Research Article

A Solution of the Inverse Kinematics Problem for a 7-Degrees-of-Freedom Serial Redundant Manipulator Using Gröbner Bases Theory

Sérgio Ricardo Xavier da Silva ¹, Leizer Schnitman ¹, and Vitalino Cesca Filho ²

¹*Mechatronics Graduation Program, Departments of Computer Science and Mechanical Engineering, Federal University of Bahia–UFBA, 02 Professor Aristides Novis Street, Salvador, BA 40210-630, Brazil*

²*Federal University of Pampa–UNIPAMPA, 111 Pedro Anunciação Avenue, Caçapava do Sul, Bagé, RS 96570-000, Brazil*

Correspondence should be addressed to Sérgio Ricardo Xavier da Silva; srxsilva@uneb.br

Received 17 December 2020; Revised 12 April 2021; Accepted 9 June 2021; Published 28 July 2021

Academic Editor: Antonio M. Gonçalves de Lima

Copyright © 2021 Sérgio Ricardo Xavier da Silva et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This article presents a solution of the inverse kinematics problem of 7-degrees-of-freedom serial redundant manipulators. A 7-degrees-of-freedom (7-DoF) redundant manipulator can avoid obstacles and thus improve operational performance. However, its inverse kinematics is difficult to solve since it has one more DoF than that necessary for reaching the whole workspace, which causes infinite solutions. In this article, Gröbner bases theory is proposed to solve the inverse kinematics. First, the Denavit–Hartenberg model for the manipulator is established. Second, different joint configurations are obtained using Gröbner bases theory. All solutions are confirmed with the aid of algebraic computing software, confirming that this method is accurate and easy to be implemented.

1. Introduction

Two well-known classic problems in the kinematics of robotic manipulators are forward and inverse kinematics. The solution of the forward kinematic problem is independent of the geometry of the robot. The same does not occur with the inverse kinematics problem since the procedure to obtain the equations depends exclusively on the geometric configuration of the robotic manipulator. This limitation makes this problem much more complicated due to the difficulty in finding a sequence of steps that facilitate resolution.

Denavit and Hartenberg [1] presented a convention to standardize the reference coordinate systems for spatial links; after ten years, the same authors developed the algorithm for solving the forward and inverse kinematics problems of articulated systems [2]. The importance of this algorithm for the kinematics analysis of robotic manipulators was presented by Paul [3] sixteen years after.

According to Sciavicco and Siciliano [4], when inverse kinematics is solved from the homogeneous transformation matrix, the following issues can be encountered: the need for additional methods and algorithms to solve the problem and the complexity of matrix calculations, which require more considerable computational efforts. An alternative method for the solution of inverse kinematics will be used in this article, namely, Gröbner bases theory.

The Gröbner bases theory consists of an algebraic algorithm that can be applied to a given set of nonzero polynomials, producing a finite set of generators so that it is possible to identify, from these generators when a specific polynomial belongs to the original given set [5]. When finding solutions for this set of generators, those are also solutions for the whole set. With the help of algebraic computing software, all these solutions can be calculated, including cases which are not so easily solved using matrix algebra or geometric applications due to the complexity of the geometry of some manipulator robots.

FIGURE 1: Input and output of Buchberger's algorithm [6].

Input: F , a finite subset of $k[x_1, \dots, x_n]$.

Output: $G = GB(F)$.

$G = F$

for each pair (f, g) of elements in G do

$h = S(f, g)$

$h = \text{normal form of } h$

if $h \neq 0$ then $G = G \cup \{h\}$ reduce the elements in G modulo each other end.

FIGURE 2: Algorithm for computing Gröbner bases [11].

An analysis of computational efficiency in the solution of the inverse kinematics problem of an anthropomorphic robot can be obtained in a previous publication [6]. Two approaches were investigated: the first approach used Paul's method and the second approach used Gröbner bases theory. Based on the computational performance data obtained, Gröbner bases theory is more efficient than Paul's method for solving the inverse kinematics of 6-degrees-of-freedom anthropomorphic robots.

However, the present work intends to go further. It brings an efficient procedure to solve the inverse kinematic problem of a 7-DoF serial redundant manipulator using Gröbner bases theory.

