Article

Task-Dependent Comfort Zone, a Base Placement Strategy for Mobile Manipulators Based on Manipulability Measures

Martin Sereinig [ID], Peter Manzl *[ID] and Johannes Gerstmayr *[ID]

Department of Mechatronics, University of Innsbruck, 6020 Innsbruck, Austria;
martin.sereinig@student.uibk.ac.at
* Correspondence:peter.manzl@uibk.ac.at (P.M.); johannes.gerstmayr@uibk.ac.at (J.G.)

Abstract: The present contribution introduces the task-dependent comfort zone as a base placement strategy for mobile manipulators using different manipulability measures. Four different manipulability measures depending on end-effector velocities, forces, stiffness, and accelerations are considered. By evaluating a discrete subspace of the manipulator workspace with these manipulability measures and using image-processing algorithms, a suitable goal position for the autonomous mobile manipulator was defined within the comfort zone. This always ensures a certain manipulator manipulablity value with a lower limit with respect to the maximum possible manipulability in the discrete subspace. Results are shown for three different mobile manipulators using the velocity-dependent manipulability measure in a simulation.

Keywords: mobile manipulation; autonomous mobile manipulator; manipulability measure; task-dependent comfort zone

1. Introduction

Mobile manipulators, which combine a mobile base platform and a serial manipulator, are becoming increasingly popular in various applications due to their flexibility and adaptability when operating in diverse environments [1,2].

To ensure their efficient and reliable operation, it is crucial to choose movement algorithms that take into account the unique characteristics of both the mobile platform and the serial manipulator. While researchers work on combining base and manipulator movement [3], the physical limitations of mobile manipulators also have to be considered. The mobile platform movement of wheeled mobile robots is imprecise in comparison to the movement of serial manipulators due to the friction and slip between the wheel and ground contact [4]. Furthermore, a safety-related reduction in velocity further highlights the need to study the division of a mobile manipulator into two subsystems. Combining their movements can lead to inaccurate positioning and potentially hazardous situations and much more complexity in the system. In this paper, a simplified approach, as opposed to a holistic one, is used to find a suitable position for the mobile platform that allows for efficient and safe manipulation while ensuring the optimal performance of the mobile platform and the manipulator. The combination of a mobile robot platform with a manipulator often results in a degree of freedom (DOF) greater than six and must be described as a robotic system with kinematic redundancy [5,6] for a certain task in three-dimensional space where a DOF of six would be sufficient. However, the inherent complexities and uncertainties in these systems pose several challenges in realizing their full potential.

Different from conventional, stationary industrial robots, which often have a DOF equal to six, redundant robots, as described above, can move while the position and orientation of the end-effector (EE) are fixed. Another advantage of such kinematically redundant systems is collision avoidance, while the EE follows a specific path [7]. Therefore, such systems can be used for technically difficult dexterity tasks, such as grasping inside

a tube or picking between two walls. The workspace of a stationary industrial robot is limited not only by the arm length but also by the allowed joint torques. The placement of a robotic manipulator base has to be done adequately for the task that should be fulfilled via the robotic system. For mobile manipulators, this is even more challenging since the base of the robot manipulator can be moved. This also introduces the possibility of changing the manipulator base for each task in respect to the task constraints. A human worker knows by nature how to operate during different tasks. They utilize their arm redundancy to take a comfortable arm position [8]. For example, soldering on a printed circuit board will be done with bent elbows in front of the body at a certain distance to ensure a good view. This position allows the worker the optimal handling of tools. One could name this the 'human work comfort zone' for this specific task. Modern mobile robots can be equipped with collaborative manipulators to fulfill different tasks, e.g., picking and placing or transportation. This type of manipulator is specially designed to be used alongside humans and cooperate with human workers. However, with some additional restrictions to the speed, force, and torque of these mobile platforms, they can be called collaborative mobile manipulators (see [9,10]).

The selection of a suitable path and a suitable path goal for a mobile manipulator is a crucial aspect of its operation, as it directly affects the robot's ability to perform tasks and its overall manipulability. In recent years, various approaches have been proposed in the literature to determine the optimal path for mobile manipulators.

For example, refs. [11–13] divides the problem and separates the mobile manipulator into two independent subsystems: the mobile base and the mounted serial manipulator. The authors of [14,15] use a different approach to combine the mobile base and the manipulator into one system with a higher DOF; well-known and often-used sampling-based planners are used in multiple variations within this approach. Sampling-based planners like the Rapidly Exploring Random Tree (RRT) [16], the Probabilisitic Road Map (PRM) [17], and the Expansive-Spaces Tree planner (EST) [18] or any of their variants (RRT-Connect, RRT-GoalBias, and Informed RRT, . . .) have been used due to their ability to handle high-dimensional spaces. Those types of planners have been utilized for mobile manipulators in which the entire system is treated as a single high-DOF system during the planning process. Other types of planners minimize a certain cost function; Covariant Hamiltonian Optimization for Motion Planning (CHOMP) [19] and Stochastic Trajectory Optimization for Motion Planning (STOMP) [20] algorithms are often used in the robotics motion-planning framework MoveIt [21]. Those planners use cost functions with various constraints like EE position and orientation, as well as joint torques, joint limits, and distance to obstacles (for collision avoidance).

As already mentioned by the authors before, finding an appropriate goal position for a mobile robot that satisfies a specific task is a challenging problem in robotics. The goal position not only has to be reachable for the robot but must also meet limitations such as, e.g., avoiding obstacles. To find an optimal solution, multiple factors, such as the robot's kinematic and dynamic properties, environment, and task requirements, must be considered. In [22], Sandakalum et al. give a comprehensive review of the current state-of-the-art path planning of mobile manipulators, as well as determining the mobile base goal position in general. To find a suitable goal position for the mobile base, as well as a goal configuration for the mounted manipulator, two different approaches should be mentioned: Firstly, the concept of a reachability map [23] uses the reachable workspace of a mounted serial manipulator. Secondly, the so-called capability map by Zhang et al. [24,25] uses the distribution of manipulability, based on Yoshikawa [26], in Cartesian space to guide the base placement of manipulators [27].

In a previous work, we defined the comfort zone as a region in the n-dimensional discrete joint space [28]. Each point in the so-defined comfort zone represents a certain joint configuration and fulfills multi-objective optimization, including various manipulability measures (velocity, force, and stiffness dependence). Furthermore, these points are

constrained with minimum/maximum velocities, forces, and stiffness. This was shown for a planar mobile robot with two translational and three rotational joints, as used in Figure 1.
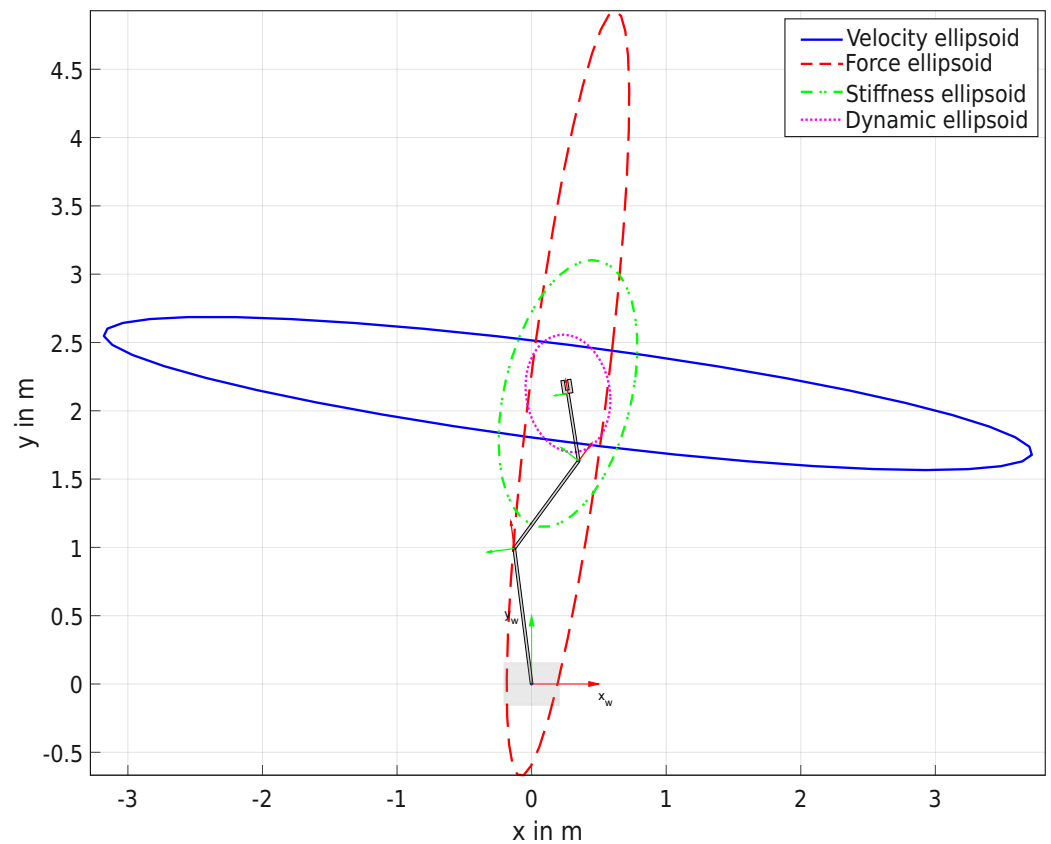

Figure 1. Geometrical representation of four different manipulability measures using a planar mobile manipulator with 5DOF.

In this paper, we propose a further method to determine a zone of possible base positions (task-dependent comfort zone) for a mobile manipulator that can be used with various individual manipulability measures, such as velocity, force, acceleration, and stiffness. Using these manipulability measures, we are able to evaluate the specific demands of different tasks and determine a zone of base positions (on a plane in Cartesian space) that can accommodate these demands. The capability of staying within this task-dependent comfort zone ensures that the mobile manipulator can perform tasks effectively and efficiently while also taking into account the need for high-speed movements, large forces, high accelerations, or high stiffness. Our results provide a comprehensive evaluation of the robot's ability to perform tasks and offer specific guidelines for selecting appropriate base-placement strategies. This enables the task-dependent, autonomous positioning of the mobile manipulator in a variety of applications. The proposed method will be shown within an example task using different types of mobile manipulators, including different serial manipulator configurations and DOF.

This paper is organized as follows: In Section 2, the Jacobian matrix is introduced, and the various manipulability measures used to evaluate the robot's performance for different tasks are described. The measures include velocity, force, acceleration, and stiffness-dependent measures based on different already-established approaches. Section 3 incorporates the manipulability measures into the proposed method to determine the task-dependent comfort zone, which represents a zone of possible positions for the mobile on a plane in Cartesian space. Section 4 describes the evaluation of the proposed method using various mobile manipulators and arm configurations. Finally, the last section summarizes our findings and provides suggestions for future work.

2. Manipulability Measures

In this section, we focus on different concepts of manipulability measures. These measures are often related to fixed-mounted industrial robots. In the literature, different measures to characterize the movement capabilities of manipulators are discussed. Specifically, here, we present an overview of manipulability measures, which are quantitative metrics used to evaluate the robot's ability to perform a specific task, based on velocities, forces, stiffness, and acceleration. Those measures will later be used to define the task-dependent comfort zone for mobile manipulators. The geometrical interpretation, so-called manipulability ellipsoids, of those measures used on force, velocity, stiffness, and dynamic manipulability measures can be seen in Figure 1. Here, for a two-dimensional case, a planar mobile robot with two translational and three rotational joints is used, and it results in this case in four manipulability ellipses.

2.1. Jacobian Matrix

The Jacobian matrix is a fundamental tool in the analysis and control of serial manipulators, serving as a representation of the relationship between the joint velocities and the EE velocity. Furthermore, it has important properties that can be leveraged to develop the manipulability measures used in this paper. The kinematic relation between the joint velocities

$$\dot{\mathbf{q}} = \frac{d\mathbf{q}}{dt} = \begin{bmatrix} \dot{q}_1 & \dot{q}_2 & \cdots & \dot{q}_n \end{bmatrix}^T \in \mathbb{R}^n \tag{1}$$

and the EE velocities

$$\dot{\boldsymbol{\zeta}} = \frac{d\boldsymbol{\zeta}}{dt} = \begin{bmatrix} \dot{\zeta}_1 & \dot{\zeta}_2 & \cdots & \dot{\zeta}_m \end{bmatrix}^T \in \mathbb{R}^m \tag{2}$$

is described by the manipulator Jacobian matrix $\mathbf{J}(\mathbf{q}) \in \mathbb{R}^{m \times n}$ with

$$\dot{\boldsymbol{\zeta}} = \underbrace{\frac{\partial \boldsymbol{\zeta}}{\partial \mathbf{q}}}_{\mathbf{J}(\mathbf{q})} \frac{\partial \mathbf{q}}{\partial t} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}} \tag{3}$$

and

$$\mathbf{J}(\mathbf{q}) = \begin{bmatrix} \frac{\partial f_1}{\partial q_1} & \frac{\partial f_1}{\partial q_2} & \cdots & \frac{\partial f_1}{\partial q_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial q_1} & \frac{\partial f_m}{\partial q_2} & \cdots & \frac{\partial f_m}{\partial q_n} \end{bmatrix}. \tag{4}$$

