TECHNISCHE UNIVERSITÄT WIEN

AUTOMATION & CONTROL INSTITUTE
INSTITUT FÜR AUTOMATISIERUNGS- & REGELUNGSTECHNIK

# Modeling and Workspace of a Mobile Manipulator

## MASTER'S THESIS

conducted in partial fulfillment of the requirements for the degree of a

Master of Science (MSc)

supervised by

Univ.-Prof.Dr. techn.A.Kugi
DI. Florian Beck

submitted at the

TU Wien
Faculty of Electrical Engineering and Information Technology
E376 Automation and Control Institute

by
Lilla Breitkopf
Vienna
Austria

Vienna, May 2023

Complex Dynamical Systems Group
A-1040 Wien, Gusshausstr. 27, Internet: http://www.acin.tuwien.ac.at

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub

# Acknowledgment

I want to express my gratitude to my professor Dr. Univ.-Prof. Dr.techn. Andreas Kugi and my supervisor Dipl.-Ing. Florian Beck for their invaluable guidance and support throughout this thesis. Their insights and expertise were instrumental in shaping my ideas and refining my research.
I would also like to extend my thanks to my friends, who have been a constant source of encouragement and motivation. I also want to express my gratitude to my friends, who have constantly motivated and supported me.
To Willi, thank you for always being my sunshine, even on the darkest days. Your positive attitude and outlook on life have been a constant source of inspiration.
To Iris, for the stimulating conversations, even at 3 AM.
To Sarah, for being the voice of motivation.
To Simon, for challenging me to think critically about every concept and reminding me that sometimes we must say goodbye to make progress.
To Felix, for being the knight in shining armor.
To Hans, for being the power flower that helped me to stay positive.
To Erich, for his meticulous attention to detail.
And to Kristof, for being the Englishman.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

# Abstract

Robotics is a vibrant field of research with great potential to assist humans in various ways, particularly with tasks in challenging and uncertain environments that require high mobility and dexterity. Autonomous mobile manipulators are, therefore, utilized more frequently compared to conventional manipulators since they can move independently, execute various tasks with high agility, and perform more time-independent or repetitive operations, such as handling and transportation.
The increasing demand for wheeled mobile manipulators in high-precision applications has resulted in new requirements emphasizing resource efficiency and predictive behavior, including trajectory planning and designing an ideal path where the robot can later anticipate further actions with the least amount of energy.
This work focuses on assessing the workspace, determining singularities, determining typical performance indicators for robots, and utilizing them. To construct trajectories, a nonlinear optimization problem was formulated, and a cost function with constraints was established. The trajectory planning is optimized in terms of workspace and manipulability of the robot, ensuring that the robot successfully reaches the desired pose while complying with the physical limitations of the system. The impact of manipulability on the resulting trajectories and how the weighting of this component affects the outcomes were given particular attention.
It was found that including manipulability in the cost function enables the robot's final configuration to be anticipatory, purposeful, and prepared for further assignments. As a result, the robot has more opportunities to perform additional tasks nearby while saving energy and resources by not having to relocate the platform. Case studies demonstrate the implementation and the different results.
Future research may find it beneficial to include performance measures in trajectory planning and to incorporate the acquired workspace information even further, as the topic has merit in several areas. Being able to accomplish additional tasks when arriving at a final configuration without consuming much energy opens further research on this topic in medical care, disaster control, or energy crises, offering promising possibilities.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

# Kurzzusammenfassung

Die Robotik ist ein dynamisches Forschungsgebiet mit großem Potenzial, den Menschen auf verschiedene Weise zu unterstützen, insbesondere bei Aufgaben in anspruchsvollen und unsicheren Umgebungen, die hohe Mobilität und Geschicklichkeit erfordern. Autonome mobile Manipulatoren werden daher im Vergleich zu konventionellen Manipulatoren häufiger eingesetzt, da sie sich selbstständig bewegen, eine Vielzahl von Aufgaben mit hoher Agilität ausführen und zeitunabhängige oder sich wiederholende Operationen, wie beispielsweise Handhabung und Transport, durchführen können.
Die steigende Nachfrage nach mobilen Manipulatoren in hochpräzisen Anwendungen hat zu neuen Anforderungen geführt, bei denen die Ressourceneffizienz und das vorausschauende Verhalten im Vordergrund stehen, einschließlich der Trajektorienplanung und des Entwurfs eines idealen Pfades, auf dem der Roboter später weitere Aktionen mit dem geringsten Energieaufwand antizipieren kann.
Diese Arbeit untersucht die Bewertung des Arbeitsraums, die Bestimmung von Singularitäten, die Ermittlung typischer Leistungsindikatoren für Roboter und deren Nutzung. Zur Konstruktion von Trajektorien wurde ein nichtlineares Optimierungsproblem formuliert und eine Kostenfunktion mit Nebenbedingungen aufgestellt. Die Trajektorienplanung wird hinsichtlich des Arbeitsraums und der Manipulierbarkeit des Roboters optimiert, während der Roboter erfolgreich die gewünschte Pose erreicht und gleichzeitig die physikalischen Grenzen des Systems einhält. Besonderes Augenmerk wurde auf die Auswirkungen der Manipulierbarkeit auf die resultierenden Trajektorien gelegt und darauf, wie sich die Gewichtung dieser Komponente auf die Ergebnisse auswirkt.
Es wurde festgestellt, dass die Manipulierbarkeit in der Kostenfunktion die Endkonfiguration des Roboters vorausschauend und zielgerichtet auf weitere Aufgaben vorbereitet. Infolgedessen hat der Roboter mehr Möglichkeiten, zusätzliche Aufgaben in der Nähe auszuführen, während er gleichzeitig Energie und Ressourcen spart, da er die Plattform nicht gezwungenermaßen weiterbewegen muss. Fallstudien veranschaulichen die Umsetzung und die unterschiedlichen Ergebnisse.
Zukünftige Forschungen könnten es als vorteilhaft erachten, Leistungsindikatoren in die Trajektorienplanung einzubeziehen und die gewonnenen Arbeitsrauminformationen noch stärker zu berücksichtigen, da das Thema in mehreren Bereichen von Nutzen ist. Die Möglichkeit, in der Endpose zusätzliche Aufgaben erfüllen zu können, ohne viel Energie zu verbrauchen, eröffnet weitere Forschungen zu diesem Thema in den Bereichen medizinische Versorgung, Katastrophenschutz oder Energiekrisen und bietet vielversprechende Möglichkeiten.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

# Contents



Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Wien Bibliothek Your knowledge hub

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

# Chapter 1

# Introduction

Autonomous mobile manipulators are increasingly used for various tasks in different application areas [1]–[3]. They are primarily concerned with tasks in arduous and precarious environments, e.g., in industry, mining, nuclear facilities, forestry, and transportation, as well as in human-machine interactions and precision work.

As a combination of a polyarticulated arm and a mobile platform [4], mobile manipulator systems offer various benefits over their stationary pendants: they are transportable and more maneuverable while having an enlarged workspace [2]. Therefore, they predominantly execute tasks based on their high mobility and dexterity.

However, the demands on the arm and the platform frequently contradict [5]. The manipulator requires a firm base to prevent falling over during its motion, while the mobile platform should be kept lightweight and agile. Thus, the structure of the mobile manipulator must be carefully engineered and designed to account for these demands. Furthermore, operation and control are inherently complex due to the numerous degrees of freedom and the unstructured working environment. Therefore, to get a better understanding of the complexity, modeling the systems is crucial.

Optimal trajectory planning for mobile manipulator systems is a challenging benchmark [6] in control engineering, especially when the manipulator has more degrees of freedom than required to complete an assigned task. Usually, the robot has not only to reach the desired position and orientation in Cartesian space, but it is reasonable to assume that it has to perform further tasks near the goal configuration. For example, a mobile manipulator has to approach an overturned book lying on a shelf and then arrange it correctly.

Such scenarios necessitate the robot arriving in a configuration that enables it to perform further tasks. Expectations like this typically require trajectory planning with a high level of manipulability [7] at the final configuration. This manipulability measure estimates the versatility of the end-effector's motion by describing the ability to reach a specific pose in the Cartesian space and the ability to change the manipulator's and the end-effector's posture at the final configuration.

The thesis deals with developing and implementing a mobile manipulator model and analyzing its kinematics and workspace. Additionally, an optimized trajectory planner is designed which is able to realize the same end-effector goal pose with different joint



Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

configurations depending on the given objectives. The focus lies on the benefits of including manipulability. The main contribution of this work is the in-depth analysis and experiments on the abovementioned topics.

This introductory chapter serves as an overview by summarizing the state of the art of mobile manipulators, trajectory planning, and performance indices, followed by the research objectives and questions. Finally, a summary of the thesis structure concludes the chapter.

## 1.1 Literature review

Robotics is currently one of the major research areas, as it is a constantly growing discipline with increasing research interest and use in diverse applications [8], [9]. According to Bogh et al. [10], autonomous mobile manipulation is a more recent, interdisciplinary field within robotics with a timeline extending over decades. Innovative techniques and technologies such as RoboCup [11] and 3D object localization [12] aim to enhance the agility and versatility of robotic systems while reducing costs.

Traditionally, mobile robots have primarily been used for transportation due to their maneuverability, while stationary manipulator robots have been favored in manufacturing due to their flexibility. Mounting a polyarticulated arm on a mobile platform combines the advantages of both systems. The end-effector is usually attached to the manipulator's free end, while the platform is equipped with either two separately driven wheels or an omnidirectional drive [13]. By integrating the maneuverability of mobile robots with the flexibility of manipulator arms, mobile manipulators can execute more complex tasks than their stationary counterparts [2], [13], [14].

As the use of mobile manipulators continues to expand across various industries and sectors, including agriculture [15], healthcare [16], and military [17], it has become increasingly important to measure and optimize their performance. This is critical for addressing each application's unique challenges and problems. For example, mobile manipulators can assist with precision agriculture tasks like crop monitoring and harvesting, improve patient care in healthcare settings, and enable the safe handling of explosive ordnance in military applications.

Manipulators, especially mobile manipulators, often possess more degrees of freedom than necessary for a particular task, known as redundancy. The kinematic redundancy of manipulators has been a focus of research over the past three decades, and it continues to be an active area of investigation. Many researchers have contributed to this field, exploring the potential benefits of enhancing performance and addressing problems, e.g., Y. Nakamura [18], Pin and Culioli [19], Zhang and Wang [20], Zhang et al. [5], Li and Li [21], and Chiaverini et al. [22].

While redundancy can pose a challenge, it also presents advantages, such as enabling different motions to accomplish the same objective. Redundant robots make it easier to attain an optimal configuration than non-redundant variants because the redundant degree of freedom provides more alternatives for the final pose. Therefore, optimizing the robot's configuration during or at the end of its movement is highly tied to this attribute.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

A limitation of any stationary manipulator is that targets outside its workspace cannot be reached. A mobile platform can effectively expand the manipulator's workspace while the manipulator offers various operational functionalities. Guaman et al. [23], among others, have analyzed how the platform and arm contribute additional degrees of freedom to the coupled system, increasing versatility. Due to the arm's higher precision compared to the platform [4], a common approach for controlling mobile manipulators is to move the platform to a desired location, park it, and then utilize the manipulator arm for high-precision manipulation.

The kinematics of standard manipulator arms are often described by the Denavit-Hartenberg method [24], which uses a minimal set of parameters. In contrast, the screw-based approach, such as the successive screw displacements method [25], provides more flexibility, albeit at the cost of increased complexity. Both modeling approaches offer unique benefits, as extensively demonstrated by Rocha et al. [25].

Mobile manipulators present challenges for planning algorithms due to their kinematic redundancy and complex dynamics. Two common approaches are separate planning for the manipulator and platform subsystems or combined planning, treating the system as a whole. Sandakalum et al. [6] recommend merging the two subsystems because separate planning may not result in optimal outcomes.

Various researchers have studied modeling and motion trajectory planning, including search-, sampling-, or optimization-based algorithms [6], [26], [27]. Common goals are optimizing movement and posture, which are usually subject to constraints, e.g., joint limitations. Given that robot performance depends on multiple factors, such as accuracy, speed, and energy efficiency, it is crucial to establish measurable and comparable performance evaluations.

Over the years, several performance indices have been developed to evaluate manipulators, including service angle, dexterity, condition number, minimum singular value, and manipulability index [28]. The workspace can also be considered a performance index, as noted by Ouyang and Shang [29].

The *service angle*, for example, describes the full range of the approach angle around a specific point of the manipulator's workspace [28]. Based on this measure, Yang and Lai [30] established the concept of the service sphere and service regions. On the other hand, the *dexterous workspace*, introduced by Kumar and Waldron [31], includes all points that the end-effector can access in any arbitrary orientation. Accordingly, the *dexterity index* [31] measures the manipulator's ability to reach different orientations at each point within its workspace, and its value varies between 0 and 1.

The Jacobian matrix is a fundamental tool in robotics [32] for performance analysis as it describes the conversion between joint angular and Cartesian end-effector velocities. One of its main benefits is helping to detect proximity to singularities, which can lead to unpredictable behavior or even failure of the manipulator. Several performance measures in robotics rely on the Jacobian matrix or can be derived from it.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

To estimate the kinematic performance, Paul and Stevenson [33] used the determinant of the Jacobian matrix. If the determinant equals zero at a given point in the manipulator's workspace, the manipulator is approaching a singularity. However, this value alone does not provide information about the degree of ill-conditioning. To address the issue of ill-conditioning, the *condition number* can be used. This measure compares the matrix's largest and smallest singular values [34] [33].

Another useful indicator based on the Jacobian matrix is the *minimum singular value* by Klein and Blaho [35]. It provides a high sensitivity near singularities and efficiently shows whether the determinant of the Jacobian matrix is approaching zero.

The *manipulability index*, introduced by T. Yoshikawa [36], is another measure based on the Jacobian matrix. It evaluates the current configuration by quantifying the extent to which the velocity of the end-effector can change depending on its current pose. The index measures the relationship between joint space and Cartesian end-effector motion and provides information about the manipulator's ability to move and apply forces in arbitrary directions.

Manipulability is fundamental in various aspects of robotics, including analysis, design, robot control, task specification, and optimization. Researchers have extensively studied the maximization of the manipulability index, such as Dufour and Wael [37] and Maric et al. [38] contributing to the field. The index is often used to solve inverse kinematics models or online control of redundant systems, as noted by Vahrenkamp et al. [39]. t also plays a significant role in developing optimal placement methods for mobile manipulation, as explained by Cui et al. [40]. The manipulability index can also be utilized for planning optimal trajectories, as emphasized by Akli et al. [41], and determining reachability data and control approaches, as shown by Andaluz et al. [42] and Vahrenkamp et al. [43].

Yamamoto and Yun [44] introduced the task-space ellipsoid as a measure of manipulability in their analysis of locomotion and manipulation from a task-space perspective, making significant contributions to the kinematic and dynamic analysis of robots. Bayle et al. [45] extended the definition of manipulability for mobile manipulators with movement-constrained, also known as nonholonomic, platforms, enabling its application in an inversion procedure aimed at solving redundancy.

To ensure that all robot objectives are achieved, manipulability is typically incorporated as a criterion paired with other factors. For example, when choosing a mobile manipulator's optimal pose, the manipulability and cycle time parameters should be added, as suggested by Merkt et al. [46]. Similarly, Teka et al. [47] recommend including the manipulability index and the manipulator's joint angles as constraints for redundant mobile manipulators. These examples demonstrate how manipulability is used as a criterion in optimization problems to influence the motion and performance of robotic systems.

As mobile manipulators gain prevalence in various industries and sectors, improving their performance through modeling, trajectory planning, redundancy, and workspace expansion is becoming increasingly important.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

## 1.2 Thesis objectives

This thesis aims to develop a kinematic model for a nine-degrees-of-freedom mobile manipulator, analyze its manipulability, and incorporate this measure into an optimal trajectory planning algorithm.

The goal is to provide a solution that balances energy consumption and execution time with a high manipulability index. Manipulability optimization is crucial in achieving a final posture farther from singular configurations, providing a more flexible motion and a wider operational space.

This work tackles the following topics:

1. Analyze and develop the kinematic model of a seven-axis manipulator and the mobile platform separately to solve the related kinematic problems.
2. Merge the manipulator and the platform into one coherent model to enable an in-depth analysis of the possible movements.
3. Create and analyze the robot's workspace to better understand its heterogeneous structure.
4. Calculate and analyze the different configurations of the robot to grasp the concept of manipulability and singular configurations.
5. Define and implement an optimization problem to plan optimal trajectories incorporating manipulability.
6. Evaluate different test scenarios to analyze the influence of manipulability on the planned trajectories and highlight the significant findings.

Special consideration is given to the numerical effort and computation time while implementing and solving the presented optimization problem to achieve a reasonable execution time and iteration number. Furthermore, a desired property of the resulting solution is that the system uses a fixed set of parameters so that tuning is not necessary for each case individually.

## 1.3 Thesis structure

This work is organized into five chapters. After the brief introduction in Chapter 1, Chapter 2 provides a detailed mathematical model of the mobile robot system, including the forward and inverse kinematics. The chapter also covers fundamental concepts for modeling robotic systems, such as geometric representation and the Denavit-Hartenberg convention. In addition, it presents the analysis and implementation of the manipulator's model combined with the differential drive platform.

Chapter 3 focuses on a detailed description of the manipulator's workspace and its manipulability. First, it discusses the problems associated with the workspace representation and proposes voxels and alpha shapes as options. Then, with the help of the mathematical model, the robot's workspace and singularities are analyzed. The last section explains the difficulties and advantages that arise with the inclusion of manipulability.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Chapter 4 discusses the basics of trajectory optimization and presents the underlying optimal control problem. It then addresses the specific implementation details of the solution to the optimization problem and presents results from various case studies.

Finally, Chapter 5 concludes the thesis by summarizing the main findings and results. It also provides recommendations for future work.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

# Chapter 2

# Mathematical modeling

This chapter deals with the derivation of the mobile manipulator's kinematic equations of motion. Figure 2.1 shows the modeled robotic system of the *KUKA LBR iiwa R820* seven-axis manipulator mounted on the *SALLY V2.0* mobile platform.

Figure 2.1: The modeled seven-axis mobile manipulator of the *KUKA LBR iiwa R820* seven-axis manipulator mounted on the *SALLY V2.0* mobile platform.

Kinematic models characterize and control the behavior of robots by considering purely geometric relationships, disregarding the connection between forces and motion. To simplify the modeling process, the mobile manipulator system is divided into two distinct subsystems, namely, a mobile platform and a robotic arm. Subsequently, these subsystems are combined to create the integrated system model. Accordingly, this chapter presents the mathematical representation of both subsystems and the combined system.

The model of the articulated manipulator arm is based on the *KUKA LBR iiwa R820* redundant seven-axis manipulator of the KUKA GmbH [48]. *LBR* stands for *'Leichtbauroboter'*, also known as *LWR 'lightweight robot'*, while *iiwa* stands for *'intelligent industrial work assistant'*. The *KUKA LBR iiwa R820* belongs to the lightweight robots with a load capacity of 14 kg and 820 mm range.

As for the mobile platform, the model is based on the *Sally V2.0* DS Automation differential drive platform [49].



Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

The serial manipulator is an open kinematic chain formed by several stiff structural elements, referred to as links, connected by rotational joints that allow relative movement between adjacent links. The major structural components are a base frame, a joint module, and a central hand. The *KUKA LBR iiwa R820* is shown in Figure 2.2.

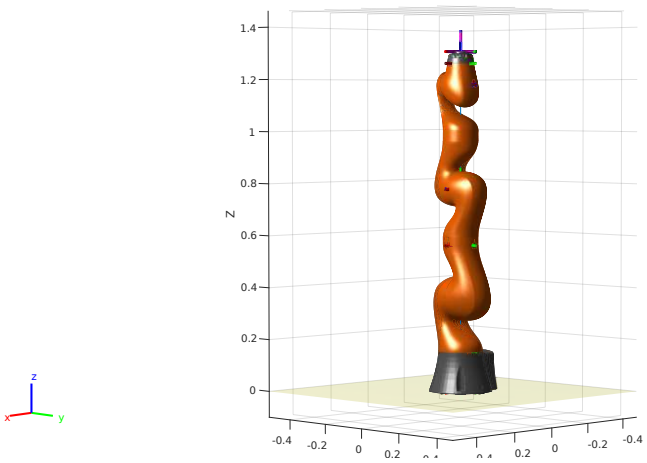


Figure 2.2: The *KUKA LBR iiwa R820* modeled in MATLAB R2021.

Table 2.1 shows the manipulator's joint limits of both the joint position and angular velocity of all seven joints according to the KUKA data sheet [50].

| $\theta$ | range of motion | speed with rated payload |
|---|---|---|
| $\theta_1$ | $\pm 170°$ | $\pm 85°/s$ |
| $\theta_2$ | $\pm 120°$ | $\pm 85°/s$ |
| $\theta_3$ | $\pm 170°$ | $\pm 100°/s$ |
| $\theta_4$ | $\pm 120°$ | $\pm 75°/s$ |
| $\theta_5$ | $\pm 170°$ | $\pm 130°/s$ |
| $\theta_6$ | $\pm 120°$ | $\pm 135°/s$ |
| $\theta_7$ | $\pm 175°$ | $\pm 135°/s$ |

Table 2.1: The *KUKA LBR iiwa R820* joint position and joint velocity limits [50].

This chapter begins with a general introduction to rigid-body structures, the definition of degrees of freedom, and redundancy. Then, it covers the following objectives regarding the manipulator arm:

1. Create a parameter-dependent model using the kinematic relationships to determine the connection between joint and Cartesian coordinates.
2. Calculate the transformation matrices to solve the forward and inverse kinematics problems.
3. Derive the Jacobian matrix to determine the relation between the joint velocity and the Cartesian end-effector velocity space.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Figure 2.3 shows the *SALLY V2.0* mobile platform.

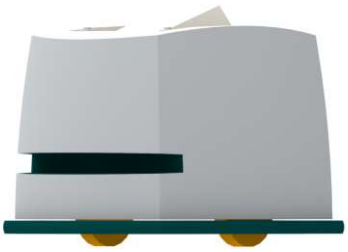

Figure 2.3: The *SALLY V2.0* mobile platform from DS Automotion.

The drive mechanism employed by *SALLY V2.0* is a differential drive system comprising two independently driven wheels arranged on a single axis. The technical data sheet [49] provides the relevant technical information, as summarized in Table 2.2.

| Sally V2.0 technical data | |
|---|---|
| track width | 0.2 $m$ |
| wheel diameter | 0.15 $m$ |
| max. forward velocity | 1.6 $m/s$ |

Table 2.2: *SALLY V2.0* mobile platform technical specifications [49].

The chapter addresses the following objectives in order to develop the platform's model:

1. Implement a parameter-dependent model, including the kinematic relationships to solve the forward kinematics.
2. Analyze and address the specific characteristics of the differential drive model, which has to be combined with the manipulator arm model.

## 2.1 Manipulator kinematics

This section provides a comprehensive overview of manipulator kinematics and is based on the theoretical principles in robot kinematics presented in Robot Modeling and Control by Spong et al. [32].

Manipulator kinematics is primarily concerned with solving problems of *forward kinematics* and *inverse kinematics*, using their results for further analysis [32]. *Forward kinematics* allows calculating the position and orientation of the robot's end-effector as a function of the manipulator's joint coordinates [32], whereas *inverse kinematics* allows finding the values of the joint coordinates based on the position and orientation of the end-effector in Cartesian coordinates [32].

As manipulators are commonly modeled as rigid-body structures, this section introduces the concept of rigid-body structures and the definitions of degrees of freedom and redundancy. It describes the pose of a rigid body in terms of position and orientation. In addition, it briefly presents three methods of defining orientation: rotation matrices, Euler angles, and quaternions, followed by the Denavit-Hartenberg convention.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

This section also derives the mathematical model of the manipulator and solves the associated forward and inverse kinematics problems. It is concluded by computing the geometric Jacobian matrix and examining its relevance in avoiding singularities by a singularity analysis.

### 2.1.1 Rigid-body structures

According to the framework outlined by Spong et al. [32], rigid-body kinematics describes the geometric behavior of a robot manipulator to a fixed reference frame, also called base or world frame, without considering the forces and moments causing the motion. The kinematic model defines the relationship between the robot's joint variables and the manipulator's end-effector pose.

A robotic manipulator's mechanical structure comprises a kinematic chain that can be divided into two subsystems: an articulated mechanism and an end-effector [32]. The articulated mechanism comprises a chain of rigid links interconnected by joints anchored at a fixed base, while the end-effector manipulates or interacts with the environment and is typically located at the chain's most distal joint.

The topology of a robotic manipulator's structure refers to the configuration of the connections between its rigid links in the articulated mechanism [32]. It can be represented as a graph, with links represented as nodes and joints represented as edges. The selected topology significantly impacts the robot's behavior and complexity [32]. Figure 2.4 displays two primary topologies in manipulators: open and closed chains.

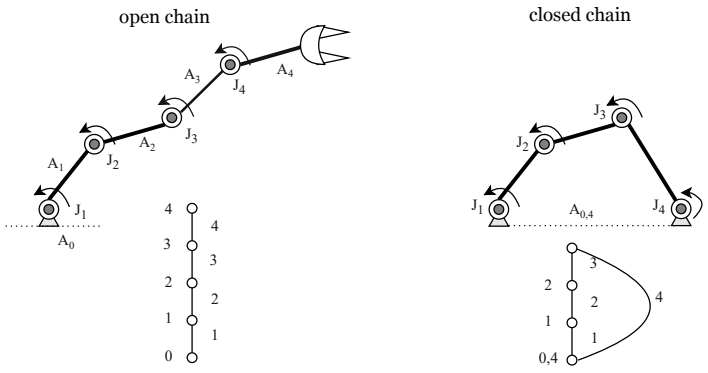


Figure 2.4: Examples of open and closed chain rigid-body topologies.

**Open chains** consist of rigid bodies that are connected in sequence [32] as a serial or tree structure, starting from a fixed base and ending at the extremity. The resulting motion is obtained by composing the elementary movements of each link in relation to the preceding one. The structure is characterized by its ease and low complexity, with typically all joints being actuated. This makes open chains particularly suitable for industrial applications, and many industrial manipulators are serial-structured open kinematic chains [32].

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

**Closed chains** are comprised of rigid bodies interconnected in a manner that forms at least one closed loop [32]. Compared to open kinematic chains, closed chains present greater complexity and technical challenges, often involving fewer actuated joints. On the other hand, they offer certain benefits, including increased rigidity and accuracy due to their higher stiffness levels [32].

### 2.1.2 Degrees of freedom and redundancy

As outlined by Spong et al. [32], a crucial parameter shaping the performance of robots is the number of *degrees of freedom (DOF)*, referring to specific modes which represent the range of possible movements for a given kinematic pair and significantly impact the usability of industrial robots. Robots are typically represented by rigid bodies, with their number of DOF dependent on the constraints these bodies must obey [32]. One such constraint is the *non-deformity*, which requires the relative distance between two points on a rigid body to remain constant.

The connection between two adjacent links is referred to as a joint, providing limited relative mobility between two rigid components. The type of joint used determines the number of constraints. The most common joint types are prismatic (translational movement in one direction), revolute (rotational movement in one direction), and spherical joints (rotation in three directions).

Prismatic and revolute joints place five constraints on the motion between two rigid-body parts, leaving one degree of freedom free. Prismatic joints allow translation along one axis, while revolute joints enable rotation around one axis. The number of DOF in an open-chain manipulator typically equals the number of joints and independent configuration parameters. Figure 2.5 depicts a schematic representation of prismatic and revolute joints.

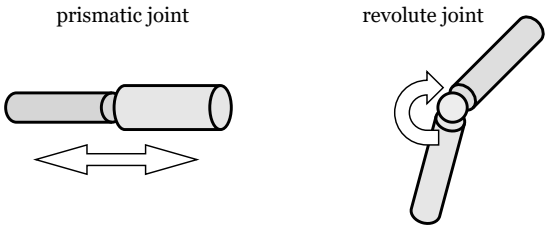


Figure 2.5: Schematic representation of prismatic and revolute joints.

The DOF of the configuration space represents the minimum number of independent displacements necessary to represent a given configuration sufficiently. The configuration space $\mathbb{R}^3$ requires a minimum of six degrees of freedom to represent the Cartesian space in the *'special Euclidean group' SE(3)* [32]. Three of them define the position, while the other three define the orientation by angles, which are commonly referred to as roll, pitch, and yaw (RPY), using aviation terminology [32].

Kinematic redundancy in a manipulator refers to a situation where the manipulator has a higher DOF than the minimum required in the task space. This allows the manipulator to solve the same task in different ways. The concept of redundancy and its associated challenges are discussed in several works, e.g., by Blaho et al. [35].

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

According to [35], although redundancy increases the complexity of planning and control, it provides several benefits, such as improved singularity and obstacle avoidance, posture optimization, and the ability to include additional objectives in the planning process.

Let a manipulator have $a$ axes that equal its number of degrees of freedom. Let $e$ be the dimension of the motion attainable by the end-effector in the task space. Let $t$ be the dimension of the assigned task in the task space. Then the following cases can occur, as outlined in [22] and [32]:

a) $a = e$: a non-redundant robot.

b) $a > e$: a kinematically redundant robot, as the manipulator's configuration can be changed without altering the end-effector's current pose (null-space motion).

c) $e = t$: the end-effector can provide the DOF required in the task space.

c) $e > t$: task redundancy, the end-effector has more DOF than required by the task space.

d) $e < t$: the dimension of the task space exceeds the dimension of the motion attainable by the end-effector.

The movement of the end-effector over time is referred to as task-space motion. On the other hand, null-space motion occurs when the end-effector stays in the same pose, but the manipulator changes its configuration. This allows for greater versatility in the robot's movements. Figure 2.6 shows the difference between task-space and null-space motions.


Figure 2.6: Example of task-space motion and null-space motion of a four-axis redundant manipulator in an $SE(2)$ task space.

A simple expression to determine the degrees of freedom is to use Gruebler's formula [24], which is widely used in robotics due to its simplicity. The formula (2.1) assumes that all constraints are independent

$$DOF = m \cdot (N - 1 - J) + \sum_{i=1}^{J} f_i, \tag{2.1}$$

where $m$ denotes the degrees of freedom of a rigid body, $N$ is the number of rigid-body parts, including the ground, $J$ is the number of joints, and $f_i$ stands for the freedom per joint type.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

The *KUKA LBR iiwa R820* has seven revolute joints, meaning that $J = 7$, $N = 8$, $f_i = 1$ for $i$ 1 to 7 and operates in an $SE(3)$ task space, so that $m = 6$. By applying Gruebler's formula, the degrees of freedom in (2.2) can be calculated as

$$DOF = 6 \cdot (8 - 1 - 7) + \sum_{i=1}^{7} 1 = 7. \tag{2.2}$$

The *KUKA LBR iiwa R820* is a robot with more than six degrees of freedom, which makes it a redundant manipulator for positioning tasks in an $SE(3)$ task space.

### 2.1.3 Position, orientation, and quaternions

This subsection strongly relies on the seminal work of B. Siciliano [51], which presents a comprehensive overview of the foundations of robotics.

The *pose* of a rigid body consists of its *position* and *orientation* [51]. The position specifies the current location of the end-effector frame origin in the three-dimensional Cartesian space, whereby the orientation represents the rotation of the end-effector frame relative to a fixed reference frame. In terms of the manipulator's task, positioning refers to placing the end-effector at a specific point within its workspace, and orientation refers to the alignment of the end-effector with the desired orientation at that position. To fully describe the pose of the rigid body in the three-dimensional Cartesian space, the position and orientation must be specified using three coordinates each [51].

Each part of a rigid body, including the end-effector, is associated with its own unique coordinate frame. The frames are defined as right-handed so that the cross products of the $x$ and $y$ axes result in the $z$ axis. These axes $x$, $y$, and $z$ form an orthonormal base, meaning they are perpendicular to each other and have a unit length. Let the coordinate system $(\mathbf{O}, x, y, z)$ denote a reference system $\mathcal{W}$ with its origin $\mathbf{O}$ at zero and its axes defined as $x$, $y$, and $z$, respectively.

Let $(\mathbf{O}', x', y', z')$ be the orthogonal coordinate system $\mathcal{B}$ attached to a rigid-body. This coordinate system is referred to as the body frame and represents the spatial orientation of the body relative to $\mathcal{W}$. As illustrated in Figure 2.7, the relative position and orientation of two coordinate systems $\mathcal{W}$ and $\mathcal{B}$ define the pose of the body.

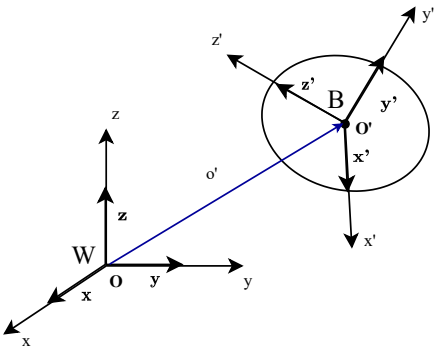


Figure 2.7: Pose of a coordinate system $\mathcal{B}$ in the reference frame $\mathcal{W}$ [51].

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

There are several ways to express the orientation of $\mathcal{B}$ with respect to $\mathcal{W}$, e.g., the *Euler-angles*, the *rotation matrix*, or *quaternions* [52]. Each of the three representations has its advantages and disadvantages.

**Euler angles** are widely used to describe the orientation of $\mathcal{B}$ in $\mathcal{W}$ by a minimal representation [51], as only three parameters are required that each describe a rotation about one of the three coordinate axes. On the other hand, Euler angles are prone to singularities, gimbal lock, and cancellation effects during interpolation.

A gimbal lock occurs when two of the three possible axes of rotation are parallel, resulting in a loss of one degree of freedom [52] [53]. However, it can also arise during interpolation between two orientations. Thus, the choice of Euler angle parametrization should be carefully considered, and two consecutive angles should not be about two parallel axes. There are 12 allowed combinations of sets, e.g., the ZYZ and the RPY angles.

The **rotation matrix** $\boldsymbol{R}$ is another representation of the orientation of $\mathcal{B}$ relative to $\mathcal{W}$. As a square $3 \times 3$ matrix, its columns serve as the basis vectors of the coordinate system and are subject to six constraints: three for being unit vectors and three for being perpendicular to each other. Despite their good analytical features, rotation matrices require more memory when stored compared to other representations.

The rotation matrix has several important characteristics:

- Its inverse is equal to its transpose, $\boldsymbol{R}^T = \boldsymbol{R}^{-1}$.
- As a result, $\boldsymbol{R}^T\boldsymbol{R} = \boldsymbol{I}$, where $\boldsymbol{I}$ denotes the $3 \times 3$ unit matrix.
- The determinant of the matrix $\det(\boldsymbol{R})$ is equal to 1, meaning that rotating a vector does not change its length.
- The matrix product of two rotation matrices is also a rotation matrix.

The set of all $3 \times 3$ real matrices that meet these conditions form the *special orthogonal group* $SO(3)$ [54]. Matrices in $SO(3)$ do not commute, meaning that their matrix multiplication order is relevant.

The basic rotation matrices about the three main axes of $\mathcal{W}$ are known as roll, pitch, and yaw, depicted by Figure 2.8. The RPY serve as the building blocks of more complex rotations.

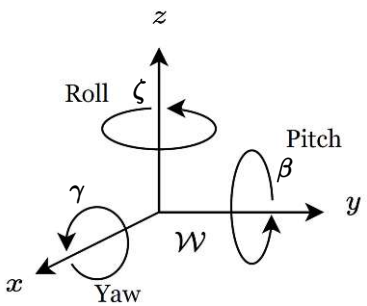


Figure 2.8: Roll, pitch, and yaw rotations about the three main Cartesian axes [32].

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Each rotation matrix is an extension of the two-dimensional rotation matrix, in which one of the coordinates remains constant during the transformation.

