CONFIGURATION SPACE

miércoles, 25 de septiembre de 2019 9:59 a. m.

Describing the robot as a point in the configuration space is useful for:

- Planning
- Control

Concept	Definition

Robot	• A collection of n_b bodies moving in an environment (workspace):
	  • R^2 or R^3
	• Number of bodies:
	  • If $n_b = 1$ **Single**-body robot
	  • If $n_b > 1$ **Multi**-body robot
	• Environment:
	  • In R^2: **Planar** Robots
	  • In R^3: **Spatial** Robots
	• **Configuration**: Minimal set of parameters that describes the position of all points of the robot. Usually organized in a vector q of parameters that are the generalized coordinates.
	  • $q = \begin{pmatrix} q_1 \\ q_2 \\ \dots \\ q_n \end{pmatrix}$
	  • q_i i-th **generalized coordinate**.

Generalized Coordinates	• **Cartesian**: used to identify the **position** of certain points of the robots (i.e. center of mass).
	  • R^2
	  • R^3
	• **Angular**: used to identify the **orientation** of bodies.
	  • Euler angles
	  • Quaternions
	  • $SO(N)$: Special Ortonormal Group of Space N
	    ○ $SO(N) = N\frac{N-1}{2}$
	  • $SE(N)$ = R^N × $SO(N)$: Special Eucledian Group of Space N

Configuration Space	• The set of all possible configurations of a robot
	• Dimension: Number of generalized coordinates, i.e. dimension of q
	• Geometry: It is the **topology** of the configuration space.
	  • **Manifold**: A space in which any neighborhood of a point is **diffeomorphic** to a neighborhood of R^n. A diffeomorphism is a continuous bijection between the space and the euclidean space R^n whose inverse is also continuous.
	  • Distances in C:
	    ○ Axioms for $d(q_A, q_B)$:
	      ▪ $d(q_A, q_B) \geq 0 \; \forall q_A, q_B$
	      ▪ $d(q_A, q_B) = 0 \Leftrightarrow q_A = q_B$
	      ▪ $d(q_A, q_B) = d(q_B, q_A)$
	      ▪ $d(q_A, q_B) + d(q_B, q_C) \geq d(q_A, q_C)$
	    ○ Euclidean Distance:
	      ▪ $d(q_A, q_B) = \lvert\lvert q_A - q_B \rvert\rvert$

- Manifold Distance:
  - **Geodesic**: Minimum length path between two points in a space. In general, geodesics cannot be computed analitically.
  - Practical:
    - B the robot
    - $B(q)$ the volume (region of W) occupied by the robot when the configuration is q
    - $P(q)$ the position of point P when the configuration is q.
    - $d(q_A, q_B) = \max_{P \in B} ||P(q_A) - P(q_B)||$
    - This means that the distance is the maximum value of the displacement of any point in the robot from one configuration to another.
    - To compute we need to calculate all possible distances and take the maximum so computability may be a problem. To simplify we compute the distances for a predefined set of points that are called **control points**.
    - $d(q_A, q_B) = \max_{P \in \{P_1, \dots, P_N\}} ||P(q_A) - P(q_B)||$
    - The problem now is choosing the right control points.

- Obstacles: The general idea is to map the obstacles from the workspace to the configuration space.
  - O_i: the i-th obstacle in W (closed subsets of W)
  - CO_i: configuration-space obstacle
    - $CO_i = \{q \in C : B(q) \cap O_i \neq \emptyset\}$
    - Set of all configurations that cause a collision between the robot and O_i
  - $CO = \cup_{i=1}^{M} CO_i$
  - $C_{free} = C - CO$ is then the free configuration space and all paths should be contained in this space. **Motion planning and control** takes place in this space.
  - Point Robot

