CSC 445 - Intro to Intelligent Robotics, Spring 2018

Uncertainty

Probabilistic Robotics

- Probabilistic robotics is about the
  - representation
  - propagation
  - reduction

  of uncertainty.

Environmental Representation

- The environment is characterized by *state*.
- There are two fundamental types of interactions between a robot and its environment.
    - Environmental sensor measurements
    - Control actions

Notation

- The state is represented as x_t where t is time.
- The measurement data at time t is denoted as z_t.
- The control data at time t is denoted as u_t.
- The notation

$$y_{t_1:t_2} = y_{t_1}, y_{t_1+1}, \ldots, y_{t_2}$$

denotes the set of all y values between from t_1 to t_2.

Representing Uncertainty

- In probabilistic robotics, uncertainty is represented explicitly using probability theory.
- The evolution of a state may be represented as a probability distribution

$$p(x_t \mid x_{0:t-1}, z_{1:t-1}, u_{1:t})$$

- That is, the state x_t is conditioned on all past states, measurements and controls.

Representing Uncertainty

- The state transition probability

$$p(x_t \mid x_{0:t-1}, u_{1:t})$$

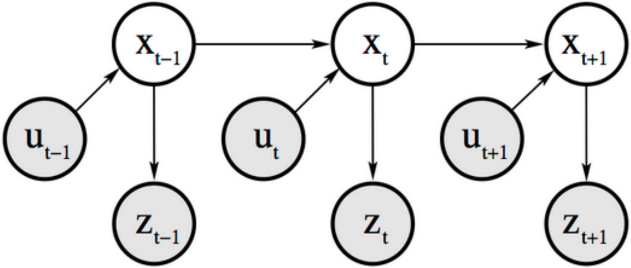
specifies how the state evolves over time as a function of the control actions.

- The measurement probability

$$p(z_t \mid x_{0:t}, z_{1:t-1}, u_{1:t})$$

specifies the probabilistic law according to which measurement z are generated from environment state x.

State Evolution Bayes Network



Complete State

- A state x_t is complete if it is the best predictor of the future.
- In other words, completeness means that knowledge of the past states, measurements, or controls carry no additional information that would help us predict the future.
- If x_t is complete, the evolution of a state may be represented as a state transition probability

$$p(x_t \mid x_{t-1}, u_t)$$

- Additionally, the measurement probability can be represented as

$$p(z_t \mid x_t)$$

Belief Distribution

- A *belief* reflects the robot's internal knowledge about the state of the environment.
- Probabilistic robotics represents beliefs with conditional probability distributions.
- A belief distribution is a posterior probability over state variables conditioned on the available data

$$bel(x_t) = p(x_t \mid z_{1:t}, u_{1:t})$$

Bayes Filter Algorithm

function $\textsc{Bayes Filter}(bel(x_{t-1}), u_t, z_t)$
    for all x_t **do**
        $\overline{bel}(x_t) = \int p(x_t \mid u_t, x_{t-1}) bel(x_{t-1}) dx_{t-1}$
        $bel(x_t) = \eta p(z_t \mid x_t) \overline{bel}(x_t)$
    return $bel(x_t)$

Different Realizations

- The Bayes filter is a framework for recursive state estimation.
- There are different realizations.
- Different properties
  - Linear vs. non-linear models for state transition and measurements.
  - Parametric vs. non-parametric
  - ...

Gaussian Filters

- A Gaussian filter represents beliefs by multivariate normal distributions

$$p(x) = \det(2\pi\Sigma)^{\frac{1}{2}} \exp\{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\}$$

- The Gaussian distribution is unimodal, so the posterior represents a single hypothesis.

The Gaussian Distribution

- A 1D Gaussian distribution is defined as

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{(x-\mu)^2}{2\sigma^2}}$$

- A n dimensional Gaussian distribution is defined as

$$p(x) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} e^{\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)}$$

where x is a vector and Σ is a covariance matrix.

Covariance Matrix

- When X is a vector, the variance is expressed as a covariance matrix Σ where

$$\sigma_{ij} = \mathrm{E}[(x_i - \mu_i)(x_j - \mu_j)]$$

- A covariance matrix has the form

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \rho_{12}\sigma_1\sigma_2 & \dots & \rho_{1n}\sigma_1\sigma_n \\ \rho_{12}\sigma_1\sigma_2 & \sigma_2^2 & \dots & \rho_{2n}\sigma_2\sigma_n \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{1n}\sigma_1\sigma_n & \rho_{2n}\sigma_2\sigma_n & \dots & \sigma_n^2 \end{bmatrix}$$

  where ρ_{ij} corresponds to the degree of correlation between the variables X_i and X_j.

