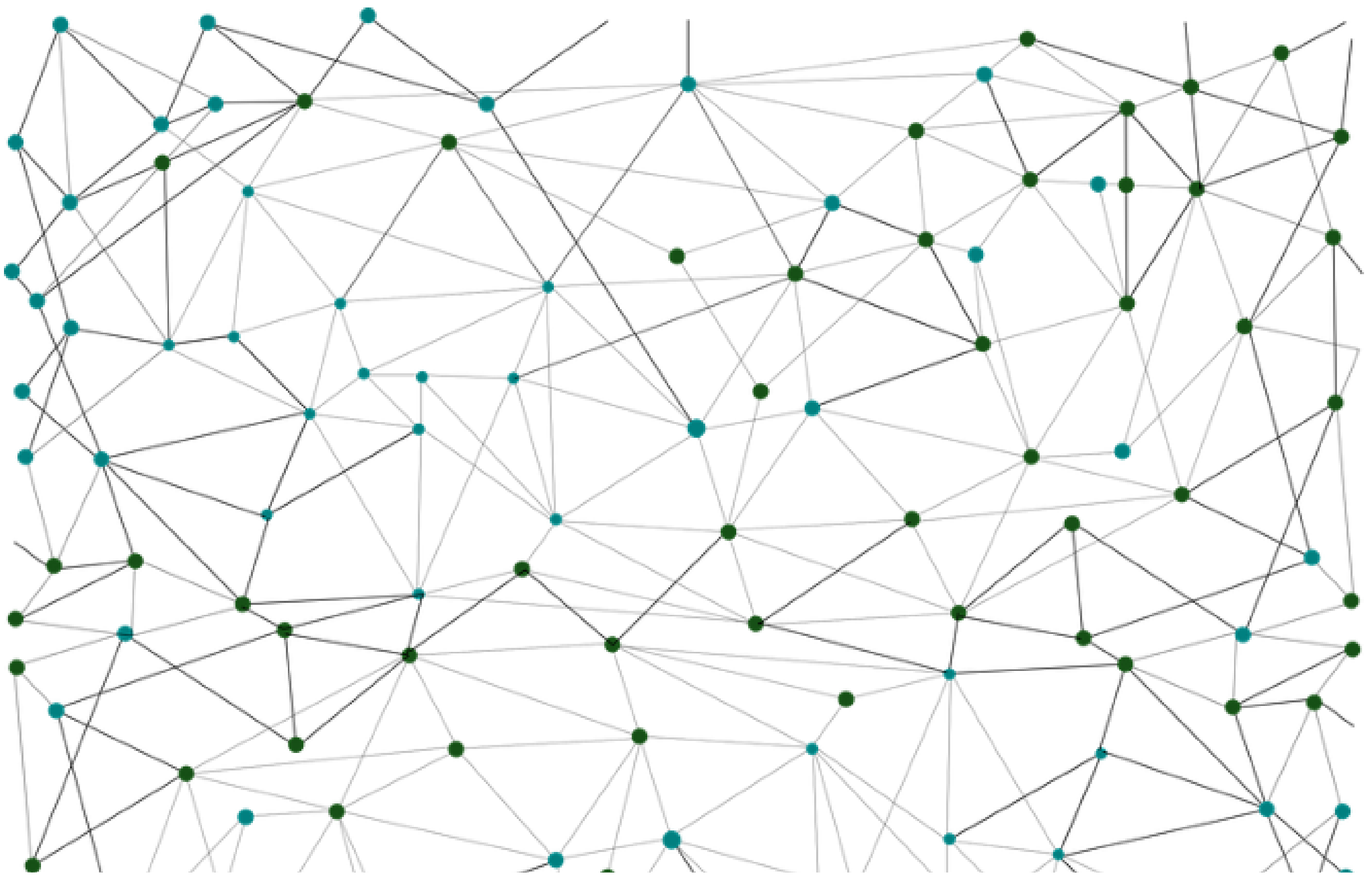
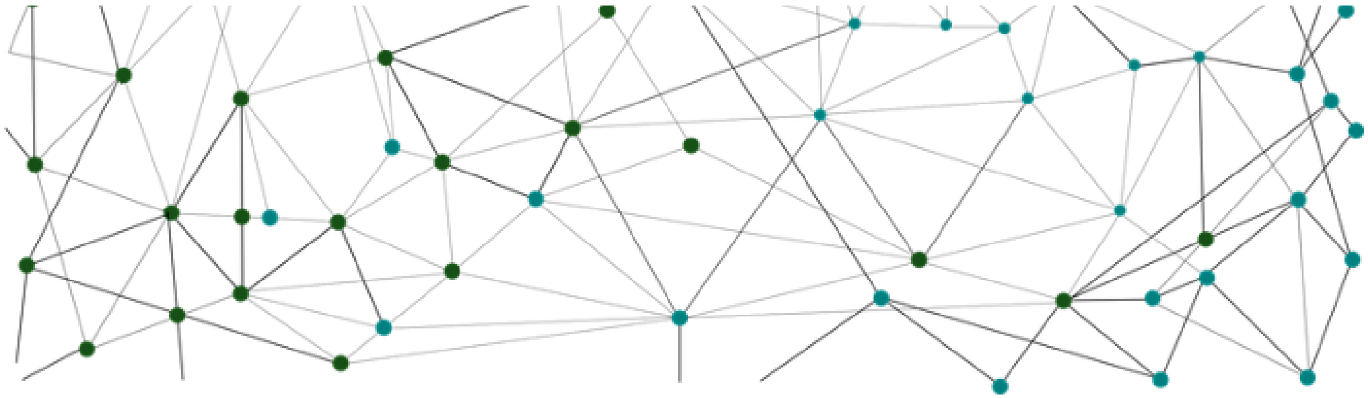
499.9K Followers · About Follow

