zer0sh0t decodes SLAM

zer0sh0t

1 Gaussian Distributions

gaussain distributions are represented by $\mathcal{N}(x; \mu, \sigma^2)$
where x is the variable, μ is the mean of x, and σ is the standard deviation of x

2 Histogram Filter

This is a discrete bayes filter

1. prediction: convolution operation

$$\overline{bel}(x) = \sum_{x_{t-1}} p(x_t|x_{t-1}, u_t) * bel(x_{t-1})$$

where,

$$x_t = state_t$$
$$x_{t-1} = state_{t-1}$$
$$u_t = control_t$$

2. measurement: multiplication operation

$$bel(x) = \alpha * p(z_t|x_t) * \overline{bel}(x)$$

where,

$$z_t = measurement_t$$
$$\alpha = scaling\ factor$$

3 Kalman Filter

This is a continuous bayes filter. This is used when the prediction function is linear

$$bel(x) \sim (\mu_{t-1}, \sigma^2_{t-1})$$
$$control \sim (u_t, \sigma^2_R)$$
$$measurement \sim (z, \sigma^2_Q)$$

1. prediction: convolution operation

$$\overline{bel}(x) = \int p(x_t|x_{t-1}, u_t) * bel(x_{t-1}) * dx_{t-1}$$

here,

$$p(x_t|x_{t-1}, u_t) \sim \mathcal{N}(x_t; a * x_{t-1} + u_t, \sigma^2_R)$$
$$bel(x_{t-1} \sim \mathcal{N}(x_{t-1}; \mu_{t-1}, \sigma^2_{t-1})$$

and, we get

$$\boxed{\bar{\mu_t} = a * \mu_{t-1} + u_t}$$

$$\boxed{\bar{\sigma_t}^2 = a^2 * \sigma_{t-1}^2 + \sigma_R^2}$$

where, a is a scaling factor
2. measurement: multiplication operation

$$bel(x) = \alpha * p(z_t|x_t) * \overline{bel}(x)$$

here,

$$p(z|x) \sim \mathcal{N}(z; c * x, \sigma_Q^2)$$

$$\overline{bel}(x) \sim \mathcal{N}(x; \overline{\mu}, \overline{\sigma}^2)$$

where, c is used to transform from state space of x to measurement space
now,

$$bel(x) = \alpha * p(z|x) * \overline{bel}(x)$$

$$bel(x) = \acute{\alpha} * e^{\frac{-(z-c*x)^2}{2*\sigma_Q^2}} * e^{\frac{-(x-\bar{\mu})^2}{2*\bar{\sigma}^2}}$$

$$bel(x) = \acute{\alpha} * e^{\frac{-(z-c*x)^2}{2*\sigma_Q^2} - \frac{(x-\bar{\mu})^2}{2*\bar{\sigma}^2}}$$

the above bel(x) is quadratic in x and we can represent this in the form of

$$bel(x) = \acute{\alpha} * e^{\frac{-(x-\mu)^2}{\sigma^2}} \tag{1}$$

suppose, there is a function f(x)

$$f(x) = \frac{1}{2} * A * (x - B)^2 + C$$

$$\frac{\partial f}{\partial x} = A * (x - B)$$

if we set

$$\frac{\partial f}{\partial x} = 0$$

we get

$$x = B$$

also,

$$\frac{\partial^2 f}{\partial x^2} = A$$

now consider the function g(x)

$$g(x) = \frac{-(z - c * x)^2}{2 * \sigma_Q^2} - \frac{(x - \bar{\mu})^2}{2 * \bar{\sigma}^2}$$

by taking first and second derivatives of this function, we get

$$A = \frac{c^2}{\sigma_Q^2} + \frac{1}{\bar{\sigma}^2}$$

$$B = \frac{\frac{c*z}{\sigma_Q^2} + \frac{\bar{\mu}}{\bar{\sigma}^2}}{\frac{c^2}{\sigma_Q^2} + \frac{1}{\bar{\sigma}^2}}$$

