Motivation

- So far, we discussed the
  - Kalman filter: Gaussian, linearization problems
- Particle filters are a way to **efficiently** represent non-Gaussian distributions
- Basic principle
  - Set of state hypotheses (“particles”)
  - Survival-of-the-fittest

Density Approximation

- Particle sets can be used to approximate densities
- The more particles fall into an interval, the higher the probability of that interval
- How to draw samples from a function/distribution?
Let us assume that $f(x) \leq 1$ for all $x$
- Sample $x$ from a uniform distribution
- Sample $c$ from $[0,1]$
- if $f(x) > c$ keep the sample
  otherwise reject the sample

\[ f(x) \leq 1 \quad \text{for all } x \]

We can even use a different distribution $g$ to generate samples from $f$
- By introducing an importance weight $w$, we can account for the “differences between $g$ and $f$”
- $w = f / g$
- $f$ is often called target
- $g$ is often called proposal
This is Easy!
We can draw samples from $p(x | z_l)$ by adding noise to the detection parameters.

Importance Sampling with Resampling

Target distribution $f : p(x | z_1, z_2, \ldots, z_n) = \frac{\prod_{l=1}^{k} p(z_i | x) \cdot p(x)}{p(z_i, z_2, \ldots, z_n)}$

Sampling distribution $g: p(x | z_l) = \frac{p(z_i | x)p(x)}{p(z_l)}$

Importance weights $w : \frac{f}{g} = \frac{p(x | z_1, z_2, \ldots, z_n)}{p(x | z_l)} = \frac{p(z_i) \prod_{l=1}^{k} p(z_i | x)}{p(z_i, z_2, \ldots, z_n)}$

Resampling

- **Given**: Set $S$ of weighted samples.

- **Wanted**: Random sample, where the probability of drawing $x_i$ is given by $w_i$.

- Typically done $n$ times with replacement to generate new sample set $S'$. 
Resampling Algorithm

1. Algorithm systematic_resampling($S,n$):
2. $S' = \emptyset$, $c_1 = w^1$
3. For $i = 2 \ldots n$
4. $c_i = c_{i-1} + w^i$
5. $u_i \sim U[0,n^{-1}], i = 1$
6. For $j = 1 \ldots n$
7. While $(u_j > c_i)$
8. $i = i + 1$
9. $S' = S' \cup \lessgtr x^i, n^{-1} >$
10. $u_j = u_j + n^{-1}$
11. Return $S'$

Also called stochastic universal sampling

Particle Filters

Sensor Information: Importance Sampling

$Bel(x) \leftarrow \alpha \frac{p(z | x) \cdot Bel'(x)}{Bel'(x)} = \alpha \frac{p(z | x)}{Bel'(x)}$
**Robot Motion**

\[ Bel'(x) \leftarrow \int p(x | u, x') Bel(x') \, dx \]

**Sensor Information: Importance Sampling**

\[
Bel(x) \leftarrow \alpha p(z | x) Bel'(x) \\
w \leftarrow \alpha p(z | x) Bel'(x) = \alpha p(z | x)
\]

**Particle Filter Algorithm**

1. Algorithm **particle_filter**( \( S_{t-1}, u_{t-1}, z_t \)):
2. \( S_t = \emptyset, \ \eta = 0 \)
3. For \( i = 1 \ldots n \) \( \rightarrow \) **Generate new samples**
4. Sample index \( j(i) \) from the discrete distribution given by \( w_{t,i} \)
5. Sample \( x'_i \) from \( p(x_i | x_{i-1}, u_{i-1}) \) using \( x_{i-1}^{(j(i)),} \) and \( u_{i-1} \)
6. \( w'_i = p(z_i | x'_i) \) \( \rightarrow \) **Compute importance weight**
7. \( \eta = \eta + w'_i \) \( \rightarrow \) **Update normalization factor**
8. \( S_t = S_t \cup \{x'_i, w'_i\} \) \( \rightarrow \) **Insert**
9. For \( i = 1 \ldots n \) \( \rightarrow \) **Normalize weights**
10. \( w'_i = w'_i / \eta \)
Particle Filter Algorithm

\[ \text{Bel}(x_t) = \eta \cdot p(z_t \mid x_t) \cdot \int p(x_t \mid x_{t-1}, u_{t-1}) \cdot \text{Bel}(x_{t-1}) \, dx_{t-1} \]