Upgrade





Particle Filter : A hero in the world of Non-Linearity and Non-Gaussian

Sharath Srinivasan Aug 14, 2019 · 8 min read ★

The superiority of particle filter technology in nonlinear and non-Gaussian systems determines its wide range of applications. In addition, the multi-modal processing capability of the particle filter is one of the reasons why it is widely used. Internationally, particle filtering has been applied in various fields.

- In the field of economics, it is used in economic data forecaThe superiority of particle filter technology in nonlinear and non-Gaussian systems determines its wide range of applications. In addition, the multi-modal processing capability of the particle filter is one of the reasons why it is widely used. Internationally, particle filtering has been applied in various fields.sting. Numbers are present everywhere, and when they are collected and recorded we refer to them as data. Machine learning is the science of learning mathematical models from data. Such models, once learned from data

- In the military field, it has been applied to radar tracking airborne objects, air-to-air, air-to-ground passive tracking
- In the field of traffic control it is applied to car or people video monitoring.
- In the field of visual analytics, a new pattern recognition model for segmenting and tracking lip contours in video sequences has been developed using particle filters.
- It is also used for global positioning of robots.

> *Questions which were unanswerable by traditional analysis methods are now being solved by means of particle simulation.*

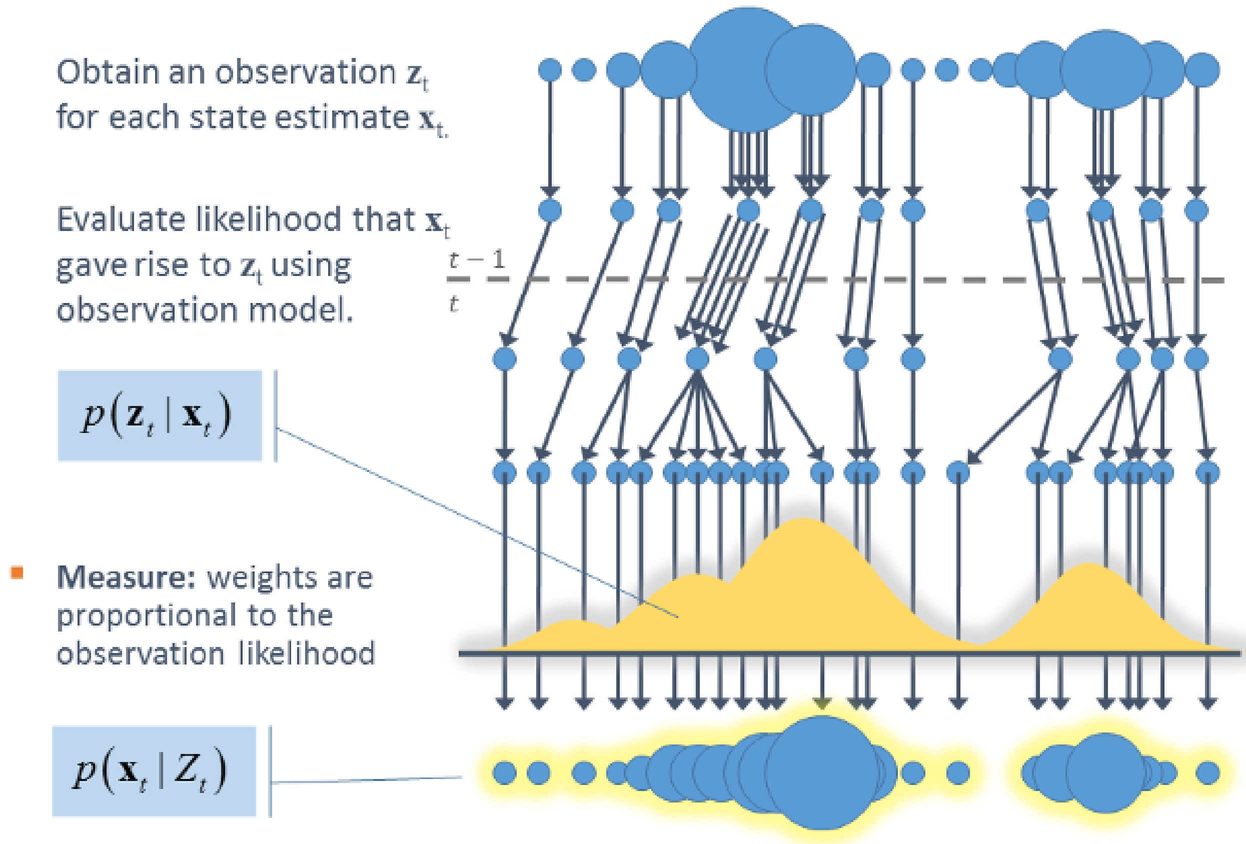
In the model selection, fault detection and diagnosis of dynamic systems, particle-based hypothesis testing, particle multi-model, particle likelihood ratio detection, and other methods have emerged.

In terms of parameter estimation, the static parameters are usually taken as part of the extended state vector, but since the parameters are static, the particles will quickly degenerate into a sample. To avoid degradation, the commonly used methods artificially increase the dynamic noise for static parameters.

The Kernel smoothing method, and the point estimation method proposed by Doucet et al. avoids direct sampling of parameters, and directly estimates the unknown parameters using maximum likelihood estimation (ML) and maximum expected value (EM) algorithm under the particle framework.

Demystifying the mysterious particle filtering

The idea of the particle filter (PF: Particle Filter) is based on *Monte Carlo methods*, which use particle sets to represent probabilities and can be used in any form of state space model. The core idea is to express its distribution by extracting random state particles from the posterior probability. It is a sequential importance sampling method (Sequential Importance Sampling).



