Particle Filters

A. Giavaras

Contents

1	Particle Filters	2
2	Introduction to Particle Filtering	2
2.1	Mathematical Formulation	3
2.2	Monte Carlo sampling	4
2.3	Importance sampling	5
2.4	Sequential Importance Sampling (SIS)	5
3	Rao-Blackwellized Particle Filter	7
4	Summary	7
4.1	Questions	7
4.2	Assignements	13
5	Answers	14

1 Particle Filters

Gaussian filtering is a useful technique in order to perform nonlinear filtering. Popular techniques that use Gaussian filtering are the extended and unscented Kalman filtering. However, these methods do not perform well when

- The models are highly nonlinear
- When the posterior distribution is significantly non-Gaussian e.g. a multimodal density

For these kinds of problems we need a different type of approximation to the posterior density $p(\mathbf{x}_t|\mathbf{z}_t;\mathbf{u}_t)$. One such approach is the particle filter method that we will discuss in this section, see [1] for a short introduction. Particle filters are numerical techniques for approximating posterior probabilities in partially observable controllable Markov chains with discrete time [1]. The idea behind this filtering technique is to aquire point-wise estimates, called samples, of the posterior $p(\mathbf{x}_k|\mathbf{y}_{1:k})$. Then, as the number of samples increases, the accuracy of the approximation increases.

2 Introduction to Particle Filtering

Particle filters have solved several hard perceptual problems in robotics. Concretely, particle filters were able to solve two important problems; the global localization problem and the kidnapped robot problem, see [1] and references therein. Another example where the particle filters approach is used successfully is the simultaneous localization and mapping (SLAM) problem [1]. In particular, FastSLAM has been demonstrated to solve problems with more than 100000 dimensions in real-time [1].

Remark 2.1. **Kidnapped Robot Problem**

In this problem a robot has to recover its pose under global uncertainty [1].

Remark 2.2. **Global Localization Problem**

Particle filters are numerical techniques for approximating posterior probabilities in partially observable controllable Markov chains with discrete time [1]. Concretely, particle filters comprise a broad family of sequential Monte Carlo algorithms for probability approximation in such environments.

Remark 2.3. **Partially Observable Markov Chains**

Remark 2.4. **Sequential Monte Carlo**

Particle filters are also known as sequential importance resampling or sequential Monte Carlo. The basis of these methods is an algorithm called sequential importance sampling, see section 2.4.

2.1 Mathematical Formulation

Let us assume that the state of a Markov chain at time t is $\mathbf{x}_t$ and that it depends on the state at time $t-1$, $\mathbf{x}_{t-1}$, and the control input $\mathbf{u}_t$ according to the so called **motion model**, see [1],

$$p(\mathbf{x}_t|\mathbf{x}_{t-1};\mathbf{u}_t) \tag{1}$$

It is assumed that the only information we have about the state of the environment at time t is the measurement $\mathbf{z}_t$ whci is a probabilistic projection of the true state $\mathbf{x}_t$. It is generated according to the so called **measurement model**

$$p(\mathbf{z}_t|\mathbf{x}_t) \tag{2}$$

Also let us assume that the initial state $\mathbf{x}_0$ is described by the distribution $p(\mathbf{x}_0)$ i.e.

$$\mathbf{x}_0 \sim p(\mathbf{x}_0) \tag{3}$$

The problem we want to solve in a partially observable Markov chain is the computation of the posterior distribution of the state $\mathbf{x}_t$ at time t. One methodology to solve this problem is the so called Bayes filters, see [1] and references therein.

In such an approach the posterior is calculated according to

$$p(\mathbf{x}_t|\mathbf{z}_t;\mathbf{u}_t) = p(\mathbf{z}_t|\mathbf{x}_t)\int p(\mathbf{x}_t|\mathbf{x}_{t-1};\mathbf{u}_t)p(\mathbf{x}_{t-1}|\mathbf{z}_{t-1};\mathbf{u}_{t-1})d\mathbf{x}_{t-1} \tag{4}$$

where the initial condition is

$$p(\mathbf{x}_0|\mathbf{z}_0;\mathbf{u}_0) = p(\mathbf{x}_0) \tag{5}$$

If states, controls and measurements are all discrete, the Markov chain is equivalent to hidden Markov models (HMM) [1] and references therein.