The EE velocity $\dot{\boldsymbol{\zeta}}$ includes linear and angular velocities, so the Jacobian matrix

$$\mathbf{J}(\mathbf{q}) = \begin{bmatrix} \mathbf{J}_T(\mathbf{q}) \\ \mathbf{J}_R(\mathbf{q}) \end{bmatrix} \tag{5}$$

can be separated into a translational part, $\mathbf{J}_T$, relating to the contribution of the joint velocities to the EE linear velocities, and a rotational part, $\mathbf{J}_R$, relating to the contribution of the joint velocities to the EE angular velocities. Moreover, the Jacobian matrix is used to map EE forces to joint forces and torques [29]. Let $\boldsymbol{\tau}_q$ denote the vector of joint torques for revolute joints and forces for prismatic joints. The principle of virtual work can be described by

$$\delta W_{\emptyset_q} = \boldsymbol{\tau}_q^T \delta \mathbf{q} \tag{6}$$

for the configuration space and

$$\delta W_{f_{EE}} = \mathbf{f}_{EE}^T \delta \boldsymbol{\zeta} = \mathbf{f}_{EE}^T \frac{\partial \boldsymbol{\zeta}}{\partial \mathbf{q}} \partial \mathbf{q} , \tag{7}$$

for the task space, with $\delta\mathbf{q}$ being the virtual displacements of the robot's joints and $\delta\boldsymbol{\xi}$ being a virtual displacement of the EE. The virtual work in joint space and task space has to be equal,

$$\delta W_{\varnothing_{\mathrm{q}}} = \delta W_{\mathrm{f}_{\mathrm{EE}}}. \tag{8}$$

Thus, Equations (6) and (7) can be reformulated to

$$\delta\mathbf{q}^{T}(\boldsymbol{\tau}_{\mathrm{q}} - \mathbf{J}^{T}\mathbf{f}_{\mathrm{EE}}) = 0. \tag{9}$$

Finally, the mapping between joint torques and EE forces can be written as

$$\boldsymbol{\tau}_{\mathrm{q}} = \mathbf{J}^{T}\mathbf{f}_{\mathrm{EE}}. \tag{10}$$

2.2. Velocity Manipulability Measure

Yoshikawa [26] uses the Jacobian manipulator to calculate a manipulability measure. This manipulability measure shows the ability of the robot's end-effector to move with a velocity in a certain direction. The calculation of those measurements uses the idea of manipulability ellipsoids [30]. It also represents a distance to a singular joint configuration. If the joint rates satisfy $||\dot{\mathbf{q}}|| = 1$, this assumes a unit sphere in the n-dimensional joint-velocity space. This unit joint-velocity condition can be written as

$$\dot{\mathbf{q}}^{T}\dot{\mathbf{q}} = 1, \tag{11}$$

and with the use of Equation (3), we get

$$\begin{aligned}
(\mathbf{J}^{-1}\dot{\boldsymbol{\xi}})^{T}\mathbf{J}^{-1}\dot{\boldsymbol{\xi}} &= \\
\dot{\boldsymbol{\xi}}^{T}\mathbf{J}^{-T}\mathbf{J}^{-1}\dot{\boldsymbol{\xi}} &= \\
\dot{\boldsymbol{\xi}}^{T}(\mathbf{J}\mathbf{J}^{T})^{-1}\dot{\boldsymbol{\xi}} &= 1.
\end{aligned} \tag{12}$$

If $\mathbf{J}$ has a full rank, the matrix $\mathbf{J}\mathbf{J}^{T}$ is square, symmetric, and positively definite.

For any symmetric and positive definite matrix $\mathbf{J}\mathbf{J}^{T}$, the set of vectors $\dot{\boldsymbol{\xi}}$ that satisfy Equation (12) defines an ellipsoid in the *m*-dimensional space. Furthermore, the volume of such an ellipsoid in the used EE task space [31] can be employed as a manipulability measure, which can be further defined as

$$\mathrm{m}_{\mathrm{v}}(\mathbf{q}) = \sqrt{\det(\mathbf{J}\mathbf{J}^{T})}. \tag{13}$$

For the case $m = n$, which means $\mathbf{J}$ is a square matrix, this relation can be simplified to

$$\mathrm{m}_{\mathrm{v}}(\mathbf{q}) = \|\det(\mathbf{J})\|, \tag{14}$$

which will not further be considered. The velocity manipulability measure in Equation (13) is proportional to the volume of the velocity ellipsoid [26,32], whose semi-axes correspond to the square roots of the eigenvalues of $\mathbf{A}_{\mathrm{v}} = \mathbf{J}\mathbf{J}^{T}$ and point in the direction of the eigenvectors of $\mathbf{A}_{\mathrm{v}}$. Near a singular configuration, the velocity ellipsoid is increasingly compressed. In the singularity, it collapses; thereby, the distance to a singularity can also be evaluated. For the two-dimensional example in Figure 1, it becomes a line. The velocity ellipsoid shown gives a quantitative measure of the ability of the EE to move in any direction. Larger values of m_{v} represent greater freedom for the movement of the EE in a specific configuration [7].

2.3. Force Manipulability Measure

Similarly, the attainable forces and moments of EE can be represented by the so-called force ellipsoid [7,30,33]. If the joint torques satisfy $||\boldsymbol{\tau}_q|| = 1$, this assumes a unit sphere in the n-dimensional joint-torque space. This unit joint-torque condition can be written as

$$\boldsymbol{\tau}_q^T \boldsymbol{\tau}_q = 1. \tag{15}$$

With the substitution of Equation (10) in (15)

$$
\begin{aligned}
(\mathbf{J}^T \mathbf{f}_{EE})^T \mathbf{J}^T \mathbf{f}_{EE} &= 1 \\
\mathbf{f}_{EE}^T \mathbf{J} \mathbf{J}^T \mathbf{f}_{EE} &= 1
\end{aligned}
\tag{16}
$$

similar results as in the calculation of Equation (13) can be obtained. With the resulting matrix $\mathbf{A}_f = (\mathbf{J}\mathbf{J}^T)^{-1}$, the so-called force ellipsoid can be calculated. The shape of this ellipsoid is defined according to the eigenvalues and eigenvectors of $\mathbf{A}_F = (\mathbf{J}\mathbf{J}^T)^{-1}$. The eigenvectors point in the same direction as those of $\mathbf{A}_v$ (the velocity ellipsoid), and the Eigenvalues are reciprocal to those of the velocity ellipsoid. With the calculated quadratic matrix $\mathbf{A}_f$, the force manipulability measure m_f with

$$m_f = \sqrt{\det(\mathbf{A}_f)} = \sqrt{\det((\mathbf{J}\mathbf{J}^T)^{-1})} \tag{17}$$

can be calculated.

2.4. Stiffness Manipulability Measure

Ajoudani et al. [34] and Chen et al. [35] discuss the Cartesian stiffness of modern industrial manipulators, depending on their joint configuration and the normal stiffness performance measure. Basics regarding stiffness calculations for articulated, rigid-body systems can be found in the work of Kövecses [36]. For a discussion of joint stiffness identification, see Dumas et al. [37]. Since stiffness (in general, $\mathbf{K}$) or compliance, $\mathbf{C} = \mathbf{K}^{-1}$, must also be taken into account when using collaborative manipulators, the following section uses a measure of the Cartesian stiffness, $\mathbf{K}_c$, of the EE. The joint stiffness k_i with $i = 1, ..., n$ of each joint is collected in a joint-stiffness matrix

$$\mathbf{K}_q = \begin{bmatrix} k_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & k_n \end{bmatrix}. \tag{18}$$

Busson et al. [38] use joint compliance and the theory of virtual work to describe the mapping between EE and joints. Sallsbury [39] shows how to calculate the Cartesian stiffness by the usage of the joint stiffness matrix $\mathbf{K}_q$ and the manipulator Jacobian $\mathbf{J}(\mathbf{q})$ by following a generalization of the linear spring relationship $F_{spring} = k_{spring}\Delta x$ with Δx describing a change in the length of a linear spring element. Using compliance, virtual displacement, and the substitution of

$$\delta\mathbf{q} = \mathbf{C}_q \boldsymbol{\tau}_q, \tag{19}$$

$$\delta\boldsymbol{\xi} = \mathbf{C}_x \mathbf{f}_{EE} \tag{20}$$

in

$$\delta\boldsymbol{\xi} = \mathbf{J}\delta\mathbf{q} \tag{21}$$

results in

$$\mathbf{C}_x \mathbf{f}_{EE} = \mathbf{J}\mathbf{C}_q \boldsymbol{\tau}_q. \tag{22}$$

After the substitution of Equation (10) in (22), one obtains

$$\mathbf{C}_x \mathbf{f}_{EE} = \mathbf{J} \mathbf{C}_q \mathbf{J}^T \mathbf{f}_{EE} \tag{23}$$

and with the elimination of forces and the usage of the relation between stiffness and compliance, the Cartesian stiffness matrix can be calculated as

$$\mathbf{K}_x = (\mathbf{J} \mathbf{K}_q^{-1} \mathbf{J}^T)^{-1}, \tag{24}$$

which can be further reformulated as

$$\mathbf{K}_x = \mathbf{J}^{-T} \mathbf{K}_q \mathbf{J}^{-1}. \tag{25}$$

Here, additional terms depending on EE forces and torques will be neglected [37]. The Cartesian stiffness described in Equation (24) can also be represented as an ellipsoid [34,40]. Eigenvalues λ of $\mathbf{K}_x$ correspond to the length of the semi-axes and can be used to characterize the stiffness of the manipulator in a particular configuration. Here, as a representation of the manipulator stiffness, the smallest eigenvalue of the Cartesian stiffness matrix,

$$\mathrm{m}_{st}(\mathbf{q}) = \min(\mathrm{eig}(\mathbf{K}_x)) \tag{26}$$

will be used as a stiffness measure.

2.5. Dynamic Manipulability Measure

The previously discussed manipulability measures only use system kinematics to evaluate the manipulability of a robotic system. The dynamic manipulability measure first defined by Yoshikawa [41] in 1985 was used by Chiacchio and Concilio [42] in 1998 to define a new dynamic manipulability ellipsoid for redundant manipulators. This new formulation of the manipulability ellipsoid is used to evaluate manipulator capabilities in terms of task-space acceleration and define a dynamic manipulability measure. For a full overview, we will present the calculations for this formulation in short form in the following section. Differentiating Equation (3) with respect to time leads to

$$\ddot{\boldsymbol{\xi}} = \mathbf{a} = \mathbf{J}(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{J}}(\mathbf{q})\dot{\mathbf{q}}. \tag{27}$$

The manipulator dynamics can be written as

$$\boldsymbol{\tau}_q = \mathbf{B}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{f}_{fric} + \boldsymbol{\tau}_q^{COM} + \mathbf{J} \mathbf{f}_{EE}, \tag{28}$$

where $\boldsymbol{\tau}_q \in \mathbf{R}^n$ is the vector of n joint torques, $\mathbf{f}_{EE} \in \mathbf{R}^m$ is the vector of m EE forces, $\mathbf{B} \in \mathbf{R}^{n \times n}$ is the inertia matrix, $\mathbf{C} \in \mathbf{R}^{n \times n}$ includes the Coriolis and centrifugal terms, and $\boldsymbol{\tau}_q^{COM}$ is the vector of gravitational force terms. For a manipulator in a given configuration and without the EE interacting with its environment, we consider the joint speed $\dot{\mathbf{q}} = 0$, the friction force $\mathbf{f}_{fric} = 0$, and the external forces $\mathbf{f}_{EE} = 0$. Using the proposed simplifications, and combining the robot dynamics with Equation (27) and solving for $\ddot{\boldsymbol{\xi}}$, leads to

$$\ddot{\boldsymbol{\xi}} = \mathbf{J} \mathbf{B}^{-1} \boldsymbol{\tau} - \mathbf{J} \mathbf{B}^{-1} \boldsymbol{\tau}_q^{COM}. \tag{29}$$

Using a scaling matrix, $\mathbf{T}_{max}$, with the maximum joint torques $\mathbf{T}_{max} = \mathrm{diag}(\tau_1^{max}, \dots, \tau_n^{max})$ and the inertia matrix $\mathbf{B}$, the matrix

$$\mathbf{Q} = (\mathbf{T}_{max}^{-1} \mathbf{B})^T \mathbf{T}_{max}^{-1} \mathbf{B} \tag{30}$$

can be introduced. Combining the results and using a pseudo-inverse, the weighted Jacobian matrix is

$$\mathbf{J}_Q^+ = \mathbf{Q}^{-1}\mathbf{J}^T(\mathbf{J}\mathbf{Q}^{-1}\mathbf{J}^T)^{-1} \tag{31}$$

and the dynamic manipulability ellipsoid can be defined as

$$(\ddot{\boldsymbol{\xi}} + \mathbf{J}\mathbf{B}^{-1}\boldsymbol{\tau}_q^{COM})^T(\mathbf{J}_Q^+)^T\mathbf{Q}\mathbf{J}_Q^+(\ddot{\boldsymbol{\xi}} + \mathbf{J}\mathbf{B}^{-1}\boldsymbol{\tau}_q^{COM}) \leq 1. \tag{32}$$

By using the core of Equation (32), this results in the definition of the dynamic acceleration matrix

$$\mathbf{N} = (\mathbf{J}_Q^+)^T\mathbf{Q}\mathbf{J}_Q^+. \tag{33}$$

When calculating the eigenvalues of the matrix $\mathbf{N}$, a dynamic manipulability measure can be defined as

$$m_a(\mathbf{q}) = \min(\mathrm{eig}(\mathbf{N})), \tag{34}$$

which represents the minimal eigenvalue of the dynamic acceleration matrix in a certain joint configuration.

In the next section, we use the manipulability measures (m_v, m_f, m_{st}, and m_a) based on force, velocity, stiffness, and acceleration, discussed here and summarized in Table 1 to develop a method for defining the task-dependent comfort zone for a mobile manipulator, taking into account different task requirements. The resulting task-dependent comfort zone will provide more specific guidelines for selecting appropriate base-placement strategies, ultimately improving the robot's ability to perform tasks. Later, we define the term manipulability measure, $m_x(\mathbf{q})$, with $x \in \{v, f, st, a\}$ in general for any measure that can evaluate a particular manipulator joint configuration using a scalar value.