Properties of the Gaussian Distribution

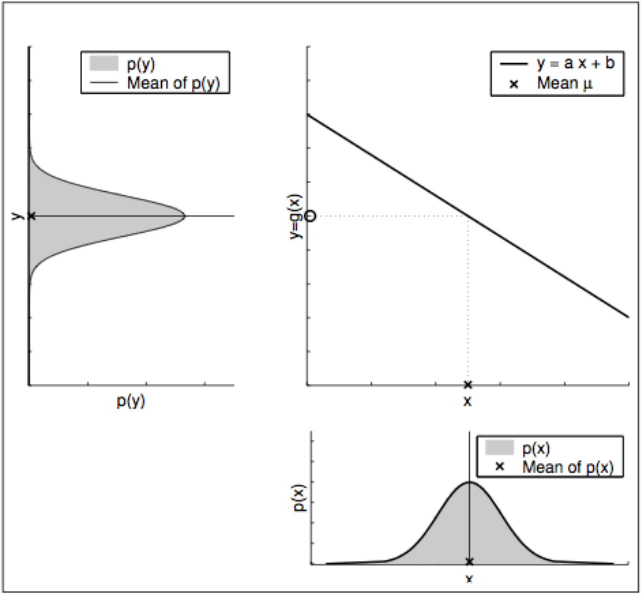
- Given two independent random variables, $X \sim \mathcal{N}(\mu_x, \sigma_x^2)$ and $Y \sim \mathcal{N}(\mu_y, \sigma_y^2)$, then

  $$aX + b \sim \mathcal{N}(a\mu_x + b, a^2\sigma_x^2)$$

  and

  $$Z = X + Y \sim \mathcal{N}(\mu_x + \mu_y, \sigma_x^2 + \sigma_y^2)$$

Properties of the Gaussian Distribution

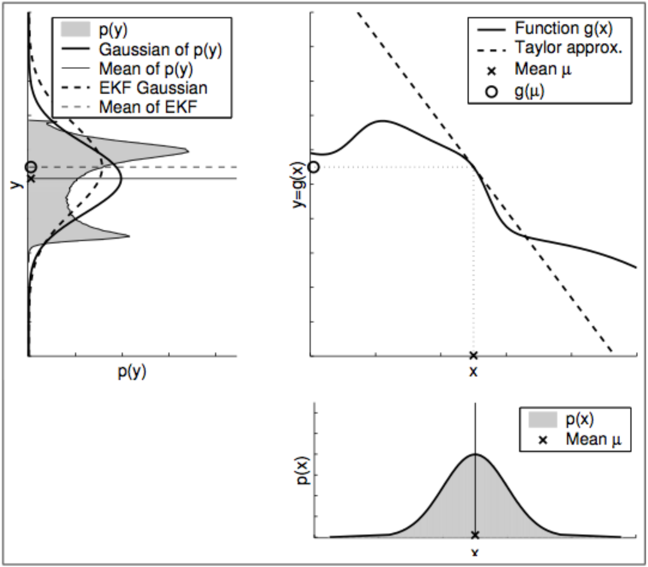


Linear Approximation of a Nonlinear Function

- Given $Y = f(X)$ with X and Y assumed to be Gaussian and $f(\cdot)$ a nonlinear function
- Approximate the function with a first order Taylor series expansion

$$Y \approx f(\mu_x) + \frac{\partial f}{\partial X}\bigg|_{x=\mu_x}(X - \mu_x)$$

Linear Approximation of a Nonlinear Function



Transforming Uncertainty

- Propagation of uncertainty is the effect of uncertainty of a random variable to the uncertainty of a function based on the random variable.
- Given a function

$$y = f(x)$$

  that maps a random variable x, to a random variable y.
- Let the standard deviation of x be given by σ_x.
- We can calculate the variance of σ_y^2 as

$$\sigma_y^2 = \left(\frac{\partial f}{\partial x}\right)^2 \sigma_x^2$$

Transforming Uncertainty

- If the function is a multivariable function that maps n inputs to m outputs, then the variances become covariance matrices.
- The covariance matrix of of y can be calculated as

$$\Sigma_y = J \Sigma_x J^T$$

where J is an $m \times n$ Jacobian matrix.

