This is the same model used for the crowd counting paper where $P_c \rightarrow$ probability of maintaining the same bearing angle. We can prove that this model w/o the noise exhibits a doubly stochastic Markov model where asymptotically both the position and heading converge to a uniform distribution.

However in the presence of noise we may not make the same statements as above, and the system is nonlinear in general.

Particle Filtering:

Instead of working towards an exact solution we need to ~~test~~ consider a stochastic approach. That is, we need to estimate the likelihood of

$$P(\underline{X}_t \mid \hat{Y}_{1:t}) \longrightarrow$$ probability of a state occupancy given all past measurement history.

Q) Why "particle" filtering?

A) In a situation where you know the "likelihood" or belief levels of a random variable $\underline{X}$, sometimes normalizing this "likelihood" to obtain a PDF legitimately might not be possible

Furthermore, to compute any "expectation" value, you would need to ensure your normalization is done correctly to get reliable numbers!

So if we have $\pi(\underline{x}) = \dfrac{\pi'(\underline{x})}{Z}$ where

$$Z = \int_S \pi'(\underline{x})\, d\underline{x}$$ ← hard to evaluate.

We can propose a "particle-based" estimate of $\pi(\underline{x})$ using $\hat{\pi}(\underline{x}) = \frac{1}{N}\sum_{i=1}^{N} \delta(\underline{x} - x^{(i)})$ where

$\underline{x}^{(i)} \in S$ are $N$ randomly sampled points.

Thus for any measurable $\Phi(\underline{x})$ we have:

$$\mathbb{E}[\Phi(\underline{x})] = \int \Phi(\underline{x})\,\hat{\pi}(\underline{x})\,d\underline{x} = \frac{1}{N}\sum_{i=1}^{N} \Phi(\underline{x}^{(i)})$$

which converges to the empirical expectation via the LLN provided $\pi(\underline{x})$ originally is well behaved et cetera.

__Deviant__: Given an empirical CDF how do you generate random samples? Also assume you have $\mathcal{U}[0,1]$ generator.

$P[x_0 \le X \le x_1] = P_0$

$x_0 < x_1 < x_2 < \cdots < x_{n-1} < x_n$

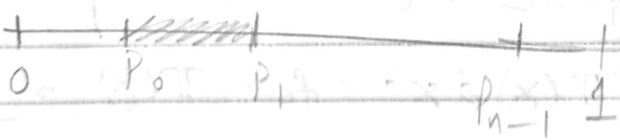
$P[x_0 \le X \le x_2] = P_1$

The trick is to observe that

$P[X_0 \le X \le X_n] = P_{n-1}$

$P[\alpha \le u \le \beta] = \beta - \alpha$ for unif $u \sim \mathcal{U}[0,1]$.

Thus, if we can partition $[0,1]$ into $P_0, P_1, \ldots, P_{n-1}$ and if we sample from $\mathcal{U}[0,1]$:




If we end up with $\theta \in [0,1]$ such that $P_i < \theta < P_{i+1}$ then $\theta$ maps to $X \in [X_i, X_{i+1})$. Thus by sampling $\mathcal{U}[0,1]$ we can effectively sample $X$ by proxy using the above mapping.

In real life we have only CDFs and no PDFs

## Rejection Sampling:

consider yet again an unnormalized distribution (or) likelihood function $\tilde{\pi}(x)$ which we are trying to normalize without explicit integration.

However, we need the normalization to perform any sort of sensible sampling or expectation sums. The particle based Monte Carlo estimate earlier is one way: but it might need a large # of sample points for LLN to kick in.

Instead, if we know that $q(\underline{x}) < \tilde{\pi}(\underline{x}) \leq B\,q(\underline{x})$ for $B \in \mathbb{R}$, $B > 0$ and measurable, normalized $q(\underline{x})$ such that we can draw samples from $q(\underline{x})$ easily, we can use an extension of the earlier derivant:

a) Sample from $q(\underline{x}) \sim \tilde{\underline{x}}$

b) Sample $u \sim \mathcal{U}[0,1]$

c) If $u \leq \dfrac{\tilde{\pi}(\tilde{\underline{x}})}{B\,q(\tilde{\underline{x}})}$ then "accept" $\tilde{\underline{x}}$ else "reject" $\tilde{\underline{x}}$.