- **Examples**:
  - Point Robot in R^2
    - $q = \begin{pmatrix} x \\ y \end{pmatrix}$
    - $C = R^2$ (same as workspace)
  - Point Robot in R^3
    - $q = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$
    - $C = R^3$ (same as workspace)
    - Obstacles: Exactly a copy of the obstacles in the workspace because the configuration space and the workspace are equivalent.
  - Disk Robot in R^2
    - $q = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix}$
      - x and y are the coordinates of the center of the disk.
      - The angle θ is only used if the orientation is important.
    - $C = R^2$ or $R^2 \times SO(2)$ (same as workspace if θ is irrelevant)
    - Obstacles: the C-obstacle in this case is larger than the original obstacle because the robot has a non-zero size so the robot must stay at a distance equal to its radius from the original shadow of the obstacle.
  - Poligonal Robot in R^2

- $q = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix}$
    - x and y are the coordinates of the center of an important point such as the baricenter of the poligon.
    - θ is a prefered orientation of the poligon.
- $C = R^2 \times SO(2)$ or $SE(2)$
- Obstacles:
    - Without Rotation: then the configuration space is still a copy of the workspace. The C-obstacle is then the shadow of the original workspace obstacle plus the area that the robot cannot reach because of its shape and size. The shape of the C-obstacle also depends on the choice of representative point.
    - With Rotation: The configuration space is no longer euclidean nor a copy of the workspace, it is now a manifold. Since the configuration space is now 3D, the C-obstacle now occupies a volume in that space that is made up by all of the 2D C-obstacles for all possible orientations of the robot.

- Polihedral Robot in R^3
    - $q = \begin{pmatrix} x \\ y \\ z \\ \alpha \\ \beta \\ \gamma \end{pmatrix}$
    - $C = R^3 \times SO(3)$
- N-link Planar Manipulator
    - $q = \begin{pmatrix} \theta_1 \\ \theta_2 \\ \dots \\ \theta_n \end{pmatrix}$
    - $C = SO(2)^n$
        - It is different form $SO(N)$ because it represent a sequence of N single body rotations instead of a single body orientation in an N dimensional space
- N-link Spatial Manipulator
    - $q = \begin{pmatrix} \theta_1 \\ \theta_2 \\ \dots \\ \theta_n \end{pmatrix}$
    - $C = SO(3)^N$
- 2R Planar Manipulator:
    - $q = \begin{pmatrix} q_1 \\ q_2 \end{pmatrix}$ where q_i represent the joint angles.
    - $C = SO(2) \times SO(2)$
    - Topology: Toroidal

Autonomous and Mobile Robotics

Prof. Giuseppe Oriolo

Configuration Space

companion slides for the blackboard lecture

Dipartimento di Ingegneria Informatica Automatica e Gestionale Antonio Ruberti
Sapienza Università di Roma