Jacobian Matrix

- Let $f(x)$ be a vector-valued function

$$f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \end{bmatrix}$$

- Let the gradient operator be the vector of (first order) partial derivatives

$$\nabla_x = \begin{bmatrix} \frac{\partial}{\partial_{x_1}} & \frac{\partial}{\partial_{x_2}} & \cdots & \frac{\partial}{\partial_{x_n}} \end{bmatrix}^T$$

- Then, the Jacobian matrix is defined as

$$F_x = \begin{bmatrix} f_1(x) \\ f_2(x) \end{bmatrix} \cdot \nabla_x = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} \cdots \frac{\partial f_1}{\partial x_n} \\ \\ \frac{\partial f_2}{\partial x_1} \cdots \frac{\partial f_2}{\partial x_n} \end{bmatrix}$$

Kalman Filter

- The Kalman filter is a realization of a Bayes filter.
- It is an estimator for the linear Gaussian case.
- It is the optimal solution for linear models and Gaussian distributions.

Linear State Transition Model

- The Kalman filter assumes that the true state as time k is evolved from the state at $(k-1)$ according to

  $$x_k = F_k x_{k-1} + B_k u_k + w_k$$

  where
  - F_k is the state transition model.
  - B_k is the control-input model.
  - w_k is process noise assumed to be drawn from a zero mean normal distribution with covariance Q_k.

Linear Observation Model

- At time k an observation (or measurement) z_k of the true state x_k is made according to

  $$z_k = H_k x_k + v_k$$

  where

  - H_k is the observation model which maps the true state space into the observed state space.
  - v_k is observation noise assumed to be drawn from a zero mean normal distribution with covariance R_k.

Kalman Filter State

- The state of a Kalman filter is represented by two variables
  - $\hat{x}_{k|k}$, the posterior state estimate at time k given the observations up to and including time k;
  - $P_{k|k}$, the posterior error covariance matrix.
- The notation $\hat{x}_{n|m}$ represents the estimate of x at time n given observations up to and including time $m \leq n$.

Kalman Filter Prediction Step

1. Predict the state estimate

$$\hat{x}_{k|k-1} = F_k \hat{x}_{k-1|k-1} + B_k u_k$$

2. Predict the estimate covariance

$$P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + Q_k$$

Kalman Filter Correction Step

1. Compute the innovation

$$\tilde{y}_k = z_k - H_k \hat{x}_{k|k-1}$$

2. Compute the innovation covariance

$$S_k = R_k + H_k P_{k|k-1} H_k^T$$

3. Compute the optimal Kalman gain

$$K_k = P_{k|k-1} H_k^T S_k^{-1}$$

4. Update the state estimate

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \tilde{y}_k$$

5. Update the estimate covariance

$$P_{k|k} = (I - K_k H_k) P_{k|k-1}$$

The Extended Kalman Filter

- The Extended Kalman Filter (EKF) is a sub-optimal extension to the original Kalman filter algorithm.
- The EKF allows for the estimation of non-linear state transition and observation models.
- This is accomplished by linearizing the mean and covariance estimates.

Nonlinear State Transition Model

- The EKF assumes that the true state as time k is evolved from the state at $(k-1)$ according to

$$x_k = f(x_{k-1}, u_k) + w_k$$

  where
  - $f(\cdot)$ is the nonlinear state transition model.
  - F_k is the Jacobian of f with respect to the state.
  - w_k is process noise assumed to be drawn from a zero mean normal distribution with covariance Q_k.

Nonlinear Observation Model

- At time k an observation (or measurement) z_k of the true state x_k is made according to

$$z_k = h(x_k) + v_k$$

  where
  - $h(\cdot)$ is the nonlinear observation model which maps the true state space into the observed state space.
  - H_k is the Jacobian of h with respect to the state.
  - v_k is observation noise assumed to be drawn from a zero mean normal distribution with covariance R_k.

Extended Kalman Filter Prediction Step

1. Predict the state estimate

$$\hat{x}_{k|k-1} = f(\hat{x}_{k-1|k-1}, u_k)$$

2. Predict the estimate covariance

$$P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + Q_k$$

Extended Kalman Filter Correction Step

1. Compute the innovation

$$\tilde{y}_k = z_k - h(\hat{x}_{k|k-1})$$

2. Compute the innovation covariance

$$S_k = H_k P_{k|k-1} H_k^T + R_k$$

3. Compute the (near optimal) Kalman gain

$$K_k = P_{k|k-1} H_k^T S_k^{-1}$$

4. Update the state estimate

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \tilde{y}_k$$

5. Update the estimate covariance

$$P_{k|k} = (I - K_k H_k) P_{k|k-1}$$