A 7-DoF redundant manipulator has one more degree of freedom than that necessary for achieving a position and orientation of the robotic manipulator end effector. This extra DoF can be used to improve operational performance, such as avoid obstacles and prevent joint limits. Meanwhile, their inverse kinematics solution is challenging because of the extra degree of freedom, which causes infinite solutions.

Tolani et al. [7] presented a set of inverse kinematic real-time algorithms for a 7-DoF nonoffset serial arm. The authors presented a work using joint limit avoidance as a primary strategy for redundancy resolution, and two algorithms were introduced to address the inverse kinematics problem of under-constrained orientation and aiming, based on numerical approaches.

Kuhlemann et al. [8] presented a robust inverse kinematics algorithm that controls the manipulator elbow redundancy. Their work, however, only addresses the inverse kinematic computation and does not allude to joint limit avoidance methods.

With the development of intelligent algorithms, genetic algorithms and neural networks are used to solve the inverse kinematics of redundant manipulators. However, these two algorithms are complex and require many calculations, which is hard to implement in real-time systems. Furthermore, it is not possible to guarantee calculation accuracy.

The main goal of this article is to solve the problem of the inverse kinematics of redundant manipulators using Gröbner bases theory, which will demonstrate that the proposed method is precise and computationally efficient.

2. Gröbner Bases Theory

The core of Gröbner bases theory is an algorithm used to solve systems of polynomial equations. It is known as Buchberger's algorithm and applies computational algebra techniques to polynomial ideals, producing a so-called Gröbner basis, a new polynomial set [6].

According to Cox et al. [9], briefly, a Gröbner basis for a polynomial equation system is a new collection of polynomials in which the null set coincides with the solution set of the original system; however, this new system is simpler to solve.

According to Sturmfels [10], a Gröbner basis can be computed so that the polynomials have a property known as the elimination property. This elimination property makes the computation of solutions of a polynomial system, which is a Gröbner basis to be mathematically efficient. Through retroactive substitutions, a solution can be found for the remaining set of variables.

It can be seen in Figure 1 that all variables can be solved consecutively because the obtained Gröbner basis is a triangular equation system.

Buchberger's algorithm receives as input a finite set $F = \{f_1, f_2, \dots, f_n\}$ of polynomials and provides a set G , which is a Gröbner basis of F . This algorithm is shown in Figure 2.

The combination $S(f, g) = L/lt(f)f - L/lt(g)g$ is called the S-polynomial of f and g , where lt stands for the leading term of each polynomial, and $L = \text{lcm}(lp(f), lp(g))$, the least common multiple among the leading power products (lp) of polynomials.

Faugère [12] developed an efficient algorithm for computing Gröbner bases: the F4 algorithm. This algorithm computes successive truncated Gröbner bases, and it replaces the classical polynomial reduction found in Buchberger's algorithm by the simultaneous reduction of several polynomials.

In the present article, the Gröbner bases were computed on an extension of the field of rational numbers, namely, the rational numbers with specific square roots included. This procedure is run automatically by the software MAPLE when square roots are written in the input. Therefore, the approximation is made only in the final evaluation of solutions, so the Gröbner bases computation does not get

FIGURE 3: Establishing coordinate frames for the Kuka LBR iiwa 7 R800.

FIGURE 4: Two coordinate frames displaced in the same three-dimensional space.

affected. Ten decimals were used in the approximation of the final solution.

With the aid of algebraic computing software, a Gröbner basis can be calculated more quickly. Many algebraic computing systems include packages to deal with Gröbner bases, e.g., MAPLE, Singular, AXIOM, muMath, Macaulay, Mathematica, Reduce, and CoCoA.

3. The Denavit–Hartenberg Model of a 7-Degrees-of-Freedom Redundant Manipulator

The LBR iiwa 7 R800 from KUKA AG is an example of a robotic manipulator with 7-DoF and having only rotary joints, i.e., it is a 7R redundant manipulator. This redundant manipulator is shown in Figure 3, with its seven frames attached to the links.

For a kinematic chain with n joints and $n - 1$ links, each joint is assigned with a coordinate system (x, y, z) . Each coordinate system can be aligned through a series of

rotations and translations. Thus, each joint may be represented by a homogeneous transformation matrix that describes the specific rotation or translation required to align the joint $i - 1$ with the joint i , which can be the combination of a rotation and a translation matrix. The product of these matrices determines the final position of the n^{th} joint, which represents the end effector of the robotic manipulator.