If $\tilde{\underline{x}}$ is accepted, say $\underline{x}^+$, we can prove that

$$P[\underline{x}^+ \in A] = \int_A \pi(x)\,dx \quad \text{for } \pi(x) = \frac{\tilde{\pi}(x)}{Z}.$$

That is, from a crude likelihood we can construct the empirical CDF w/o explicitly normalizing.

Eg: Any bounded $|g(x)| < M$, we can use this technique to sample $e^{-x^2/2}g(x)$ by picking $q \sim N(0,1)$ and $B = M\sqrt{2\pi}$ etc.

* A key factor in deciding the efficiency of rejection sampling is the value of $B \in \mathbb{R}_+$. If B is too small then for any $x \sim \mathcal{U}[0,1]$ you would be very likely to accept a sample. But it so happens that for high dimen cases, B scales with the dimensionality in the exponent i.e, even reasonable values of B blow up at high "d" leading to very little or no acceptance

Importance Sampling:

Once again, if we have a measurable normalized ~~easy~~ easy to sample $q(x)$ and $\tilde{\pi}(x)$ which is not normalized $\Rightarrow \pi(x) = \frac{\tilde{\pi}(x)}{Z}$, we now only require

$$q(x) = 0 \quad \forall \, x \text{ where } \pi(x) = 0 .$$

Now, we have $\mathbb{E}_{\pi}[\Phi(x)] = \int \pi(x)\Phi(x)\,dx$

$$\Rightarrow \mathbb{E}_{\pi}(\Phi(x)) = \mathbb{E}_q\left[\Phi(x)\left\{\frac{\pi(x)}{q(x)}\right\}\right]$$

$$= \mathbb{E}_q\left[\Phi(x)\,W(x)\right]$$

$W(x)$ is called the "true" importance weight map, which is well defined owing to the criterion for choosing $q(x)$. However, we only know

$\hat{W}(x) = \frac{\tilde{\pi}(x)}{q(x)}$ which is the "relative weight"

$$= \frac{W(x)}{Z} \Rightarrow \mathbb{E}_q\left[\Phi(x)W(x)\right] = \mathbb{E}_{\pi}(\phi(x))$$

$$= \frac{1}{Z}\,\mathbb{E}_q\left[\phi(x)\hat{W}(x)\right]$$

Now, since $q$ is easy to sample,

$$\hat{I}(\Phi) = \frac{1}{NZ} \sum_{i=1}^{N} \Phi(\underline{x}^{(i)}) \tilde{w}(\underline{x}^{(i)})$$

where $\underline{x}^{(i)} \sim q(\underline{x})$.

Also, $\mathbb{E}_q[\tilde{w}(x)] = \int q(x) \frac{\tilde{\pi}(x)}{q(x)} dx = Z$.

So, we can also estimate $Z$ by sampling $q$!

$$\hat{Z} = \frac{1}{N} \sum_1^{'} \tilde{w}(\underline{x}^{(i)}) \quad \text{and we are done.}$$

$$\Rightarrow \hat{I} = \frac{\sum_1^{'} \Phi(\underline{x}^{(i)}) \tilde{w}(\underline{x}^{(i)})}{\sum_1^{'} \tilde{w}(\underline{x}^{(i)})}$$

or equivalently,

$$\hat{I} = \sum_1^{'} w_j \Phi(\underline{x}^{(i)}) \quad \text{where}$$

$$w_j = \frac{\tilde{w}(\underline{x}^{(i)})}{\sum_1^{'} \tilde{w}(\underline{x}^{(i)})} \quad \text{captures the normalization}$$

Thus, we can write $\hat{\pi}(x) = \sum_{j=1}^{N} w_j \delta(\underline{x} - \underline{x}^{(j)})$ as a "particle" based approximate PDF.

Of course, we also need to figure out how many points "N" to sample for good representations, and define the effective sample size

$$N_{eff} = \frac{1}{\sum_1^{'} w_i^2}$$

and resample until we get a good fit or low variability in $w$'s

Sequential Monte Carlo :

This technique is applicable for systems that exhibit the following properties :

a) Both state evolution in time and measurement model are non linear

b) The states belong to a Markov chain ie, the next state can be fully determined by the current one (DMC property)

c) The measurement relies only on the current state (DMC).

If a) was modified to a linear evolution mechanism then the most optimal solution would have been the Kalman Filter (provided noise processes are Gaussian as well).

Model : * State evolution : $X_{t+1} | X_t \sim f_t(X_{t+1} | X_t)$
* Initial condition : $X_1 \sim \mu(X_1)$.
* Measurement : $Y_t | X_t \sim g_t(Y_t | X_t)$.