- ***Yaw*** is the counterclockwise rotation of $\gamma$ about the $x$-axis, given by

$$\boldsymbol{R}_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\gamma) & -\sin(\gamma) \\ 0 & \sin(\gamma) & \cos(\gamma) \end{bmatrix}. \tag{2.3}$$

- ***Pitch*** is a counterclockwise rotation of $\beta$ about the $y$-axis, given by

$$\boldsymbol{R}_y = \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix}. \tag{2.4}$$

- ***Roll*** is a counterclockwise rotation of $\zeta$ about the $z$-axis, given by

$$\boldsymbol{R}_z = \begin{bmatrix} \cos(\zeta) & -\sin(\zeta) & 0 \\ \sin(\zeta) & \cos(\zeta) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{2.5}$$

Any arbitrary rotation $\boldsymbol{R}$ can be achieved by multiplying the basic rotation matrices about the principal axes of the reference coordinate system $\mathcal{W}$ in a given sequence, whereby the order of rotations is relevant.

**Quaternions** offer a reasonable tradeoff between memory and analytical features, providing a compact and efficient way to describe three-dimensional orientation, requiring four parameters instead of the nine needed by a rotation matrix [54]. Introduced by William Hamilton in 1843, they are widely used in computer graphics, robotics, and computer vision due to their improved numerical stability, avoidance of problems like gimbal lock, and ability to interpolate between orientations without discontinuities [54].

Quaternions are hypercomplex numbers [53] and serve as extensions of complex numbers. They are composed of a scalar and a vector, expressed as

$$\mathbf{r} = x_0 + x_1\mathbf{i} + x_2\mathbf{j} + x_3\mathbf{k}, \tag{2.6}$$

with $x_0$, $x_1$, $x_2$ and $x_3$ as real numbers and $\mathbf{i}$, $\mathbf{j}$, and $\mathbf{k}$ as the basic quaternions, which can be interpreted as unit vectors [54].

### 2.1.4 Homogeneous transformations

The pose of $\mathcal{B}$ relative to $\mathcal{W}$ is described by its position and orientation. The position of $\mathcal{B}$ can be represented by a translation vector $\boldsymbol{t}$ of dimensions $3 \times 1$ and the orientation by a $3 \times 3$ rotation matrix $\boldsymbol{R}$.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

A convenient and compact representation is achieved by combining $\boldsymbol{t}$ and $\boldsymbol{R}$ into a single $4 \times 4$ *homogeneous transformation matrix* $\boldsymbol{T} \in SE(3)$ [55], which is constructed as

$$\boldsymbol{T} = \begin{bmatrix} R_{11} & R_{12} & R_{13} & x \\ R_{21} & R_{22} & R_{23} & y \\ R_{31} & R_{32} & R_{33} & z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{R}_{3\times3} & \boldsymbol{t}_{3\times1} \\ \boldsymbol{0}_{1\times3} & 1 \end{bmatrix}, \tag{2.7}$$

with the rotation matrix $\boldsymbol{R} \in SO(3)$, the $\boldsymbol{t}_{3\times1}$ translation vector, $\boldsymbol{0}_{1\times3}$ as the perspective vector and 1 as the global scaling scalar. It should be noted that $\boldsymbol{T}$ always represents a rotation followed by a translation and not the other way around.

The transformation matrix $\boldsymbol{T}$ satisfies properties similar to those of the rotation matrix $\boldsymbol{R}$. The inverse $\boldsymbol{T^{-1}}$ of $\boldsymbol{T}$ exists such that $\boldsymbol{TT^{-1}} = \boldsymbol{I}$, where $\boldsymbol{I}$ equals the $4 \times 4$ identity matrix. The product of two transformation matrices also results in a transformation matrix. However, the matrix multiplication is not commutative, meaning that the order of multiplication is relevant. The set of all transformation matrices forms $SE(3)$.

To benefit from the advantages of the transformation matrix, it is necessary to convert the Cartesian coordinates into *homogeneous coordinates*. This allows representing $N$-dimensional coordinates using $N + 1$ numbers. The conversion appends a scalar variable, $w$, to the three-dimensional Cartesian coordinates, which results in a point represented as $\begin{bmatrix} x & y & z & w \end{bmatrix}^T$ in homogeneous coordinates, as opposed to $\begin{bmatrix} X & Y & Z \end{bmatrix}^T$ in Cartesian space. The homogeneous coordinates are scale-invariant, meaning that different homogeneous coordinates can represent the same point in Cartesian space, depending on the scaling factor $w$.

The conversion from homogeneous coordinates to Cartesian is as follows

$$\begin{bmatrix} X & Y & Z \end{bmatrix}^T = \begin{bmatrix} \frac{x}{w} & \frac{y}{w} & \frac{z}{w} \end{bmatrix}^T. \tag{2.8}$$

To create the transformation matrix, the $3 \times 1$ Cartesian vectors are converted into $4 \times 1$ homogeneous vectors by simply setting the scaling factor $w = 1$ and appending it at the end of each vector. This representation is known as the homogeneous coordinate representation [55].

Assuming a rotation described by the RPY, with the abbreviations $c_\zeta = \cos(\zeta)$, $c_\beta = \cos(\beta)$, $c_\gamma = \cos(\gamma)$, $s_\zeta = \sin(\zeta)$, $s_\beta = \sin(\beta)$, and $s_\gamma = \sin(\gamma)$ representing the angles, the transformation matrix reads as

$$\boldsymbol{T} = \begin{bmatrix} c_\zeta c_\beta & c_\zeta s_\beta s_\gamma - s_\zeta c_\gamma & c_\zeta s_\beta c_\gamma + s_\zeta s_\gamma & x \\ s_\zeta c_\beta & s_\zeta s_\beta s_\gamma + c_\zeta c_\gamma & s_\zeta s_\beta c_\gamma - c_\zeta s_\gamma & y \\ -s_\beta & c_\beta s_\gamma & c_\beta c_\gamma & z \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2.9}$$

The relationship between coordinate systems can be calculated by multiplying the transformation matrices of the joints

$$\boldsymbol{T}_0^j = \boldsymbol{T}_0^1 \boldsymbol{T}_1^2 ... \boldsymbol{T}_{j-1}^j, \tag{2.10}$$

where $\boldsymbol{T}_{i-1}^i$ denotes the transformation from the $(i-1)^{th}$ coordinate system to the $i^{th}$ coordinate system.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

### 2.1.5 Denavit-Hartenberg Convention

Describing the pose of a rigid body in three-dimensional space requires six variables, three for the position and three for the orientation. However, working with six parameters can be tedious.

To address this issue, Jacques Denavit and Richard S. Hartenberg published the **Denavit-Hartenberg** (D-H) **convention** in 1955 as a compact method for setting up coordinate systems for rigid-body systems [56] [57]. This convention describes the transformation between consecutive coordinate frames using four variables: link length, twist, offset, and joint angle [57]. Using the D-H parameters, the number of parameters required to describe the pose of a rigid body reduces from six to four.

With this compact representation, an anthropomorphic manipulator with seven degrees of freedom requires only $7 \cdot 4 = 28$ independent parameters, compared to $7 \cdot 6 = 42$ using the canonical approach, allowing manipulation to be more efficient.

The D-H convention can transform any arbitrary coordinate system into another by applying two rotations and two offsets in a specific order. It provides a general description of robotic geometry and establishes coordinate systems for manipulators using homogeneous transformations [57].

Typically, rigid-body configurations are represented by right-handed frames so that each link is assigned to a local coordinate system attached to it using orthogonal coordinate frames. The center of the joint must coincide with the coordinate frame to which it is attached, and the $z$-axis has to point in the direction of the joint.

There are two ways [53] [55] to define the parameters: the *standard D-H convention* and the *modified D-H convention*. These two conventions slightly differ in the parameter definition.

The standard D-H convention, as shown in Figure 2.9, assumes that the $i^{th}$ ($\mathbf{O}_i$, $x_i$, $y_i$, $z_i$) coordinate frame is at the $(i+1)^{th}$ joint and the $z_i$ axis is placed in the direction of the $(i+1)^{th}$ joint [51]. Furthermore, the $x_i$ axis falls in the shared normal (normal transverse direction) of the axes of the $(i+1)^{th}$ and $i^{th}$ joint (in $z$-direction) and points to the $(i+1)^{th}$ joint. The robot's base frame ($\mathbf{O}_0$, $x_0$, $y_0$, $z_0$) can be set arbitrarily. The only constraint is that the $z_0$ axis and the joint's rotation axis must be collinear.

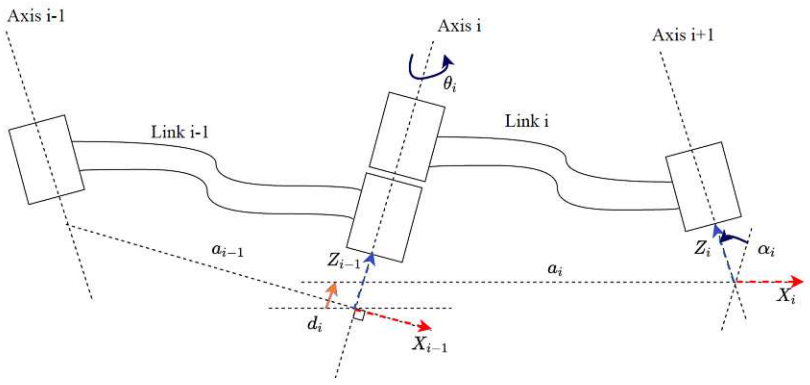


Figure 2.9: Standard Denavit-Hartenberg parametrization [32].

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

The four standard D-H parameters are defined as follows:

- $d_i$: the physical length of the link, the distance from the origin of $(i-1)^{th}$ coordinate frame to the point where the common perpendicular intersects axis $z_{i-1}$. For prismatic joints, this is the joint variable.
- $a_i$: the distance between the $z_{i-1}$ and $z_i$ axes along the $x_i$ axis, the common normal of two consecutive links.
- $\alpha_i$: the angle between two consecutive links, from the $z_{i-1}$ axis to the $z_i$ axis on the $x_i$ axis.
- $\theta_i$: the angle between the current link and the next link, between $x_{i-1}$ and $x_i$ axis on $z_{i-1}$ axis. For revolute joints, this is the joint variable.

By using the D-H convention, the homogeneous transformation matrix $\boldsymbol{T}_{i-1}^{i}$ can be defined from the $(i-1)^{th}$ to the $i^{th}$ frame as

$$\boldsymbol{T}_{i-1}^{i} = \begin{bmatrix} \boldsymbol{R}_z(\theta_i) & \mathbf{0}_{3\times1} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{0}_{3\times3} & \boldsymbol{t}_x(a_i) \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{0}_{3\times3} & \boldsymbol{t}_z(d_i) \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{R}_x(\alpha_i) & \mathbf{0}_{3\times1} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix}. \tag{2.11}$$

The D-H convention is by no means unique. If a given manipulator has only revolute joints, the generalized coordinates $\boldsymbol{q}$ represent the joint angles. The parameters are usually expressed in form of a matrix, where each row represents a single link. The corresponding columns are $a$ (m), $\alpha$ (radians), $d$ (m) and $\theta$ (radians).

One possible set of D-H parameters describing the kinematic chain of the *KUKA LBR iiwa R820* is shown in Table 2.3, based on the KUKA data sheet [50].

| $a(m)$ | $\alpha(radians)$ | $d(m)$ | $\theta(radians)$ |
|---|---|---|---|
| 0 | $-\pi/2$ | 0.36 | $\theta_1$ |
| 0 | $\pi/2$ | 0 | $\theta_2$ |
| 0 | $\pi/2$ | 0.42 | $\theta_3$ |
| 0 | $-\pi/2$ | 0 | $\theta_4$ |
| 0 | $-\pi/2$ | 0.40 | $\theta_5$ |
| 0 | $\pi/2$ | 0 | $\theta_6$ |
| 0 | 0 | 0.06 | $\theta_7$ |

Table 2.3: Selected D-H parameters for the *KUKA LBR iiwa R820*.

### 2.1.6 Manipulator rigid-body model

With the help of the rigid-body structure, the transformation matrices can be expressed according to (2.11) between two adjacent body parts based on the D-H parameters

$$\boldsymbol{T}_{\boldsymbol{i-1}}^{\boldsymbol{i}} = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i)\cos(\alpha_i) & \sin(\theta_i)\sin(\alpha_i) & a_i\cos(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i)\cos(\alpha_i) & -\cos(\theta_i)\sin(\alpha_i) & a_i\sin(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2.12}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

$\boldsymbol{T_0^1}$ (base to first rigid body) and $\boldsymbol{T_1^2}$ (first to second rigid body) serve as examples

$$\boldsymbol{T_0^1} = \begin{bmatrix} \cos(\theta_1) & 0 & -\sin(\theta_1) & 0 \\ \sin(\theta_1) & 0 & \cos(\theta_1) & 0 \\ 0 & -1 & 0 & 0.36 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{2.13}$$

$$\boldsymbol{T_1^2} = \begin{bmatrix} \cos(\theta_2) & 0 & -\sin(\theta_2) & 0 \\ \sin(\theta_2) & 0 & \cos(\theta_2) & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2.14}$$

The entire transformation matrix from the base to the end effector may be obtained by multiplying the transformation matrices

$$\boldsymbol{T_0^7} = \boldsymbol{T_0^1 T_1^2 T_2^3 T_3^4 T_4^5 T_5^6 T_6^7}. \tag{2.15}$$

This leads to the transformation matrix $\boldsymbol{T_0^7}$ from the base to the end-effector

$$\boldsymbol{T_0^7} = \begin{bmatrix} \boldsymbol{R_0^7} & \boldsymbol{t_0^7} \\ \boldsymbol{0}_{1\times 3} & 1 \end{bmatrix} = \begin{bmatrix} (R_0^7)_{11} & (R_0^7)_{12} & (R_0^7)_{13} & (t_0^7)_1 \\ (R_0^7)_{21} & (R_0^7)_{22} & (R_0^7)_{23} & (t_0^7)_2 \\ (R_0^7)_{31} & (R_0^7)_{32} & (R_0^7)_{33} & (t_0^7)_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2.16}$$

In order to speed up the optimization algorithm solver, it is necessary to implement $\boldsymbol{T_0^7}$ with symbolic joint variable expressions, despite the resulting elements being long and highly nonlinear expressions. The exact expressions for $\boldsymbol{R_0^7}$ and $\boldsymbol{t_0^7}$ can be found in Appendix A1.

The robot was implemented in the MATLAB R2021 environment as a rigid-body structure. Parameters such as joint type (revolute, prismatic, or fixed), joint limits, adjacent rigid-body parts, mass, the center of mass, and inertia can be specified individually for each joint and rigid-body combination.

Accordingly, the *KUKA LBR iiwa R820* rigid-body model in MATLAB consists of seven rigid-body parts and seven revolute joints. Figure 2.10 depicts the manipulator in its so-called home configuration, which is given by the *KUKA LBR iiwa R820* data sheet [50] as [0° 25° 0° 90° 0° 0° 0°].

### 2.1.7 Forward and inverse kinematics

For open-chain manipulators, the kinematic analysis involves two fundamental problems [32]: the forward (direct) kinematics to find the pose of the end-effector relative given the joint variables and inverse kinematics to find the joint variables given the end-effector pose.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

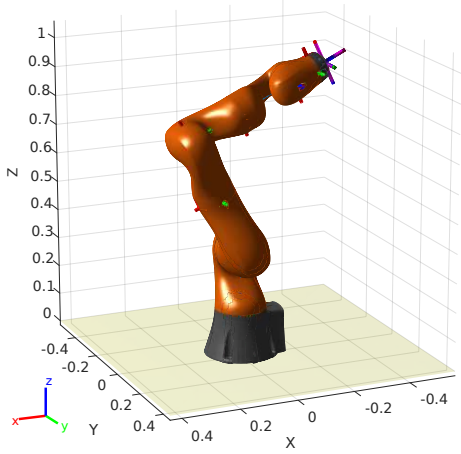


Figure 2.10: *KUKA LBR iiwa R820* in home configuration [0° 25° 0° 90° 0° 0° 0°].

The **forward kinematics** maps the joint variable coordinates to the position and orientation of the end-effector in the Cartesian space, describing its pose as a function of the joint space [24]. The time-dependent forward kinematic equation reads as

$$\mathbf{x}_e(t) = \mathbf{k}(\mathbf{q}(t)), \tag{2.17}$$

where $\mathbf{x}_e(t) \in \mathbb{R}^m$ represents the configuration of the end-effector by a minimal set of coordinates, $\mathbf{q} \in \mathbb{R}^n$ represents the set of the joint variables, and $\mathbf{k}(\mathbf{q}(t)) \in \mathbb{R}^m$ is the nonlinear vector function, which allows the computation of the end-effector pose.

In this specific case, the forward kinematics is covered by the transformation matrix $\mathbf{T}_0^7$ as defined in (2.16). The translation vector $\boldsymbol{t}_0^7$ provides the position in vector form, while the orientation is given by the submatrix $\boldsymbol{R}_0^7$ as a rotation matrix. To obtain an expression for the orientation angles, the matrix was converted into Euler angles.

The **inverse kinematics** aims to find a configuration of the rigid-body structure so that the end-effector has the desired position and orientation in space. It means that the inverse relationship from the end-effector's configuration space to the joint space is required

$$\mathbf{q}(t) = \mathbf{k}^{-1}(\mathbf{x}_e(t)). \tag{2.18}$$

However, inverse kinematics is more complex than forward kinematics as some fundamental challenges arise. While the mapping from joint space to the end-effector's configuration space is unique, the inverse mapping may vary from zero to infinite possible solutions, depending on the desired position and orientation. Ideally, a unique, analytical, and closed-form solution exists, but such a solution can only be obtained for a restricted set of manipulators having specific properties. For arbitrary robots, there is no general analytic solution.

Numerical methods are often preferred over analytical methods in inverse kinematics because they can handle complex and arbitrary robots and are more efficient in finding solutions. These methods are based on matrix inversion and optimization and can

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

incorporate additional constraints, such as joint limits, to reduce the number of feasible solutions. Despite their advantages, finding a solution using numerical methods can be challenging. They typically require an initial guess followed by an iterative optimization process using methods such as Newton or gradient methods.

When multiple solutions exist, they can be differentiated by several criteria to reduce the set of solutions, e.g., minimizing movement from the current position, using the closest solution concept by moving the links with the lightest weight, or collision avoidance.

To simplify the complexity resulting from the redundancy of the seven degrees-of-freedom manipulator, kinematic decoupling can be applied. One analytical decoupling method is known as *Pieper's method* [58], which is applied to manipulators with spherical wrists. Spherical wrists can be considered when three axes intersect at a point. Applying Pieper's method, the inverse kinematics problem is solved in two stages: the inverse kinematics for the position (of the wrist) and the inverse kinematics for the orientation (of the end-effector with respect to the wrist).

### 2.1.8 Analytical and geometrical Jacobian

Forward and inverse kinematics map the joint positions into the end-effector's configuration space and vice versa. However, as the end-effector frame moves and changes its pose, the joint variables also vary [32].

Therefore, to establish a unique relationship between the different types of representational velocities and the differential kinematics of the system, the mapping between the joint velocity space and the end-effector velocity space must be defined [55]. This involves determining the joint variable vector, which is a highly nonlinear function, and, thus, makes differential kinematics calculations nontrivial.

The resulting relationship between joint and end-effector velocities is described by the **Jacobian matrix** [55]. This matrix also connects the end-effector wrenches, joint forces, and torques [55]. There are two types of Jacobian matrices [32]: geometric and analytical.

The *analytical Jacobian* $\mathbf{J}_A$ is directly derived from the forward kinematics by $\frac{\partial \boldsymbol{k}}{\partial \boldsymbol{q}}$ and provides differential quantities in the operational space, while the *geometric Jacobian* $\mathbf{J}_G$ is based on the geometric relation and has quantities with clear geometric meaning.

Although the analytical and geometric Jacobians result in the same linear velocities, their interpretation of the resulting angular velocities differ. For the analytical Jacobian, the angular velocities depend on the local coordinate system of $SE(3)$ and the chosen angles (e.g., roll-pitch-yaw) to represent the orientation of the end-effector. In contrast, the geometric Jacobian calculates the angular velocities around the three orthogonal axes ($x$,$y$,$z$).

As the analytical and geometric Jacobians are related, a mapping exists between them [32], which can be expressed with the help of a parametrization-dependent transformation matrix $\mathbf{E}(\mathbf{L})$ with the choice $\mathbf{L} = [\zeta\ \beta\ \gamma]^T$, and $\zeta$, $\beta$ and $\gamma$ denoting the roll-pitch-yaw angles [53].

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

The matrix $\mathbf{E}(\mathbf{L})$ then is defined as

$$\mathbf{E}(\mathbf{L}) = \begin{bmatrix} 1 & 0 & \sin(\beta) \\ 0 & \cos(\zeta) & -\cos(\beta)\sin(\zeta) \\ 0 & \sin(\zeta) & \cos(\beta)\cos(\zeta) \end{bmatrix}. \tag{2.19}$$

Furthermore, $\mathbf{E}(\mathbf{L})$ can also map the joint velocity limits to end-effector velocity limits. Then the relation between the analytical and geometric Jacobians can be expressed as

$$\mathbf{J}_A(\mathbf{q}) = \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{E}^{-1}(\mathbf{L}) \end{bmatrix} \mathbf{J}_G(\mathbf{q}). \tag{2.20}$$

The **analytical Jacobian** $\boldsymbol{J}_A$ is directly derived from the forward kinematics. Let a minimum set of coordinates $\mathbf{x}_e(t)$ represent the configuration, as in 2.17. The velocities are the time derivatives of these coordinates.

The matrix can be obtained by differentiating the time-dependent joint variables.

$$\dot{\boldsymbol{x}}_e = \frac{\partial \boldsymbol{k}}{\partial \boldsymbol{q}} \frac{\partial \boldsymbol{q}}{\partial t}. \tag{2.21}$$

Now the analytical Jacobian can be expressed as

$$\boldsymbol{J}_A(\boldsymbol{q}) = \frac{\partial \boldsymbol{k}}{\partial \boldsymbol{q}}, \tag{2.22}$$

where $\boldsymbol{J}_A$ denotes the analytical Jacobian matrix $\boldsymbol{J}_A \in R^{m\times n}$, with $n$ as the number of joints, and $m$ as the dimension of the operational space.

With $\boldsymbol{t}_{\mathbf{0}}^{\mathbf{7}}$ and $\mathbf{L}(\mathbf{q})$ as the Euler angle representation of $\boldsymbol{R}_{\mathbf{0}}^{\mathbf{7}}$ from (2.16), the analytical Jacobian matrix reads as

$$\boldsymbol{J}_A = \begin{bmatrix} \frac{\partial \boldsymbol{t}}{\partial q_1} & \frac{\partial \boldsymbol{t}}{\partial q_2} & \cdots & \frac{\partial \boldsymbol{t}}{\partial q_7} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial \boldsymbol{L}}{\partial q_1} & \frac{\partial \boldsymbol{L}}{\partial q_2} & \cdots & \frac{\partial \boldsymbol{L}}{\partial q_7} \end{bmatrix}. \tag{2.23}$$

In contrast, the **geometric Jacobian** $\boldsymbol{J}_G$ maps the generalized velocities $\dot{\mathbf{q}}$ to the end-effector velocities $\boldsymbol{w}_e$ as

$$\boldsymbol{w}_e = \begin{bmatrix} \boldsymbol{v}_e \\ \boldsymbol{\omega}_e \end{bmatrix} = \boldsymbol{J}_G(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{2.24}$$

where $\boldsymbol{v}_e$ denotes the linear velocity vector and $\boldsymbol{\omega}_e$ the angular velocity vector.

The matrix $\boldsymbol{J}_G \in R^{6\times n}$ is defined as

$$\boldsymbol{J}_G = \begin{bmatrix} \boldsymbol{j}_{P,1} & \boldsymbol{j}_{P,2} & \cdots & \boldsymbol{j}_{P,n} \\ & \cdots & \cdots & \\ \boldsymbol{j}_{O,1} & \boldsymbol{j}_{O,2} & \cdots & \boldsymbol{j}_{O,n} \end{bmatrix}, \tag{2.25}$$

with the vectors $\boldsymbol{j}_{P,i} \in \mathbb{R}^{3\times1}$, $\boldsymbol{j}_{O,i} \in \mathbb{R}^{3\times1}$, $i$= 1...$n$, and $n$ as the number of joints, so that the $i^{th}$ column of the matrix depends on the $i^{th}$ joint type.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

To express $\boldsymbol{j}_{P,i}$ and $\boldsymbol{j}_{O,i}$, let $\boldsymbol{z}_0$ be $\boldsymbol{z}_0 = [0\ 0\ 1]^T$ and $\boldsymbol{z}_i$ denoting the first three elements of the third column of the transformation matrix $\boldsymbol{T}_0^i$. Furthermore, let $\boldsymbol{t}_{0,E}$ be the forward kinematics for the end-effector position relative to the base frame, while $\boldsymbol{t}_{0,i-1}$ is the partial forward kinematics in the base frame position $\boldsymbol{O}_{i-1}$. These values can be obtained from the corresponding transformation matrix.

Then the $i^{th}$ column of $\boldsymbol{J}_G$ can be calculated as

$$\begin{bmatrix} \boldsymbol{j}_{P,i} \\ \boldsymbol{j}_{O,i} \end{bmatrix} = \begin{bmatrix} \boldsymbol{z}_{i-1} \times (\boldsymbol{t}_{i-1,E}) \\ \boldsymbol{z}_{i-1} \end{bmatrix}. \tag{2.26}$$

For revolute joints, $\boldsymbol{t}_{i-1,E}$ is calculated as

$$\boldsymbol{t}_{i-1,E} = \boldsymbol{t}_{0,E} - \boldsymbol{t}_{0,i-1}. \tag{2.27}$$

As the modeled *KUKA LBR iiwa 820* has seven revolute joints and zero prismatic joints, the geometric Jacobian $\boldsymbol{J}_7^0(\boldsymbol{q})$ be determined as

$$\boldsymbol{J}_7^0(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{z}_0 \times (\boldsymbol{t}_{0,E}) & \boldsymbol{z}_1 \times (\boldsymbol{t}_{1,E}) & \dots & \boldsymbol{z}_6 \times (\boldsymbol{t}_{6,E}) \\ \boldsymbol{z}_0 & \boldsymbol{z}_1 & \dots & \boldsymbol{z}_6 \end{bmatrix}. \tag{2.28}$$

The symbolic geometric Jacobian $\boldsymbol{J}_7^0(\boldsymbol{q}) \in \mathbb{R}^{6\times 7}$ is derived and provided in Appendix A2. Due to the highly nonlinear matrix elements, the resulting symbolic expression is complex. Despite this complexity, using the symbolic geometric Jacobian, like the transformation matrix, is essential for saving computational resources.

### 2.1.9 The inverse Jacobian

For a robot to follow a desired end-effector path, the controller needs to compute the inverse position and velocity kinematics. However, singularities can cause this calculation to fail, causing the kinematics and dynamics of the rigid-body structure to become indeterminate [24]. Singularities are more likely to occur when the path is specified in Cartesian space instead of joint space, leading to the loss of one or more degrees of freedom in the manipulator [32].

When solving the inverse kinematics problem, it is necessary to find the inverse of the Jacobian matrix. For $\boldsymbol{J}_A$, the inverse calculation of (2.21) can be expressed as

$$\dot{\boldsymbol{q}} = \boldsymbol{J}_A^{-1}(\boldsymbol{q})\dot{\boldsymbol{x}}_e. \tag{2.29}$$

Similarly, in case of $\boldsymbol{J}_G$ and (2.24), the equation reads as

$$\dot{\boldsymbol{q}} = \boldsymbol{J}_G^{-1}(\boldsymbol{q})\boldsymbol{w}_e, \tag{2.30}$$

If the Jacobian is an $n \times n$ nonsingular matrix, the inverse matrix exists, and the desired end-effector velocities can immediately be mapped into their joint-space counterparts.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

When the Jacobian matrix is not square, a pseudo-inverse matrix can be used instead. One of the most common forms of the pseudoinverse is the *Moore-Penrose pseudoinverse*, which guarantees that, while trying to match the desired velocity, the joint movement is minimal [59] [60]. Now, the pseudoinverse $\boldsymbol{J}_A^+$ of the real matrix $\boldsymbol{J}_A$ can be expressed as

$$\boldsymbol{J}_A^+ = (\boldsymbol{J}_A^T \boldsymbol{J}_A)^{-1} \boldsymbol{J}_A^T. \tag{2.31}$$

Singularities occur when the determinant of $(\boldsymbol{J}_A^T \boldsymbol{J}_A)$ equals zero [61].

With the help of (2.31), the velocity mapping from Cartesian space to joint space calculates as

$$\dot{\boldsymbol{q}} = \boldsymbol{J}_A^+(\boldsymbol{q}) \dot{\boldsymbol{x}}_e. \tag{2.32}$$

The same reasoning applies to the geometric Jacobian $\boldsymbol{J}_G$, and the approach can also be extended to this case.

To evaluate the Jacobian matrix's ill-conditionedness, the *Singular Value Decomposition* (SVD) [55] is a useful tool that detects discontinuities and instability near singularities and from which the pseudo-inverse matrix can be computed. Subsection 3.2.1 will explain the SVD in detail. Analyzing the singularities enables evaluating the manipulator's configuration and how it affects its degrees of freedom [32].

### 2.1.10 Singularity analysis

Singularity analysis is crucial for robot control and manipulation, as it aims to identify when singular configurations occur and how to avoid them. The types and complexity of singularities depend on the number, type, and arrangement of joints, and they fall into two categories: *internal singularities* and *boundary singularities* [62].

*Boundary singularities* occur when a robot reaches its joint limits, decreasing the number of independent columns in the Jacobian matrix and a singular configuration. *Internal singularities* occur when links become collinear, resulting in a reduced rank in the kinematic Jacobian matrix.

Redundant manipulators may approach singular configurations where the Jacobian matrix loses rank, and its pseudoinverse becomes singular. This poses safety risks, especially for large industrial robots, as small Cartesian space velocities require high joint-space velocities, reducing trajectory accuracy and requiring high torques to maintain control. When a joint reaches its limits, an immediate rank reduction of the Jacobian follows, necessitating limits on joint-space velocities to mitigate risk. To prevent boundary singularities, joint limits can be monitored, and the robot's range of motion can be constrained.

Joint limits impose a natural constraint on achievable movements, e.g., when the manipulator becomes fully stretched in one direction. Scott B. Nokleby and Ron P. Podhorodeski [61] [63] proposed a methodology for analyzing singularities in the case of redundant manipulators. This method involves selecting linearly independent joint screws and constructing a submatrix from the Jacobian matrix. Nevertheless, Waldron et al. [64] recommend changing the reference frame when calculating the Jacobian matrix to simplify the matrix elements.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

In this thesis, to investigate singular configurations of the *KUKA LBR iiwa R820*, the method and results presented by Beck et al. [65] and Zhang et al. [66] are used, and Boudreau and Podhorodeski [67] and the KUKA documentation [68] are used for comparison. Accordingly, the resulting singularity conditions can be summed up as

1. $\theta_2 = 0°$ and $\theta_3 = \pm 90°$ as internal singularity,
2. $\theta_4 = 0°$ as boundary singularity,
3. $\theta_2 = 0°$ and $\theta_6 = 0°$ as internal singularity,
4. $\theta_5 = \pm 90°$ and $\theta_6 = 0°$ as internal singularity.

Figure 2.11 demonstrates the abovementioned configurations.

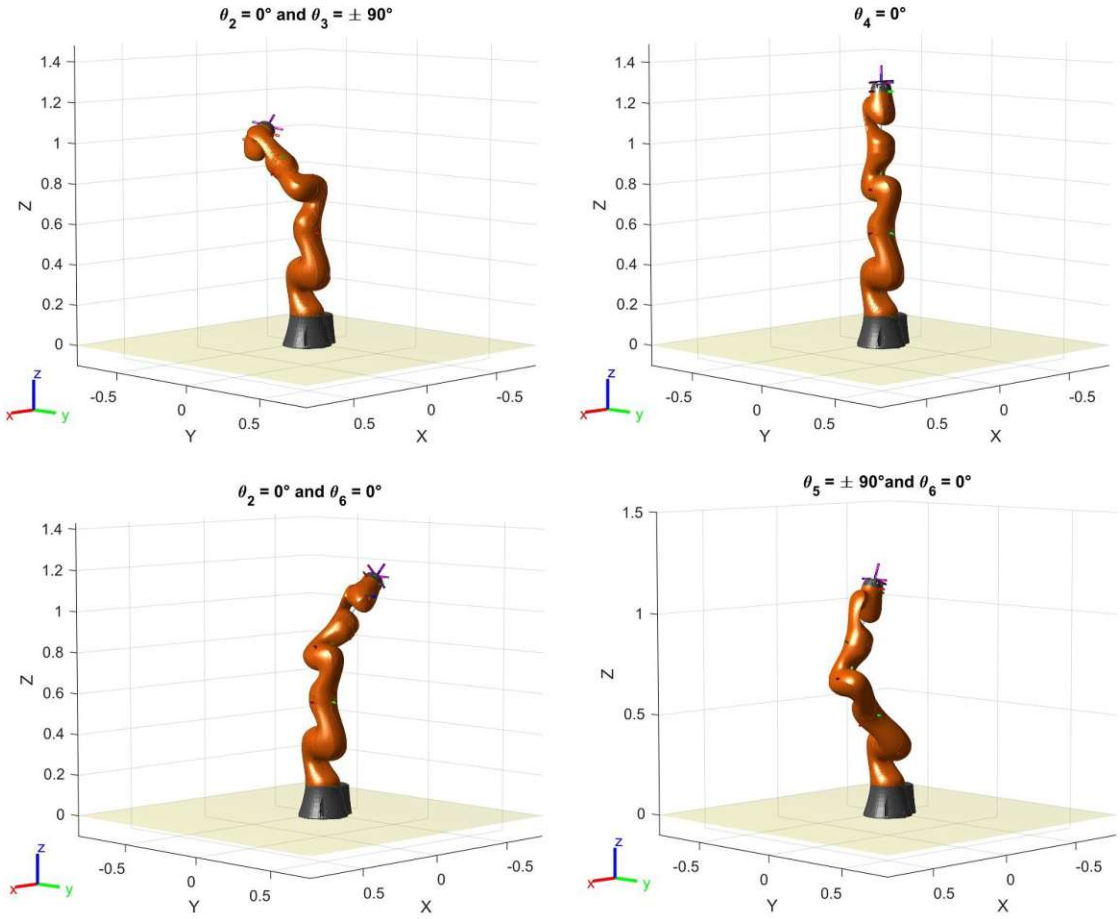


Figure 2.11: *KUKA LBR iiwa R820* singular configurations.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

## 2.2 Differential drive kinematics

This section is based on the theory presented in *Modern Robotics: Mechanics, Planning, and Control* by Lynch K. and Park F. [24].

Differential drives are considered one of the simpler drive configurations for wheeled mobile robots [24]. Typically, they are used in small, inexpensive robots for indoor tasks. This type of drive system includes two driven wheels mounted on collinear axes, each controlled by a separate drive unit. By varying the relative velocity of the wheels, the robot can turn by modifying its point of rotation.

The *Sally V2.0* differential drive mobile platform has a pair of driven wheels aligned on the same axis and a non-motorized wheel that supports the system [49].

### 2.2.1 Kinematic model

To derive the kinematic equations, mobile platforms are typically modeled as two-dimensional structures.

Let the global reference frame ($\mathbf{O}_G$, $X_G$, $Y_G$) be denoted by $\mathcal{G}$ and a local reference frame ($\mathbf{O}_L$, $X_L$, $Y_L$) as $\mathcal{L}$ attached to a rigid body on a two-dimensional horizontal surface. Furthermore, let the origin $\mathbf{O}_L$ of $\mathcal{L}$ be the axis-midpoint of the driven wheels and let the $X_L$ axis always be parallel to the forward direction of the chassis [53].