The breakdown of the concept

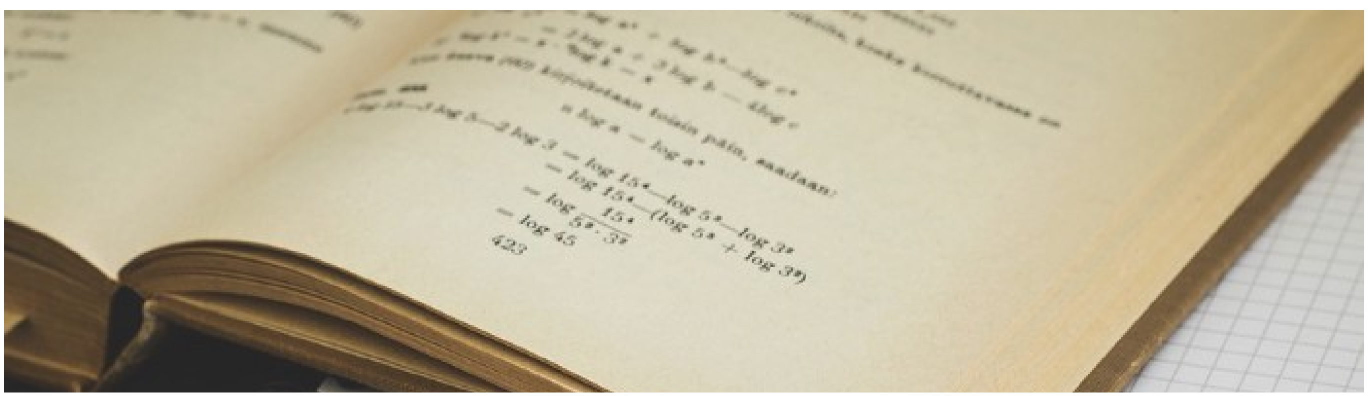
In simple terms, the particle filtering method refers to the process of obtaining the state minimum variance distribution by finding a set of random samples propagating in the state space to approximate the probability density function and replacing the integral operation with the sample mean.

> *The sample here refers to the particle, and when the number of samples $N \rightarrow \propto$, it can approximate any form of probability density distribution.*

Although the probability distribution in the algorithm is only an approximation of the real distribution, due to the non-parametric characteristics, it can get rid of the constraint that the random quantity must satisfy the Gaussian distribution when solving the nonlinear filtering problem, and can express a wider distribution than the Gaussian model.

It also has a stronger ability to model the nonlinear characteristics of variable parameters. Therefore, particle filtering can accurately express the posterior probability distribution based on observation and control quantities, which can be used to solve the SLAM problem.

Development of Particle filtering



Markov Chain Monte Carlo improvement strategy

The Markov Chain Monte Carlo (MCMC) method produces samples from the target distribution by constructing Markov chains with good convergence. In each iteration of the SIS, the MCMC is combined to move the particles to different locations to avoid degradation, and the Markov chain can push the particles closer to the *probability density function* (PDF). Make the sample distribution more reasonable.

Unscented Particle Filter (UPF)

The Unscented Kalman Filter (UKF) was proposed by Julier et al. The EKF (Extended Kalman Filter) approximates the nonlinear term using a first-order Taylor expansion and approximates the state distribution with a Gaussian distribution. The UKF is similar to the EKF, with a Gaussian distribution approximating the state distribution, but without linearization, only a few samples called Sigma points are used.

After these points pass the nonlinear model, the obtained mean and variance can be accurate to the second-order term of the nonlinear expansion Taylor, which makes the nonlinear filtering more accurate. Merwe et al. proposed the use of UKF to generate the importance distribution of PF, called Unscented Particle Filter (UPF). The importance distribution generated by UKF is larger than the overlap of the real state PDF, and the estimation accuracy is higher.