then, from equation 1

$$A = 1/\sigma^2$$

$$B = \mu$$

now,

$$\sigma^2 = \frac{1}{(\frac{c}{\sigma_Q^2} + \frac{1}{\bar{\sigma}^2})} \tag{2}$$

$$\mu = \sigma^2 * (\frac{c}{\sigma_Q^2} + \frac{\bar{\mu}}{\bar{\sigma}^2}) \tag{3}$$

rewriting μ in a different way, we get

$$\mu = \sigma^2 * (\frac{c}{\sigma_Q^2} * (z - c * \bar{\mu} + c * \bar{\mu}) + \frac{\bar{\mu}}{\bar{\sigma}^2})$$

$$\mu = \sigma^2 * (\frac{c}{\sigma_Q^2} * (z - c * \bar{\mu}) + \frac{c^2 * \bar{\mu}}{\sigma^2} + \frac{\bar{\mu}}{\bar{\sigma}^2})$$

then, from equation (2)

$$\mu = \sigma^2 * (\frac{c}{\sigma_Q^2} * (z - c * \bar{\mu}) + \frac{\bar{\mu}}{\sigma^2})$$

$$\mu = \frac{\sigma^2}{\sigma_Q^2} * c * (z - c * \bar{\mu}) + \bar{\mu}$$

$$\mu = \bar{\mu} + K * (z - c * \bar{\mu})$$

where, K is the kalman gain

$$K = \frac{\sigma^2 * c}{\sigma_Q^2}$$

substituting σ^2 from equation (2)

$$K = \frac{c}{\sigma_Q^2 * (\frac{c}{\sigma_Q^2} + \frac{1}{\bar{\sigma}^2})}$$

multiplying and dividing by $\bar{\sigma}^2$

$$K = \frac{c * \bar{\sigma}^2}{\sigma_Q^2 * (\frac{c}{\sigma_Q^2} + \frac{1}{\bar{\sigma}^2}) * \bar{\sigma}^2}$$

$$K = \frac{c * \bar{\sigma}^2}{c^2 * \bar{\sigma}^2 + \sigma_Q^2}$$

rewriting σ^2 from equation (2)

$$\sigma^2 = \frac{1}{(\frac{c}{\sigma_Q^2} + \frac{1}{\bar{\sigma}^2})}$$

$$\sigma^2 = \frac{\sigma_Q^2 * \bar{\sigma}^2}{c^2 * \bar{\sigma}^2 + \sigma_Q^2}$$

$$\sigma^2 = (1 - \frac{c^2 * \bar{\sigma}^2}{c^2 * \bar{\sigma}^2 + \sigma_Q^2}) * \bar{\sigma}^2$$

$$\sigma^2 = (1 - K * c) * \bar{\sigma}^2$$

finally, we get

$$\boxed{K = \frac{c * \bar{\sigma}^2}{c^2 * \bar{\sigma}^2 + \sigma_Q^2}}$$

$$\boxed{\mu = \bar{\mu} + K * (z - c * \bar{\mu})}$$

$$\boxed{\sigma^2 = (1 - K * c) * \bar{\sigma}^2}$$

4 Extended Kalman Filter

This filter is used when the prediction function is non-linear

assume this 3x1 state vector and 2x1 control vector for computational simplicity:

$$state = \begin{vmatrix} x \\ y \\ \theta \end{vmatrix}$$

$$control = \begin{vmatrix} l \\ r \end{vmatrix}$$

and,

$$\begin{aligned} &bel(x) \sim (\mu_{t-1}, \Sigma_{t-1}) \\ &control \sim (u_t, \Sigma_{con}) \\ &measurement \sim (z, Q) \end{aligned}$$