Then, the rigid body has three degrees of freedom (DOF): two independent DOF for the position ($x_l$, $y_l$) in $\mathcal{G}$ and one for the orientation (the angle $\theta_l$) along the vertical axis $X_G$. Here $\theta_l$ denotes the heading angle, the angular difference between $\mathcal{G}$ and $\mathcal{L}$, as the velocity vector always points in the forward direction of the robot.

The pose of $\mathcal{L}$ can be expressed as a vector $\boldsymbol{\eta}$ in $\mathcal{G}$ as

$$\boldsymbol{\eta} = \begin{bmatrix} x_l & y_l & \theta_l \end{bmatrix}^T. \tag{2.33}$$

These three DOF are sufficient to describe any pose of the body in a 2D global reference frame, see Figure 2.12.

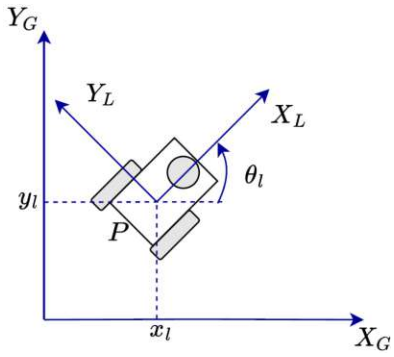


Figure 2.12: Global and local coordinate frames for differential drive platform.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Mapping the orientation between $\mathcal{L}$ and $\mathcal{G}$ can be done by an orthogonal rotation matrix $\mathbf{R}(\theta)$

$$\boldsymbol{R}(\theta_l) = \begin{bmatrix} \cos(\theta_l) & \sin(\theta_l) & 0 \\ -\sin(\theta_l) & \cos(\theta_l) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{2.34}$$

Differential-drive robots are commonly considered highly maneuverable. The robot's *differentiable degrees of freedom (DDOF)* refer to the number of independent parameters that describe the motion of a mobile platform. They determine the directions in which the robot can move, and the trajectories can be traversed.

The *Sally V2.0* mobile platform has two DDOF. The robot can move in the $X_L$ direction and rotate $\mathcal{L}$ relative to $\mathcal{G}$. However, it is unable to move instantaneously in the $Y_L$ direction. Any pose in $\mathcal{G}$ can be reached, whereby all movements in a particular direction require at least a translational motion.

Furthermore, some simplifications and assumptions are necessary to simplify the calculations. It is assumed that:

- the plane of a given wheel is continuously perpendicular to the ground,
- a point on the wheel is in constant contact with the ground (the robot has ideal suspension),
- wheels do not slip at the point of contact, e.g., the relative speed there is zero (nonholonomic constraint),
- internal degrees of freedom are not taken into account,
- the wheels have fixed positions on the chassis, and as such, the geometry of the chassis does not change over time.

### 2.2.2 Derivation of forward kinematics equations

This section presents the nonlinear kinematic equations of the nonholonomic mobile platform as outlined in [24] and [53].

The geometrical parameters of the model are the radius $r$ of the driven wheels and the axis length $2d$, which is the distance of the parallel planes of rotation of the wheels. The mobile platform is described by velocities, where $u_R$ denotes the angular and $v_R$ the linear velocity of the right wheel and $u_L$ the angular and $v_L$ the linear velocity of the left wheel. Then the time-continuous model of $\dot{\boldsymbol{\eta}}$ can be expressed as

$$\dot{\boldsymbol{\eta}} = \begin{bmatrix} \dot{x}_l \\ \dot{y}_l \\ \dot{\theta}_l \end{bmatrix} = f(d,\, r,\, \theta_l,\, u_R,\, u_L) = \begin{bmatrix} \frac{r}{2}\cos(\theta_l) & \frac{r}{2}\cos(\theta_l) \\ \frac{r}{2}\sin(\theta_l) & \frac{r}{2}\sin(\theta_l) \\ -\frac{r}{2d} & \frac{r}{2d} \end{bmatrix} \begin{bmatrix} u_L \\ u_R \end{bmatrix}. \tag{2.35}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

A single standard wheel cannot have a lateral motion along its horizontal axis. This axis is called the *zero motion line*. Figure 2.13 depicts when the local $X_L$ axis shows in the direction of motion, and the local $Y_L$ axis represents the zero motion line.

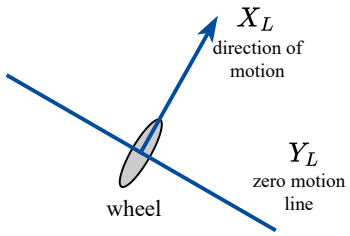


Figure 2.13: A wheel and its zero motion line, with $Y_L$ representing the zero motion line, while the $X_L$ shows in the direction of motion.

Two steerable wheels mounted on the same axis share a common zero motion line. To change the heading angle $\theta_l$, the robot must rotate around a point located along the common axis of the left and right wheels, which forms a circle of radius $R$. The origin of this circle, which lies along the zero motion line, is the *ICC - Instantaneous Center of Curvature* (ICC), also called as *Instantaneous Center of Rotation* (ICR), see Figure 2.14. The rate of rotation around ICC is the angular velocity $\omega(t)$.

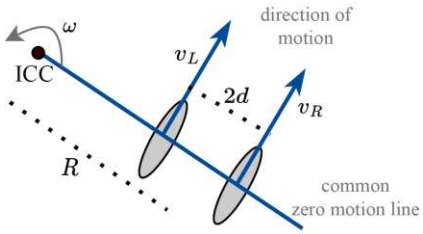


Figure 2.14: Differential drive wheels on the same axis sharing a common zero motion line, rotating around ICC with the radius $R$ and the angular velocity $\omega(t)$.

A differential-drive robot is controlled by the velocities $v_L(t)$ and $v_R(t)$ of its two separately driven wheels. The common velocity $v(t)$ reads as

$$v(t) = \frac{v_R(t) + v_L(t)}{2}, \tag{2.36}$$

and the angular velocity $\omega(t)$ of the platform can be expressed as

$$\omega(t) = \frac{v_R(t) - v_L(t)}{2d}. \tag{2.37}$$

The radius $R$ can be calculated as the distance between the mid-point of the differential-drive axis and the ICC

$$R = \frac{2d}{2}\frac{v_L + v_R}{v_R - v_L}. \tag{2.38}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

As the rate of rotation $\omega(t)$ around the ICC must be equal for both wheels, the following equations hold

$$v_R(t) = \omega(t)\,(R + d)\,, \tag{2.39a}$$
$$v_L(t) = \omega(t)\,(R - d)\,. \tag{2.39b}$$

The ICC constrains wheels on the same chassis, as their zero motion lines must intersect at the same center of rotation, and they move with the same angular velocity $\omega(t)$, see Figure 2.15.

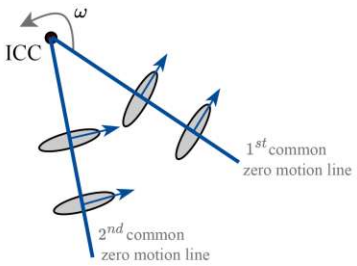


Figure 2.15: Two pairs of rotating wheels on the same chassis sharing the same center of rotation.

The midpoint of the rear wheel axis can be represented by $x_l(t)$ and $y_l(t)$ in Cartesian space so that the ICC can be expressed as

$$ICC = [x_l - R\sin(\theta_l), y_l + R\cos(\theta_l)]. \tag{2.40}$$

Depending on the wheel velocities, different maneuvers are possible. In case that $v_L = v_R \neq 0$, then $R = \infty$, and $\omega = 0$, the wheels are driven in the same direction at the same speed, and the robot moves forward along a straight line. If one of the wheels slows down, the robot turns in the direction of the decelerated wheel. In particular, when one of the wheel velocities equals zero, the ICC coincides with that wheel. If the wheels are rotated in opposite directions at the same speed $v_L = -v_R$, the robot rotates in place around the midpoint of the wheel axis and $R = 0$. The sharpness of the turn depends on the difference in speed between the rotations of the two wheels. Furthermore, as the robot turns around the ICC, the heading angle and the position change.

Using the ICC, the new position can be calculated using a rotation matrix from (2.34).

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

## 2.3 System coupling

The manipulator and the mobile platform models can be combined to represent the full robot kinematics. The resulting model provides a comprehensive understanding of the robot's movements and positioning, considering both the platform's and the manipulator's movements.

The merged model has nine degrees of freedom, seven coming from the manipulator and two from the platform. The resulting redundancy means additional three degrees of freedom for the system to solve tasks in an $SE(3)$ task space. The platform's role is to increase the robot's workspace by advancing and changing the pose of the manipulator's base.

The arm is mounted at a 45° angle and a height of 685 mm on the platform, as illustrated in Figure 2.1. The movement of the platform and the arm mounting can be represented by the transformation matrices $\boldsymbol{T}_{\mathbf{sally}}$ and $\boldsymbol{T}_{\mathbf{mount}}$. The transformation matrix $\boldsymbol{T}_{\mathbf{sally}}$ reads as

$$\boldsymbol{T}_{\mathbf{sally}} = \begin{bmatrix} \cos(\theta_C) & -\sin(\theta_C) & 0 & x_C \\ \sin(\theta_C) & \cos(\theta_C) & 0 & y_C \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2.41}$$

The arm mounting is represented by a constant matrix $\boldsymbol{T}_{\mathbf{mount}}$ in the form

$$\boldsymbol{T}_{\mathbf{mount}} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ -0.7071 & -0.7071 & 0 & 0 \\ 0.7071 & -0.7071 & 0 & 0.685 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2.42}$$

Now, $\boldsymbol{T}_{\mathbf{sally}}$ and $\boldsymbol{T}_{\mathbf{mount}}$ can be multiplied with the manipulator's transformation matrix $\boldsymbol{T}_{\mathbf{0}}^{\mathbf{7}}$ from (2.16) to obtain the transformation matrix $\boldsymbol{T}_{\mathbf{final}}$ for the combined system

$$\boldsymbol{T}_{\mathbf{final}} = \boldsymbol{T}_{\mathbf{sally}}\ \boldsymbol{T}_{\mathbf{mount}}\ \boldsymbol{T}_{\mathbf{0}}^{\mathbf{7}}. \tag{2.43}$$

When determining a robotic system's Jacobian matrix and manipulability, it is common to consider the entire system. However, this work prioritizes the arm's maneuverability and thus only considers the Jacobian matrix of the *KUKA LBR iiwa R820*. As a result, the manipulability and the Jacobian of the mobile manipulator depend only on the manipulator's configuration and are not affected by the movements of the mobile platform.

A simulation tool was developed in MATLAB to understand better and visualize the robot's movements, as the MATLAB Robotics Systems Toolbox offers a rigid-body model representation using the D-H convention. However, it locks the manipulator models at the origin of the global coordinate system.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

To replicate the robot's unrestricted behavior, the differential drive was mapped to a rigid-body system, a simple revolute-prismatic-revolute rigid-body tree. The first two revolute-prismatic joints reenact the position that the differential drive can achieve as the revolute joint rotates the platform and the prismatic joint maintains it at the according distance from the map's center. The second revolute joint simulates the platform's heading angle.

This approach enabled the visualization of the robot's movements using the Toolbox, as shown in Figure 2.16.

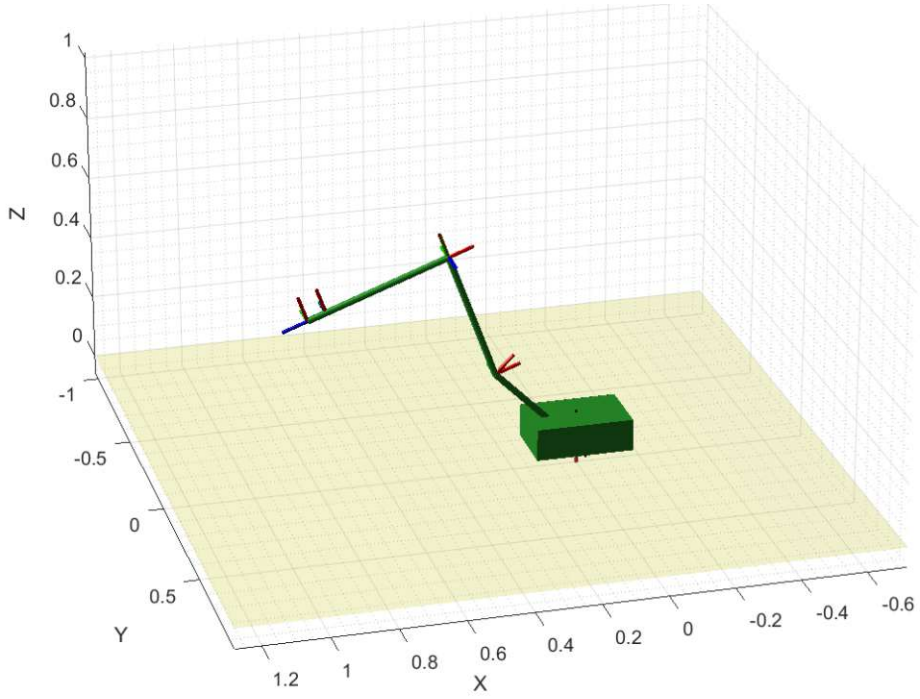


Figure 2.16: Visualization of the implemented robot model in MATLAB 2021.

In the following chapter, the workspace of the robot will be thoroughly defined and analyzed using the established model. Furthermore, calculating specific performance indices will enhance the validity and accuracy of the results obtained.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

# Chapter 3

# Workspace and manipulability

The robot's workspace will be studied in this chapter based on the previously developed model in Section 2.3. Furthermore, the model will be used to compute and study several performance metrics, particularly the manipulability index.

The collection of all positions that the robot's end-effector can reach using any combination of joint angles is known as the manipulator's *workspace*. The size and shape of the workspace depend on the geometry, kinematics, joint restrictions, and designated mounting point of the manipulator.

Mobile manipulators, carried by a wheeled platform, offer a wider range of applications due to their ability to access different environments, compared to regular robotic arms with a stationary base. To evaluate and optimize their capabilities in an objective manner, the use of performance indices is essential.

The *scope* of performance measures can be divided into local and global indices, see, e.g., [28]. *Local indices* are configuration-dependent indicators of the local properties of the robot that change depending on the configuration. *Global indices* have a generic, single value for the robot over the entire workspace and are independent of the configuration. In most cases, they are expanded versions of local indices.

Local indices like the condition number [28] and the manipulability index [36] are based on the Jacobian matrix because they depend on the robot's current configuration. An example of a global index is the Global Conditioning Index (GCI) [28], which assesses the overall performance.

The three subgroups of *performance characteristics* are kinematic, dynamic, and simple indices [28]. *Kinematic indices*, which assess the kinematic behavior, depend on the structure of the robot and, thus, are based on the Jacobian matrix. *Dynamic indices* describe dynamic features and rely heavily on inertial traits. *Simple indices*, like the service angle [30] and the dexterity index [69], do not fit into either group and form their own.

Depending on the *application*, indices are distinguished between intrinsic and extrinsic indices [28]. *Intrinsic indices* are inherent traits and are task-independent, such as condition number, manipulability, and dexterity. *Extrinsic indices* quantify how well the robot can solve a given task, for example, the power manipulability index [28] or the robot-task conformance index [28].



Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

As shown earlier, it is crucial to understand how freely the end-effector can move during task performance. This can be evaluated using a local kinematic intrinsic index, the manipulability index. It is common practice to use manipulability indices to provide a quantifiable measurement for modifying the movement and applying forces in any direction. They can be obtained from the robot's kinematics and further optimized by imposing various constraints, providing insights into force and motion transmission. The manipulability ellipsoid and manipulability measurements must be introduced to analyze this performance aspect from a kinematics approach. Therefore, this chapter will strongly focus on this performance index.

## 3.1 Workspace of the manipulator

Knowing the workspace's quality and geometry helps gain a better understanding of the manipulator's performance. The workplace is often not homogeneous since the robot may access certain areas quickly while others may take longer. As a result, determining the boundary surfaces and consequently analyzing the various sections is critical.

In order to obtain information about the workspace, two methods are commonly used, as outlined in [70]: analytical and numerical methods.

*Analytical methods* involve closed-form descriptions of the workspace boundary surfaces and rely on kinematic equations, but can be highly complex due to nonlinearities and matrix inversions. Furthermore, their use is limited to specific types of manipulators. On the other hand, *numerical methods* are more versatile and can be adapted to a wider range of manipulators. However, they solely provide approximate boundary surfaces and are less accurate than the boundaries established by analytical approaches.

The workspace is expected to have a concave volume and the shape of a sphere with an interior cavity. This cavity depicts the restricted area where the manipulator is located, hence is inaccessible to it.

Section 3.1 aims to define a representation encompassing both the volume and the boundary surfaces. After defining the workspace, it is subdivided into two subspaces: a reachable workspace, which consists of the points that the end-effector can access, and a dexterous workspace, which includes all the locations that are not only reachable but also allow the end-effector to have any orientation. The reachable workspace can be stored offline, which speeds up online tasks like path planning. Adding quality measurements to the workspace data makes it possible to determine which areas are easier to manipulate than others and can further improve performance.

### 3.1.1 Monte Carlo method

The Monte Carlo method, as shown by Rastegar and Perel [71], is a widely used and straightforward technique for estimating the manipulator's workspace. It is a numerical, random sampling method that uses forward kinematics to generate the workspace volume and boundary surfaces as it generates uniformly distributed configurations at random samples.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

The Monte Carlo method is easy to apply and works well for most manipulators, which makes it particularly suitable for robots with kinematic redundancy. It generates most sampling points within the workspace and fewer on the boundary surfaces. The method's drawback is that, like all numerical methods, it yields an inaccurate workspace, which might complicate motion planning.

The method solely considers the manipulator's forward kinematics solution. The joint values are randomly distributed between their limits, resulting in a pose of the end-effector, which is then translated into a Cartesian coordinate point with the help of the forward kinematics. The approach can generate an arbitrary number of reachable end-effector points. The higher the number of created points, the higher the resolution of the workspace. Figure 3.1 shows the results of generating point sets with an increasing number of data points.

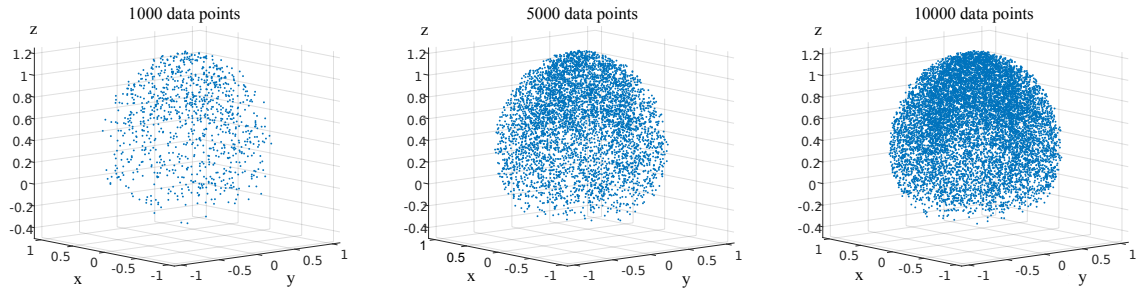


Figure 3.1: Workspace point sets generated by the Monte Carlo method with an increasing number of data points.

A point set representation of the workspace is insufficient as it lacks continuity, and testing every possible joint arrangement to determine every end-effector configuration is not feasible. In order to address this issue, the acquired data must be refined and smoothed, which helps to accurately describe the reachable volume and obtain the workspace's boundary surfaces.

Over the years, researchers presented various approaches to depict the manipulator's workspace, including *voxel-based estimation* [72] and the *alpha-shape method* [73], both of which will be discussed in this work.

### 3.1.2 Voxelization

A common strategy for improving the workspace representation is to discretize the workspace and tessellate it into a grid of uniform cuboids. This technique is known as voxelization [72], which is a time-efficient approach to model and represent volumetric data.

Voxels are small, three-dimensional volumes, splitting up the space into small, uniform components. The required resolution determines the size of the cuboids. Each point of the point set generated by the Monte Carlo method gets assigned to a cell of the grid.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

The approach consists of the following steps:

1. Creating a seed workspace by generating a point set with the Monte Carlo method (e.g., 1000 points).
2. Enveloping the full point cloud by a cuboid (maximum and minimum values of the points on all three coordinate axes give the side lengths), which then will be used for further tessellation of the workspace.
3. Discretizing the enveloping cuboid by dividing it into uniform cells. The cuboids' size depends on the desired resolution, as shown in Figure 3.2. The workspace can be modeled more accurately with higher resolution and more data points.

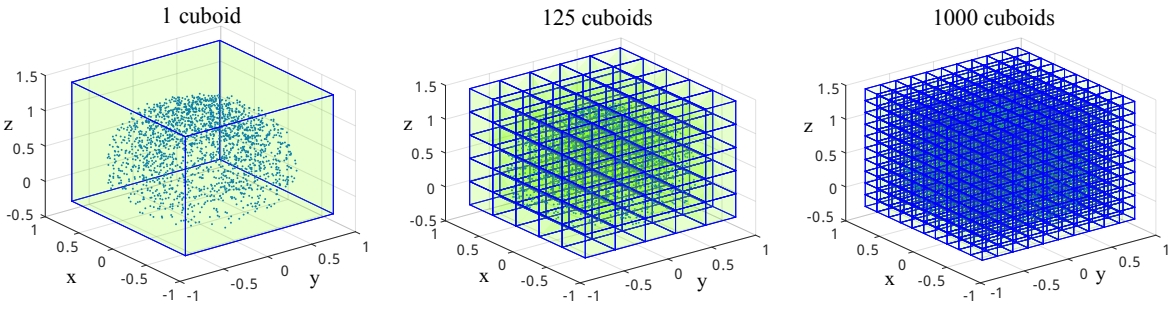


Figure 3.2: The enveloping cuboid of the workspace is tessellated by different resolutions depending on the size of the cuboids.

4. Equipping the voxels with the binary hit/no-hit variable. Only the elements containing at least one Monte Carlo data point are assigned a value of 1 and subsequently displayed. The cells without any data points are given a value of 0. When the point set is large, it is common for multiple points to be located within the same cell, depending on the chosen resolution.

   Figure 3.3 displays a 100-point Monte Carlo point cloud, and the unique cuboids containing at least one data point are well visible.

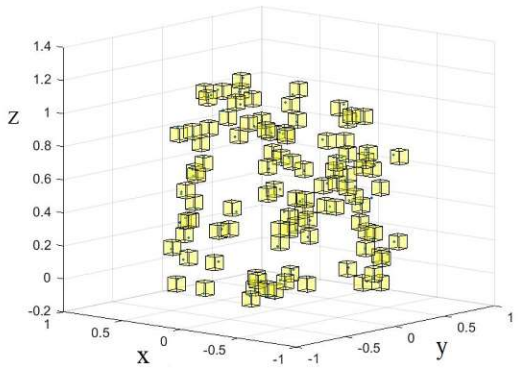


Figure 3.3: A Monte Carlo point cloud with 100 data points, showing the unique cuboids containing at least one data point.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

As outlined in [74], modeling the workspace at a higher resolution can produce remarkably accurate results but requires more computational resources. Current technology is optimized to render polygons, and there is no specialized hardware for rendering high-resolution voxels adequately. Therefore, voxels require significantly more memory in comparison to polygons.

### 3.1.3 Alpha Shape method

The alpha shape method is a different way to represent and organize unstructured data elements by taking advantage of the tessellation of a point cloud [75]. The purpose of this approach, which is based on the *Delaunay triangulation algorithm* [73] [75], is to generate a bounding hull that contours the seed point cloud. In terms of the $\alpha$-shape method, the underlying triangulation generates a mesh of triangles covering the majority of the data points.

The Delaunay triangulation of a finite point set $\mathcal{S} \in \mathbb{R}^2$ means that the circumcircle or for $\mathcal{S} \in \mathbb{R}^3$ the circumsphere of every triangle is empty so that neither of the triangles has points in their interior. The algorithm examines if, in case of $\mathcal{S} \in \mathbb{R}^2$ a circle, or in case of $\mathcal{S} \in \mathbb{R}^3$ a sphere, which crosses three points, includes any other data point. If so, the triangle is rejected. If not, the triangle is accepted.

This indicates that the triangles are larger in areas with few data points, whereas the triangles are considerably smaller in densely populated regions. The drawback of this approach is having too large triangles, resulting in a hull that is significantly larger than the actual object.

Figure 3.4 compares a Delaunay triangulation to a non-Delaunay triangulation and demonstrates that the Delaunay algorithm does not have uniformly sized triangles.

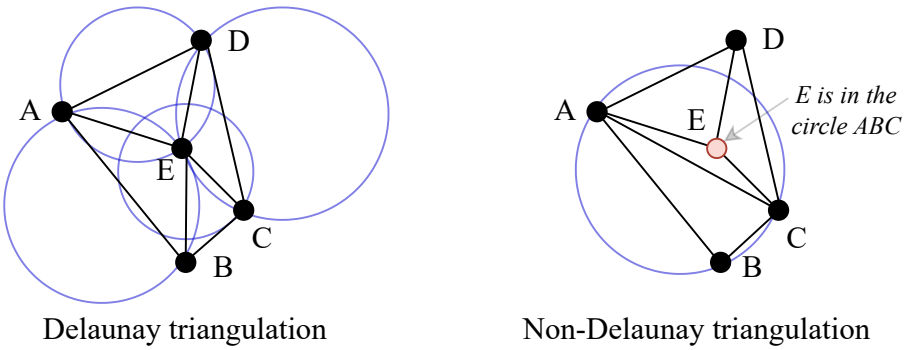


Figure 3.4: A Delaunay triangulation compared to a non-Delaunay triangulation.

The $\alpha$-shape approach is frequently used to generate a nonconvex hull for point clouds [73] [75]. Even though the Delaunay triangulation serves as the foundation, it is enhanced by an additional parameter, $\alpha$, which is bounded between 0 and $\infty$. This $\alpha$ serves as the predefined upper limit for the radius of any circle through the three vertices of the triangle. This constraint requires all radii to be equal to or less than this value. If $\alpha$ equals $\infty$, the contouring is convex. If not, the shape degenerates as $\alpha$ decreases.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

The steps of the $\alpha$-shape approach are as follows:

1. **Generation of the seed point cloud.**
   The smoothness of the resulting alpha shape and the value of the $\alpha$ parameter depend on the size of the point cloud, which is generated by the Monte Carlo method. Increasing the dataset provides better resolution and allows for improved visualization of the workspace. As an initial attempt, a point cloud size of 10000 data points is used.
2. **Hull construction using the Delaunay triangulation.**
   The resulting polyhedron envelopes the whole set of the Monte Carlo points. Furthermore, the hull can be better observed when the point cloud is set to invisible, see Figure 3.5.

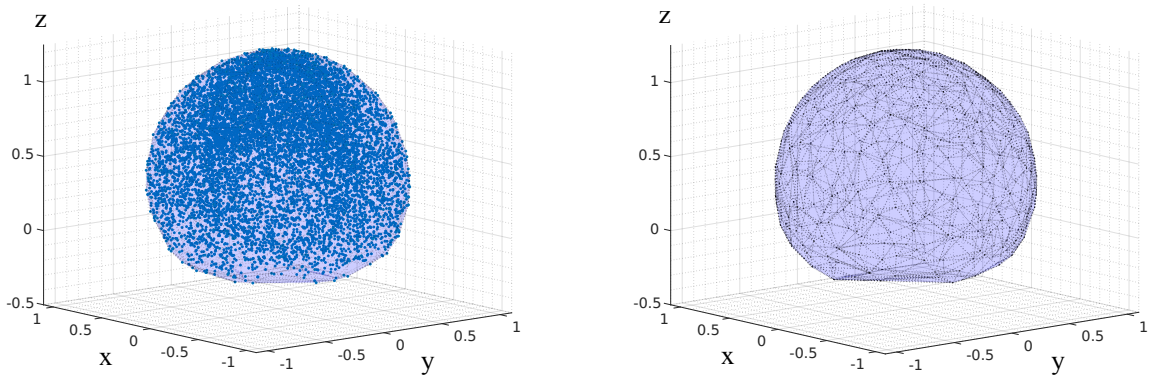


Figure 3.5: Polyhedron enveloping Monte Carlo points using Delaunay triangulation.

3. **Optimizing the $\alpha$ parameter for a more precise shape.**
   The alpha shape can be adjusted using the $\alpha$ parameter for a better fit by either tightening or loosening the volume. To accurately capture the features of a workplace, such as dents and holes, the alpha shape needs to be able to represent inward lines, which makes convex contouring unsuitable.

There is a correlation between the chosen $\alpha$ and the number of data points of the point cloud. Depending on the selected $\alpha$, it produces a different shape space. The larger the cloud, the smaller the $\alpha$ can be chosen while maintaining a coherent structure.

Initially, the value of $\alpha$ is set to the default $\alpha = \infty$, which results in a convex shape without any cavity. To reveal a central void in the shape, $\alpha$ had to be decreased. However, any value above $\alpha > 0.4$ still resulted in a convex hull, necessitating further reduction of $\alpha$.

The shape starts to show visible changes once $\alpha < 0.4$. At $\alpha = 0.35$, a small cavity is revealed, which expands further with decreasing $\alpha$. At $\alpha = 0.18$, a noticeably larger cavity is observed. Below $\alpha < 0.1$, the shape starts to disintegrate into smaller pieces, and at $\alpha = 0.05$, it is merely a large number of small polyhedrons.

The resulting alpha shapes for various values of $\alpha$ are presented in Figure 3.6, illustrating the changes in the shape as the value of $\alpha$ is decreased.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

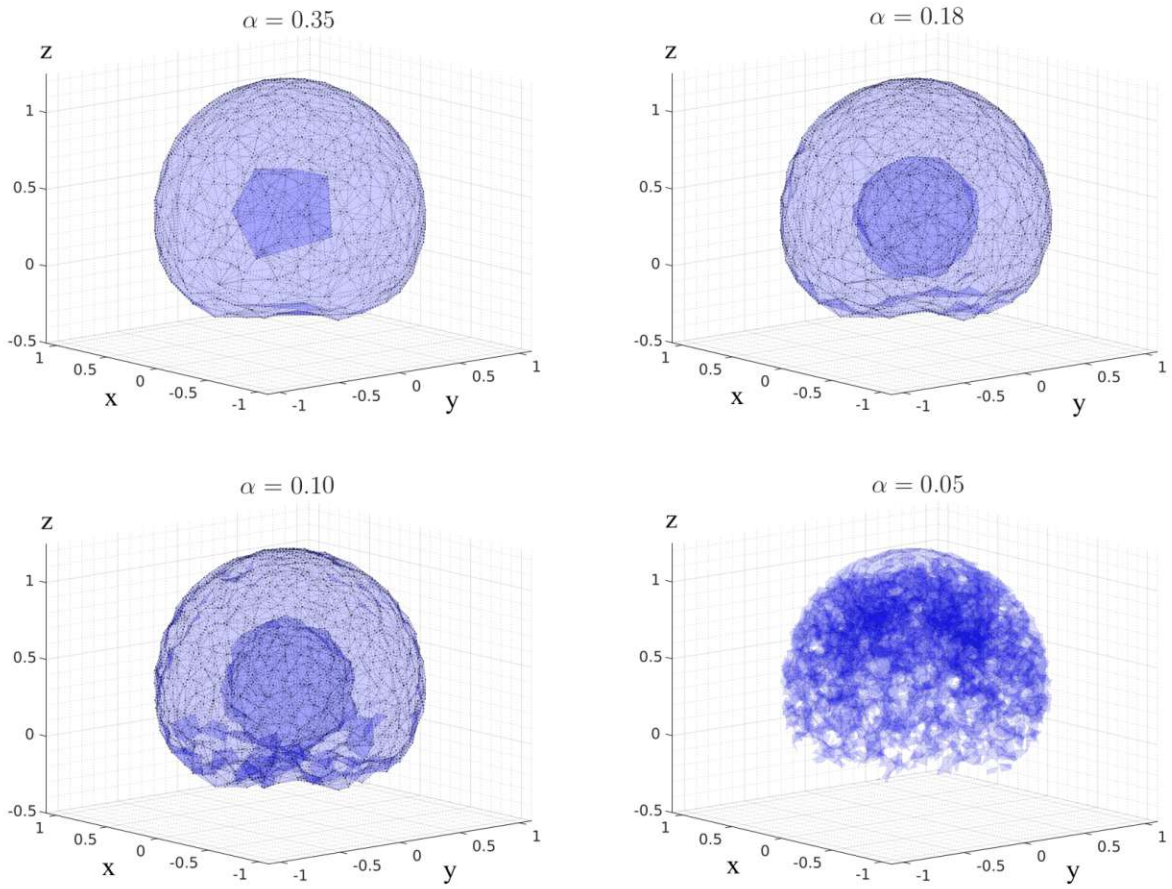


Figure 3.6: Illustration of alpha shapes for varying $\alpha$, demonstrating the changes in the shape as $\alpha$ decreases.

As it is case specific, the optimal value of $\alpha$ has to be determined empirically depending on the data size by adjusting it. The point cloud has no predefined shape and may feature inward bendings or caverns, making the experimental adaptation of $\alpha$ necessary.

The challenge is to find an optimal balance between the shape, the volume, and the surface hull. As it can be seen, for 10000 data points, an optimal $\alpha$ value is around $\alpha = 0.18$.

### 3.1.4 Workspace results

This section concludes the findings of the manipulator's workspace evaluation. The method of representation was carefully considered, and it was determined that using alpha shapes provides numerous advantages over voxels, including precision in the boundary representation, improved efficiency, better representation of curved surfaces, easier analysis, and improved visual representation. The results of using alpha shapes to represent the manipulator's workspace are shown and discussed.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

To improve the resolution of the workspace, a larger point cloud with 50000 data points was generated with the Monte Carlo method, and after empirical testing, an $\alpha$ value of $\alpha = 0.05$ was selected. Figure 3.7 shows the resulting shape and the cavity cross-section.

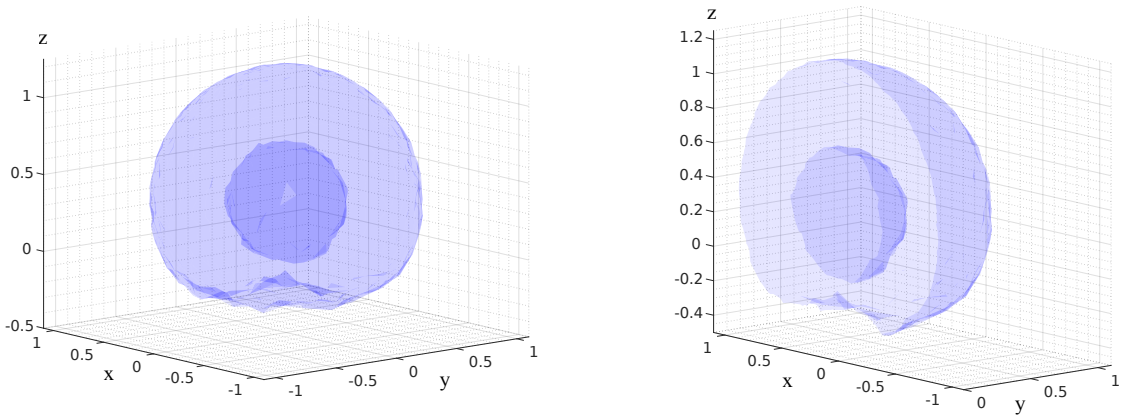


Figure 3.7: Alpha shape and cross-section of the workspace generated from a point cloud of 50000 data points using an $\alpha = 0.05$.

As the manipulator is mounted on the mobile platform, the workspace has a vertical offset and is tilted, as depicted in Figure 3.8.