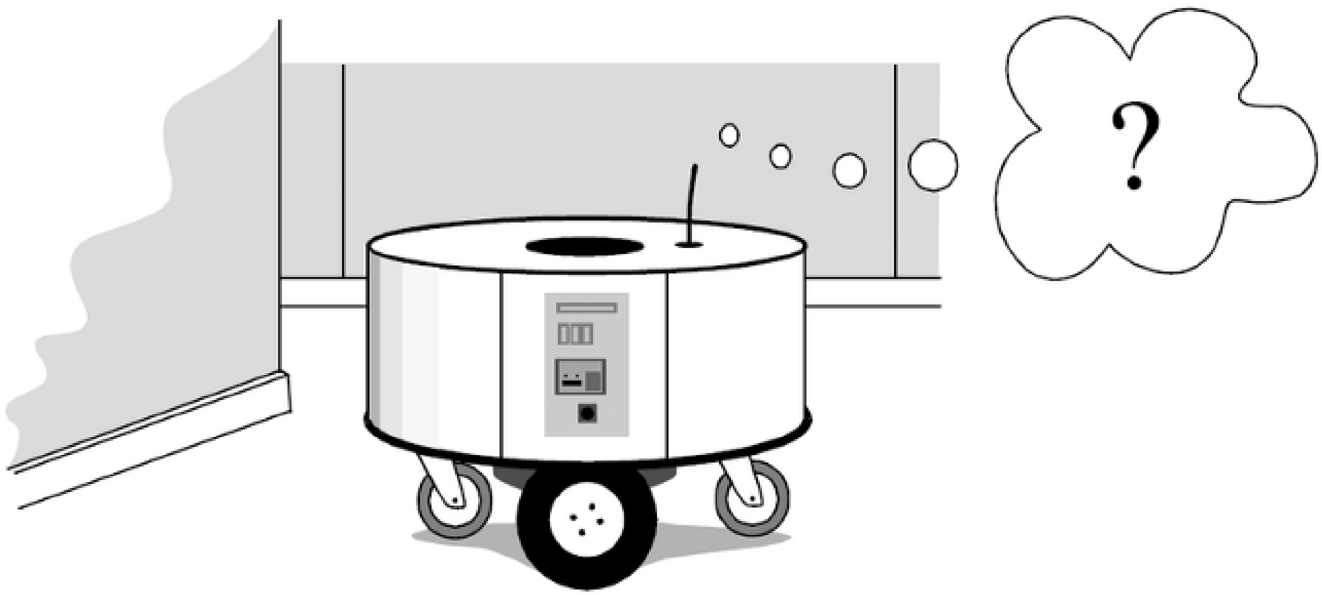
Particle Filter in Data Science and Robotics

Particle filters are now widely employed in the estimation of models for financial markets, in particular for stochastic volatility models.

