Norwegian University of Science and Technology Department of Mathematical Sciences	TMA4180 Optimisation I Spring 2017 **Exercise set "Project I"**

1 Robots

We consider a planar robot consisting of n rigid segments of lengths ℓ_i, $1 \leq i \leq n$, that are connected by revolute joints. The hand or tool is attached, by means of a freely orientable joint, to the last segment, while the first segment is attached to the origin of the plane $\mathbb{R}^2$, again by means of a revolute joint. A sketch of such a robot with $n = 3$ segments is given in Figure 1.

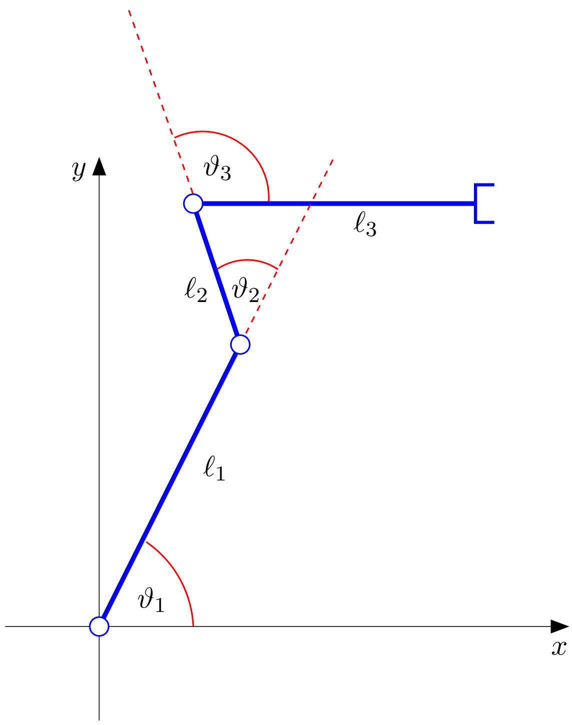

Figure 1: Sketch of the type of planar robots we consider in this project. The segments of the robot are of (fixed) lengths ℓ_i, $i = 1, 2, 3$. The joint angle ϑ_1 denotes the angle between the first segment and the x-axis. The subsequent joint angles ϑ_i denote the angle between the i^{th} and the $(i-1)^{\text{st}}$ segment. Note that, in this particular configuration, the angle ϑ_3 is negative, whereas the other two angles are positive.

The set of all points that are reachable by the hand is called the *configuration space* of the robot, and will be denoted by $\mathcal{C}$. The set of all possible joint parameters is denoted by $\mathcal{J}$ and called the *joint space* of the robot. In the situation we consider here, we can describe the robot completely by the set of the angles ϑ_i, $i = 2, \ldots, n$, between the i^{th} and $(i-1)^{\text{st}}$

segment of the robot, as well as the angle ϑ_1 between the x-axis and the first segment (again, see Figure 1). In that way, we can view the joint space $\mathcal{J}$ as a subset of $\mathbb{R}^n$.

In case the joints may move freely without any restrictions, the joint space is equal to the whole $\mathbb{R}^n$. More usual in practice, however, is a situation where each joint may only vary between a minimal and a maximal angle, say $\vartheta_i^{\min}$ and $\vartheta_i^{\max}$. In such a case, the joint space would be a box of the form

$$\mathcal{J} = [\vartheta_1^{\min}, \vartheta_1^{\max}] \times \cdots \times [\vartheta_n^{\min}, \vartheta_n^{\max}] \subset \mathbb{R}^n.$$

2 Forward and Inverse Kinematics

A first very basic question in robotics is the problem of determining the configuration of the robot given the joint parameters. More formally, this means that one wants to find a function $F\colon \mathcal{J} \to \mathcal{C}$ mapping the joint parameters to the position of the end-effector. This is called the *forward kinematic problem*. For the type of robots and the parametrisation of the joints we consider here, convince yourself that F takes the form

$$F(\vartheta) = \sum_{i=1}^{n} \ell_i \begin{pmatrix} \cos \sum_{j=1}^{i} \vartheta_j \\ \sin \sum_{j=1}^{i} \vartheta_j \end{pmatrix},$$

where $\vartheta = (\vartheta_1, \ldots, \vartheta_n)$ is the vector of joint angles. More general, the position of the k^{th} joint is given by

$$\sum_{i=1}^{k} \ell_i \begin{pmatrix} \cos \sum_{j=1}^{i} \vartheta_j \\ \sin \sum_{j=1}^{i} \vartheta_j \end{pmatrix}.$$

In contrast, in the *inverse kinematic problem* a target configuration $p \in \mathbb{R}^2$ is given, and one wants to find joint parameters $\vartheta \in \mathcal{J}$ that generate this configuration. In other words, one wants to solve the equation

$$F(\vartheta) = p \tag{1}$$

for ϑ. When trying to solve this equation, however, one faces (at least) two difficulties. First, a solution of this equation only exists, if the point p is actually an element of the configuration space $\mathcal{C}$ of the robot, and, depending on possible restrictions on the joints of the robot, this configuration space may be difficult or even impossible to describe analytically. Second, even if a solution exists, it cannot be expected to be unique, as one has two equations (one for each coordinate of p) but n unknowns (one for each joint), and n is usually larger than two.

As an alternative to solving the equation $F(\vartheta) = p$, it may therefore make sense to solve instead the optimisation problem

$$\min_{\vartheta \in \mathcal{J}} \frac{1}{2} \|F(\vartheta) - p\|_2^2. \tag{2}$$

That is, instead of finding joint parameters that satisfy (1) exactly, one only tries to approximate the point p as close as possible within the constraints given by the robot.

3 Tasks

The goal of this project is the numerical solution of the optimisation problem (2). For simplicity we will assume in the following that we don't have any restrictions on the joints and thus $\mathcal{J} = \mathbb{R}^n$.[1]

3.1 Theoretical Investigations

1 Provide a mathematical description of the configuration space $\mathcal{C}$ in case $\mathcal{J} = \mathbb{R}^n$.

2 Show that the problem (2) has a solution.

3 Discuss the properties of the optimisation problem (2).

The discussion of the properties might, for instance, address the following questions: Is it possible to say something about uniqueness of solutions? Are there local minima? Stationary points that are not local minima? Are the minimisers isolated? Is the problem convex?

Note also that the properties of the problem might depend both on the specifications of the robot and the point p. In your study of the optimisation problem, you should therefore discuss whether these properties are generic to the problem or they are specific to certain situations.

3.2 Implementation

4 Implement a numerical method that, given segment lengths ℓ_i, $1 \leq i \leq n$, and a target configuration $p \in \mathbb{R}^2$, computes a solution of the optimisation problem (2) with $\mathcal{J} = \mathbb{R}^n$.

You may implement any method you like for the solution of this unconstrained optimisation problem, but provide some justification for your choice in the report, ideally based on the previous theoretical discussion.

Also, while not necessary for solving the inverse kinematic problem (2), it is strongly recommended to implement first a function solving *and plotting* the forward problem. That is, implement a function that, given segment lengths ℓ_i, $1 \leq i \leq n$, and joint angles ϑ_i, $1 \leq i \leq n$, calculates and plots the position of the robot's joints in the plane. This should be useful both for presenting your results in the report, but also for debugging purposes.

[1] In the second project, where we will continue our investigations of planar robots, this restriction will be dropped at some point, though.

5 Test your numerical method in a number of different situations and discuss the results.

As a suggestion, you might test your method for the following settings:

- $n = 3$, $\ell = (3, 2, 2)$, $p = (3, 2)$.
- $n = 3$, $\ell = (1, 4, 1)$, $p = (1, 1)$.
- $n = 4$, $\ell = (3, 2, 1, 1)$, $p = (3, 2)$.
- $n = 4$, $\ell = (3, 2, 1, 1)$, $p = (0, 0)$.

(The first and the third test case are completely generic and should not pose any difficulties. In the second and fourth test case, your program might run into problems, though, depending on your implementation and possible handling of exceptional cases.)

In the discussion of the results, you should at least address the questions of convergence (or: failure of convergence in specific situations) and speed of convergence. Also, try to relate your numerical results with what the theory would predict, whenever this is possible.