Table 1. Summary of manipulability measures used.

Symbol	Formula	Description
m_v	Equation (13)	Proportional to the volume of the EE velocity ellipsoid, which represents the ability to move the EE with a certain velocity in all directions.
m_f	Equation (17)	Proportional to the volume of the EE force ellipsoid, which represents the ability to act with a certain force in all directions.
m_a	Equation (26)	Represents the minimum eigenvalue of the Cartesian stiffness matrix, which characterizes the smallest stiffness in a certain configuration.
m_a	Equation (34)	Represents the minimum eigenvalue of the weighted dynamic manipulability matrix, which characterizes the smallest acceleration in a certain direction.

3. Task-Dependent Comfort Zone

In our previous work, the comfort zone was defined by combining the manipulability measures with the help of a multi-objective optimization [28]. In this approach, with multi-objective optimization, the problem arises in which the used measures are related, and in some cases, they are inversely proportional. Here, a different approach is considered to increase the variable usability of the proposed algorithm by choosing one measure at a time. However, care was taken to retain the possibility of performing a combination of the used measures.

3.1. Task Classification

In modern industrial environments, the number of instances of and the need for human–robot interaction, and thereby collaborative tasks, continue to increase. Thus, tasks can be classified according to requirements for a robotic system [43]. As humans, we can classify tasks according to forces on our bodies arising from interactions with the environment. This can be done due to our great and fast sensory capabilities and by learning how to interact with the environment [44]. To classify a robotic task, many different approaches are possible. For example, the motion constraints and the total force/torque acting on the system could be used [45]. In Table 2, we show a selection of tasks and try to

classify them according to their velocity, force, stiffness, and acceleration requirements so as to then choose the suitable manipulability measure for a certain task. Of course, these are just general examples, and the specific requirements for each task will depend on the details of the task. It shows that our concept of a task-dependent comfort zone can be extended by using a different manipulability measure. Depending on the task, a directional manipulability measure [46] may be preferable, as opposed to the more general approach discussed in Section 2.2.

Table 2. Possible classification of robotic tasks.

Task Type	Velocity	Force	Stiffness	Acceleration
Pick and Place	High	Moderate	Low	High
Assembly	Moderate	High	High	Moderate
Painting	Moderate	Low	Low	Low
Milling	Low	High	High	Low

3.2. Motivation

For any mobile manipulator, which is a combination of a mobile robot platform and a serial manipulator, the $p + m$ joint variables can be separated as

$$\mathbf{q} = [\mathbf{q}_\mathrm{P}, \ \mathbf{q}_\mathrm{M}]^T = [q_{\mathrm{P}1}, \ q_{\mathrm{P}2}, \ \ldots, \ q_{\mathrm{P}p}, \ q_{\mathrm{M}1}, \ q_{\mathrm{M}2}, \ \ldots, \ q_{\mathrm{M}m}]^T, \tag{35}$$

where p is the number of joints of the mobile platform, and m is the number of joints of the serial manipulator. Any pose of a mobile robot platform on a plane can be described with a three-dimensional vector:

$$\mathbf{p} = (x_\mathrm{P}, y_\mathrm{P}, \theta_\mathrm{P}) \in \mathbb{R} \times \mathbb{R} \times [0, 2\pi), \tag{36}$$

x_P and y_P are the coordinates, and θ_P is the rotation around the local z-axis of the mobile robot. The pose of the EE of a robot manipulator can be described using

$$\boldsymbol{\xi}_\mathrm{EE} = [\mathbf{x}_\mathrm{EE}, \ \boldsymbol{\phi}_\mathrm{EE}]^T = [x_\mathrm{EE}, \ y_\mathrm{EE}, \ z_\mathrm{EE}, \ \phi_{x,\mathrm{EE}}, \ \phi_{y,\mathrm{EE}}, \ \phi_{z,\mathrm{EE}}]^T, \tag{37}$$

with $\boldsymbol{\phi}_\mathrm{EE}$ standing for the description of the orientation. We use Euler angles with rotation around the (x, y, z)-axis, and $\mathbf{x}_\mathrm{EE}$ is the position of the EE. This can also be expressed using the homogeneous transformation matrix $\mathbf{T}_\mathrm{EE}$. The problem is now to find feasible placements for the mobile base $\mathcal{P} = (\mathbf{p}_1, \ldots, \mathbf{p}_{n_p})$ with $\mathbf{p}_i = (x_\mathrm{P}, y_\mathrm{P}, \theta_\mathrm{P}) \in \mathbb{R} \times \mathbb{R} \times [0, 2\pi) \forall i \in \{1, 2, 3, \ldots, n_p\}$, where the EE can achieve the desired pose described by the homogeneous transformation matrix $\mathbf{T}_\mathrm{des}$. The manipulation of an object located on a desk is used to motivate the employed algorithm. The task focuses on the fast velocity of the EE. Thus, in this description, only the velocity-dependent manipulability measure will be taken into account. This task can be seen in a lot of standard industrial applications, as well as in the rules of the RoboCup@Work competition [47]. The schematic drawing shown in Figure 2a depicts the principle of the task setup.

The pose of the object is described by the desired pose vector $\boldsymbol{\xi}_\mathrm{EE,des} = [\mathbf{x}_\mathrm{EE,des}, \ \boldsymbol{\phi}_\mathrm{EE,des}]^T$, and it will, further on, be used as the starting point to calculate the task-dependent comfort zone for a suitable mobile base placement. Note that the calculation of the manipulability measures does not include the joints of the mobile platform because excluding the platform joints allows for the reservation of this degree of freedom for navigation within the designated comfort zone, and the orientation of the platform is often influenced by external factors, such as spatial constraints.
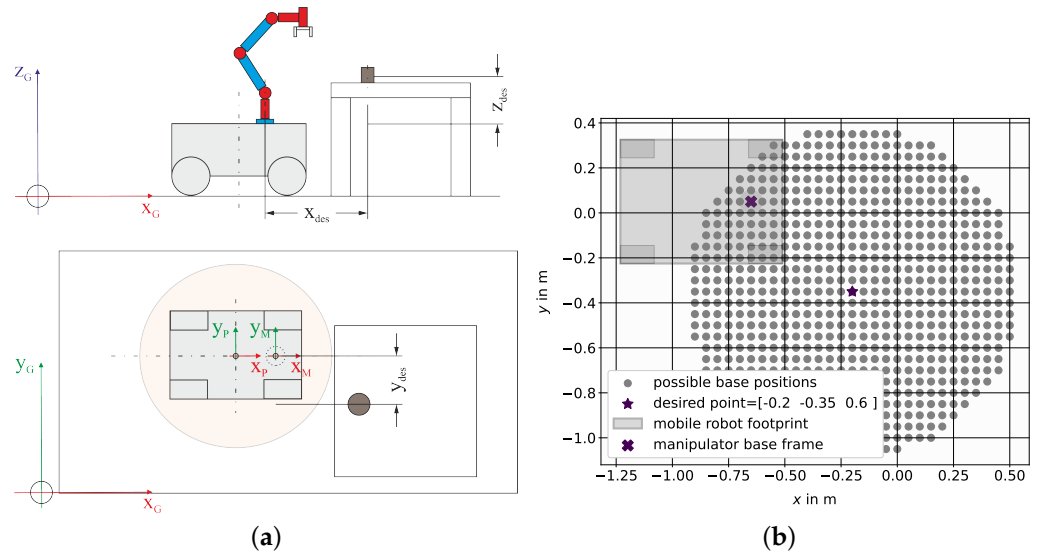
Figure 2. Schematic overview for a pick-and-place task: (**a**) mobile platform (light gray), manipulator (links in blue and joints in red), and object (dark gray). (**b**) Feasible discrete mobile base positions for a desired EE goal.

3.3. Recommended Workspace

While using a serial link-structured robotic manipulator with rotational joints, the values for the minimal and maximal EE (x, y)-coordinates, $x_{\mathrm{EE,min/max}}$ and $y_{\mathrm{EE,min/max}}$, are dependent of the desired z position. A subspace of the manipulators' workspace where all positions and all orientations of the EE can be achieved is called a recommended or dexterous workspace. To achieve all orientations $\boldsymbol{\phi}_{\mathrm{EE,des}}$ of the EE, the desired position $\mathbf{P}_{\mathrm{EE,des}} = [x_{\mathrm{EE,des}}, y_{\mathrm{EE,des}}, z_{\mathrm{EE,des}}]^T$ should be inside the recommended workspace of the robotic manipulator.

The workspace boundaries and workspace characterization of serial manipulators have been extensively researched [7,48–50]. Here, for simplification, the outer limits of the recommended workspace for serial link manipulators are described using a triaxial ellipsoid as an approximation,

$$\frac{(x - x_{\mathrm{M,Base}})^2}{a^2} + \frac{(y - y_{\mathrm{M,Base}})^2}{b^2} + \frac{(z - z_{\mathrm{M,Base}})^2}{c^2} = 1,\tag{38}$$

with parameters a,b,c beeing the length of the three half-axes, which are specific to the serial manipulator used, and the origin of the serial robot manipulator base frame is

$$\mathbf{P}_{\mathrm{M,Base}} = [x_{\mathrm{M,Base}}, \ y_{\mathrm{M,Base}}, \ z_{\mathrm{M,Base}}]^T.\tag{39}$$

Unreachable points within this simplified workspace are excluded while the proposed comfort zone is calculated.

By varying the (x, y)-positions of the mobile platform while holding the desired position of the EE fixed in the world frame, the configuration of the robotic manipulator can be changed. This set of configurations will be searched to find the comfort zone for the mobile manipulator. To find the inequality constraint equation for a point inside the recommended workspace, a plane is used to cut the recommended workspace ellipsoid per Equation (38) at point $\mathbf{P}_{\mathrm{des}}$ with varying (x, y)-coordinates and a fixed z-coordinate. Therefore, this plane, in its normal form, is

$$\mathbf{x} \cdot \mathbf{n}_0 = \mathbf{P}_{\mathrm{EE,des}} \cdot \mathbf{n}_0\tag{40}$$

with $\mathbf{n}_0 = [0,\ 0,\ 1]^T$ said to be parallel to the (x,y)-plane, resulting in

$$z = z_{\text{EE,des}}. \tag{41}$$

When combining Equations (38) and (41), an ellipse parallel to the (x,y)-plane with its center at the point $\mathbf{P}_{\text{M,Base}}$:

$$E_{\text{EE}}(x,y) := \frac{x^2}{a^2(1 - \frac{z^2_{\text{EE,des}}}{c^2})} + \frac{y^2}{b^2(1 - \frac{z^2_{\text{EE,des}}}{c^2})} = 1 \tag{42}$$

and the axes lengths

$$\hat{a} = \sqrt{a^2(1 - \frac{z^2_{\text{EE,des}}}{c^2})} \tag{43}$$

$$\hat{b} = \sqrt{b^2(1 - \frac{z^2_{\text{EE,des}}}{c^2})} \tag{44}$$

can be found. Using the boundary condition

$$\frac{x^2}{\hat{a}} + \frac{y^2}{\hat{b}} \le 1 \tag{45}$$

defined by the ellipse in Equation (42), a set of points, $\mathcal{P}_{\text{EE}} \in \mathbb{R}^2$, can be calculated. This set includes all points that can be reached via the EE while the mobile base is placed at the point $\mathbf{P}_{\text{M,Base}}$, and it can be defined as

$$\mathcal{P}_{\text{EE}} = \left\{ \mathbf{P}_{\text{EE}}(x,y) \in \mathbb{R}^2 \mid \frac{x^2}{\hat{a}} + \frac{y^2}{\hat{b}} \le 1 \right\}. \tag{46}$$

Using the desired EE position $\mathbf{P}_{\text{EE,des}}$ and all points $\mathbf{P}_{\text{EE}} \in \mathcal{P}_{\text{EE}}$, a new set

$$\mathcal{P}_{\text{Base}} = \left\{ \mathbf{P}_{\text{Base}} \in \mathbb{R}^2 \mid \mathbf{P}_{\text{Base}}(x,y) = \mathbf{P}_{\text{EE}}(x,y) + \mathbf{P}_{\text{EE,des}} \ \forall \ \mathbf{P}_{\text{EE}}(x,y) \right\} \tag{47}$$

of feasible base positions can be found. Assuming a planar movement in the (x,y)-plane of the mobile base, and due to Equation (41), z_{Base} is set to zero, and it will further be neglected. Each element in the new set $\mathcal{P}_{\text{Base}}$ represents a suitable position of the mobile base to ensure the desired EE position, $\mathbf{P}_{\text{des}}$, using different manipulator-joint configurations. By evaluation of the continuous set $\mathcal{P}_{\text{Base}}$ with discrete values for $\mathbf{x} = \hat{\mathbf{x}}$ and $\mathbf{y} = \hat{\mathbf{y}}$ with

$$\hat{\mathbf{x}} = [x_{\min}, x_{\min} + \Delta x, \dots, x_{\max} - \Delta x, x_{\max}] \tag{48}$$

and

$$\hat{\mathbf{y}} = [y_{\min}, y_{\min} + \Delta x, \dots, y_{\max} - \Delta x, y_{\max}] \tag{49}$$

the discrete set

$$\hat{\mathcal{P}}_{\text{Base}} = \left\{ \hat{\mathbf{P}}_{\text{Base},i}(\hat{\mathbf{x}}, \hat{\mathbf{y}}) \in \mathcal{P}_{\text{Base}} \right\} \tag{50}$$

with n_{Base} points can be found. Figure 2b shows an exemplary set of $n_{\text{Base}} = 460$ feasible positions for a mobile base with a mounted serial manipulator. The discrete set of points $\hat{\mathcal{P}}_{\text{Base}}$ is used to find a subset, defined as a task-dependent comfort zone (see Equation (54)) for mobile manipulator placement. Thereby, different manipulability measures, $m_v(\mathbf{q})$, $m_f(\mathbf{q})$, $m_{\text{st}}(\mathbf{q})$, $m_a(\mathbf{q})$, as described in the previous sections, are used to evaluate the corresponding joint configuration for each point in the discrete set.