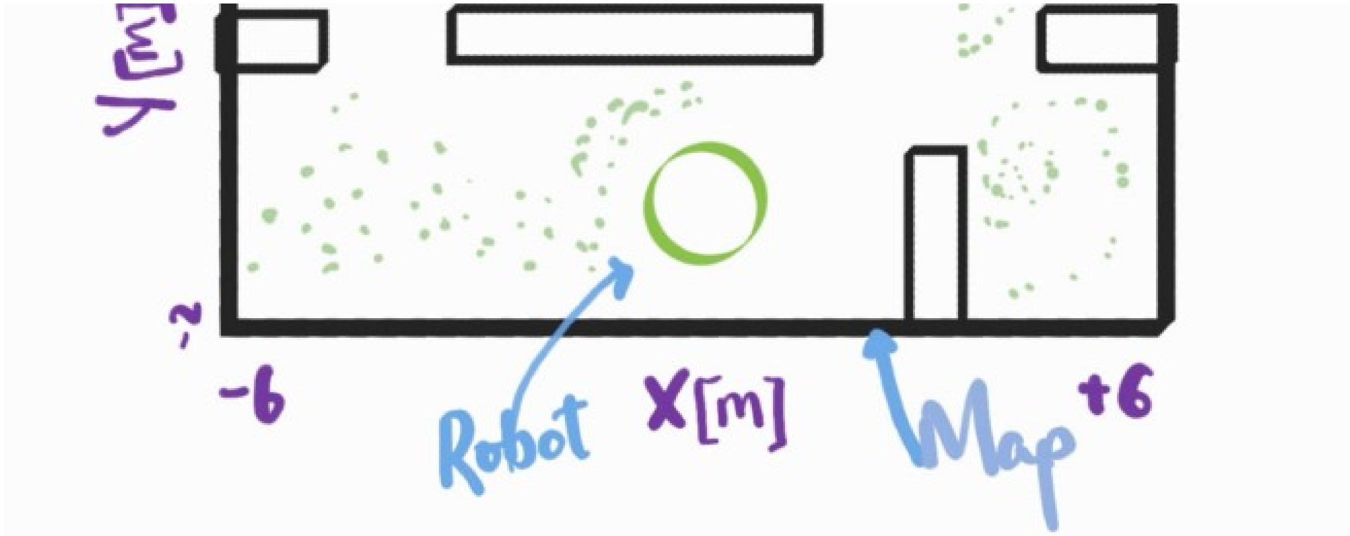
> *Particle filters methods are recursive Bayesian filters which provide a convenient and attractive approach to approximate the posterior distributions when the model is nonlinear and when the noises are not Gaussian.*

These techniques provide general solutions to many problems, where linearisation and Gaussian approximations are intractable or would yield too low performances. Non-Gaussian noise assumptions and incorporation of constraints on the state variables can also be performed in a natural way. Moreover, particle filter methods are very flexible, easy to implement, parallelizable and applicable in very general settings.

