

CPS: Praktikum 5

Probabilistic Roadmap und A*-Pfadsuche

Version: 2025-12-28



This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Germany License. To view a copy, visit <http://creativecommons.org/licenses/by-nc-sa/3.0/de/> or write to Creative Commons, 171 Second Street, Suite 300, San Francisco, CA 94105, USA.

Der vorbereitete Code für dieses Praktikum ist ein mittels Vibe-Coding entstandenes, aus zwölf Python-Dateien bestehendes Programm. Es simuliert einen Roboterarm, der ungewöhnlicherweise ausschließlich Endlosgelenke hat (Schleifringe o. ä. statt Kabel),^{c1} in einem Raum mit Hindernissen. Der Roboterarm besteht aus drehbaren Quadern; die Hindernisse sind achsenparallele Quader. Das Programm hat zwei Betriebsarten: In der ersten Betriebsart baut es eine Probabilistic Roadmap auf (Konfigurationen würfeln, auf Kollisionsfreiheit testen, nächste Nachbarn verbinden, die Verbindungen auf Kollisionsfreiheit testen, daraus einen Graph bauen) und speichert diese als JSON-Datei. (Beim vorbereiteten Code ist bereits eine solche Datei für 10.000 Knoten enthalten.) In seiner zweiten Betriebsart lädt das Programm diese JSON-Datei und zeigt eine interaktive 3D-Darstellung, in der man Endpunkte des Roboterarms auswählen kann, die dieser dann anfährt – falls es einen Pfad im Graphen gibt. Der kürzeste Pfad wird mit dem A*-Algorithmus gefunden.

c1 text added by jl

Aufgaben **ohne KI**:

- In welchem (abstrakten) Raum liegen die Knoten der Probabilistic Roadmap?
- Wie werden Abstände in diesem Raum berechnet?
- Was ist die Struktur der JSON-Datei?
- Wozu und wie werden im interaktiven Teil Events benutzt?
- Überprüfen Sie, ob bei der Animation der kürzeste Weg von einem Winkel zum nächsten verwendet wird.^{c2}
- Wodurch benötigt der Aufbau der Probabilistic Roadmap so viel Zeit? Wie könnte man ihn beschleunigen? (Setzen Sie das dann mit KI-Hilfe um.)
- Wie könnte man die Abstandsberechnung ändern, um weniger Zickzackbewegungen zu erhalten? (Setzen Sie das dann mit KI-Hilfe um.)
- Wie könnte man organisch weiche Bewegungen erzeugen? (Setzen Sie das dann mit KI-Hilfe um.)
- Wie könnte man eine nachträgliche Feinjustage vornehmen, um auch andere als die gewürfelten Endpunkte zu erreichen? (Setzen Sie das dann mit KI-Hilfe um.)

c2 text added by jl