_{t-1} is the movement measure by the odometer during the interval $[t-1, t]$, and z_t is the observation of sensors at time t. Based on the Markov hypothesis, the PDF could be recursively updated according to the Bayesian rules, *i.e.*,

$$B(x_t) \propto p(z_t|x_t) \int p(x_t|x_{t-1}, u_{t-1}) B(x_{t-1}) \mathrm{d}x_{t-1} \tag{2}$$

as is shown in Fig. 1.

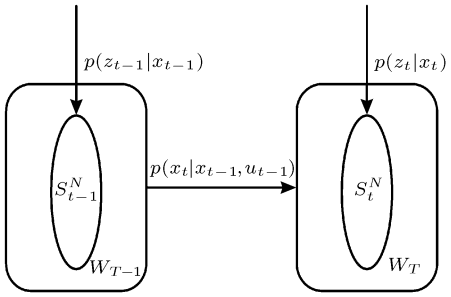

Fig. 1 Updating scheme of basic particle filters

Particle filters represent $B(x_t)$ with weighted sample sets $S_t^N = \{< x_t^i, w_t^i > | i = 0, 1, 2, \cdots, N\}$ (x is the state estimation, w is the corresponding weight, and the weighted samples $< X, w >$ are named particles). The particle sets are updated using the method of SISR. Details of this algorithm are the followings.

1) Importance sampling: draw samples from S_{t-1}^N according to the weights of particles; the combination of such samples X' is coincidental to $B(x_{t-1})$, which is the posterior PDF of the past time.

2) Movement updating: according to the move model $p(x'|x, u)$, update the samples in X' by the measured movement u_{t-1} and get the new samples set X.

3) Particle weighting: based on Bayesian rules, weight each sample x_t^i in X by observation model $w_t^i = p(z_t|x_t^i)$ and the observation data z_t.

Recursively calculating the above three steps generates the basic particle filters in robot localization, which may estimate the robot states on-the-fly. Further more, from the algorithm, one sees that particle filters do not specify the types of movement model or observation model, or the shape of environment maps. Therefore, particle filters are capable of solving the problem of robot localization. The accuracy and robustness of particle filters are closely related with the dimension of particle state and the size of particle sets. Unluckily, the computational complexity is proportional to the size of particle sets and increases exponentially with the dimension of particles state. Consequently, the focused improvement to particle filters is seeking for more efficient trade-off between accuracy, simplification and robustness of the particle filtering algorithms. Some available modifications are listed as follows.

A. Discarding sensor data

The idea of discarding the sensor data is to initialize the proper dimension of particles and enlarge the size of particle sets to get the off-line optimization of the estimation performance. Then during the real time running, one can update the particle filter as often as possible and discard the sensor data that could not be processed timely. Because this method will not make full use of the information collected by the robot on-the-fly, the accuracy and robustness of localization are not acceptable.

B. KLD sampling

One can make use of Kullback-Leibler distance to update the necessary particle size in a certain error bound between the sample-based maximum likelihood estimate and the current approximation of the true posterior[11]. Therefore, KLD sampling achieves improvement of real time performance while maintaining the localization accuracy of particle filter in robot localization. However, this method only meets the real-time requirements partially with the adaptive size of particles.

C. Real time particle filters

Kwok *et al.* distributes the samples among the different observation arrivals during a filter update[12]. Fig. 2 illustrates this approach. Several particle sets, named particle sets window, are maintained during the procedure, and they are combined as integrity to update the sample set by the sensor observations. This approach can realize the aim of real time while using all the observation data. However, the robustness is not satisfactory. As shown in Fig. 2 the particle sets windows are discrete, and the updating of particle sets does not make full use of the known information. For example, the information of set S_{t^1} is not used when updating the particle set S_{t^2}.