3.4. Manipulability Measure Norm, Combination, and Constraints

In order to obtain an equal weighting of the different measures and set suitable boundary conditions, the measures are normed with the help of the minimum and maximum values out of a set of possible points, $\hat{\mathcal{P}}_{\text{Base}}$. The used manipulability measure m_x can be normalized into a range, $0 \leq \hat{m}_x \leq 1$, using $f_{\text{Norm}}(m_x) = \hat{m}_x$ with

$$f_{\text{Norm}}(m_{x,i}) = \frac{m_{x,i} - m_{x,\text{min}}}{m_{x,\text{max}} - m_{x,\text{min}}} \, , \tag{51}$$

$m_{x,\text{max}}$ is the maximum, and $m_{x,\text{min}}$ is the minimum value out of the given set of manipulability measures. This is shown through an example in Figure 3a. Furthermore, the measures can be combined using a weighted sum with

$$\hat{m}_{c,i} = k_v \hat{m}_{v,i} + k_f \hat{m}_{f,i} + k_{st} \hat{m}_{st,i} + k_a \hat{m}_{a,i} \, , \forall i \in \{0, 1, \ldots, n_{\text{Base}}\} \, , \tag{52}$$

where $0 \leq k_x \leq 1$ is used with $x \in \{v, f, st, a\}$ and $k_v + k_f + k_{st} + k_a = 1$ to maintain normalization.

(**a**)　　　　　　　　　　(**b**)
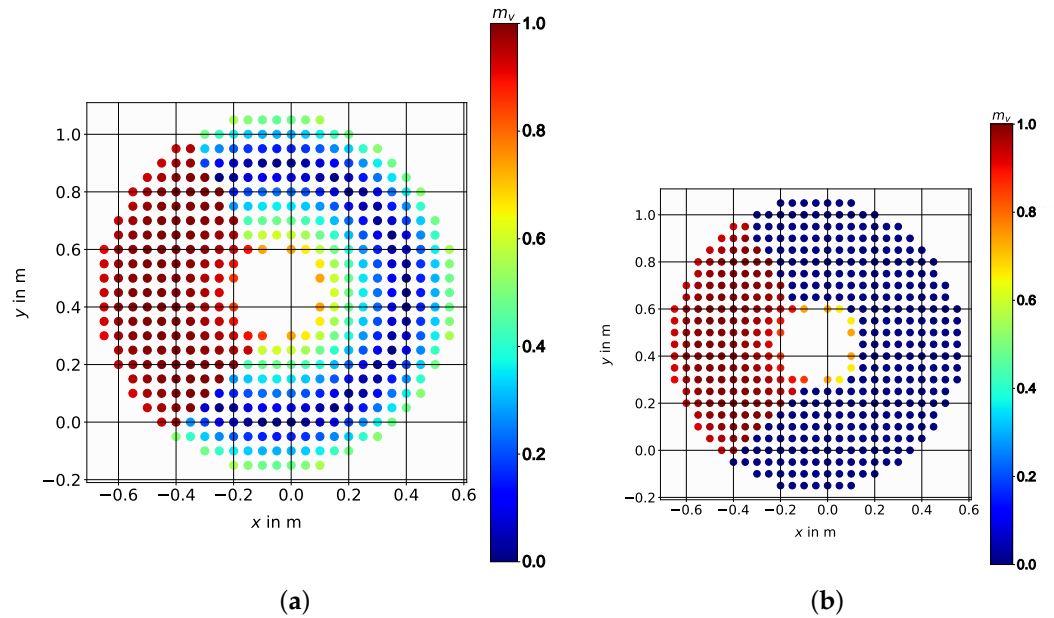
Figure 3. Discrete point-plot representation of feasible base position points for the Stanford Robotics Platform. The quality of the points is described by the manipulability measure m_v; the red points have high manipulability. The manipulability measure is dependent on the angular velocity of the EE for a desired EE pose, $\xi_{\text{EE,des}} = [-0.05, \ 0.45, \ 0.65, \ \frac{\pi}{4}, \ 0, \ \frac{-\pi}{4}]^T$. (**a**) Normalized values of m_v from 0 to 1; (**b**) comfort zone $\mathcal{A}_{\text{zone}}$ for a desired EE point, represented using red dots with a minimum of 85% manipulability.

3.5. Comfort Zone Definition

The discrete set of points $\mathcal{P}_{\text{Base}}$ can be constrained into a feasible zone by using constraints for the manipulability measure

$$\hat{m}_{x,\text{low}} \leq \hat{m}_{x,i} \leq \hat{m}_{x,\text{up}} \, , \tag{53}$$

where $\hat{m}_{x,\text{low}}$ and $\hat{m}_{x,\text{up}}$ are the lower and upper constraints for each individual measure, $(\hat{m}_v, \hat{m}_f, \hat{m}_{st}, \hat{m}_a, \hat{m}_c)$. Using these constraints, a zone of possible base positions can be defined. The defined task-dependent comfort zone $\mathcal{A}_{\text{zone}}$ for mobile manipulators with

$$\mathcal{A}_{\text{zone}} = \{P_{\text{zone}}(x,y) \in \mathbb{R}^2 \mid P_{\text{zone},i}(x,y) \subset \hat{\mathcal{P}}_{\text{Base}}$$
$$\wedge (\hat{m}_{x,\text{low}} \leq \hat{m}_{x,i} \leq \hat{m}_{x,\text{up}}) \forall i = [1, \ldots, n_{\text{Base}}]\} \tag{54}$$

includes all points P_{zone}, such that the corresponding joint configuration and its manipulability measure fulfills the set constraints, as shown in Figure 3b.

3.6. Comfort Zone Mesh Representation and Target Points

With the use of the discrete points defined by the comfort zone Equation (54), a square mesh, as shown in Figure 4a, is built. For simplification, and to avoid further computational effort with more precise discretization, the mean value out of four corner points for each square is used to interpolate the manipulability measure inside one square. We assume that, in this square, around each point, no unfeasible solution in the manipulator configuration will arise. The whole region is now used as a feasible goal for the mobile base. Nevertheless, a defined point to aim for is needed, and therefore, we use the mesh representation and transform it into pixel coordinates where each square (center point) will be represented using an individual pixel. The resulting image, shown in Figure 4b, will then be used for further investigations with image-processing algorithms from scikit-image (https://scikit-image.org/, accessed on 3 August 2024), an image-processing algorithm collection in Python [51], and OpenCV (https://docs.opencv.org/4.x/d6/d00/tutorial_py_root.html, accessed on 3 August 2024), a library of Python bindings designed to solve computer vision problems. The following algorithms are used in three steps to determine a possible goal position inside the comfort zone: Firstly, the identification of possible disconnected areas uses the scikit-image function `label` [52]. Secondly, the comparison of the resulting areas uses the scikit-image function `regionprops` [53] and, lastly, we use the OpenCV functions `distanceTransform` [54] and `minMaxLoc` to find the centerpoint and radius of the largest possible inscribed circle; see Figure 5.

(**a**) (**b**)
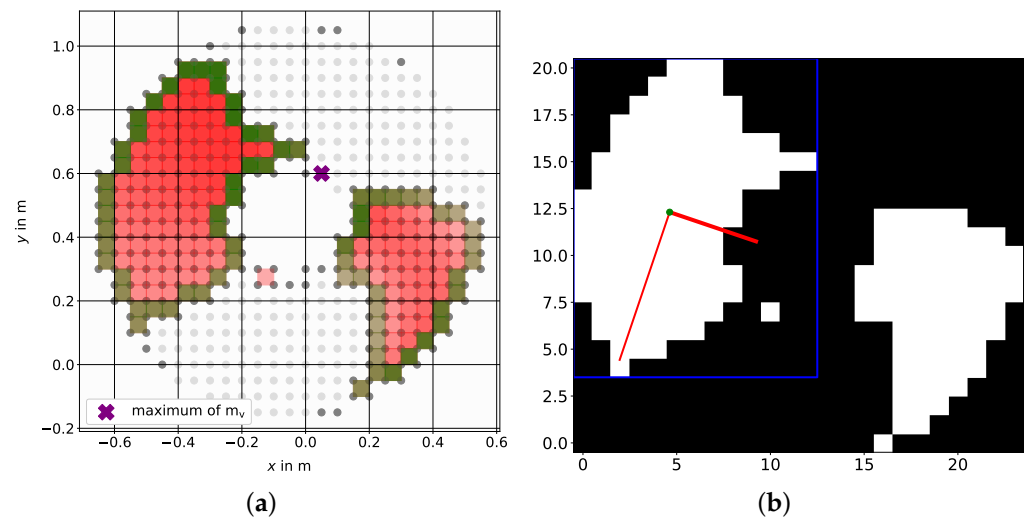
Figure 4. Comfort zone example for the Stanford mobile manipulator, transformation to pixel representation. Each red/green square in (**a**) is transformed to a white pixel in (**b**). Purple x shows the maximum value of the used manipulability mv. Area boarders are represented by green squares.

Figure 5 also shows that only the largest area found will be considered in the further investigation. Therefore, other points that fulfill the manipulability constraints will be neglected even when the highest manipulability can be found outside the proposed comfort zone. Using the center of the largest possible inscribed circle in the comfort zone increases the robustness and enables moving around within this zone without a substantial loss of manipulability.
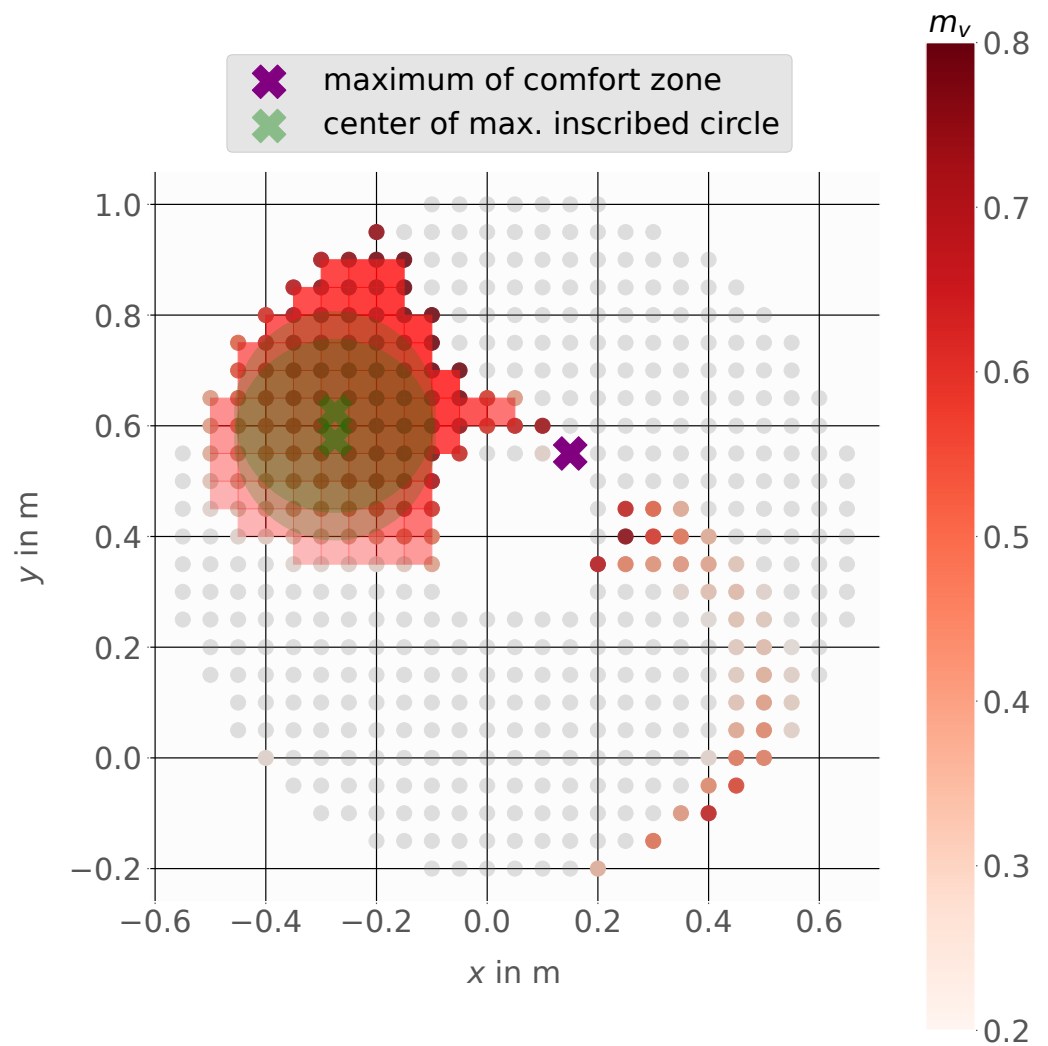
Figure 5. Comfort zone square mesh representation. The green circle represents the largest possible inscribed circle in the comfort zone with its center point marked in green. The maximum value of the used manipulability m_v is marked in purple. Each red square represents a suitable mobile base position for the desired EE goal at $z_{EE,des}$ with at least 85% of the maximum manipulability.