Robot Localization using Particle filters



An Indoor robot navigating its way inside a office

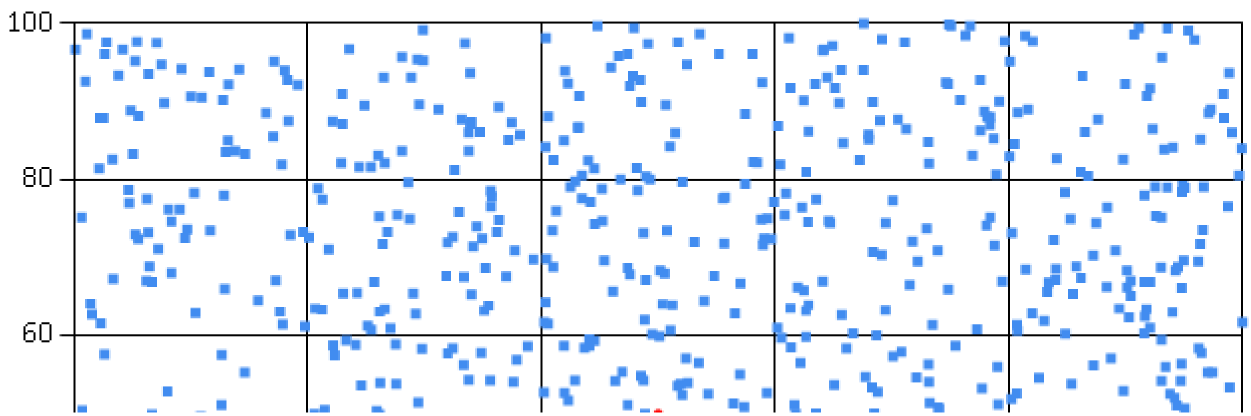


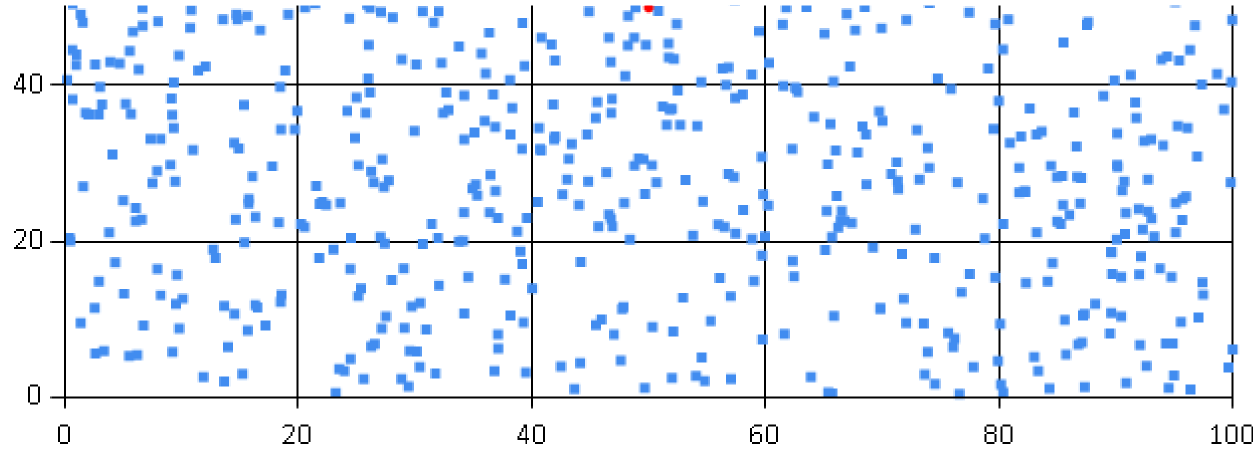
The lost robot on a map with the approximation of its location

Particle filter solves many problems in applied robotics. Assume we might have moving objects that we want to track. Maybe the objects are fighter jets and missiles, or maybe we are tracking people playing cricket in a field. It doesn't really matter. Let's think about the characteristics of a three-dimensional robotic problem:

- **Multimodal**: We want to track zero, one, or more than one object simultaneously
- **Occlusions**: One object can hide another, resulting in one measurement for multiple objects.

- **Non-linear behavior**: Aircraft are buffeted by winds, balls move in parabolas, and people collide into each other.
- **Non-linear measurements**: Radar gives us the distance to an object. Converting that to an (x,y,z) coordinate requires a square root, which is nonlinear.
- **Non-Gaussian noise:** as objects move across a background the computer vision can mistake part of the background for the object.
