16-843  – Manipulation Algorithms

Configuration Spaces

Katharina Muelling
Administrivia

• Schedule for student presentations is now on the course website – papers posted 2 weeks in advance

• First student presentations: Thursday, Sept 15 (not Tues Sept 13)

• Optional talk: Michael Koval PhD defense, Wednesday Sept 7, 1:00 p.m., NSH 3305
Problem Definition

Consider the problem of robot motion planning:

Goal

Robot:

neobotix-robot.com
Problem Definition

Consider the problem of robot motion planning:

- Should we model the position of the robot as a point?
- What about collision?

Robot: neobotix-robot.com
Consider the problem of robot motion planning:

How to avoid collisions of parts of the robot with the environment if modeled as a point?
Consider the problem of robot motion planning:

How to avoid collisions of parts of the robot with the environment if modeled as point?

Robot:

Expand obstacles
Reduce robot
Problem Definition

Consider the problem of robot motion planning:

How do we do this for those robots?

Cannot treat those robots as points!
Can we create a space in which all robots can be treated as points?

**Outline**

- Single point representation
- Algorithms

**Configuration Space**
- Robot Configuration Space
- Obstacle Representation
**Configuration Space**

**Definitions**

**Workspace:** The world in which the robot lives and occupies space. The set of reachable points by an end-effector.

3-parameter specification $(x, y, \theta)$

6-parameter specification $(x, y, z, \alpha, \beta, \gamma)$

rotation representation
**Configuration Space**

**Definitions**

**Configuration:**

The *configuration* of a robot is a specification of the position of every point of the system.

The configuration $\mathbf{q}$ is usually expressed as a vector of length $n$, where $n$ defines the degree of freedom of the robot.

$$\mathbf{q} = [q_1, q_2, \ldots, q_n]$$

**Configuration Space:**

The set $C$ of all possible configurations.

$$\mathbf{q} \in C$$

Read:
Spatial Planning: a Configuration Space
T. Lozano-Perez, 1980
Configura0on Space

Examples

\[ q = (x, y) \in \mathbb{R}^2 \]

\[ A(q) : C \rightarrow W \]

A(q) maps from point to the physical robot’s location in \( \mathbb{R}^2 \)
Examples

\[ \mathbf{q} = (x, y, \theta) \in SE(2) \]

\( A(\mathbf{q}) : C \rightarrow W \)

\( A(\mathbf{q}) \) maps from point to the physical robot’s location in \( \mathbb{R}^2 \)
**Examples**

\[ q = (\theta_1, \theta_2) \in S^1 \times S^1 \]

\[
A(q) : C \rightarrow W
\]

\( A(q) \) maps from point to the physical robot’s location in \( \mathbb{R}^2 \)
Configuration Space

Topography of this space is not necessarily the one of a Cartesian space.
Topology of this space is in not necessarily the one of a Cartesian space.

Animation
Topography of this space is in not necessarily the one of a Cartesian space.

Animation
Topology of this space is in not necessarily the one of a Cartesian space.

Animation
Topology of this space is in not necessarily the one of a Cartesian space.

Animation
Topology of this space is in not necessarily the one of a Cartesian space.

Animation
Configuration Space Obstacles

Where do we put the obstacle in configuration space?

World

Configuration Space

\[ q = (x, y) \in \mathbb{R}^2 \]
Configuration Space Obstacles

Obstacle representation in configuration space
- Obstacles make certain configurations in configuration space impossible
- Remove the configurations in configuration space that cause the robot to collide with obstacles or cause some specific links of the robot to collide

Obstacle Region:
\[ C_{obs} = \{ q \in C \mid A(q) \cap O \neq \emptyset \} \]

Free Space:
\[ C_{free} = C \setminus C_{obs} \]

Note: \( C_{free} \) is open set. Sometimes need to consider closure of \( C_{free} \).
Configuration Space Obstacles

Motion planning:
Find a path in $C$ such that no configuration along the path intersects with an obstacle

The Piano Mover’s Problem:
1) A World $W = \mathbb{R}^2$ or $W = \mathbb{R}^3$
2) An obstacle region $O \subset W$
3) A robot $A$ defined in $W$: $A \subset W$
4) $C_{\text{free}} = C \setminus C_{\text{obs}}$
5) A configuration $q_I \in C_{\text{free}}$ as initial configuration and $q_G \in C_{\text{free}}$ as goal configuration (query pair)
6) Compute continuous path $\tau : [0,1] \rightarrow C_{\text{free}}$ such that $\tau(0) = q_I$ and $\tau(1) = q_G$
Configuration Space Obstacles

Where do we put the obstacle in configuration space?

World

Configuration Space
\( \mathbf{q} = (x, y) \in \mathbb{R}^2 \)
Configuration Space Obstacles

Where do we put the obstacle in configuration space?