- Draw \( x_{t-1} \) from \( \text{Bel}(x_{t-1}) \)
- Draw \( x_t \) from \( p(x_t \mid x_{t-1}, u_{t-1}) \)

Importance factor for \( x_t \):

\[ w_t' = \frac{\text{target distribution}}{\text{proposal distribution}} = \frac{\eta \cdot p(z_t \mid x_t) \cdot p(x_t \mid x_{t-1}, u_{t-1}) \cdot \text{Bel}(x_{t-1})}{p(x_t \mid x_{t-1}, u_{t-1}) \cdot \text{Bel}(x_{t-1})} \propto p(z_t \mid x_t) \]

Motion Model Reminder

Proximity Sensor Model Reminder

Laser sensor

Sonar sensor
Using Ceiling Maps for Localization

Vision-based Localization

Under a Light

Measurement $z$: $P(z|x)$

$P(z|x)$:
Next to a Light

Elsewhere

Global Localization Using Vision

Recovery from Failure
Localization for AIBO robots

Adaptive Sampling

KLD-Sampling Sonar

Adapt number of particles on the fly based on statistical approximation measure

KLD-Sampling Laser
Particle Filter Projection

Density Extraction

Sampling Variance

CSE-571
Robotics

Bayes Filter Implementations

Discrete filters
**Discrete Bayes Filter Algorithm**

1. **Algorithm** *Discrete_Bayes_filter* (*Bel(x), d*):
   2. \( \eta = 0 \)
   3. If *d* is a perceptual data item *z* then
      4. For all *x* do
         5. \( Bel'(x) = P(z \mid x)Bel(x) \)
         6. \( \eta = \eta + Bel'(x) \)
         7. For all *x* do
         8. \( Bel'(x) = \eta^{-1}Bel(x) \)
   9. Else if *d* is an action data item *u* then
      10. For all *x* do
        11. \( Bel'(x) = \sum_{x'} P(x \mid u, x') Bel(x') \)
      12. Return \( Bel'(x) \)

**Piecewise Constant Representation**

\( Bel(x_i) = \langle x, y, \theta > \)

\((0, 0, 0)\)

**Grid-based Localization**
Sonars and Occupancy Grid Map

**Idea:** Represent density using a variant of Octrees

Tree-based Representations

- Efficient in space and time
- Multi-resolution

Localization Algorithms - Comparison

<table>
<thead>
<tr>
<th></th>
<th>Kalman filter</th>
<th>Multi-hypothesis tracking</th>
<th>Topological maps</th>
<th>Grid-based (fixed/variable)</th>
<th>Particle filter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensors</td>
<td>Gaussian</td>
<td>Gaussian</td>
<td>Features</td>
<td>Non-Gaussian</td>
<td>Non-Gaussian</td>
</tr>
<tr>
<td>Posterior</td>
<td>Gaussian</td>
<td>Multi-modal</td>
<td>Piecewise constant</td>
<td>Piecewise constant</td>
<td>Samples</td>
</tr>
<tr>
<td>Efficiency (memory)</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>-/o</td>
<td>+/+</td>
</tr>
<tr>
<td>Efficiency (time)</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>o/+</td>
<td>+/+</td>
</tr>
<tr>
<td>Implementation</td>
<td>+</td>
<td>0</td>
<td>+</td>
<td>+/-</td>
<td>++</td>
</tr>
<tr>
<td>Accuracy</td>
<td>++</td>
<td>++</td>
<td>-</td>
<td>+/++</td>
<td>++</td>
</tr>
<tr>
<td>Robustness</td>
<td>-</td>
<td>+</td>
<td>+</td>
<td>++</td>
<td>+/+</td>
</tr>
<tr>
<td>Global localization</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>