- **Continuous:** the object's position and velocity (i.e. the state space) can smoothly vary over time.
- **Multivariate**: we want to track several attributes, such as position, velocity, turn rates, etc.
- **Unknown process model**: we may not know the process model of the system

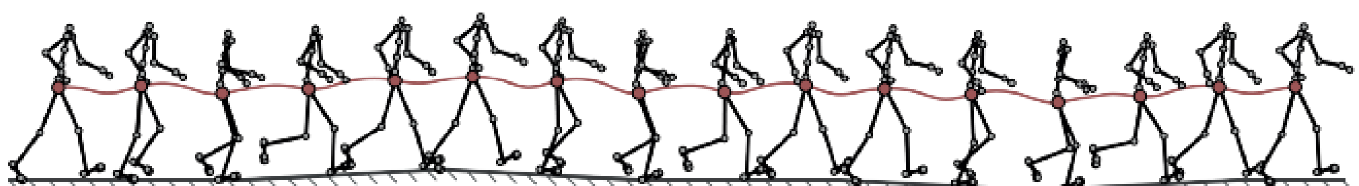




The initial assumption of the localized pose is noisy as we can see the points spread across the grid (map). As the robot moves further the particles converge which explains the convergence of the particles.

A variety of robots say an indoor robot navigating in a warehouse could use particle filters to localize itself based on the input from a range-finding sensor such as a 2D-Laser scanner or Particle Filters for self driving car could be applied to fuse sensory input and identify lane markings on the road. A drone could use particle filter to optimise optical flow and.

Reinforcement Learning algorithm and Particle Filters



Reinforcement learning, in the context of artificial intelligence, is a type of dynamic programming that learns and perfects an algorithm using a system of reward and punishment.