Remark 2.5. **Hidden Markov Models**

More than often in robotics, we deal with continuous state spaces. In such cases solving analytically equation 4 is not possible. In fact, analysical solutions for equation 4 exist is very specialized cases [1].

the basic idea behind particle filters is to approximate the posterior, $p(\mathbf{x}_t|\mathbf{z}_t;\mathbf{u}_t)$, using a set of

The basic idea behind particle filtering is to use a non-parametric representation of the posterior

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \approx \sum_{i=1}^{N} w_k^{(i)} \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}) \tag{6}$$

where $\mathbf{x}_k^{(i)}$ are particles and $w_k^{(i)}$ are associated weights.

We can then perform filtering by propagating $\mathbf{x}_k^{(i)}$ over time and updating the weights.

2.2 Monte Carlo sampling

Given independent samples $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(N)} \sim p(\mathbf{x})$ we can approximate

$$E[\mathbf{g}(\mathbf{x})] \approx \frac{1}{N} \sum_{i=1}^{N} \mathbf{g}(\mathbf{x}^{(i)}) \tag{7}$$

$$p(\mathbf{x}) \approx \frac{1}{N} \sum_{i=1}^{N} \delta(\mathbf{x} - \mathbf{x}^{(i)}) \tag{8}$$

Monte Carlo approximations have the following characteristics

- Non-parametric approximation to $p(\mathbf{x})$
- Approximates all kinds of densities $p(\mathbf{x})$
- Does not suffer from the curse of dimensionality

Remark 2.6. **Curse of Dimensionality**

However, we should note that it is often difficult to generate samples from $p(\mathbf{x})$

2.3 Importance sampling

Importance sampling can be used when it is difficult to sample from $p(\mathbf{x})$. Importance sampling generates samples $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(N)}$ from a a proposal density $q(\mathbf{x})$. We can then set $p(\mathbf{x})$ as

$$p(\mathbf{x}) \approx \sum_{i=1}^{N} w^{(i)} \delta(\mathbf{x} - \mathbf{x}^{(i)}) \tag{9}$$

where the weights are given by

$$w^{(i)} = \frac{\tilde{w}^{(i)}}{\sum_{n=1}^{N} \tilde{w}^{(i)}}, \quad \tilde{w}^{(i)} = \frac{p(\mathbf{x}^{(i)})}{q(\mathbf{x}^{(i)})} \tag{10}$$

Importance sampling is a flexible and powerful tool. It performs well as long as

- It is easy to sample from $q(\mathbf{x})$
- The support of $q(\mathbf{x})$ contains the support of $p(\mathbf{x})$
- $q(\mathbf{x})$ is similar to $p(\mathbf{x})$

2.4 Sequential Importance Sampling (SIS)

Recall that our goal is to recursively and accurately approximate the filtering density $p(\mathbf{x}_k|\mathbf{y}_{1:k})$. In order to do so we assumed that both the motion and measurement models, i.e. $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ and $p(\mathbf{y}_k|\mathbf{x}_k)$ respectively, can be evaluated point-wise. For example, for the model

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}) + \mathbf{q}_{k-1}, \quad \mathbf{q}_{k-1} \sim N(\mathbf{0}, \mathbf{Q}_{k-1}) \tag{11}$$
$$\mathbf{y}_k = h(\mathbf{x}_{k-1}) + \mathbf{r}_{k-1}, \quad \mathbf{r}_{k-1} \sim N(\mathbf{0}, \mathbf{R}_{k-1}) \tag{12}$$

then the density

$$p(\mathbf{x}_k|\mathbf{x}_{k-1}) = N(\mathbf{x}_k; f(\mathbf{x}_{k-1}), \mathbf{Q}_{k-1}) \tag{13}$$

is generally easy to be evaluated for any values of $\mathbf{x}_k$ and $\mathbf{x}_{k-1}$.