A homogeneous matrix (1) is used to describe the direction vectors for the axes x_2 , y_2 , and z_2 and a reference system $\{2\}$ in terms of the direction vectors for the axes x_1 , y_1 , and z_1 of a reference system $\{1\}$:

$$\begin{bmatrix} n_x & o_x & a_x & x \\ n_y & o_y & a_y & y \\ n_z & o_z & a_z & z \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (1)$$

The two coordinate systems are represented in Figure 4.

Transformation operators (rotation and translation) are used to align frames. A_{mn} represents the homogeneous transformation matrix that relates points in the reference system n to the reference system m . In this way, $A_{0n} = A_{01} \cdot A_{12} \cdot A_{23} \cdot \dots \cdot A_{(n-1)n}$.

A Cartesian coordinate system must be assigned to each joint of the manipulator robot so that each connected link is assigned with a specific coordinate axis. Each coordinate system is determined based on the following rules: the z_{k-1} axis lies along the axis of motion (rotation or translation) of the joint k ; the x_k axis is perpendicular to the z_{k-1} axis, with its positive direction being towards the z_k axis direction, and the y_k axis is chosen so that the three axes form an orthogonal system satisfying the right-hand rule [13].

The four parameters a_k , d_k , α_k , and θ_k shown in Figure 5 represent a_k , the distance along x_k , from the origin O_k to the intersection of the axes x_k and z_{k-1} , d_k , the distance along z_{k-1} , from O_{k-1} to the intersection of the axes x_k and z_{k-1} , α_k , the angle of the z_{k-1} axis towards the z_k axis, measured around the x_k , and θ_k , the angle of the x_{k-1} axis towards the x_k axis, measured around the z_{k-1} axis (signal obtained by the right-hand rule).

TABLE 1: Denavit–Hartenberg parameters of 7-DoF redundant manipulator.

i	α_i (rad)	a_i (mm)	d_i (mm)	θ_i (rad)
1	$-\frac{\pi}{2}$	0	d_1	θ_1
2	$\frac{\pi}{2}$	0	0	θ_2
3	$-\frac{\pi}{2}$	0	d_3	θ_3
4	$\frac{\pi}{2}$	0	0	θ_4
5	$-\frac{\pi}{2}$	0	d_5	θ_5
6	$\frac{\pi}{2}$	0	0	θ_6
7	0	0	d_7	θ_7

According to Tsai [15], in the representation of Denavit–Hartenberg, each matrix is given by the product of four fundamental transformations involving rotations (R) and translations (Tr), as can be observed in (2). We use s_{θ_k} for $\sin(\theta_k)$ and c_{θ_k} for $\cos(\theta_k)$:

$${}^{k-1}A_k = R_{z_k, \theta_k} \cdot Tr_{z_k, d_k} \cdot Tr_{x_k, a_k} \cdot R_{x_k, \alpha_k} \quad (2)$$

$$= \begin{bmatrix} c_{\theta_k} & -s_{\theta_k} & 0 & 0 \\ s_{\theta_k} & c_{\theta_k} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_k \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & a_k \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_{\alpha_k} & -s_{\alpha_k} & 0 \\ 0 & s_{\alpha_k} & c_{\alpha_k} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

The term R_{z_k, θ_k} represents the rotation θ_k around the z_k axis, Tr_{z_k, d_k} is the translation d_k along the z_k axis, Tr_{x_k, a_k} is the translation a_k along the x_k axis, and, finally, R_{x_k, α_k} represents the rotation α_k around the x_k axis.

This operation results in the matrix represented by (3). This matrix defines the mapping of coordinates between two consecutive manipulator links of the robot:

$${}^{k-1}A_k = \begin{bmatrix} c_{\theta_k} & -s_{\theta_k}c_{\alpha_k} & s_{\theta_k}s_{\alpha_k} & a_kc_{\theta_k} \\ s_{\theta_k} & c_{\theta_k}c_{\alpha_k} & -c_{\theta_k}s_{\alpha_k} & a_k s_{\theta_k} \\ 0 & s_{\alpha_k} & c_{\alpha_k} & d_k \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (3)$$

Two of the four parameters used as a reference have a direct relationship with the component of the joint variables. For rotational joints, the joint variable is the angle θ_k , while the joint variable is the distance between the links d_k if it is prismatic. These parameters are obtained from the algorithm proposed by Denavit and Hartenberg.