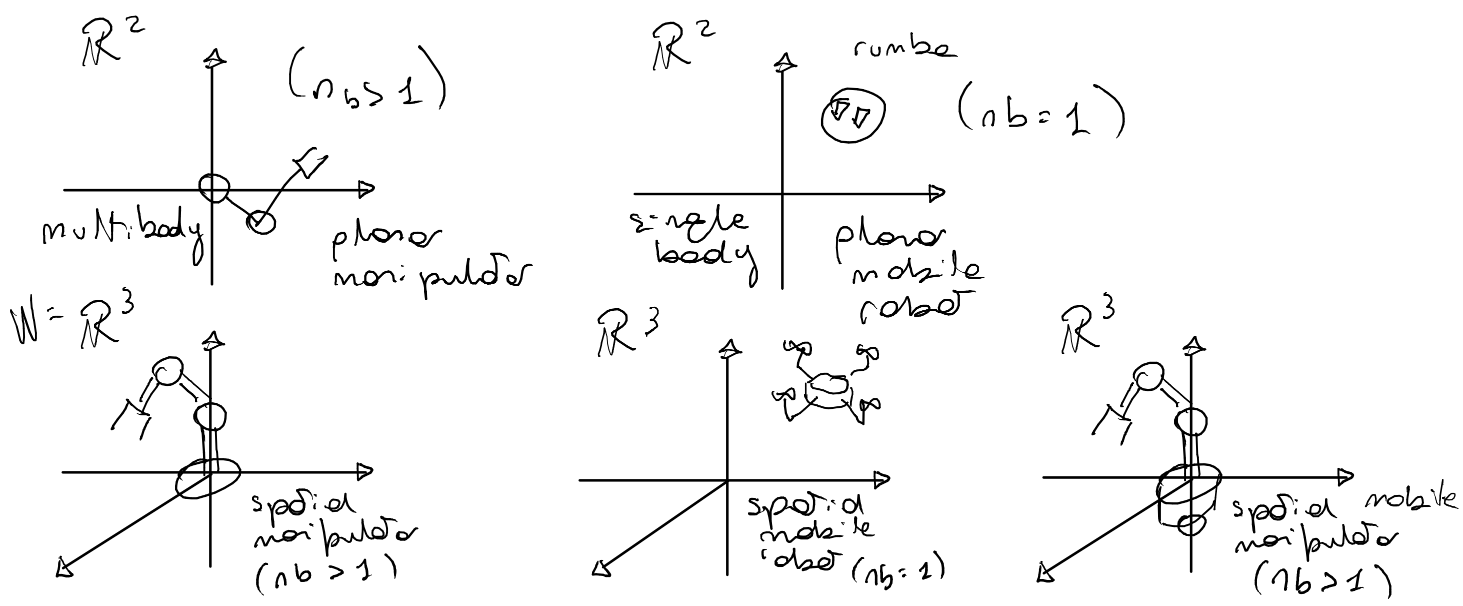
Configuration space

- idea: represent the robot as a point in a suitable space
- useful tool:
  - planning
  - control

We will discuss also:
- dimension
- topology
- distance
- obstacles

Definitions

- robot: a system of nb rigid bodies (collection), moving in a workspace $\mathbb{R}^N$



• Configuration of a ROBOT

A minimal set of parameters that allows to identify the position of each point of B in W

$$q = \begin{pmatrix} q_1 \\ \vdots \\ q_n \end{pmatrix}$$ generalized coordinates

configuration (vector)

• Generalized coordinates :

- Cartesian: used to define the (Cartesian) position of some points of B

  → take values in $\mathbb{R}^N$, $N = 2$ or 3

- Angular: used to define the orientation of some bodies of B

  → take values in $SO(2)$ or $SO(3)$
  (special orthogonal in 2 dim or 3 dim)
  2 dim: planar rotations; 3 dim: spatial rotations

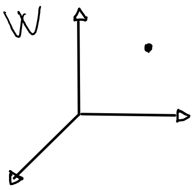
• Configuration space (C-space) C

the set of all configuration that the robot can assume

dimension of $C = n$

examples

• a point robot in $\mathbb{R}^3$

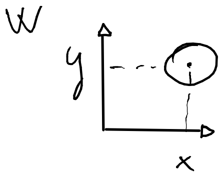


$q = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$

$n = 3$

$C = \mathbb{R}^3$

• A disk robot in $\mathbb{R}^2$ (roomba)

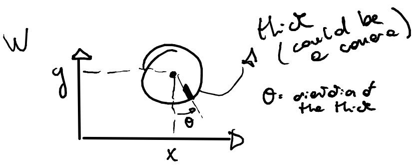


$q = \begin{pmatrix} x \\ y \end{pmatrix}$

$n = 2$

$C = \mathbb{R}^2$

• a disk robot in $\mathbb{R}^2$ with a thick

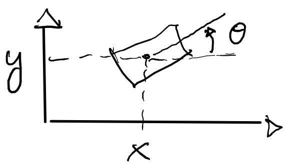


$q = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix}$ $n = 3$

$C = \mathbb{R}^2 \times SO(2)$

planar rotation θ in $SO(2)$

• A polygonal robot in $\mathbb{R}^2$ (car)



$q = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix}$ $n = 3$

$C = \mathbb{R}^2 \times SO(2)$ → $SE(2)$ special euclidean space (space of rototranslation)

