Expectation-Maximisation Maximum Likelihood Estimation for Inertial/Visual SLAM

Zoran Sjanic* † Martin Skoglund*, and Fredrik Gustafsson*
*Linköping University †Saab Aeronautics

Problem description

The Simultaneous Localisation and Mapping (SLAM) problem aims at estimating the state of a moving platform simultaneously with building a map of the local environment. There are essentially three classes of algorithms. EKF-SLAM and FastSLAM solve the problem on-line, while Nonlinear Least Squares (NLS) is a batch method. All of them scales badly with either the state dimension, the map dimension or the batch length.

Contribution

We investigate the EM algorithm for solving a generalized version of the NLS problem. This EM-SLAM algorithm solves two simpler problems iteratively, hence it scales much better with dimensions. The iterations switch between state estimation, where we propose an Extended Rauch-Tung-Striebel smoother, and map estimation, where a quasi-Newton method is suggested.

EM-SLAM

A state space model for visual/inertial SLAM can be described by

$$x_t = f(x_{t-1}, u_t, w_t),$$
$$y_t = h_t(x_t, \theta) + e_t,$$

where the measurement noise $e_t \sim \mathcal{N}(0, R)$, the process noise, w_t, is considered white with mean zero and covariance Q. u_t are inputs given by the inertial sensors from which the navigation states, $x = [p, v, q]^T$, are computed. The function h relates camera measurements, states and parameters. The parameter vector θ, consists of landmark coordinates. Contrary to traditional SLAM state space models the map is seen as a parameter which parametrises the measurement likelihood function. From this state space formulation it is straightforward to define the joint likelihood for the batch problem

$$p_\theta(Y, X) = \prod_{t=1}^{N} p_\theta(y_t|x_t)p(x_t|x_{t-1}). \tag{2}$$

where the platform states, X, are considered to be latent variables. Notice that the process model does not depend explicitly on the parameter θ, which simplifies the expectation step significantly.

E-step

Given the joint likelihood from (2) the expectation step have the following form

$$\mathcal{Q}(\theta, \theta_k) = \mathbf{E}_{\theta_k}\left\{\log\left[\prod_{t=1}^{N} p_\theta(y_t|x_t)p(x_t|x_{t-1})\right]\middle|Y\right\}, \tag{3}$$

where the measurement likelihood is given by the PDF $p_\theta(y_t|x_t) = p_\theta(e_t) = p_\theta(y_t - h_t(x_t, \theta))$ and the state transition density, $p(x_t|x_{t-1})$, does not depend on θ. Assuming that the likelihood has Gaussian distribution the expectation (3) becomes

$$\mathcal{Q}(\theta, \theta_k) = \text{const.} - \mathbf{E}_{\theta_k}\left\{\sum_{t=1}^{N}\frac{1}{2}\|y_t - h_t(x_t,\theta)\|^2_{R_t^{-1}} + \log p(x_t|x_{t-1})\middle|Y\right\}$$
$$= -\sum_{t=1}^{N}\mathbf{E}_{\theta_k}\left\{\frac{1}{2}\|y_t - h_t(x_t,\theta)\|^2_{R_t^{-1}}\middle|Y\right\} + \text{const.} \tag{4}$$

where all the terms not depending on θ are lumped into a constant term. The measurement function is approximated as

$$\mathcal{Q}(\theta, \theta_k) \approx \text{const.} - \frac{1}{2}\sum_{t=1}^{N}\Big(\|y_t - h_t(\hat{x}_{t|N},\theta)\|^2_{R^{-1}} +$$
$$\text{Tr}(R^{-1}\nabla_x h_t(\hat{x}_{t|N},\theta)P^s_{t|N}(\nabla_x h_t(\hat{x}_{t|N},\theta))^T)\Big) \tag{5}$$

Here, $\hat{x}_{t|N}$ is the smoothed estimate of the latent variable and $P^s_{t|N}$ is its covariance.

Algorithm 1 Extended Rauch-Tung-Striebel Smoother (E-RTS)

Input: measurements $Y = \{y_1, \dots, y_N\}$, inputs $U = \{u_1, \dots, u_N\}$, covariance matrices Q and R, parameter estimate θ_k
Output: smoothed state estimates $\hat{x}^s = \{\hat{x}_{1|N}, \dots, \hat{x}_{N|N}\}$, covariances $P^s_{1:N|N}$

Run a forward Extended Kalman filter (EKF) where measurement equation uses fixed value of the parameter $\theta = \theta_k$, and store time and measurement updates for states, $\hat{x}_{t|t}$, $\hat{x}_{t|t-1}$, the covariances $P_{t|t}$, $P_{t|t-1}$ and the Jacobians of the dynamics, $F_{t-1} = \frac{\partial}{\partial x}f(x_{t-1}, u_t, w_t)|_{x_{t-1}=\hat{x}_{t-1|t-1}, w_t=0}$.
$P^s_{N|N} := P_{N|N}$
for $t = N : 2$ **do**
  $S_{t-1} := P_{t-1|t-1}F^T_{t-1}P^{-1}_{t|t-1}$
  $\hat{x}_{t-1|N} := \hat{x}_{t-1|t-1} + S_{t-1}(\hat{x}_{t|N} - \hat{x}_{t|t-1})$
  $P^s_{t-1|N} := P_{t-1|t-1} + S_{t-1}(P^s_{t|N} - P_{t|t-1})S^T_{t-1}$
end for