4. Comfort Zone Simulation Examples

We want to show the usage of the comfort zone for a given pick task and also reveal that the approach can be applied to different types of mobile manipulators. All of the mobile manipulators used (Stanford Robotics Platform, LeoBot, and Kairos) are equipped with a holonomic mobile base to achieve omnidirectional movement. Each of these bases supports different types of serial manipulators. The results of this evaluation will provide insight into the practical usefulness of the proposed approach and its ability to offer specific guidelines for selecting appropriate base placements for different mobile manipulators. For the simulations, a discrete workspace with a discretization of $\Delta x = \Delta y = 0.05$ m and $\Delta z = 0.01$ m is used. The region of interest is a subdomain of the manipulators' workspace. Therefore, a cuboid with $1.6 \times 1.6 \times 0.4$ m, as shown in Figure 6, is used. The simulation examples use either a desired EE pose, $\boldsymbol{\xi}_{EE,des} = [\mathbf{x}_{EE,des}, \boldsymbol{\phi}_{EE,des}]^T$, or a desired EE position, $\mathbf{x}_{EE,des} = [x_{EE,des}, y_{EE,des}, z_{EE,des}]^T$, where the orientation of the EE is not considered in the inverse kinematics calculation. For serial manipulators with a defined number of inverse kinematics solutions (UR5 and PUMA), all feasible solutions (in terms of self-collision between the manipulator and the base) can be checked; for Panda, a numerical calculation of the inverse kinematics is used, so only one solution is checked. Also, within the results shown, only either translational or rotational parts

of the Jacobian matrix are considered while the manipulability measures are calculated. Using the full Jacobian matrix would lead to problems, as discussed in [55]. Thereby, the usage of the full Jacobian matrix has been shown to have no physical consistency due to the fact that the rotational and translational parts have different physical units. All the simulations and calculations are done using Python and the flexible multibody systems framework Exudyn [56] (https://github.com/jgerstmayr/exudyn, accessed on 3 August 2024). In the simulation examples, the inverse kinematics algorithm ikine_LMS according to Levenberg–Marquadt with Sugihara's tweak [57] from Peter Corke's robotics toolbox [58] (https://github.com/petercorke/robotics-toolbox-python, accessed on 3 August 2024), was used.
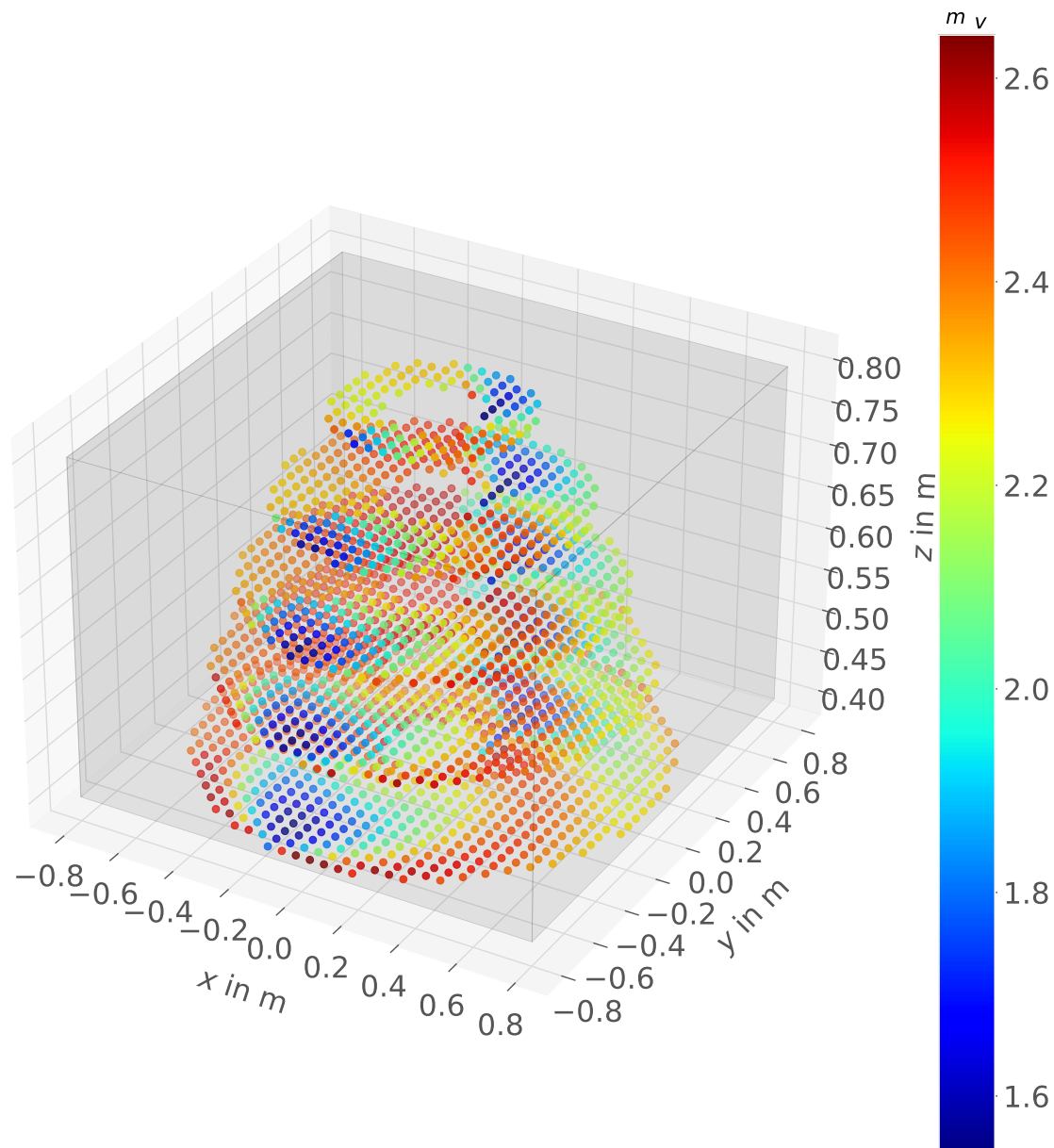
Figure 6. Discrete subdomain of the Puma560 manipulator workspace shown as a gray cuboid. Evaluation of all discrete points regarding their manipulability m_v (not normalized; $\Delta z = 0.05$).

4.1. Stanford Robotics Platform

The Stanford Robotics Platformis a mobile manipulator with a Puma560 serial manipulator (build by Unimation in Danbury, Connecticut, USA) mounted on a base using four so-called powered caster modules [59]. In Figure 7b, we show a 3D-CAD drawing, as

well as the kinematic chain. The Stanford Robotics Platformwas developed and built by the Stanford University Computer Science Department in the year 2000 [60], and it combines a Unimate Puma560 manipulator with six rotational joints and a Euler wrist configuration mounted on a Nomadic XR4000 holonomic mobile base. The Puma560 manipulator is widely used in research as an example of a classic six-DOF serial manipulator. The kinematic chain and its parameter are well described in the literature [31,61,62]. The resulting system with 9 DOF is shown in Figure 7a, including the mobile platform joints P_1, P_2, and P_3 and the manipulator joints M_1 to M_6.

(**a**)   (**b**)
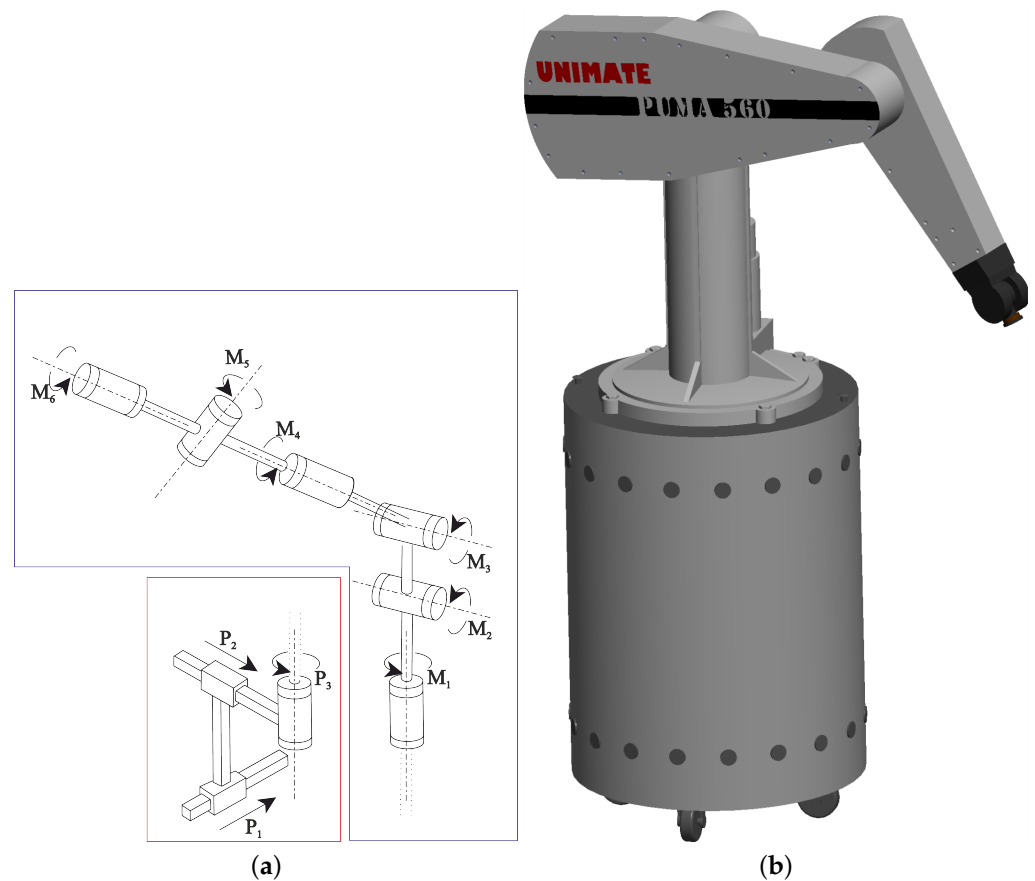
Figure 7. Serial manipulator with six rotational joints mounted on a Nomadic XR4000 holonomic mobile base. (**a**) Kinematic chain of the Stanford Robotics Platform; (**b**) Stanford Robotics Platform built by Stanford University.

In Figure 8, multiple possible placements for the mobile manipulator are shown. Both figures show the same results for the pose and position while using manipulability m_v for which only linear velocities are considered. Figure 9 shows the results for m_v based on only angular velocities whereby, especially in Figure 9a, the main purpose of the proposed comfort zone can be seen. Here, the maximum value of the manipulability is outside the proposed comfort zone; therefore, using this position leads to a lack of manipulability during operation for even a small movement of the mobile base. Using one of the proposed points (green), which are the centers of the maximum inscribed circles (light green), always leads to a manipulability of at least 65% of the maximum value.
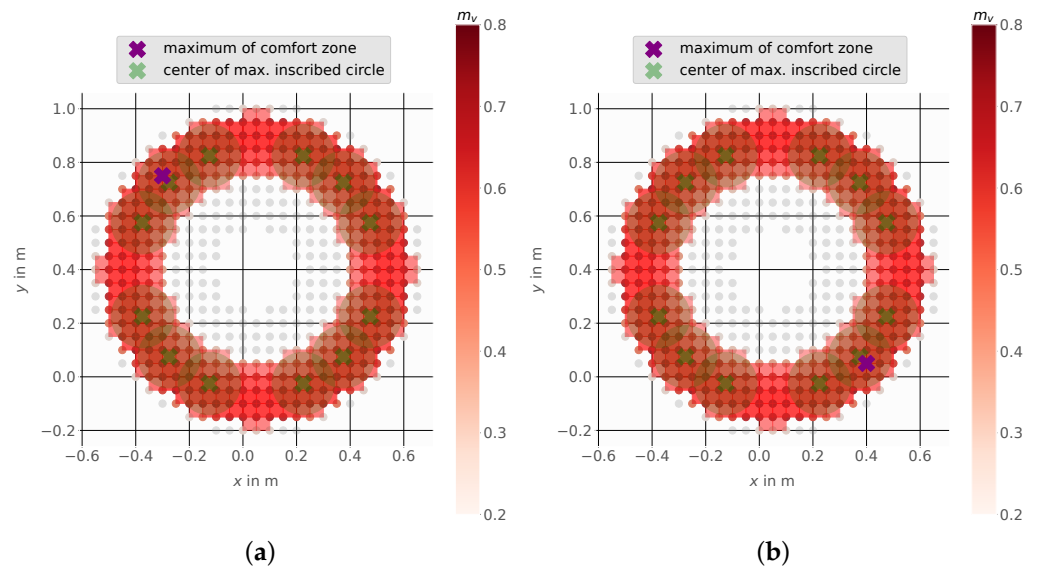
Figure 8. Comfort zone for the Stanford Robotics Platform using the linear velocity **v**-dependent manipulability measure m_v with a minimum of 65% manipulability for the EE pose (**a**) ($\xi_{EE,des} = [0.05, 0.40, 1.55, \frac{\pi}{4}, 0, \frac{-\pi}{4}]^T$) and EE position (**b**) ($\xi_{EE,des} = [0.05, 0.40, 1.55]^T$).
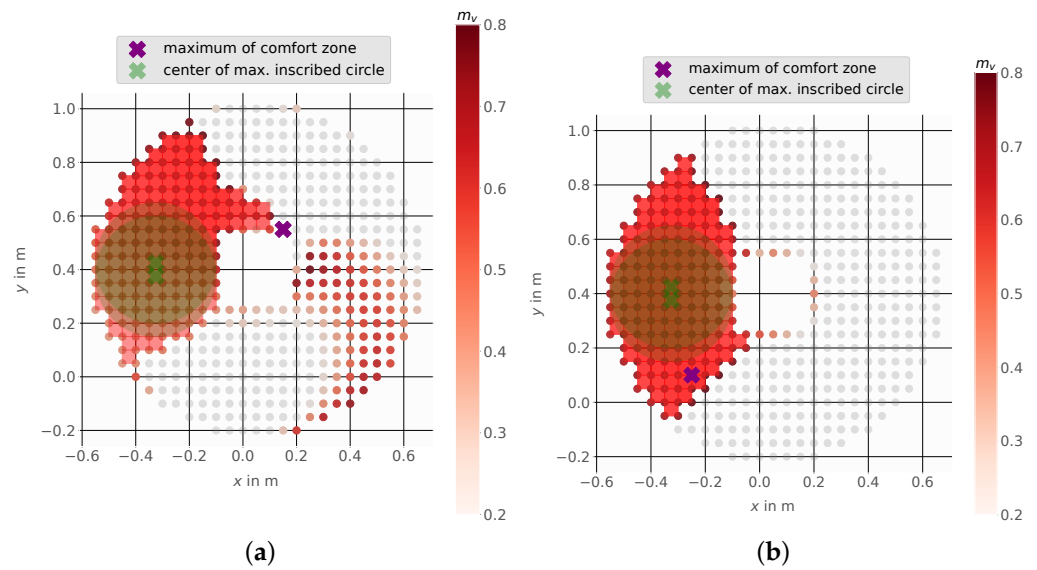

Figure 9. Comfort zone for the Stanford Robotics Platform using the angular velocity ω-dependent manipulability measure m_v with a minimum of 65% manipulability for the EE pose (**a**) ($\xi_{EE,des} = [0.05, 0.40, 1.55, \frac{\pi}{4}, 0, \frac{-\pi}{4}]^T$) and EE position (**b**) ($\xi_{EE,des} = [0.05, 0.40, 1.55]^T$).