Particle filters are also known as sequential importance resampling or sequential Monte Carlo. The basis of these methods is an algorithm called sequential importance sampling. The standard SIS algorithm is outlined below

Remark 2.7. **Standard SIS Algorithm**

1. For $i = 1, \ldots, N$ at each time k do:
   - Draw $\mathbf{x}_k^{(i)} \sim q(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)}, \mathbf{y}_k)$
   - Compute weights
   $$w_k^{(i)} \propto w_{k-1}^{(i)} \frac{p(\mathbf{y}_k|\mathbf{x}_k^{(i)})q(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)})}{q(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)}, \mathbf{y}_k)} \tag{14}$$
   - Normalize the weights
2. Approximate
$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \approx \sum_{i=1}^{N} w_k^{(i)} \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}) \tag{15}$$

Now that we have a method to approximate $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ we ask ourselves what is the MMSE estimate of $\mathbf{x}_k$. We can calculate the MMSE estimate from this approximation via

$$\hat{\mathbf{x}}_k = \sum_{i}^{N} w_k^{(i)} \mathbf{x}_k^{(i)} \tag{16}$$

We can view the particle filter representation as approximating the posterior PDF with a posterior PMF where the weight, $w_k^{(i)}$, gives us the discrete probability that the state is $\mathbf{x}_k^{(i)}$, Using the definition of expected value on this PMF gives us the solution above.

Example 2.1. Nonlinear Filter Benchmark

We will demonstrate particle filtering using the following benchmark for nonlinear filters

$$x_k = \frac{x_{k-1}}{2} + \frac{25x_{k-1}}{1 + x_{k-1}^2} + 8\cos(1.2k) + q_{k-1} \tag{17}$$

$$y_k = \frac{x_k^2}{20} + r_k \tag{18}$$

where $q_{k-1} \sim N(0, 10)$ and $r_k \sim N(0, 1)$.

3 Rao-Blackwellized Particle Filter

4 Summary

This section briefly touched upon the particle-filter algorithm. Although, this is a powerful algorithm, the application of particle-filters to robotics is not without any problems. Specifically, a range of problems occurs due to the fact that no matter how detailed the probabilistic model, it will still be wrong and make false independence assumptions [1]. Furthermore, in robotics, all models lack important state variables that systematically affect sensor and actuator noise. Moreover, probabilistic inference is further complicated by the fact that robot systems must make decisions in real-time. The latter prohibits the use of vanilla particle-filters, as they exhibit exponential time performance, in many perceptual problems.

4.1 Questions

1. Which of the following statements, regarding the usefulness of particle filters, are true?
   (a) Particle filters are useful as they can handle almost any models
   (b) Particle filters are useful as within reasonable computational complexity, the particle filter always gives us the best performance.
   (c) Particle filters are useful as they give us a compact description of the posterior density.
   (d) Particle filters are useful as they give a non-parametric description of the posterior.
2. Assuming that we describe our posterior using the following approximation
$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \approx \sum_{i=1}^{N} w_k^{(i)} \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}) \tag{19}$$
   what is the MMSE estimate of $\mathbf{x}_k$?
   (a) $\hat{\mathbf{x}}_k = \sum_i^N w_k^{(i)} \mathbf{x}_k^{(i)}$
   (b) $\hat{\mathbf{x}}_k = \mathbf{x}_k^{(j)}$ where $j = argmax_i w_k^{(i)}$
   (c) $\hat{\mathbf{x}}_k = \frac{1}{N} \sum_i^N \mathbf{x}_k^{(i)}$
   (d) It is not possible to calculate a MMSE estimate from this approximation.

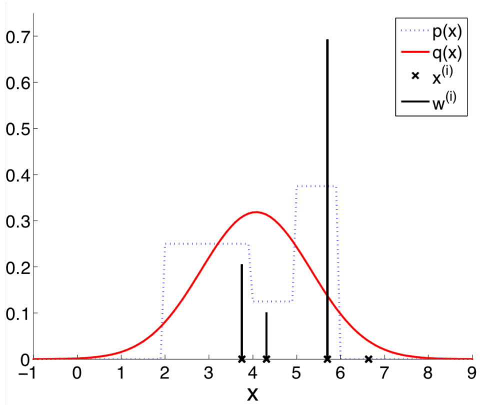
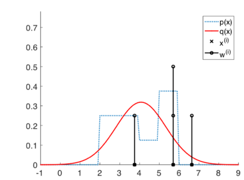
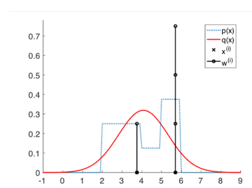


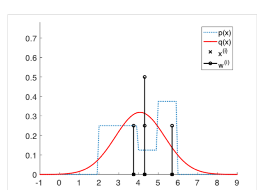
Fig. 1: Question figure.