M-step

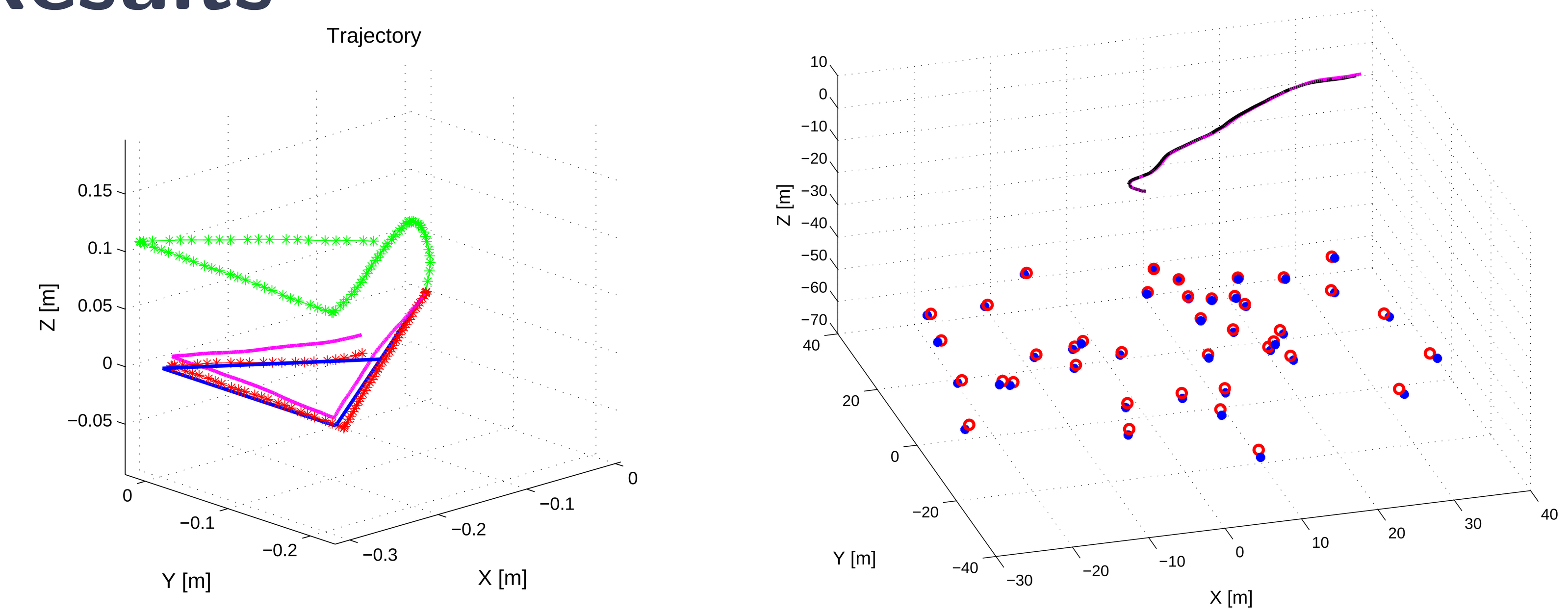
Maximisation of the $\mathcal{Q}$-function can be done using standard optimisation software. For our particular setting, the function to be minimised is a nonlinear function of the parameters. We use a quasi-Newton method called BFGS since it is quite efficient but other choices are also possible.

Algorithm 2 M-step (Quasi-Newton minimisation with BFGS Hessian update)

Input: smoothed states $\hat{x}^s$, measurements Y, initial parameters θ_k, inverse Hessian approximation $B_0 \approx \left(\nabla^2_\theta\mathcal{Q}(\theta_k, \theta_k)\right)^{-1}$, termination threshold ε. **Output:** θ_{k+1}.

$i := 0$, *terminate* := **false**, $\theta^i := \theta_k$
while not *terminate* **do**
  Compute search direction: $p_i := -B_i\nabla_\theta\mathcal{Q}(\theta^i, \theta_k)$
  Update the parameter: $\theta^{i+1} := \theta^i + \alpha_i p_i$ where α_i is the step length computed by line search ensuring decrease in cost
  Compute: $s_i = \theta_{i+1} - \theta_i$, $r_i = \nabla_\theta\mathcal{Q}(\theta^{i+1}, \theta_k) - \nabla_\theta\mathcal{Q}(\theta^i, \theta_k)$
  Update the inverse Hessian: $B_{i+1} := \left(I - \frac{s_i r_i^T}{r_i^T s_i}\right)B_i\left(I - \frac{r_i s_i^T}{r_i^T s_i}\right) + \frac{s_i s_i^T}{r_i^T s_i}$
  if $\|\nabla_\theta\mathcal{Q}(\theta^{i+1}, \theta_k)\| < \varepsilon$ **then**
    terminate := **true**
  else
    $i := i + 1$
  end if
end while
$\theta_{k+1} := \theta^{i+1}$

Results



Left: Trajectory estimates from EM-SLAM (magenta) and standard NLS-SLAM (red), both initialised with a linear method (green) and compared with ground truth (blue). Right: The setup used in the MC simulations. True trajectory is in black and true landmarks are red circles. One of 30 simulation results is depicted as magenta trajectory and blue landmarks.

This work has been submitted to IEEE Transactions on Robotics. For more details please refer to the draft in Zoran Sjanic's PhD thesis.