4.2. Mobile Manipulator LeoBot

The mobile manipulator LeoBot [63], shown in Figure 10, uses Mecanum wheels for base motion with a mounted Franka Emika Panda seven-DOF serial manipulator. Figure 10a shows the kinematic chain of LeoBot which achieves a total of 10 DOF, including the mobile platform joints P_1, P_2, and P_3 and the manipulator joints M_1 to M_7. The Franka Emika Panda is widely used in research and development, and its parameters, including link masses, inertias, and center of mass, were recently investigated by researchers [64]. Figure 11 shows the results using m_v based on linear velocities. Figure 11a shows that a mobile base placement at the proposed point (the center of the maximal possible inscribed circle) offers the advantage of staying within a zone with a minimum 30% manipulability even when the platform is moved. Also, here, the maximal manipulability is outside the comfort zone, and a small movement there would lead to a great loss in manipulability.

The center points of all the inscribed circles found would be feasible mobile-platform goal positions with higher flexibility in base movement and ensure a solution for the inverse kinematics. A selection between them could be made via a comparison of the individual manipulability of each point, but for all points here, the manipulability is at least higher than or equal to 30% of the maximum value.
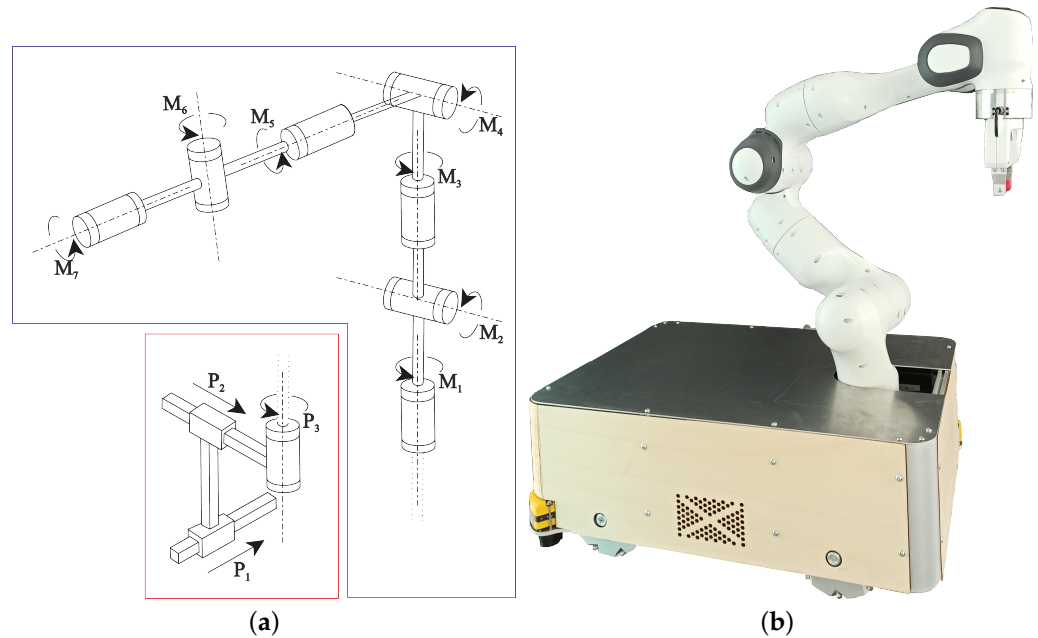

Figure 10. LeoBot mobile manipulator developed and built by the Department of Mechatronics at the University of Innsbruck in the year 2020. The mobile manipulator combines a Franka–Emika–Panda manipulator with seven rotational joints mounted on a holonomic mobile base using Mecanum wheels. (**a**) Kinematic chain of the mobile manipulator LeoBot; (**b**) mobile manipulator LeoBot.
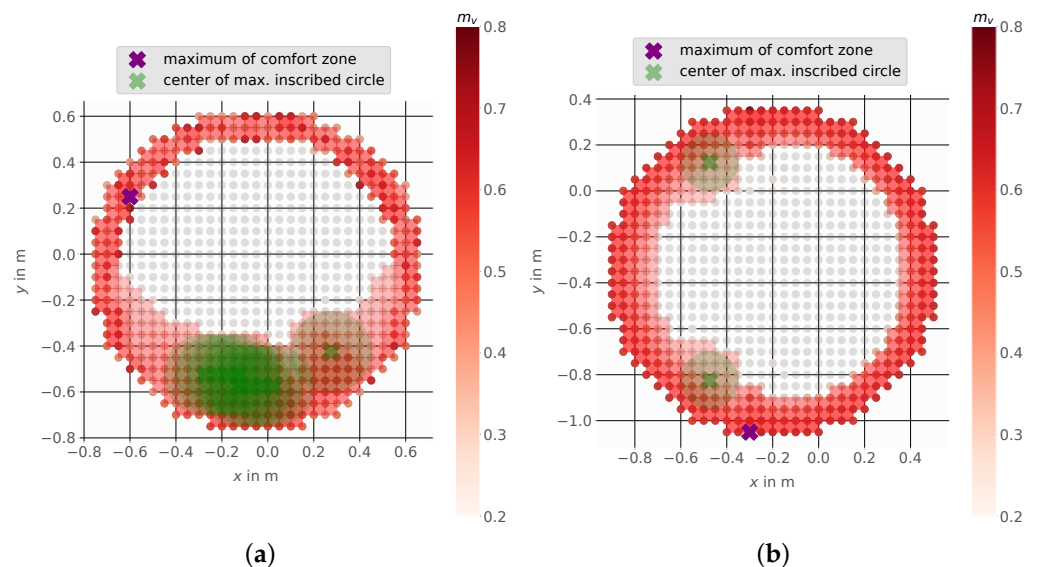
Figure 11. Comfort zone for the mobile manipulator LeoBot using the linear velocity **v**-dependent manipulability measure m_v with a minimum of 30% manipulability for the EE pose (**a**) ($\boldsymbol{\xi}_{EE,des} = [-0.05, -0.05, 0.65, \frac{\pi}{4}, 0, \frac{-\pi}{4}]^T$) and EE position (**b**) ($\boldsymbol{\xi}_{EE,des} = [-0.2, -0.35, 0.65]^T$).

Figure 12 shows the results using m_v based on angular velocities. The mobile base placement at the proposed point (the center of the maximal possible inscribed circle)

achieves a large zone with a minimum 30% manipulability. The maximum value of manipulability is located outside of the comfort zone. So, using that as a base goal position would quickly lead to a lack of manipulability.
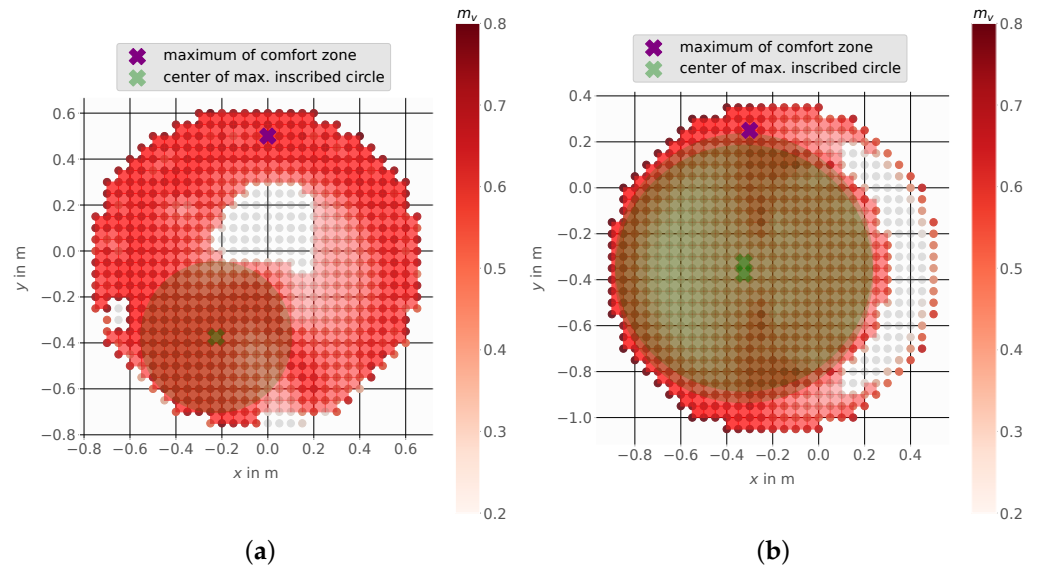

(**a**)   (**b**)

Figure 12. Comfort zone for the mobile manipulator LeoBot using the angular velocity ω-dependent manipulability measure m_v with a minimum of 30% manipulability for the EE pose (**a**) ($\zeta_{EE,des} = [-0.05, \ -0.05, \ 0.65, \ \frac{\pi}{4}, \ 0, \ \frac{-\pi}{4}]^T$) and EE position (**b**) ($\zeta_{EE,des} = [-0.2, \ -0.35, \ 0.65]^T$).

4.3. Mobile Manipulator Kairos

The mobile manipulator Kairos from the company Robotnik (Valencia, Spain) (https://robotnik.eu/, accessed on 3 August 2024), combines a mobile base using Mecanum wheels with a mounted Universal Robot UR5 serial manipulator, as shown in Figure 13. Figure 13a shows the kinematic chain of Kairos, which achieves in total of nine DOF, including the mobile platform joints P_1, P_2, and P_3 and the manipulator joints M_1 to M_6. The Universal Robot UR5 manipulator is often used in medium-sized companies, as well as in research and development. Its parameters, including link inertias, can be found in [65,66]. Figure 14 shows the results using m_v based on linear velocities. Figure 14a shows that a mobile base placement at the proposed point (the center of the maximal possible inscribed circle) offers the advantage of staying within a zone of minimum 30% manipulability, even when the platform is moved. The maximum value of the manipulability is located outside the comfort zone. Hence, using this position, already, a small movement of the mobile platform would lead to a manipulability lower than 30% of the maximal value, and the same is also valid for Figure 14b.

Figure 15 shows the results using m_v based on angular velocities. The mobile base placement at the proposed point (the center of the maximal possible inscribed circle) achieves a large zone of minimum 30% manipulability. In Figure 15b, the maximum value of manipulability is located near the middle of the comfort zone. Here, a small area is excluded from the comfort zone, so a base movement into this area could lead to a failure of the task—as the inverse kinematics could not find a solution for this example. The center points of all inscribed circles would be feasible mobile-platform goal positions with higher flexibility in base movement and ensure a solution for the inverse kinematics. A selection between them could be made via a comparison of the individual manipulability of each point, but for all points here, the manipulability is at least higher than or equal to 30% of the maximum value. Figure 16 shows the mobile manipulator Kairos within a Python simulation. There, Exudyn (a C++ based Python framework for flexible, multibody systems' simulation) was used to simulate the mobile platform, as well as the manipulator. The mobile platform and the manipulator links are represented as rigid bodies, and the