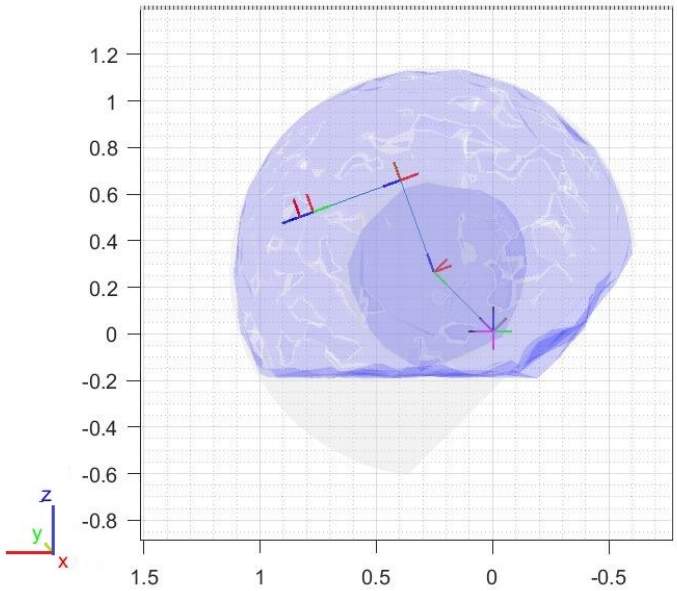


Figure 3.8: Modified workspace of *KUKA LBR iiwa R820* considering the vertical offset and the mounting angle on the mobile platform.

The workspace is now adequately represented, including the volume and boundary surfaces, and the workspace data can be efficiently computed and saved. This enables further analysis, such as evaluating performance indices, e.g., the manipulability index.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

## 3.2 Manipulability measures

Manipulability refers to the robot's dexterity [53] and ability to generate Cartesian velocities and apply forces within its workspace [36]. It provides a quantitative measure of how well the velocity of the end-effector can be changed depending on its current pose: it indicates the relative capability to move in different directions. Furthermore, it provides information regarding the proximity to singularities. To accurately measure manipulability, mathematical approaches are used. These measures can be derived from the robot's kinematics and optimized by incorporating constraints.

There are various ways of defining manipulability, leading to various manipulability indices. Most of these indices can be related to and derived from the so-called *manipulability ellipsoid*, a concept described by Wen and Wilfinger [76].

### 3.2.1 Manipulability ellipsoids

The manipulability ellipsoid is a graphical representation of the capacity of a robot's end-effector to change its position and orientation at a given configuration. Ideally, the end-effector should have isotropic movement, meaning it should be equally capable of moving in all directions. This ability can be represented by a unit sphere in the $n$-dimensional Euclidean space. However, isotropic movement is not always possible, and the sphere may deteriorate.

The dimensions of the manipulability ellipsoid provide valuable information about the robot's possible end-effector velocities, and the shape of the ellipsoid varies depending on the robot's current configuration. However, as discussed in Subsection 2.1.10, a robot's configuration may be singular or close to being so, which can result in the sphere collapsing and being reduced to a line, thereby making the motion in one of the Cartesian directions impossible. Therefore, the manipulability ellipsoid is a valuable tool, providing insights into the manipulator's ability to execute tasks and reach various parts of its workspace.

As shown in Subsection 2.1.8, the Jacobian matrix $\mathbf{J}_G$ maps the joint velocities to Cartesian velocities. Now, to derive a generic expression for the ellipsoid of the end-effector velocity, let the manipulator be with $n$ degrees of freedom acting in an $m$ dimensional task space. Assuming that $m \leq n$, the velocity mapping from Cartesian space to joint space follows as in (2.24), see [53]. In the redundant case when $m < n$, the pseudoinverse Jacobian $\boldsymbol{J}_G^+$ is used.

Assuming that the joint velocity is within the unit sphere, i.e.

$$\|\dot{\mathbf{q}}\|_2^2 = 1, \tag{3.1}$$

this unit sphere is mapped to the Cartesian velocity space as

$$\|\dot{\mathbf{q}}\|_2^2 = \dot{\mathbf{w}}_e^T (\mathbf{J}_G^{-1})^T \mathbf{J}_G^{-1} \dot{\mathbf{w}}_e, \tag{3.2}$$

which is the expression for the ellipsoid of the end-effector velocity if $\mathbf{J}_G^{-1}$ exists. Otherwise, the pseudoinverse $\boldsymbol{J}_G^+$ must be used.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

The ellipsoids represent the characteristics of the feasible motion and are called *manipulability ellipsoids*. Their proximity to collapsing can be used to calculate how near the robot is to a singular configuration, i.e., if the ellipsoid volume is zero, the robot is in a singular configuration.

According to the *singular value decomposition* (SVD) [55], any matrix $\boldsymbol{M} \in \mathbb{R}^{m \times n}$ can be decomposed into the product of three matrices, $\boldsymbol{U}$, $\boldsymbol{D}$ and $\boldsymbol{V}$ in the form

$$\mathbf{M} = \mathbf{U}\mathbf{D}\mathbf{V}^T, \tag{3.3}$$

where $\boldsymbol{U} \in \mathbb{R}^{m \times m}$ and $\boldsymbol{V}^T \in \mathbb{R}^{n \times n}$ are orthogonal matrices. Furthermore, $\boldsymbol{D} \in \mathbb{R}^{m \times n}$ is a diagonal matrix with non-negative diagonal elements. It contains the singular values ($\sigma_1$,$\sigma_2$, .. $\sigma_m$) on the diagonal of the matrix. The matrix $\boldsymbol{M}$ is singular if at least one of these $\sigma$ values is zero. The other two matrices, $\boldsymbol{U}$ and $\boldsymbol{V}$, are the orthonormal bases for the range and the null space. The non-zero singular values correspond to given columns in the $\mathbf{U}$ matrix ($\mathbf{u}_1$, $\mathbf{u}_2$, ... $\mathbf{u}_m$).

Applying the SVD to $\mathbf{J}_G$ allows calculating the manipulability ellipsoids. The principal axes, which are the axes of symmetry of the manipulability ellipsoid, are given by $\sigma_1\mathbf{u}_1$,$\sigma_2\mathbf{u}_2$ ... $\sigma_m\mathbf{u}_m$ in an ordered way from the highest to the lowest singular value. The end-effector can move most easily in the $\sigma_1\mathbf{u}_1$ direction, which represents the major axis. In contrast, the axis of the least movement is given by $\sigma_m\mathbf{u}_m$.

In cases when the ellipsoid is a sphere, all directions are equally favorable. Furthermore, the manipulability ellipsoid's size also indicates the end-effector's movement speed, as a larger ellipsoid corresponds to a faster movement for a given joint velocity [55].

By using the manipulability ellipsoid to represent the capabilities of the end-effector's motion, it is possible to derive various manipulability measures to evaluate the degree to which the robot approaches singularity and analyze its performance further. These indices, known as manipulability measures, provide insights into the robot's capabilities and limitations [28]. Four of these measures will be introduced next.

The first measure is the *minimum singular value* [28] of the singular value decomposition, which is the length of the shortest axis. The second index is the ratio of the ellipsoid's longest to the shortest axis, known as the *condition number* [28]. These two indices provide information about the stability of the manipulator's motion and the shape of the manipulability ellipsoid. The third measure is the *isotropy index* [28], which is the reciprocal value of the condition number. This index addresses how isotropic the motion capabilities of the end-effector are. The final metric is the square root of the product of the eigenvalues, called *Yoshikawa's measure of manipulability* [36], which is proportional to the ellipsoid's volume and provides information about the overall manipulability of the end-effector.

### 3.2.2 Minimum singular value

The minimum singular value $\sigma_m$ is the smallest value resulting from the singular value decomposition (SVD), as shown in (3.3). For each non-zero singular value in the $\boldsymbol{D}$ matrix, there is a corresponding column in $\boldsymbol{U}$, which spans the space of solutions.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

As $\sigma_m$ approaches zero, the manipulator is closer to singularity, and its ability to perform subsequent tasks becomes limited [28]. This is a straightforward measure. However, the SVD method suffers from high computational costs, which limit its practical use for online applications.

### 3.2.3 Condition Number and Isotropy Index

Salisbury and Craig [77] expanded the concept of manipulability by introducing the condition number $\Gamma_C(\mathbf{J}_G)$. By calculating the SVD with (3.3) of $\mathbf{J}_G$ and then comparing the largest $\sigma_1$ and the smallest $\sigma_m$ singular values, $\Gamma_C(\mathbf{J}_G)$ can be obtained in the form

$$\Gamma_C(\mathbf{J}_G) = \frac{\sigma_1}{\sigma_m}. \tag{3.4}$$

A higher ratio indicates a more ill-conditioned matrix, provoking larger joint velocities. Thus, this measure gives a clear insight into the stability of the manipulator's motion by providing the ratio of the ellipsoid's longest to the shortest axis.

The manipulability ellipsoid is said to be isotropic and can move in all directions if this measure, which is lower-bounded by 1, equals 1. This value has no upper bound and approaches infinity as the robot draws closer to a singularity. The geometrical interpretation of this measure is the length ratio of the major and minor semi-axis of the manipulability ellipsoid. Yoshikawa [78] regarded this measure as an indicator of the velocity ellipsoid's directional uniformity.

As can be seen, this metric only traces the most difficult and easiest directions of motion and disregards the other ones. A disadvantage is that it can be sensitive to small changes in the input data, which can lead to significant differences in the value of the condition number.

The isotropy index $\Gamma_I(\mathbf{J}_G)$ evaluates the directional uniformity of the manipulability ellipsoid. It is defined as the reciprocal of the condition number $\Gamma_C(\mathbf{J}_G)$ from (3.4).

It was introduced as an alternative to the condition number to avoid issues that arise when the Jacobian matrix becomes ill-conditioned and $\Gamma_C(\mathbf{J}_G)$ approaches infinity. Similarly to the condition number, the disadvantage of the isotropy index is that it solely considers two singular values, disregarding the others.

### 3.2.4 Yoshikawa's measure of manipulability

In [36], T. Yoshikawa introduced the manipulability index $w$ to measure the kinematic performance. It is a quantitative scalar measure based on the Jacobian matrix. In the case of non-redundant manipulators, the measure can be calculated simply with the help of the determinant

$$w = \det(\mathbf{J}_G). \tag{3.5}$$

For redundant manipulators, $w$ is defined as the square root of the determinant of the Jacobian matrix $\mathbf{J}_G$ times its transpose

$$w = \sqrt{\det(\mathbf{J}_G \mathbf{J}_G^T)}. \tag{3.6}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

As outlined and explained by T. Yoshikawa in [36], this measure is also related to the manipulability ellipsoid, or more precisely, to the ellipsoid's volume space

$$V = \delta w, \tag{3.7}$$

where $\delta$ is a constant coefficient whose value depends on the number of singular values. This implies that the volume is directly proportional to $w$, which can be expressed by the singular values $\sigma_1, \sigma_2 \dots \sigma_m$ of the Jacobian matrix

$$w = \sigma_1 \sigma_2 \dots \sigma_m. \tag{3.8}$$

The manipulability index is considered by Tadokoro et al. [79] to be a more accurate measure of dexterity compared to the minimum singular value and the condition number. This is because, contrary to the indices only evaluating motion in one or two dimensions, the manipulability index considers the end-effector's motion in all directions. Furthermore, the manipulability index is invariant to changes in the reference frame, unlike the condition number and the minimum singular value.

## 3.3 Singularity simulations

As demonstrated in Section 2.1.10, the *KUKA LBR iiwa R820* has four singular configurations. Verifying whether these criteria result in singularity is possible with the manipulability indices.
The conditions to be examined are :

1. $\theta_2 = 0°$ and $\theta_3 = \pm 90°$,
2. $\theta_4 = 0°$,
3. $\theta_2 = 0°$ and $\theta_6 = 0°$,
4. $\theta_5 = \pm 90°$ and $\theta_6 = 0°$.

In the following, the four singular configurations are evaluated by three manipulability indices: the minimum singular value, the condition number, and the Yoshikawa manipulability index. The results are depicted from Figure 3.9 to Figure 3.12.

As the minimum singular value evaluates merely the least favorable direction, it is expected to provide only modest information. Even though this measure indicates when the configuration is approaching a singularity, it provides no further data about the other directions. Nevertheless, the minimum singular value is the most effective measure of distance from singular configurations.

As it compares the largest and lowest singular values, the condition number is expected to increase rapidly and significantly as the system approaches a singular configuration.

The Yoshikawa manipulability index is anticipated to be a more comprehensive measure of the workspace as it considers the whole Jacobian matrix. This measure is also expected to be smoother and to provide more nuanced results.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

**Case 1:** $\theta_2 = 0°$ **and** $\theta_3 = \pm 90°$: the joint angles are set to the configuration $[0° \ \theta_2 \ \theta_3 \ 90° \ 0° \ 90° \ 0°]$, $\theta_2$ and $\theta_3$ vary.

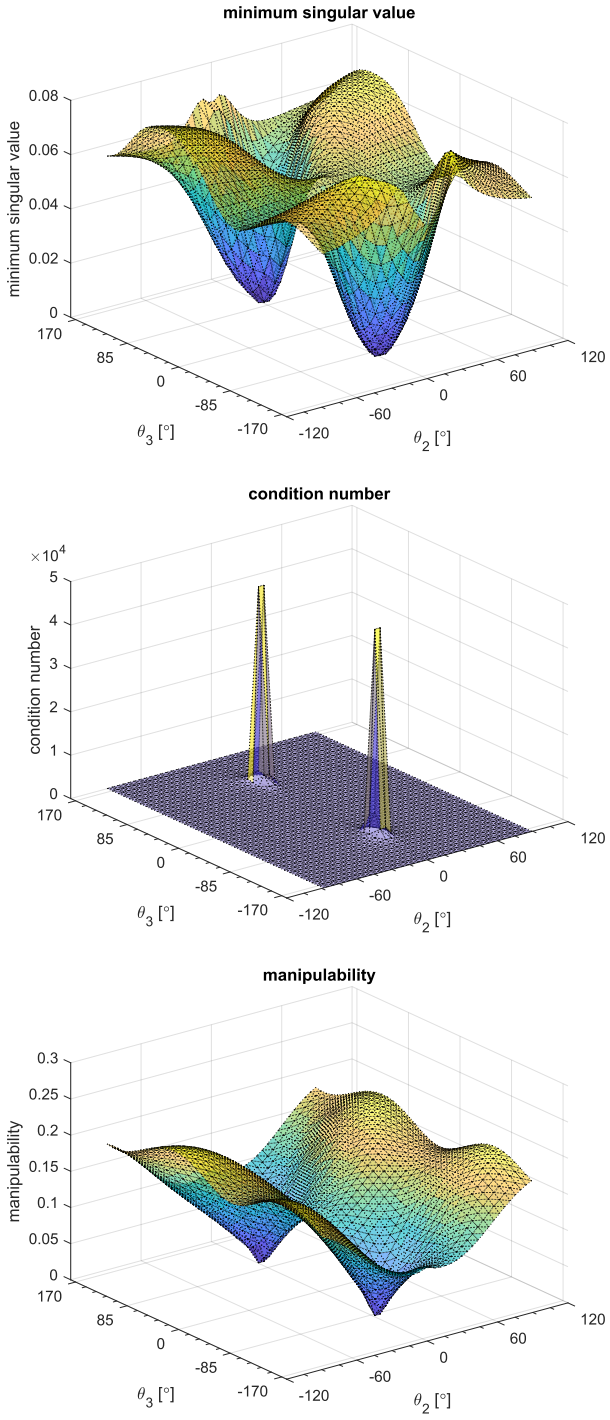


Figure 3.9: Performance indices for the singular case $\theta_2 = 0°$ and $\theta_3 = \pm 90°$.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

**Case 2:** $\theta_4 = 0°$: the joint angles are set to the configuration [0° 25° 0° $\theta_4$ 0° 90° 0°] and only $\theta_4$ is variable.

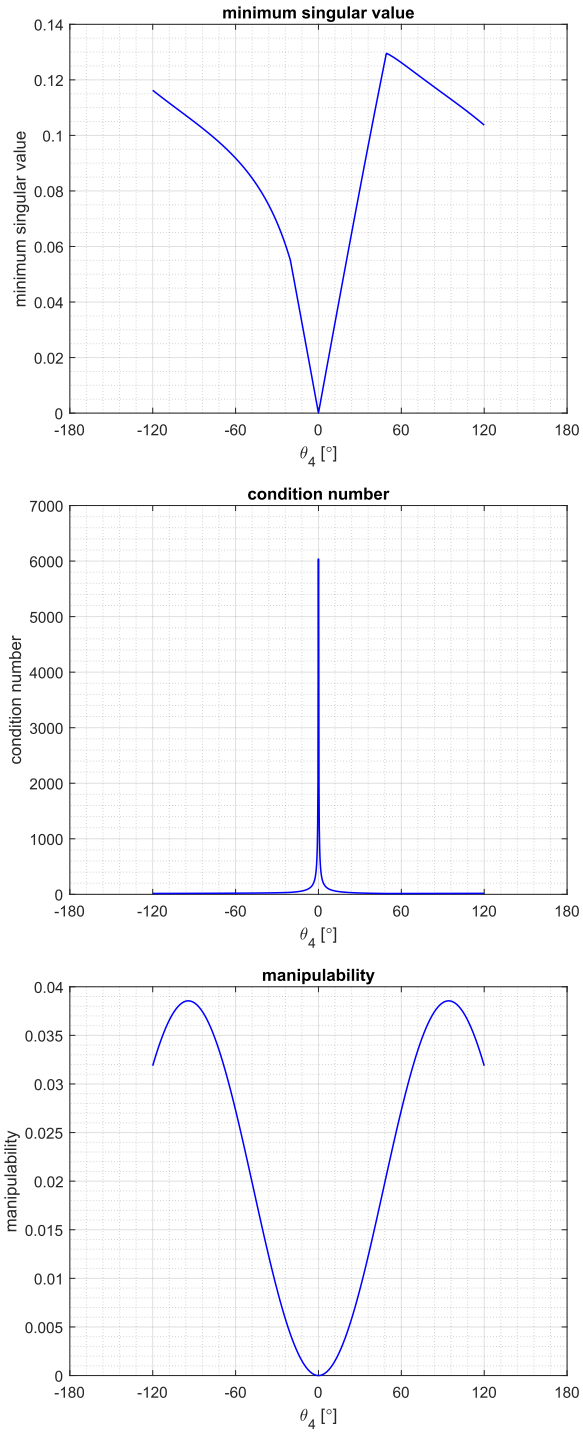


Figure 3.10: Performance indices for the singular case $\theta_4 = 0°$.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

**Case 3:** $\theta_2 = 0°$ **and** $\theta_6 = 0°$: the joint angles are set to the configuration $[0° \ \theta_2 \ 0° \ 90° \ 0° \ \theta_6 \ 0°]$, $\theta_2$ and $\theta_6$ vary.

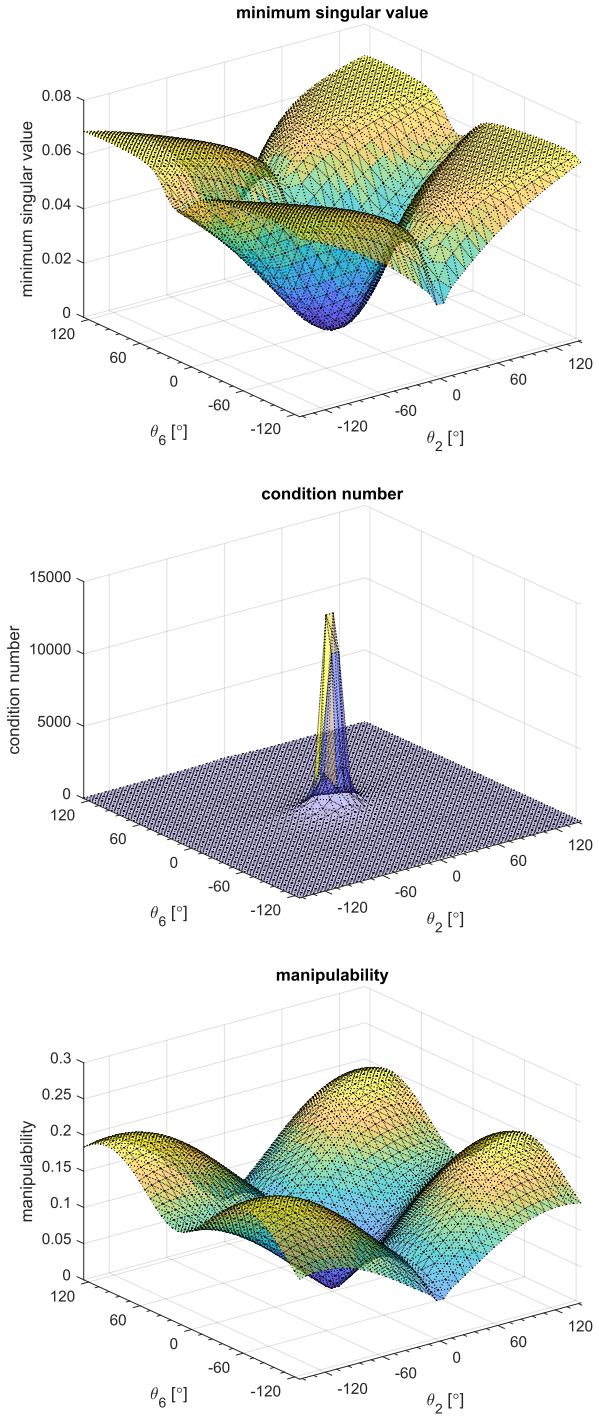


Figure 3.11: Performance indices for the singular case $\theta_2 = 0°$ and $\theta_6 = 0°$.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

**Case 4:** $\theta_5 = \pm 90°$ **and** $\theta_6 = 0°$: the joint angles are set to the configuration $[0° \ 25° \ 0° \ 90° \ \theta_5 \ \theta_6 \ 0°]$, $\theta_5$ and $\theta_6$ vary.

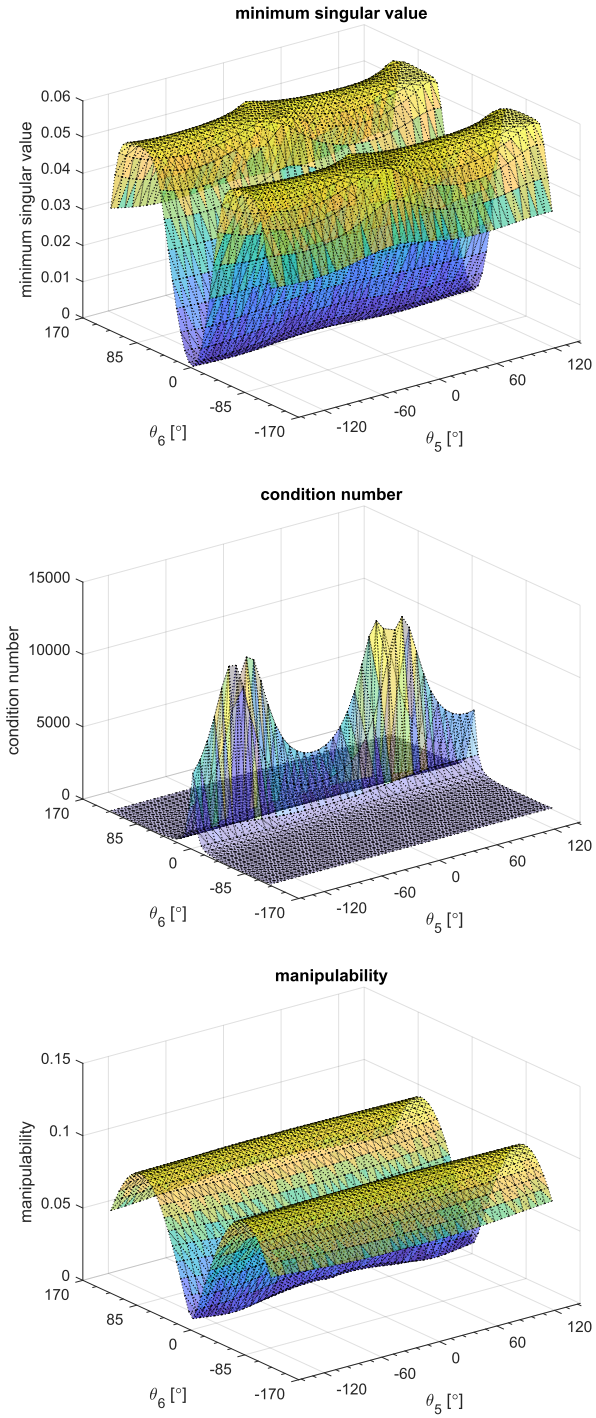


Figure 3.12: Performance indices for the singular case $\theta_5 = \pm 90°$ and $\theta_6 = 0°$.

As the results show, the minimum singular value and the condition number, even though they indicate the presence of a singularity, do not calculate the manipulability in the whole workspace as the Yoshikawa index does.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Contrary to the other two measures, the Yoshikawa manipulability index considers the end-effector motion in all directions. This is especially true for the condition number, which increases rapidly near and at singularities but otherwise remains flat. On the other hand, low condition numbers can be utilized in trajectory planning since robot configurations with small condition numbers help prevent significant variations in the joint rates.

## 3.4 Yoshikawa manipulability histogram

The Monte Carlo method can be employed with the forward kinematics to generate a data set with the guarantee that the configurations are reachable for the manipulator.

Now, to get a better understanding of the workspace, the Yoshikawa manipulability measure can be calculated for each data point. Then, the results can be presented as a histogram chart to demonstrate the frequency and distribution of the manipulability values.

The histogram can be generated for different point clouds. However, comparing the results is challenging because of the sample size and bin width disparities. This can be avoided by having a constant bin width and normalizing the histograms such that the bars total up to one. Figure 3.13 shows the results for three different sample sizes.

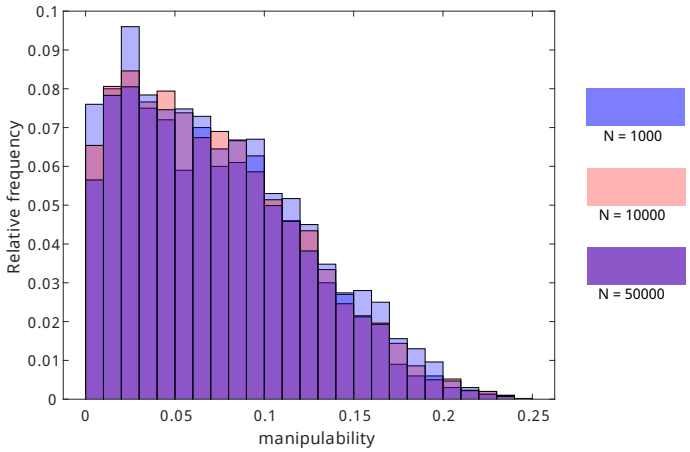


Figure 3.13: Histograms with different data set sizes follow a similar distribution.

The experiments validate the predictions, as a similarly decreasing distribution across all data sets can be observed regardless of the data size. To model the point set, a *gamma distribution* was chosen due to its flexibility in modeling right-skewed data [80], which reads as

$$\gamma(x,a,b) = \frac{1}{b^a \Gamma(a)} x^{a-1} e^{-x/b}, \tag{3.9}$$

where $x$ is a non-negative real number, $\Gamma(a)$ is the gamma function evaluated at $a$, and $a$ and $b$ are positive real numbers representing the shape and scale of the distribution, respectively. The function parameters can be estimated by using curve fitting, which was performed using MATLAB. A gamma distribution with the parameters $a = 1.754$ and $b = 0.043$ fits the point cloud data well, as shown in Figure 3.14.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

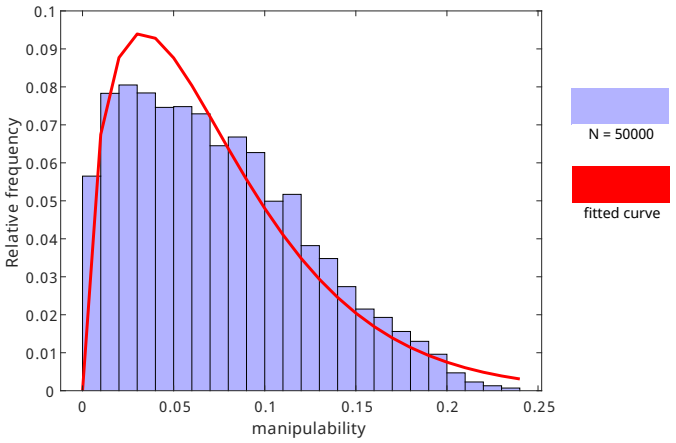


Figure 3.14: Curve fitting by a gamma distribution with $a = 1.754$ and $b = 0.043$.

In the following Section, the data is categorized into groups with varying degrees of manipulability, which enables a heterogeneous representation.

## 3.5 Workspace heterogeneity

The selected approach is the *alpha shape method*, as introduced in Section 3.1.3. The sampling size is $N = 50000$ with $\alpha = 0.1$ to have a large enough data set, which was then extended with the Yoshikawa manipulability index for each data point. The points were then grouped in clusters to generate boundary surfaces. The different sections of the heterogeneous workspace are depicted in Figure 3.15 and Figure 3.16.

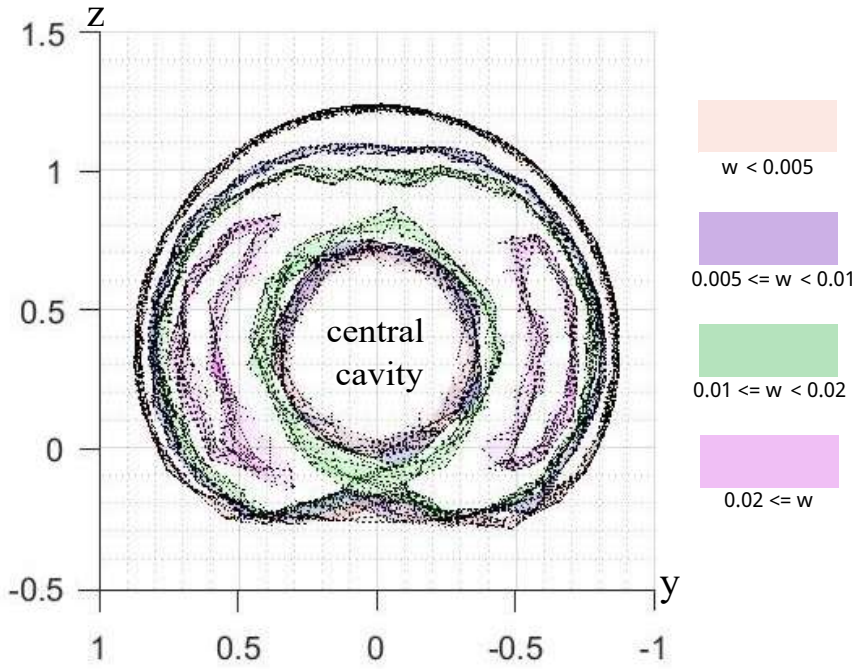


Figure 3.15: $\alpha$-shape cross section boundary representation.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

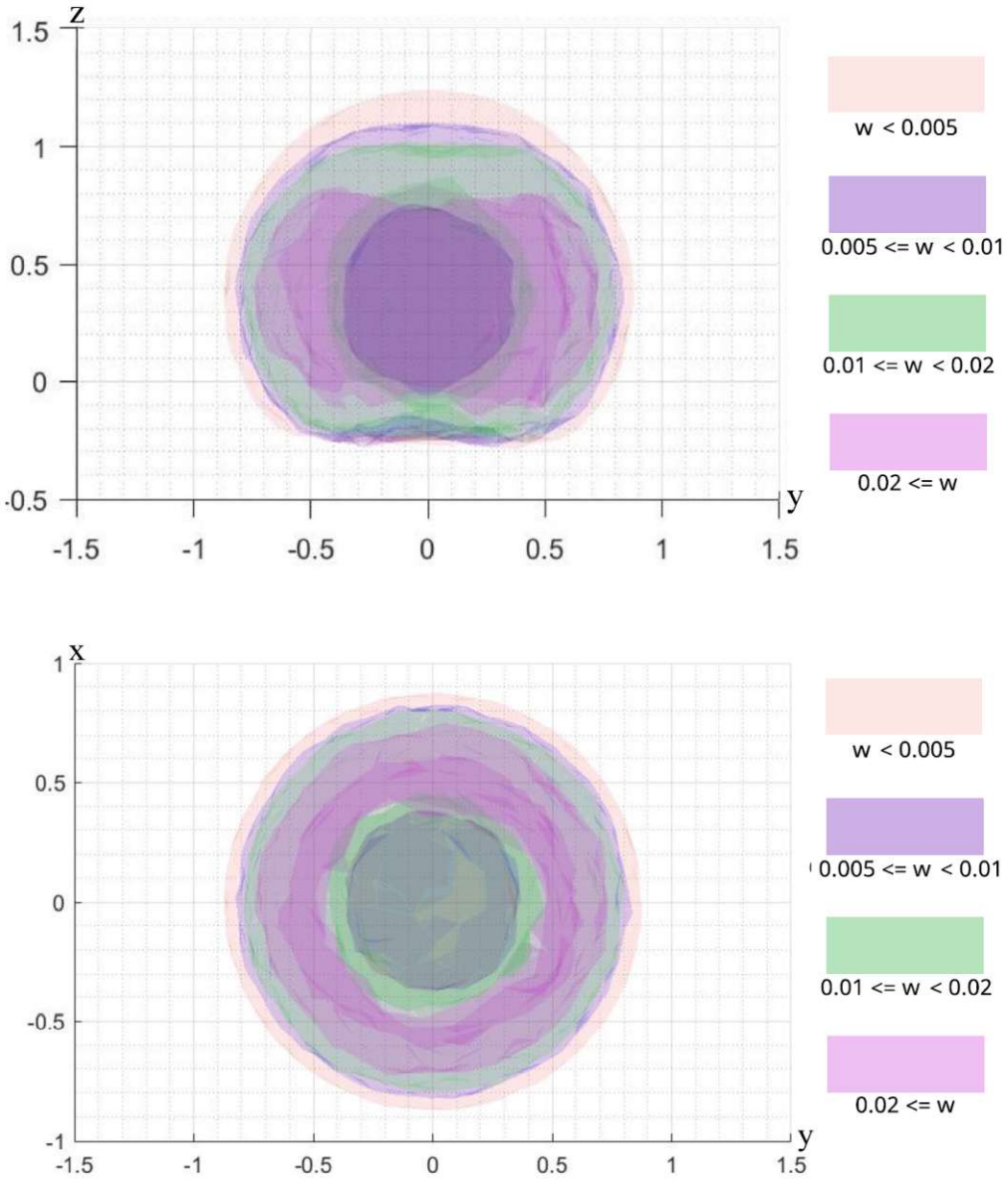


Figure 3.16: Heterogeneous workspace cross-section and upper view.

As expected, the lower the minimum allowed manipulability index value is, the larger the considered workspace becomes. The outside border and the internal cavity are rigid boundaries limiting the workplace. The highest manipulability is in the interior of the volume, far away from the central cavity.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

# Chapter 4

# Trajectory planning and optimization

Trajectory planning is the generation of trajectories in joint or Cartesian space to coordinate the different joint and wheel variables. This allows the end-effector of the mobile manipulator to reach desired points in space with the desired orientation while taking into account the limits of the actuators and the robot in general. In line with the ideas advanced in [32] [24], and [81], the thesis considers trajectory planning as a dynamic optimization problem subject to the robot's kinematics and other constraints.

This chapter aims to determine the optimal control inputs for the mobile manipulator to achieve optimal trajectories to the desired pose. This is accomplished by using the mathematical model containing seven actuated joints and two actuated wheels, as established in Chapter 2.

In this thesis, the offline trajectory planning employs the angular jerk as the control input.

The problem is first stated and specified, including the cost function, the system's various restrictions, and the task itself. The kinodynamics of the robot serves as a basis for the planning task, including the kinematic constraints, which are referred to as *interior* or *robot* constraints. The *exterior* or *workspace* constraints include the initial and goal conditions.