By multiplying all the transformation matrices obtained by the Denavit–Hartenberg algorithm, we obtain the homogeneous transformation matrix that goes from the base to the end effector, that is, by varying k from 0 to n . This resulting matrix can be considered as the solution to the forward kinematics problem. Briefly, for a manipulator with n degrees of freedom,

FIGURE 5: Representation of Denavit–Hartenberg parameters [14].

$$T = A_{0n} = A_{01} \cdot A_{12} \cdot A_{23} \cdot \dots \cdot A_{(n-1)n} = \begin{bmatrix} n_x & o_x & a_x & x \\ n_y & o_y & a_y & y \\ n_z & o_z & a_z & z \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (4)$$

where each matrix A_{ij} is a Denavit–Hartenberg matrix, defined in (3).

All Denavit–Hartenberg parameters of this manipulator are listed in Table 1, where i represents the joint number.

After constructing all the transformation matrices (5) to (11) using the Denavit–Hartenberg parameters and multiplying them (12), we obtain the homogeneous transformation matrix (13), solution of the 7-DoF redundant manipulator forward kinematics problem. We now use s_k for $\sin(\theta_k)$ and c_k for $\cos(\theta_k)$:

$${}^0A_1 = \begin{bmatrix} c_1 & 0 & -s_1 & 0 \\ s_1 & 0 & c_1 & 0 \\ 0 & -1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (5)$$

$${}^1A_2 = \begin{bmatrix} c_2 & 0 & s_2 & 0 \\ s_2 & 0 & -c_2 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (6)$$

$${}^2A_3 = \begin{bmatrix} c_3 & 0 & -s_3 & 0 \\ s_3 & 0 & c_3 & 0 \\ 0 & -1 & 0 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (7)$$

$${}^3A_4 = \begin{bmatrix} c_4 & 0 & s_4 & 0 \\ s_4 & 0 & -c_4 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (8)$$

$${}^4A_5 = \begin{bmatrix} c_5 & 0 & -s_5 & 0 \\ s_5 & 0 & c_5 & 0 \\ 0 & -1 & 0 & d_5 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (9)$$

$${}^5A_6 = \begin{bmatrix} c_6 & 0 & s_6 & 0 \\ s_6 & 0 & -c_6 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (10)$$

$${}^6A_7 = \begin{bmatrix} c_7 & -s_7 & 0 & 0 \\ s_7 & c_7 & 0 & 0 \\ 0 & 0 & 1 & d_7 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (11)$$

$$T = {}^0A_1 \cdot {}^1A_2 \cdot {}^2A_3 \cdot {}^3A_4 \cdot {}^4A_5 \cdot {}^5A_6 \cdot {}^6A_7, \quad (12)$$

$$T = \begin{bmatrix} n_x & o_x & a_x & x \\ n_y & o_y & a_y & y \\ n_z & o_z & a_z & z \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (13)$$

where the entries of T are given by

$$\begin{aligned} n_x = & (((((c_1c_2c_3 - s_1s_3)c_4 - c_1s_2s_4)c_5 + (-c_1c_2s_3 - c_3s_1)s_5)c_6 \\ & + (-c_1c_2c_3 - s_1s_3)s_4 - c_1s_2c_4)s_6)c_7 \\ & + (-c_1c_2c_3 - s_1s_3)c_4 - c_1s_2s_4)s_5 + (-c_1c_2s_3 - c_3s_1)c_5)s_7, \end{aligned} \quad (14)$$

$$\begin{aligned} n_y = & (((((c_2c_3s_1 + c_1s_3)c_4 - s_1s_2s_4)c_5 + (-c_2s_1s_3 + c_1c_3)s_5)s_6 \\ & + (-c_2c_3s_1 + c_1s_3)s_4 - s_1s_2c_4)s_6)c_7 \\ & + (-((c_2c_3s_1 + c_1s_3)c_4 - s_1s_2s_4)s_5 + (-c_2s_1s_3 + c_1c_3)c_5)s_7, \end{aligned} \quad (15)$$

$$\begin{aligned} n_z = & ((((-c_3c_4s_2 - c_2s_4)c_5 + s_2s_3s_5)c_6 + (c_3s_2s_4 - c_2c_4)s_6)c_7 \\ & + (-c_3c_4s_2 - c_2s_4)s_5 + s_2s_3c_5)s_7, \end{aligned} \quad (16)$$