Goal : compute $p(X_t | Y_{1:t})$.

Step 1 : $p(X_t | Y_{1:t}) = p(X_t | Y_t, Y_{1:t-1})$

$$\Rightarrow P(X_t | Y_{1:t}) = \frac{p(X_t, Y_t | Y_{1:t-1})}{P(Y_t | Y_{1:t-1})}$$

$$= \frac{p(Y_t | X_t, Y_{1:t-1}) \, P(X_t | Y_{1:t-1})}{P(Y_t | Y_{1:t-1})}$$

Now, $p(Y_t | X_t, Y_{1:t-1}) = p(Y_t | X_t) = g_t(Y_t | X_t)$
(Axiom c)

$$\Rightarrow p(x_t \mid y_{1:t}) = \frac{\underline{g_t(y_t \mid x_t)\, p(x_t \mid y_{1:t-1})}}{p(y_t \mid y_{1:t-1})}$$

Observe that the two underscored terms are

a) $p(x_t \mid y_{1:t-1}) \longrightarrow$ "prediction density", if we see carefully, the only way to describe this entity is by rolling back all the way to the prior ie, a recursion on time!

$$p(x_t \mid y_{1:t-1}) = \int p(x_t, x_{t-1} \mid y_{1:t-1})\, dx_{t-1}$$

$$= \int p(x_t \mid x_{t-1}, y_{1:t-1})\, p(x_{t-1} \mid y_{1:t-1})\, dx_{t-1}$$

From Markov property $p(x_t \mid x_{t-1}, y_{1:t-1}) = p(x_t \mid x_{t-1})$

$$\Rightarrow p(x_t \mid y_{1:t-1}) = \int p(x_t \mid x_{t-1})\, p(x_{t-1} \mid y_{1:t-1})\, dx_{t-1}$$

$$= \int f_t(x_t \mid x_{t-1})\, \underline{p(x_{t-1} \mid y_{1:t-1})}\, dx_{t-1}$$

Recursive!

b) $p(y_t \mid y_{1:t-1}) \longrightarrow$ "the normalization" factor, for which we can use our importance sampling technique! Let us hope that $p(x_{t-1} \mid y_{1:t-1})$ has a "particle based" representation

$$\hat{p}(x_{t-1} \mid y_{1:t-1}) = \sum_{i=1}^{N} w_{t-1}^{i}\, \delta(\underline{x}_{t-1} - \underline{x}_{t-1}^{i})$$

(of course $N$, etc. are assumed to be chosen semi optimally)

$$\Rightarrow \hat{p}(x_t | y_{1:t-1}) = \int f_t(x_t | x_{t-1}) \left( \sum_i w_{t-1}^i \delta(x_{t-1} - x_{t-1}^i) \right) dx_{t-1}$$

$$= \sum_{i=1}^{N} w_{t-1}^i f_t(x_t | x_{t-1}^i).$$

(we simply substituted our particle approximation into the summation).

Plug this into eqn (a), we get:

$$p(x_t | y_{1:t}) = \frac{1}{p(y_t | y_{1:t-1})} \sum_{i=1}^{N} \left[ g_t(y_t | x_t) w_{t-1}^i \times f_t(x_t | x_{t-1}^i) \right]$$

Now, if our proxy density for importance sampling is $q(x_t | y_{1:t})$, then we have a $\pi$ analogue:

$$\hat{p}(x_t | y_{1:t}) = g_t(y_t | x_t) \sum_i w_{t-1}^i f_t(x_t | x_{t-1}^i)$$

$$\Rightarrow \tilde{w}(x_t | y_{1:t}) = \frac{\hat{p}}{q} = \frac{g_t(y_t | x_t) \sum_i w_{t-1}^i f_t(x_t | x_{t-1}^i)}{q(x_t | y_{1:t})}$$

If we set $q(x_t | y_{1:t}) = \hat{p}(x_t | y_{1:t-1})$

$$\Rightarrow \tilde{w}(x_t) = g_t(y_t | x_t)$$

This mode is known as the "bootstrap particle filter".

Notation: $p(x_t | y_{1:t})$ is known as the filtering PDF at time "$t$".

