ME/CS 132(b)

Notes on Feature-based EKF SLAM

(Version 2: May 5, 2017)

1 Introduction

These notes review the basic equations of EKF (extended Kalman Filter) SLAM (Simultaneous Localization and Mapping).

1.1 The Notion of a "Map"

Let $\vec{p}_k^R$ denote the robot's position at time t_k. We assume that the robot's exteroception sensor suite can identify and recognize "landmarks" which will make up the map (where the landmarks play the role of a "beacon"). We will assume that the robot can process the sensory data associated with the landmark so that a unique point can be associated to each landmark. Hence, each landmark can be associated to the coordinates of that point. Let $\vec{p}^{l_j}$ denote the coordinates of the j^{th} landmark. While we assume that the landmarks are stationary, we assume that any measurements of their location is noisy, and therefore the estimate of the landmark's location overy time will hopefully converge to its true relative location.

An estimator, in this case an Extended Kalman Filter (EKF) will be used to update the estimate of the robot's position and the landmark's position. The state of the filter will consist of the robot's state as well as the landmark positions. Let $\vec{z}_k$ denote the system state

$$\vec{z}_k = \begin{bmatrix} \vec{p}_k^R \\ \vec{p}_k^{l_1} \\ \vdots \\ \vec{p}^{l_N} \end{bmatrix} \triangleq \begin{bmatrix} \vec{p}_k^R \\ \vec{M}_k^L \end{bmatrix}$$

where N is the number of landmarks, and $\vec{M}_k^L$ is the vector of all landmark positions, which defines the "map" which is to be maintained by the SLAM algorithm.

1.2 The System Dynamic model

We assume that the dynamics of the robot's motion can be captured by a discrete time model of the following form:

$$\vec{p}_{k+1}^R = f(\vec{p}_k^R, u_k) + \vec{\eta}_k$$

where $\vec{p}_k^R$ is the robot's position and u_k is the robot's control input at time k, while $\vec{\eta}_k$ models process disturbances at t_k.

Since the landmarks are assumed to be stationary, their true positions are assumed to be constant over time:

$$\vec{p}^{\,l_j}_{k+1} \;=\; \vec{p}^{\,l_j}_{k} \qquad \forall\, j = 1, \ldots, N \;.$$

Hence, the dynamics take the form:

$$\vec{z}_{k+1} \;=\; \begin{bmatrix} \vec{p}^{\,R}_{k+1} \\ \vec{M}^{L}_{k+1} \end{bmatrix} \;=\; \tilde{f}(\vec{z}_k, u_k) \;=\; \begin{bmatrix} f(\vec{p}^{\,R}_k, u_k) \\ \vec{M}^{L}_{k} \end{bmatrix}$$

We assume that the robot can measure the range and bearing to the landmark. Let us assume for now that the robot can "see" all N landmarks. Then, the measurment is:

$$\vec{y}_k \;=\; \begin{bmatrix} r^{l_1}_k \\ \phi^{l_1}_k \\ \vdots \\ r^{l_N}_k \\ \phi^{l_N}_k \end{bmatrix} + \vec{\omega}_k = h(\vec{r}^{\,L}_k(\vec{z}_k), \vec{\phi}^{\,L}_k(\vec{z}_k)) + \vec{\omega}_k$$

where $r^{l_j}_k$ and $\phi^{l_j}_k$ are the range and bearing measurements of the j^{th} landmark:

$$r^{l_j}_k(\vec{p}^{\,R}_k, \vec{p}^{\,l_j}_k) \;=\; ||\vec{p}^{\,R}_k - \vec{p}^{\,l_j}_k|| = \sqrt{(x^R_k - x^{l_j}_k)^2 + (y^R_k - y^{l_j}_k)^2} \tag{1}$$

$$\phi^{l_j}_k(\vec{p}^{\,R}_k, \vec{p}^{\,l_j}_k) \;=\; Atan2[(y^R_k - y^{l_j}_k), (x^R_k - x^{l_j}_k)] + \theta^R_k \tag{2}$$

and $\vec{\omega}_k$ represents measurement noise. The vectors $\vec{r}^{\,L}_k$ and $\vec{\phi}^{\,R}_k$ represent all range and bearing measurments at time t_k. The measurement function $h(\cdot)$ is a function of both the robot's state and the map states.

