1 Trajectory Optimization

Motivation

Above we discussed MPE and MAP estimation in discrete state spaces, and we we want a recipe for doing this in continuous state spaces as well. In this chapter we focus on trajectory optimization, which is useful for drones, autonomous cars, and any type of mobile robot. Because all these examples operate in continuous state spaces, we need a way to do MAP estimation in those spaces. As we will see below, this involves fusing information from multiple sensors, and hence trajectory optimization is an instance of "sensor fusion".

1.1 A Motivating Example

Figure 1: A motivating car localization example, where red indicate GPS measurements, green are odometry measurements, and blue are landmark measurements.

Assume we have an autonomous vehicle, driving on the highway, and

1. we have regular GPS measurements;
2. we have odometry measurements from sensors on the wheels;
3. from time to time, we observe a landmark whose absolute location in the world is known;
4. optionally, we know that GPS is biased: atmospheric effects often add a systematic error, e.g., it always thinks we are two meters to the left.

The situation is illustrated in Figure 1. How can we fuse all that information into an estimate of where the car currently is?

1.2 Factor Graph Representation

We can represent the unknown variables and the measurements on them using a factor graph, as shown in Figure 2. In this example we have 5 variables for $x_0 ... x_4$, and factors for all measurements discussed above. There is one extra factor, which represent a prior on the initial location of the car. "Odometry factors" are binary (connected to two unknowns) whereas all other variables are unary.

Exercise

Modify the factor graph for the case that we have an (unknown) bias b, as discussed above.

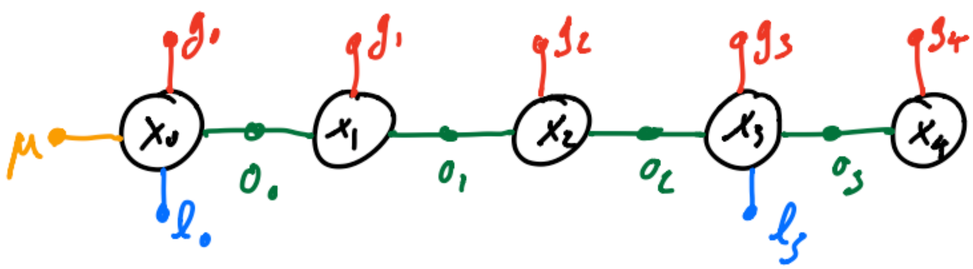

Figure 2: The factor graph corresponding to Figure 1.

1.3 A 1-D Version is Linear!

A MAP solution in continuous spaces is to maximize the posterior probability density of the unknowns given all the measurements. In 1D, the above problem will be a linear least squares problem.

To see this, we first need to become specific about variables and measurement models. First, in a simple 1D example, let's define our unknowns as the car location $x_k \in R$ at time t_k. Then the measurement functions are:

1. GPS: $h_{GPS}(x_k) = x_k$
2. Odometry from time t_k to time t_{k+1}: $h_{ODO}(x_k, x_{k+1}) = x_{k+1} - x_k$
3. Landmark observations: $h_{LM}(x_k; l_k) = l_k - x_k$, where $l_k \in R$ is the location of the landmark at time t_k. In other words, we just measure the *signed* distance to the landmark[1].
4. In case GPS is biased, we modify the GPS measurement model: $h_{GPS}(x_k) = x_k + b$

1.4 Trajectory Optimization and Bayes Law

To find the optimal trajectory $X^K = \{x_k\}_{k=0}^{K-1}$, we want to maximize the posterior density

$$X^{K*} = \arg\max p(X^K | G^K, O^{K-1}, Z^K)$$

given all GPS measurements G^K, odometry measurements O^{K-1}, and landmark measurements Z^K. Here a superscript means: all things up to and including time t_{K-1}.

