Parallel Tempered Particle Filter

Dimitri Marinakis
Department of Computer Science, University of Victoria
dmarinak@gmail.com

***Abstract*— In this paper, we present the concept of running multiple configuration-exchanging particle filters in parallel; each characterizing an increasingly 'smoothed' version of the target density *via* the technique of sampling at high temperatures. This technique is used in Markov Chain Monte Carlo to improve mixing where it is known as parallel tempering.**

I. Introduction

In this paper, we present the application of parallel tempering to particle filters in the domain of mobile robotics. The concept is to run multiple configuration-exchanging particle filters in parallel; each at a different *temperature*. The filter running at temperature $\tau = 1$ attempts to track the target distribution, while those running at higher temperatures track increasingly 'smoothed' versions of the target distribution. The motivation for employing filters at a temperature $\tau > 1$ is that the over-dispersed PDFs can help maintain diversity which can be passed down to the 'colder' filters. We present an example of how a parallel tempered particle filter (PTPF) can be applied to the problem of mobile robot localization. The approach, however, should also apply to the more general problem of SLAM in robotics; *e.g.* to aid loop closing.

We first give some background on the application of parallel tempering to Markov Chain Monte Carlo. We then provide details for how parallel tempering could be implemented in particle filters and applied to localization in mobile robotics.

II. Parallel Tempering Markov Chain Monte Carlo (MCMC)

Parallel tempering [3] is a MCMC variant in which multiple configuration-exchanging chains of different *temperatures* are simulated in parallel. The temperature of a chain can be thought of as specifying the relative 'smoothness' of its target distribution. Usually, a chain C_k of temperature τ, will use the density: $\pi_k = (\pi)^{\frac{1}{\tau}}$. While the lowest temperature chain attempts to sample from the target distribution, π, the higher temperature chains sample potentially easier to characterize versions of the original target distribution.

During a simulation, after a number of within chain proposals, two consecutive chains C_i and C_{i+1} are selected randomly and their current configurations X_i and X_{i+1} are exchanged (or not) according the Metropolis-Hastings [7] [4] acceptance ratio:

$$\alpha = \min\left(1, \frac{\pi_i(X_{i+1})\pi_{i+1}(X_i)}{\pi_i(X_i)\pi_{i+1}(X_{i+1})}\right) \quad . \tag{1}$$

Parallel tempering achieves good performance by allowing high temperature chains to make fast, less-restrained exploration of the underlying probability landscape. Promising realizations discovered by these 'hot' chains are fed down to colder chains, and ultimately to the principle chain. The objective is faster mixing than in the single chain variant, and hence more complex target distributions require less computational effort to characterize through sampling. Parallel tempering MCMC is used in fields such as physics and biology and has been applied to localization in hybrid sensor network / mobile robot systems [6]. The concept of exploiting relaxed versions of the final problem is related to simulated annealing [5], although in simulated annealing the temperature is managed in a sequential fashion as opposed to in parallel.

In the next section, we consider the application of parallel tempering to a particle filter, which can be considered an on-line variant of MCMC.

III. A Parallel Tempered Particle Filter for Mobile Robot Localization

A. Prior Work: Monte Carlo Localization (MCL)

First we briefly outline the application of a particle filter to robot localization such as was presented by Fox *et al.* [2] [1]. See these references for a detailed description of the approach.

MCL recursively computes the density $p(x_k|Z^k, U^{k-1})$ for the robot's location x_k at time step k given all landmark measurements obtained up to that point Z^k and all motion control inputs given to the robot up to that point U^{k-1}. Given the robot's location at $t = k - 1$ the approach computes the conditional distribution for the robot at $t = k$ using a motion model and a measurement model. This is done using using a predictive (or proposal) phase that samples directly from the motion model and then an update phase that uses importance sampling to correct for the influence of the latest landmark observation using the measurement model.

B. A Tempered Particle Filter (TPF)

The same steps used in MCL can be followed here to recursively compute an over-dispersed density for the location of the robot at time step $t = k$:

$$\pi_k(x_k)^{\frac{1}{\tau}} = p(x_k|Z^k, U^{k-1})^{\frac{1}{\tau}}$$

for a given temperature τ.

1) Tempered Predictive Phase: To apply tempering to the predictive phase of MCL, we wish to sample from an over-dispersed motion model; *i.e.* at time step k, the predictive particle i is generated as follows:

$$q_k^i \sim p(x_k^i | x_{k-1}^i, u_{k-1})^{\frac{1}{\tau}}$$

where u_{k-1} represents the control input for the motion of the robot and x_k^i is the robot location sample associated with particle i.

2) Tempered Update Phase: Likewise, during the update phase, the weight assigned to each predictive particle i prior to re-sampling is computed:

$$w_k^i \propto p(z_k | q_k^i)^{\frac{1}{\tau}}$$

where z_k represents the landmark observed at time step k. New samples are then drawn, with replacement, from the weighted set of predictive samples leading to a set of particles that give a representation of the density at temperature τ for this time step.