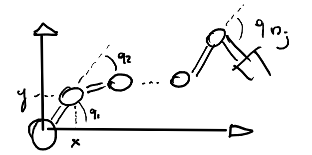
- A polynomial robot in $\mathbb{R}^3$

$$q = \begin{pmatrix} x \\ y \\ z \\ \alpha \\ \beta \\ \gamma \end{pmatrix} \begin{matrix} \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} \text{cartesian coordinates} \\ \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} \text{rotation coordinates} \end{matrix}$$

$n = 6$

$$C = \underbrace{\mathbb{R}^3 \times SO(3)}_{SE(3)}$$

- A planar manipulator with n_j revolute joints (or nR robot)



by taking:

$\begin{cases} \text{center of each body} \\ \text{orientation " "} \end{cases} \Rightarrow 3 \cdot n_j$ coords $\rightarrow$ Is this a minimal set? NO!

Each joint has 2 constraints : $3n_j - 2n_j = n_j$

$$q = \begin{pmatrix} q_1 \\ \vdots \\ q_n \end{pmatrix} \rightarrow C = SO(2) \times SO(2) \times \dots \times SO(2) \quad n_j \text{ times}$$
$$= (SO(2))^{n_j}$$

- A spatial manipulator with n_j revolute joints

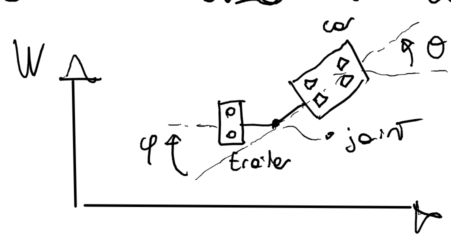
$6n_j$: by taking $\begin{cases} \text{center for each body} \\ \text{orientation for each node} \end{cases}$

$5n_j$: constraints

$6n_j - 5n_j = n_j$

$$\Rightarrow C = (SO(2))^{n_j}$$

- a car-like robot with a trailer in $\mathbb{R}^2$



6 = position & orientation trailer + position & orientation car

2 = constraints from joint

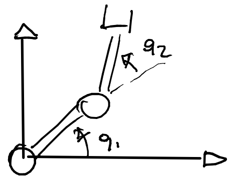
$6 - 2 = 4$

$$q = \begin{pmatrix} x \\ y \\ \theta \\ \varphi \end{pmatrix} \qquad C = \mathbb{R}^2 \times SO(2) \times SO(2)$$

• Topology of C_1

What kind of space is C_1?

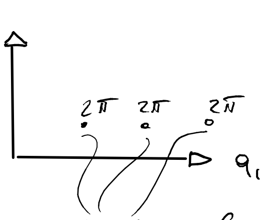
example: a 2R manipulator in $\mathbb{R}^2$



