CSE 483: Mobile Robotics

Lecture by: Prof. K. Madhava Krishna
Scribe: J. Krishna Murthy

Lecture # 04
Date: 12th August, 2016 (Friday)

Linearization of the Probabilistic Motion Model

Until the last lecture, we've discussed multivariate gaussian distributions. In this lecture, we will begin modeling the robot's state probabilistically using multivariate Gaussians. These concepts serve as a build up to the *Extended Kalman Filter*, a technique for probabilistic state estimation.

1 Probabilistic Motion Model

As discussed in the previous lecture, a robot represents it's state $\mathbf{X_t}$ at time t in terms of the mean $\mu_\mathbf{t}$ and the covariance $\mathbf{\Sigma_t}$. Let us denote by $bel(\mathbf{X_t})$ the belief of the robot that it is indeed in state $\mathbf{X_t}$ at time t.

$$bel(\mathbf{X_t}) = \mathcal{N}(\mu_\mathbf{t}, \mathbf{\Sigma_t}) = \frac{1}{\eta} exp \left\{ -\frac{1}{2}(\mathbf{X_t} - \mu_\mathbf{t})^T \mathbf{\Sigma}^{-1} (\mathbf{X_t} - \mu_\mathbf{t}) \right\} \tag{1}$$

Let the robot's belief at time $t+1$ be denoted by $bel(\mathbf{X_{t+1}})$. Intuitively, the robot's belief indicates *where the robot **thinks** it is*. It is important to understand that *where the robot **thinks** it is* and *where the robot **actually** is* need not coincide. The primary reason for this is noise in the robot's motion.

Let $\mathbf{u_{t+1}}$ be the control applied to the robot at time t.

$$\mathbf{u_{t+1}} = \begin{bmatrix} T \\ \phi \end{bmatrix} \tag{2}$$

In equation 2, T represents the translation and ϕ represents the rotation applied to the robot by the control $\mathbf{u_{t+1}}$. It must be noted that the control is also a random variable characterized by a multivariate Gaussian distribution, to accomodate for the fact that there could be noise in the control. Note that Kalman Filters work only under assumptions of Gaussian noise.

A (probabilistic) *motion model* is a probability distribution over the set of robot's states, conditioned over the current state and the next control. In essence, a (probabilistic) motion model maps the current state and the next control to the next state. The motion model for a differential drive robot can now be defined in terms of the current state $\mathbf{X_t}$ and the control $\mathbf{u_{t+1}}$ as follows.

$$\begin{bmatrix} \hat{\mu}_{x,t+1} \\ \hat{\mu}_{y,t+1} \\ \hat{\mu}_{\theta,t+1} \end{bmatrix} = \begin{bmatrix} \mu_{x,t} \\ \mu_{y,t} \\ \mu_{\theta,t} \end{bmatrix} + \begin{bmatrix} Tcos(\mu_{\theta,t} + \phi) \\ Tsin(\mu_{\theta,t} + \phi) \\ \phi \end{bmatrix} \tag{3}$$

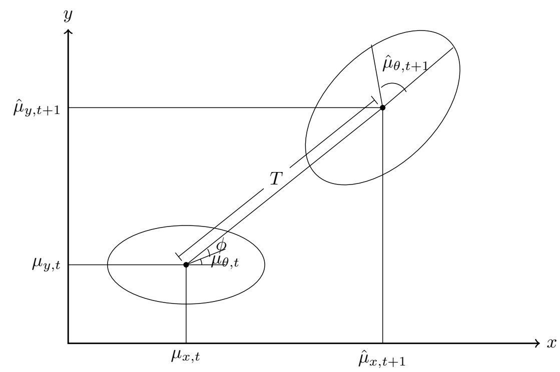

Figure 1: A diagramatic representation of the robot's (probabilistic) motion model. Notice how the uncertainty grows with time. The reasons for this will become clear soon.

In equation 3, $\hat{\mu}_{x,t+1}$, $\hat{\mu}_{y,t+1}$, and $\hat{\mu}_{\theta,t+1}$ together form $\hat{\mu}_{\mathbf{t+1}}$, the next state estimate. This is diagramatically represented in Fig. 1.

1.1 A Note on Gaussians

There are certain desirable properties of Gaussian distributions which make them attractive models for being used in many robotics applications. Here, we make note of one important property that motivates their use in the Extended Kalman Filter (EKF).

Assume that $\mathbf{X}$ is a Gaussian random variable with mean $\mathbf{a}$ and covariance $\mathbf{B}$, i.e., $\mathbf{X} = \mathcal{N}(\mathbf{a}, \mathbf{B})$. Now, consider a linear transformation A that transforms $\mathbf{X}$ to $\mathbf{Y}$.

$$\mathbf{Y}_{m\times 1} = A_{m\times n}\mathbf{X}_{n\times 1} \tag{4}$$

The subscripts indicate the dimensions of each vector or matrix. It turns out that $\mathbf{Y}$ is again a Gaussian distribution with mean $A\mathbf{a}$ and covariance $A\mathbf{B}A^T$, i.e., $\mathbf{Y} = \mathcal{N}(A\mathbf{a}, A\mathbf{B}A^T)$. In nutshell, any linear transformation applied on a Gaussian distribution results in a Gaussian distribution.

2 Linearization of the Motion Model

Assume that the state vector $\mathbf{X}$ is subject to a non-linear transform.

$$\mathbf{Y} = f(\mathbf{X}) \tag{5}$$

In the above equation, f is a non-linear function, and hence $\mathbf{Y}$ no longer follows a multivariate Gaussian distribution. So, to be able to use the Kalman filter for such functions f, we *linearize* the function about a point (say) x_0.

We use Taylor series expansion for linearizing the function. The following equation performs Taylor expansion.

$$\mathbf{Y}(x_0) = f(x_0) + \left(\frac{\partial f}{\partial \mathbf{X}}\right)^T \Big|_{x_0} (x - x_0) + (x - x_0)^T \left(\frac{\partial^2 f}{\partial \mathbf{X}^2}\right)\Big|_{x_0} (x - x_0) \tag{6}$$

In the above equation, the matrix of first-order partial derivatives is referred to as the *Jacobian*, whereas the matrix of second-order partial derivatives is referred to as the *Hessian*. To linearize $\mathbf{Y}$, we can ignore the second order (Hessian) term, and the above equation can be written in the vecotr form as follows.

$$\mathbf{Y} = f(\mathbf{X_0}) + \mathbf{J}(\mathbf{X} - \mathbf{X_0}) \tag{7}$$

Now, since $\mathbf{Y}$ is linear, and since $\mathbf{X}$ follows a multivariate Gaussian distribution, $\mathbf{Y}$ also follows a multivariate Gaussian distribution. More precisely, $\mathbf{Y} = \mathcal{N}(\mathbf{Ja}, \mathbf{JBJ^T})$.

Coming back to the motion model specified in equation 3, we notice that both the current state and the next control are drawn from independent Gaussian distributions. Hence, the motion model has two Jacobian matrices; one corresponds to the state vector, and the other corresponds to the control vector.

$$\hat{\mu}_{\mathbf{t+1}} = f(\mu_{\mathbf{t}}, \mathbf{u_t}) \tag{8}$$

$$\hat{\mathbf{\Sigma}}_{\mathbf{t+1}} = \mathbf{F\Sigma_t F^T} + \mathbf{G\Sigma_u G^T} \tag{9}$$

Here, $\mathbf{F}$ and $\mathbf{G}$ correspond to the Jacobians with respect to the state and the control vectors respectively.

$$\hat{\mathbf{\Sigma}}_{\mathbf{t+1}} = \begin{bmatrix}\mathbf{F} & \mathbf{G}\end{bmatrix} \begin{bmatrix}\mathbf{\Sigma_t} & 0 \\ 0 & \mathbf{\Sigma_t}\end{bmatrix} \begin{bmatrix}\mathbf{F}^T \\ \mathbf{G}^T\end{bmatrix} \tag{10}$$

$$\mathbf{F} = \frac{\partial f}{\partial \mu_{\mathbf{t}}} = \begin{bmatrix} \frac{\partial \hat{\mu}_{x,t+1}}{\partial \mu_{x,t}} & \frac{\partial \hat{\mu}_{x,t+1}}{\partial \mu_{y,t}} & \frac{\partial \hat{\mu}_{x,t+1}}{\partial \mu_{\theta,t}} \\ \frac{\partial \hat{\mu}_{y,t+1}}{\partial \mu_{x,t}} & \frac{\partial \hat{\mu}_{y,t+1}}{\partial \mu_{y,t}} & \frac{\partial \hat{\mu}_{y,t+1}}{\partial \mu_{\theta,t}} \\ \frac{\partial \hat{\mu}_{\theta,t+1}}{\partial \mu_{x,t}} & \frac{\partial \hat{\mu}_{\theta,t+1}}{\partial \mu_{y,t}} & \frac{\partial \hat{\mu}_{\theta,t+1}}{\partial \mu_{\theta,t}} \end{bmatrix} \tag{11}$$

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & -Tsin(\mu_{\theta,t} + \phi) \\ 0 & 1 & Tcos(\mu_{\theta,t} + \phi) \\ 0 & 0 & 1 \end{bmatrix} \tag{12}$$

$$\mathbf{G} = \frac{\partial f}{\partial \mathbf{u_t}} = \begin{bmatrix} \frac{\partial \hat{\mu}_{x,t+1}}{\partial T} & \frac{\partial \hat{\mu}_{x,t+1}}{\partial \phi} \\ \frac{\partial \hat{\mu}_{y,t+1}}{\partial T} & \frac{\partial \hat{\mu}_{y,t+1}}{\partial \phi} \\ \frac{\partial \hat{\mu}_{\theta,t+1}}{\partial T} & \frac{\partial \hat{\mu}_{\theta,t+1}}{\partial \phi} \end{bmatrix} \tag{13}$$

$$\mathbf{G} = \begin{bmatrix} cos(\mu_{\theta,t} + \phi) & -Tsin(\mu_{\theta,t} + \phi) \\ sin(\mu_{\theta,t} + \phi) & Tcos(\mu_{\theta,t} + \phi) \\ 0 & 1 \end{bmatrix} \tag{14}$$

Now that we have computed the Jacobians with respect to the state and the control, the robot's motion model can be linearized by using equation 9.

Equation 9 provides a revelation of how the uncertainty in robot state evolves over time. Since $\mathbf{\Sigma_t}$ and $\mathbf{\Sigma_u}$ are covariance matrices, they are symmetric and are positive semi-definite. Hence, $\hat{\mathbf{\Sigma}}_{\mathbf{t+1}}$ increases with time, since it is a summation of positive semi-definite quantities. So, the uncertainty in the robot's state increases with time, if left uncorrected. In the next lecture, we will look at methods of reducing this uncertainty by incorporating *measurements*.