The posterior density can be gotten by Bayes law, as before:

$$\begin{aligned} p(X^K | G^K, O^{K-1}, Z^K) &\propto p(X^K) l(X^K; G^K) l(X^K; O^{K-1}) l(X^K; Z^K) \\ &= p(x_0) \prod_k l(x_k; g_k) \prod_k l(x_k, x_{k+1}; o_k) \prod_k l(x_k; z_k, l_j) \end{aligned} \tag{1}$$

where we assumed that the only prior information is where we start out, i.e., $p(X^K) = p(x_0)$.

We now need to evaluate the likelihoods $l(.)$. Let us take GPS as an example. Remember that the likelihood $l(x_k; g_k)$of the state x_k given the GPS measurement g_l is any function proportional to the conditional density $p(g_k|x_k)$. When we assume a measurement is corrupted by Gaussian noise, we have seen before that - for the GPS measurements for example - we have

$$p(g_k|x_k) = \mathcal{N}(z; h(x_k), R) = \frac{1}{\sqrt{|2\pi R|}} \exp\left\{ -\frac{1}{2} \|h_{GPS}(x_k) - g_k\|_R^2 \right\} \tag{2}$$

[1] Absolute distance would be a nonlinear measurement.

Hence, a good likelihood is

$$l(x_k; g_k) = \exp\left\{-\frac{1}{2}\|h_{GPS}(x_k) - g_k\|_R^2\right\} = \exp\left\{-\frac{1}{2}\|x_k - g_k\|_R^2\right\} \tag{3}$$

where we used the fact that $h_{GPS}(x_k) = x_k$. The normalizing constant of the Gaussian does not depend on x_k, and hence it does not matter for our trajectory optimization problem.

To find the MAP solution, equation 1 would then multiply many of these likelihoods together, and then we would have to maximize the posterior with respect to the variables X^K. Unfortunately, it is not obvious how to do this.

1.5 Trajectory Optimization as Least-Squares

In continuous spaces, we can take the negative log, and instead minimize a linear least-squares error. To see this, we now apply the most amazing trick in continuous MAP estimation: we take the negative log of Equation 1, turning a maximization of products into a minimization of sums instead:

$$X^{K*} = \arg\min\{-\log p(x_0)\} + \sum_k nll(x_k; g_k) + \sum_k nll(x_k, x_{k+1}; o_k) + \sum_k nll(x_k; z_k, l_k). \tag{4}$$

Because all our measurement functions are linear, this becomes a linear least-squares problem. E.g., for GPS we have:

$$-\log l(x_k; g_k) = -\log\exp\left\{-\frac{1}{2}\|x_k - g_k\|_R^2\right\} = \frac{1}{2}\|x_k - g_k\|_R^2$$

Doing this for all measurements, we obtain our final objective function:

$$X^{K*} = \arg\min \frac{1}{2}\|x_0 - \mu\|_P^2 + \sum_k \frac{1}{2}\|x_k - g_k\|_R^2 + \sum_k \frac{1}{2}\|x_{k+1} - x_k - o_k\|_Q^2 + \sum_j \frac{1}{2}\|x_k - l_k - z_k\|_P^2. \tag{5}$$

where μ and P are the mean and covariance, respectively, of the prior on the first car location.

The factor graph can now be interpreted as a graphical representation of this linear least-Squares (LLS) problem, where every factor corresponds to a quadratic term, connected to the variables that play a role in it. To solve an LLS problem, we can use any number standard solvers, but our favorite solver is GTSAM, which allows us to specify the factor graph directly in code, using "GaussianFactors" in the linear-Gaussian case. But the nice property of it is that can also be used to solve nonlinear problems, which will be discussed in the next chapter.

Exercises

1. Modify the LLS objective for the case that we have an (unknown) bias b, as discussed above.

2. What are we now optimizing over?

Summary

We briefly summarize what we learned in this section:

1. Autonomous driving provides a simple motivating example.

2. We can represent the problem graphically using a factor graph.

3. In 1-D, this problem is linear, although we will not be so lucky in 2D.
4. We then turn the MAP estimate of the trajectory into a trajectory optimization problem.
5. Finally, by converting to (negative) log-space, we obtain an easy linear least-squares problem.