$$\begin{aligned} o_x = & -((((c_1c_2c_3 - s_1s_3)c_4 - c_1s_2s_4)c_5 + (-c_1c_2s_3 - c_3s_1)s_5)c_6 \\ & + (-c_1c_2c_3 - s_1s_3)s_4 - c_1s_2c_4)s_6)s_7 \\ & + (-((c_1c_2c_3 - s_1s_3)c_4 - c_1s_2s_4)s_5 \\ & + (-c_1c_2s_3 - c_3s_1)c_5)c_7, \end{aligned} \quad (17)$$

$$\begin{aligned} o_y = & -((((s_1c_2c_3 + c_1s_3)c_4 - s_1s_2s_4)c_5 + (-s_1c_2s_3 + c_1c_3)s_5)c_6 \\ & + (-s_1c_2c_3 + c_1s_3)s_4 - s_1s_2c_4)s_6)s_7 \\ & + (-((s_1c_2c_3 + c_1s_3)c_4 - s_1s_2s_4)s_5 + (-s_1c_2s_3 + c_1c_3)c_5)c_7, \end{aligned} \quad (18)$$

$$\begin{aligned} o_z = & -(((((-c_3c_4s_2 - c_2s_4)c_5 + s_2s_3s_5)c_6 + (c_3s_2s_4 - c_2c_4)s_6)s_7 \\ & + (-c_3c_4s_2 - c_2s_4)s_5 + s_2s_3c_5)c_7, \end{aligned} \quad (19)$$

$$\begin{aligned} a_x = & (((c_1c_2c_3 - s_1s_3)c_4 - c_1s_2s_4)c_5 + (-c_1c_2s_3 - c_3s_1)s_5)s_6 \\ & - (-c_1c_2c_3 - s_1s_3)s_4 - c_1s_2c_4)c_6, \end{aligned} \quad (20)$$

$$\begin{aligned} a_y = & (((c_2c_3s_1 + c_1s_3)c_4 - s_1s_2s_4)c_5 + (-c_2s_1s_3 + c_1c_3)s_5)s_6 \\ & - (-c_2c_3s_1 + c_1s_3)s_4 - s_1s_2c_4)c_6, \end{aligned} \quad (21)$$

$$\begin{aligned} a_z = & ((-c_3c_4s_2 - c_2s_4)c_5 + s_2s_3s_5)s_6 - (c_3s_2s_4 - c_2c_4)c_6, \end{aligned} \quad (22)$$

$$\begin{aligned} x = & (((((c_1c_2c_3 - s_1s_3)c_4 - c_1s_2s_4)c_5 + (-c_1c_2s_3 - c_3s_1)s_5)s_6 \\ & - (-c_1c_2c_3 - s_1s_3)s_4 - c_1s_2c_4)c_6)d_7 \\ & + ((c_1c_2c_3 - s_1s_3)s_4 + c_1s_2c_4)d_5 + c_1s_2d_3, \end{aligned} \quad (23)$$

$$\begin{aligned} y = & (((((c_2c_3s_1 + c_1s_3)c_4 - s_1s_2s_4)c_5 + (-c_2s_1s_3 + c_1c_3)s_5)s_6 \\ & - (-c_2c_3s_1 + c_1s_3)s_4 - s_1s_2c_4)c_6)d_7 + ((c_2c_3s_1 + c_1s_3)s_4 \\ & + s_1s_2c_4)d_5 + s_1s_2d_3, \end{aligned} \quad (24)$$

$$\begin{aligned} z = & ((((-c_3c_4s_2 - c_2s_4)c_5 + s_2s_3s_5)s_6 - (c_3s_2s_4 - c_2c_4)c_6)d_7 \\ & + (-c_3s_2s_4 + c_2c_4)d_5 + c_2d_3 + d_1. \end{aligned} \quad (25)$$

For the solution of inverse kinematics of a robotic manipulator with seven degrees of freedom, we have 12 equations and seven unknowns. However, among the nine equations that arise from the portion of the rotational matrix of T , only three are independent. These added to the equations of the portion of the position vector of T , which results in six equations with seven unknowns. Such equations are nonlinear and transcendental, which can be quite tricky to solve.

According to Ghosal [16], in a redundant manipulator, the number of joint variables will always be higher than the number of equations.