1. prediction

$$\boxed{\bar{\mu_t} = g(\mu_{t-1}, u_t)}$$

$$\boxed{\bar{\Sigma_t} = G_t * \Sigma_{t-1} * G_t^T + V_t * \Sigma_{con} * V_T^T}$$

where, function g outputs a 3x1 state vector

$$g = \begin{vmatrix} g1 \\ g2 \\ g3 \end{vmatrix}$$

and,

$$G_t = \frac{\partial g}{\partial state}$$

G_t states how much change in Σ_{t-1} propagates to $\bar{\Sigma_t}$

$$V_t = \frac{\partial g}{\partial control}$$

V_t states how much change in Σ_{con} propagates to $\bar{\Sigma_t}$

2. correction

$$z_t = h(x_t)$$

where, h is the measurement function

$$\boxed{K = \bar{\Sigma_t} * H^T * (H * \bar{\Sigma_t} * H^T + Q)^{-1}}$$

$$\boxed{\mu = \bar{\mu} + K * (z - h(\bar{\mu_t}))}$$

$$\boxed{\Sigma = (I - K * H) * \bar{\Sigma}}$$

where,

$$H = \frac{\partial h}{\partial control}$$

H states how much change in $\bar{\Sigma}$ propagates to Σ

5 Particle Filter

let, M be a non-zero number, then

1. prediction

$$\boxed{sample\ \bar{x}_t \sim p(x_t|x_{t-1}, u_t)\ for\ M\ times}$$

2. correction

$$\boxed{weight\ w_t = p(z_t|\bar{x}_t)}$$

then, importance sampling technique is used to sample M particles where each particle is weighted by it's corresponding w_t

6 EKF SLAM

In EKF SLAM, the map of the environment is predicted along with the current position of the robot, so the state vector is modified to fit the problem setup

$$state = \begin{vmatrix} x \\ y \\ \theta \\ m \end{vmatrix}$$

where, m is a set of relevant features in the map. also, both prediction and measurement functions (g and h respectively) take map as an input along with the robot location

7 Fast SLAM

In Fast SLAM, particle filter is used to model position of the robot and kalman filter is used to model map

$$p(x_{1:t}, m|z_{1:t}, u_{1:t}, c_{1:t}) = p(x_{1:t}|z_{1:t}, u_{1:t}, c_{1:t}) * \prod_{1}^{N} p(m_i|x_{1:t}, z_{1:t}, c_{1:t})$$

where, $c_{1:t}$ is a set of data associations/landmark correspondences of a particle and,

$$measurement \sim (z, Q)$$

1. prediction

$$\boxed{sample\ \bar{x}_t \sim p(x_t|x_{t-1}, u_t)\ for\ M\ times}$$

where, M is the number of particles

2. correction

for every particle and for every measurement, likelihood of correspondence is computed. a counter variable is maintained for each landmark and if this value goes below zero, the landmark is removed from the map

first, the counter variable is decreased for all landmarks in the visibility range by -1. then,

$$\begin{aligned}
&\bar{z} = h(state, m_k) \\
&H = \frac{\partial h}{\partial landmark}\bigg|_{(state, m_k)} \\
&Q_l = H * \Sigma_k * H^T + Q_t \\
&\delta_z = z - \bar{z} \\
&l = \frac{1}{2 * \pi * \sqrt{det(Q_l)}} * e^{\frac{-\delta_z^T * Q_l^{-1} * \delta_z}{2}}
\end{aligned}$$

where, Σ_k is the covariance of the landmark position

if the likelihood is below the minimum threshold, a new landmark is initialized and the minimum threshold value is used for computing weight of this particle

$$\begin{aligned}
&m_k = h^{-1}(state, z) \\
&H = \left.\frac{\partial h}{\partial landmark}\right|_{(state,m)} \\
&\Sigma_k = H^{-1} * Q_t * (H^{-1})^T \\
&counter = 1
\end{aligned}$$

else if there is a correspondence, the landmark position is updated using kalman filter and the corresponding likelihood is used for computing weight of this particle

$$\begin{aligned}
H &= \left.\frac{\partial h}{\partial landmark}\right|_{(state,m_k)} \\
K &= \Sigma_{old} * H^T * Q_l^{-1} \\
\mu_{new} &= \mu_{old} + K * (z - h(state, \mu_{old})) \\
\Sigma_{new} &= (I - K * H) * \Sigma_{old} \\
&counter + +
\end{aligned}$$

then, all the landmarks with a negative counter are removed and all the likelihoods are multiplied to get the weight of the particle and they are resampled again using importance sampling technique