$$q = \begin{pmatrix} q_1 \\ q_2 \end{pmatrix}$$

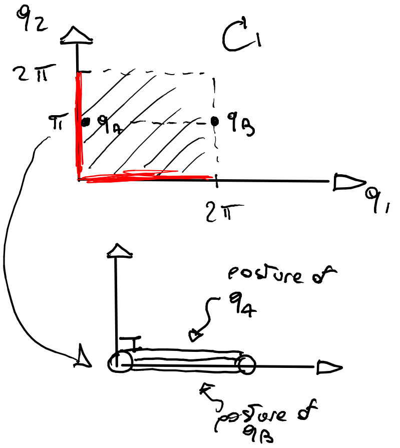
$$C_1 = SO(2) \times SO(2)$$

loss of injectivity



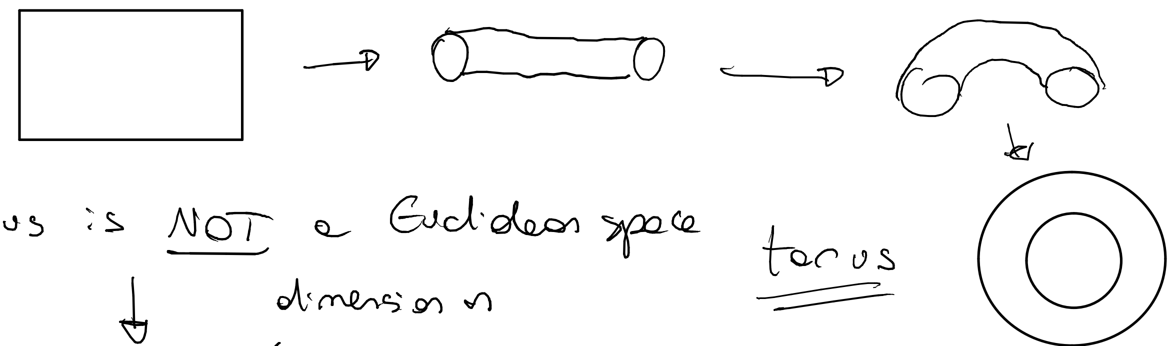
multiple configs correspond to the same posture of the robot (not injective)

Is this a good representation of C_1? NO



"for" in this representation of C_1 they correspond to almost the same posture of robot!

• Correct representation of C_1



A torus is NOT a Euclidean space dimension n

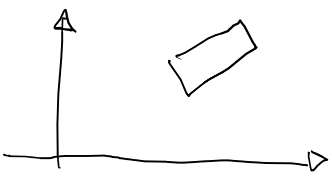
torus

need an embedding in $\mathbb{R}^3$ for visualization

A manifold (varietà)

Space where each neighborhood of a point is homeomorphic ("looks like") to a neighborhood of $\mathbb{R}^n$

• polygonal robot in $\mathbb{R}^2$ → in $\mathbb{R}^3$

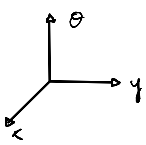


$$q = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix}$$

$$C_1 = \mathbb{R}^2 \times SO(2)$$

manifold → Every time we have orientation in the coordinates we have manifold

In $\mathbb{R}^3$, representing C_1, embedding does not help visualization



Distance

I need:
- $d(q_A, q_B) \geq 0$
- $d(q_A, q_B) = 0$ iff $q_A = q_B$
- $d(q_A, q_B) = d(q_B, q_A)$
- $d(q_A, q_B) + d(q_B, q_C) \leq d(q_A, q_C)$

<u>problem</u>: C_i is not a Euclidean space
⇒ cannot use Euclidean distance

Therefore we have to compute <u>distances on manifolds</u>
→ Use Geodesics (path of shorter length between two points)
↳ only known for simple manifolds

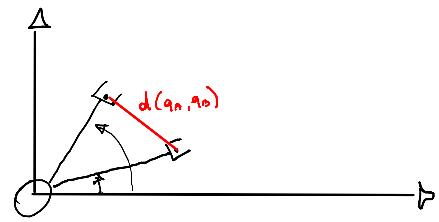
in <u>Robotics</u>

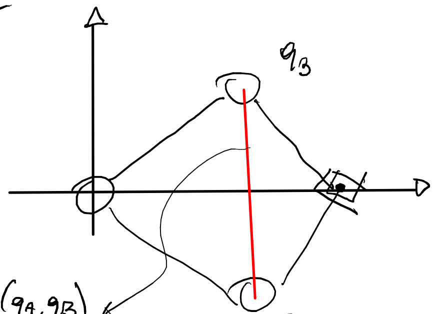
$B(q)$ region of W occupied by the robot when the configuration is q
$p(q)$ position of point p (of the robot) in W when the configuration is q

distance:

$$d(q_A, q_B) = \max_{p \in B} \underbrace{\| p(q_A) - p(q_B) \|}_{\text{on Euclidean distance}} \longrightarrow \text{DISPLACEMENT METRIC}$$

single joint



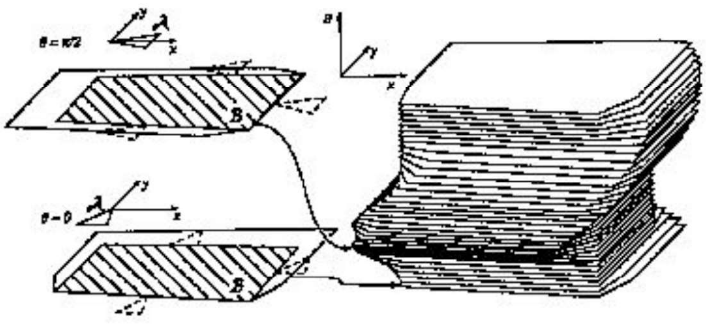


2 joints

$d(q_A, q_B)$ (maximum displacement between 2 configs → on the 2nd joint)