This chapter also introduces the employed technique and its parameters, which were implemented and tested in the MATLAB R2021 environment. Finally, the results of the optimization are presented, along with an investigation into the influence of integrating weighted manipulability in the optimal control problem and the impact of the weighting factor on the solution.

## 4.1 Optimal control problem

The process of choosing a dynamic system's control and state trajectories over time to minimize an optimality criterion is known as *optimal control*, which will be covered in this section.

The initial step is to analyze the planning problem itself. Therefore, this first part aims to define the trajectory planning problem through differential equations, algebraic equalities, and inequalities, including cost function and constraints.



Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

### 4.1.1 Problem formulation

The considered dynamic optimization problem formulation reads as

$$\min_{\mathbf{x},\mathbf{u},t_f} \quad J(\mathbf{x},\mathbf{u},t_f) = \varphi(t_f,\mathbf{x}(t_f)) + \int_{t_0}^{t_f} l(t,\mathbf{x}(t),\mathbf{u}(t))dt \qquad \text{cost functional} \quad (4.1a)$$

$$\text{s.t.} \quad \dot{\mathbf{x}} = \mathbf{f}(t,\mathbf{x},\mathbf{u}) \qquad \text{system dynamics} \quad (4.1b)$$

$$\mathbf{x}(t_0) = \mathbf{x}_0 \qquad \text{initial conditions} \quad (4.1c)$$

$$\boldsymbol{\psi}(t_f,\mathbf{x}(t_f)) = \mathbf{0} \qquad \text{terminal conditions} \quad (4.1d)$$

$$h_j(\mathbf{x},\mathbf{u}) \leq 0 \ , \ j = 1,\dots,q \qquad \text{inequality constraints} \quad (4.1e)$$

The state $\mathbf{x} \in \mathbb{R}^n$ represents the state of the nonlinear system, while the input variable $\mathbf{u} \in \mathbb{R}^m$ is the control input applied to it. $\varphi(t_f,\mathbf{x}(t_f))$ is the terminal cost, which is a function of the final state of the system and $l(t,\mathbf{x}(t),\mathbf{u}(t))$ is the running cost, which penalizes the system for deviations from the desired state and control inputs over the time interval $[t_0,t_f]$. The terminal conditions are given by the vector function $\boldsymbol{\psi}(t_f,\mathbf{x}(t_f)) = \mathbf{0}$, which specifies constraints on the final state of the system. The inequality constraints are given by the functions $h_j(\mathbf{x},\mathbf{u}) \leq 0$, $j = 1,\dots,q$, specifying further constraints on the state and input variables.

The goal of (4.1a) is to find an optimal control input $\mathbf{u}(t)$, $t \in [t_0,t_f]$ and state trajectory $\mathbf{x}(t)$, $t \in [t_0,t_f]$ that minimize a cost functional $J(\mathbf{x},\mathbf{u},t_f)$, subject to the system dynamics (4.1b), initial (4.1c) and terminal conditions (4.1d), and inequality constraints (4.1e). The terminal time $t_f$ is also an optimization variable that determines the final time of the state trajectory.

The given trajectory planning problem aims to satisfy the following requirements:

1. The fundamental task is to move the end-effector from its start pose to the desired pose in Cartesian space.
2. The resulting trajectories must be sufficiently smooth: the angular positions, velocities, and accelerations must be continuous and differentiable.
3. The task termination time is limited, and the movement must end within this time constraint.

The state variables $\mathbf{x}$ of the manipulator are given by $\mathbf{x}_1 = \mathbf{q}$, $\mathbf{x}_2 = \dot{\mathbf{q}}$, $\mathbf{x}_3 = \ddot{\mathbf{q}}$, denoting the joint angles, velocities, and accelerations, respectively, with $\mathbf{q} = [q_1,\dots\ q_7]$ representing the seven joint angles. The state variables of the mobile platform are according to (2.33) $x_4 = x_l$, $x_5 = y_l$ and $x_6 = \theta_l$ representing the current position and the heading angle, see Section 2.2.1. The wheel velocities $v_L$ and $v_R$ are summarized as $\mathbf{v} = [v_L,v_R]^T$ and are represented as state variables by $\mathbf{x}_7 = \mathbf{v}$, their accelerations as $\mathbf{x}_8 = \dot{\mathbf{v}}$, respectively.

The summarized state variables can be expressed as

$$\mathbf{x} = [\mathbf{q}^T, \dot{\mathbf{q}}^T, \ddot{\mathbf{q}}^T, x_l, y_l, \theta_l, \mathbf{v}^T, \dot{\mathbf{v}}^T]^T. \quad (4.2)$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

This leads to the following dynamic system

$$\dot{\mathbf{x}} = \frac{d}{dt}\begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \\ \ddot{\mathbf{q}} \\ x_l \\ y_l \\ \theta_l \\ \mathbf{v} \\ \dot{\mathbf{v}} \end{bmatrix} = \mathbf{f}(\mathbf{x}, t, \mathbf{u}_m, \mathbf{u}_p), \tag{4.3}$$

where $\mathbf{f}(\mathbf{x}, t, \mathbf{u}_m, \mathbf{u}_p)$ represents the dynamics of the system, $\mathbf{u}_m(t) \in \mathbb{R}^7$ is the control input for the manipulator and $\mathbf{u}_p(t) \in \mathbb{R}^2$ is the control input for the mobile platform. The system dynamics $\mathbf{f}(\mathbf{x}, t, \mathbf{u}_m, \mathbf{u}_p)$ will be derived in the subsequent Subsection 4.1.2 and presented in (4.7).

To control the manipulator, the jerks $\mathbf{j}(t) = \mathbf{u}_m(t)$ are used as control variables. For the mobile platform, individual left and right wheel jerks $j_L(t) = u_{p,L}(t)$ and $j_R(t) = u_{p,R}(t)$ are used, which are summed up in $\mathbf{u}_p(t) = [u_{p,L}, u_{p,R}]^T$. This modification changes the platform's system dynamics slightly, as the wheel velocities and accelerations are now calculated from the jerk inputs, similar to how the manipulator's joint velocities and accelerations are calculated.

### 4.1.2 System dynamics and constraints

Constraints play an important role in optimization problems as they define the feasible region for the possible values of the optimization variables.

They can be classified as equality or inequality constraints. Distinguishing between them is important because it enables the optimization algorithm to handle each type of constraint properly. Equality constraints are used to ensure that certain conditions are met, while inequality constraints limit the range of possible solutions. Another classification of constraints is their source, internal or external.

*Internal* or *robot constraints* reflect the physical nature of the robot and impose kinematic equality constraints on its motion. They can be defined for both the *KUKA LBR iiwa R820* manipulator and the *SALLY V2.0* mobile platform.

As for the manipulator, the kinematic equality constraints of motion can be expressed as a simple integrator chain

$$\begin{bmatrix} \dot{\mathbf{x}}_1 \\ \dot{\mathbf{x}}_2 \\ \dot{\mathbf{x}}_3 \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{0}_{7\times7} & \mathbf{I}_{7\times7} & \mathbf{0}_{7\times7} \\ \mathbf{0}_{7\times7} & \mathbf{0}_{7\times7} & \mathbf{I}_{7\times7} \\ \mathbf{0}_{7\times7} & \mathbf{0}_{7\times7} & \mathbf{0}_{7\times7} \end{bmatrix}}_{\mathbf{A}_m} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \end{bmatrix} + \underbrace{\begin{bmatrix} \mathbf{0}_{7\times7} \\ \mathbf{0}_{7\times7} \\ \mathbf{I}_{7\times7} \end{bmatrix}}_{\mathbf{B}_m} \mathbf{u}_m, \tag{4.4}$$

with $\mathbf{A}_m \in \mathbb{R}^{21\times21}$ as the system matrix and $\mathbf{B}_m \in \mathbb{R}^{21\times7}$ as the control matrix.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

In addition to the manipulator constraints, the mobile platform has its own constraints that affect its position, orientation, and rear-axis velocity. To model these constraints, a time-continuous model was introduced in (2.35), representing the constraint equations for the differential drive. This model takes into account the non-holonomic nature of the mobile platform, where the robot can only move in certain directions due to its mechanical constraints. The platform dynamics are given by

$$\begin{bmatrix} \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} \frac{1}{2}\cos(x_6) & -\frac{1}{2}\cos(x_6) \\ \frac{1}{2}\sin(x_6) & \frac{1}{2}\sin(x_6) \\ -\frac{1}{2d} & \frac{1}{2d} \end{bmatrix} \mathbf{x}_7. \tag{4.5}$$

The remaining kinematics of the mobile platform can be expressed as an integrator chain

$$\begin{bmatrix} \dot{\mathbf{x}}_7 \\ \dot{\mathbf{x}}_8 \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{0}_{2\times2} & \mathbf{I}_{2\times2} \\ \mathbf{0}_{2\times2} & \mathbf{0}_{2\times2} \end{bmatrix}}_{\mathbf{A}_p} \begin{bmatrix} \mathbf{x}_7 \\ \mathbf{x}_8 \end{bmatrix} + \underbrace{\begin{bmatrix} \mathbf{0}_{2\times2} \\ \mathbf{I}_{2\times2} \end{bmatrix}}_{\mathbf{B}_p} \mathbf{u}_p, \tag{4.6}$$

with $\mathbf{A}_p \in \mathbb{R}^{4\times4}$ as the system matrix and $\mathbf{B}_p \in \mathbb{R}^{4\times2}$ as the control matrix.

The coupled system's dynamics, as in (4.3), can now be summed up as

$$\begin{bmatrix} \dot{\mathbf{x}}_1 \\ \dot{\mathbf{x}}_2 \\ \dot{\mathbf{x}}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \\ \dot{\mathbf{x}}_7 \\ \dot{\mathbf{x}}_8 \end{bmatrix} = \mathbf{f}(\mathbf{x},t,\mathbf{u}_m,\mathbf{u}_p) = \begin{bmatrix} \mathbf{A}_m \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \end{bmatrix} + \mathbf{B}_m\mathbf{u}_m \\ \begin{bmatrix} \frac{1}{2}\cos(x_6) & -\frac{1}{2}\cos(x_6) \\ \frac{1}{2}\sin(x_6) & \frac{1}{2}\sin(x_6) \\ -\frac{1}{2d} & \frac{1}{2d} \end{bmatrix} \mathbf{x}_7 \\ \mathbf{A}_p \begin{bmatrix} \mathbf{x}_7 \\ \mathbf{x}_8 \end{bmatrix} + \mathbf{B}_p\mathbf{u}_p \end{bmatrix}. \tag{4.7}$$

The robot's joint positions, velocities, and accelerations, denoted as $\mathbf{x}_1(t)$, $\mathbf{x}_2(t)$, and $\mathbf{x}_3(t)$ respectively, are constrained within the specified minimum and maximum values. Similarly, the wheel velocities and accelerations, denoted as $\mathbf{x}_7$ and $\mathbf{x}_8$, are also bounded

$$\mathbf{x}_{i,min} \leq \mathbf{x}_i(t) \leq \mathbf{x}_{i,max}, \text{ with } i \in \{1,2,3,7,8\}. \tag{4.8}$$

Finally, $t_f$ is also limited

$$0 < t_f \leq t_{max}. \tag{4.9}$$

There are a total of 52 inequality constraints for the manipulator and mobile platform system. The manipulator has 42 constraints due to the upper and lower limits on the seven joints, velocities, and accelerations $(2 \cdot 3 \cdot 7)$. The mobile platform has 8 constraints for the wheel velocities and accelerations $(2 \cdot 2 \cdot 2)$ and 2 constraints for the motion time.

*External* or *workspace constraints* reflect the restrictions on the workspace and robot due to the specific task. The robot begins moving from a predetermined standstill configuration which corresponds to the initial conditions. The manipulator's starting configuration $\mathbf{x}_1(t_0)$ is given as the home configuration, while the platform's position

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

and heading angle are at the origin, as well as the initial velocities and accelerations are zero

$$\mathbf{x}_1(t_0) = \mathbf{x}_{1,0} = [0^\circ\ 25^\circ\ 0^\circ\ 90^\circ\ 0^\circ\ 0^\circ\ 0^\circ], \tag{4.10a}$$
$$\mathbf{x}_i(t_0) = \mathbf{0},\ \text{with } i \in \{2,3,7,8\}, \tag{4.10b}$$
$$x_i(t_0) = 0,\ \text{with } i \in \{4,5,6\}. \tag{4.10c}$$

In order to achieve the desired end pose of the robot in Cartesian space, the position and orientation must be accurately calculated. To accomplish this, the entire kinematic chain of the mobile manipulator is utilized as a transformation matrix. For the orientation of the robot, quaternions are used to calculate the final orientation of the robot.

The terminal conditions must be met at $t = t_f$, and the robot must reach the desired position $\mathbf{p}_d$ and orientation $\mathbf{o}_d$. The position $\mathbf{p}_{EE,f}$ and orientation $\mathbf{o}_{EE,f}$ of the end-effector in the Cartesian space at $t_f$ are expressed by $\mathbf{p}_{EE,f} = [x_{EE}(t_f),\ y_{EE}(t_f),\ z_{EE}(t_f)]^T$ and $\mathbf{o}_{EE,f} = [\gamma_{EE}(t_f),\ \beta_{EE}(t_f),\ \zeta_{EE}(t_f)]^T$, respectively. The terminal conditions are expressed as

$$\mathbf{p}_{EE,f} = \mathbf{p}_d = [x_d(t_f),\ y_d(t_f),\ z_d(t_f)]^T, \tag{4.11a}$$
$$\mathbf{o}_{EE,f} = \mathbf{o}_d = [\gamma_d(t_f),\ \beta_d(t_f),\ \zeta_d(t_f)]^T. \tag{4.11b}$$

Additionally, the robot must come to a complete stop when it reaches the intended destination, therefore at $t = t_f$, both the velocity and acceleration of the manipulator and the platform must be zero

$$\mathbf{x}_i(t_f) = \mathbf{0}, \text{with } i \in \{2,3,7,8\}. \tag{4.12}$$

### 4.1.3 Optimality criteria and cost function

As noted and outlined by Chiddarwar and Babu [81], trajectory planning can be guided by several optimality criteria, including

(1) minimum energy
(2) minimum acceleration
(3) minimum jerk
(4) minimum execution time
(5) maximum manipulability

Each criterion provides a way to balance the trade-offs that arise when determining the optimal trajectory. Considering the specific requirements and limitations of the system and application, a suitable set of criteria can be selected to ensure that the resulting trajectory satisfies the specifications.

*Minimum energy* based trajectory planning seeks to generate a trajectory that requires the least amount of energy [81]. This criterion is particularly useful in applications with scarce energy resources, such as underwater or space exploration. By minimizing the energy required for motion, this criterion results in smoother trajectories with less stress on the actuators, reducing their efforts and potential for wear and tear.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

*Minimum acceleration* trajectory planning aims to minimize the integral of the squared acceleration values [81]. This results in smooth joint trajectories and continuously differentiable velocity profiles, making it a desirable criterion in many applications. However, its effectiveness depends on the chosen parametrization.

Jerk is the third derivative of position and the time derivative of acceleration, playing a significant role in determining the smoothness of a system's motion. *Minimum jerk criterion* trajectories aim to minimize this quantity in order to produce a smooth motion [81]. This criterion reduces the potential for wear and tear on the system and helps avoid the excitation of resonance frequencies that can cause vibrations and errors. The trade-off is that higher levels of jerk enable the system to adjust faster to changes in movement.

The minimum jerk criterion can be expressed as an optimization problem that minimizes the integral of the squared jerks over the duration of motion, denoted by $t_f$, and the time range from $t_0$ to $t_f$

$$\min_{x} \quad \int_{t_0}^{t_f} \dddot{x}^2(t)\, dt. \tag{4.13}$$

Higher-order derivatives, such as the fourth (snap), fifth (crackle), or sixth (pop) derivatives, have not been considered for producing smoother trajectories, as Richardson et al. [82] demonstrated that as the order of the derivative increases above three, the solution of $x(t)$ approaches a step function, resulting in extreme velocity changes.

When using the *minimum execution time* criterion, the goal is to minimize the duration of motion

$$\min_{t_f} \quad \int_{t_0}^{t_f} 1\ dt. \tag{4.14}$$

Considering the time in the cost function is highly recommended [81]. Taking the duration of the trajectory as a design variable impedes generating extremely time-consuming but low-energy-requiring paths and avoids an overly long processing time, which is typically competing with smoothness objectives.

*Maximum manipulability* is a practical design criterion helping the robot to have a high degree of freedom during or at the end of its movement [81]. Furthermore, it helps in avoiding singularities. Yoshikawa's manipulability index [36], as stated in Section 3.2.4, can be determined with the help of the Jacobian matrix $\mathbf{J}_G(\mathbf{q}(t))$ calculated at $t_f$ in the form (3.6), which reads as

$$m(\mathbf{q}(t_f)) = \sqrt{\det(\mathbf{J}_{G,f}\mathbf{J}_{G,f}^T)}. \tag{4.15}$$

The index equals zero only in singular configurations. Due to the time-consuming computation, only the final posture of the end-effector at $t_f$ is evaluated in the cost function. The cost function to maximize the manipulability reads as

$$\max_{\mathbf{q}(t_f)} \quad m(\mathbf{q}(t_f)), \tag{4.16}$$

which is equivalent to the following minimization problem

$$\min_{\mathbf{q}(t_f)} \quad -m(\mathbf{q}(t_f)). \tag{4.17}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

However, the issue of designing the minimization problem in this form is that the negative sign causes direct compensation of other cost factors. Therefore, it is preferable to extend the formulation of the maximum manipulability criterion with a scalar $0 < \varepsilon \ll 1$ as follows, which helps to prevent division by zero

$$\min_{\mathbf{q}(t_f)} \quad \frac{1}{\varepsilon + \sqrt{\det(\mathbf{J}_{G,f}\mathbf{J}_{G,f}^T)}}. \tag{4.18}$$

Besides fulfilling the main task and adhering to the constraints, the execution time should be decreased. Simultaneously, the robot's final configuration should be selected so that future movement in the immediate vicinity is simple, preferably without moving the platform.

We select the *minimum jerk* (4.13), *minimum time* (4.14) and the *maximum manipulability* (4.18) as optimality criteria for the optimization problem. The designed cost function reflects a multi-objective dynamic optimization problem, as the cost function consists of three individual optimality criteria combined into one objective. Accordingly, the cost function reads as

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}_m,\mathbf{u}_p,t_f} \quad & J(\mathbf{x},\mathbf{u}_m,\mathbf{u}_p,t_f) \quad = w_1 \int_{t_0}^{t_f} \mathbf{u}_m^T(t)\mathbf{u}_m(t)\,dt + w_2 \int_{t_0}^{t_f} \mathbf{u}_p^T(t)\mathbf{u}_p(t)\,dt + \\ & + w_3 \int_{t_0}^{t_f} 1\,dt + w_4 \frac{1}{\varepsilon + \sqrt{\det(\mathbf{J}_{G,f}\mathbf{J}_{G,f}^T)}}\,, \end{aligned} \tag{4.19}$$

with $w_1$, $w_2$, $w_3$ and $w_4$ representing the positive scalar weighting factors.

### 4.1.4 Multi-objective dynamic optimization problem

Summing up, the dynamic optimization problem reads as

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}_m,\mathbf{u}_p,t_f} \quad & J(\mathbf{x},\mathbf{u}_m,\mathbf{u}_p,t_f) = w_1 \int_{t_0}^{t_f} \mathbf{u}_m^T(t)\mathbf{u}_m(t)\,dt + w_2 \int_{t_0}^{t_f} \mathbf{u}_p^T(t)\mathbf{u}_p(t)\,dt + \\ & + w_3 \int_{t_0}^{t_f} 1\,dt + \frac{w_4}{\varepsilon + \sqrt{\det(\mathbf{J}_{G,f}\mathbf{J}_{G,f}^T)}}, \qquad w_1,\ w_2,\ w_3,\ w_4,\ \varepsilon > 0 \end{aligned} \tag{4.20a}$$

$$\text{s.t.} \quad \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},t,\mathbf{u}_m,\mathbf{u}_p) \tag{4.20b}$$

$$\mathbf{x}_1(t_0) = \mathbf{x}_{1,0} \tag{4.20c}$$

$$\mathbf{x}_i(t_0) = \mathbf{0}, \ \text{with } i \in \{2,3,7,8\} \tag{4.20d}$$

$$x_i(t_0) = 0, \ \text{with } i \in \{4,5,6\} \tag{4.20e}$$

$$\mathbf{x}_i(t_f) = \mathbf{0}, \ \text{with } i \in \{2,3,7,8\} \tag{4.20f}$$

$$\mathbf{p}_{EE,f} = \mathbf{p}_d \tag{4.20g}$$

$$\mathbf{o}_{EE,f} = \mathbf{o}_d \tag{4.20h}$$

$$h_i(\mathbf{x},t,\mathbf{u}_m,\mathbf{u}_p) \leq \mathbf{0}, \ i = 1,2\ldots,52. \tag{4.20i}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

### 4.1.5 Discretized optimization problem

Direct transcription techniques will be used to transform the problem into a discrete optimization problem [83]. This involves dividing the time domain into a finite number of intervals, resulting in a finite-dimensional nonlinear programming problem (NLP) where all state and control variables are discretized in time. Due to the complexity of the resulting NLP, highly efficient solvers are necessary for finding the optimal solution.

Let the number of waypoints be $N$. The division of the time domain $[t_0, t_f]$ into $N$ finite intervals $[\Delta t_0, \Delta t_1 \ldots, \Delta t_{N-1}]$ can be expressed such as

$$\sum_{i=0}^{N-1} \Delta t_i = t_f. \tag{4.21}$$

The intervals can be chosen to be equidistant or have variable lengths. Using non-equidistant intervals can reduce the number of required waypoints, but convergence may be slower. Additionally, incorporating time intervals as optimization variables adds complexity to the problem and requires more computational resources.

In light of the trade-off between convergence speed and the number of required waypoints, equidistant time intervals were selected

$$\Delta t = \frac{t_f}{N}. \tag{4.22}$$

The system dynamics of the manipulator need to be expressed as a difference equation. To do so, the equations of motion, which are a simple integrator chain (4.4), must be discretized. The state vector $\mathbf{x}_{i,k}$ with $i \in \{1,2,3\}$, which represents the state at each time step $k\Delta t_i$, is a function of the previous state and the input vector $\mathbf{u}_{m,k}$. The evolution of the state is described by the state transition matrix $\mathbf{\Phi}_m$ given by

$$\mathbf{\Phi}_m = \exp(\boldsymbol{A}_m \Delta t). \tag{4.23}$$

The effect of inputs on the system is captured by the input matrix $\mathbf{\Gamma}_m$, given by

$$\mathbf{\Gamma}_m = \int_0^{\Delta t} \exp(\boldsymbol{A}_m \tau) \mathrm{d}\tau \boldsymbol{B}_m. \tag{4.24}$$

Thus, the discrete-time system reads as

$$\begin{bmatrix} \mathbf{x}_{1,k+1} \\ \mathbf{x}_{2,k+1} \\ \mathbf{x}_{3,k+1} \end{bmatrix} = \mathbf{\Phi}_m \begin{bmatrix} \mathbf{x}_{1,k} \\ \mathbf{x}_{2,k} \\ \mathbf{x}_{3,k} \end{bmatrix} + \mathbf{\Gamma}_m \mathbf{u}_{m,k}. \tag{4.25}$$

In addition to the manipulator dynamics, the dynamics of the mobile platform must also be considered as

$$\begin{bmatrix} \mathbf{x}_{7,k+1} \\ \mathbf{x}_{8,k+1} \end{bmatrix} = \mathbf{\Phi}_p \begin{bmatrix} \mathbf{x}_{7,k} \\ \mathbf{x}_{8,k} \end{bmatrix} + \mathbf{\Gamma}_p \mathbf{u}_{p,k}, \tag{4.26}$$

with $\mathbf{\Phi}_p = \exp(\boldsymbol{A}_p \Delta t)$ and $\mathbf{\Gamma}_p = \int_0^{\Delta t} \exp(\boldsymbol{A}_p \tau) \mathrm{d}\tau \boldsymbol{B}_p$, according to (4.6).

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

To obtain a time-discrete model for the mobile platform, integration methods such as the forward Euler method can be used [84], which approximates the time derivative at time $k\Delta t_i$ by the difference quotient $\frac{1}{\Delta t}(x((k+1)\Delta t_i) - x(k\Delta t_i))$, where $\Delta t$ is the time step. Applying this to each equation in (4.5) yields

$$x_{4,k+1} = x_{4,k} + \frac{2d}{2}\frac{x_{7,1,k}+x_{7,2,k}}{x_{7,2,k}-x_{7,1,k}}\left(\sin\left(\frac{x_{7,2,k}-x_{7,1,k}}{2d}\Delta t + x_{6,k}\right) - \sin(x_{6,k})\right), \tag{4.27}$$

$$x_{5,k+1} = x_{5,k} - \frac{2d}{2}\frac{x_{7,1,k}+x_{7,2,k}}{x_{7,2,k}-x_{7,1,k}}\left(\cos\left(\frac{x_{7,2,k}-x_{7,1,k}}{2d}\Delta t + x_{6,k}\right) - \cos(x_{6,k})\right), \tag{4.28}$$

$$x_{6,k+1} = x_{6,k} + \frac{\Delta t}{2d}\left(x_{7,2,k} - x_{7,1,k}\right). \tag{4.29}$$

The discrete-time model of the whole system takes the form

$$\begin{bmatrix} \mathbf{x}_{1,k+1} \\ \mathbf{x}_{2,k+1} \\ \mathbf{x}_{3,k+1} \\ x_{4,k+1} \\ x_{5,k+1} \\ x_{6,k+1} \\ \mathbf{x}_{7,k+1} \\ \mathbf{x}_{8,k+1} \end{bmatrix} = \mathbf{F}(\mathbf{x}_k, \mathbf{u}_{m,k}, \mathbf{u}_{p,k}) = \begin{bmatrix} \boldsymbol{\Phi}_m \begin{bmatrix} \mathbf{x}_{1,k} \\ \mathbf{x}_{2,k} \\ \mathbf{x}_{3,k} \end{bmatrix} + \boldsymbol{\Gamma}_m \mathbf{u}_{m,k} \\ x_{4,k} + \frac{2d}{2}\frac{x_{7,1,k}+x_{7,2,k}}{x_{7,2,k}-x_{7,1,k}}\left(\sin\left(\frac{x_{7,2,k}-x_{7,1,k}}{2d}\Delta t + x_{6,k}\right) - \sin(x_{6,k})\right) \\ x_{5,k} - \frac{2d}{2}\frac{x_{7,1,k}+x_{7,2,k}}{x_{7,2,k}-x_{7,1,k}}\left(\cos\left(\frac{x_{7,2,k}-x_{7,1,k}}{2d}\Delta t + x_{6,k}\right) - \cos(x_{6,k})\right) \\ x_{6,k} + \left(\frac{x_{7,2,k}-x_{7,1,k}}{2d}\right)\Delta t \\ \boldsymbol{\Phi}_p \begin{bmatrix} \mathbf{x}_{7,k} \\ \mathbf{x}_{8,k} \end{bmatrix} + \boldsymbol{\Gamma}_p \mathbf{u}_{p,k} \end{bmatrix}. \tag{4.30}$$

The jerks are chosen as the control variables as $\mathbf{u}_{m,k}$ and $\mathbf{u}_{p,k}$, and are considered to be constant over the corresponding time interval $k\Delta t_i$. The initial and terminal conditions do not change.

Now the discrete optimization problem can be formulated as

$$\min_{\substack{\mathbf{x}_0,..,\mathbf{x}_N \\ \mathbf{u}_{m,0},..,\mathbf{u}_{m,N-1} \\ \mathbf{u}_{p,0},..,\mathbf{u}_{p,N-1} \\ t_f}} \quad J(\mathbf{x},\mathbf{u}_m,\mathbf{u}_p,t_f) = w_1 \sum_{i=0}^{N-1} \mathbf{u}_{m,i}^T \mathbf{u}_{m,i} + w_2 \sum_{i=0}^{N-1} \mathbf{u}_{p,i}^T \mathbf{u}_{p,i} + \tag{4.31a}$$

$$w_3 \sum_{i=0}^{N-1} \Delta t_i + \frac{w_4}{\varepsilon + \sqrt{\det(\mathbf{J}_{G,f}\mathbf{J}_{G,f}^T)}}, \quad w_1,\ w_2,\ w_3,\ w_4,\ \varepsilon > 0$$

$$\text{s.t.} \quad \mathbf{x}_{k+1} = \mathbf{F}(\mathbf{x}_k, \mathbf{u}_{m,k}, \mathbf{u}_{p,k}) \tag{4.31b}$$

$$\mathbf{x}_{1,0} = \mathbf{x}_{1,0} \tag{4.31c}$$

$$\mathbf{x}_{i,0} = \mathbf{0}, \text{ with } i \in \{2,3,7,8\} \tag{4.31d}$$

$$x_{i,0} = 0, \text{ with } i \in \{4,5,6\} \tag{4.31e}$$

$$\mathbf{x}_{i,N} = \mathbf{0}, \text{ with } i \in \{2,3,7,8\} \tag{4.31f}$$

$$\mathbf{p}_{EE,f} = \mathbf{o}_d \tag{4.31g}$$

$$\mathbf{o}_{EE,f} = \mathbf{o}_d \tag{4.31h}$$

$$\mathbf{x}_{i,min} \leq \mathbf{x}_{i,k} \leq \mathbf{x}_{i,max}, \text{ with } i \in \{1,2,3,7,8\} \tag{4.31i}$$

$$t_{min} \leq t_f \leq t_{max} \tag{4.31j}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

## 4.2 Implementation

The finite-dimensional NLP was solved in MATLAB R2021a. The first step was choosing a solver to solve the designed problem (4.31) and then implementing it with some necessary adjustments. Then, using the obtained waypoints, a smooth trajectory is calculated with the help of piecewise polynomial curves with minimum support, also known as *B-splines*. This approach paves the way for future experimentation and testing of various cost function versions with varying weights for the different optimality criteria.

### 4.2.1 Solver specification

The gradient-based nonlinear programming solver *fmincon* was selected to solve the designed problem. It is defined as

$$\mathbf{X} = fmincon(costfun,\ \mathbf{X}_0,\ \mathbf{A},\ \mathbf{b},\ \mathbf{A}_{eq},\ \mathbf{b}_{eq},\ \mathbf{lb},\ \mathbf{ub},\ nonlcon,\ options), \tag{4.32}$$

with the function $costfun$ to be minimized, $\mathbf{X}_0$ as the initial guess, specified as real vector array, $\mathbf{A}$, $\mathbf{b}$, $\mathbf{A}_{eq}$, $\mathbf{b}_{eq}$ as linear equality and inequality constraints, $\mathbf{lb}$ and $\mathbf{ub}$ as lower and upper bounds on the variables, specified as vector arrays.

The *nonlcon* function sums up the nonlinear constraints, including the inequality and equality constraints. The versatile *options* include, among others, the optimization algorithm, the tolerance of constraint violation, and the maximum number of iterations.

The matrix to be optimized is the $\mathbf{X} \in \mathbb{R}^{N\times 38}$ matrix, where $N$ equals the number of rows and is a variable parameter, depending on the number of waypoints. Having additional waypoints can lead to a more efficient but also more complicated and resource-intensive trajectory, therefore, necessitating more testing. The columns correspond to

- $7 \times 3 = 21$ columns for the manipulator joint positions, velocities, accelerations,
- $2 \times 2 = 4$ columns for the platform wheel velocities and accelerations,
- 3 columns for the platform's position and heading angle,
- 9 columns for the jerk controls,
- 1 column for the time intervals.

The cost function $costfun$ is implemented according to the discrete optimization problem (4.31). To reduce the solver's computational effort, the Jacobian matrix is pre-calculated and implemented in symbolic form. The adjustable weight parameters $w_1$ to $w_4$ were used to adjust the cost function, and the parameter $w_4$ was used to study the impact of the manipulability index on the solution.

The convergence of $fmincon$ and other numerical, nonlinear optimization algorithms is sensitive to the initial guess. The choice of the initial guess determines how fast the algorithm converges to a solution and, for functions with more than one local extrema, to which local optimum it converges.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Based on these arguments, the goal location determines the initial guess $\mathbf{X}_0$. The robot's surroundings are separated into four areas: the original workspace, one segment of 90°, and two segments of 135°. Depending on the location of the goal position, the robot obtains a different initial guess. The areas are illustrated in Figure 4.1.

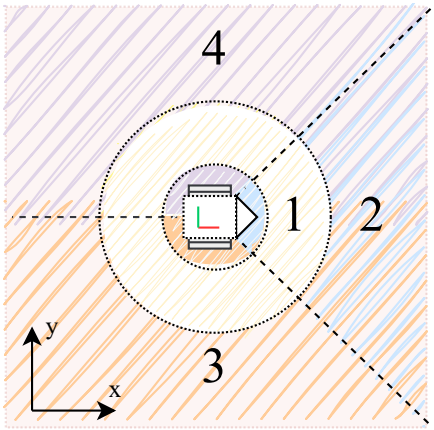


Figure 4.1: Different initial guess depending on the goal position, marked with four segments.

The segments are:

- **Segment 1**: If the goal pose is located in the initial workspace area of the robot, it can be assumed that the robot will not move away from its initial position unless the optimization is subject to a high level of expected manipulability at the final configuration.
- **Segment 2**: If the goal is located in this area, it is reasonable to assume that the robot must move the platform forward; hence, the wheel control inputs get an equal, modest initial value.
- **Segment 3**: If the goal is located in this area, it is realistic to expect the robot to turn right. Therefore, the left wheel's control input is higher than the right wheel's. For the robot to reach the rear region behind the workplace, it must turn since the platform cannot drive backward.
- **Segment 4**: In this segment, a left turn is intended, thereby increasing the right wheel's initial control values.

The lower bounds **lb** and upper bounds **ub** correspond to the system's physical limitations, in particular, the velocity and acceleration limits and the joint position restrictions. The time intervals are also bounded between plausible boundaries to help the solver to converge to a solution faster.

The *nonlcon* function aggregates all equality and inequality constraints, including the system kinematics and the requirement to reach the goal position and orientation. Achieving the desired goal pose requires additional consideration. The end-effector's position must equal the desired target position at the last waypoint, which is the translation vector of (2.43).

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Reaching the desired final orientation is a more challenging task, as even though expressing the desired orientation is more intuitive in Euler angles, they have discontinuities that vary depending on the used convention. The Euler angles can be converted into a rotation matrix, as described in Subsection 2.1.3. However, the matrix representation is redundant as only four of its nine elements are independent.

The alternative approach employs quaternions to avoid the rotation matrix as a set of equality constraints. The lack of discontinuities significantly benefits quaternions over Euler angles and rotation matrices. Another advantage is the faster calculation of the distance between two quaternions.

Thus, the explicit requirement that the rotation matrices or the Euler angles be identical is not part of the orientation equality condition. Instead, at the end of the movement, the distance between the desired quaternion $\mathbf{o}_d$ and the end-effector orientation quaternion $\mathbf{o}_{EE,f}$ must equal zero, as defined in Section Section 4.1. The in-built MATLAB function *dist* computes the distance between the two quaternions, which is measured as the minimum angle of rotation between the two quaternions, expressed as the magnitude of the difference. The result is a number between 0 and $\pi$.

Accordingly, the constraint regarding the desired orientation at $t_f$ is implemented as

$$\text{dist}(\mathbf{o}_d, \mathbf{o}_{EE,f}) = 0. \tag{4.33}$$

The solver is given specification choices in the form of the optimization *options*. This approach requires explicit specification of the optimization algorithm, the maximum number of permitted function evaluations, and the bound-honoring.

The solution algorithm must be chosen carefully (interior-point, trust-region-reflective, sequential quadratic programming, or active-set). Each offers unique benefits and drawbacks, e.g., available gradient or Hessian information, depending on the scale size.