For three-dimensional space, six variables can describe the position and orientation of the robotic manipulator end effector. Therefore, a manipulator with six degrees of freedom can reach any given joints configuration in its working space. However, this kind of manipulator cannot reach a given pose if its link paths have obstacles or fail to realize a desired directional motion if its end effector is close to the singularity.

4. Inverse Kinematics Using Gröbner Bases Theory

The inverse kinematics problem consists of finding all combinations of joint configurations that will position the end

effector of a robot at a given point in space [17]. To solve this problem, one must determine polynomial equations that model the movement of the robotic arm in each joint configuration. With all these elements, one must seek a set of solutions from the possible movements of each joint.

To determine all joint configurations to position and orient the end effector on a given point in space, the following joint values, $\theta = [(\pi/6), (\pi/4), \pi, \pi, (\pi/3), (\pi/3), -(\pi/3)]$ radians, and the nonzero parameters, $d_1 = 380$, $d_3 = 328$, $d_5 = 323$, and $d_7 = 80$, were replaced in (12), thus obtaining

$$T = \begin{bmatrix} \frac{(7\sqrt{3} + 6)\sqrt{2}}{32} - \frac{\sqrt{3}}{16} & \frac{5}{16} + \frac{(6\sqrt{3} - 3)\sqrt{2}}{32} & \frac{\sqrt{3}((\sqrt{2} + 2)\sqrt{3} - 2\sqrt{2})}{16} & 15\sqrt{2} + 30 - \frac{35\sqrt{6}}{4} \\ \frac{3}{16} + \frac{(2\sqrt{3} + 7)\sqrt{2}}{32} & \frac{\sqrt{3}(2\sqrt{6} - \sqrt{2} - 10)}{32} & \frac{(\sqrt{3} - 2)\sqrt{2}}{16} - \frac{3\sqrt{3}}{8} & (-30 + 5\sqrt{2})\sqrt{3} - \frac{35\sqrt{2}}{4} \\ -\frac{7\sqrt{2}}{16} + \frac{\sqrt{6}}{8} & \frac{\sqrt{6}(1 + 2\sqrt{3})}{16} & -\frac{\sqrt{2}(2 + \sqrt{3})}{8} & 380 - 10\sqrt{6} - \frac{35\sqrt{2}}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (26)$$

From this point, equations (14) to (25) are rewritten by substituting $n_x, n_y, n_z, o_x, o_y, o_z, a_x, a_y, a_z, x, y$, and z by the respective values of the obtained numerical matrix, which exactly equals to matrix (26). All these 12 polynomials, added to the seven polynomials obtained from the fundamental trigonometric relation for each joint variable, are

used to generate a Gröbner basis. The lexicographic order $c_7 > s_7 > c_6 > s_6 > c_5 > s_5 > c_4 > s_4 > c_3 > s_3 > c_2 > s_2 > c_1 > s_1$ was selected for this robot.

Performing 3511 floating-point operations (169 additions, 606 multiplications, and 342 functions), we obtain the following Gröbner basis consisting of 20 polynomials:

$$g_1 = 16s_1^4 - 8s_1^2 + 1, \quad (27)$$

$$g_2 = 8\sqrt{3}s_1^3 - 5\sqrt{3}s_1 + 3c_1, \quad (28)$$

$$g_3 = -4s_1^3\sqrt{2} + 4s_1^2s_2 + \sqrt{2}s_1 - s_2, \quad (29)$$

$$g_4 = -2s_1s_2\sqrt{2} + 2s_1^2 + s_2^2, \quad (30)$$

$$g_5 = -4\sqrt{2}s_1^2 + 8s_1s_2 - 3\sqrt{2} + 4c_2, \quad (31)$$

$$g_6 = 4s_3\sqrt{6}s_1^2 - 8\sqrt{3}s_1s_2s_3 + s_3\sqrt{6} + 4c_3s_1^2 - c_3, \quad (32)$$

$$g_7 = 28\sqrt{3}s_1^3s_3^2 + 6\sqrt{6}s_2s_3^2 - 4\sqrt{3}s_1^3 - 19\sqrt{3}s_1s_3^2 - 6\sqrt{2}c_3s_1s_3 + 6c_3s_2s_3 + \sqrt{3}s_1, \quad (33)$$