To obtain $\hat{p}(x_t | y_{1:t})$ we need to use the "$t-1$" filtering PDF and perform importance sampling with $q(x_t) = \hat{p}(x_t | y_{1:t-1})$

So we need to draw samples $\{x_t^i, i=1,2,\dots,N\}$ from

$$q(x_t) = \sum_1^{} w_{t-1}^i f_t(x_t \mid x_{t-1}^i)$$

Clearly $q$ is made up of a weighted sum of PDFs itself, so we first choose which weight component to resolve and then use its coefficient PDF to sample.

That is,

a) Consider $(w_{t-1}^1, w_{t-1}^2, \dots w_{t-1}^N)$ as a PMF and select an index $1 \le k \le N$ with prob. $w_{t-1}^k$. Once this is done, $\underline{x}_{t-1} = \underline{x}_{t-1}^k$ due to the dirac delta.

b) Repeat this $N$ times to get $x_{t-1}^{k_1}, x_{t-1}^{k_2}, \dots, x_{t-1}^{k_N}$ and for each of these sample

$$x_t^{k_i} \sim f(x_t \mid x_{t-1}^{k_i})$$

c) Regenerate $w_t^1, w_t^2, \dots, w_t^N$ as

$$w_t^i = \frac{g(y_t \mid x_t^{k_i})}{\sum_{j=1}^{N} g(y_t \mid x_t^{k_j})}$$

of course at $t=1$, $x_1^{k_i} \sim \mu(x_1)$ as the base case prior! Notice that we have not used "$y_t$" information) anywhere for weight selection at time "$t$", only for recallibration!

Thus, the bootstrap particle filter might be inefficient if $y_t$ has significant content to alter the dynamics of the system.

# Particle filtering for motion tracking

Once again, it is to be understood that the "particles" being filtered are actually sampled data points from a multidimensional space.

**Input:** Total "tracking time" $T$, Number of particles $I$, Measurements $\psi_{1:T}$.

**Process:** 1) Assume a prior $\xi_1(\underline{x}_1)$ at $t=1$, if nothing provided, assume $1/I$ (uniform). Sample "$I$" particles $\underline{x}_1^1, \underline{x}_1^2, \ldots, \underline{x}_1^I$. Recall that each state vector has content of location and bearing angle

$$\Rightarrow \underline{x}_t^i \in \mathbb{R}^3. \quad (\text{particle space})$$

2) The first set of importance weights come from the measurement model: (so do the rest of them)

$$w_1^i = \frac{p(\psi_1 \mid \underline{x}_1 = \underline{x}_1^i)}{\sum_{j=1}^{I} p(\psi_1 \mid \underline{x}_1 = \underline{x}_1^j)}$$

3) The estimate (MMSE) for the target state is

$$\hat{\underline{x}}_1 = \mathbb{E}\{\underline{x}_1 \mid \underline{\psi}_1\} = \sum_{i=1}^{I} w_1^i \underline{x}_1^i$$

This is also compatible with Gaussian noise model since conditional expectation gives the MMSE!

4) Now use $w_{t-1}^i$ as the prior for particle sampling for $t = 2, 3, \ldots, T$, to obtain

$$\tilde{\underline{x}}_{t-1}^1, \tilde{x}_{t-1}^2, \ldots, \tilde{x}_{t-1}^I$$

5) Here is where the paper gets a little ambiguous and/or there is a typo that went uncorrected! It says "sample $\underline{x}_t^i \sim g(\tilde{\underline{x}}_{t-1}^i)$ and $g$ is not defined anywhere?! Oh no it is the motion dynamics!

We use the motion model to sample

$$\underline{x}_t^1, \underline{x}_t^2, \ldots, \underline{x}_t^I \sim g(\underline{x}_t \mid \tilde{\underline{x}}_{t-1}^i)$$

for $i = 1, 2, \ldots I$ respectively

6) Compute

$$w_t^i = \frac{p(\psi_t \mid \underline{x}_t = \underline{x}_t^i)}{\sum_{j=1}^{I} p(\psi_t \mid \underline{x}_t = \underline{x}_t^j)}$$

7) Estimate $\hat{x}_t = \sum_{i=1}^{I} w_t^i x_t^i$.

Thus, the AoA paper uses the classic bootstrap flavor of the particle filter.

Implementation of tracking

3 static RX setup on 3 laptops with Intel 5300 NIC.

However, the TX uses iPerf tool to broadcast packets, as it handles the problem of time synchronization with multiple RX operation.

For each time window $|T_{win}|$, the data at each receiver is processed as if it was from a receiver array of length $v|T_{win}|$ (?)