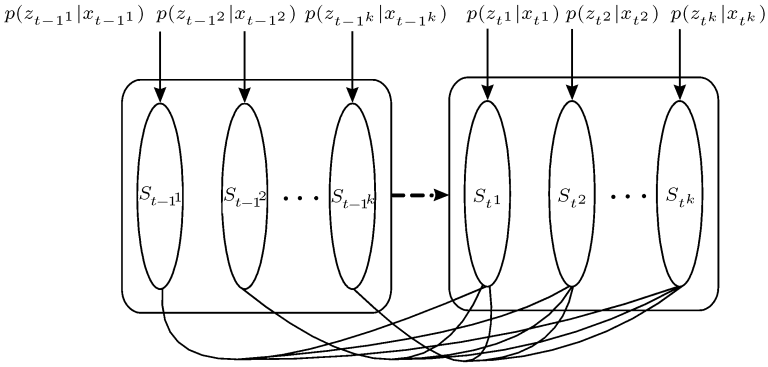

Fig. 2 Update scheme of real time particle filters

3 Novel adaptive particle filters

In particle filters, the difference between the estimation of the robot state presented by particle sets and the actual robot state is named the system uncertainty. The system uncertainty comes from tow aspects: the first is the uncertainty of the initial state of robot, and the second is the model error and the noise of sensor measures. During each iteration, particle filter uses the noised sensor measures to update the particle sets. Therefore it is required that the iteration should maintain the efficient estimation information in the previous particle sets while trying best to draw information from the noised sensor measures. Accuracy, simplification and robustness are considered by the following method of consecutive window particle filters.

A. Probability retracing

In particle filters, the un-normalized weight w is the probability that the actual state is at the sample. Usually the PDF of robot states is of generalized Gaussian distribution. So the more accurate the estimation sample is, the bigger the corresponding weight w is. Consequently, w indicates the efficiency of the estimation sample. At the same time, the average un-normalized weight $\bar{W}$ of all samples in one particle set S^N can be used to measure the efficiency of this particle set, which is named particle set efficiency. The bigger the particle set efficiency $\bar{W}$ is, the more accurate the state estimation of this particle set is.

This paper introduces probability retrieval to generate the initial particle set with big particle set efficiency. Let the robot be away from the obstacle L in the direction θ; then the obstacle will arrive at the robot by retracing L along the direction $-\theta$. For the observations are always with noise, the position that the obstacle retraces to is the probability estimation about the robot state. In practice, the obstacle retraces according to the observation models that generate the distribution of estimation states, which is named probability retracing. In a deterministic environment, probability retracing only rests on the observation at that time, and the more accurate the sensors are, the more efficient the probability retracing is. So we could use probability retracing to generate the initial particle set with high particle set efficiency. The procedure of this algorithm is as follows:

1) Find the direction of the initial robot direction in the map by the line-matching method in [14].

2) In the map, represent obvious edges of the obstacle along the observation direction θ with discrete sample set X_θ.

3) Retrace the samples in X_θ along the direction $-\theta$ to generate new sample set X'_θ.

4) Repeat steps 2) and 3), and retrace all the obvious obstacle edges along all observation directions to get sample set $X' = \bigcup_\theta X'_\theta$.

5) Weight the samples in X' by the method of K-D tree[15] and generate the initial particle set S_0.

The initial particle set generated by probability retracing has high particle set efficiency. At the same time, the initialization procedure need not to be operated in real time, so the probability retracing will not influence the real-time performance.

B. Consecutive window filtering

In the basic particle filters, the importance sampling is just applied to the particle set S_{t-1}^N, which means that the state estimation at next time only bases on the current estimation, other than any of the previous estimations. This is a first order Markov process, as shown in Fig. 1.

For the movement model and the sensor model are not accurate, and in reality the next state estimation is related to the previous estimations, one can update the estimation according to multi-estimation. The proposed consecutive window filtering is illustrated in Fig. 3.

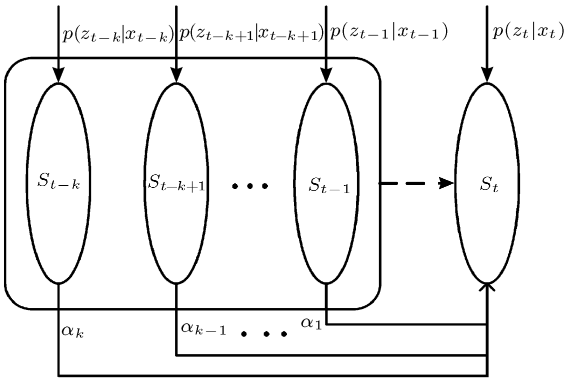

Fig. 3 Updating scheme of consecutive window filtering

In Fig. 3, the importance sampling of particle set is based on the past k times$'$ estimations

$$B(x_t) \propto \sum_{i=1}^{k} \alpha_i p(z_t|x_t) \int \cdots \int B(x_{t-i}) \prod_{j=t-i+1}^{t} p(x_j|x_{j-1}, u_{j-1}) \mathrm{d}x_{t-i+1} \cdots \mathrm{d}x_{t-1} \tag{3}$$