2 Localization Updates: Naive case

Assume that an estimate (and covariance of the estimate) of the robot state and landmark map is available at time t_k: $\hat{z}_{k|k}$, $P_{k|k}$. Further, let us assume that the robot moves from its location at time t_k to a new location at time t_{k+1}. At t_{k+1} let us unrealistically assume that the robot can measure the range and bearing to *all* N landmarks. In this case, we use the EKF to update the estimate of the robot's position, as well as to improve the estimate of the relative locations of the landmarks.

Dynamic State Update: The dynamic update of the state estimate and its covariance is given by:

$$\hat{z}_{k+1|k} \;=\; \tilde{f}(\hat{z}_{k|k}, u_k) \tag{3}$$

$$P_{k+1|k} \;=\; \tilde{F}_k P_{k|k} \tilde{F}^T_k + V_k \tag{4}$$

where V_k is the covariance[1] and matrix F_k is the linearization of the robot's dynamics at the current best state estimate:

$$F_k = \frac{\partial}{\partial \vec{z}} \tilde{f}(\vec{z}_k, u_k)\big|_{(\vec{z},\vec{u})=(\hat{z}_{k|k},\vec{u}_k)}.$$

Note that F_k has an appealing structure:

$$\tilde{F}_k = \begin{bmatrix} \frac{\partial f(\vec{p}^R, u_k)}{\partial \vec{p}^R} & 0 & \cdots & 0 \\ 0 & I & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & I \end{bmatrix}_{(\vec{p}^R,\vec{u}_k)=(\vec{p}^R_{k|k},\vec{u}_k)} \triangleq \begin{bmatrix} F_k & 0 \\ 0 & I \end{bmatrix} \tag{5}$$

Note that the covariance matrix of the system state estimate, $P_{k|k}$, has the following structure:

$$P_{k|k} = \begin{bmatrix} P^{RR}_{k|k} & P^R L_{k|k} \\ P^{LR}_{k|k} & P^{LL}_{k|k} \end{bmatrix} \tag{6}$$

where P^{RR} is the uncertainty in the robot's state estimate, P^{LL} is the uncertainty in the landmark location estimates, and $P^{RL} = (P^{LR})^T$ is the correlation between the robot's undercertainty and the landmarks' uncertainties. Because of this structure, Equation (4) has the simplified structure:

$$P_{k+1|k} = \begin{bmatrix} F_k & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} P^{RR}_{k|k} & P^{RL}_{k|k} \\ P^{LR}_{k|k} & P^{LL}_{k|k} \end{bmatrix} \begin{bmatrix} F_k & 0 \\ 0 & I \end{bmatrix}^T + V_k = \begin{bmatrix} F_k P^{RR}_{k|k} F_k^T & F_k P^{RL}_{k|k} \\ (F_k P^{RL}_{k|k})^T & P^{LL}_{k|k} \end{bmatrix} + V_k.$$

Since the landmarks are assumed to be stationary (but with imprecisely known locations), their covariance does not change during the dynamic update! When the map becomes large (e.g., dozens of landmarks or more), this structure also leads to computational savings.

Measurement Update: The measurement update of the Kalman Filter is:

$$\vec{\nu}_{k+1} = \vec{y}_{k+1} - h(\hat{z}_{k+1|k}) \tag{7}$$
$$S_{k+1} = H_{k+1} P_{k+1|k} H^T_{k+1} + Q_k \tag{8}$$
$$K_{k+1} = P_{k+1|k} H^T_{k+1} S^{-1}_{k+1} \tag{9}$$
$$\hat{z}_{k+1|k+1} = \hat{z}_{k+1|k} + K_{k+1} \vec{\nu}_{k+1} \tag{10}$$
$$P_{k+1|k+1} = (I - K_{k+1} H_{k+1}) P_{k+1|k} \tag{11}$$

where Q_k is the covariance of the measurement noise, and H_{k+1} is the linearization of the meaasurement equation:

$$H_{k+1} = \frac{\partial h(z)}{\partial z}\Big|_{z=\hat{z}_{k+1|k}}. \tag{12}$$

[1]The covariance of the disturbance, which is assumed to be a zero mean Gaussian continuous random variable, can be found as $\mathbb{E}_{\vec{\eta}_k}[\vec{\eta}_k \vec{\eta}_k^T]$, where $\mathbb{E}$ denotes expectation.

