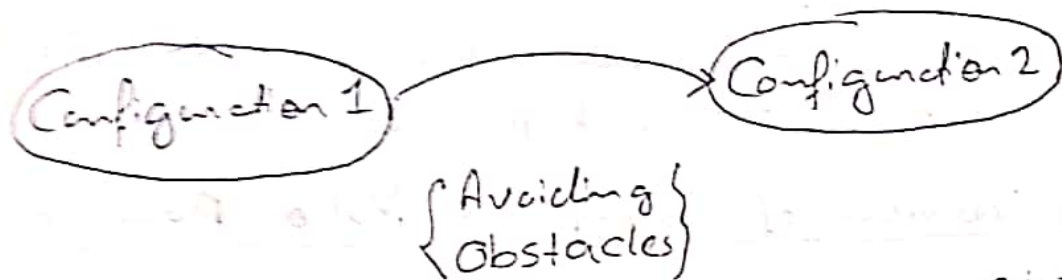


1

Introduction

(Automatic motion planning)

⇒ The goal is to be able to specify a task in a high level language and have the robot automatically compile this specification into a set of low-level motion primitives, or feedback controllers, to accomplish the task.



⇒ The Key Problem is to make sure no point on the robot hit an obstacle, so we need a way to represent the location of all the points on the robot.

→ This representation is called the **Configuration** of the robot.

→ **Configuration Space** is the space of all configurations the robot can achieve.

⇒ The Configuration Space is generally non-Euclidean.

⇒ The dimensions of the Configuration Space is equal to number of independent variables in the representation of the Configuration, also known as degree of freedom (DOF).

⇒ The problem is to find a curve in the configuration space that connects the start and goal points and avoids all configuration space obstacles that arise due to obstacle in the space.

(Nonholonomic constraint)

⇒ When a robot has more degrees of freedom than required to complete its task, the robot is called redundant.

⇒ When a robot has many extra degree of freedom, then it is called hyper-redundant.

★ Overview of Concepts in Motion Planning

① Task

Navigation

{ Problem of finding a collision-free motion for the robot system from one configuration to another. }

Coverage

Localization

mapping

{ Problem of passing a sensor or tool over all points in a space }

{ Mapping is the problem of exploring & sensing an unknown environment }

{ Problem of using map to interpret sensor data to determine the configuration of the robot. }

★ ② Properties of the Robot

Configuration space

degree of freedom

Kinematic / dynamic description

Motion Constraints

③ Properties of Algorithm

Optimal / nonoptimal motion

Computation & Memory Complexity

Completeness

Meaning that they will always find a solution to the motion planning problem when one exists or indicate failure in finite time.

Online / Offline

Sensor-based / world model

* Mathematical Style

⇒ Robots are assumed to operate in a plane (\mathbb{R}^2) or three dimensional (\mathbb{R}^3) ambient space, sometimes called the workspace W .

⇒ let WO_i be the i^{th} obstacle.

$$W_{\text{free}} = W \setminus \bigcup_i WO_i \quad \left\{ \begin{array}{l} \setminus \text{ is subtraction} \\ \text{operator} \end{array} \right.$$

⇒ Motion planning, however does not usually occur in the workspace. Instead it occurs in the Configuration Space Q (also called C-space).

$\Rightarrow R(q)$ is to denote the set of points of the ambient space occupied by the robot at Configuration q .

\Rightarrow An obstacle in the Configuration Space corresponds to Configurations of the robot that intersect an obstacle in the Workspace.

$$QO_i = \{q \mid R(q) \cap WO_i \neq \emptyset\}$$

$$Q_{free} = Q \setminus \bigcup_i QO_i$$

Path planning \rightarrow { Continuous Curve on the Configuration Space }

$C: [0, 1] \rightarrow Q$ where $C(0) = q_{start}$ $C(1) = q_{goal}$
and $C(s) \in Q_{free} \forall s \in [0, 1]$

\Rightarrow When path is parameterized by time t , then $C(t)$ is a trajectory.

\Rightarrow Finding a feasible trajectory is called trajectory planning or motion planning.

$$q = [q_1, q_2, \dots, q_n]^T$$