Reinforcement Learning(RL) algorithm based on Particle Filters are computationally inexpensive and have a very low memory footprint. RL algorithm that includes direct global policy search, based on particle filters, would perform a search in the policy space defined by the selected policy parameterization.

The particle filter could be a core concept of the RL algorithm, could be account for guiding the exploration and exploitation, by creating particles, each of which represents a whole policy. Due to the ability of particle filters to perform a global search, the resulting RL algorithm should be capable of direct global search in policy space, which is a significant improvement over traditional local search based policy RL.

Algorithm 1 Reinforcement Learning based on Particle Filters (RLPF)

Input: parameterized policy π, policy parameter space Θ, reward function $R(\theta)$ where $\theta \in \Theta$, reward transformation function g, total number of trials N, exploration probability function $P_{explore}(n)$, initial noise ϵ_0, noise decay factor λ.

Let $S = \{\,\}$ {A set of policy particles}
for $n = 1$ **to** N **do**
  Draw $v \sim U(0,1)$ {Uniform sampling $\in$ (0,1)}
  if $v < P_{explore}(n)$ **then**
    {Exploration step: global random sampling}
    Draw $\theta_n \sim U(\Theta)$ {Uniform sampling}

else
 {Exploitation step: particle filter sampling}
 Let $h(0) = 0$
 for $i = 1$ **to** $|S|$ **do**
  $w_i = \frac{g(R_i)}{\sum_{j=1}^{|S|} g(R_j)}$ {Importance weight}
  $h(i) = h(i-1) + w_i$ {Aux. function}
 end for
 Draw $z \sim U(0,1)$ {Uniform sampling}
 Let $y = h^{-1}(z)$
 Let $k = \lceil y \rceil$ {$\lceil \cdot \rceil$ is the ceiling function}
 Select policy particle $p_k = \langle \theta_k, \tau_k, R_k, w_k \rangle$
 Let $\epsilon_n = \epsilon_0 \lambda^{(n-L-1)}$ {noise with exp. decay}
 Let $\theta_n = \theta_k + \epsilon_n$
end if
Perform trial(s) $\tau_n(\pi(\theta_n))$
Estimate return R_n
Let $w_n = 0$
Create new policy particle $p_n = \langle \theta_n, \tau_n, R_n, w_n \rangle$
$S = S \cup \{p_n\}$
end for

This adds value as it inherits many advantages from particle filters, among which:

- It is very simple to implement and can be realized easily in embedded systems for online learning
- It can use adaptive computation depending on the available resources (both time- and CPU-wise) by changing the value of the σ parameter

- It can concentrate the effort of the RL exploration on the most important parts of the policy space,
- It can exhibit adaptive convergence rate depending on the requirements for precision and time, by changing the initial noise level and the decay factor λ

The Detriment of Particle Filtering

Although the particle filter algorithm can be used as an effective means to solve the SLAM problem, there are still some problems in the algorithm. The main problem is that a large number of samples are needed to closely approximate the posterior probability density of the system. The more complex the environment the robot faces, the more samples are needed to describe the posterior probability distribution, and the more complex the algorithm.

Therefore, an adaptive sampling strategy that can effectively reduce the number of samples is the focus of the algorithm. In addition, the re-sampling phase can result in loss of sample validity and diversity, leading to sample depletion. How to maintain the validity and diversity of particles and overcome the depletion of samples is also re-sampling the focus of this algorithm.

In reinforcement learning, particle filter as a member of global search methods, generally requires more trials in order to converge, because the scope of the search is the largest possible — the whole policy space.

Particle filters do not have a strict proof of convergence. In theory, we have traded a 'proof for local convergence' with a global search method which has no proof of global convergence but is at least guaranteed not to get stuck at local optima.

Reinforcement Learning   Localization   Self Driving Cars   Algorithms   Robotics