**Stochastic Optimal Control-based Framework for Robot Motion planning**

Ammar Mohamed Nasher Abbas, Ali-akbar Agha-mohammadi, Suman Chakravorty

Department of Aerospace Engineering, University of Texas A&M

---

**Introduction**

In the robot motion planning under uncertainty, we are seeking a policy that brings the robot from a start point A to a goal point B such that the probability of collision with obstacles is minimized.

Which path (policy) to choose?

---

**The system Model**

The system model describes the evolution of the system state. It is algebraically expressed by:

\[
\begin{align*}
\dot{x} &= f(x, u, w) \\
\dot{u} &= f(u, v, z) \\
\end{align*}
\]

A unicycle robot would be represented by:

- x-y coordinates
- Linear velocity
- Linear velocity noise
- Heading angle
- Angular velocity
- Angular velocity noise
- State of the robot at the time step
- Motion planner (controller)

In red, the mobile robot sensor. In white, the landmarks.

---

**The Observation Model (Sensor model)**

The observation model expresses the evolution of the observation. It is algebraically expressed by:

\[
\begin{align*}
Z_k &= h(x_k, u_k, v_k, z_k) \\
\end{align*}
\]

The sensor is expressed algebraically by:

Sensor measurements

Sensor noise

where the Euclidean distance is calculated as follows:

\[
\begin{align*}
\|x, y\|^2 &= (x - x_l)^2 + (y - y_l)^2 \\
\end{align*}
\]

In the robot motion planning under uncertainty, we are seeking a policy that brings the robot from a start point A to a goal point B such that the probability of collision with obstacles is minimized.

---

**The State Estimator (filter)**

The state estimator generates the “belief” of the system which can be computed recursively by:

\[
\begin{align*}
\hat{x}_k &= K_k z_k + \hat{x}_{k-1} \\
\end{align*}
\]

Kalman Filter

Here we focus on the Kalman filtering as the state estimator. Suppose we have the following linear Gaussian system:

\[
\begin{align*}
\dot{x}_k &= Ax_k + Bu_k + Cw_k \\
\end{align*}
\]

where the Euclidean distance is calculated as follows:

\[
\begin{align*}
\|x, y\|^2 &= (x - x_l)^2 + (y - y_l)^2 \\
\end{align*}
\]

Following is the sensor model that measures the distance of the robot from some landmarks.

Example

Kalman Filter performs the estimation in two steps: prediction and update.

1. In the prediction step, the belief state is predicted:

\[
\begin{align*}
\hat{P}_k &= A\hat{P}_{k-1}A^T + Q_k \\
\end{align*}
\]

2. In the update step, the next belief state is calculated:

\[
\begin{align*}
\hat{P}_k &= (I - KH_k)\hat{P}_k \\
\end{align*}
\]

---

**Motion Planner (controller)**

Motion planner (controller) generates the actions for the robot to move from the current state to the goal state.

The last major block in the stochastic control is where decisions are being made to generate control according to:

Cost-to-go:

\[
J^* = \min_{\pi(b_k)} J^*(b_k)
\]

---

**References**


