Simultaneous Localization And Mapping via Extended Kalman Filter

Giovanni Ciatto
giovanni.ciatto@studio.unibo.it

November 12, 2016

1 Introduction

In the context of robotics, Simultaneous Localization And Mapping (SLAM) is the problem of a robot dynamically constructing a map of an unknown environment and concurrently localizing itself within it, while exploring such environment. It may seem a chicken-or-egg problem, since a map is needed for localization and a frame of reference is needed for building a map, but the problem has been studied and solved both from a theoretical (probabilistic) and practical point of view in [TBF05].

SLAM involves a robotic agent being at least able to move within, and gather information about, the environment. Wheels and laser scanners are common choices to satisfy such requirements. Generally speaking, we will refer to them as *actuators* and *exteroceptive sensors*, respectively. Optionally, the robot may be equipped with one or more *odometric* sensors allowing it to actually measure its own movement, which should otherwise be inferred, e.g., by the speed imposed to the wheels. Generally speaking we call them *proprioceptive sensors*.

It is well understood in robotics that sensory data and inferred movement information are inherently noisy. From a probabilistic point of view this means that, if the robot is keeping track of its own position, the *uncertainty* of about such position increases as the robot moves. Conversely, supposing the robot is able to detect some objects (*landmarks*, in jargon) within its surroundings, recognize them for a number of observations, and estimate the relative position between itself and each landmark, then it can use such information to reduce the uncertainty about its own position. Such a correlation between position estimation and landmark measurement is clearly explained in [Bre14, Unit C].

The generic approach to SLAM requires the following models to be properly defined:

Motion model: describes how the robot updates the estimation of its own position and orientation according to the proprioceptive sensor data. It depends on the degrees of freedom of the robot and the nature of the available data. For instance, in this report, we consider the case of a differential robot constrained to move on a plane. So the robot pose variables are x, y and θ (the *bearing*, in jargon) and the proprioceptive data consist of the last velocity values v_l and v_r imposed to the wheels motors.

Inverse Observation model: describes how exteroceptive sensor data is used to deduce the landmarks positions, taking into account the current estimation of the robot position and orientation too. It depends on the nature of the data and the number of dimensions required to localize a landmark on the map. For example, in this report, we consider the case of a laser sensor providing, for each landmark, both its distance and angle w.r.t. the laser sensor. So the exteroceptive data consist of (ρ, α) pairs, which are used to deduce the landmark position (x_m, y_m) on the map.

Direct Observation model: describes how to predict the expected exteroceptive sensor data for a known landmark. From a conceptual point of view, it's the inverse function of the Inverse Observation model. E.g., for our concerns, the Direct Observation Model takes into account the current estimation of the robot position and orientation (x, y, θ) and some known landmark position (x_m, y_m) and computes the expected sensor data (ρ, α).

Such models are exploited by the following conceptual flow in order to produce a new estimation of the current position and orientation. It takes into account the previous estimation, and the last available proprioceptive and exteroceptive data:

1. *Prediction* phase: a basic estimation of the new position is achieved by combining the previous estimation with the proprioceptive data according to the motion model;

2. *Recognition* phase: the information from the exteroceptive data is analyzed in order to understand if it corresponds to an already known landmark or a new one. This normally exploits the inverse observation model: if it's a new landmark it must be added to the map;

3. *Observation* phase: when an already-known landmark is found, the current exteroceptive data relative to such a landmark (the *measurement*) is compared to the estimation achieved through the direct observation model for the same landmark (the *expected* measurement);

4. *Correction* phase: the difference between the measurement and the expected measurement is used to improve the basic estimation from the prediction phase and the landmark position estimation.

Our contribution consists of a practical tutorial for solving the SLAM problem through the *Extended Kalman Filter* (EKF), which supposes the robot state and the landmarks positions to be random vectors having a multivariate normal distribution, whose principal moments are estimated according to the phases described above.

2 Models and Reference Frames

Appendix

2.1 Affine transform on the plane

Here we recall the affine transformation equation mapping each point from a source 2D plane to another translated, scaled and rotated 2D plane:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} s_x \cdot \cos x & -\sin x \\ \sin x & s_y \cdot \cos x \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} \tag{1}$$

and its inverse:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \frac{1}{s_x} \cdot \cos x & \sin x \\ -\sin x & \frac{1}{s_y} \cdot \cos x \end{pmatrix} \cdot \begin{pmatrix} x' \\ y' \end{pmatrix} - \begin{pmatrix} t_x \\ t_y \end{pmatrix} \tag{2}$$

where $(x,\, y)^\top$ is a point in the source reference frame, $(x',\, y')^\top$ is the transformed reference frame, $(t_x,\, t_y)^\top$ is a translation vector, s_x and s_y are scale factors for the horizontal and vertical coordinates, respectively, and θ is the angle between the source reference frame abscissas axis and the transformed frame one.