3. Consider Figure 12.

   Perform resampling on the density and illustrate the result. Assume that the numbers 0.65, 0.03, 0.84 and 0.93 are drawn uniformly from $[0, 1]$. Which one of the following figures illustrates the resampled particles?

   (a) 

   (b) 

   (c) 

4. Bearing only tracking with a constant velocity motion in 2D, what is $\mathbf{x}_k^l$, $\mathbf{u}_k^l$ and $\mathbf{y}_k$ in this case

(a) $\mathbf{x}_k^l$: position, $\mathbf{u}_k^l$: velocity, $\mathbf{y}_k$: bearing to target

(b) $\mathbf{x}_k^l$: velocity, $\mathbf{u}_k^l$: position, $\mathbf{y}_k$: bearing to target

(c) $\mathbf{x}_k^l$: velocity, $\mathbf{u}_k^l$: bearing to target, $\mathbf{y}_k$: position

5. Which of the following is true about the SIS (Sequential Importance Sampling) particle filter?

   (a) It outputs strictly Gaussian posterior distribution approximations.

   (b) It can approximate multi-modal state distributions.

   (c) It eventually degenerates to just a few particles with significant weights.

   (d) It is a special case of a sigma-point filter.

6. With the particle filters we approximate the posterior as

$$p(\mathbf{x}_k|\mathbf{y}_{1:k} \approx \sum_{i=1}^{N} w_k^{(i)} \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}) \tag{20}$$

   Which of the following statements regarding this approximation are true?

   (a) $\mathbf{x}_k$ can take any value in $[\min_i \mathbf{x}_k^{(i)}, \max_i \mathbf{x}_k^{(i)}]$

   (b) We can view this as a discrete distribution where $P(\mathbf{x}_k = \mathbf{x}_k^{(i)}|\mathbf{y}_{1:k}) = w_k^{(i)}$.

   (c) It follows that $E[\mathbf{x}_k|\mathbf{y}_{1:k}] \approx \sum_{i=1}^{N} w_k^{(i)} \mathbf{x}_k^{(i)}$.

   (d) We can get arbitratry fine approximation by just in-creasing the number of particles, N .

7. Consider Figure 2

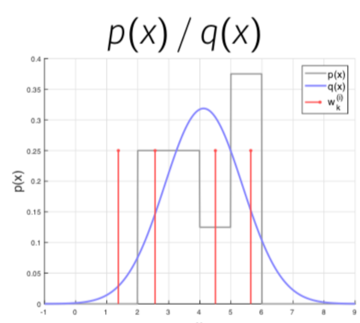


Fig. 2: Figure for exe

   Which one of the Figures 3, 4, 5 and 6 is the importance sampling approximation of $p(x)$ using $q(x)$?

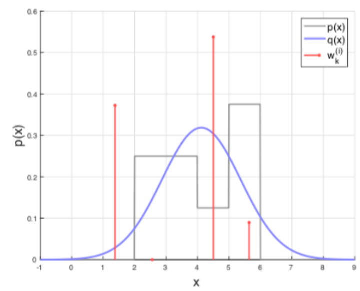
(a) 

Fig. 3: Option A

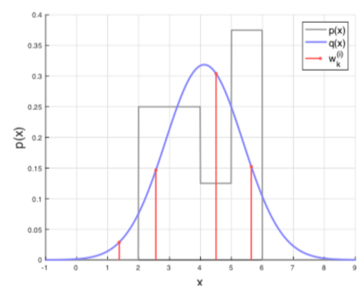
(b) 

Fig. 4: Option B

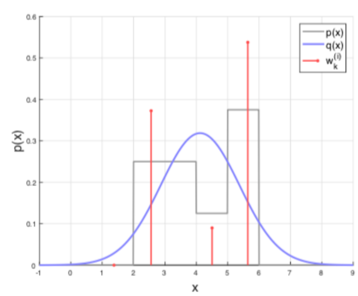
(c) 

Fig. 5: Option C

8. Consider Figure 7

   Which one of the Figures 8, 9, 10 and 11 is the importance sampling approximation of $p(x)$ using $q(x)$?