C-obstacles when rotations are involved

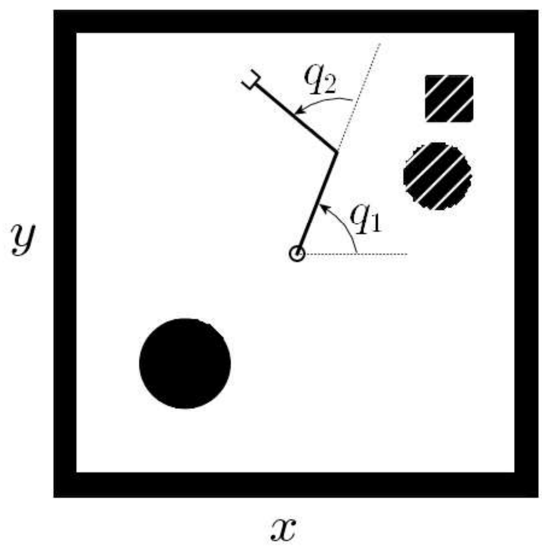
for a polygonal robot free to translate and rotate on the plane

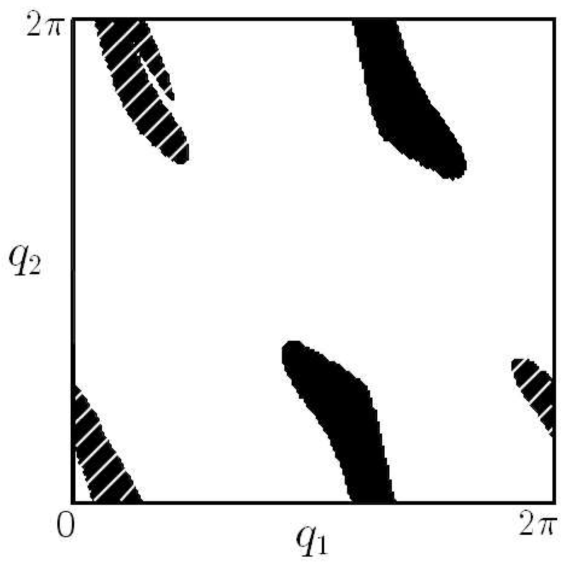


"grow and stack"