2.2 Multivariate normal distribution

Here we recall the notion of multivariate normal distribution and its linearity and geometric properties. Let $\mathbf{x} \in \mathbb{R}^n$ be a normally distributed vector having mean $\boldsymbol{\mu} \in \mathbb{R}^n$ and covariances matrix $\Sigma \in \mathcal{M}_{n\times n}(\mathbb{R})$, then we write:

$$\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu},\, \Sigma)$$

meaning that the probability density function of $\mathbf{x}$ is the multidimensional Gaussian function:

$$pdf(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^n \cdot |\Sigma|}} \cdot \mathrm{e}^{-\frac{1}{2}\cdot(\mathbf{x}-\boldsymbol{\mu})^\top \cdot \Sigma^{-1} \cdot (\mathbf{x}-\boldsymbol{\mu})}$$

Linearity. Let $\mathbf{y} \in \mathbb{R}^m$ be a random vector, obtained by linearly combining a number of normally distributed independent random vectors $\mathbf{x}_i \sim \mathcal{N}(\boldsymbol{\mu}_i,\, \Sigma_i)$:

$$\mathbf{y} = A_1 \cdot \mathbf{x}_1 + A_2 \cdot \mathbf{x}_2 + \ldots + \mathbf{b}$$

where $A_i \in \mathcal{M}_{m\times n}(\mathbb{R})$ are transformation matrices and $\mathbf{b} \in \mathbb{R}^m$ is a constant vector, then $\mathbf{y}$ is normally distributed too, having mean $\boldsymbol{\mu}_y$ and covariances matrix Σ_y, expressed as follows:

$$\boldsymbol{\mu}_y = \mathbf{b} + \sum_i A_i \cdot \boldsymbol{\mu}_i \tag{3}$$

$$\Sigma_y = \sum_i A_i \cdot \Sigma_i \cdot A_i^\top \tag{4}$$

Representation. Usually, a Multivariate normal distribution $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ is imagined and represented as an hyper-ellipsoid centered in $\boldsymbol{\mu}$, whose axes are the (left) singular vectors of Σ. Such an ellipsoid is scaled (w.r.t. each axis) according to the singular values of Σ. In order to produce a rendering of such an hyper-ellipsoid (which is an ordinary ellipse in the 2D case), it is sufficient to produce a singular-values-decomposition of the covariances matrix:

$$\Sigma \overset{svd}{=} V \cdot D \cdot V^\top = \begin{pmatrix} \mathbf{v}_1 & \cdots & \mathbf{v}_n \end{pmatrix} \cdot \begin{pmatrix} d_1^2 & 0 & \cdots & 0 \\ 0 & d_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & d_n^2 \end{pmatrix} \cdot \begin{pmatrix} \mathbf{v}_1^\top \\ \vdots \\ \mathbf{v}_n^\top \end{pmatrix}$$

where the i-th column of V, namely $\mathbf{v}_i$, is the versor identifying the direction of the i-th axis of the ellipsoid, and the i-th diagonal element of D, namely d_i^2 can be thought to be the variance of the distribution according to the i-th axis, i.e., d_i can be thought to be its standard deviation.

In the 1-dimensional case it is common to represent the k-standard deviation interval, i.e., the circular interval centered on the mean and including each point whose distance from the mean is lower than k times the standard deviation. Analogously, the k-th ellipsoid centered in $\boldsymbol{\mu}$ can be represented for the n-dimensional case by applying the following affine transformation to each point on the unary hyper-sphere:

$$k \cdot \begin{pmatrix} \mathbf{v}_1 & \cdots & \mathbf{v}_n \end{pmatrix} \cdot \begin{pmatrix} d_1 & 0 & \cdots & 0 \\ 0 & d_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & d_n \end{pmatrix} \cdot \mathbf{c} + \boldsymbol{\mu}$$

where $\mathbf{c} \in \{(x_1, \ldots, x_n)^\top \mid x_1^2 + \ldots + x_n^2 = 1\}$.

References

[Bre14] Claus Brenner. Slam lectures. https://www.youtube.com/playlist?list=PLpUPoM7Rgzi_7YWn14Va2FODh7LzADBSm, 2014.

[Fri13] Albin Frischenschlager. Slam algorithm. http://ti.tuwien.ac.at/cps/teaching/courses/networked-embedded-systems/seminar.pdf, 2013.

[RB] Søren Riisgaard and Morten Rufus Blas. Slam for dummies. https://ocw.mit.edu/courses/aeronautics-and-astronautics/16-412j-cognitive-robotics-spring-2005/projects/1aslam_blas_repo.pdf.

[Sol14] Joan Solà. Simultaneous localization and mapping with the extended kalman filter. http://www.iri.upc.edu/people/jsola/JoanSola/objectes/curs_SLAM/SLAM2D/SLAM%20course.pdf, 2014.

[TBF05] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.