The past k times$'$ particle sets are combined to a particle set window. The relative weights of the particle sets in the particle set window are denoted as $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \cdots, \alpha_k]$, which is the normalized particle set efficiencies

$$\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \cdots, \alpha_k] = [\bar{W}_{t-1}, \bar{W}_{t-2}, \cdots, \bar{W}_{t-k}] \Big/ \sum_{i=t-1}^{t-k} \bar{W}_i \tag{4}$$

Compared to the real time particle filters, the particle set S_t^N in consecutive window filtering at time t derives from the window of particle sets, which includes particle sets $S_{t-k}^N, S_{t-k+1}^N, \cdots, S_{t-1}^N$, other than the particle sets $S_{\lfloor t/k \rfloor \times k-k}^N, S_{\lfloor t/k \rfloor \times k-k+1}^N, \cdots, S_{\lfloor t/k \rfloor \times k-1}^N$. Since the particle sets used to update in consecutive window filtering are latter than those in real time particle filters, consecutive window filtering is more efficient and more robust than real time particle filters.

C. Adaptive particle set size

In order to improve the efficiency of the algorithm further, KLD sampling for reference is used to update the particle set size. In KLD sampling, one divides the map into small grids. In the thresholds ε and δ, one calculates the next particle set size according to the numbers of grids b, which is held by the current particle set

$$N = \frac{b-1}{2\varepsilon} \left\{ 1 - \frac{2}{9(b-1)} + \sqrt{\frac{2}{9(b-1)}} z_{1-\delta} \right\}^3 \tag{5}$$

In consecutive window filtering, one calculates N in each particle set. Then according to the particle set efficiencies, one makes use of the weighted sum of N as the next particle set size, which is

$$N_{t+1} = \sum_{i=0}^{k-1} \alpha_i \frac{b_{t-i}-1}{2\varepsilon} \left\{ 1 - \frac{2}{9(b_{t-i}-1)} + \sqrt{\frac{2}{9(b_{t-i}-1)}} z_{1-\delta} \right\}^3 \tag{6}$$

where b_{t-i} is the number of grids held by particle set S_{t-i} in the map.

The combination of probability retracing, consecutive window filtering and the adaptive particle set size forms the novel adaptive particle filters, whose procedure is listed as follows,

1) Employ probability retracing to generate the initial particle set S_0;

2) Update particle filters using equation (3) to get the current particle set S_t, and calculate the corresponding particle set efficiency $\bar{W}_0$;

3) Based on equation (6), calculate the next particle set size N_{t+1};

4) $t = t + 1$, and go back to 2).

4 Experiment results

The computer to do experiments has a PIII800 CPU and 256M memory. The map of the tested office room is illustrated by a 600×400 bitmap, and each pixel denotes the space of 3cm×3cm. The grid in the map is 5×5, which denotes a 15cm×15cm space in the office room. The sensor data and the movement measures are collected by the robot with one odometer and four sonar sensors. Both the odometer model and the sonar sensor model agree with [16]. In the experiments, the number of particles updated in each iterative is named the valid updating particle number.

When the valid updating particle numbers of different algorithms are identical, the proposed novel adaptive particle filter (APF) is compared with the basic particle filter (MCL), and the other improved particle filters: KLD sampling, real time particle filters (RTPF). From Fig. 4, one can see that when the valid updating particle number is not too large, the proposed particle filters consume lest time and the real time performance is the best. However, when the valid particle number arises to some large threshold (10E5 for example), KLD sampling outperforms the proposed particle filters. The improvement by consecutive window filtering is not prominent compared to that from the adaptive particle set size.

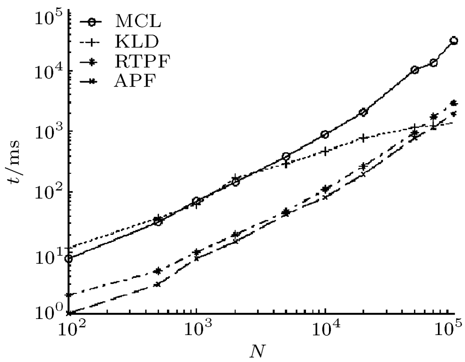

Fig. 4 Time consumption of each iterative in different particle filters

The robustness performance between the novel adaptive particle filters and the basic particle filters is compared in Fig. 5. For the proposed particle filters employ probability retracing to generate the