**World**

Configuration Space

\[ q = (x, y) \in \mathbb{R}^2 \]


Configuration Space Obstacles

**Minkowski Difference**
Formally we define the obstacle configuration space with the Minkowski difference, if
- \( C = \mathbb{R}^n, \ n = 1,2,3 \)
- Robot is rigid body

Minkowski Sum:

\[
A \oplus B = \{a + b \mid a \in A, b \in B\}
\]

Minkowski Difference:

\[
A \ominus B = \{a - b \mid a \in A, b \in B\}
\]
Minkowski Difference

$$A \oplus B = \{a + b \mid a \in A, b \in B\}$$

Configuration Space Obstacles
Configuration Space Obstacles

Minkowski Difference

\[ A \oplus B = \{ a + b \mid a \in A, b \in B \} \]

\[ A \ominus B = \{ a - b \mid a \in A, b \in B \} \]
Configuration Space Obstacles

Minkowski Difference

\[ A \oplus B = \{ a + b \mid a \in A, b \in B \} \]

\[ A \ominus B = \{ a - b \mid a \in A, b \in B \} \]
Configuration Space Obstacles

Minkowski Difference

\[ C_{obs} = O \oplus -A(0) \]

\[ = O \ominus A(0) \]

Object \( O \)  Robot \( A(0) \)  \( C_{obs} \)

Gets difficult in high dimensions.
**Star algorithm**

2D Workspace, translation

$C_{obs}$: convex polygon
Configuration Space Obstacles

**Star algorithm**
2D Workspace, translation
\( C_{obs} \): convex polygon

---

**Key observation:** every edge of \( C_{obs} \) is a translated edge from either \( A \) or \( O \)
**Star algorithm**
2D Workspace, translation
$C_{obs}$: convex polygon

**Problem:** Determine the order of these edges!

Let $\alpha_i$ denote the angles of the inward edge normals around $A$
Let $\beta_i$ denote the outward edge normals to $O$
Sort all angles in around $S^1$
Configuration Space Obstacles

**Star algorithm**
2D Workspace, translation
$C_{obs}$: convex polygon

**So far**: identified edges
**Left**: solid representation of $C_{obs}$

**Idea**: Create convex obstacles by intersecting half-planes

Two types of contacts:
- Vertex of $A$ and Edge of $O$
**Configuration Space Obstacles**

**Star algorithm**
2D Workspace, translation
\( C_{obs} \): convex polygon

**So far:** identified edges

**Left:** solid representation of \( C_{obs} \)

**Idea:** Create convex obstacles by intersecting half-planes

Two types of contacts:
- Vertex of \( A \) and Edge of \( O \)
- Edge of \( A \) and Vertex of \( O \)
Configuration Space Obstacles

**Star algorithm**
2D Workspace, translation
$C_{obs}$: convex polygon

**So far**: identified edges

**Left**: solid representation of $C_{obs}$

**Idea**: Create convex obstacles by intersecting half-planes

Contact occurs when $n$ and $v$ are perpendicular

$$n \cdot v = 0$$
**Configuration Space Obstacles**

**Star algorithm**
2D Workspace, translation
$C_{obs}$: convex polygon

**So far**: identified edges
**Left**: solid representation of $C_{obs}$

**Idea**: Create convex obstacles by intersecting half-planes

Contact occurs when $n$ and $v$ are perpendicular

\[ n \cdot v(x_t, y_t) = 0 \]

\[ H = \{(x_t, y_t) \in C \mid n \cdot v(x_t, y_t) \leq 0\} \]
Configuration Space Obstacles

**Star algorithm**
3D Workspace, translation
$C_{obs}$: polyhedral

generates

What happens when the robot translates *and* rotates in the plane?
What happens when the robot translates *and* rotates in the plane?

Need to compute $C_{obs}$ based on the third dimension $\theta$. 
What happens when the robot translates \textit{and} rotates in the plane?
What happens when the robot translates \textit{and} rotates in the plane?
What happens when the robot translates *and* rotates in the plane?

\[ \theta = 90 \]
What happens when the robot translates \textit{and} rotates in the plane?
Articulated Robots

Types of collisions:

• **Objects**: collision with objects make certain configuration in configuration space unattainable due to possible collision

• **Robot**: links of the robot collide with each other

\[
W = \mathbb{R}^3
\]

Consider each link independently: \(A_i(q)\)

Collision pairs: \((i, j) \in P\) \(\forall i \neq j\)

Self collision:

\[
\bigcup_{[i,j \in P]} \{q \in C \mid A_i(q) \cap A_j(q) \neq \emptyset\}
\]
Articulated Robots

Types of collisions:

• **Objects**: collision with objects make certain configuration in configuration space unattainable due to possible collision

• **Robot**: links of the robot collide with each other

\[ W = \mathbb{R}^2 \text{ or } W = \mathbb{R}^3 \]

\[ O \subset W \]

\[ \{ A_1, A_2, \ldots, A_n \} \]

Consider each link independently: \( A_i(q) \)

Link – Object collision:

\[
C_{obs} = \left( \bigcup_{i=1}^{m} \{ q \in C \mid A_i(q) \cap O \neq \emptyset \} \right) \cup \left( \bigcup_{[i,j] \in P} \{ q \in C \mid A_i(q) \cap A_j(q) \neq \emptyset \} \right)
\]
Articulated Robots

From Howie Choset

Configuration space

\[ \theta_1 \]

\[ \theta_2 \]