



Article

KUKA[®]–Kinova[®] Holonomic Mobile Manipulator

Vasil Popov , Andon V. Topalov *, Tihomir Stoyanov and Sevil Ahmed-Shieva 

Faculty of Electronics and Automation, Technical University—Sofia, Plovdiv Branch, 4000 Plovdiv, Bulgaria; vasil_popov@tu-plovdiv.bg (V.P.); stoyanov.tihomir@tu-plovdiv.bg (T.S.); sevil.ahmed@tu-plovdiv.bg (S.A.-S.)
* Correspondence: topalov@tu-plovdiv.bg; Tel.: +359-882-389-026

Abstract: We have proposed an open-source holonomic mobile manipulator composed of the KUKA youBot holonomic mobile platform with four Swedish wheels and a stationary aboard six-degrees-of-freedom Kinova Jaco Gen 2H manipulator, and we have developed corresponding kinematic problems. We have defined forward and inverse analytic Jacobians and designed Jacobian algorithms of forward and inverse mobile manipulator kinematics. An experimental test conducted with the designed laboratory prototype of the investigated mobile manipulator with the described kinematics was used to verify the obtained theoretical results. The goal of the test was to keep constant the position of the gripper in 3D space while the mobile platform is moving to some extent in the 2D workspace.

Keywords: forward kinematics; inverse kinematics; holonomic mobile manipulator; Denavit–Hartenberg; Pieper’s algorithm; Jacobian algorithm; KUKA youBot; Kinova Jaco



Citation: Popov, V.; Topalov, A.V.; Stoyanov, T.; Ahmed-Shieva, S. Kinematic Modeling with Experimental Validation of a KUKA[®]–Kinova[®] Holonomic Mobile Manipulator. *Electronics* **2024**, *13*, 1534. <https://doi.org/10.3390/electronics13081534>

Academic Editor: Luca Patanè

Received: 15 January 2024

Revised: 3 April 2024

Accepted: 11 April 2024

Published: 17 April 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In the present day, robots with various configurations (mobile robots, robot manipulators, mobile manipulators, etc.) are very rapidly advancing in performing different application tasks in our everyday lives. There are also many companies that offer on the market service robots dedicated to helping disabled people in their daily activities. Currently, various approaches for human–robot interactions are gaining popularity, including, for example, manual guidance, among others. It should be noted that such machines, especially in their recently investigated collaborative implementation, must correspond to the safety standards developed in the EU [1–3]. The widespread use of robots also requires a high level of understanding of the problems involved into this field of science [4–8]. In the present work, we describe existing research into the mathematical modeling of the kinematics of a holonomic mobile manipulator that has remarkable mobility and handling capabilities. The combined machinery contains two devices: a mobile platform and a 6 DoF manipulator. The chosen mobile platform, KUKA youBot, has four omnidirectional wheels, and it is widely used for scientific research tasks [9]. Installed on the top of the mobile platform robot manipulator, Kinova Jaco Gen2 is often used to assist disabled people by helping them with their daily activities, such as drinking water, holding a spoon, etc. The manipulator has also been selected for the experiments because it is supported by numerous software tools that make its control open to users and provide an opportunity to explore different control approaches. Such a combination of mobile platforms and manipulators widens the fields of application of the resulting mobile manipulators. Examples of similar assemblies can be found in [10–12]. All those efforts aimed to extend the working space of manipulators. However, the assembly of a mobile platform and a manipulator requires relevant kinematic models. An interesting solution can be found in [13], where authors present a configuration of a mobile manipulator based on Boston Dynamics Spot that is additionally equipped with a robotic arm. Some of the mobile manipulators with

open-chain kinematics have been studied and are presented in [14–20]. More recent studies on the inverse kinematics frameworks of mobile manipulators are given in [21–24].

In the present work, we derive the forward kinematics, inverse kinematics, Jacobian and inverse Jacobian of the investigated mobile manipulator. The obtained mathematical models are then applied to control the movement of the laboratory prototype mobile manipulator in the task of maintaining a constant position and orientation of the manipulator's end-effector in 3D space while the mobile platform is moving to some extent in the 2D workspace. This is achieved by controlling the speed of each mobile manipulator link and the speed of each mobile platform wheel. Afterwards, the efficacy of the proposed approach is evaluated by running a real-world experiment.

The article is structured in the following way: In Section 2, we describe the components of the mobile manipulator and derive their kinematic models. Then, the forward and inverse kinematics of the mobile manipulator and Jacobian algorithms are derived. In Section 3, the achieved results from the conducted experimental test carried out with the constructed laboratory prototype are presented. Section 4 contains conclusions.

2. Devices and Mathematical Models

2.1. The Mobile Manipulator

The assembled structure consists of two essential devices: a holonomic mobile platform and a six-degrees-of-freedom (6 DoF) robot manipulator (Figure 1). The mobile platform is a KUKA youBot holonomic platform, and the implemented spherical wrist robot is a Jaco Gen2 manipulator produced by the Canadian company Kinova.

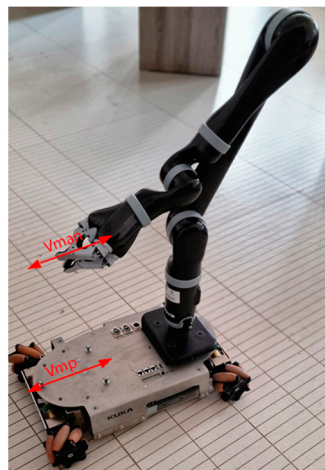


Figure 1. Mobile manipulator and velocities.

This paper discusses the mathematical derivation of many important aspects that are related to the designed mobile manipulator: the forward and inverse kinematics of the mobile platform, forward and inverse kinematics of the manipulator Jaco Gen2, combined homogeneous transformation between the mobile platform and the end-effector, manipulator's Jacobian matrix, manipulator's inverse Jacobian matrix, and synchronization between the velocities of these two individual components. The obtained theoretical results have been verified via the conducted experimental test using a laboratory prototype of the above mobile manipulator. The implemented control software of the prototype uses ROS Melodic (Robot Operating System)-implemented packages, which were either provided by the manufacturers of the devices or are freely available from the web. The transfer of the control signals and feedback values is provided by the ROS's topics, the ROS's actions, and the ROS's services. Each one of these types of communication is used depending on the existing situation. The presented results show the velocities of the mobile platform V_{mp}

and the robot manipulator V_{man} . These velocities allow for achieving a stable positioning of the end-effector during a movement of the mobile platform. The resulting velocity is V_{res} .

$$V_{res} = \pm V_{mp} \pm V_{man} \quad (1)$$

2.2. The Mobile Platform KUKA youBot

The KUKA youBot mobile platform gives us the opportunity to expand the working space of the manipulator. Thus, the proposed structure of mobile manipulator enhances its functionalities.

However, in the design, this combination also increases the complexity of the whole robot system structure while increasing the mobile robot's own weight and energy consumption. In order to complete complex work, the manipulator and omnidirectional vehicle must be in perfect harmony, and they also need to reduce energy loss. Therefore, the study of omnidirectional mobile operating robot has important theoretical and practical value.

The next figure (Figure 2) shows the most significant dimensions of the KUKA youBot mobile platform and its structure.

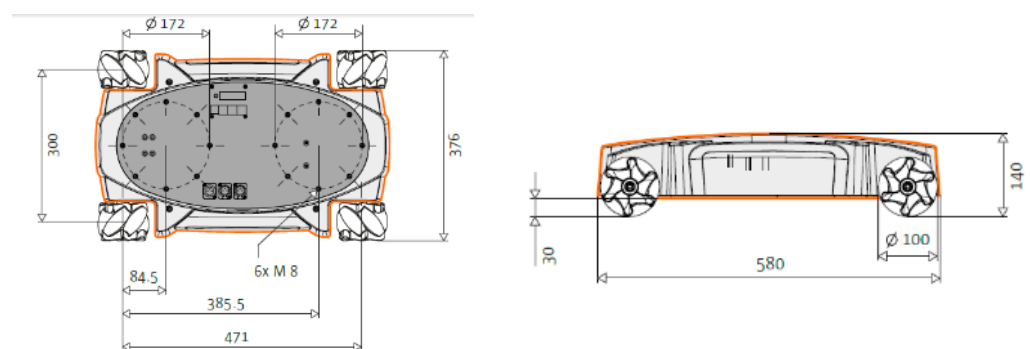


Figure 2. Geometric dimensions (in [mm]) of the mobile platform KUKA youBot.

The mobile platform is controlled by an onboard computer with embedded Intel Atom dual-core CPU and 2 GB RAM, 32 GB SSD, WLAN and a USB interface. Available interfaces are USB and EtherCAT/Ethernet. The existing WiFi connection makes possible the establishment of ROS communication.

The conventional approach for description of the motion of the mobile platform in a 2D plane requires the definition of coordinate frames. To specify the position and the orientation of the mobile platform onto the plane, we have to establish a relationship between the reference frame of the plane and the local frame of the platform. The axes X_I and Y_I define an arbitrary inertial basis on the plane as the global reference frame from some origin $O : \{X_I, Y_I\}$. To indicate the position and orientation of the mobile platform, we choose a geometric center of the platform chassis-point P , and, thus, it is the origin of the platform's local reference frame. The axes $\{X_R, Y_R\}$ define the mobile platform's local reference frame. The column vector (2) specifies the position and the orientation of the platform local reference frame in the global reference frame [25,26]:

$$P = \begin{bmatrix} x \\ y \\ \mu \end{bmatrix} \quad (2)$$

where x and y are the displacements along the X_I axis and Y_I axis, and the orientation is defined by μ .

The KUKA youBot mobile robotic platform is controlled by varying the rotational speed of each of its omnidirectional wheels (Figure 3b).