3 Some Issues to Consider in an Actual EKF SLAM Implementation

An actual SLAM implementation has to incorporate *many* additional issues. Here we only summarize a few issues, and then some procedures to address these issues.

- Adding new landmarks to the map.
- Updating the map when only some of the landmarks are visible.
- Associating sensory data to landmark identity.

3.1 Incorporating new states into the map

Let us assume that at time t_k the robot has "built" a map consisting of $N-1$ landmarks. At time t_{k+1} the robot finds a *new* landmark, l_{new}, and chooses to incorporate it into its map to be the N^{th} landmark. Hence, at t_k the augmented state vector is $\vec{z}_k = \begin{bmatrix} \vec{p}_k^R & \vec{p}_k^{l_1} & \ldots & \vec{p}_k^{l_{N-1}} \end{bmatrix}^T$. At t_{k+1} the state vector is enlarged: $\vec{z}_k = \begin{bmatrix} \vec{p}_k^R & \vec{p}_k^{l_1} & \ldots & \vec{p}_k^{l_{N-1}} & \vec{p}_k^{l_{new}} \end{bmatrix}^T$. The addition of this new state also implies that the covariance matrix will grow in size, and that it must be properly initialized to include a reasonable estimate of the newly found landmark's position uncertainty.

Let us assume that the newly enlarged covariance matrix has the structure

$$\begin{bmatrix} P^{RR} & P^{RL} & P^{Rl_{new}} \\ P^{LR} & P^{LL} & P^{Ll_{new}} \\ P^{l_{new}R} & P^{l_{new}L} & P^{l_{new},l_{new}} \end{bmatrix}$$

where R indicates robot-related states, L indicates the states of the previously known landmarks, and l_{new} indicates the states associated with the newly found landmark.

Note that to update the covariance matrix during the measurement update, we will need to linearize the measurement equations. Hence, we need to have an initial estimate of the landmark's location (and the uncertainty in that location) in order to process all of the measurements at t_{k+1}. A logical estimate of the landmark's position is to add the robot-to-landmark distance (as measured using the range and bearing sensors) to the best estimate of the robot's current position (at $t_{k+1|k}$) to

$$\hat{p}_{k+1|k}^{l_{new}} = \hat{p}_{k+1|k}^R + \begin{bmatrix} r_{k+1}^{l_{new}} \cos(\phi_{k+1}^{l_{new}} + \hat{\theta}_k) \\ r_{k+1}^{l_{new}} \sin(\phi_{k+1}^{l_{new}} + \hat{\theta}_k) \end{bmatrix} \qquad (13)$$

If we assume that the noise experienced during the measurement of the new landmarks's location at t_{k+1} is uncorrelated with the noise in the robot's position at time t_k (an excellent assumption), then it would be natural to estimate the uncertainty of the robot's position at t_{k+1} to be

$$P^{l_{new}l_{new}}_{k+1|k+1} = J_{l_{new}R} P^{RR}_{k+1|k} J^T_{l_{new}R} + Q_{k+1}$$

where Q_{k+1} is the covariance, or uncertainty, in the measurement of the landmark's position (relative to the robot). The matrix $J_{l_{new}R}$ is a *Jacobian matrix* which measures the sensitivity of the landmark's location to changes in the robot's position. The Jacobian can generally be calculated as follows. If we have a formula for the landmark's position as a function of the robot's position (e.g., exactly the formula (13), then the Jacobian is the linearization of this formula:

$$J_{l_{new}R} = \frac{\partial \vec{p}^{l_{new}}}{\partial \vec{p}^R}.$$

However, after initializing the landmark's location and uncertainty, we will carry out a complete measurement update step at t_{k+1}, which will incorporate the measurement uncertainty related to the new landmark. Hence, we would "double count" the measurement uncertaintly in this proposed approach. I.e., conceptually we would like to estimate $P^{l_{new}l_{new}}_{k+1|k}$, the covariance *before* we take into account the measurement (and its uncertainty) of the new landmark at t_{k+1} upon the entire system state. Hence,

$$P^{l_{new}l_{new}}_{k+1|k+1} = J_{l_{new}R} P^{RR}_{k+1|k} J^T_{l_{new}R}$$

Before completing the measurement update at t_{k+1}, we also need initial estimates for $P^{Rl_{new}}$ and $P^{Ll_{new}}$ at $t_{k+1|k}$. If one carefully scrutinizes the construction of the covariance formulas in the measurement update equations, then a reasonable estimate of these matrices can be constructed as:

$$P^{Rl_{new}}_{k+1|k} = P^{RR} J^T_{l_{new}R} \tag{14}$$

$$P^{l_{new}L}_{k+1|k} = J_{l_{new}R} P^{RL} \tag{15}$$

3.2 Updating the Map with Limited Landmark Visibility

At any given instant, not all of the previously found landmarks are likely to be visible. This lack of visibility needs to be taken into account when updating the robot and landmark state estimates.

Let us assume that at time t_k, N landmarks can be viewed, and the EKF framework is used to find the best estimate of the system state and its uncertainty at t_k: $\hat{z}_{k|k}$, $P_{k|k}$. Now, assume that during the robot's movement between time t_k and time t_{k+1} that the j^{th} landmark becomes hidden. The dynamic update step is not affected by the disappearance of landmark l_j before the measurement update at t_{k+1}:

$$\hat{z}_{k+1|k} = \tilde{f}(\hat{z}_{k|k}, u_k); \qquad P_{k+1|k} = \tilde{F}_k P_{k|k} \tilde{F}^T_k + v_k.$$

But, the measurement update *will* be affected by the change in the number of landmarks. Let the measurement vector at t_{k+1} be denoted by $\vec{y}^{Nj}$:

$$\vec{y}^{N\backslash j} = \begin{pmatrix} r^{l_1}_{k+1} & \phi^{l_1}_{k+1} & \cdots & r^{l_{j-1}}_{k+1} & \phi^{l_{j-1}}_{k+1} & r^{l_{j+1}}_{k+1} & \phi^{l_{j+1}}_{k+1} & \vdots & r^{l_N}_{k+1} & \phi^{l_N}_{k+1} \end{pmatrix} \tag{16}$$

When all N landmarks are visible, $\vec{y}$ has dimension $2N$. When landmark l_j disappears, $\vec{y}^{N\backslash j}$ has dimension $2(N-1)$.

There are two ways to see how the disappearance of landmark l_j will affect the measurement update–they both lead to the same outcome.

In the first conceptual approach, assume that the state vector $\vec{z}$ still incorporates the entire map of all landmarks, including the non-visible landmark. Let $h^{N\backslash j}(\vec{z})$ denote the measurement function, where the j^{th} landmark measurement is not incorporated (as in Equation (16)). Then the linearization of the measurement function (as needed to compute Equations (8)-(11)) produces a $2(N-1) \times (2N+3)$ matrix $H^{N\backslash j}_{k+1}$

$$H^{N\backslash j}_{k+1} = \frac{\partial h^{N\backslash j}}{\partial z}\Big|_{z=z_{k\hat{+}1|k}}.$$

However, note that

$$\frac{\partial h^{N\backslash j}}{\partial r^{l_j}} = \frac{\partial h^{N\backslash j}}{\partial \phi^{l_j}} = 0$$

and hence the linearization of the measurement equation, H_{k+1} will have a row of zeros associated with the missing landmark index. In the linearized covariance update equation (8), these zeros will multiply the j^{th} row and j^{th} column of $P_{k+1|k}$, causing no updates to the elements in the j^{th} row and column of $Pk+1|k$.