C-obstacles when rotations are involved

for a 2R planar manipulator, scene 1

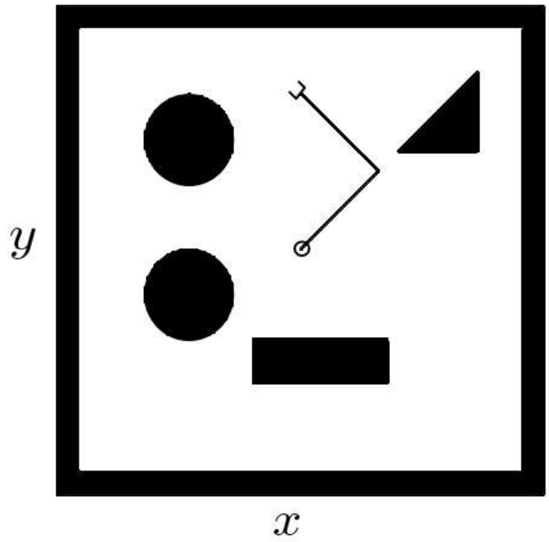


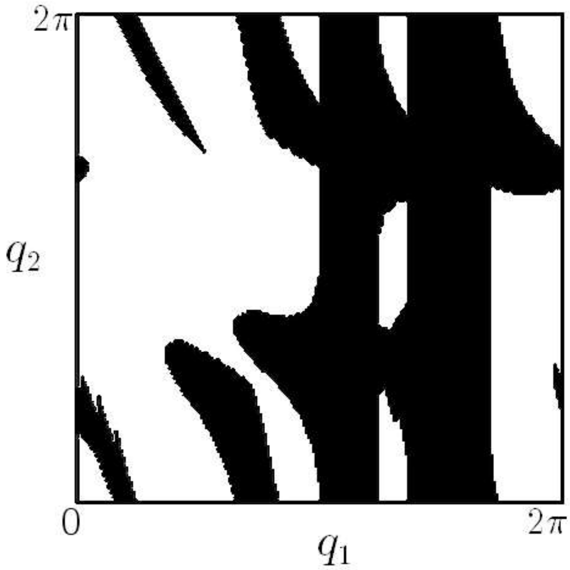


disjoint workspace obstacles may merge in C-space

C-obstacles when rotations are involved

for a 2R planar manipulator, scene 2





the free configuration space may be disconnected

Problems

Describe the nature (including the dimension) of the configuration space for a mobile manipulator consisting of a unicycle-like vehicle carrying a sixDOF anthropomorphic arm, providing a choice of generalized coordinates for the system.

The configuration of the mobile manipulator is

$$q = [\underbrace{x \; y}_{} \; \theta_0 \; \theta_1 \; \theta_2 \; \theta_3 \; \theta_4 \; \theta_5 \; \theta_6]^T$$

x, y: coordinates of the contact point of the wheel with the ground (equivalently of the wheel centre)

θ_0: unicycle orientation (wrt x axis)

$\theta_1 \dots \theta_6$: manipulator joint variables

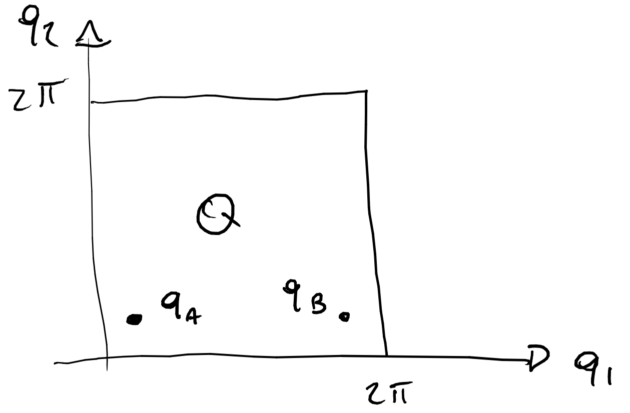
- Configuration space

$$\mathcal{C} = \mathbb{R}^2 \times \underbrace{SO(2) \times \dots \times SO(2)}_{7 \text{ times}} \qquad \dim(\mathcal{C}) = 9$$

With reference to a 2R manipulator, modify the definition (12.2) of configuration space distance so as to take into account the fact that the manipulator posture does not change if the joint variables q1 and q2 are increased (or decreased) by a multiple of 2π.

* definition 12.2 : Euclidean norm $d_2(q_A, q_B) = \| q_A - q_B \|$

Assume that the configuration q takes values in the subset Q



This can be obtained by computing the joint variables q_1 and q_2.

Given two configurations $q_A = (q_{1A}, q_{2A})$, $q_B = (q_{1B}, q_{2B})$ in Q, define

$$\Delta_1 = \min\left(|q_{1,A} - q_{1,B}|,\ 2\pi - |q_{1,A} - q_{1,B}|\right)$$

$$\Delta_2 = \min\left(|q_{2,A} - q_{2,B}|,\ 2\pi - |q_{2,A} - q_{2,B}|\right)$$

and let $d_3(q_A, q_B) = \sqrt{\Delta_1^2 + \Delta_2^2}$

This definition of configuration space distance clearly satisfies the requirement of the problem.