$$g_8 = c_3^2 + s_3^2 - 1, \quad (34)$$

$$g_9 = 140\sqrt{6}s_1^3s_3 - 95\sqrt{6}s_1s_3 + 60\sqrt{3}s_2s_3 + 30c_3s_2\sqrt{2} - 60c_3s_1 + 1938s_4, \quad (35)$$

$$g_{10} = c_4 + 1, \quad (36)$$

$$g_{11} = 164\sqrt{6} - 984\sqrt{3} + 969\sqrt{2} - 5814 + (-656\sqrt{6} - 3936\sqrt{3} - 3876\sqrt{2})s_1^2 + 3936\sqrt{6}s_1s_2 - 3876\sqrt{3}s_3 + (10496\sqrt{3} + 7752\sqrt{2} - 15504)s_1^2s_3^2 + 7752s_5^2 + (-10496\sqrt{6} + 15504\sqrt{2})s_3^2s_2s_1 + (2624\sqrt{3} - 1938\sqrt{2})s_3^2, \quad (37)$$

$$g_{12} = (-9\sqrt{2} - 1)c_5s_1 + (3\sqrt{6} - \sqrt{3})s_1s_5 + 6c_5s_2 + (12\sqrt{2} + 4)s_1^3c_5 + (-4\sqrt{6} + 4\sqrt{3})s_1^3s_5 - 2\sqrt{3}s_2s_5, \quad (38)$$

$$g_{13} = (323\sqrt{6} + 1938\sqrt{3} + 328\sqrt{2} - 1968)s_5 + (969\sqrt{2} - 5814)c_5 - 7752c_3s_3s_5 + 7872\sqrt{2}s_5s_1s_2 + (-1292\sqrt{6} - 1312\sqrt{2} - 7872)s_1^2s_5 - 3876\sqrt{2}c_5s_1^2 + 7752c_5s_3^2, \quad (39)$$

$$g_{14} = (-4592\sqrt{6} - 7752)s_1^2s_3s_5 + (323\sqrt{6} + 969\sqrt{3})c_5s_3 + (-820\sqrt{6} + 969)s_3s_5 + (-969\sqrt{3} + 1968)c_3s_5 + 2907c_3c_5 - 1292\sqrt{6}c_5s_1^2s_3 + (7872\sqrt{3} + 7752\sqrt{2})s_5s_3s_2s_1 - 3936\sqrt{2}c_3s_1s_2s_5, \quad (40)$$

$$g_{15} = -323\sqrt{6} - 1938\sqrt{3} + 1148\sqrt{2} + 984 + (1292\sqrt{6} - 4592\sqrt{2} + 3936)s_1^2 - 3936s_1s_2\sqrt{2} + 3876c_3s_3 + 7752c_5s_5 + (-2584\sqrt{6} - 15504\sqrt{3} + 5248\sqrt{2})s_1^2s_3^2 + 15504\sqrt{6}s_1s_2s_3^2 + (646\sqrt{6} - 1312\sqrt{2})s_3^2, \quad (41)$$

$$g_{16} = -164\sqrt{6} + 984\sqrt{3} - 969\sqrt{2} - 1938 + (656\sqrt{6} + 3936\sqrt{3} + 3876\sqrt{2})s_1^2 - 3936\sqrt{6}s_1s_2 + 7752c_5^2 + 3876\sqrt{3}c_3s_3 + (-10496\sqrt{3} - 7752\sqrt{2} + 15504)s_1^2s_3^2 + (10496\sqrt{6} - 15504\sqrt{2})s_3^2s_2s_1 + (-2624\sqrt{3} + 1938\sqrt{2})s_3^2, \quad (42)$$

$$g_{17} = (-656\sqrt{6} - 969\sqrt{2} - 17442)c_5s_3s_1 + (-1938\sqrt{6} - 11628\sqrt{3} - 3936\sqrt{2} + 11808)s_2s_3s_5 + (-3936\sqrt{3} + 11628)c_3s_1s_5 + 5814s_6 + 1968\sqrt{6}c_3s_2s_5 + (-23256\sqrt{6} - 7752\sqrt{3} + 27552\sqrt{2} - 15744)s_1^3s_3s_5 + (2624\sqrt{6} + 3876\sqrt{2} + 23256)c_5s_1^3s_3 + (17442\sqrt{6} + 5814\sqrt{3} - 18696\sqrt{2} + 11808)s_1s_3s_5, \quad (43)$$