Mecanum wheels are modeled with a contact-friction model based on the rolling disk, as described in [4]. In Figure 16a, the task starts, and the comfort zone (possible mobile-base goal placements) is drawn using red boxes. The height of each box corresponds to the manipulability measure used, m_v. Figure 16b shows the mobile manipulator within the pick task when the mobile base is already in its goal position.
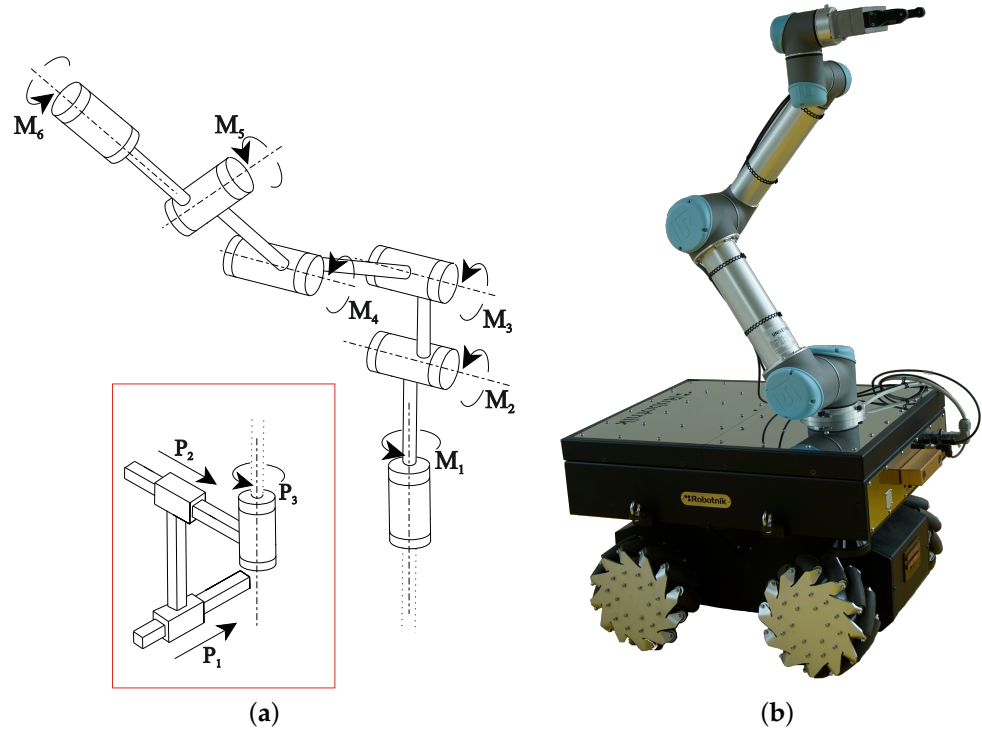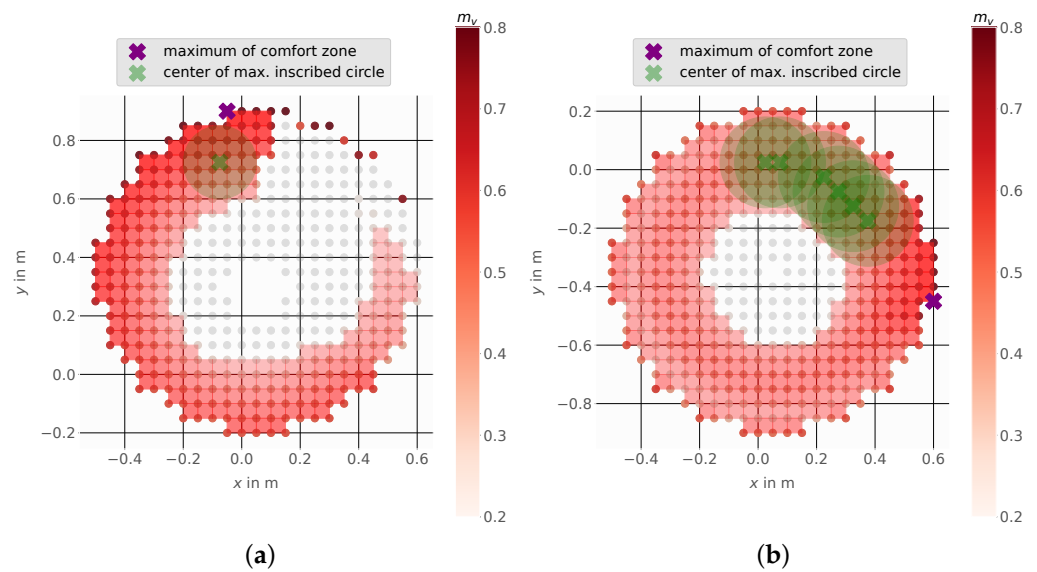

Figure 13. Mobile manipulator Kairos developed and built by the company Robotnik. The mobile manipulator combines a Universal Robot UR5 manipulator with six rotational joints mounted on a holonomic mobile base using Mecanum wheels. (**a**) Kinematic chain of the mobile manipulator Kairos; (**b**) mobile manipulator Kairos.

Figure 14. Comfort zone for the mobile manipulator Kairos using the linear velocity **v**-dependent manipulability measure m_v with a minimum of 30% manipulability for the EE pose (**a**) ($\zeta_{EE,des} = [0.05, 0.35, 1.1, \frac{\pi}{4}, 0, \frac{-\pi}{4}]$) and EE position (**b**) ($\zeta_{EE,des} = [0.05, -0.35, 1.1]$).
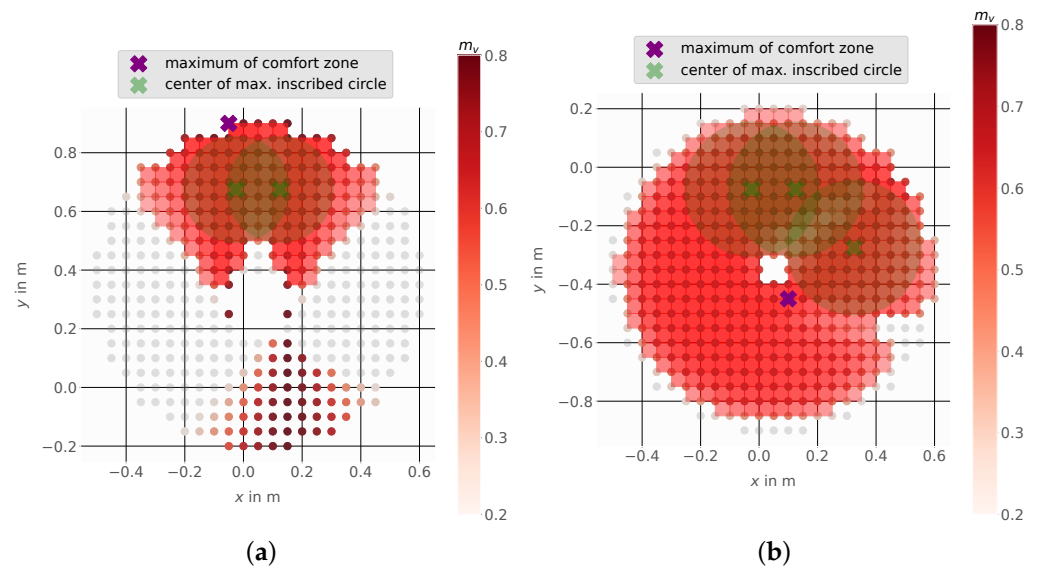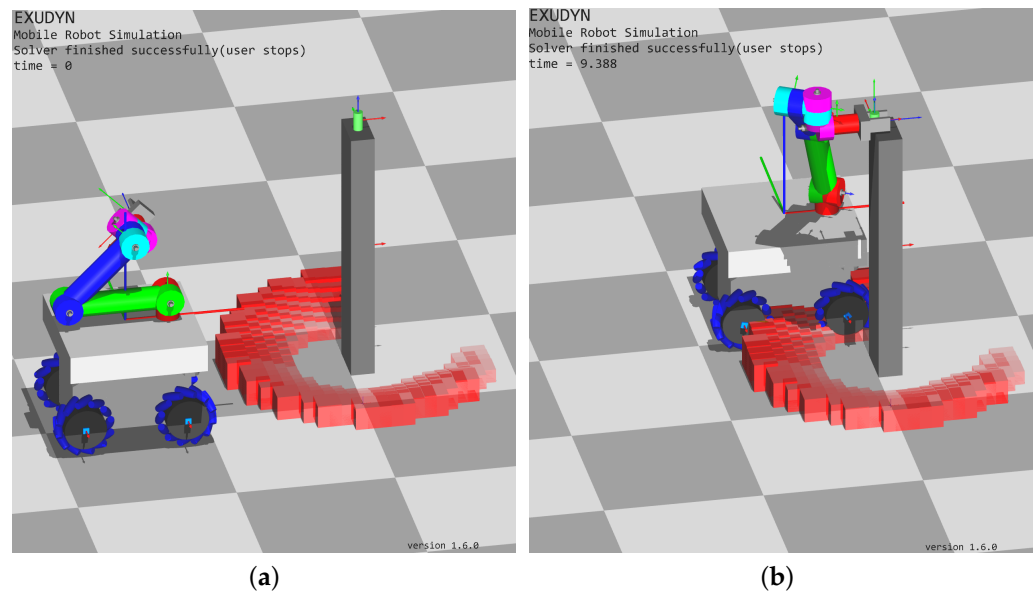
Figure 15. Comfort zone for the mobile manipulator Kairos using the angular velocity ω-dependent manipulability measure m_v with a minimum of 30% manipulability for the EE pose (**a**)($\zeta_{EE,des} = [0.05,\ 0.35,\ 1.1,\ \frac{\pi}{4},\ 0,\ \frac{-\pi}{4}]$) and EE position (**b**) ($\zeta_{EE,des} = [0.05,\ -0.35,\ 1.1]$).

Figure 16. Mobile manipulator Kairos approaching a target object using the task-dependent comfort zone (red boxes) for mobile manipulator base placement. Values as shown in Figure 14a: (**a**) mobile manipulator Kairos in the start position; (**b**) mobile manipulator Kairos in the pick-object position.

5. Conclusions

In this work, we have presented a method to determine a zone of possible base positions, the so-called *task-dependent comfort zone*, for mobile manipulators based on different manipulability measures. Our method takes into account the specific demands of different tasks, such as the need for high-speed movements, large forces, high accelerations, or heightened stiffness. To find a suitable goal position for a mobile manipulator, we used the discretization of the manipulator workspace and different image-processing methods.

We evaluated our method using Python computation with different holonomic mobile manipulators. First, we used the Stanford Robotics Platform with a mounted Puma560 (6DOF, spherical wrist). Secondly, we used the mobile manipulator LeoBot with a mounted

Franka Emika Panda (7DOF), and lastly, we used the mobile manipulator Kairos with a mounted Universal Robot UR5 (6DOF).

Our results show that manipulability measures could be used to develop a task-dependent comfort zone for mobile manipulators, providing more specific guidelines for selecting appropriate mobile-manipulator base placement strategies. In path-planning algorithms, a fixed goal position for the mobile base is usually given, although in many cases, the exact positioning of the platform is not as important as the goal of the manipulators' end-effector.

Once the *task-dependent comfort zone* is calculated, it can help find feasible mobile-manipulator base positions in dynamically changing environments. Our study highlights the importance of considering manipulability measures when determining mobile-manipulator base positions, and it shows the potential of our proposed method for use in practical applications. Future work could explore the use of our method within real-world scenarios to further evaluate its effectiveness. Therefore, combining the concept of the task-dependent comfort zone with a path planning algorithm to use it within the robot framework ROS should be considered, especially a combination with the `costmap2D` algorithm [67] used in the ROS-Navigation stack. With the help of this algorithm, the proposed task-dependent comfort zone could be included as an additional cost map layer in the ROS-Navigation stack [68].

References

1. Brandstötter, M.; Mirkovic, D.; Hofbaur, M. Mobile Manipulation—Eine altbekannte Technologie findet durch sensitive Robotertechnologie Einzug in die Industrie. In Proceedings of the C-AR2017—Conference on Automation and Robotics, Linz, Austria, 16–17 May 2017; pp. 1–6.
2. Sereinig, M.; Werth, W.; Faller, L.M. A review of the challenges in mobile manipulation: Systems design and RoboCup challenges. *Elektrotech. Informationstechnik* **2020**, *137*, 297–308. [CrossRef]
3. Haviland, J.; Sünderhauf, N.; Corke, P. A Holistic Approach to Reactive Mobile Manipulation. *IEEE Robot. Autom. Lett.* **2022**, *7*, 3122–3129. [CrossRef]
4. Manzl, P.; Gerstmayr, J. An Improved Dynamic Model of the Mecanum Wheel for Multibody Simulations. In Proceedings of the Volume 9: 17th International Conference on Multibody Systems, Nonlinear Dynamics, and Control (MSNDC), International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Boston, MA, USA, 17–20 August 2021. [CrossRef]
5. Luca, A.D.; Oriolo, G.; Giordano, P.R. Kinematic modeling and redundancy resolution for nonholonomic mobile manipulators. In Proceedings of the 2006 IEEE International Conference on Robotics and Automation, ICRA 2006, Orlando, FL, USA, 15–19 May 2006; pp. 1867–1873. [CrossRef]
6. Chiaverini, S. Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators. *IEEE Trans. Robot. Autom.* **1997**, *13*, 398–410. [CrossRef]
7. Siciliano, B.; Khatib, O. *Springer Handbook of Robotics*; Springer Handbooks; Springer International Publishing: Berlin/Heideilberg, Germany, 2016.
8. Zanchettin, A.M.; Rocco, P.; Bascetta, L.; Symeonidis, I.; Peldschus, S. Kinematic motion analysis of the human arm during a manipulation task. In Proceedings of the ISR 2010 (41st International Symposium on Robotics) and ROBOTIK 2010 (6th German Conference on Robotics), Munich, Germany, 7–9 June 2010; Volume 2, pp. 1–6.

9. *ISO15066*; Robots and Robotic Devices—Collaborative Robots. International Organization for Standardization. **Edition 1, 2016**. Available online: https://www.iso.org/obp/ui/en/#iso:std:iso:ts:15066:ed-1:v1:en (accessed on 3 August 2024).

10. *ISO10218*; Robots and Robotic Devices — Safety Requirements for Industrial Robots — Part 1: Robots. International Organization for Standardization. **Edition 2, 2011**. Available online: https://www.iso.org/obp/ui/en/#iso:std:iso:10218:-1:ed-2:v1:en (accessed on 3 August 2024).