In the second conceptual approach, we assume that at t_{k+1} the actual map shrinks in size–i.e. the j^{th} landmark state is removed from the map: $\vec{z}^{N\backslash j}_{k+1} = \begin{pmatrix} \vec{p}^{T}_{k+1} & \vec{p}^{l_1}_{k+1} & \cdots & \vec{p}^{l_{j-1}}_{k+1} & \vec{p}^{l_{j+1}}_{k+1} & \cdots & \vec{p}^{l_N}_{k+1} \end{pmatrix}$. Hence, in the measurement update takes the form:

$$y^{N\backslash j}_{k+1} = h^{N\backslash j}(\vec{z}^{N\backslash j}) + \vec{\omega}_k \ .$$

In order to carry out the measurement updates one must construct a new covariance matrix $P^{N\backslash j}_{k+1|k}$ by removing the j^{th} column and j^{th} row of $P_{k+1|k}$.

3.3 A Simplified Data Association Solution

The EKF SLAM approach is based on the important assumption that each of the landmarks is *identifiable* during each measurement update. That is, it is assumed that one can reliably associate a specific landmark index to each landmark data measurement. In practice, this *data association* problem can be difficult to solve.

A very simple data association algorithm can be constructed as follows. Assume that at t_k the labels of each landmark are known. Assume that the range and bearing to the same N

land marks can be found at t_{k+1}. However, the labels (or indices) of these landmarks are not known, meaning that we don't know how to associated the landmark measurements at t_{k+1} to the labels determined at t_k.

Assume, for the purposes of the data association process, that an arbitrary labeling is assigned to the landmark measurements at t_{k+1}. The goal is now to find the proper matching of the measurements at t_{k+1} with the labels of t_k. To assist in the proper matching of data to labels, calculate the following *innovations*:

$$\nu_{k+1}^{ij} \;=\; y_{k+1}^{i} - h^{j}(\hat{p}_{k+1|k}^{l_j})$$

where y_{k+1}^i is the landmark data which is temporarily assigned index i at t_{k+1}, and $h^j(\hat{p}_{k+1|k}^{l_j})$ is the predicted location of landmark j (whose identity was properly established at t_k) after the robot moves to its new position at t_{k+1}. Hence, ν^{ij} is the *data association error* between the tentative pairing of landmark data i acquired at time t_k with landmark index j at t_k.

Naively, for landmark j a t_k, the landmark k at measurement time t_{k+1} is assumed to be the best match to l_j if $k = \arg\min_i \nu^{ij}$. However, this naive argument must be slightly modified so that we measure the "error" in the most suitable way.

Let S_{k+1} denote the innovation covariance, Equation (8), at t_{k+1}. Let the covariance associated with the innovation ν^{ij} be dnoted S^{ij}. This covariance is computed as:

$$S_{k+1}^{ij} = H_{k+1}^{j} P_{k+1|k} (H_{k+1}^{j})^{T} + Q_{ii}$$

where H^j is the linearization of the measurement equation associated with the j^{th} landmark at time t_k.

$$H_{k+1}^{j} = \begin{pmatrix} \frac{\partial h^j}{\partial \vec{p}^R} & 0 & \cdots & \frac{\partial h^j}{\partial \vec{p}^{l_j}} & 0 & \cdots & 0 \end{pmatrix}\Bigg|_{\vec{p}_{k+1|k}^{R},\vec{p}_{k+1|k}^{l_j}} .$$

Using this covariance, the *association error* arising from the belief that the i^{th} measurement at t_{k+1} is associated with the j^{th} landmark at time t_k is given by:

$$\chi_{ij}^{2} \;=\; \nu^{ij}(S^{ij})^{-1}\nu^{ij}.$$

The most likely landmark at t_{k+1} to associated with landmark l_j at t_k is the one which minimizes this association error:

$$l_i = \arg\min_{i} \;\; \chi_{ij}^{2} \; .$$