The chosen solver algorithm is the *sequential quadratic programming* (*SQP*) algorithm, which is a state-of-art approach under the nonlinear programming methods. SQP is a good choice as it is vital to use a strategy that respects the constraints at all iterations, offers a reasonably trustworthy certificate of infeasibility, and can benefit from a good initial guess. Each major iteration approximates the Hessian of the Lagrangian function by using a quasi-Newton updating approach which is then utilized to create a QP subproblem. The result of the subproblem serves as the search direction for a line search technique.

The option *HonorBounds* was set to *true* so that the algorithm always honors the bounds. To assure better outcomes, the parameter *MaxFunctionEvaluations* was also increased from its default value to a higher value, $8e+04$.

Due to the high level of complexity, it was not feasible to express the manipulability index, the cost function, the gradients of the constraints, and the Hessian matrix symbolically.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

### 4.2.2 Trajectory generation

The next step is to create a smooth trajectory using the data. After fmincon solves the discrete optimization problem, the matrix **X** delivers all the necessary information for the trajectory waypoints. It is critical since a hypothetical controller must provide the control inputs in considerably shorter time intervals.

Robots are frequently required to perform tasks that require configuration profiles with complicated geometries. They are typically described by a set of waypoints. After the solver determines the waypoints and the end time, the trajectory is equally sampled between the start and end time. As a result, the objective is to generate a piecewise cubic B-spline trajectory for the waypoints. Because of their versatility, spline functions are widely utilized in robot trajectory planning, among many other applications.

Splines of the so-called B-form, or B-splines, are frequently favored because they are significantly simpler to compute, enabling quick and easy local adjustments without recomputing the entire trajectory. Using B-spline trajectories with specified time intervals between waypoints allows for creating more through points while maintaining the integrity of the remaining trajectory so that local adjustments can be easily performed.

### 4.2.3 Visualization and animation

Visualizing the end-effector's movement in Cartesian space is critical, as visual feedback is essential for understanding how the modeled robot performs. The MATLAB Toolbox supports displaying rigid-body systems. It is challenging to determine whether the results are correct based solely on the resulting location, velocity, and acceleration profiles. Therefore, to directly illustrate the behavior of the mobile manipulator, the generated trajectories were played as a video.

First, interpolated trajectories were displayed between arbitrary start and end configurations, as Figure 4.2 illustrates.

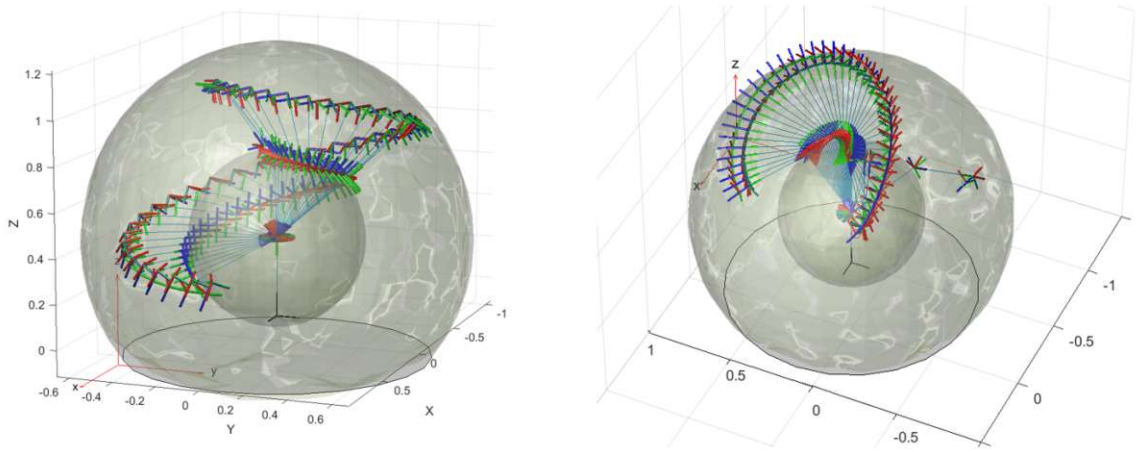


Figure 4.2: Interpolated manipulator trajectories between start and end configurations.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

It can be seen that using simple interpolation to generate trajectories for manipulators does not yield optimal results, often resulting in excessive and unnecessary movements. Furthermore, achieving goal poses located within the inner sphere or beyond the manipulator's workspace is not possible without repositioning the manipulator's base, which can be accomplished through the aid of a mobile platform.

Simulations provide a simpler way to identify faulty behavior and highlight variations among optimization solutions. Therefore, the conversion of the platform into a rigid-body structure, as described in Section 2.3, allowed the simulation of the complete mobile manipulator in MATLAB and provided insight into the calculated trajectories. This modification facilitated the visualization of the platform's movement during the simulations. As depicted in Figure 4.3, the black dots represent the platform's trajectory.

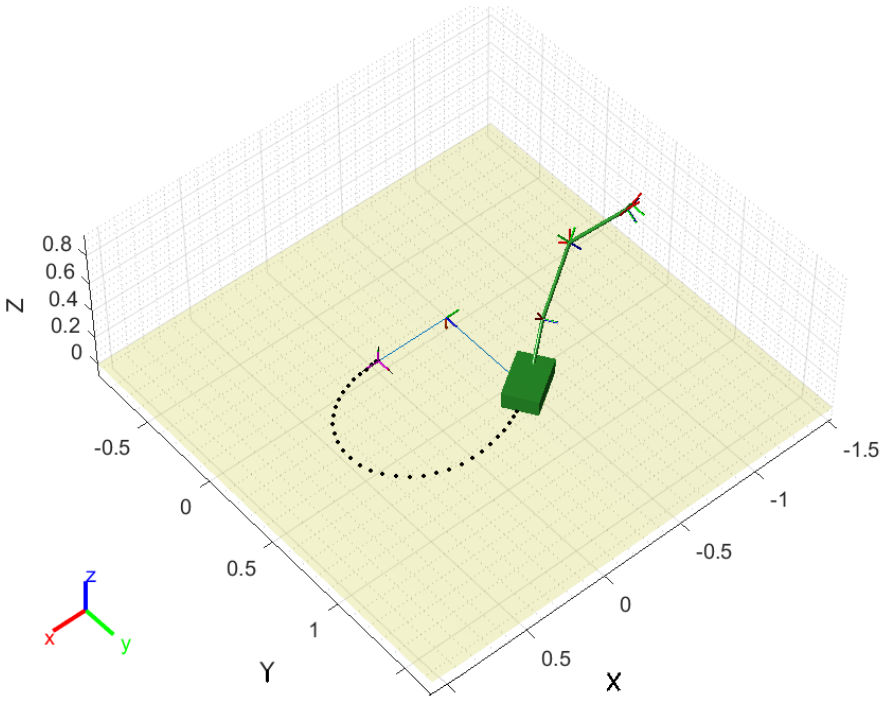


Figure 4.3: The mobile manipulator model with the platform trajectory depicted.

## 4.3 Case studies and results

The main objective of this section is to evaluate the described discrete optimization and trajectory planning results, in particular, the impact of the different weightings of the manipulability index in the cost function.

Two assumptions are made at this point. First, it is assumed that moving the platform instead of the manipulator requires more energy since the platform is heavier than the arm. Second, the manipulator is generally more accurate than the platform, allowing it to position and orient the end-effector precisely. These assumptions suggest that the arm should lead the movement, and the platform should only move when necessary.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

The jerk-based optimization favors the manipulator, resulting in limited movement of the platform. This can cause the arm to remain fully stretched near the edge of the workspace or in a disadvantageous posture. It is undesirable, though, as one would anticipate that the desired pose is selected with the understanding that the robot would subsequently need to perform additional tasks near the target. This issue is demonstrated in Figure 4.4, where different results are obtained depending on the initial robot position, some of which are visibly inadequate.

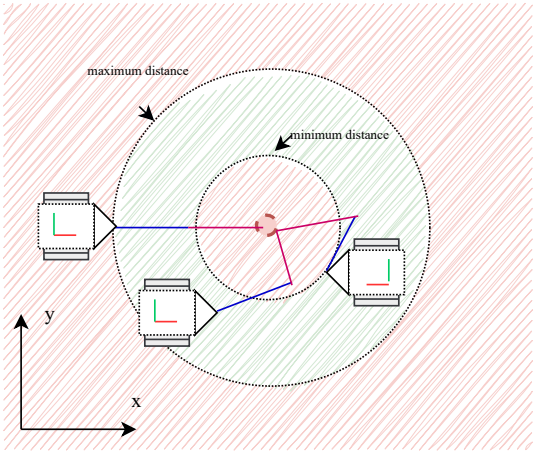


Figure 4.4: Depending on the current position where the platform stays, the manipulator arm has different possibilities to reach the desired goal.

One would like to use the arm to achieve the desired position rather than moving the platform, but in a manner that leaves the robot in the desired pose with a high degree of dexterity at the end of the movement. Thus, if the platform needs to move from its original position, it should do so without stopping at the workspace's boundary so that the manipulator can eventually achieve a more beneficial configuration.

The goal is to analyze when it makes sense to move the platform and when it is sufficient to move with the arm, which, of course, highly depends on the given task. Consider a scenario when the target pose is already in the workspace. The robot would not move the platform, even if the final configuration is excessively disadvantageous. By including manipulability in the cost function, the platform will move to achieve a more favorable final posture.

To investigate the aforementioned considerations, five distinct scenarios have been identified that can serve as test cases. Each scenario provides valuable insights into the system's behavior and capabilities, which are presented in Table 4.1.

The scenarios were tested by increasing the weight $w_4$ assigned to the manipulability criterion in the cost function from $w_4 = 0$ to $w_4 = 1000$. The hypothesis suggests that as the manipulability criterion increases, the platform will move more, and the duration of the movement will increase.

It is also anticipated that the higher importance of the manipulability criterion will result in increased energy consumption, longer execution times, and higher computational costs of the optimization algorithm.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

| | Description | $x$,$y$,$z$ in $m$ | | | $\alpha$,$\beta$,$\gamma$ in $deg$ | | |
|---|---|---|---|---|---|---|---|
| Case 1 | goal is outside the workspace (front) | 1.2 | -1.0 | 0.8 | -34° | -120° | 50° |
| Case 2 | goal is is outside the workspace (back) | -1.2 | 0.5 | 0.8 | -34° | 120° | 50° |
| Case 3 | goal is in immediate proximity | 0.1 | 0.1 | 0.4 | -34° | 120° | 50° |
| Case 4 | goal is in the front part of the workspace | 0.9 | 0.0 | 0.7 | -34° | 120° | 50° |
| Case 5 | goal is in the back part of the workspace | 0.0 | 0.0 | 0.8 | -34° | -120° | 50° |

Table 4.1: Selected study cases with description and desired pose.

Figure 4.5 illustrates the relative position of the selected goal poses with respect to the workspace, as viewed from above and from the side.

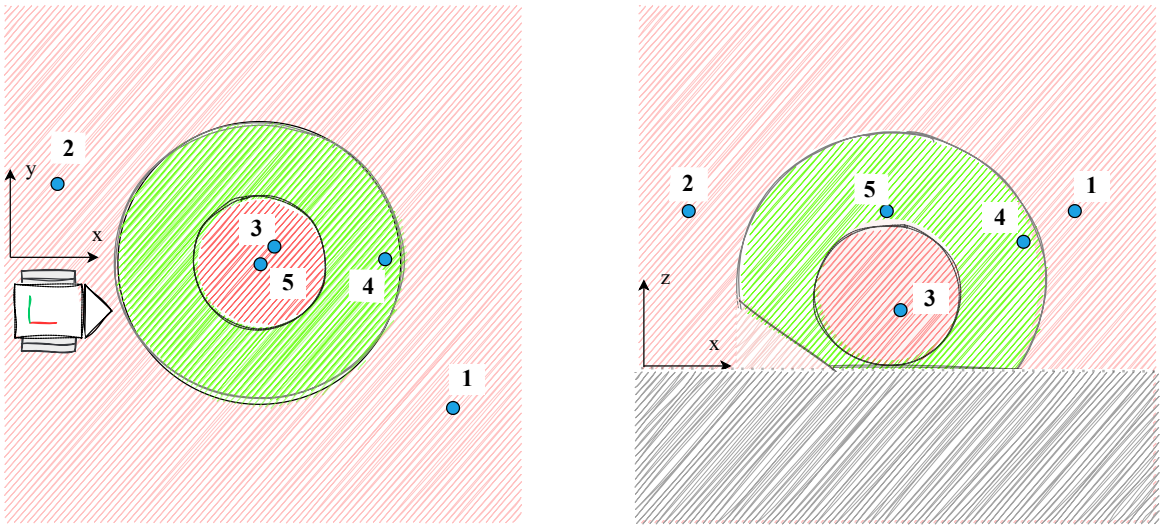


Figure 4.5: Sketches of the relative position of the selected goal poses with respect to the workspace, as viewed from above and from the side.

The desired pose and orientation are specified as constraints in the optimization problem. Due to solver tolerance, achieving the exact desired values may not always be possible. Instead, the optimization problem is solved subject to a certain level of error or tolerance. It is crucial to consider the impact of this error on the overall system performance and ensure it remains within acceptable limits.

The observed factors are:

1. *orientation error*: the distance between the end-effector's desired orientation and its final orientation in degrees.
2. *position error*: the Euclidean distance between the end-effector's desired position and its final position in m.
3. *platform effort*: the share of the cost function due to the platform's movement:

$$\sum_{i=0}^{N-1} \mathbf{u}_{p,i}^T \mathbf{u}_{p,i}. \tag{4.34}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

4. *manipulator effort*: the share of the cost function due to the manipulator's movement:
$$\sum_{i=0}^{N-1} \mathbf{u}_{m,i}^T \mathbf{u}_{m,i}. \tag{4.35}$$
5. *task execution time (in s)*: the required time $t_f$ to execute the movement of the mobile manipulator as in (4.21).
6. *manipulability measure*: the Yoshikawa manipulability measure with $\mathbf{J}_{G,f}$ as the Jacobian matrix at the final time $t_f$ as in (4.15).
7. *evaluation time*: the time the solver requires to find a solution, in s.
8. *iteration number*: number of iterations which the solver needs.

The scenarios were designed with a fixed number of waypoints $N = 15$, while additional tests were conducted with different numbers of waypoints that are not presented in this work. In the simulation results, the start and final poses of the end-effector are distinguished by turquoise and orange colors, respectively.

### 4.3.1 Case study 1

For the first case study **C1**, the desired goal is outside of the initial workspace, requiring the mobile robot to drive away from the initial position:

| | | | |
|---|---|---|---|
| Goal position (m) | $x = 1.2$ | $y = -1.0$ | $z = 0.8$ |
| Goal orientation (°) | $\alpha = -34°$ | $\beta = -120°$ | $\gamma = 50°$ |

The expected behavior is, excluding the high manipulability criterion, that the robot tries to move the platform as little as possible, resulting in a completely stretched arm, ending the movement as soon as possible.

When the manipulability criterion is included with an increasing $w_4$, it is expected that the execution time elongates as the platform drives closer to the end pose and that the manipulability value increases as well. Table 4.2 summarizes the results.

| $w_4$ | 0 | $\frac{1}{100}$ | $\frac{1}{10}$ | 1 | 2 | 5 | 10 | 1000 |
|---|---|---|---|---|---|---|---|---|
| Δ orientation (°) | 2.3425 | 2.3388 | 2.4859 | 5.1761 | 6.1225 | 6.4470 | 6.0298 | 6.6349 |
| Δ position (m) | 0.0054 | 0.0055 | 0.0058 | 0.0062 | 0.0061 | 0.0056 | 0.0287 | 0.0324 |
| platform effort | 0.4850 | 0.4930 | 0.5350 | 0.6080 | 0.6210 | 0.6370 | 1.3060 | 1.5150 |
| manipulator effort | 0.1910 | 0.1823 | 0.2085 | 0.1698 | 0.1837 | 0.1838 | 0.1869 | 0.0806 |
| exec. time(s) | 7.9192 | 7.9206 | 7.9592 | 8.1656 | 8.2325 | 8.2726 | 10.055 | 10.327 |
| manipulability | 0.0594 | 0.0708 | 0.1187 | 0.1764 | 0.1819 | 0.1835 | 0.1835 | 0.1835 |
| eval. time (s) | 23.68 | 38.28 | 38.28 | 37.25 | 35.48 | 34.89 | 33.42 | 43.18 |
| iteration # | 65 | 67 | 64 | 61 | 55 | 55 | 52 | 72 |

Table 4.2: C1 results with the desired pose located beyond the initial workspace but still positioned ahead of the robot.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

As the results show, the more weight is given to the manipulability, the more energy the platform requires compared to the manipulator, as the base has to travel further. More significant are the results of the execution time and the manipulability, as shown in Figure 4.6 on a logarithmic scale.

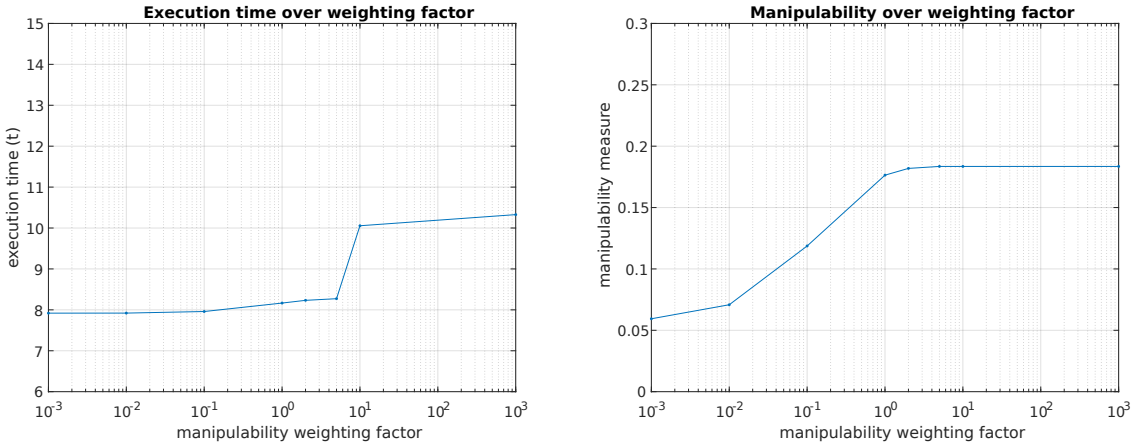


Figure 4.6: C1 execution time and Yoshikawa manipulability index over increasing $w_4$.

It can be observed that around the factor of $w_4 = 1 - 5$, the manipulability starts to reach its local maximum, and the curve gets saturated. Choosing an appropriate weighting factor is decisive and should be selected to improve the final posture while not significantly deteriorating the execution time and energy profiles.

A visual comparison between two scenarios with the excluded manipulability criterion $w_4 = 0$ in Figure 4.7 and a suitably included $w_4 = 2$ in Figure 4.8 showcase this result.

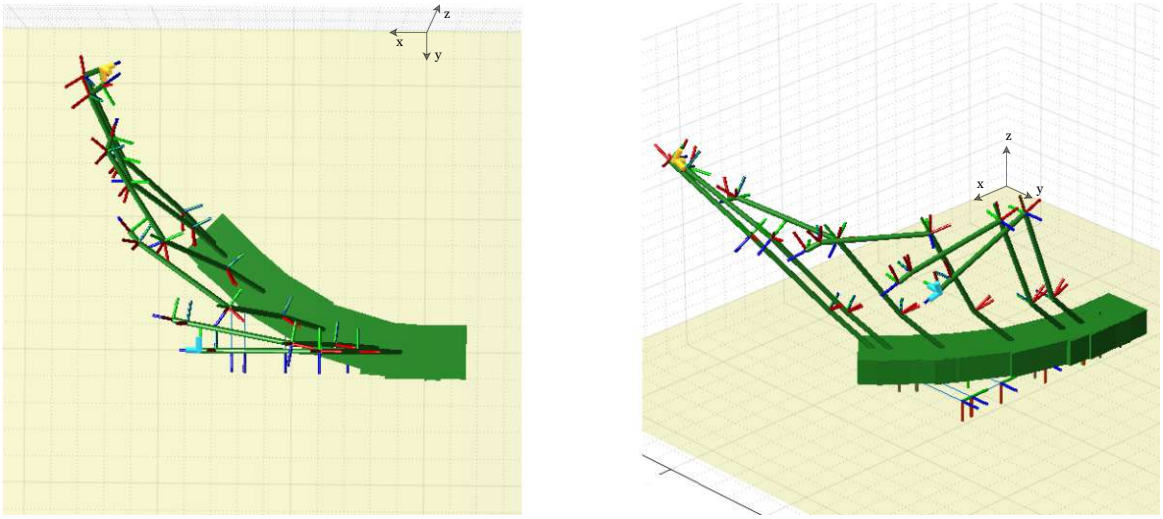

Figure 4.7: C1 trajectory results with $w_4 = 0$ weighting of the manipulability criterion.

In the case of $w_4 = 0$, the robot ends the movement with a stretched configuration, which is not beneficial for subsequent activities.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

In comparison, with $w_4 = 2$, the robot reaches a beneficial posture while still having only slightly increased execution time.

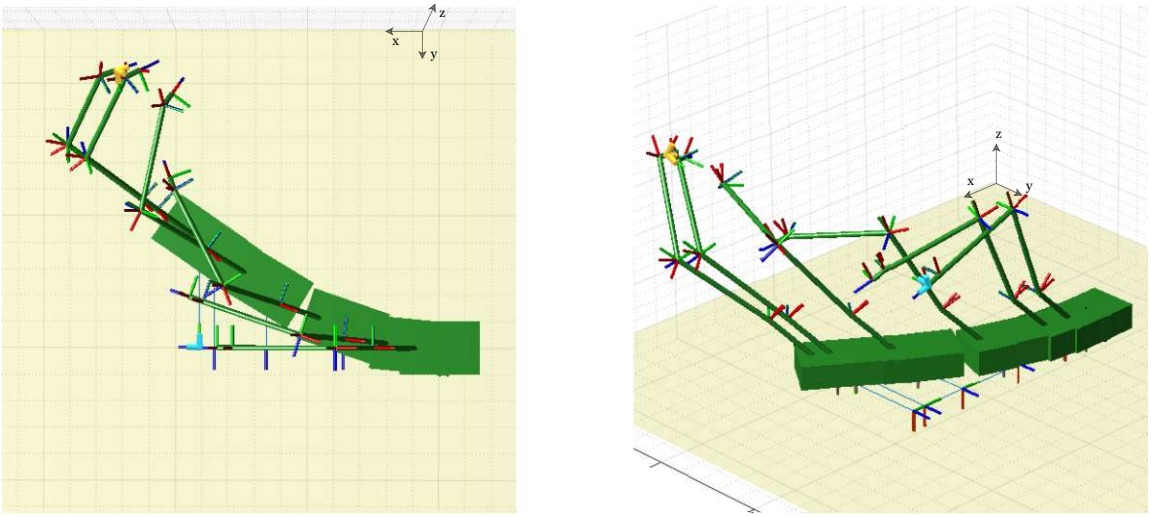

Figure 4.8: C1 trajectory results with $w_4 = 2$ weighting of the manipulability criterion.

For the case $w_4 = 2$, Figure 4.9 illustrates that the manipulator has to execute large joint changes to reach a high manipulability at the end-configuration.

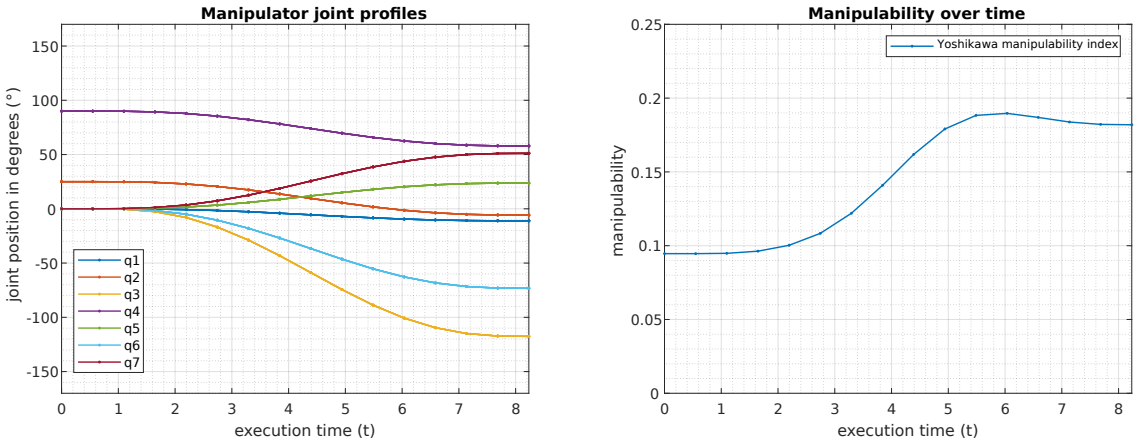


Figure 4.9: C1 joint profiles and Yoshikawa manipulability index over time for $w_4 = 2$.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

### 4.3.2 Case study 2

For the second case study **C2**, the desired pose was chosen outside of the initial workspace, behind the robot, requiring the platform to turn around before reaching it:

| | | | |
|---|---|---|---|
| Goal position (m) | $x = -1.2$ | $y = 0.5$ | $z = 0.8$ |
| Goal orientation (°) | $\alpha = -34°$ | $\beta = 120°$ | $\gamma = 50°$ |

This is a demanding task as the base cannot reverse and must execute an almost 180° turn in order to reach the goal pose. As a result, it is anticipated that the robot will require additional execution time to perform the turn and move the platform accordingly. The results are presented in Table 4.3.

| $w_4$ | 0 | $\frac{1}{100}$ | $\frac{1}{10}$ | 1 | 2 | 5 | 10 | 1000 |
|---|---|---|---|---|---|---|---|---|
| $\Delta$ orientation (°) | 6.4863 | 6.4440 | 2.1278 | 4.2918 | 3.5759 | 2.3034 | 3.3278 | 6.6560 |
| $\Delta$ position (m) | 0.0293 | 0.0294 | 0.0300 | 0.0326 | 0.0333 | 0.0352 | 0.0368 | 0.0110 |
| platform effort | 1.3140 | 1.3230 | 1.3390 | 1.3770 | 1.3990 | 1.4940 | 1.5480 | 3.0110 |
| manipulator effort | 0.0454 | 0.0383 | 0.0476 | 0.1336 | 0.1379 | 0.1277 | 0.1284 | 0.0211 |
| exec. time(s) | 9.7196 | 9.7231 | 9.7811 | 10.068 | 10.124 | 10.331 | 10.445 | 12.698 |
| manipulability | 0.0095 | 0.0300 | 0.0895 | 0.1953 | 0.2020 | 0.2049 | 0.2076 | 0.2102 |
| eval. time (s) | 55.41 | 69.05 | 65.28 | 49.76 | 50.75 | 41.43 | 40.68 | 40.88 |
| iteration # | 76 | 69 | 78 | 85 | 70 | 66 | 50 | 66 |

Table 4.3: C2 results with the desired pose located outside and behind the initial workspace.

The results confirm that as the weight of the manipulability criterion increases, the platform's traveling distance also increases, leading to improvements in manipulability and final posture. Figure 4.10 clearly shows that the execution time suffers a significant increase after the manipulability weighting factor $w_4$ exceeds 2.

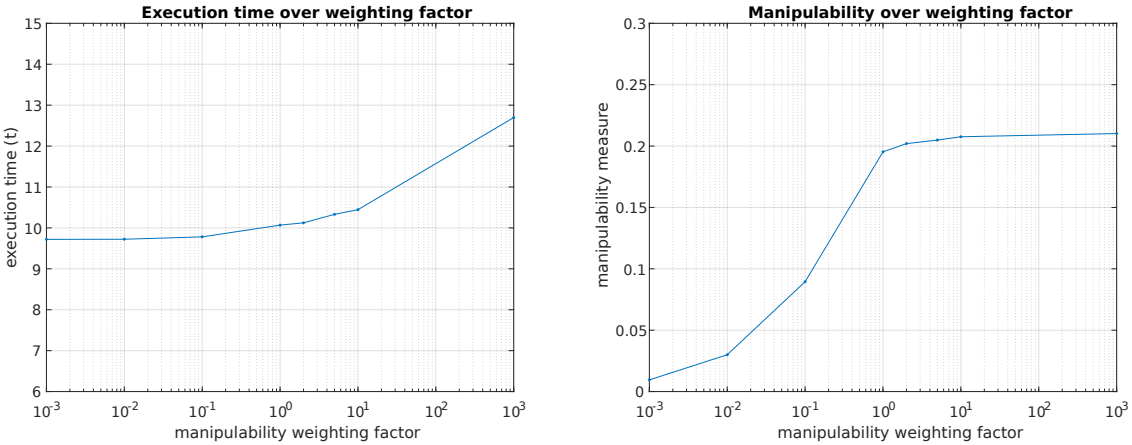


Figure 4.10: C2 execution time and Yoshikawa manipulability index over increasing $w_4$.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

A visual comparison between two scenarios with $w_4 = 0$ and $w_4 = 2$ showcases the conclusion that increasing the manipulability weight improves the final posture. As shown in Figure 4.11, a trajectory with $w_4 = 0$ from the side and upper view demonstrates the need for improvement.

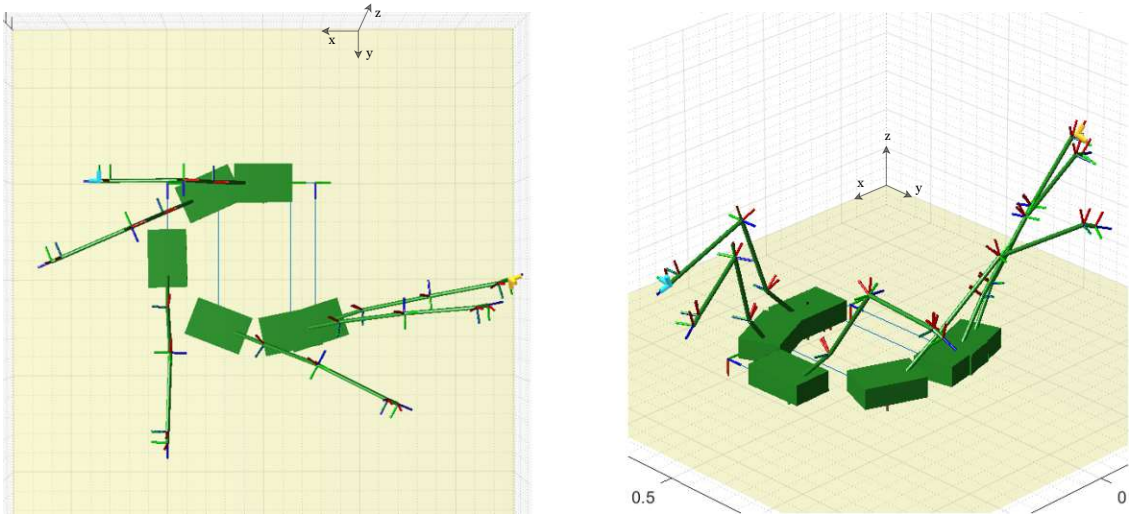


Figure 4.11: C2 trajectory results with $w_4 = 0$ weighting of the manipulability criterion.

In comparison, Figure 4.12 illustrates the final trajectory and configuration for $w_4 = 2$. As evident from the figure, the end-effector reaches a more desirable configuration.

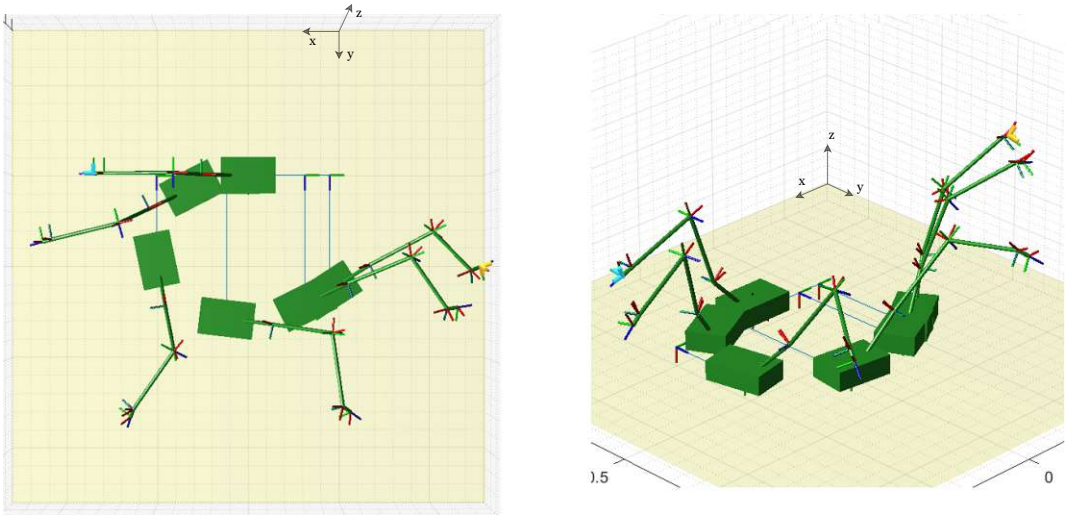


Figure 4.12: C2 trajectory results with $w_4 = 2$ weighting of the manipulability criterion.

Furthermore, as Figure 4.13 shows, the resulting trajectory is smooth, and the final posture reaches a high level of manipulability. Additionally, the joint trajectories remain within their respective limits, indicating that the robot's motion is feasible and safe.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

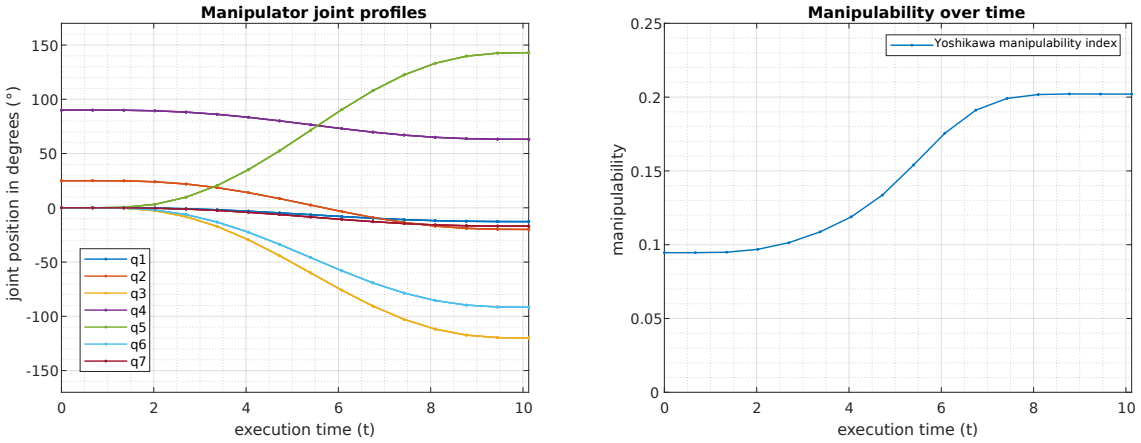


Figure 4.13: C2 joint profiles and Yoshikawa manipulability index over time for $w_4 = 2$.