9. Which of the following is true about the SIS (Sequential Importance Sampling) particle filter?

   (a) It outputs strictly Gaussian posterior distribution approximations.

   (b) It can approximate multi-modal state distributions.

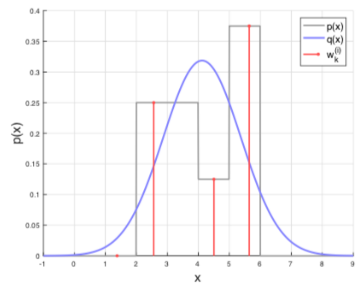
(d) 

Fig. 6: Option D

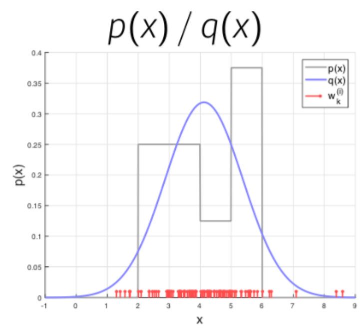


Fig. 7: Figure for exe

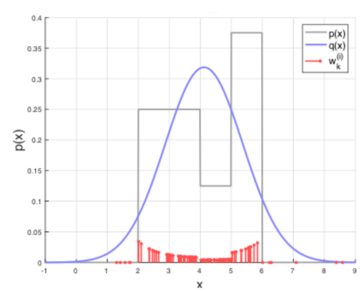
(a) 

Fig. 8: Option A

(c) It eventually degenerates to just a few particles with significant weights.

(d) It is a special case of a sigma-point filter.

10. Assume you want to compute the mean of a function $g(\mathbf{x})$ where $\mathbf{x}$ is distributed according to $p(\mathbf{x})$ which cannot be sampled from. To compute an approximate mean, you use importance sampling with a proposal density $q(\mathbf{x})$ and the normalized weights $w^{(i)}$. What of the following is true?

    (a) The proposal density $q(\mathbf{x})$ must be proportional to $p(\mathbf{x})$

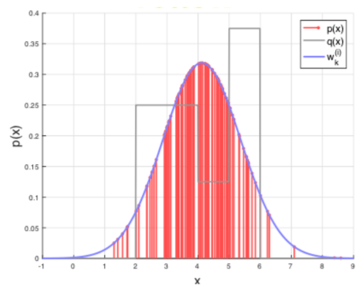
(b)


Fig. 9: Option B

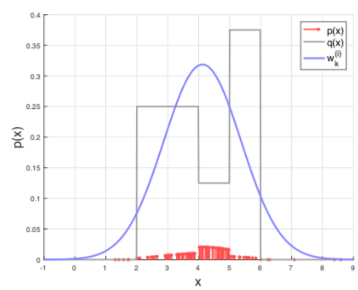
(c)


Fig. 10: Option C

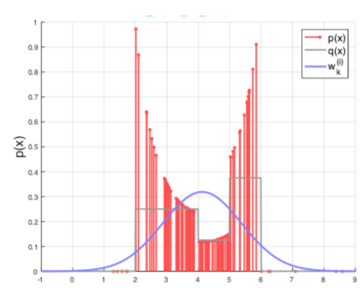
(d)


Fig. 11: Option D

(b) Samples $\mathbf{x}^{(i)}$ are drawn from $g(\mathbf{x})$

(c) $p(\mathbf{x})$ is approximated by the samples $\mathbf{x}^{(i)}$ as $p(\mathbf{x}) \approx \frac{1}{N}\sum_{i=1}^{N}\delta(\mathbf{x}-\mathbf{x}^{(i)})$

(d) The mean of any function $g(\mathbf{x})$ is approximated as $E[g(\mathbf{x})] \approx \sum_{i=1}^{N} g(\mathbf{x}^{(i)})w^{(i)}$

(e) You must evaluate the densities $p(\mathbf{x}^{(i)})$ and $q(\mathbf{x}^{(i)})$ for each sample in order to compute the mean of $g(\mathbf{x})$ by importance sampling.

11. Now you also want to compute the covariance of $g(\mathbf{x})$. What of the following is true?

(a) The proposal density $q(\mathbf{x})$ must have similar support as $g(\mathbf{x})$

(b) Covariance of any function $g(\mathbf{x})$ can be approximated using importance sampling as $\text{Cov}(g(\mathbf{x})) = \sum_{i=1}^{N} \text{Cov}(g(\mathbf{x}^{(i)}))w^{(i)}$

(c) When approximating the covariance of $g(\mathbf{x})$ using importance sampling, one can reuse the samples $\mathbf{x}^{(i)}$ and weights $w^{(i)}$ from when approximating $E[g(\mathbf{x})]$ with importance sampling.

12. What of the following is true about the SIR (Sequential Importance Resampling) particle filter?

    (a) It solves the degeneracy problem.

    (b) It should be performed at every time step.

    (c) The number of samples reduces after resampling.

    (d) Its performance depends on the quality of the importance distribtution. correct

    (e) The bootstrap filter is a typical variation of SIR.

13. Which of the following statements regarding resampling are true?

    (a) By resampling we make an approximation of our particle approximation of $p(\mathbf{x}_k|\mathbf{y}_{1:k})$

    (b) By resampling we get a more accurate approximation of $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ than what we had before we resampled.

    (c) By resampling we move the particles in a similar manner as we do in the measurement update of a Gaussian filter.

    (d) By resampling we focus our particles to high probability areas so they are not wasted in improbable states.

14. Resampling can sometimes result in lack of diversity. Consider the following toy example: There are two rooms, and the robot is unsure about which room it is in. The non-informative sensor shows equal probability of being in either room. We start with N particles equally distributed between the two rooms. How are particles distributed after a sufficiently long time if resampling is done at each time step?

    (a) We don't know. It depends on the distribution of last time step.

    (b) Each room will have the same number of particles.

    (c) The particles will converge to one of the two rooms.

4.2 Assignements

5 Answers

1. Which of the following statements, regarding the usefulness of particle filters, are true?
   (a) Particle filters are useful as they can handle almost any models
   (b) Particle filters are useful as within reasonable computational complexity, the particle filter always gives us the best performance.
   (c) Particle filters are useful as they give us a compact description of the posterior density.
   (d) Particle filters are useful as they give a non-parametric description of the posterior.

   Answer:

   The main advantage with particle filters is that they offer a non-parametric description of our posterior. As a consequence of this, it can be used to handle almost any model as long as we can evaluate them point-wise. However, it is hard to argue that we get a compact description of our posterior as we typically use a couple of thousand parameters or even more. Further, it is not a solution that should be used to solve all problems, for example, if we have linear and Gaussian models we should use the ordinary Kalman filter we know that this gives us the optimal solution. So correct options are A and D.

2. Assuming that we describe our posterior using the following approximation

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \approx \sum_{i=1}^{N} w_k^{(i)} \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}) \tag{21}$$

   what is the MMSE estimate of $\mathbf{x}_k$?
   (a) $\hat{\mathbf{x}}_k = \sum_i^N w_k^{(i)} \mathbf{x}_k^{(i)}$
   (b) $\hat{\mathbf{x}}_k = \mathbf{x}_k^{(j)}$ where $j = argmax_i w_k^{(i)}$
   (c) $\hat{\mathbf{x}}_k = \frac{1}{N} \sum_i^N \mathbf{x}_k^{(i)}$
   (d) It is not possible to calculate a MMSE estimate from this approximation.

   Answer:

   We can view the particle filter representation as approximating the posterior PDF with a posterior PMF where the weight, $w_k^{(i)}$, gives us the discrete probability that the state is $\mathbf{x}_k^{(i)}$, Using the definition of expected value on this PMF gives us the solution above. So the correct answer is option A.

3. Consider Figure 12.

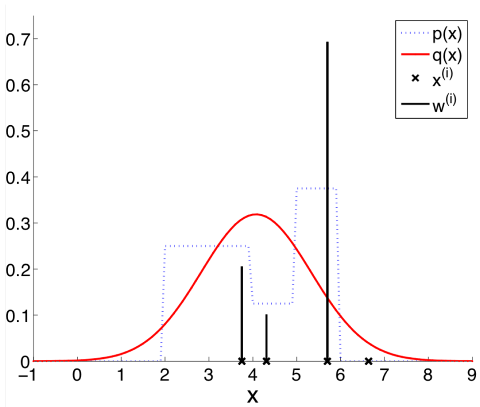
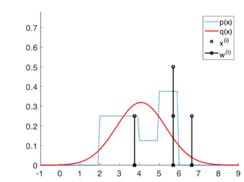
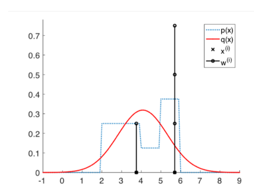
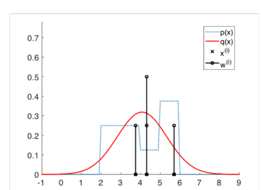


Fig. 12: Question figure.

Perform resampling on the density and illustrate the result. Assume that the numbers 0.65, 0.03, 0.84 and 0.93 are drawn uniformly from $[0, 1]$. Which one of the following figures illustrates the resampled particles?

(a) 

(b) 

(c) 

Answer:

If particles are ordered in ascending order, $x^{(1)} < \ldots < x^{(4)}$ resampling gives the following $x^{(1)} = 3.8$ and $x^{(2)} = x^{(3)} = x^{(4)} = 5.7$. Thus, option B is the correct answer.

4. Bearing only tracking with a constant velocity motion in 2D, what is $\mathbf{x}_k^l$, $\mathbf{u}_k^l$ and $\mathbf{y}_k$ in this case

   (a) $\mathbf{x}_k^l$: position, $\mathbf{u}_k^l$: velocity, $\mathbf{y}_k$: bearing to target

   (b) $\mathbf{x}_k^l$: velocity, $\mathbf{u}_k^l$: position, $\mathbf{y}_k$: bearing to target

   (c) $\mathbf{x}_k^l$: velocity, $\mathbf{u}_k^l$: bearing to target, $\mathbf{y}_k$: position

   Answer: Correct answer is option B. Our bearing measurements depend nonlinearly on our position state while the velocity inters linearly if we condition on the position state.

5. With the particle filters we approximate the posterior as

$$p(\mathbf{x}_k|\mathbf{y}_{1:k} \approx \sum_{i=1}^{N} w_k^{(i)} \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}) \tag{22}$$

   Which of the following statements regarding this approximation are true?

   (a) $\mathbf{x}_k$ can take any value in $[\min_i \mathbf{x}_k^{(i)}, \max_i \mathbf{x}_k^{(i)}]$

   (b) We can view this as a discrete distribution where $P(\mathbf{x}_k = \mathbf{x}_k^{(i)}|\mathbf{y}_{1:k}) = w_k^{(i)}$.

   (c) It follows that $E[\mathbf{x}_k|\mathbf{y}_{1:k}] \approx \sum_{i=1}^{N} w_k^{(i)} \mathbf{x}_k^{(i)}$.

   (d) We can get arbitratry fine approximation by just in-creasing the number of particles, N .

   Answer:

   Options B, C and D should be selected.

6. Consider Figure 2 Which one of the Figures 3, 4, 5 and 6 is the importance sampling approximation of $p(x)$ using $q(x)$?

   Answer

   Figure C.

7. Consider Figure 7

   Which one of the Figures 8, 9, 10 and 11 is the importance sampling approximation of $p(x)$ using $q(x)$?

   Answer

   Figure A.