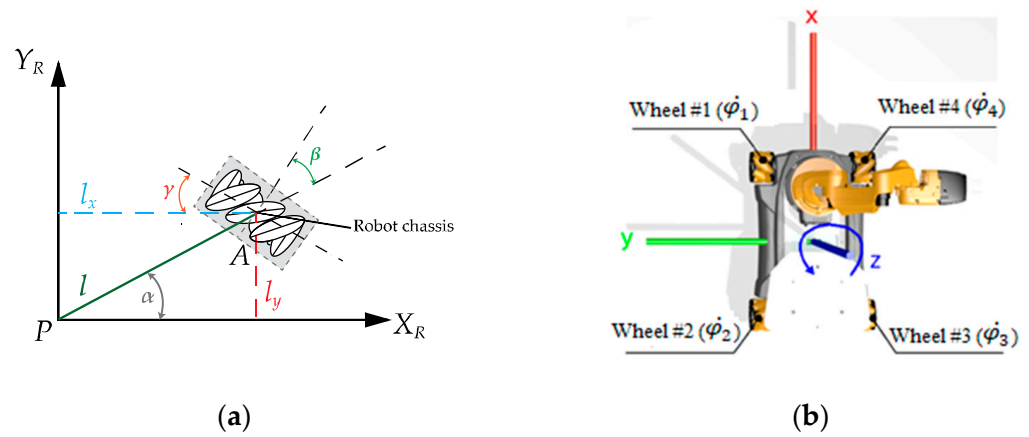


Figure 3. A sketch with the necessary notations to calculate the forward kinematics of the mobile platform: (a) wheel parameters and (b) wheels and the corresponding speeds.

The kinematic model describes the relationship between the rotational speeds of the four wheels $\dot{\varphi}_1, \dot{\varphi}_2, \dot{\varphi}_3$ and $\dot{\varphi}_4$ and the speed of the mobile platform in the 2D plane. The types of wheels and robot chassis geometry define this relationship. Each wheel introduces a set of mathematical constraints on the movement of the platform.

According to [25], using the constraints, we can obtain the kinematic model of the mobile robot. For a Swedish (mecanum) wheel, the following equations must be satisfied [25,26]:

$$\begin{aligned} V_1 R(\mu) \dot{P} - r \dot{\varphi}_i \cos(\gamma_i) &= 0 \\ V_2 R(\mu) \dot{P} - r \dot{\varphi}_i \cos(\gamma_i) - r_{sw} \dot{\varphi}_{sw} &= 0, \end{aligned} \tag{3}$$

where V_1 and V_2 denote the following three-element vectors:

$$V_1 = [\sin(\alpha_i + \beta_i + \gamma_i) \quad -\cos(\alpha_i + \beta_i + \gamma_i) \quad -l \cos(\beta_i + \gamma_i)]$$

$$V_2 = [\cos(\alpha_i + \beta_i + \gamma_i) \quad \sin(\alpha_i + \beta_i + \gamma_i) \quad -l \sin(\beta_i + \gamma_i)]$$

the matrix R is the rotational matrix, and r denotes the radius of the wheels.

$$R = \begin{bmatrix} \cos \mu & \sin \mu & 0 \\ -\sin \mu & \cos \mu & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The index i denotes the number of platform wheels.

Considering the construction parameters of the platform: the length $l_x = 235.5$ mm, $l_y = 150$ mm and the angle $\gamma = \pm 45^\circ$, (the robot dimensions are shown in (Figure 2) and the notations in (Figure 3a)), the angles α_i and β_i can be calculated according to the following equations [26]:

$$\begin{aligned} \alpha_1 &= \arctg\left(\frac{l_y}{l_x}\right), \alpha_2 = 180^\circ - \arctg\left(\frac{l_y}{l_x}\right) \\ \alpha_3 &= 180^\circ + \arctg\left(\frac{l_y}{l_x}\right), \alpha_4 = \arctg\left(\frac{-l_y}{l_x}\right) \end{aligned} \tag{4}$$

where, for the considered platform, $l_x = 235.5$ mm and $l_y = 150$ mm.

According to the theory of kinematics of the Swedish wheel [25], and considering the fact that the axes of the four wheels are perpendicular to PX_R , then using the notations in Figure 3a, it is possible to express the angle β_i and the length l :

$$\beta_i = 90 - \alpha_i; \tag{5}$$

$$l = \sqrt{l_x^2 + l_y^2} \approx 0.279 \text{ m}$$

Table 1 contains the kinematic parameters of the mobile platform.

Table 1. Kinematic parameters of the mobile platform.

	α [deg]	β [deg]	γ [deg]	l [m]	r [m]
Wheel (1)	32.5	57.5	−45	0.279	0.05
Wheel (2)	147.5	−57.5	45	0.279	0.05
Wheel (3)	−147.5	−122.5	−45	0.279	0.05
Wheel (4)	−32.5	122.5	45	0.279	0.05

The forward kinematics of the mobile platform is described by the following matrix equation [26]:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\mu} \end{bmatrix} = \frac{r}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 \\ \frac{-1}{(l_x+l_y)} & \frac{-1}{(l_x+l_y)} & \frac{1}{(l_x+l_y)} & \frac{1}{(l_x+l_y)} \end{bmatrix} \begin{bmatrix} \dot{\varphi}_1 \\ \dot{\varphi}_2 \\ \dot{\varphi}_3 \\ \dot{\varphi}_4 \end{bmatrix} \quad (6)$$

And the inverse kinematic model is

$$\begin{bmatrix} \dot{\varphi}_1 \\ \dot{\varphi}_2 \\ \dot{\varphi}_3 \\ \dot{\varphi}_4 \end{bmatrix} = \frac{1}{r} \begin{bmatrix} 1 & -1 & -(l_x + l_y) \\ 1 & 1 & -(l_x + l_y) \\ 1 & -1 & (l_x + l_y) \\ 1 & 1 & (l_x + l_y) \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\mu} \end{bmatrix} \quad (7)$$

2.3. The Manipulator KINOVA Jaco Gen 2

An overview of the robot manipulator Jaco Gen2 and its dimensions are shown on (Figure 4) [27].

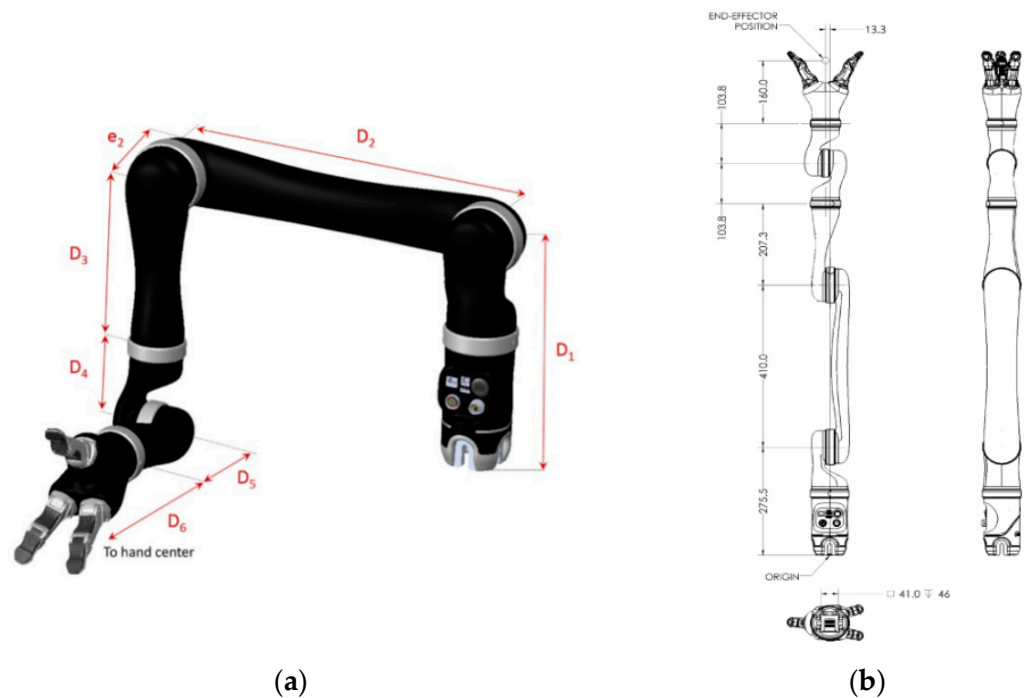


Figure 4. Notations (a) and dimensions (b) of Jaco Gen 2 manipulator.

The geometric dimensions are shown in Table 2; these values are essential components of forward kinematics and notations: $D_1, D_2, D_3, D_4, D_5, D_6$ and e_2 are structural elements of the homogeneous transformations.

Table 2. Structural elements of the homogeneous transformations [27].

Parameters	Descriptions	Length (m)
D ₁	Base to shoulder	0.2755
D ₂	Upper Length (shoulder to elbow)	0.41
D ₃	Forearm length (elbow to wrist)	0.2073
D ₄	First wrist length	0.1038
D ₅	Second wrist length	0.1038
D ₆	Wrist to center of the hand	0.16
e ₂	Joint 3–4 lateral offset	0.0133

In Table 3, some physical characteristics of the manipulator Jaco Gen2 Spherical Wrist are given.

Table 3. Physical characteristics of the manipulator Jaco Gen2 [27].

Characteristics	Values
Total weight	4.4 kg
Reach	98.4 cm
Maximum payload	2.6 kg (mid-range)/2.2 kg (full-range)
Materials	carbon fiber/aluminum
Maximum linear arm speed	20 cm/sec
Power supply voltage	18–29 VDC
Average power	25 W (15 W standby)
Peak power	100 W
Communication protocol	RS485
Communication cables	20 pins flat flex cable
Water resistance	IPX2
Operating temperature	–10 to 40

Kinova Jaco Gen2 has specifically designed actuators mechanics and electronics. They use DC brushless motors with harmonic drive technology and are equipped with encoders, a torque sensor, a current sensor, a temperature sensor and accelerometers [23]. Kinova’s actuators and parameters are presented in Figure 5 and Table 4, respectively.

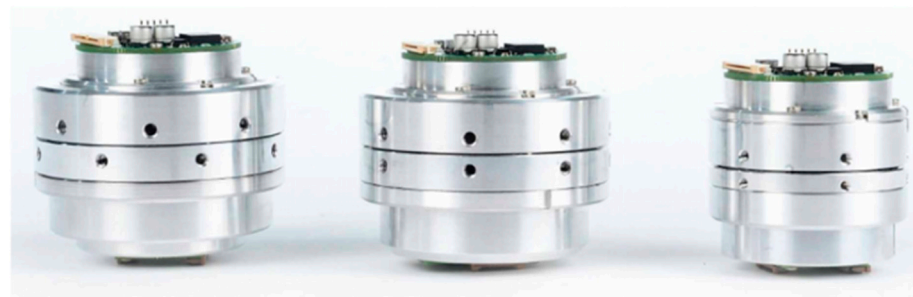


Figure 5. Kinova’s actuators: K-75+, K-75 and K-58.

Table 4. Kinova Jaco Gen 2’s actuator specifications.

Title	K–75+	K–75	K–58
Nominal torque (Nm)	12	9.2	3.6
Peak torque (Nm)	30.5	18	6.8
No load speed (rpm)	12.2	9.8	20.3
Nominal speed (rpm)	9.4	7	15
Weight (g)	570	587	357
Reduction ratio	136	160	110
Angular ranges, software limited (turns)		±27.7	
Communication protocol		RS485	

Multiple functionalities are offered by the software provided by the company Kinova (Boisbriand, QC, Canada). Using the Development Center and the Torque Console [27], researchers can monitor the manipulator’s state, activate admittance, switch between Cartesian and angular control and define trajectories. There is a C++ programming function integrated that can be used to access the API, for example, the API function can deactivate auto-avoidance behavior. Kinova Jaco 2 is an integrable device in ROS environment. The kinova-ros stack (set of ROS packages) provides a ROS interface for Kinova Jaco2. The stack is developed above the Kinova C++ API functions, which communicate with the DSP inside the robot base [27]. Using our knowledge of the robot’s structure and measurements, we define its forward kinematic table, using classical Denavit–Hartenberg convention (Table 5).

Table 5. Table formed using Denavit–Hartenberg parameters, classical approach [27].

i	α_i	a_i	d_i	θ_i
1	$\frac{\pi}{2}$	0	D_1	θ_1
2	π	D_2	0	θ_2
3	$\frac{\pi}{2}$	0	$-e_2$	θ_3
4	$\frac{\pi}{2}$	0	$-(D_3 + D_4)$	θ_4
5	$\frac{\pi}{2}$	0	0	θ_5
6	π	0	$-(D_5 + D_6)$	θ_6

The classical Denavit–Hartenberg convention is defined by the following matrix [28]:

$${}^i{}_{i-1}T = \begin{bmatrix} c\theta_i & -s\theta_i c\alpha_i & s\theta_i s\alpha_i & a_i c\theta_i \\ s\theta_i & c\theta_i c\alpha_i & -c\theta_i s\alpha_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{8}$$

The modified Denavit–Hartenberg convention is defined by the following matrix [29]:

$${}^i{}_{i-1}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1} d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{9}$$

We mark the trigonometric functions $c\theta_i$ and $s\theta_i$ with $\cos \theta_i$ and $\sin \theta_i$, respectively. Additionally, $c\theta_{ij}$ and $s\theta_{ij}$ are equal to $\cos(\theta_i + \theta_j)$ and $\sin(\theta_i + \theta_j)$, respectively.

The Manipulator Jaco Gen 2 Inverse Kinematics

All approaches to obtain the inverse kinematics of a robot manipulator proposed in the scientific literature can be divided into two classes: (i) approaches using numerical solutions and (ii) approaches based on closed-form solutions. Due to their iterative nature, approaches based on the numerical solutions are much slower than those using closed-form solution. So, in the most applications, especially for online implementations, the numerical approach for solving inverse kinematics is not suitable.

“Closed form” denotes a solution method based on analytic expressions or the solution of polynomials of degree 4 or less. Although, in the general case, the six-degrees-of-freedom manipulator does not have a closed–form solution, for some particular manipulator structures characterized by either several intersecting joint axes or several angles α_i equal to 0 or ± 90 degrees, closed-form solutions exist.

Such an approach to obtain a closed-form solution is Pieper’s approach that studies manipulators with six degrees of freedom where three of their consecutive axes intersect at a point [29]. The inverse kinematic problem calculates the desired angular position $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5$, and θ_6 of each rotational joint, according to the predefined position and orien-

tation in the final homogeneous transformation 0_6T . We have implemented Pieper’s method for all six revolute joints of Jaco Gen 2, with the last three axes intersecting. The idea in Pieper’s solution is to split the calculation into two separate problems—the first one, taking into account the first three joints, and the second one, considering the last three joints. A brief formulation of the algorithm can be presented as follows:

1. Locate the intersection point of the last three joint axes;
2. Calculate the position of this intersection point, given that we know the desired position and orientation of the end-effector;
3. Solve inverse kinematics for first three joints;
4. Compute 0_3T and determine 3_6T ;
5. Solve the inverse kinematics for the last three joints.

Using the modified Denavit–Hartenberg convention, given by (9) we can derive the homogeneous transformation (Table 6):

Table 6. Table formed by Denavit–Hartenberg parameters, using a modified approach.

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	$\frac{\pi}{2}$	0	0	θ_2
3	0	D_2	e_2	θ_3
4	$-\frac{\pi}{2}$	0	$D_3 + D_4$	θ_4
5	$\frac{\pi}{2}$	0	0	θ_5
6	$-\frac{\pi}{2}$	0	0	θ_6

Where $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5$ and θ_6 are the known angular displacements as follows:

$${}^0_1T = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{10}$$

$${}^1_2T = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s\theta_2 & -c\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{11}$$

$${}^2_3T = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & 0.41 \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0.0133 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{12}$$

$${}^3_4T = \begin{bmatrix} c\theta_4 & -s\theta_4 & 0 & 0 \\ 0 & 0 & 1 & 0.3111 \\ -s\theta_4 & -c\theta_4 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{13}$$

$${}^4_5T = \begin{bmatrix} c\theta_5 & -s\theta_5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s\theta_5 & c\theta_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{14}$$

$${}^5_6T = \begin{bmatrix} c\theta_6 & -s\theta_6 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s\theta_6 & -c\theta_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{15}$$

The complete model of forward kinematics, using a modified Denavit–Hartenberg convention [29], taking into account the actual angular displacement of each rotational joint, is

$${}^0_6T = {}^0_1T(\theta_1){}^1_2T(\theta_2){}^2_3T(\theta_3){}^3_4T(\theta_4){}^4_5T(\theta_5){}^5_6T(\theta_6) \tag{16}$$

In order to solve the inverse kinematics, we have to calculate the angular position of each rotational joint, knowing the desired orientation and position 0_6T :

$${}^0_6T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^0_1T(\theta_1){}^1_2T(\theta_2){}^2_3T(\theta_3){}^3_4T(\theta_4){}^4_5T(\theta_5){}^5_6T(\theta_6) \tag{17}$$

Values $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5$ and θ_6 in (17) are the desired angular positions, and 0_6T contents are the predefined desired orientations and positions. Using (17), we put ${}^0_1T(\theta_1)$ on the left-hand side of the equation:

$$\left[{}^0_1T(\theta_1)\right]^{-1}{}^0_6T = {}^1_2T(\theta_2){}^2_3T(\theta_3){}^3_4T(\theta_4){}^4_5T(\theta_5){}^5_6T(\theta_6) \tag{18}$$

Inverting the matrix 0_1T , we can write (18)

$$\begin{bmatrix} c\theta_1 & s\theta_1 & 0 & 0 \\ -s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^1_6T \tag{19}$$

According to Pieper’s approach, we equate the element placed on row 2, column 4 inside the matrix 1_6T with the element placed on row 2, column 4 from the matrix $\left[{}^0_1T(\theta_1)\right]^{-1}{}^0_6T$. The unknown angular displacement θ_1 can now be expressed by (20).

$$-s\theta_1 p_x + c\theta_1 p_y = 0.0133 \tag{20}$$

To solve an equation of this form, we make the following trigonometric substitutions:

$$\begin{aligned} p_x &= \rho c\phi \\ p_y &= \rho s\phi \end{aligned} \tag{21}$$

where ρ is the distance from the origin of the base coordinate system to the point defined by the x and y coordinates of the end-effector:

$$\rho = \sqrt{p_x^2 + p_y^2} = \text{Atan2}(p_y, p_x) \tag{22}$$

Substituting (21) into (20) [29],

$$c\theta_1 s\phi - s\theta_1 c\phi = \frac{0.0133}{\rho} \tag{23}$$

From the difference-of-angles formula,

$$s(\phi - \theta_1) = \frac{0.0133}{\rho} \tag{24}$$

Hence,

$$c(\phi - \theta_1) = \pm \sqrt{1 - \frac{0.00017689}{\rho^2}} \tag{25}$$

and so

$$\phi - \theta_1 = \text{Atan2}\left(\frac{0.0133}{\rho}, \pm\sqrt{1 - \frac{0.00017689}{\rho^2}}\right) \tag{26}$$

The solution for θ_1 can be written as

$$\theta_1 = \text{Atan2}(p_y, p_x) - \text{Atan2}\left(0.0133, \pm\sqrt{p_x^2 + p_y^2 - 0.00017689}\right) \tag{27}$$

We have found two possible solutions for the desired value of θ_1 , corresponding to the plus-or-minus sign in (27). Now that θ_1 is known, the left-hand side of (19) is known. By equating the elements on row 1, column 4 and row 3, column 4 from both sides of (19), we can write Equation (28) as follows [29]:

$$\begin{aligned} c\theta_1 p_x + s\theta_1 p_y &= 0.41c\theta_2 - 0.3111s\theta_{23} \\ -p_z &= 0.41s\theta_2 + 0.3111c\theta_{23} \end{aligned} \tag{28}$$

After taking the squares of Equations (28) and (20) and adding them, the resulting equations are obtained:

$$-(D_3 + D_4)s\theta_3 = \frac{p_x^2 + p_y^2 + p_z^2 - D_2^2 - e_2^2 - (D_3 + D_4)^2}{2D_2} \tag{29}$$

$$s\theta_3 = \frac{0.265 - p_x^2 - p_y^2 - p_z^2}{0.2551} = K \tag{30}$$

$$c\theta_3 = \pm\sqrt{1 - K^2} \tag{31}$$

$$\theta_3 = \text{Atan2}\left(K, \pm\sqrt{1 - K^2}\right) \tag{32}$$

The plus or minus sign in (32) leads to two different solutions for the angle θ_3 . We can rewrite Equation (17) so that the entire left-hand side becomes a function of three parameters θ_1, θ_2 and θ_3 . The only unknown parameter is θ_2 :

$$\begin{bmatrix} 0 \\ 3 \end{bmatrix} T(\theta_2) \tag{-1} \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^3_4T(\theta_4) {}^4_5T(\theta_5) {}^5_6T(\theta_6) \tag{33}$$

or

$$\begin{aligned} &\begin{bmatrix} c\theta_1 c\theta_{23} & s\theta_1 c\theta_{23} & -s\theta_{23} & -0.41c\theta_3 \\ -c\theta_1 s\theta_{23} & -s\theta_1 s\theta_{23} & -c\theta_{23} & 0.41s\theta_3 \\ -s\theta_1 & c\theta_1 & 0 & 0.0133 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \\ &= \begin{bmatrix} c\theta_4 c\theta_5 c\theta_6 - s\theta_4 s\theta_6 & -c\theta_6 s\theta_4 - c\theta_4 c\theta_5 s\theta_6 & -c\theta_4 s\theta_5 & 0 \\ c\theta_6 s\theta_5 & s\theta_5 s\theta_6 & c\theta_5 & 0.3111 \\ -c\theta_4 s\theta_6 - c\theta_5 c\theta_6 s\theta_4 & c\theta_5 s\theta_4 s\theta_6 - c\theta_4 c\theta_6 & s\theta_4 s\theta_5 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{34}$$

If we equate the elements on row 1, column 4 and row 2, column 4 from both sides of (34), we can write [29]

$$\begin{aligned} c\theta_1 c\theta_{23} p_x + s\theta_1 c\theta_{23} p_y - s\theta_{23} p_z - 0.41c\theta_3 &= 0 \\ -c\theta_1 s\theta_{23} p_x - s\theta_1 s\theta_{23} p_y - c\theta_{23} p_z + 0.41s\theta_3 &= 0.3111 \end{aligned} \tag{35}$$

These equations can be solved simultaneously for $s\theta_{23}$ and $c\theta_{23}$, resulting in

$$\begin{aligned} s\theta_{23} &= \frac{0.41(c\theta_1s\theta_3p_x+s\theta_1s\theta_3p_y-c\theta_3p_z)}{p_z^2+(c\theta_1p_x+s\theta_1p_y)^2} \\ c\theta_{23} &= \frac{0.41(c\theta_1c\theta_3p_x+s\theta_1c\theta_3p_y+s\theta_3p_z)-0.3111}{p_z^2+(c\theta_1p_x+s\theta_1p_y)^2} \end{aligned} \tag{36}$$

The denominators are equal and positive, so we solve for the sum of θ_2 and θ_3 as follows [29]:

$$\theta_{23} = \text{Atan2}(arg1, arg2) \tag{37}$$

where

$$\begin{aligned} arg1 &= [c\theta_1c\theta_{23}p_x + s\theta_1c\theta_{23}p_y - 0.41c\theta_3] \\ \text{and } arg2 &= [0.41(c\theta_1c\theta_3p_x + s\theta_1c\theta_3p_y + s\theta_3p_z) - 0.3111]. \end{aligned}$$

Equation (37) computes four values of θ_{23} , according to the four possible combinations of solutions for θ_1 and θ_3 ; then, four possible solutions for the desired angular position θ_2 are computed as follows:

$$\theta_2 = \theta_{23} - \theta_3 \tag{38}$$

The entire left side of (34) is known. If we equate the elements on row 1, column 3 and row 3, column 3 from both sides of (34), we can write:

$$\begin{aligned} r_{13}c\theta_1c\theta_{23} + r_{23}s\theta_1c\theta_{23} - r_{33}s\theta_{23} &= -c\theta_4s\theta_5 \\ -r_{13}s\theta_1 + r_{23}c\theta_1 &= s\theta_4s\theta_5 \end{aligned} \tag{39}$$

As long as $s\theta_5 \neq 0$, the solution for θ_4 is

$$\theta_4 = \text{Atan2}(-r_{13}s\theta_1 + r_{23}c\theta_1, r_{13}c\theta_1c\theta_{23} + r_{23}s\theta_1c\theta_{23} - r_{33}s\theta_{23}) \tag{40}$$

When $\theta_5 = 0$, the manipulator is in a configuration in which axes 6 and 4 line up and cause the same motion of the last link of the robot. In this case, all that can be solved for is the sum or difference of θ_4 and θ_6 . This situation is detected by checking whether both arguments of the Equation (40) are near to zero. If so, θ_4 is chosen arbitrarily, and when θ_6 is computed later, it will be computed accordingly.

If we consider (17), we can rewrite it so that the entire left-hand side is a function of only knowns and θ_4 , rewriting it as follows [29]:

$$[{}^0_4T(\theta_4)]^{-1} \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^4_5T(\theta_5) {}^5_6T(\theta_6) \tag{41}$$

$$= \begin{bmatrix} c\theta_1c\theta_{23}c\theta_4 + s\theta_1s\theta_4 & s\theta_1c\theta_{23}c\theta_4 - c\theta_1s\theta_4 & -s\theta_{23}c\theta_4 & 0.0133s\theta_4 - 0.41c\theta_3c\theta_4 \\ -c\theta_1c\theta_{23}s\theta_4 + s\theta_1c\theta_4 & -s\theta_1c\theta_{23}s\theta_4 - c\theta_1c\theta_4 & s\theta_{23}s\theta_4 & 0.0133c\theta_4 + 0.41c\theta_3s\theta_4 \\ -c\theta_1s\theta_{23} & -s\theta_1s\theta_{23} & -c\theta_{23} & 0.41s\theta_3 - D_3 - D_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{42}$$

$${}^4_5T(\theta_5) {}^5_6T(\theta_6) = \begin{bmatrix} c\theta_5c\theta_6 & -c\theta_5s\theta_6 & -s\theta_5 & 0 \\ s\theta_6 & c\theta_6 & 0 & 0 \\ c\theta_6s\theta_5 & -s\theta_5s\theta_6 & c\theta_5 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{43}$$

If we equate the elements on row 1, column 3 and row 3, column 3 from the both sides of (42), we obtain

$$\begin{aligned} r_{13}(c\theta_1c\theta_{23}c\theta_4 + r_{23}(s\theta_1c\theta_{23} - c\theta_1s\theta_4) - r_{33}s\theta_{23}c\theta_4) &= -s\theta_5 \\ -r_{13}c\theta_1s\theta_{23} - r_{23}s\theta_1s\theta_{23} - r_{33}c\theta_{23} &= c\theta_5 \end{aligned} \tag{44}$$

We solve for the desired value of θ_5 as follows:

$$\theta_5 = \text{Atan2}(s\theta_5, c\theta_5) \tag{45}$$

Applying the same method one more time, we compute ${}^0_5T^{-1}$ and write (17) in the following form:

$$\left[{}^0_5T(\theta_5) \right]^{-1} \begin{bmatrix} r_{11} & r_{12} & r_{13} & P_x \\ r_{21} & r_{22} & r_{23} & P_y \\ r_{31} & r_{32} & r_{33} & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^5_6T(\theta_6) \tag{46}$$

If we equate the elements on row 3, column 1 and row 1, column 1 from the both sides of (42), we can write

$$\begin{aligned} -r_{11}(c\theta_1c\theta_{23}s\theta_4 - s\theta_1c\theta_4) - r_{21}(s\theta_1c\theta_{23}s\theta_4 + c\theta_1c\theta_4) + r_{31}s\theta_{23}s\theta_4 &= s\theta_6 \\ r_{11}[(c\theta_1c\theta_{23}c\theta_4 + s\theta_1s\theta_4)c\theta_5 - c\theta_1s\theta_{23}s\theta_5] + \\ + r_{21}[(s\theta_1c\theta_{23}c\theta_4 - c\theta_1s\theta_4)c\theta_5 - s\theta_1s\theta_{23}s\theta_5] - \\ - r_{31}(s\theta_{23}c\theta_4c\theta_5 + c\theta_{23}s\theta_5) &= c\theta_6 \end{aligned} \tag{47}$$

$$\theta_6 = \text{Atan2}(s\theta_6, c\theta_6) \tag{48}$$

Because of the plus or minus signs appearing in (27) and (32), these equations compute four solutions. Additionally, there are four more solutions obtained by the wrist of the manipulator. For each of the four solutions computed above, we obtain the flipped solution by [29]

$$\begin{aligned} \theta'_4 &= \theta_4 + 180^\circ \\ \theta'_5 &= -\theta_5 \\ \theta'_6 &= \theta_6 + 180^\circ \end{aligned} \tag{49}$$

2.4. The Designed Mobile Manipulator’s Common Chain of Frames

To achieve a successful combination between the KUKA youBot mobile platform and the Kinova Jaco Gen 2 robot manipulator, we have to describe the position of the end-effector according the local frame, attached to the center of the mobile chassis (platform). In order to deploy the mathematical equations, we have to describe the connection between the Kinova manipulator’s base frame and the KUKA mobile platform frame using the classical Denavit–Hartenberg convention (18). The aim is to derive a homogeneous transformation ${}^{\text{mobP}}_{\text{manip}}T$ between these two frames; the transformation will be sized as a 4×4 matrix, and the matrix will include the rotations between frame $R^{\text{mobP}}_{\text{manip}}(t)$ and vector $o^{\text{mobP}}_{\text{manip}}(t)$ that describes the displacement between the origins of these two frames. The origin of the Kinova manipulator base frame is in negative direction according the x axis of the mobile platform, a fact that causes a negative sign to be attached to the Denavit–Hartenberg parameter a . The distance between the origins is denoted as h_2 .

$${}^{\text{mobP}}_{\text{manip}}T = \begin{bmatrix} R^{\text{mobP}}_{\text{manip}}(t) & o^{\text{mobP}}_{\text{manip}}(t) \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & -1 & -l_2 \\ -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{50}$$

To calculate the position and the orientation of the end-effector, we have to provide the product of seven homogeneous transformations:

$${}^{\text{mob}}P_6^{\text{manip}} = {}^{\text{mob}}P_{\text{manip}}^0 T_1^0(\theta_1) T_2^1(\theta_2) T_3^2(\theta_3) T_4^3(\theta_4) T_5^4(\theta_5) T_6^5(\theta_6) \quad (51)$$

The final result is shown in (Figure 6).

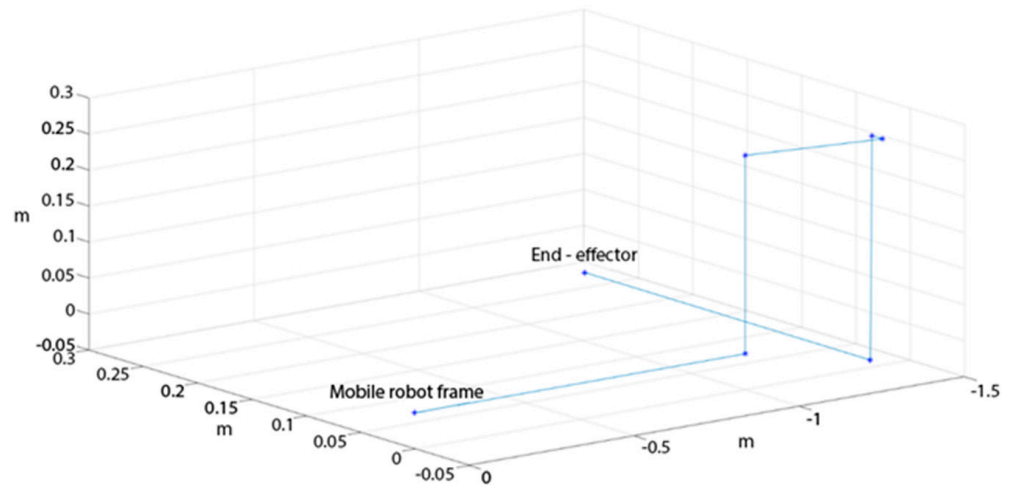


Figure 6. Position of the end-effector according to the mobile platform frame. The manipulator is placed in home position.

2.5. Jacobian

Since the Kinova Jaco Gen 2 manipulator is mounted upon the mobile platform KUKA youBot, in order to be able to maintain the stable position of the manipulator’s end-effector, while the platform is moving within its 2D workspace, we have to calculate the Jacobian matrix to determine the instantaneous velocity of the gripper along the Cartesian y axis (y axis of the base coordinate system). The manipulator’s y axis coincides with the mobile platform’s x axis. The goal that we imposed for our experiments is even simplified: to keep the position of the gripper stable, according to the inertial coordinate system of the environment, while the mobile platform is moving along a straight line, its x axis, for example. To achieve the purpose, we set the velocity of the youBot mobile platform according to the velocity of the gripper, using the opposite direction. The joint velocities are accessible through the messages, provided by the ROS stack, a concatenated set of ROS packages. The calculation of the Jacobi matrix is an essential mathematical element for the successful conduct of the experiments.

2.5.1. Linear Velocity

Consider frame $\{B\}$ to be attached to a rigid body. We must describe the motion of point BQ relative to the frame $\{A\}$, where we consider $\{A\}$ to be fixed. Frame $\{B\}$ is located relative to frame $\{A\}$, the position vector is ${}^A P_{BORG}$, the rotation matrix is ${}^A R_B$, and we will assume that the orientation of ${}^A R_B$ is constant with time, that is, the motion of point Q relative to $\{A\}$ is due to ${}^A P_{BORG}$ [29]. Equation (52) solves the linear velocity of point Q in terms of $\{A\}$. We have to express both components of the velocity in terms of $\{A\}$ and sum them, keeping the relative orientation between these two frames constant as follows:

$${}^A V_Q = {}^A V_{BORG} + {}^A R_B {}^B V_Q \quad (52)$$

2.5.2. Rotational Velocity

If the orientation between two frames is changing in time, the point that is fixed in frame $\{B\}$ is indicated by vector BQ . The rotational velocity of the frame $\{B\}$ relative to frame $\{A\}$ is expressed by a vector ${}^A \Omega_B$. We will consider that the linear velocity vector

${}^B V_Q$ is constant equal to zero. Point Q will have only rotational velocity ${}^A \Omega_B$ according frame {A}. We can observe both the magnitude and the direction of the change in vector ${}^B Q$, as viewed from {A}. The differential change in ${}^A Q$ is perpendicular to ${}^A Q$ and ${}^A \Omega_B$. Additionally, it is observable that the magnitude of the differential change is

$$|\Delta Q| = (|{}^A Q| \sin \theta (|{}^A \Omega_B| \Delta t)) \tag{53}$$

The vector cross-product is suggested by a defined condition of magnitude and direction. These conditions are satisfied by the following equation:

$${}^A V_Q = {}^A \Omega_B \times {}^A Q \tag{54}$$

Generally, the vector Q can also be changed according the frame {B}. This concept is described by

$$\begin{aligned} {}^A V_Q &= {}^A ({}^B V_Q) + {}^A \Omega_B \times {}^A Q = {}^A {}_B^R V_Q + {}^A \Omega_B \times {}^A {}_B^R Q = \\ &= {}^A V_{BORG} + {}^B R^B V_Q + {}^A \Omega_B \times {}^B R^B Q \end{aligned} \tag{55}$$

When the geometric approach is described, however, to investigate more precisely, we must describe also the mathematical approach.

To derive the relationship between the derivative of an orthonormal matrix and a certain skew-symmetric matrix, we will use $n \times n$ orthonormal matrix R:

$$RR^T = I_n \tag{56}$$

the matrix I_n is an $n \times n$ identity matrix. Differentiating (56), we derive

$$\dot{R}R^T + R\dot{R}^T = 0_n \tag{57}$$

The matrix 0_n is an $n \times n$ zero matrix; thus, (57) can be written as

$$\dot{R}R^T + \left(\dot{R}R^T\right)^T = S + S^T = 0_n, \tag{58}$$

where S is a skew-symmetric matrix. The relationship between the derivative of orthonormal matrices and skew-symmetric matrices is

$$S = \dot{R}R^{-1} \tag{59}$$

Let us consider a vector ${}^B P$ that is fixed in frame {B}. Its description in another frame {A} is

$${}^A P = {}^B R^B P \tag{60}$$

If frame {B} is rotating, it means that the derivative ${}^B \dot{R}$ is not a zero, so the resulting vector ${}^A P$ will also be changing:

$${}^A \dot{P} = {}^B \dot{R}^B P \tag{61}$$

Using the notation for velocity that it will give and (59), we have:

$${}^A V_P = {}^B \dot{R}^B P = {}^A \dot{R}^A R_{-1}^A P = {}^A S^A P \tag{62}$$

The skew-symmetric matrix in (59) is called the angular-velocity matrix. We assign the elements in a skew-symmetric matrix S as follows:

$$S = \begin{bmatrix} 0 & -\Omega_x & \Omega_y \\ \Omega_x & 0 & -\Omega_z \\ -\Omega_y & \Omega_z & 0 \end{bmatrix} \tag{63}$$

The column vector Ω , which corresponds to the 3×3 angular-velocity matrix, is an angular-velocity vector:

$$\Omega = \begin{bmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix} \tag{64}$$

Applying vector cross-product gives

$$SP = \Omega \times P \tag{65}$$

where P is any vector.

Hence, (59) can be written as follows:

$${}^A V_P = {}^A \Omega_B \times {}^A P \tag{66}$$

where the notation for Ω indicates that it is the angular-velocity vector specifying the motion of the frame $\{B\}$ with respect to frame $\{A\}$ [29].

2.5.3. Motion of the Robot’s Links

Notations v_i and ω_i are the linear and angular velocities of the origin of link frame $\{i\}$. The robot’s structure is a chain of connected links, where each one is capable of motion relative to its neighbors. To calculate the velocities of each link, the starting point is the robot’s base. The velocities of link $i + 1$ will be calculated according to the velocity of link i plus new velocity components added by link $i + 1$. We have to divide the velocity into two subparts: the linear velocity of the origin of the link frame and the rotational velocity of the link.

Rotational velocities can be added when both ω vectors are written with respect to the same frame. Therefore, the angular velocity of link $i + 1$ is the same as that of link i plus a new component caused by rotational velocity at joint $i + 1$. This statement can be written according to frame $\{i\}$ as follows:

$${}^i \omega_{i+1} = {}^i \omega_i + {}^i R_{i+1} \dot{\theta}_{i+1} {}^{i+1} \hat{Z}_{i+1} \tag{67}$$

Also

$$\dot{\theta}_{i+1} {}^{i+1} \hat{Z}_{i+1} = {}^{i+1} \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_{i+1} \end{bmatrix} \tag{68}$$

In order to represent the added rotational component due to motion at the joint in the frame $\{i + 1\}$, we have made use of the rotational matrix related to frames $\{i\}$ and $\{i + 1\}$. The rotational matrix describes the rotation of joint $i + 1$ using its description in frame $\{i\}$, and we can add the two components of angular velocities [29]. We can find the description of the angular velocity of link $i + 1$ according to frame $\{i + 1\}$ using:

$${}^{i+1} \omega_{i+1} = {}^{i+1} R^i \omega_i + \dot{\theta}_{i+1} {}^{i+1} \hat{Z}_{i+1} \tag{69}$$

The linear velocity of the origin of the frame $\{i + 1\}$ is equal to the velocity of frame $\{i\}$ plus a new component caused by the rotation of link i . It is described by the last part of (55), in frame $\{i\}$. One term vanishes because ${}^iP_{i+1}$ is constant. We have

$${}^iV_{i+1} = {}^iV_i + {}^i\omega_i \times {}^iP_{i+1} \tag{70}$$

Premultiplying both sides by ${}^{i+1}R$, we compute

$${}^{i+1}V_{i+1} = {}^{i+1}R \left({}^iV_i + {}^i\omega_i \times {}^iP_{i+1} \right) \tag{71}$$

Equations (69) and (71) are important results related to revolute joints. The corresponding relationships for the case when joint $i + 1$ is prismatic are

$$\begin{aligned} {}^{i+1}V_{i+1} &= {}^{i+1}R \left({}^iV_i + {}^i\omega_i \times {}^iP_{i+1} \right) + \dot{d}_{i+1} {}^{i+1}\hat{Z}_{i+1} \\ {}^{i+1}\omega_{i+1} &= {}^{i+1}R {}^i\omega_i \end{aligned} \tag{72}$$

Applying Equation (72) or (69) and (71) successively from link to link, we can compute ${}^N\omega_N$ and Nv_N .

The multidimensional form of the derivatives is known as Jacobian. If we have six functions and the number of the independent parameters is also six, we have [29]

$$\begin{aligned} y_1 &= f_1(x_1, x_2, x_3, x_4, x_5, x_6) \\ y_2 &= f_2(x_1, x_2, x_3, x_4, x_5, x_6) \\ &\vdots \\ y_6 &= f_6(x_1, x_2, x_3, x_4, x_5, x_6) \end{aligned} \tag{73}$$

Using a vector notation (72) can be written as

$$Y = F(X) \tag{74}$$

We can use the chain rule to calculate δy_i as follows [29]:

$$\begin{aligned} \delta y_1 &= \frac{\partial f_1}{\partial x_1} \delta x_1 + \frac{\partial f_1}{\partial x_2} \delta x_2 + \dots + \frac{\partial f_1}{\partial x_6} \delta x_6 \\ \delta y_2 &= \frac{\partial f_2}{\partial x_1} \delta x_1 + \frac{\partial f_2}{\partial x_2} \delta x_2 + \dots + \frac{\partial f_2}{\partial x_6} \delta x_6 \\ &\vdots \\ \delta y_6 &= \frac{\partial f_6}{\partial x_1} \delta x_1 + \frac{\partial f_6}{\partial x_2} \delta x_2 + \dots + \frac{\partial f_6}{\partial x_6} \delta x_6 \end{aligned} \tag{75}$$

In vector notation, (76) is

$$\delta Y = \frac{\partial F}{\partial X} \delta X \tag{76}$$

The Jacobian is a 6×6 matrix. It contains partial derivatives, and the notion of the matrix is J . If the functions $f_1(X)$ through $f_6(X)$ are nonlinear, then the partial derivatives are functions of x_i , so we can use the following notation:

$$\delta Y = J(X) \delta X \tag{77}$$

The Jacobian can be presented as mapping velocities in X to Y

$$\dot{Y} = J(X) \dot{X} \tag{78}$$

At each new instance, X has changed, which means that the linear transformation $J(X)$ has also changing. In robotics, the general purpose of Jacobian is that it relates joint velocities to the Cartesian velocities of the tip of the arm [29]

$${}^0\boldsymbol{v} = {}^0J(\Theta)\dot{\Theta} \quad (79)$$

where the vector of joint displacements is Θ , and the Cartesian velocities are represented by the vector \boldsymbol{v} . The superscript in front of the vector notations indicates in which frame the resulting Cartesian velocities are expressed. The shape of the vector $\dot{\Theta}$ is 6×1 and the vector ${}^0\boldsymbol{v}$ has the shape 6×1 . The vector ${}^0\boldsymbol{v}$ is split into two parts:

$${}^0\boldsymbol{v} = \begin{bmatrix} {}^0\boldsymbol{v} \\ {}^0\boldsymbol{\omega} \end{bmatrix} \quad (80)$$

The first three elements of ${}^0\boldsymbol{v}$ represent linear velocity, and the second three elements represent angular velocity. The Jacobian has a number of rows equal to the degrees of freedom in the Cartesian space, with columns representing the number of joints of the manipulator. The Jacobian can be found by directly differentiating the kinematic equations of the manipulator. This approach is easily implemented for linear velocities, but there is no 3×1 orientation vector whose derivative is $\boldsymbol{\omega}$. That is the reason why we introduced Equations (69) and (71).

When the Jacobian is nonsingular, we can calculate joint velocities from the given Cartesian velocities:

$$\dot{\Theta} = J^{-1}(\Theta){}^0\boldsymbol{v} \quad (81)$$

2.6. Maintaining the Constant Position of the Manipulator's Gripper While the Mobile Platform Is Moving

The position and orientation of the working tool of Kinova Jaco Gen2 are calculated by multiplying homogeneous transformations from (82) to (87). These transformations are derived using a classical Denavit–Hartenberg convention, as is presented in (18):

$${}^0_1T = \begin{bmatrix} c\theta_1 & 0 & s\theta_1 & 0 \\ -s\theta_1 & 0 & c\theta_1 & 0 \\ 0 & -1 & 0 & D_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (82)$$

$${}^1_2T = \begin{bmatrix} c\theta_2 & s\theta_2 & 0 & D_2c\theta_2 \\ s\theta_2 & c\theta_2 & 0 & D_2s\theta_2 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (83)$$

$${}^2_3T = \begin{bmatrix} c\theta_3 & 0 & s\theta_3 & 0 \\ s\theta_3 & 0 & -c\theta_3 & 0 \\ 0 & 1 & 0 & -e_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (84)$$

$${}^3_4T = \begin{bmatrix} c\theta_4 & 0 & s\theta_4 & 0 \\ s\theta_4 & 0 & -c\theta_4 & 0 \\ 0 & 1 & 0 & -(D_3 + D_4) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (85)$$

$${}^4_5T = \begin{bmatrix} c\theta_5 & 0 & s\theta_5 & 0 \\ s\theta_5 & 0 & -c\theta_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (86)$$

$${}^5_6T = \begin{bmatrix} c\theta_6 & s\theta_6 & 0 & 0 \\ s\theta_6 & c\theta_6 & 0 & 0 \\ 0 & 1 & -1 & -(D_5 + D_6) \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{87}$$

The vector Θ contains all the angular positions of the joints:

$$\Theta = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6]^T \tag{88}$$

The values of Θ are instantaneous. Since the investigated mobile manipulator has only rotational joints, we can use (69) and (71) to calculate the Jacobian. The output will be a vector with velocities in the Cartesian space, while the input values are the velocities of the manipulator’s joints $\dot{\Theta}$. As the values of Θ and $\dot{\Theta}$ were previously saved, we will use them to calculate 0v for each instance. The calculation of the Jacobian is performed in steps; in each step, the algorithm calculates a column of the matrix J.

$$\begin{aligned} {}^0_2R &= {}^0_2T(0:2,0:2) = [{}^0_1T(\theta_1){}^1_2T(\theta_2)](0:2,0:2) \\ {}^0_3R &= {}^0_3T(0:2,0:2) = [{}^0_1T(\theta_1){}^1_2T(\theta_2){}^2_3T(\theta_3)](0:2,0:2) \\ {}^0_4R &= {}^0_4T(0:2,0:2) = [{}^0_1T(\theta_1){}^1_2T(\theta_2){}^2_3T(\theta_3){}^3_4T(\theta_4)](0:2,0:2) \\ {}^0_5R &= {}^0_5T(0:2,0:2) = [{}^0_1T(\theta_1){}^1_2T(\theta_2){}^2_3T(\theta_3){}^3_4T(\theta_4){}^4_5T(\theta_5)](0:2,0:2) \\ {}^0_6R &= {}^0_6T(0:2,0:2) = [{}^0_1T(\theta_1){}^1_2T(\theta_2){}^2_3T(\theta_3){}^3_4T(\theta_4){}^4_5T(\theta_5){}^5_6T(\theta_6)](0:2,0:2) \end{aligned} \tag{89}$$

Using (86), we can calculate

$$J(0:5,0) = \begin{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times \left({}^0_6T(0:3,3) - \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \right) \\ \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \end{bmatrix} \tag{90}$$

$$J(0:5,1) = \begin{bmatrix} {}^0_1R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times ({}^0_6T(0:3,3) - {}^0_1T(0:3,3)) \\ {}^0_1R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \end{bmatrix} \tag{91}$$

$$J(0:5,2) = \begin{bmatrix} {}^0_2R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times ({}^0_6T(0:3,3) - {}^0_2T(0:3,3)) \\ {}^0_2R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \end{bmatrix} \tag{92}$$

$$J(0:5,3) = \begin{bmatrix} {}^0_3R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times ({}^0_6T(0:3,3) - {}^0_3T(0:3,3)) \\ {}^0_3R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \end{bmatrix} \tag{93}$$

$$J(0:5,4) = \begin{bmatrix} {}^0_4\mathbf{R} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times ({}^0_6\mathbf{T}(0:3,3) - {}^0_4\mathbf{T}(0:3,3)) \\ {}^0_4\mathbf{R} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \end{bmatrix} \quad (94)$$

$$J(0:5,5) = \begin{bmatrix} {}^0_5\mathbf{R} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times ({}^0_6\mathbf{T}(0:3,3) - {}^0_5\mathbf{T}(0:3,3)) \\ {}^0_5\mathbf{R} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \end{bmatrix} \quad (95)$$

The final Jacobian is derived by combining Equations (90) to (95).

$$\mathbf{J} = [\mathbf{J}(0:5,0), \mathbf{J}(0:5,1), \mathbf{J}(0:5,2), \mathbf{J}(0:5,3), \mathbf{J}(0:5,4), \mathbf{J}(0:5,5)] \quad (96)$$

The velocities of the joints are represented by

$$\dot{\Theta} = [\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3, \dot{\theta}_4, \dot{\theta}_5, \dot{\theta}_6]^T \quad (97)$$

The final Cartesian velocities are calculated using (96) and (97) by

$$\begin{bmatrix} {}^0\mathbf{v} \\ {}^0\boldsymbol{\omega} \end{bmatrix} = \mathbf{J}(\Theta)\dot{\Theta} \quad (98)$$

If we have to calculate $\dot{\Theta}$, we use the inverse Jacobian

$$\mathbf{J}^{-1} = \frac{1}{\det(\mathbf{J})} \text{adj}(\mathbf{J}) \quad (99)$$

$$\dot{\Theta} = \mathbf{J}^{-1}(\Theta) \begin{bmatrix} {}^0\mathbf{v} \\ {}^0\boldsymbol{\omega} \end{bmatrix} \quad (100)$$

3. Experimental Results

The proposed approach is validated by conducting a real-world experiment with the laboratory prototype of the proposed mobile manipulator. Both components of the introduced manipulator are ROS-supported. This gives a universal point of view regarding the implemented solutions, despite the differences in the architecture of the mobile manipulator subsystems.

Using the message provided by the Kinova ROS stack: `/${kinova_robotType}_driver'/out/joint_state`, we are able to observe the velocities and angular positions for each robot joint. The calculated movement speeds along the Kinova Jaco Gen 2's Cartesian coordinate system's x , y and z axes and angular velocities of ω_x , ω_y , and ω_z are fed to KUKA youBot. The data recorded during the experiments are used to display the results. The velocities are measured in radians per second. Based on this, we structure a matrix that contains the joint velocities during the experiment. Using these measurements, we are able to

determine the end-effector’s velocities along the Cartesian coordinate system’s axis and its angular velocities.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = J_{6 \times 6} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \\ \dot{\theta}_4 \\ \dot{\theta}_5 \\ \dot{\theta}_6 \end{bmatrix}_{6 \times 1} \tag{101}$$

where $\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3, \dot{\theta}_4, \dot{\theta}_5$ and $\dot{\theta}_6$ are joint velocities. Figure 7 shows the velocity ramp for each joint during the validation.

The values of the vector elements that represent the linear and angular velocities during the experiment are shown in Figure 9. It visualizes the results of applying formula (98). The velocities of the joints are applied to the earlier proposed algorithm. It calculates the vector that is given in (98). The values inside the vector provide a stable position for the end-effector in the 3D space, despite the movable base.

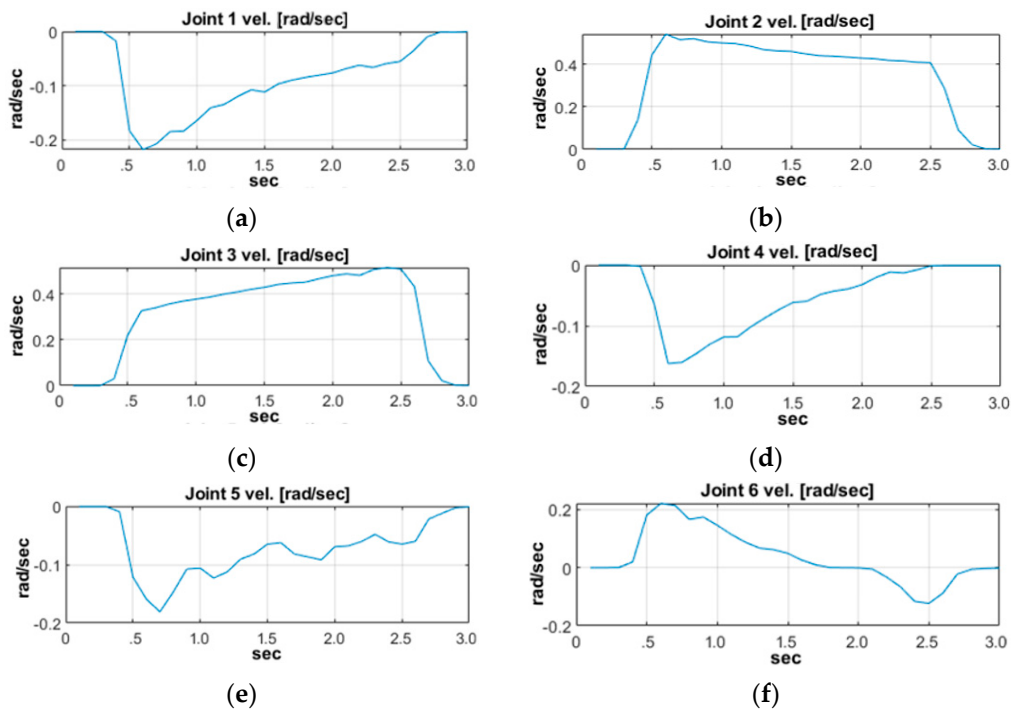


Figure 7. Joint velocities: (a) velocity of the first joint; (b) velocity of the second joint; (c) velocity of the third joint; (d) velocity of the fourth joint; (e) velocity of the fifth joint; (f) velocity of the sixth joint.

The positions along the axes and the orientations around them for the end-effector in the Cartesian coordinate system are shown in Figure 8.

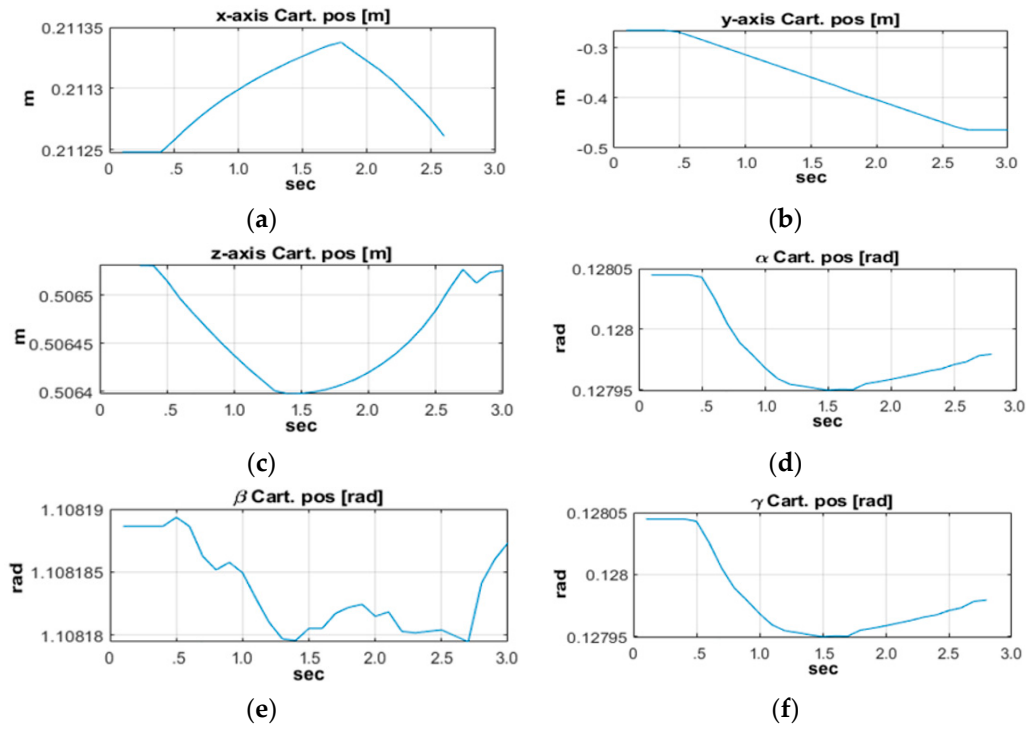


Figure 8. Measured positions and orientations of the end-effector. There are 3 positions and 3 orientations according to the manipulator’s base (Cartesian) coordinate system: (a) position along the x axis; (b) position along the y axis; (c) position along the z axis; (d) orientation around the x axis— α ; (e) orientation around the y axis— β ; (f) orientation around the z axis— γ .

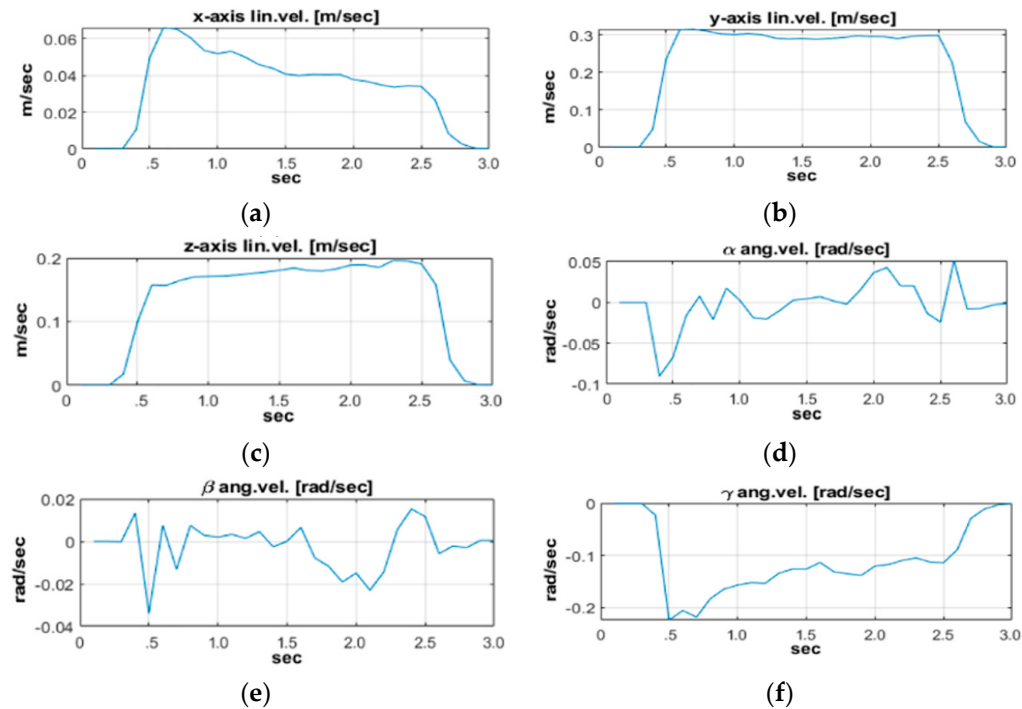


Figure 9. Calculated velocities according to the manipulator’s base coordinate system, along each axis and around each axis. These velocities applied to the base provide constant position of the end-effector during its movement according to the adjusted joint velocities vector $\dot{\Theta}$: (a) velocity along the x axis; (b) velocity along the y axis; (c) velocity along the z axis; (d) velocity around the x axis; (e) velocity around the y axis; (f) velocity around the z axis.

The displacement between the positions of the end-effector of the Kinova Jaco Gen 2 manipulator and the KUKA youBot mobile platform during the experiment is presented in Figure 10.

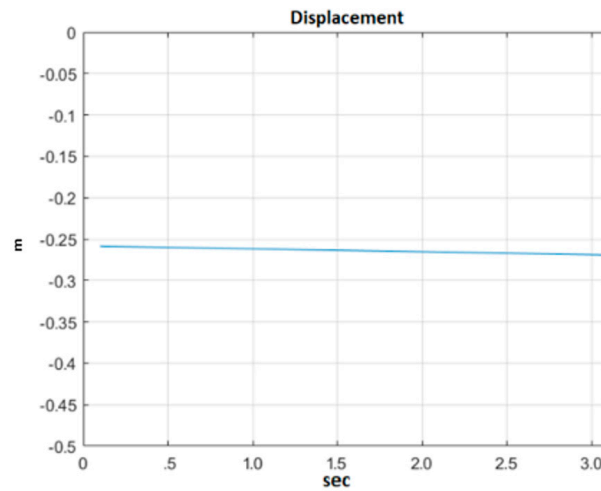


Figure 10. Displacement between the position of the movable base coordinate system and the position of the end-effector.

According to the proposed approach, we calculated the manipulator’s velocity along the axes of its base frame to keep its end-effector’s position constant during the movement of the base. The base is attached to the mobile platform KUKA youBot. To validate the proposed approach, we visualized manipulator’s position along its *y* axis during the movement of the mobile robot. The manipulator’s home position along *y* axis is around -0.25 m. The horizontality of the line shows that the velocities of the manipulator and mobile robot are approximately the same, slightly observable slope may be caused by not perfect environment conditions like the slippage of the mobile platform’s wheels.

Figure 11 shows the positions of the mobile platform and the manipulator gripper during the experiment. Additionally, the starting and the final positions of their movements are visible.

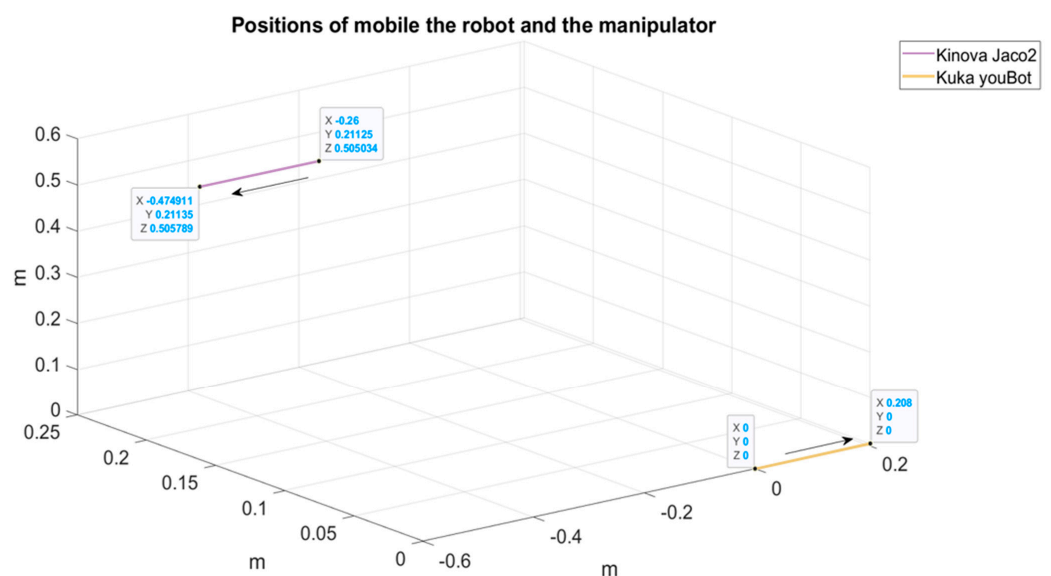


Figure 11. The positions of the mobile platform and manipulator gripper during the experiment.

4. Conclusions

We have proposed an affordable open-source mobile manipulator composed of the KUKA youBot mobile platform and a Kinova Jaco manipulator, and we have developed corresponding kinematic problems. The latter have been verified by running experiments with the designed laboratory prototype. The proposed open-source character of the mobile manipulator and the initial software provided by the ROS community can help robot researchers to concentrate directly on their research topics. The latter can more easily complete challenging research and development tasks, which will accelerate the technology's transfer and application.

Author Contributions: Conceptualization, A.V.T. and S.A.-S.; methodology, V.P. and S.A.-S.; software, V.P. and T.S.; validation, T.S. and V.P.; formal analysis, V.P.; investigation, S.A.-S.; resources, A.V.T. and S.A.-S.; data curation, T.S.; writing—original draft preparation, V.P.; writing—review and editing, A.V.T., T.S. and S.A.-S.; visualization, V.P. and T.S.; supervision, A.V.T.; project administration, S.A.-S.; funding acquisition, S.A.-S. and V.P. All authors have read and agreed to the published version of the manuscript.

Funding: The European Regional Development Fund within the Operational Programme “Bulgarian national recovery and resilience plan”, via the procedure for the direct provision of grants “Establishing of a network of research higher education institutions in Bulgaria”, and under Project BG-RRP-2.004-0005, “Improving the research capacity and quality to achieve international recognition and resilience of TU-Sofia (IDEAS)”.

Data Availability Statement: Data are available upon request.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; the collection, analyses, or interpretation of data; the writing of the manuscript; or the decision to publish the results.

References

1. ISO/TS 15066:2016; Robots and Robotic Devices—Collaborative Robots. International Organization for Standardization: Geneva, Switzerland, 2016. Available online: <https://www.iso.org/standard/62996.html> (accessed on 16 January 2024).
2. ISO 10218-1:2011; Robots and Robotic Devices—Safety Requirements for Industrial Robots—Part 1: Robots. International Organization for Standardization: Geneva, Switzerland, 2011. Available online: <https://www.iso.org/standard/51330.html> (accessed on 16 January 2024).
3. ISO 10218-2:2011; Robots and Robotic Devices—Safety Requirements for Industrial Robots—Part 2: Robot Systems and Integration. International Organization for Standardization: Geneva, Switzerland, 2011. Available online: <https://www.iso.org/standard/41571.html> (accessed on 16 January 2024).
4. Neto, P.; Pires, J.N.; Moreira, A.P. Accelerometer-based control of an industrial robotic arm. In Proceedings of the Robot and Human Interactive Communication RO-MAN 2009, Toyama, Japan, 27 September–2 October 2009. [CrossRef]
5. Ben Abdallah, I.; Bouteraa, Y. Gesture Control of 3DOF Robotic Arm Teleoperated by Kinect Sensor. In Proceedings of the 19th International Conference on Sciences and Techniques of Automatic Control & Computer Engineering (STA), Sousse, Tunisia, 24–26 March 2019. [CrossRef]
6. Ahmed, S.A.; Ramadan, M.M.; Fathy, E. Motion Control of Robot by Using Kinect Sensor. *Res. J. Appl. Sci. Eng. Technol.* **2014**, *8*, 1384–1388. [CrossRef]
7. Waldherr, S.; Romero, R.; Thrun, S.A. Gesture Based Interface for Human-Robot Interaction. *Auton. Robot.* **2000**, *9*, 151–173. [CrossRef]
8. Tchon, K.; Jakubiak, J.; Muszynski, R. Mobile manipulators with implicit aboard kinematics. In Proceedings of the 11th International Conference on Advanced Robotics ICAR 2003, Coimbra, Portugal, 30 June–3 July 2003. [CrossRef]
9. Bischoff, R.; Huggenberger, U.; Prassler, E. KUKA youBot—A mobile manipulator for research and education. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011. [CrossRef]
10. Natarajan, S.; Kasperski, S.; Eich, M. An Autonomous Mobile Manipulator for Collecting Sample Containers. In Proceedings of the International Symposium on Artificial Intelligence, Robotics and Automation in Space, Montreal, Canada, 17–19 June 2014.
11. Pepe, A.; Chiaravalli, D.; Melchiorri, C. A Hybrid Teleoperation Control Scheme for a Single-Arm Mobile Manipulator with Omnidirectional Wheels. In Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, Republic of Korea, 9–14 October 2016. [CrossRef]
12. Adiwahono, A.H.; Chua, Y.; Tee, K.P.; Liu, B. Automated door opening scheme for non-holonomic mobile manipulator. In Proceedings of the 2013 13th International Conference on Control, Automation and Systems (ICCAS 2013), Gwangju, Republic of Korea, 20–23 October 2013. [CrossRef]

13. Zimmermann, S.; Poranne, R.; Coros, S. Go Fetch!—Dynamic Grasps using Boston Dynamics Spot with External Robotic Arm. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021. [CrossRef]
14. Yamamoto, Y.; Yun, X. Coordinating locomotion and manipulation of a mobile manipulator. *IEEE Trans. Autom. Contr.* **1994**, *39*, 1326–1332. [CrossRef]
15. Seraji, H. A unified approach to motion control of mobile manipulators. *Int. J. Rob. Res.* **1998**, *17*, 107–118. [CrossRef]
16. Khatib, O.; Yokoi, K.; Chang, K.; Casai, A. Robots in human environments: Basic autonomous capabilities. *Int. J. Rob. Res.* **1999**, *18*, 684–696. [CrossRef]
17. Yamamoto, Y.; Yun, X. Unified analysis of mobility and manipulability of mobile manipulators. In Proceedings of the Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C), Detroit, MI, USA, 10–15 May 1999; pp. 1200–1206. [CrossRef]
18. Tchon, K.; Jakubiak, J.; Muszynski, R. Kinematics of mobile manipulators: A control theoretic perspective. *Arch. Control Sci.* **2001**, *11*, 79–105.
19. Sharma, S.; Kraetzschmar, G.K.; Scheurer, C.; Bischoff, R. Unified Closed Form Inverse Kinematics for the KUKA youBot. In Proceedings of the ROBOTIK 2012, 7th German Conference on Robotics, Munich, Germany, 21–22 May 2012; ISBN 978-3-8007-3418-4.
20. Liu, Y.; Li, H.; Ding, L.; Liu, L.; Liu, T.; Wang, J.; Gao, H. An Omnidirectional Mobile Operating Robot Based on Mecanum Wheel. In Proceedings of the 2017 2nd International Conference on Advanced Robotics and Mechatronics (ICARM), Hefei and Tai'an, China, 27–31 August 2017. [CrossRef]
21. Li, X.; Luo, L.Z.; Zhao, H.; Ge, D.S.; Ding, H. Inverse Kinematics Solution Based on Redundancy Modeling and Desired Behaviors Optimization for Dual Mobile Manipulators. *J. Intell. Robot. Syst.* **2023**, *108*, 37. [CrossRef]
22. Zhang, X.F.; Li, G.F.; Xiao, F.; Jiang, D.; Tao, B.; Kong, J.Y.; Jiang, G.Z.; Liu, Y. An inverse kinematics framework of mobile manipulator based on unique domain constraint. *Mech. Mach. Theory* **2023**, *183*, 105273. [CrossRef]
23. Colucci, G.; Botta, A.; Tagliavini, L.; Cavallone, P.; Baglieri, L.; Quaglia, G. Kinematic Modeling and Motion Planning of the Mobile Manipulator Agri.Q for Precision Agriculture. *Machines* **2022**, *5*, 321. [CrossRef]
24. Huang, Y.; Li, D.L.; Dong, K.J.; Zhang, W.C.; Gao, X.M. Inverse Kinematics Analysis and Mechatronics Design of Mobile Parallel Manipulator for Assisted Assembly. In Proceedings of the 2022 IEEE International Conference on Mechatronics and Automation (IEEE ICMA 2022), Guilin, China, 7–10 August 2022. [CrossRef]
25. Siegwart, R.; Nourbakhsh, I.R.; Scaramuzza, D. *Introduction to Autonomous Mobile Robots*; The MIT Press: Cambridge, MA, USA, 2011.
26. Ahmed, S.A.; Shakev, N.G.; Topalov, A.V.; Popov, V.L. Model-Free Detection and Following of Moving Objects by an Omnidirectional Mobile Robot using 2D Range Data. *IFAC-PapersOnLine* **2018**, *51*, 226–231. [CrossRef]
27. User Guide KINOVA@GEN2 Ultra Lightweight Robot. Available online: https://artifactory.kinovaapps.com/ui/api/v1/download?repoKey=generic-documentation-public&path=Documentation%252FGen2%252FTechnical%2520documentation%252FUser%2520Guide%252FUG-009_KINOVA_Gen2_Ultra_lightweight_robot_User_guide_EN_R03.pdf (accessed on 21 October 2023).
28. Craig, J.J. *Introduction to Robotics: Mechanics and Control*, 4th ed.; Pearson Prentice Hall: London, UK, 2017.
29. Lewis, W.L.; Dawson, D.M.; Abdallag, C.T. *Robot Manipulator Control Theory and Practice*, 2nd ed.; Publisher Marcel Dekker, Inc.: New York, NY, USA, 2004.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.