### 4.3.3 Case study 3

The third case study, **C3**, examines the scenario where the desired pose is located within the inner, non-reachable sphere of the initial workspace, which is in the immediate vicinity of the robot's body. This requires the robot to move away from its initial position to reach the goal.

| | | | |
|---|---|---|---|
| Goal position (m) | $x = 0.1$ | $y = 0.1$ | $z = 0.4$ |
| Goal orientation (°) | $\alpha = -34°$ | $\beta = 120°$ | $\gamma = 50°$ |

The results of the experiment are presented in Table 4.4.

| $w_4$ | 0 | $\frac{1}{100}$ | $\frac{1}{10}$ | 1 | 2 | 5 | 10 | 1000 |
|---|---|---|---|---|---|---|---|---|
| $\Delta$ orientation (°) | 7.4340 | 7.4804 | 7.4229 | 2.6456 | 2.2461 | 2.1745 | 2.4046 | 6.4013 |
| $\Delta$ position (m) | 0.0056 | 0.0056 | 0.0058 | 0.0072 | 0.0038 | 0.0017 | 0.0364 | 0.1364 |
| platform effort | 0.1300 | 0.1400 | 0.1750 | 0.4450 | 1.0540 | 1.1580 | 1.5470 | 4.2750 |
| manipulator effort | 0.4727 | 0.4583 | 0.4170 | 0.4424 | 0.1739 | 0.1544 | 0.1815 | 0.0218 |
| exec. time(s) | 7.8252 | 7.8289 | 7.8364 | 8.5453 | 9.4579 | 9.6726 | 10.540 | 14.593 |
| manipulability | 0.0871 | 0.0921 | 0.1039 | 0.1474 | 0.2265 | 0.2403 | 0.2436 | 0.2469 |
| eval. time (s) | 24.24 | 54.68 | 49.24 | 58.36 | 49.41 | 42.89 | 37.68 | 52.69 |
| iteration # | 79 | 85 | 79 | 95 | 83 | 68 | 59 | 83 |

Table 4.4: C3 results with the desired pose located within the initial workspace's inner, non-reachable sphere.

When the desired goal pose is in close proximity to the robot, the manipulator needs to move away from its initial position. Increasing the weight of the manipulability criterion in the cost function enhances manipulability but at the expense of higher energy consumption and execution time. However, the manipulability plateaus beyond a certain weight, as depicted in Figure 4.14, while energy and execution time keep increasing without notable improvement in the final configuration.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

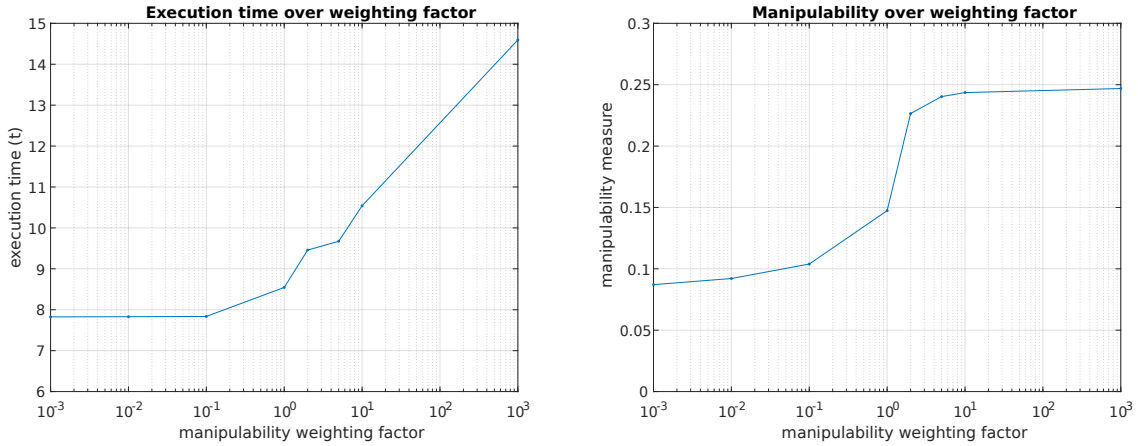


Figure 4.14: C3 execution time and Yoshikawa manipulability index over increasing $w_4$.

Increasing the manipulability criterion's weight in the cost function necessitates larger movements, with the robot turning around almost completely, resulting in a considerable increase in task execution time. Simulations confirm this behavior, showing that when $w_4 = 0$, the manipulator ends up in a final posture close to a singular configuration, as illustrated in Figure 4.15.

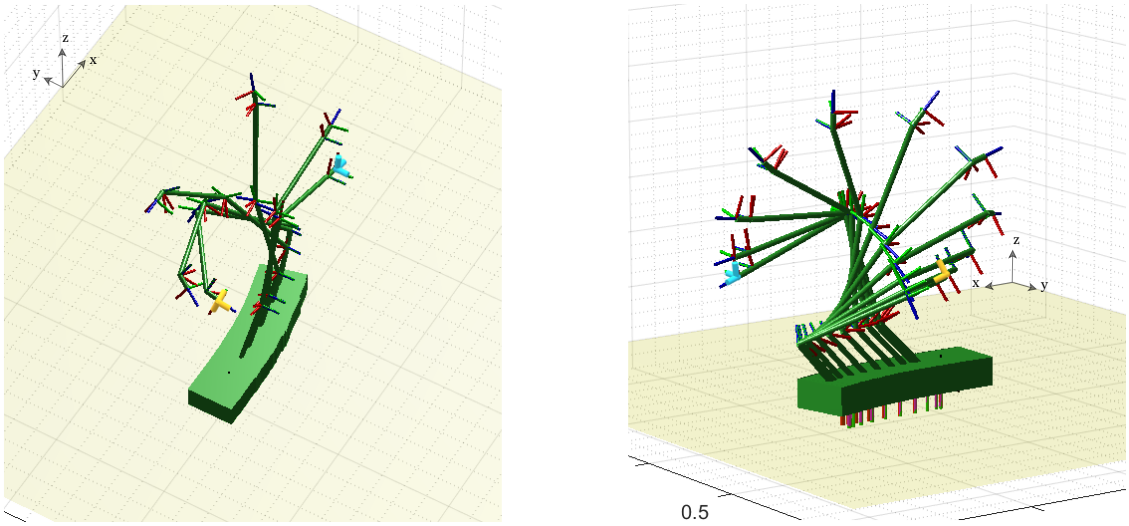


Figure 4.15: C3 trajectory results with $w_4 = 0$ weighting of the manipulability criterion.

When the manipulability is included in the cost function with a relative weight of $w_4 = 2$, the manipulator's movement results in the robot turning almost 180°, significantly increasing the task execution time and platform movement. Despite the resulting increase in task execution time, the final configuration enables the robot to reach nearby points without requiring the platform to move as significantly, as illustrated by Figure 4.16.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

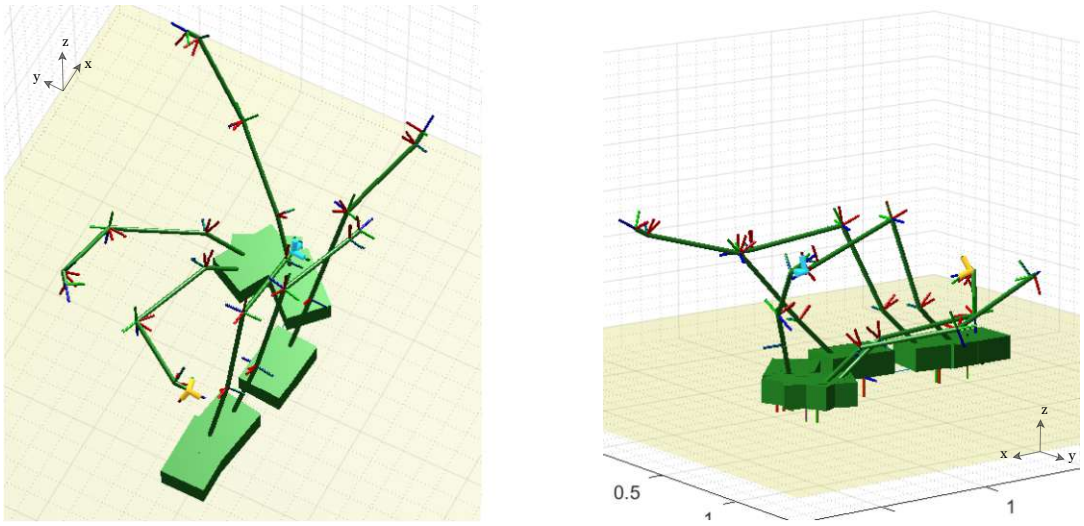


Figure 4.16: C3 trajectory results with $w_4 = 2$ weighting of the manipulability criterion.

Figure 4.17 demonstrates the smooth trajectories of the manipulator's joints for $w_4 = 2$ and visualizes the increased manipulability as a result of the major adjustments made to the base's final orientation.

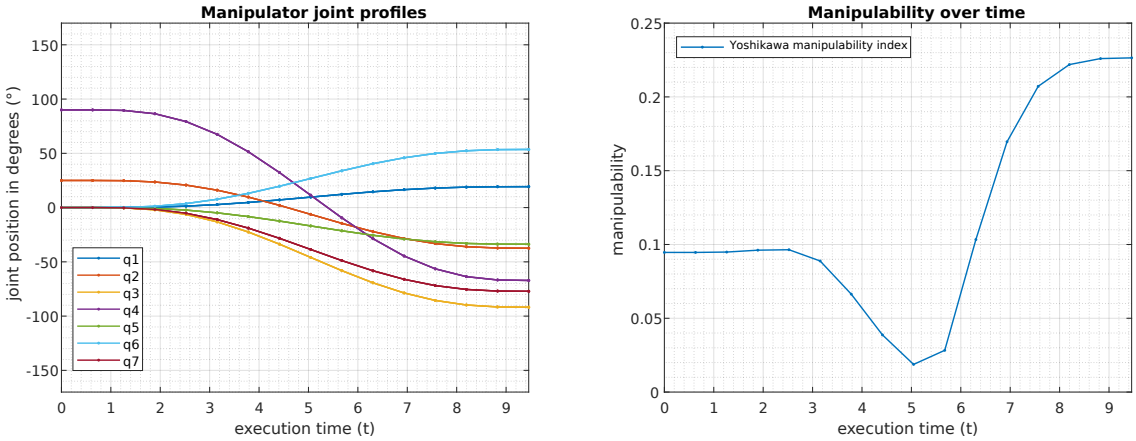


Figure 4.17: C3 joint profiles and Yoshikawa manipulability index over time for $w_4 = 2$.

### 4.3.4 Case study 4

In scenario **C4**, the manipulator is tasked with achieving a goal pose initially located within its initial workspace, where the platform usually remains stationary. However, when the manipulability criterion is included in the optimization process, it may be necessary to move the platform to achieve a better final configuration.

| | | | |
|---|---|---|---|
| Goal position (m) | $x = 0.9$ | $y = 0.0$ | $z = 0.7$ |
| Goal orientation (°) | $\alpha = -34°$ | $\beta = 120°$ | $\gamma = 50°$ |

The results are summarized in Table 4.5.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

| $w_4$ | 0 | $\frac{1}{100}$ | $\frac{1}{10}$ | 1 | 2 | 5 | 10 | 1000 |
|---|---|---|---|---|---|---|---|---|
| $\Delta$ orientation (°) | 6.9524 | 6.9052 | 6.8221 | 5.5672 | 5.0719 | 4.6579 | 4.5331 | 4.4641 |
| $\Delta$ position (m) | 0.0017 | 0.0019 | 0.0024 | 0.0020 | 0.0024 | 0.0029 | 0.0033 | 0.0029 |
| platform effort | 0.0000 | 0.0021 | 0.0110 | 0.0207 | 0.0260 | 0.0320 | 0.0400 | 0.0930 |
| manipulator effort | 0.4348 | 0.4280 | 0.4118 | 0.5133 | 0.5383 | 0.5471 | 0.5455 | 0.5150 |
| exec. time(s) | 6.8914 | 6.8923 | 6.9042 | 7.1962 | 7.2901 | 7.3427 | 7.3641 | 7.4543 |
| manipulability | 0.1083 | 0.1174 | 0.1375 | 0.1720 | 0.1789 | 0.1808 | 0.1811 | 0.1815 |
| eval. time (s) | 21.43 | 41.01 | 38.21 | 40.91 | 38.49 | 36.66 | 33.42 | 35.95 |
| iteration # | 72 | 71 | 66 | 67 | 63 | 59 | 52 | 55 |

Table 4.5: C4 results with the desired pose in the initial workspace's front section.

The execution time required to achieve this goal does not change significantly, even when the cost function includes the manipulability criterion. However, when the weighting factor of the manipulability is increased up to $w_4 = 5$, a significantly better final posture is achieved, as shown in Figure 4.18. This contrasts with the default solution, where the manipulator is solely responsible for achieving the final pose.

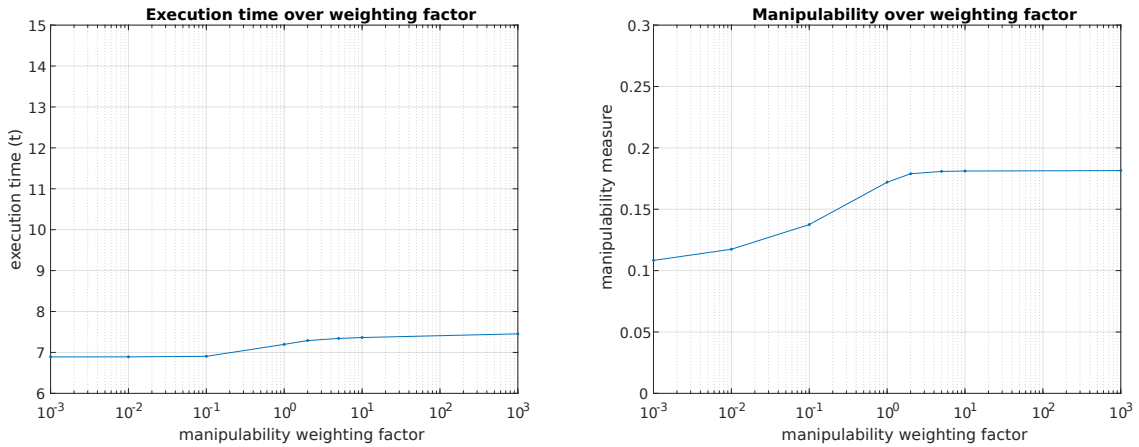


Figure 4.18: C4 execution time and Yoshikawa manipulability index over increasing $w_4$.

Simulations show no significant differences in the robot's motion, as the goal pose is near the initial configuration. Figure 4.19 illustrates the resulting trajectory for $w_4 = 0$.

In contrast, as shown in Figure 4.20, increasing the weighting factor of the manipulability criterion to $w_4 = 2$ results in a final configuration that is not only more favorable but also takes into account the end-effector orientation, which can be beneficial for subsequent tasks.

The resulting joint and manipulability profiles for $w_4 = 2$ are depicted in Figure 4.21.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

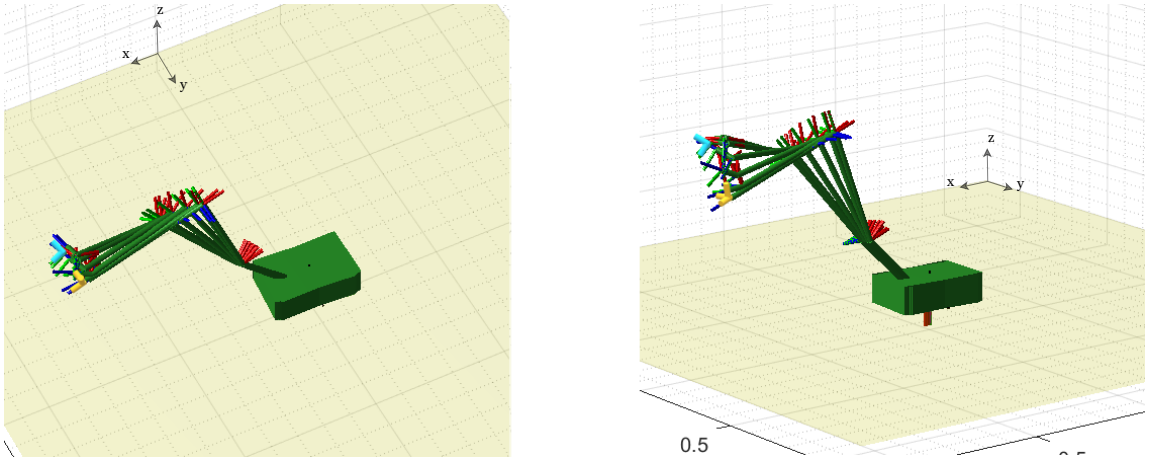


Figure 4.19: C4 trajectory results with $w_4 = 0$ weighting of the manipulability criterion.

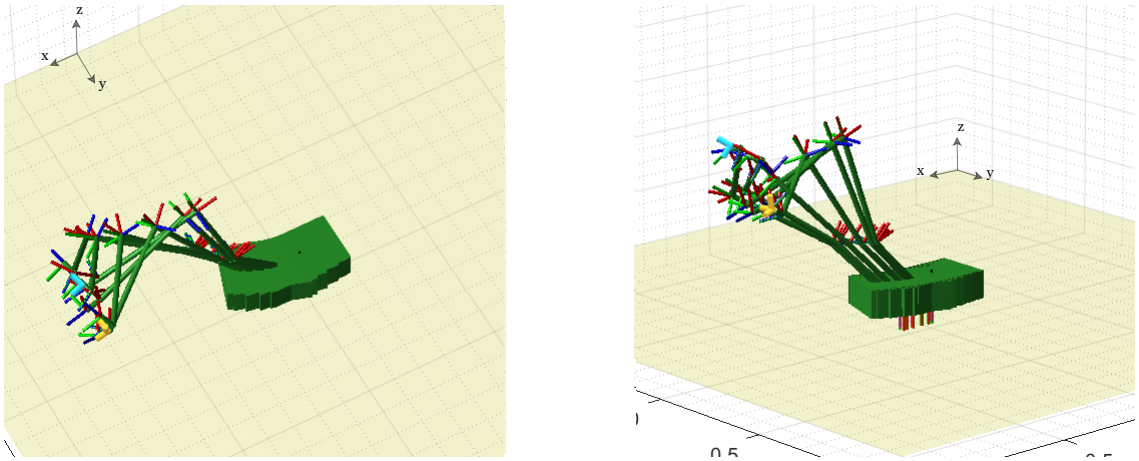


Figure 4.20: C4 trajectory results with $w_4 = 2$ weighting of the manipulability criterion.

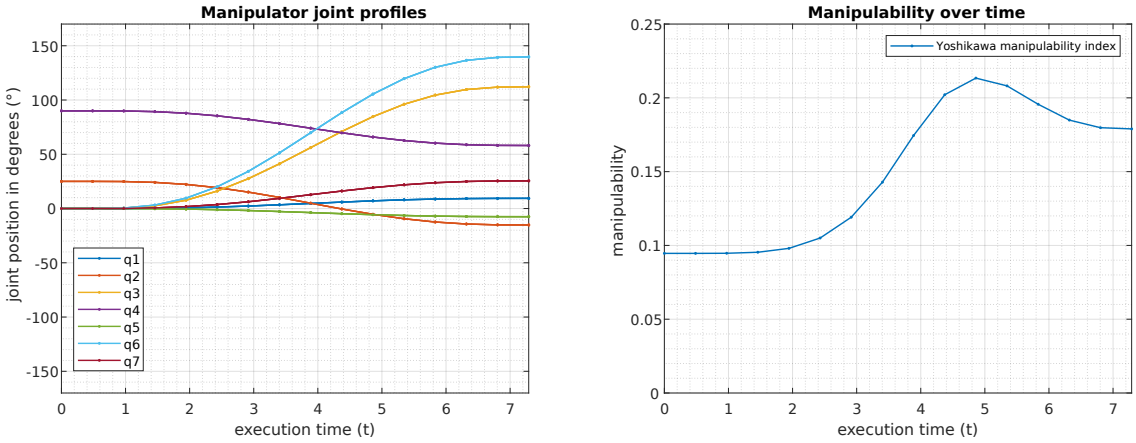


Figure 4.21: C4 joint profiles and Yoshikawa manipulability index over time for $w_4 = 2$.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

### 4.3.5 Case study 5

The final case study **C5** examines the situation where the goal pose is situated in the rear section of the robot's workspace.

| | | | |
|---|---|---|---|
| Goal position (m) | $x = 0.0$ | $y = 0.0$ | $z = 0.8$ |
| Goal orientation (°) | $\alpha = -34°$ | $\beta = -120°$ | $\gamma = 50°$ |

By default, the robot adjusts only its manipulator arm to achieve a desired pose without moving the base. However, this configuration may not be optimal for subsequent tasks. Incorporating the manipulability in the cost function yields a notably improved final configuration, as demonstrated in the results presented in 4.6.

| $w_4$ | 0 | $\frac{1}{100}$ | $\frac{1}{10}$ | 1 | 2 | 5 | 10 | 1000 |
|---|---|---|---|---|---|---|---|---|
| Δ orientation (°) | 3.1696 | 3.2626 | 2.8359 | 5.4614 | 6.2226 | 6.5750 | 6.4275 | 5.7218 |
| Δ position (m) | 0.0017 | 0.0017 | 0.0017 | 0.0017 | 0.0017 | 0.0020 | 0.0065 | 0.0307 |
| platform effort | 0.0000 | 0.0010 | 0.0040 | 0.0510 | 0.0460 | 0.0437 | 0.1810 | 1.8230 |
| manipulator effort | 0.4107 | 0.4139 | 0.3912 | 0.4878 | 0.5063 | 0.5266 | 0.5382 | 0.1076 |
| exec. time(s) | 6.6520 | 6.6532 | 6.7676 | 7.2032 | 7.2574 | 7.3120 | 7.8634 | 10.634 |
| manipulability | 0.0218 | 0.0265 | 0.0330 | 0.0881 | 0.0899 | 0.0907 | 0.0971 | 0.1887 |
| eval. time (s) | 51.97 | 81.11 | 69.85 | 69.15 | 73.13 | 82.32 | 71.58 | 132.1 |
| iteration # | 81 | 83 | 69 | 69 | 77 | 92 | 62 | 119 |

Table 4.6: C5 results with the desired pose located in the rear portion of the workspace.

As shown in Figure 4.22, there is a possibility to improve the final configuration through the inclusion of manipulability. However, this results in a significant increase in execution time, as the base starts to execute a 180° turn. The trajectory for $w_4 = 0$ is illustrated in Figure 4.23, while the beginning of the base turning can be observed in Figure 4.24 due to $w_4 = 2$.

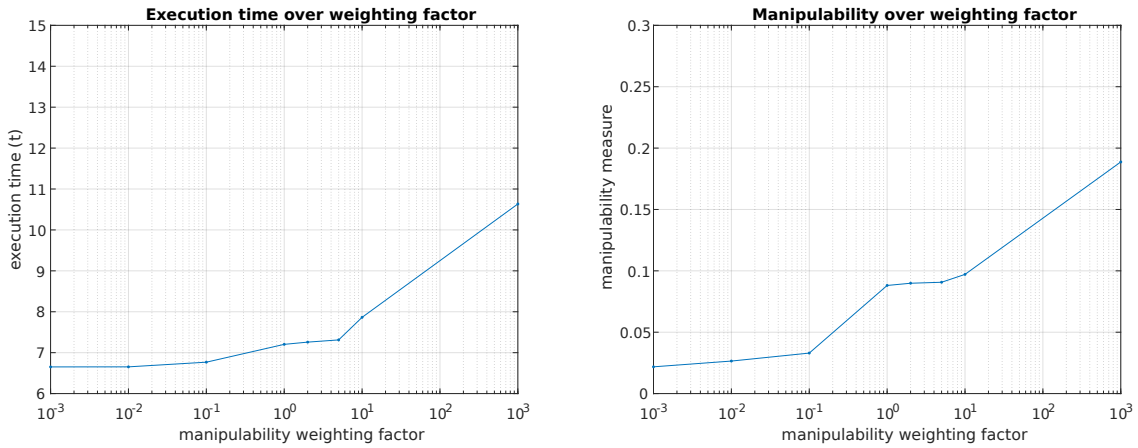


Figure 4.22: C5 execution time and Yoshikawa manipulability index over increasing $w_4$.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

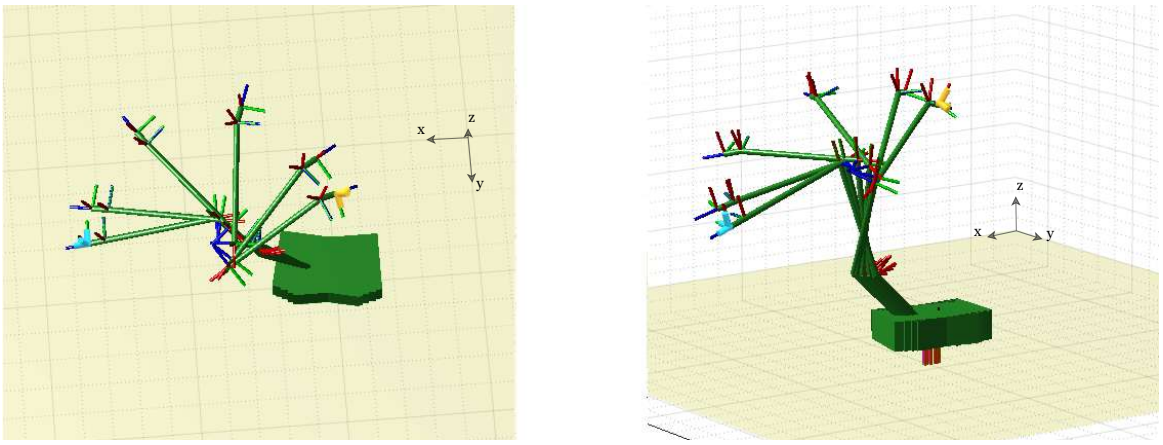


Figure 4.23: C5 trajectory results with $w_4 = 0$ weighting of the manipulability criterion.

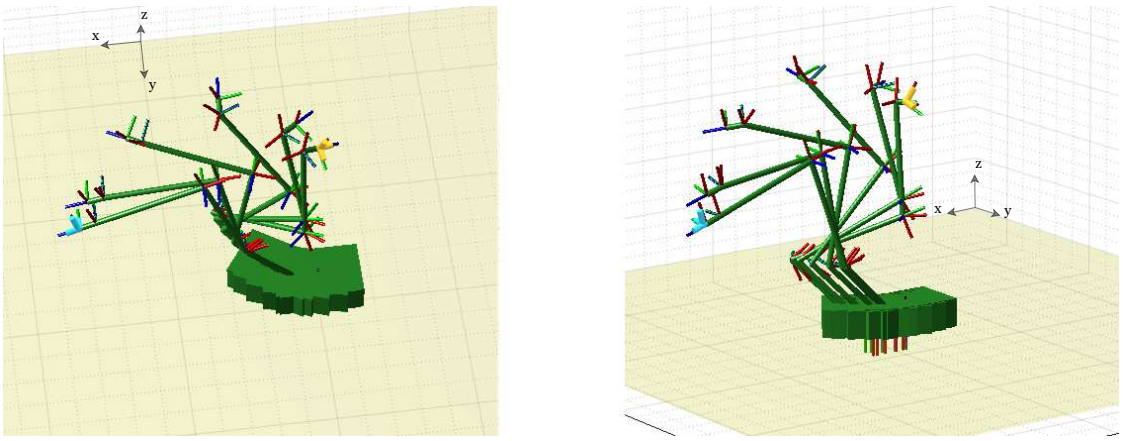


Figure 4.24: C5 trajectory results with $w_4 = 2$ weighting of the manipulability criterion.

Figure 4.25 demonstrates that the selected final Cartesian goal pose results in low manipulability values but still leads to a significant change in joint configuration.

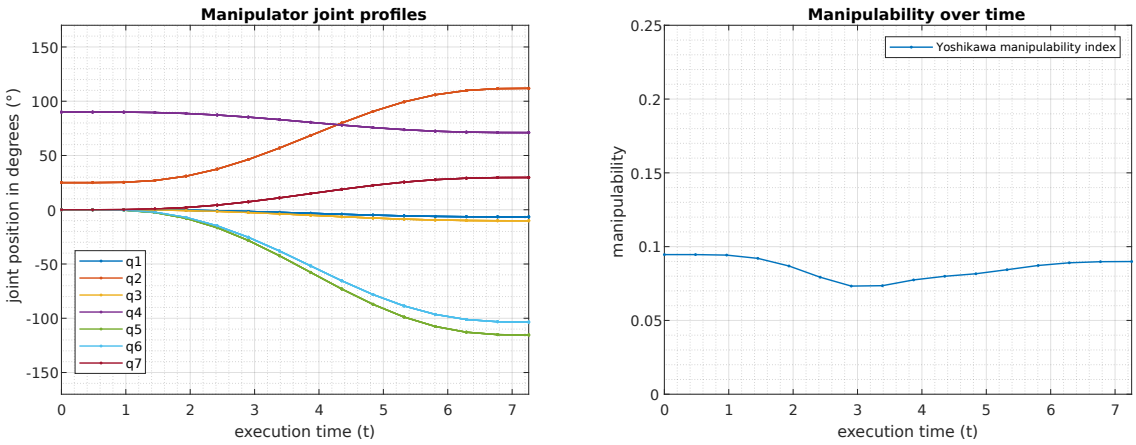


Figure 4.25: C5 joint profiles and Yoshikawa manipulability index over time for $w_4 = 2$.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

# Chapter 5

# Conclusion

This thesis aimed to develop and implement a nonlinear jerk-based trajectory optimization algorithm for a mobile manipulator robot. In real-world scenarios, it is reasonable to assume that a mobile robot may need to execute additional tasks in close proximity to the new position after relocating its base. To achieve this efficiently, trajectory planning must ensure that the robot's final configuration enables the robot to execute subsequent tasks. Accordingly, the primary research question addressed in this thesis is how to generate trajectories that meet these requirements while accounting for the robot's movement characteristics.

By integrating manipulability as a performance index, this thesis demonstrates that it is possible to generate trajectories that guarantee a high manipulability of the manipulator in its final configuration. The results show that the weighting factor for the manipulability in the cost function of the optimization problem plays a critical role in shaping the final configuration and influencing the execution time of the generated trajectories. It was illustrated that the produced trajectories are efficient in terms of energy consumption, control inputs, and achieving the ideal posture.

This work places significant emphasis on developing and implementing the mobile manipulator model, serving as a basis for trajectory planning and optimization. The first step involved the derivation of a mathematical model of the manipulator using kinematic equations based on the Denavit-Hartenberg convention. Transformation and Jacobian matrices were calculated to solve the forward and inverse kinematics problems. This enabled us to determine the singular configurations of the manipulator, which are critical because, at these configurations, the manipulator loses its ability to move in certain directions.

A Monte-Carlo-based method was combined with the forward kinematics solution to sample the manipulator's workspace. The workspace is represented as a bounding hull generated using alpha shape polyhedra. Finally, manipulability measures were introduced, and various performance indices were analyzed to determine the most suitable one for the subsequent optimization problem. It was found that the Yoshikawa manipulability index was the most suitable measure for improving the trajectory planning process, particularly near singular configurations.

The next step was formulating an optimal control problem to generate trajectories for the mobile manipulator. The problem was subject to the system's kinematics and



Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

was designed to minimize a cost function with several criteria: minimizing the total execution time, minimizing the jerk, and maximizing the manipulability index. The problem was discretized, and a numerical method was used to find the optimal solution.

A series of simulations were conducted to validate the effectiveness of the proposed trajectory planning and optimization method. The results show a significant improvement in terms of task completion time and overall efficiency compared to traditional approaches. The analysis reveals a range in which the manipulability of the final configuration significantly improves while the impact on execution time is relatively small, thus suggesting the existence of a trade-off between these criteria. Below this range, the manipulability effect becomes negligible while the execution time increases. Above the range, the manipulability does not change significantly anymore, and the execution time rapidly increases without further benefits. These findings suggest that a careful selection of the corresponding weighting factor based on the conducted tests should be one to five times higher than other terms, substantially impacting the final configuration.

While this work demonstrates promising results, it is important to acknowledge its limitations. The described control problem was tested exclusively on the model, not the actual mobile manipulator, which may introduce biases and discrepancies due to idealized assumptions. Additionally, the baseline scenario assumed a level of precision in joint and wheel control that is not always achievable in practical applications, where system imperfections and estimation errors are common.

Numerous ways to improve the current implementation offer exciting opportunities for future research. One area of focus could be exploring efficient approaches for integrating the heterogeneity of the workspace information into the optimization problem. Another potential improvement could be the cost function, where implementing a method for symbolic manipulability value or an estimation technique with lower computational demand could yield significant progress. Furthermore, non-equidistant time segments and flexible time constraints could be another approach. While equidistant time segments were ultimately selected for simplicity and computational efficiency, alternative approaches could be explored in future work.

Efforts were made to optimize the computational time of the trajectory. One approach considered was the manual definition of gradients and the Hessian matrix for the cost function and constraints. However, due to its high computational demands, this method was not practical within the scope of the thesis, and the current implementation is unsuitable for real-time solutions. Finally, testing the current solution on a physical robot would provide valuable insights into its ability to replicate real-world behaviors.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

# Appendix

## A1 Symbolic transformation matrix expressions

The symbolic transformation matrix expressions of the $\boldsymbol{t}_{\boldsymbol{0}}^{\boldsymbol{7}}$ and the rotation matrix $\boldsymbol{R}_{\boldsymbol{0}}^{\boldsymbol{7}}$ of the transformation matrix (2.15), with the abbreviations $\sin(\theta_i) = s_i$ and $\cos(\theta_i) = c_i$.

$$\boldsymbol{t}_{\boldsymbol{0}}^{\boldsymbol{7}} = \begin{bmatrix} \frac{21c_1s_2}{50} + \frac{3c_6\sigma_{10}}{50} - \frac{3s_6\sigma_9}{50} + \frac{2s_4\sigma_{19}}{5} + \frac{2c_1c_4s_2}{5} \\ \frac{21s_1s_2}{50} - \frac{3c_6\sigma_8}{50} - \frac{2s_4\sigma_{18}}{5} + \frac{3s_6\sigma_7}{50} + \frac{2c_4s_1s_2}{5} \\ \frac{21c_2}{50} + \frac{2c_2c_4}{5} + \frac{3s_6\sigma_{11}}{50} + \frac{3c_6\sigma_{12}}{50} + \frac{2c_3s_2s_4}{5} + \frac{9}{25} \end{bmatrix} \tag{.1}$$

$$\boldsymbol{R}_{\boldsymbol{0}}^{\boldsymbol{7}} = \begin{bmatrix} s_7\sigma_5 - c_7\sigma_2 & c_7\sigma_5 + s_7\sigma_2 & c_6\sigma_{10} - s_6\sigma_9 \\ c_7\sigma_1 - s_7\sigma_4 & -c_7\sigma_4 - s_7\sigma_1 & s_6\sigma_7 - c_6\sigma_8 \\ c_7\sigma_3 - s_7\sigma_6 & -c_7\sigma_6 - s_7\sigma_3 & s_6\sigma_{11} + c_6\sigma_{12} \end{bmatrix} \tag{.2}$$

$$\begin{aligned}
\sigma_1 &= \sin(\theta_6)\,\sigma_8 + \cos(\theta_6)\,\sigma_7 \\
\sigma_2 &= \sin(\theta_6)\,\sigma_{10} + \cos(\theta_6)\,\sigma_9 \\
\sigma_3 &= \cos(\theta_6)\,\sigma_{11} - \sin(\theta_6)\,\sigma_{12} \\
\sigma_4 &= \sin(\theta_5)\,\sigma_{13} - \cos(\theta_5)\,\sigma_{14} \\
\sigma_5 &= \sin(\theta_5)\,\sigma_{15} - \cos(\theta_5)\,\sigma_{16} \\
\sigma_6 &= \sin(\theta_5)\,\sigma_{17} - \cos(\theta_5)\sin(\theta_2)\sin(\theta_3) \\
\sigma_7 &= \cos(\theta_5)\,\sigma_{13} + \sin(\theta_5)\,\sigma_{14} \\
\sigma_8 &= \sin(\theta_4)\,\sigma_{18} - \cos(\theta_4)\sin(\theta_1)\sin(\theta_2) \\
\sigma_9 &= \cos(\theta_5)\,\sigma_{15} + \sin(\theta_5)\,\sigma_{16} \\
\sigma_{10} &= \sin(\theta_4)\,\sigma_{19} + \cos(\theta_1)\cos(\theta_4)\sin(\theta_2) \\
\sigma_{11} &= \cos(\theta_5)\,\sigma_{17} + \sin(\theta_2)\sin(\theta_3)\sin(\theta_5) \\
\sigma_{12} &= \cos(\theta_2)\cos(\theta_4) + \cos(\theta_3)\sin(\theta_2)\sin(\theta_4) \\
\sigma_{13} &= \cos(\theta_4)\,\sigma_{18} + \sin(\theta_1)\sin(\theta_2)\sin(\theta_4) \\
\sigma_{14} &= \cos(\theta_1)\cos(\theta_3) - \cos(\theta_2)\sin(\theta_1)\sin(\theta_3) \\
\sigma_{15} &= \cos(\theta_4)\,\sigma_{19} - \cos(\theta_1)\sin(\theta_2)\sin(\theta_4) \\
\sigma_{16} &= \cos(\theta_3)\sin(\theta_1) + \cos(\theta_1)\cos(\theta_2)\sin(\theta_3) \\
\sigma_{17} &= \cos(\theta_2)\sin(\theta_4) - \cos(\theta_3)\cos(\theta_4)\sin(\theta_2) \\
\sigma_{18} &= \cos(\theta_1)\sin(\theta_3) + \cos(\theta_2)\cos(\theta_3)\sin(\theta_1) \\
\sigma_{19} &= \sin(\theta_1)\sin(\theta_3) - \cos(\theta_1)\cos(\theta_2)\cos(\theta_3)
\end{aligned}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

# A2 Geometric Jacobian

The symbolic geometric Jacobian matrix expressions of (2.28) were calculated with Maple 2021. Commands and results follow in this section.