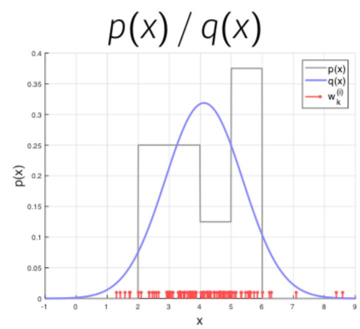


Fig. 13: Figure for exe

8. Which of the following is true about the SIS (Sequential Importance Sampling) particle filter?
   (a) It outputs strictly Gaussian posterior distribution approximations.
   (b) It can approximate multi-modal state distributions.
   (c) It eventually degenerates to just a few particles with significant weights.
   (d) It is a special case of a sigma-point filter.

   Answer:

   Options B and C should be selected.

9. Assume you want to compute the mean of a function $g(\mathbf{x})$ where $\mathbf{x}$ is distributed according to $p(\mathbf{x})$ which cannot be sampled from. To compute an approximate mean, you use importance sampling with a proposal density $q(\mathbf{x})$ and the normalized weights $w^{(i)}$. What of the following is true?
   (a) The proposal density $q(\mathbf{x})$ must be proportional to $p(\mathbf{x})$
   (b) Samples $\mathbf{x}^{(i)}$ are drawn from $g(\mathbf{x})$
   (c) $p(\mathbf{x})$ is approximated by the samples $\mathbf{x}^{(i)}$ as $p(\mathbf{x}) \approx \frac{1}{N}\sum_{i=1}^{N}\delta(\mathbf{x}-\mathbf{x}^{(i)})$
   (d) The mean of any function $g(\mathbf{x})$ is approximated as $E[g(\mathbf{x})] \approx \sum_{i=1}^{N} g(\mathbf{x}^{(i)})w^{(i)}$
   (e) You must evaluate the densities $p(\mathbf{x}^{(i)})$ and $q(\mathbf{x}^{(i)})$ for each sample in order to compute the mean of $g(\mathbf{x})$ by importance sampling.

   Answer:

   Options D and E should be selected.

10. Now you also want to compute the covariance of $g(\mathbf{x})$. What of the following is true?
    (a) The proposal density $q(\mathbf{x})$ must have similar support as $g(\mathbf{x})$

(b) Covariance of any function $g(\mathbf{x})$ can be approximated using importance sampling as $\mathrm{Cov}(g(\mathbf{x})) = \sum_{i=1}^{N} \mathrm{Cov}(g(\mathbf{x}^{(i)}))w^{(i)}$

(c) When approximating the covariance of $g(\mathbf{x})$ using importance sampling, one can reuse the samples $\mathbf{x}^{(i)}$ and weights $w^{(i)}$ from when approximating $E[g(\mathbf{x})]$ with importance sampling.

Answer:

Option C should be selected.

11. What of the following is true about the SIR (Sequential Importance Resampling) particle filter?

(a) It solves the degeneracy problem.

(b) It should be performed at every time step.

(c) The number of samples reduces after resampling.

(d) Its performance depends on the quality of the importance distribtution. correct

(e) The bootstrap filter is a typical variation of SIR.

Answer:

Options A, D and E should be selected.

12. Which of the following statements regarding resampling are true?

(a) By resampling we make an approximation of our particle approximation of $p(\mathbf{x}_k|\mathbf{y}_{1:k})$

(b) By resampling we get a more accurate approximation of $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ than what we had before we resampled.

(c) By resampling we move the particles in a similar manner as we do in the measurement update of a Gaussian filter.

(d) By resampling we focus our particles to high probability areas so they are not wasted in improbable states.

Answer:

Options A and D should be selected.

13. Resampling can sometimes result in lack of diversity. Consider the following toy example: There are two rooms, and the robot is unsure about which room it is in. The non-informative sensor shows equal probability of being in either room. We start with N particles equally distributed between the two rooms. How are particles distributed after a sufficiently long time if resampling is done at each time step?

(a) We don't know. It depends on the distribution of last time step.

(b) Each room will have the same number of particles.

(c) The particles will converge to one of the two rooms.

Answer:

Option C should be selected.

References

[1] Thurn S., *Particle Filters in Robotics* In Proceedings of Uncertainty in AI (UAI), 2002.

[2] Thurn S., Burgard W., Fox Do, *Probabilistic Robotics* MIT Press, 2006.