3) Recursively Maintain Ratio to Density at $\tau = 1$: Additionally, we recursively compute and maintain the ratio of the posterior for each particle i with respect to the density at temperature $\tau = 1$:

$$\begin{aligned} r_k^i &= \frac{\pi_k(x_k^i)}{\pi_k(x_k^i)^{\frac{1}{\tau}}} \\ &= r_{k-1}^i \left[\frac{p(x_k^i | x_{k-1}^i, u_{k-1}) p(z_k | x_k^i)}{p(x_k^i | x_{k-1}^i, u_{k-1})^{\frac{1}{\tau}} p(z_k | x_k^i)^{\frac{1}{\tau}}} \right]. \end{aligned} \quad (2)$$

For a given set of particles at a temperature $\tau > 1$ and $t = k$ we can recover an estimate for the target distribution ($\tau = 1$) by weighing the set of particles by their respective r_k^i values, normalizing, and then re-sampling as in the update step of MCL.

C. A Parallel Tempered Particle Filter (PTPF)

The approach described above allows us to compute a particle based density for the robot's location for an arbitrary temperature. Here, we will show that by extending this approach to multiple temperatures we can construct a parallel tempered particle filter.

Given a set of temperatures $T = \{\tau_1, \tau_2, \ldots, \tau_M\}$, we run in parallel a number of instances $f_1, f_2, \ldots, f_M$ of the TPF; each instance f_m maintaining the density for the robot's location at the temperature τ_m. Additionally, for each particle in each filter f_m, we maintain the density ratio to $\tau = 1$ for *each* temperature $\tau \in T$ (see Equation 2). Hence, for each f_m, the following matrix R_k^m of ratio values is maintained at time step k:

$$R_k^m = \begin{bmatrix} r_k^{1,\tau_1} & \ldots & r_k^{1,\tau_M} \\ \vdots & & \\ r_k^{N,\tau_1} & \ldots & r_k^{N,\tau_M} \end{bmatrix}$$

where N is the number of particles in f_m and the first superscript on r corresponds to the particle number.

This ratio information $R_k^1, \ldots, R_k^m$ now allows the exchange of particles among any two TPFs f_i, f_j at time step k. According to the M-H acceptance ratio, (see Equation 1), particle i of filter f_i and particle j of filter f_j may be exchanged with probability:

$$\begin{aligned} \alpha &= \min\left(1, \frac{\pi_k(x_k^j)^{\frac{1}{\tau_i}} \pi_k(x_k^i)^{\frac{1}{\tau_j}}}{\pi_k(x_k^j)^{\frac{1}{\tau_j}} \pi_k(x_k^i)^{\frac{1}{\tau_i}}}\right) \\ &= \min\left(1, \frac{r_k^{j,\tau_j} r_k^{i,\tau_i}}{r_k^{j,\tau_i} r_k^{i,\tau_j}}\right) \quad . \end{aligned}$$

When applied to MCMC, a typical implementation of parallel tempering allows the exchange of configurations among two chains of consecutive temperature. The analogous implementation with PTPFs would see a round of potential exchanges among filters of consecutive temperatures at the end of each time step. For example, M filters could be run, each at an increased temperature and each with N particles. Then after each time step, starting with the 'hottest' filter, each particle i could be tested for an exchange with the corresponding particle i in the filter f_{z-1}.

There have been advancements in the application of particle filters to mobile robotics since the introduction of MCL, such as an improved proposal mechanism [9] [8]. The approach we have described here for extending a particle filter using parallel tempering should apply as long as the motion and measurement model can be parameterized.

IV. Conclusion

In this paper, we presented the concept of parallel tempered particle filters and provided an example of their application to the task of localization in mobile robotics. PTPFs should be more robust to the particle depletion problem in which a filter becomes over confident and loses areas of support for the target distribution. The technique should be especially helpful where complex, multi-modal distributions are common, such as in range-only SLAM. Future work will look at validating the approach experimentally.

References

[1] F. Dellaert, D. Fox, W. Burgard, and S. Thrun. Monte carlo localization for mobile robots. In *ICRA'99*, May 1999.

[2] D. Fox, W. Burgard, F. Dellaert, and S. Thrun. Monte carlo localization: Efficient position estimation for mobile robots. In *AAAI'99*, July 1999.

[3] C. J. Geyer. Markov chain monte carlo maximum likelihood. In *Computing Science and Statistics: Proc. of the 23rd Symposium on the Interface*, pages 156–163, 1991.

[4] W. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57:97–109, 1970.

[5] S. Kirkpatrick. Optimization by simulated annealing: Quantitative studies. *Journal of Statistical Physics*, 34(5-6):975–986, 1984.

[6] D. Marinakis, D. Meger, I. Rekleitis, and G. Dudek. Hybrid inference for sensor network localization using a mobile robot. In *AAAI'07*, pages 1089–1094, Vancouver, Canada, July 2007.

[7] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller. Equation of state calculation by fast computing machines. *Journal of Chemical Physics*, 21:1087–1092, 1953.

[8] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In *IJCAI'03*, Acapulco, Mexico, 2003.

[9] R. van der Merwe, A. Doucet, N. de Freitas, and E. Wan. The unscented particle filter. In *Neural Information Processing Systems*, Dec. 2000.