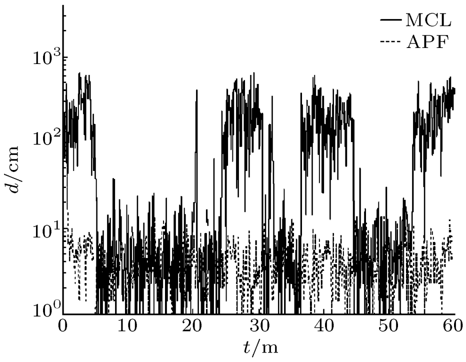

Fig. 5 The comparison of robustness between the novel adaptive particle filters and the basic particle filters

initial particle set, in the left end of the curves, its estimated error is lower than the basic particle filters. The result means that the proposed particle filters can achieve accurate initial location. One can also see that the robustness performance of the novel particle filters is excellent compared to the basic particle filters.

The above experiments demonstrate that the proposed adaptive particle filters can improve the robustness and reduce the computational complexity in robot localization compared with the basic particle filters and other improvement methods, and can successfully solve the nonlinear, non-Gaussian state estimation problem of robot localization.

5 Conclusion

In this article, we propose the adaptive particle filters, which employ the technologies including probability retracing, consecutive window filtering, and adaptive particle set size. The novel method makes good trade-off among the accuracy, simplification and robustness performance of particle filters. Computer simulations demonstrate that the proposed method can successfully solve the nonlinear, non-Gaussian state estimation problem of robot localization.

References

1 Burgard W, Fox D, Hennig D, Schmidt T. Estimating the absolute position of a mobile robot using position probability grids. In: Proceedings of the Fourteenth National Conference on Artificial Intelligence (AAAI-96), 1996. 896～901
2 Roumeliotis S I, Bekey G A. Bayesian estimation and Kalman filtering: A unified framework for mobile robot localization. In: Proceedings of the IEEE International Conference on Robotics & Automation. San Francisco, CA, USA, 2000. 2985～2992
3 Jensfelt P, Wijk O, Austin D, Andersson M. Feature based condensation for mobile robot localization. In: Proceedings of the IEEE International Conference on Robotics & Automation. San Francisco, CA, USA, 2000. 2531～2537
4 Gordon N J, Salmond D J, Smith A F M. A novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings on Radar and Signal Processing*, 1993, **140**: 107～113
5 Doucet A, Freitas N, Gordon N. Sequential Monte Carlo Methods in Practice. New York: Springer, 2001
6 Doucet A, Godsill S, Andrieu C. On sequential Monte Carlo sampling methods for Bayesian filtering. *Stat. Comput*, 2000, **3**: 197～208
7 Leonard J, Durrant-Whyte H. Directed Sonar Sensing for Mobile Robot Navigation. Cambridge, MA, USA: Kluwer Academic Publishers, 1992
8 Schiele B, Crowley J. A comparison of position estimation techniques using occupancy grids. In: Proceedings of ICRA-94. 1994. 1628～1634
9 Gutmann J S, Schlegel C. AMOS: Comparison of scanmatching approaches for self-localization in indoor environments. In: Proceedings of Euromicro. IEEE Computer Society Press. 1996,
10 Thrun S, Fox D, Burgard W. Robust Monte Carlo localization for mobile robots. *Artificial Intelligence*, 2000, **128**(1-2): 99～114
11 Fox D. KLD-sampling: Adaptive particle filters. *International Journal of Robotics Research*, 2003, **22**(12): 985～1003
12 Kwok C, Fox D, Meila M. Real time particle filters. *Proceedings of the IEEE*, 2004, **3**(92): 469～484
13 Kwok C, Fox D, Meila M. Adaptive real-time particle filters for robot localization. In: Proceedings of IEEE International Conference on Robotics and Automation. Taipei, Taiwan, 2003. **2**: 2836～2841
14 Gutmann J, Weigel T, Nebel B. A fast, accurate, and robust method for self-localization in polygonal environments using laser-range-finders. *Advanced Robotics Journal*, 2001, **14**(8): 651～668
15 Moore A. Efficient Memory-based Learning for Robot Control. Cambridge: University of Cambridge, 1990
16 Ioannis M, Rekleitis. Cooperative Localization and Multi-Robot Exploration. Montreal, Quebec, Canada: School of Computer Science, McGill University, 2003

JIANG Zheng-Wei Received his master degree from Tsinghua University in 2005. His research interests include information retrieval and visual data mining.

GU Yuan-Tao Received his Ph. D. degree from Tsinghua University in 2003. He is now a lecturer of the Department of Electronic Engineering, Tsinghua University. His research interests include signal processing and image retrieval.