



hochschule mannheim

**test First line  
second line title**

Daniel Koch

Bachelor-Thesis  
Studiengang Informatik

Fakultät für Informatik  
Hochschule Mannheim

22.07.2020

Betreuer

Prof. Dr. Thomas Ihme, Hochschule Mannheim

**Koch, Daniel:**

TEST / Daniel Koch. –

Bachelor-Thesis, Mannheim: Hochschule Mannheim, 2020. 13 Seiten.

**Koch, Daniel:**

TEST / Daniel Koch. –

Bachelor Thesis, Mannheim: University of Applied Sciences Mannheim, 2020. 13 pages.

## **Erklärung**

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Ich bin damit einverstanden, dass meine Arbeit veröffentlicht wird, d. h. dass die Arbeit elektronisch gespeichert, in andere Formate konvertiert, auf den Servern der Hochschule Mannheim öffentlich zugänglich gemacht und über das Internet verbreitet werden darf.

Mannheim, 22.07.2020

Daniel Koch



# Abstract

*TEST*

TEST.

*TEST*

TEST.



# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Ziel der Arbeit . . . . .	1
1.3	Aufbau der Arbeit . . . . .	1
<b>2</b>	<b>Grundlagen</b>	<b>3</b>
2.1	Aufbau des Laufroboters . . . . .	3
2.2	Robotik . . . . .	3
2.2.1	Koordinatensysteme . . . . .	3
2.2.2	Direkte Kinematik . . . . .	3
2.2.3	Inverse Kinematik . . . . .	3
2.2.4	Laufplanung . . . . .	3
2.3	Frameworks . . . . .	3
2.3.1	Robot Operating System . . . . .	3
2.3.2	Gazebo . . . . .	3
2.3.3	MeshLab . . . . .	3
<b>3</b>	<b>Funktionsweise des Random Samplings</b>	<b>5</b>
3.1	Erzeugen gültiger Lösungen . . . . .	5
3.2	Bewertung gültiger Lösungen . . . . .	5
3.3	Heuristiken zur Bewertung des Algorithmus . . . . .	5
<b>4</b>	<b>Portierung des Laufplaners nach ROS und Gazebo</b>	<b>7</b>
4.1	Analyse bestehender Laufplaner . . . . .	7
4.2	Allgemeiner Aufbau des Pakets . . . . .	7
4.3	Aufsetzen der Simulation . . . . .	7
4.3.1	Aufsetzen des Roboter-Modells mittels urdf . . . . .	7
4.3.2	Definition der Gelenkmotoren mittels ros_control . . . . .	7
4.3.3	Aufsetzen der Umgebung mittels Gazebo . . . . .	8
4.3.4	Aufsetzen der Fußsteuerung des Laufroboters . . . . .	8
4.4	Aufsetzen von Laufalgorithmen . . . . .	8
4.4.1	Implementierung . . . . .	8
4.4.2	Generierung von Bewegungen als xml-Datei . . . . .	8
4.4.3	Einlesen und Abspielen der Bewegungen . . . . .	8

<b>5 Testen der Ergebnisse</b>	<b>9</b>
<b>6 Zusammenfassung</b>	<b>11</b>
<b>7 Ausblick</b>	<b>13</b>
<b>Abkürzungsverzeichnis</b>	<b>vii</b>
<b>Tabellenverzeichnis</b>	<b>ix</b>
<b>Abbildungsverzeichnis</b>	<b>xi</b>
<b>Quellcodeverzeichnis</b>	<b>xiii</b>



# **Kapitel 1**

## **Einleitung**

### **1.1 Motivation**

### **1.2 Ziel der Arbeit**

### **1.3 Aufbau der Arbeit**



## **Kapitel 2**

# **Grundlagen**

### **2.1 Aufbau des Laufroboters**

### **2.2 Robotik**

#### **2.2.1 Koordinatensysteme**

#### **2.2.2 Direkte Kinematik**

#### **2.2.3 Inverse Kinematik**

#### **2.2.4 Laufplanung**

statische Laufalgorithmen, reaktive, planende Laufalgorithmen like RandomSampling

### **2.3 Frameworks**

#### **2.3.1 Robot Operating System**

#### **2.3.2 Gazebo**

#### **2.3.3 MeshLab**



## **Kapitel 3**

# **Funktionsweise des Random Samplings**

### **3.1 Erzeugen gültiger Lösungen**

### **3.2 Bewertung gültiger Lösungen**

### **3.3 Heuristiken zur Bewertung des Algorithmus**



## **Kapitel 4**

# **Portierung des Laufplaners nach ROS und Gazebo**

### **4.1 Analyse bestehender Laufplaner**

Hermes, Ruffler, und AKrobat GitHub (gibt es dazu eine Arbeit?)

### **4.2 Allgemeiner Aufbau des Pakets**

Ordnerstruktur oder vllt in Baustein, Laufzeit und Verteilungssicht?

### **4.3 Aufsetzen der Simulation**

#### **4.3.1 Aufsetzen des Roboter-Modells mittels urdf**

Notes: urdf, xacro, Collisions, Inertia + Berechnung + STL-Dateien ( Vereinfachung durch einfaches Geometry Object wenn möglich, sonst vereinfachtes STL, sonst das Original STL) / MeshLab

#### **4.3.2 Definition der Gelenkmotoren mittels ros\_control**

Notes: URDF-File, config file, Controller

### **4.3.3 Aufsetzen der Umgebung mittels Gazebo**

launch-files

### **4.3.4 Aufsetzen der Fußsteuerung des Laufroboters**

## **4.4 Aufsetzen von Laufalgorithmen**

### **4.4.1 Implementierung**

### **4.4.2 Generierung von Bewegungen als xml-Datei**

### **4.4.3 Einlesen und Abspielen der Bewegungen**



## **Kapitel 5**

### **Testen der Ergebnisse**



## **Kapitel 6**

### **Zusammenfassung**



## **Kapitel 7**

## **Ausblick**



# **Abkürzungsverzeichnis**





# **Tabellenverzeichnis**



# **Abbildungsverzeichnis**



## Listings