11. Rastegarpanah, A.; Gonzalez, H.C.; Stolkin, R. Semi-Autonomous Behaviour Tree-Based Framework for Sorting Electric Vehicle Batteries Components. *Robotics* **2021**, *10*, 82. [CrossRef]

12. Saoji, S.; Rosell, J. Flexibly configuring task and motion planning problems for mobile manipulators. In Proceedings of the 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Vienna, Austria, 8–11 September 2020; Volume 1, pp. 1285–1288. [CrossRef]

13. Castaman, N.; Pagello, E.; Menegatti, E.; Pretto, A. Receding Horizon Task and Motion Planning in Changing Environments. *Robot. Auton. Syst.* **2021**, *145*, 103863. [CrossRef]

14. Brock, O.; Kavraki, L. Decomposition-based motion planning: A framework for real-time motion planning in high-dimensional configuration spaces. In Proceedings of the 2001 ICRA IEEE International Conference on Robotics and Automation (Cat. No.01CH37164), Seoul, Republic of Korea, 21–26 May 2001; Volume 2, pp. 1469–1474. . [CrossRef]

15. Su, J.; Xie, W. Motion Planning and Coordination for Robot Systems Based on Representation Space. *IEEE Trans. Syst. Man Cybern. Part B* **2011**, *41*, 248–259. [CrossRef]

16. LaValle, S.M.; Kuffner, J.J. Randomized Kinodynamic Planning. *Int. J. Robot. Res.* **2001**, *41*, 378–400. [CrossRef]

17. Kavraki, L.; Svestka, P.; Latombe, J.C.; Overmars, M. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Trans. Robot. Autom.* **1996**, *12*, 566–580. [CrossRef]

18. Hsu, D.; Latombe, J.C.; Motwani, R. Path planning in expansive configuration spaces. In Proceedings of the International Conference on Robotics and Automation, Albuquerque, NM, USA, 20–25 April 1997; Volume 3, pp. 2719–2726. [CrossRef]

19. Ratliff, N.; Zucker, M.; Bagnell, J.A.; Srinivasa, S. CHOMP: Gradient optimization techniques for efficient motion planning. In Proceedings of the 2009 IEEE International Conference on Robotics and Automation, Kobe, Japan, 12–17 May 2009; pp. 489–494. [CrossRef]

20. Kalakrishnan, M.; Chitta, S.; Theodorou, E.; Pastor, P.; Schaal, S. STOMP: Stochastic trajectory optimization for motion planning. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 4569–4574. [CrossRef]

21. Coleman, D.; Sucan, I.; Chitta, S.; Correll, N. Reducing the Barrier to Entry of Complex Robotic Software: A MoveIt! Case Study. *arXiv* **2014**. [CrossRef]

22. Sandakalum, T.; Ang, M.H. Motion Planning for Mobile Manipulators: A Systematic Review. *Machines* **2022**, *10*, 97. [CrossRef]

23. Makhal, A.; Goins, A.K. Reuleaux: Robot Base Placement by Reachability Analysis. In Proceedings of the 2018 Second IEEE International Conference on Robotic Computing (IRC), Laguna Hills, CA, USA, 13 January–1 February 2018; pp. 137–142.

24. Zhang, H.; Sheng, Q.; Sun, Y.; Sheng, X.; Xiong, Z.; Zhu, X. A novel coordinated motion planner based on capability map for autonomous mobile manipulator. *Robot. Auton. Syst.* **2020**, *129*, 103554. [CrossRef]

25. Zhang, H.; Sheng, Q.; Hu, J.; Sheng, X.; Xiong, Z.; Zhu, X. Cooperative Transportation with Mobile Manipulator: A Capability Map-Based Framework for Physical Human–Robot Collaboration. *IEEE/ASME Trans. Mechatron.* **2022**, *27*, 4396–4405. [CrossRef]

26. Yoshikawa, T. Manipulability of Robotic Mechanisms. *Int. J. Robot. Res.* **1985**, *4*, 3–9. [CrossRef]

27. Zacharias, F.; Borst, C.; Hirzinger, G. Capturing robot workspace structure: Representing robot capabilities. In Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, CA, USA, 29 October–2 November 2007; pp. 3229–3236. [CrossRef]

28. Sereinig, M.; Manzl, P.; Gerstmayr, J. Komfortzone mobiler Manipulatoren. In Proceedings of the Sechste IFToMM D-A-CH Konferenz 2020, Campus Technik Lienz, Lienz, Austria, 27–28 February 2020; Volume 2020. [CrossRef]

29. Spong, M.; Hutchinson, S.; Vidyasagar, M. *Robot Modeling and Control*; Wiley: Hoboken, NJ, USA, 2005.

30. Lynch, K.M.; Park, F.C. *Modern Robotics: Mechanics, Planning, and Control*, 1st ed.; Cambridge University Press: Cambridge, MA, USA, 2017.

31. Corke, P. *Robotics, Vision and Control: Fundamental Algorithms in MATLAB*, 1st ed.; Springer Publishing Company, Inc.: Berlin/Heidelberg, Germany, 2013.

32. Kim, J.-O.; Khosla, K. Dexterity measures for design and control of manipulators. In Proceedings of the IROS '91: IEEE/RSJ International Workshop on Intelligent Robots and Systems, Osaka, Japan, 3–5 November 1991; Volume 2, pp. 758–763.

33. Melchiorri, C. Force manipulability ellipsoids for general manipulation systems. *IFAC Proc. Vol.* **1994**, *27*, 235–240. [CrossRef]

34. Ajoudani, A.; Tsagarakis, N.G.; Bicchi, A. Choosing Poses for Force and Stiffness Control. *IEEE Trans. Robot.* **2017**, *33*, 1483–1490. [CrossRef]

35. Chen, C.; Peng, F.; Yan, R.; Li, Y.; Wei, D.; Fan, Z.; Tang, X.; Zhu, Z. Stiffness performance index based posture and feed orientation optimization in robotic milling process. *Robot. Comput.-Integr. Manuf.* **2019**, *55*, 29–40. [CrossRef]

36. Kövecses, J.; Angeles, J. The stiffness matrix in elastically articulated rigid-body systems. *Multibody Syst. Dyn.* **2007**, *18*, 169–184. [CrossRef]

37. Dumas, C.; Caro, S.; Cherif, M.; Garnier, S.; Furet, B. Joint stiffness identification of industrial serial robots. *Robotica* **2012**, *30*, 649–659. [CrossRef]

38. Busson, D.; Bearee, R.; Olabi, A. Task-oriented rigidity optimization for 7 DOF redundant manipulators. *IFAC-PapersOnLine* **2017**, *50*, 14588–14593. [CrossRef]
39. Salisbury, J.K. Active stiffness control of a manipulator in cartesian coordinates. In Proceedings of the 19th IEEE Conference on Decision and Control including the Symposium on Adaptive Processes, Albuquerqe, NM, USA, 10–12 December 1980; pp. 95–100.
40. Guo, Y.; Dong, H.; Ke, Y. Stiffness-oriented posture optimization in robotic machining applications. *Robot. Comput.-Integr. Manuf.* **2015**, *35*, 69–76. [CrossRef]
41. Yoshikawa, T. Dynamic Manipulability of Robot Manipulators. *Trans. Soc. Instrum. Control. Eng.* **1985**, *21*, 970–975. [CrossRef]
42. Chiacchio, P.; Concilio, M. The dynamic manipulability ellipsoid for redundant manipulators. *IEEE Int. Conf. Robot. Autom.* **1998**, *1*, 95–100. [CrossRef]
43. El Makrini, I.; Merckaert, K.; De Winter, J.; Lefeber, D.; Vanderborght, B. Task allocation for improved ergonomics in Human-Robot Collaborative Assembly. *Interact. Stud.* **2019**, *20*, 103–134. [CrossRef]
44. Burdet, E.; Osu, R.; Franklin, D.W.; Milner, T.E.; Kawato, M. The central nervous system stabilizes unstable dynamics by learning optimal impedance. *Nature* **2001**, *414*, 446–449. [CrossRef]
45. Khatib, O. A unified approach for motion and force control of robot manipulators: The operational space formulation. *IEEE J. Robot. Autom.* **1987**, *3*, 43–53. [CrossRef]
46. Chen, H.; Lee, S.I.; Do, J.H.; Lee, J.M. Directional Manipulability to Improve Performance Index of Dual Arm Manipulator for Object Grasping. In *Intelligent Autonomous Systems 12: Volume 1 Proceedings of the 12th International Conference IAS-12, Jeju Island, Republic of Korea, 26–29 June 2012*; Lee, S., Cho, H., Yoon, K.J., Lee, J., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 595–602. [CrossRef]
47. Kraetzschmar, G.K.; Hochgeschwender, N.; Nowak, W.; Hegger, F.; Schneider, S.;Dwiputra, R.; Berghofer, J.; Bischoff, R. RoboCup@Work: Competing for the Factory of the Future. In *RoboCup 2014: Robot World Cup XVIII*; Springer International Publishing: Berlin/Heidelberg, Germany, 2015; pp. 171–182. [CrossRef]
48. Abdel-Malek, K.; Yang, J. Workspace boundaries of serial manipulators using manifold stratification. *Int. J. Adv. Manuf. Technol.* **2006**, *28*, 1211–1229. [CrossRef]
49. Cao, Y.; Lu, K.; Li, X.; Zang, Y. Accurate Numerical Methods for Computing 2D and 3D Robot Workspace. *Int. J. Adv. Robot. Syst.* **2011**, *8*, 76. [CrossRef]
50. Patel, S.; Sobh, T. Task based synthesis of serial manipulators. *J. Adv. Res.* **2015**, *6*, 479–492. [CrossRef]
51. Van der Walt, S.; Schönberger, J.L.; Nunez-Iglesias, J.; Boulogne, F.; Warner, J.D.; Yager, N.; Gouillart, E.; Yu, T.; The Scikit-Image Contributors. scikit-image: Image processing in Python. *PeerJ* **2014**, *2*, e453. [CrossRef]
52. Fiorio, C.; Gustedt, J. Two linear time Union-Find strategies for image processing. *Theor. Comput. Sci.* **1996**, *154*, 165–181. [CrossRef]
53. Jähne, B. *Digital Image Processing*, 6th ed.; Springer: Berlin/Heidelberg, Germany, 2005. [CrossRef]
54. Felzenszwalb, P.; Huttenlocher, D. Distance Transforms of Sampled Functions. *Theory Comput.* **2004**, *8*, 415–428. [CrossRef]
55. Doty, K.L.; Melchiorri, C.; Schwartz, E.M.; Bonivento, C. Robot manipulability. *IEEE Trans. Robot. Autom.* **1995**, *11*, 462–468. [CrossRef]
56. Gerstmayr, J. Exudyn—A C++ based Python package for flexible multibody systems. *Multibody Syst. Dyn.* **2023**, *60*, 533–561. [CrossRef]
57. Sugihara, T. Solvability-Unconcerned Inverse Kinematics by the Levenberg–Marquardt Method. *IEEE Trans. Robot.* **2011**, *27*, 984–991. [CrossRef]
58. Corke, P.; Haviland, J. Not your grandmother's toolbox—The Robotics Toolbox reinvented for Python. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 11357–11363.
59. Holmberg, R.; Khatib, O. Development and Control of a Holonomic Mobile Robot for Mobile Manipulation Tasks. *Int. J. Robot. Res.* **2000**, *19*, 1066–1074. [CrossRef]
60. Chang, K.S.; Holmberg, R.; Khatib, O. The augmented object model: Cooperative manipulation and parallel mechanism dynamics. In Proceedings of the ICRA, IEEE International Conference on Robotics and Automation, San Francisco, CA, USA, 24–28 April 2000; Volume 1, pp. 470–475. [CrossRef]
61. Corke, P.I. *Visual Control of Robots: High-Performance Visual Serving*; John Wiley & Sons Inc.: Hoboken, NJ, USA, 1997.
62. Kim, H.; Streit, D. Configuration dependent stiffness of the PUMA 560 manipulator: Analytical and experimental results. *Mech. Mach. Theory* **1995**, *30*, 1269–1277. [CrossRef]
63. Sereinig, M.; Manzl, P.; Hofmann, P.; Neurauter, R.; Pieber, M.; Gerstmayr, J. Omnidirectional Mobile Manipulator LeoBot for Industrial Environments, Developed for Research and Teaching. In Proceedings of the RoboCup 2022; Eguchi, A., Lau, N., Paetzel-Prüsmann, M., Wanichanon, T., Eds.; Springer International Publishing: Berlin/Heidelberg, Germany, 2023; pp. 127–139. [CrossRef]
64. Gaz, C.; Cognetti, M.; Oliva, A.; Robuffo Giordano, P.; De Luca, A. Dynamic Identification of the Franka Emika Panda Robot with Retrieval of Feasible Parameters Using Penalty-Based Optimization. *IEEE Robot. Autom. Lett.* **2019**, *4*, 4147–4154. [CrossRef]
65. Kebria, P.M.; Al-wais, S.; Abdi, H.; Nahavandi, S. Kinematic and dynamic modelling of UR5 manipulator. In Proceedings of the 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Budapest, Hungary, 9–12 October 2016; pp. 4229–4234.

66. Kovincic, N.; Müller, A.; Gattringer, H.; Weyrer, M.; Schlotzhauer, A.; Brandstötter, M. Dynamic parameter identification of the Universal Robots UR5. In Proceedings of the ARW & OAGM Workshop 2019, Graz, Austria, 9–10 May 2019; pp. 44–53. [CrossRef]
67. Lu, D.V.; Hershberger, D.; Smart, W.D. Layered costmaps for context-sensitive navigation. In Proceedings of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, 14–18 September 2014; pp. 709–715. [CrossRef]
68. Marder-Eppstein, E.; Berger, E.; Foote, T.; Gerkey, B.; Konolige, K. The Office Marathon: Robust navigation in an indoor office environment. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation, Anchorage, AL, USA, 3–8 May 2010; pp. 300–307. [CrossRef]