### Maple Commands

$\mathbf{z0} = [0; 0; 1]$
$\mathbf{z1} = \mathbf{T01}(1:3,3)$
$\mathbf{z2} = \mathbf{T02}(1:3,3)$
$\mathbf{z3} = \mathbf{T03}(1:3,3)$
$\mathbf{z4} = \mathbf{T04}(1:3,3)$
$\mathbf{z5} = \mathbf{T05}(1:3,3)$
$\mathbf{z6} = \mathbf{T06}(1:3,3)$

$\mathbf{o0} = [0; 0; 0]$
$\mathbf{o1} = \mathbf{T01}(1:3,4)$
$\mathbf{o2} = \mathbf{T02}(1:3,4)$
$\mathbf{o3} = \mathbf{T03}(1:3,4)$
$\mathbf{o4} = \mathbf{T04}(1:3,4)$
$\mathbf{o5} = \mathbf{T05}(1:3,4)$
$\mathbf{o6} = \mathbf{T06}(1:3,4)$
$\mathbf{o7} = \mathbf{T07}(1:3,4)$

$\mathbf{Jv1} = \text{cross}(\mathbf{z0},(\mathbf{o7} - \mathbf{z0}))$
$\mathbf{Jv2} = \text{cross}(\mathbf{z1},(\mathbf{o7} - \mathbf{z1}))$
$\mathbf{Jv3} = \text{cross}(\mathbf{z2},(\mathbf{o7} - \mathbf{z2}))$
$\mathbf{Jv4} = \text{cross}(\mathbf{z3},(\mathbf{o7} - \mathbf{z3}))$
$\mathbf{Jv5} = \text{cross}(\mathbf{z4},(\mathbf{o7} - \mathbf{z4}))$
$\mathbf{Jv6} = \text{cross}(\mathbf{z5},(\mathbf{o7} - \mathbf{z5}))$
$\mathbf{Jv7} = \text{cross}(\mathbf{z6},(\mathbf{o7} - \mathbf{z6}))$

$\mathbf{Jv} = [\mathbf{Jv1}, \mathbf{Jv2}, \mathbf{Jv3}, \mathbf{Jv4}, \mathbf{Jv5}, \mathbf{Jv6}, \mathbf{Jv7}]$
$\mathbf{Jw} = [\mathbf{z0}, \mathbf{z1}, \mathbf{z2}, \mathbf{z3}, \mathbf{z4}, \mathbf{z5}, \mathbf{z6}]$
$\mathbf{Jacobian} = [\mathbf{Jv}; \mathbf{Jw}]$
**Results**

$$\mathbf{Jacobian} = \boldsymbol{J}_7^0(\boldsymbol{q}) = \begin{bmatrix} \delta_1 & c_1\sigma_3 & \delta_{11} & \delta_2 & \delta_5 & \delta_9 & \delta_{15} \\ \sigma_1 & s_1\sigma_3 & \delta_{12} & \delta_3 & \delta_6 & \delta_{13} & \delta_{16} \\ 0 & \delta_8 & \delta_{10} & \delta_4 & \delta_7 & \delta_{14} & \delta_{17} \\ 0 & s_1 & c_1 s_2 & \sigma_{33} - \sigma_{34} & \sigma_{35} - s_4\sigma_{39} & \sigma_{24} - s_5\sigma_{32} & \sigma_{41} \\ 0 & -c_1 & s_1 s_2 & \sigma_{36} & \sigma_{28} & \sigma_5 & \sigma_4 \\ 1 & 0 & -c_2 & s_2 s_3 & -\sigma_{31} - \sigma_{30} & \sigma_7 & \sigma_{42} \end{bmatrix} \tag{.3}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

$$\sin(\theta_i) = s_i$$
$$\cos(\theta_i) = c_i$$
$$\delta_1 = \sigma_{14} - \sigma_{17} - \sigma_{15} - \sigma_{18} - \sigma_{16}$$
$$\delta_2 = -\sigma_{36}\left(\sigma_{22} + \sigma_{19} + \sigma_{20} + \sigma_{21}\right) - \sin\left(\theta_2\right)\,\sin\left(\theta_3\right)\,\left(\sigma_{17} + \sigma_{15} - \sigma_{14} + \sigma_{16}\right)$$
$$\delta_3 = -(\sigma_{34} - \sigma_{33})\left(\sigma_{22} + \sigma_{19} + \sigma_{20} + \sigma_{21}\right) - \sin\left(\theta_2\right)\,\sin\left(\theta_3\right)\,\left(\sigma_{11} - \sigma_{10} + \sigma_{13} - \sigma_{12}\right)$$
$$\delta_4 = \sigma_{36}\left(\sigma_{11} - \sigma_{10} + \sigma_{13} - \sigma_{12}\right) - (\sigma_{34} - \sigma_{33})\left(\sigma_{17} + \sigma_{15} - \sigma_{14} + \sigma_{16}\right)$$
$$\delta_5 = (\sigma_{31} + \sigma_{30})\left(\sigma_{17} + \sigma_{15} - \sigma_{14} + \sigma_{16}\right) - \sigma_{28}\left(\sigma_{22} + \sigma_{19} + \sigma_{20} + \sigma_{21}\right)$$
$$\delta_6 = (\sigma_{31} + \sigma_{30})\left(\sigma_{11} - \sigma_{10} + \sigma_{13} - \sigma_{12}\right) - \sigma_{26}\left(\sigma_{22} + \sigma_{19} + \sigma_{20} + \sigma_{21}\right)$$
$$\delta_7 = \sigma_{28}\left(\sigma_{11} - \sigma_{10} + \sigma_{13} - \sigma_{12}\right) - \sigma_{26}\left(\sigma_{17} + \sigma_{15} - \sigma_{14} + \sigma_{16}\right)$$
$$\delta_8 = \cos\left(\theta_1\right)\,\sigma_1 + \sin\left(\theta_1\right)\,\sigma_2$$
$$\delta_9 = -\sigma_5\left(\sigma_{19} + \sigma_{20}\right) - \sigma_7\left(\sigma_{17} - \sigma_{14}\right)$$
$$\delta_{10} = \cos\left(\theta_1\right)\,\sin\left(\theta_2\right)\,\sigma_2 - \sin\left(\theta_1\right)\,\sin\left(\theta_2\right)\,\sigma_1$$
$$\delta_{11} = \cos\left(\theta_2\right)\,\sigma_2 - \sin\left(\theta_1\right)\,\sin\left(\theta_2\right)\,\sigma_3$$
$$\delta_{12} = \cos\left(\theta_1\right)\,\sin\left(\theta_2\right)\,\sigma_3 - \cos\left(\theta_2\right)\,\sigma_1$$
$$\delta_{13} = -\sigma_9\left(\sigma_{19} + \sigma_{20}\right) - \sigma_7\left(\sigma_{11} - \sigma_{10}\right)$$
$$\delta_{14} = \left(\sigma_{11} - \sigma_{10}\right)\sigma_5 - \sigma_9\left(\sigma_{17} - \sigma_{14}\right)$$
$$\delta_{15} = \sigma_8\left(\sigma_{17} - \sigma_{14}\right) - \left(\sigma_{19} + \sigma_{20}\right)\sigma_4$$
$$\delta_{16} = \left(\sigma_{11} - \sigma_{10}\right)\sigma_8 - \sigma_6\left(\sigma_{19} + \sigma_{20}\right)$$
$$\delta_{17} = \left(\sigma_{11} - \sigma_{10}\right)\sigma_4 - \sigma_6\left(\sigma_{17} - \sigma_{14}\right)$$
$$\sigma_1 = \frac{2\,\cos(\theta_1)\,\sin(\theta_2)}{5} - \sigma_{11} + \sigma_{10} - \sigma_{13} + \sigma_{12}$$
$$\sigma_2 = \sigma_{18} + \sigma_{17} + \sigma_{15} - \sigma_{14} + \sigma_{16}$$
$$\sigma_3 = \frac{2\,\cos(\theta_2)}{5} + \sigma_{22} + \sigma_{19} + \sigma_{20} + \sigma_{21}$$
$$\sigma_4 = \cos\left(\theta_6\right)\,\sigma_{28} - \sin\left(\theta_6\right)\,\sigma_{27}$$
$$\sigma_5 = \sin\left(\theta_5\right)\,\sigma_{37} - \cos\left(\theta_5\right)\,\sigma_{36}$$
$$\sigma_6 = \cos\left(\theta_6\right)\,\sigma_{26} - \sin\left(\theta_6\right)\,\sigma_{25}$$
$$\sigma_7 = \sin\left(\theta_5\right)\,\sigma_{38} - \cos\left(\theta_5\right)\,\sin\left(\theta_2\right)\,\sin\left(\theta_3\right)$$
$$\sigma_8 = \sin\left(\theta_6\right)\,\sigma_{29} + \sigma_{23}$$
$$\sigma_9 = \sin\left(\theta_5\right)\,\sigma_{32} - \sigma_{24}$$
$$\sigma_{10} = \frac{2\,\sin(\theta_6)\,\sigma_{25}}{25}$$
$$\sigma_{11} = \frac{2\,\cos(\theta_6)\,\sigma_{26}}{25}$$
$$\sigma_{12} = \frac{2\,\cos(\theta_1)\,\cos(\theta_4)\,\sin(\theta_2)}{5}$$
$$\sigma_{13} = \frac{2\,\sin(\theta_4)\,\sigma_{39}}{5}$$
$$\sigma_{14} = \frac{2\,\sin(\theta_6)\,\sigma_{27}}{25}$$
$$\sigma_{15} = \frac{2\,\sin(\theta_4)\,\sigma_{40}}{5}$$
$$\sigma_{16} = \frac{2\,\cos(\theta_4)\,\sin(\theta_1)\,\sin(\theta_2)}{5}$$
$$\sigma_{17} = \frac{2\,\cos(\theta_6)\,\sigma_{28}}{25}$$
$$\sigma_{18} = \frac{2\,\sin(\theta_1)\,\sin(\theta_2)}{5}$$
$$\sigma_{19} = \frac{2\,\sin(\theta_6)\,\sigma_{29}}{25}$$
$$\sigma_{20} = \frac{2\,\cos(\theta_6)\,(\sigma_{31}+\sigma_{30})}{25}$$
$$\sigma_{21} = \frac{2\,\cos(\theta_3)\,\sin(\theta_2)\,\sin(\theta_4)}{5}$$
$$\sigma_{22} = \frac{2\,\cos(\theta_2)\,\cos(\theta_4)}{5}$$
$$\sigma_{23} = \cos\left(\theta_6\right)\,\left(\sigma_{31} + \sigma_{30}\right)$$
$$\sigma_{24} = \cos\left(\theta_5\right)\,\left(\sigma_{34} - \sigma_{33}\right)$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

$$\sigma_{25} = \cos(\theta_5)\ \sigma_{32} + \sin(\theta_5)\ (\sigma_{34} - \sigma_{33})$$
$$\sigma_{26} = \sin(\theta_4)\ \sigma_{39} - \sigma_{35}$$
$$\sigma_{27} = \cos(\theta_5)\ \sigma_{37} + \sin(\theta_5)\ \sigma_{36}$$
$$\sigma_{28} = \sin(\theta_4)\ \sigma_{40} + \cos(\theta_4)\ \sin(\theta_1)\ \sin(\theta_2)$$
$$\sigma_{29} = \cos(\theta_5)\ \sigma_{38} + \sin(\theta_2)\ \sin(\theta_3)\ \sin(\theta_5)$$
$$\sigma_{30} = \cos(\theta_3)\ \sin(\theta_2)\ \sin(\theta_4)$$
$$\sigma_{31} = \cos(\theta_2)\ \cos(\theta_4)$$
$$\sigma_{32} = \cos(\theta_4)\ \sigma_{39} + \cos(\theta_1)\ \sin(\theta_2)\ \sin(\theta_4)$$
$$\sigma_{33} = \cos(\theta_1)\ \cos(\theta_2)\ \sin(\theta_3)$$
$$\sigma_{34} = \cos(\theta_3)\ \sin(\theta_1)$$
$$\sigma_{35} = \cos(\theta_1)\ \cos(\theta_4)\ \sin(\theta_2)$$
$$\sigma_{36} = \cos(\theta_1)\ \cos(\theta_3) + \cos(\theta_2)\ \sin(\theta_1)\ \sin(\theta_3)$$
$$\sigma_{37} = \cos(\theta_4)\ \sigma_{40} - \sin(\theta_1)\ \sin(\theta_2)\ \sin(\theta_4)$$
$$\sigma_{38} = \cos(\theta_2)\ \sin(\theta_4) - \cos(\theta_3)\ \cos(\theta_4)\ \sin(\theta_2)$$
$$\sigma_{39} = \sin(\theta_1)\ \sin(\theta_3) + \cos(\theta_1)\ \cos(\theta_2)\ \cos(\theta_3)$$
$$\sigma_{40} = \cos(\theta_1)\ \sin(\theta_3) - \cos(\theta_2)\ \cos(\theta_3)\ \sin(\theta_1)$$
$$\sigma_{41} = \sin(\theta_6)\ \sigma_{25} - \cos(\theta_6)\ \sigma_{26}$$
$$\sigma_{42} = -\sin(\theta_6)\ \sigma_{29} - \sigma_{23}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

# List of Figures





Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

# List of Tables




Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

# Bibliography


[1] O. Madsen, S. Bøgh, C. Schou, R. Andersen, J. Damgaard, M. Pedersen, and V. Krueger, "Integration of mobile manipulators in an industrial production," *Industrial Robot*, vol. 42, no. 1, pp. 11–18, 2015 (cit. on p. 1).

[2] M. Yang, E. Yang, R. C. Zante, M. Post, and X. Liu, "Collaborative mobile industrial manipulator: A review of system architecture and applications," in *Proceedings of the 25th International Conference on Automation and Computing (ICAC)*, 2019, pp. 1–6 (cit. on pp. 1, 2).

[3] Y. Xiong, P. From, and V. Isler, "Design and evaluation of a novel cable-driven gripper with perception capabilities for strawberry picking robots," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 7384–7391 (cit. on p. 1).

[4] B. Hamner, S. Koterba, J. Shi, R. Simmons, and S. Singh, "An autonomous mobile manipulator for assembly tasks," *Autonomous Robots*, vol. 28, no. 1, pp. 131–149, 2010 (cit. on pp. 1, 3).

[5] Y. Zhang, J. Wang, and Y. Xia, "A dual neural network for redundancy resolution of kinematically redundant manipulators subject to joint limits and joint velocity limits," *IEEE Transactions on Neural Networks*, vol. 14, no. 3, pp. 658–667, 2003 (cit. on pp. 1, 2).

[6] T. Sandakalum and M. H. Ang, "Motion planning for mobile manipulators: A systematic review," *Machines*, vol. 10, no. 2, 97, 2022. [Online]. Available: `https://www.mdpi.com/2075-1702/10/2/97` (visited on 04/24/2023) (cit. on pp. 1, 3).

[7] F. Valero, V. Mata, F. Rubio, and J.-L. Suñer, "Influence of a manipulability index on trajectory planning for robots in a workspace with obstacles," in *Advances in Robot Kinematics: Theory and Applications*. Springer Dordrecht, 2002, pp. 67–76 (cit. on p. 1).

[8] M. Dadvar and S. Habibian, "Contemporary research trends in response robotics," *ROBOMECH Journal*, vol. 9, no. 1, 9, 2022. [Online]. Available: `https://doi.org/10.1186/s40648-022-00221-z` (visited on 04/24/2023) (cit. on p. 2).



Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

[9] F. Rubio, F. Valero, and C. Llopis-Albert, "A review of mobile robots: Concepts, methods, theoretical framework, and applications," *International Journal of Advanced Robotic Systems*, vol. 16, no. 2, 172988141983959, 2019. [Online]. Available: `https://journals.sagepub.com/doi/10.1177/1729881419839596` (visited on 05/14/2023) (cit. on p. 2).

[10] M. Hvilshøj, S. Bøgh, O. Nielsen, and O. Madsen, "Autonomous industrial mobile manipulation (AIMM): Past, present and future," *Industrial Robot*, vol. 39, no. 2, pp. 120–135, 2012 (cit. on p. 2).

[11] T. Wisspeintner, T. van der Zant, L. Iocchi, and S. Schiffer, "Robocup@home: Scientific competition and benchmarking for domestic service robots," *Interaction Studies*, vol. 10, no. 3, pp. 392–426, 2009 (cit. on p. 2).

[12] A. Andreopoulos, S. Hasler, H. Wersing, H. Janssen, J. Tsotsos, and E. Körner, "Active 3D Object Localization Using a Humanoid Robot," *IEEE Transactions on Robotics*, vol. 27, no. 1, pp. 47–64, 2011 (cit. on p. 2).

[13] S. Bøgh, M. Hvilshøj, M. Kristiansen, and O. Madsen, "Autonomous Industrial Mobile Manipulation (AIMM): From Research to Industry," in *Proceedings of the 42nd International Symposium on Robotics*, 2011 (cit. on p. 2).

[14] B. Hamner, S. Koterba, J. Shi, R. Simmons, and S. Singh, "An autonomous mobile manipulator for assembly tasks," *Autonomous Robots*, vol. 28, no. 1, pp. 131–149, 2010 (cit. on p. 2).

[15] C. Cheng, J. Fu, H. Su, and L. Ren, "Recent advancements in agriculture robots: Benefits and challenges," *Machines*, vol. 11, no. 1, 48, 2023. [Online]. Available: `https://www.mdpi.com/2075-1702/11/1/48` (visited on 05/14/2023) (cit. on p. 2).

[16] Z. Li, P. Moran, Q. Dong, R. J. Shaw, and K. Hauser, "Development of a tele-nursing mobile manipulator for remote care-giving in quarantine areas," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 3581–3586 (cit. on p. 2).

[17] G. Marchant, B. Allenby, R. Arkin, E. Barrett, J. Borenstein, L. Gaudet, O. Kittrie, P. Lin, G. Lucas, R. O'Meara, and J. Silberman, "International governance of autonomous military robots," in *Handbook of Unmanned Aerial Vehicles*. Springer Dordrecht, 2015, pp. 2879–2910 (cit. on p. 2).

[18] Y. Nakamura, *Advanced robotics: redundancy and optimization*. Addison-Wesley, 1991 (cit. on p. 2).

[19] F. G. Pin and J.-C. Culioli, "Optimal positioning of combined mobile platform-manipulator systems for material handling tasks," *Journal of Intelligent and Robotic Systems*, vol. 6, no. 2, pp. 165–182, 1992 (cit. on p. 2).

[20] Y. Zhang and J. Wang, "Obstacle avoidance for kinematically redundant manipulators using a dual neural network," *IEEE transactions on systems, man, and cybernetics*, vol. 34, no. 1, pp. 752–759, 2004 (cit. on p. 2).

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[21] Z. Li and S. Li, "Model-based recurrent neural network for redundancy resolution of manipulator with remote centre of motion constraints," *International Journal of Systems Science*, vol. 53, no. 1, pp. 1–14, 2022 (cit. on p. 2).

[22] S. Chiaverini, G. Oriolo, and I. D. Walker, "Kinematically redundant manipulators," in *Springer Handbook of Robotics*. Springer Berlin Heidelberg, 2008, pp. 245–268 (cit. on pp. 2, 12).

[23] R. Guaman, R. Garcia Alvarado, A. Martínez Rocamora, and F. Auat Cheein, "Workspace analysis of a mobile manipulator with obstacle avoidance in 3D printing tasks," *Applied Sciences*, vol. 11, no. 17, pp. 1–16, 2021 (cit. on p. 3).

[24] K. M. Lynch and F. C. Park, *Modern Robotics: Mechanics, Planning, and Control*. Cambridge University Press, 2017 (cit. on pp. 3, 12, 20, 23, 26, 27, 51).

[25] C. Rocha, C. Tonetto, and A. Dias, "A comparison between the Denavit–Hartenberg and the screw-based methods used in kinematic modeling of robot manipulators," *Robotics and Computer-Integrated Manufacturing*, vol. 27, no. 4, pp. 723–728, 2011 (cit. on p. 3).

[26] M. Chen and A. Zalzala, "Dynamic modelling and genetic-based trajectory generation for non-holonomic mobile manipulators," *Control Engineering Practice*, vol. 5, no. 1, pp. 39–48, 1997 (cit. on p. 3).

[27] M. Korayem, M. Nazemizadeh, and V. Azimirad, "Optimal trajectory planning of wheeled mobile manipulators in cluttered environments using potential functions," *Scientia Iranica*, vol. 18, no. 5, pp. 1138–1147, 2011 (cit. on p. 3).

[28] S. Patel and T. Sobh, "Manipulator performance measures - a comprehensive literature survey," *Journal of Intelligent and Robotic Systems*, vol. 77, no. 3, pp. 547–570, 2014 (cit. on pp. 3, 32, 41, 42).

[29] B. Ouyang and W. Shang, "Wrench-feasible workspace based optimization of the fixed and moving platforms for cable-driven parallel manipulators," *Robotics and Computer-Integrated Manufacturing*, vol. 30, no. 6, pp. 629–635, 2014 (cit. on p. 3).

[30] D. C. H. Yang and Z. C. Lai, "On the dexterity of robotic manipulators—service angle," *Journal of Mechanisms Transmissions and Automation in Design*, vol. 107, no. 2, pp. 262–270, 1985 (cit. on pp. 3, 32).

[31] A. Kumar and K. J. Waldron, "The workspaces of a mechanical manipulator," *Journal of Mechanical Design*, vol. 103, no. 3, pp. 665–672, 1981 (cit. on p. 3).

[32] M. Spong, S. Hutchinson, and M. Vidyasagar, *Robot Modeling and Control*. John Wiley & Sons, Inc., 2020 (cit. on pp. 3, 9–12, 14, 17, 19, 21, 23, 24, 51).

[33] R. P. Paul and C. N. Stevenson, "Kinematics of robot wrists," *The International Journal of Robotics Research*, vol. 2, no. 1, pp. 31–38, 1983 (cit. on p. 4).

[34] J. W. Demmel, "The geometry of III-conditioning," *Journal of Complexity*, vol. 3, no. 2, pp. 201–229, 1987 (cit. on p. 4).

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[35] C. A. Klein and B. E. Blaho, "Dexterity measures for the design and control of kinematically redundant manipulators," *The International Journal of Robotics Research*, vol. 6, no. 2, pp. 72–83, 1987 (cit. on pp. 4, 11, 12).

[36] T. Yoshikawa, "Manipulability of robotic mechanisms," *The International Journal of Robotics Research*, vol. 4, no. 2, pp. 3–9, 1985 (cit. on pp. 4, 32, 40–43, 56).

[37] K. Dufour and W. Suleiman, "On maximizing manipulability index while solving a kinematics task," *Journal of Intelligent and Robotic Systems*, vol. 100, no. 5, pp. 3–13, 2020 (cit. on p. 4).

[38] F. Marić, O. Limoyo, L. Petrovic, T. Ablett, I. Petrovic, and J. Kelly, "Fast manipulability maximization using continuous-time trajectory optimization," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 8258–8264 (cit. on p. 4).

[39] N. Vahrenkamp and T. Asfour, "Representing the robot's workspace through constrained manipulability analysis," *Autonomous Robots*, vol. 38, no. 1, pp. 1–14, 2014 (cit. on p. 4).

[40] Z. Cui, P. Cao, Y.-M. Shao, D.-H. Qian, and X.-C. Wang, "Trajectory planning for a redundant mobile manipulator using avoidance manipulability," in *Proceedings of the IEEE International Conference on Automation and Logistics*, 2009, pp. 283–288 (cit. on p. 4).

[41] I. Akli, M. Haddad, B. Bouzouia, and N. Achour, "Trajectory generation for operational task execution with manipulability analysis," in *Proceedings of the International Conference on Advanced Robotics (ICAR)*, 2011, pp. 94–99 (cit. on p. 4).

[42] V. Andaluz, F. Roberti, J. Toibero, and R. Carelli, "Adaptive unified motion control of mobile manipulators," *Control Engineering Practice*, vol. 20, no. 12, pp. 1337–1352, 2012 (cit. on p. 4).

[43] N. Vahrenkamp, T. Asfour, and R. Dillmann, "Efficient inverse kinematics computation based on reachability analysis," *International Journal of Humanoid Robotics*, vol. 9, no. 4, p. 1 250 035, 2012. [Online]. Available: `https://www.worldscientific.com/doi/abs/10.1142/S0219843612500351` (visited on 05/15/2023) (cit. on p. 4).

[44] Y. Yamamoto and X. Yun, "Unified analysis on mobility and manipulability of mobile manipulators," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, vol. 2, 1999, pp. 1200–1206 (cit. on p. 4).

[45] B. Bayle, J.-Y. Fourquet, and M. Renaud, "Manipulability analysis for mobile manipulators," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, vol. 2, 2001, pp. 1251–1256 (cit. on p. 4).

[46] W. Merkt, Y. Yang, T. Stouraitis, C. E. Mower, M. Fallon, and S. Vijayakumar, "Robust shared autonomy for mobile manipulation with continuous scene monitoring," in *Proceedings of the IEEE Conference on Automation Science and Engineering (CASE)*, 2017, pp. 130–137 (cit. on p. 4).

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[47] B. Teka, H. Jangid, R. Raja, and A. Dutta, "Advanced KSOM based Redundancy Resolution of a Mobile Manipulator System for Motion on an Uneven Terrain," in *Proceedings of the Advances in Robotics*, Association for Computing Machinery, 2017, pp. 1–6 (cit. on p. 4).

[48] R. Bischoff, J. Kurth, G. Schreiber, R. Koeppe, A. Albu-Schaeffer, A. Beyer, O. Eiberger, S. Haddadin, A. Stemmer, G. Grunwald, and G. Hirzinger, "The KUKA-DLR Lightweight Robot arm - a new reference platform for robotics research and manufacturing," in *Proceedings of the (ISR) International Symposium on Robotics and German Conference on Robotics (ROBOTIK)*, 2010, pp. 1–8 (cit. on p. 7).

[49] *SALLY Flexibles Mini-FTF*, DS Automotion GmbH, 2022. [Online]. Available: `https://www.ds-automotion.com/fileadmin/user_upload/Fahrzeug_Renderings/Sally/PDF/SALLY_Onepager_2022_DE.pdf` (visited on 04/22/2023) (cit. on pp. 7, 9, 26).

[50] *LBR iiwa LBR iiwa 7 R800, LBR iiwa 14 R820 Specification*, KUKA Roboter GmbH, 2015. [Online]. Available: `https://www.oir.caltech.edu/twiki_oir/pub/Palomar/ZTF/KUKARoboticArmMaterial/Spez_LBR_iiwa_en.pdf` (visited on 04/29/2023) (cit. on pp. 8, 18, 19).

[51] B. Siciliano, L. Sciavicco, V. Luigi, and G. Oriolo, *Robotics: Modelling, Planning and Control*. Springer London, 2011 (cit. on pp. 13, 14, 17).

[52] E. Hemingway and O. O'Reilly, "Perspectives on euler angle singularities, gimbal lock, and the orthogonality of applied forces and applied moments," *Multibody System Dynamics*, vol. 44, no. 9, pp. 31–56, 2018 (cit. on p. 14).

[53] P. I. Corke, *Robotics, Vision & Control: Fundamental Algorithms in MATLAB*, Second. Springer Cham, 2017, ISBN 978-3-319-54413-7 (cit. on pp. 14, 15, 17, 21, 26, 27, 40).

[54] B. Hall, *Lie Groups, Lie Algebras, and Representations*. Springer Cham, 2013 (cit. on pp. 14, 15).

[55] B. Siciliano and O. Khatib, *Springer Handbook of Robotics*. Springer Berlin Heidelberg, 2008 (cit. on pp. 16, 17, 21, 24, 41).

[56] F. Steinparz, "Co-ordinate transformation and robot control with Denavit-Hartenberg matrices," *Journal of Microcomputer Applications*, vol. 8, no. 4, pp. 303–316, 1985 (cit. on p. 17).

[57] P. Corke, "A Simple and Systematic Approach to Assigning Denavit–Hartenberg Parameters," *IEEE Transactions on Robotics*, vol. 23, no. 3, pp. 590–594, 2007 (cit. on p. 17).

[58] D. L. Pieper, "The kinematics of manipulators under computer control," PhD Thesis, Stanford University, Computer Science Department, 1968 (cit. on p. 21).

[59] E. H. Moore, "On the reciprocal of the general algebraic matrix," *Bulletin of the American Mathematical Society*, vol. 26, no. 9, 394–395, 1920 (cit. on p. 24).

[60] R. Penrose, "A generalized inverse for matrices," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 51, no. 3, 406—413, 1955 (cit. on p. 24).

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[61] S. B. Nokleby and R. P. Podhorodeski, "Reciprocity-based resolution of velocity degeneracies (singularities) for redundant manipulators," *Mechanism and Machine Theory*, vol. 36, no. 3, pp. 397–409, 2001 (cit. on p. 24).

[62] Z.-H. Kang, C.-A. Cheng, and h. Huang, "A singularity handling algorithm based on operational space control for six-degree-of-freedom anthropomorphic manipulators," *International Journal of Advanced Robotic Systems*, vol. 16, no. 3, p. 172 988 141 985 891, 2019. [Online]. Available: `https://journals.sagepub.com/doi/10.1177/1729881419858910` (visited on 05/15/2023) (cit. on p. 24).

[63] S. B. Nokleby and R. P. Podhorodeski, "Velocity degeneracy determination for the kinematically redundant CSA/ISE STEAR testbed manipulator," *Journal of Robotic Systems*, vol. 17, no. 11, pp. 633–642, 2000 (cit. on p. 24).

[64] K. J Waldron, S.-L. Wang, and S. J Bolin, "A study of the jacobian matrix of serial manipulators," *Journal of mechanical design (1990)*, vol. 107, no. 2, pp. 230–237, 1985 (cit. on p. 24).

[65] F. Beck, M. N. Vu, C. Hartl-Nesic, and A. Kugi, *Singlularity avoidance with application to online trajectory optimization for serial manipulators*, 2022 (cit. on p. 25).

[66] L. Zhang, S. Guo, Y. Huang, and X. Xiong, "Kinematic singularity analysis and simulation for 7dof anthropomorphic manipulator," *International Journal of Mechatronics and Applied Mechanics*, vol. 1, no. 6, pp. 157–164, 2019 (cit. on p. 25).

[67] R. Boudreau and R. Podhorodeski, "Singularity analysis of a kinematically simple class of 7-jointed revolute manipulators," *Transactions of the Canadian Society for Mechanical Engineering*, vol. 34, no. 1, pp. 105–117, 2010 (cit. on p. 25).

[68] *KUKA Sunrise.OS KUKA Sunrise.Workbench, Operating and Programming Instructions for System Integrators*, KUKA Roboter GmbH, 2016 (cit. on p. 25).

[69] C. Gosselin, "Dexterity indices for planar and spatial robotic manipulators," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, vol. 1, 1990, pp. 650–655 (cit. on p. 32).

[70] K. Abdel-Malek, F. Adkins, H.-J. Yeh, and E. Haug, "On the determination of boundaries to manipulator workspaces," *Robotics and Computer-Integrated Manufacturing*, vol. 13, no. 1, pp. 63–72, 1997 (cit. on p. 33).

[71] J. Rastegar and D. Perel, "Generation of manipulator workspace boundary geometry using the monte carlo method and interactive computer graphics," *Journal of Mechanical Design*, vol. 112, no. 3, pp. 452–454, 1990 (cit. on p. 33).

[72] P. Anderson-Sprecher and R. Simmons, "Voxel-based motion bounding and workspace estimation for robotic manipulators," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 2141–2146 (cit. on p. 34).

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[73] X. Xu and K. Harada, "Automatic surface reconstruction with alpha-shape method," *The Visual Computer*, vol. 19, no. 7, pp. 431–443, 2003 (cit. on pp. 34, 36).

[74] P. Mileff and J. Dudra, "Simplified voxel based visualization," *Production Systems and Information Engineering*, vol. 8, no. 1, pp. 5–18, 2019 (cit. on p. 36).

[75] W. Zhou and H. Yan, "Alpha shape and delaunay triangulation in studies of protein-related interactions," *Briefings in bioinformatics*, vol. 15, no. 1, pp. 54–64, 2014 (cit. on p. 36).

[76] J. Wen and L. Wilfinger, "Kinematic manipulability of general constrained rigid multibody systems," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 3, pp. 558–567, 1999 (cit. on p. 40).

[77] J. K. Salisbury and J. J. Craig, "Articulated hands: Force control and kinematic issues," *The International Journal of Robotics Research*, vol. 1, no. 1, pp. 4–17, 1982 (cit. on p. 42).

[78] T. Yoshikawa, "Dynamic manipulability of robot manipulators," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 1985, pp. 1033–1038 (cit. on p. 42).

[79] S. Tadokoro, I. Kimura, and T. Takamori, "Dexterity measure for trajectory planning and kinematic design of redundant manipulators," Proceedings of the Annual Conference of IEEE Industrial Electronics Society - (IECON), 1989, pp. 415–420 (cit. on p. 43).

[80] W. G. Manning, A. Basu, and J. Mullahy, "Generalized modeling approaches to risk adjustment of skewed outcomes data," *Journal of Health Economics*, vol. 24, no. 3, pp. 465–488, 2005 (cit. on p. 48).

[81] S. Chiddarwar and N. Babu, "Optimal trajectory planning for industrial robot along a specified path with payload constraint using trigonometric splines," *International Journal of Automation and Control*, vol. 6, no. 1, pp. 39–65, 2012 (cit. on pp. 51, 55, 56).

[82] M. J. E. Richardson and T. Flash, "Comparing smooth arm movements with the two-thirds power law and the related segmented-control hypothesis," *Journal of Neuroscience*, vol. 22, no. 18, pp. 8201–8211, 2002 (cit. on p. 56).

[83] P. Foehn, A. Romero, and D. Scaramuzza, "Time-optimal planning for quadrotor waypoint flight," *Science Robotics*, vol. 6, no. 56, eabh1221, 2021. [Online]. Available: `https://www.science.org/doi/10.1126/scirobotics.abh1221` (visited on 04/24/2023) (cit. on p. 58).

[84] E. P. Z. Bartosiewicz, "Euler's discretization and dynamic equivalence of nonlinear control systems," in *Nonlinear control in the year 2000*. Springer London, 2001, vol. 2, pp. 183–191 (cit. on p. 59).

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN