Interactive Robotics Education Tool

Purpose

Our goal is to create a demonstration tool that supports robotics education by visualizing different aspects of robotics algorithms and concepts, e.g., robot kinematics, motion planning, and controls. Target audiences include university students, advanced high school students, and community members interested in learning about how robots work.

Deliverables

We would like the team to create a client-based web application that shows a robot in a 2D environment, lets the user make changes to this environment and visualizes the steps and functionality of different algorithms. The building blocks of the robot environment and the user interface are the same for many different robotics topics and algorithms, and we would like the team to create these building blocks in such a way that they can be reused for creating multiple demonstration tools.
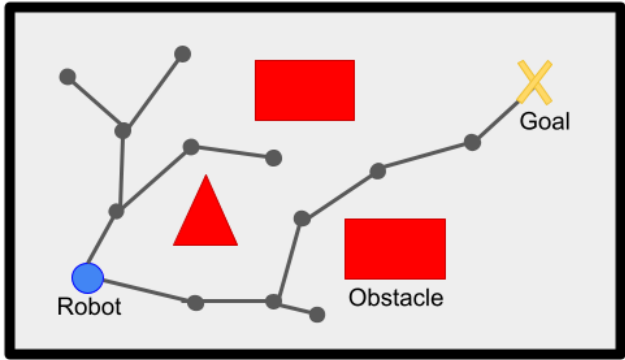
A typical 2D robot scenario consists of a rectangular area that contains the robot, several obstacles, and a goal. The robot's movement over time as well as the path it is following should be visualized. There should be UI elements for starting and stopping the execution of an algorithm and UI elements for setting algorithm parameters. Users should be able to explore how changes they make to the environment and the algorithm parameters affect the performance of the algorithm. The drawing below shows a sketch for a path planning demonstration tool consisting of the described elements. We already use a demonstration tool for a single topic, which can be found here as an example: https://robotics.cs.unc.edu/C-space/ .

The code will be made publicly available via a BSD-style license

Prerequisite Knowledge

No prior robotics knowledge is required. An undergraduate data structures class covering graphs (e.g., COMP 210) provides all necessary background knowledge, and code for the robotics algorithms will be provided. If the team is interested in deeper understanding of a particular algorithm to enable a better performing web application, we would be happy to provide a short tutorial.

Robot Path Planning Application Layout

Robot Environment

✏️	Draw new Obstacle
🗑️	Remove Obstacle

▶️ Start   ⏸️ Pause   ⏹️ Stop   ❌ Clear Environment

Parameter 1: 3.5

Parameter 2: 10.8