$$g_{18} = 41\sqrt{6} - 82\sqrt{3} - 323 + 646c_6 + (-164\sqrt{6} - 328\sqrt{3})s_1^2 + 328\sqrt{6}s_1s_2, \quad (44)$$

$$g_{19} = (323\sqrt{2} + 5814)c_5s_3s_1 + (646\sqrt{6} + 3876\sqrt{3} + 1968)s_2s_3s_5 + (-656\sqrt{3} - 3876)c_3s_1s_5 + 1938s_7 + 328\sqrt{6}c_3s_2s_5 + (7752\sqrt{6} + 2584\sqrt{3} + 4592\sqrt{2})s_1^3s_3s_5 + (-1292\sqrt{2} - 7752)c_5s_1^3s_3 + (-5814\sqrt{6} - 1938\sqrt{3} - 3116\sqrt{2})s_1s_3s_5, \quad (45)$$

$$g_{20} = (-323\sqrt{6} - 5814\sqrt{3})c_5s_3s_1 + (13776\sqrt{3} - 1938\sqrt{2} - 11628)s_2s_3s_5 + (3876\sqrt{3} - 13776)c_3s_1s_5 + 5814c_7 + 6888\sqrt{2}c_3s_2s_5 + (32144\sqrt{6} - 23256\sqrt{2} - 7752)s_1^3s_3s_5 + (1292\sqrt{6} + 7752\sqrt{3})c_5s_1^3s_3 + (-21812\sqrt{6} + 17442\sqrt{2} + 5814)s_1s_3s_5. \quad (46)$$

To determine all the values of $\theta = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7]$, which allow the robot to assume the configuration (position + orientation) given by (26), the solve command is used in MAPLE.

Performing 5191 floating-point operations (169 additions, 606 multiplications, and 552 functions), four different joint configurations are obtained, as shown in Table 2.

The set of angles designated in this section can be seen in line 2 of Table 2. The corresponding postures for the four inverse kinematic solutions are shown in Figure 6.

Adding the 3511 floating-point operations to obtain the Gröbner basis to the number of floating-point operations to

obtain the four different joint configurations from the generated Gröbner basis (5191 flop), 8702 floating-point operations are obtained.

In computing, floating-point operations per second (FLOPS, flops, or flop/s) is a measure of computer performance, useful in fields of scientific computations that require floating-point calculations. For such cases, it is a more accurate measure than measuring instructions per second. The similar term FLOP is often used for floating-point operations, for example, as a unit of counting floating-point operations carried out by an algorithm or computer hardware.

TABLE 2: Joints configurations of 7-DoF redundant manipulator obtained from Gröbner bases theory.

θ_1 (rad)	θ_2 (rad)	θ_3 (rad)	θ_4 (rad)	θ_5 (rad)	θ_6 (rad)	θ_7 (rad)
-2.617993878	-0.7853981635	0	3.141592654	1.047197551	1.047197551	-1.047197551
0.5235987758	0.7853981635	3.141592654	3.141592654	1.047197551	1.047197551	-1.047197551
0.5235987758	0.7853981635	0	3.141592654	1.047197551	-1.047197551	2.094395103
-2.617993878	-0.7853981635	3.141592654	3.141592654	1.047197551	-1.047197551	2.094395103

FIGURE 6: Postures for the four inverse kinematic solutions of the Kuka LBR iiwa 7 R800.

To confirm the computational efficiency for the solution of the inverse kinematics of 7-degrees-of-freedom serial redundant manipulators using Gröbner bases theory, a new configuration of joints will be analyzed. To meet this goal,

they are replaced arbitrarily, and the following joint values, $\theta = [(\pi/6), 0, \pi, -(\pi/6), (\pi/3), -(\pi/3), (\pi/6)]$ radians, and the nonzero parameters, $d_1 = 380$, $d_3 = 328$, $d_5 = 323$, and $d_7 = 80$, were replaced in (12), thus obtaining

$$T = \begin{bmatrix} \frac{(3/16 + \sqrt{3}/8)\sqrt{3}}{2} + \frac{3\sqrt{3}}{16} + \frac{1}{8} - \frac{3}{32} - \frac{\sqrt{3}}{16} + \frac{(3\sqrt{3}/8 + 1/4)\sqrt{3}}{2} & \frac{(-3/8 + \sqrt{3}/4)\sqrt{3}}{2} + \frac{\sqrt{3}}{8} & -40(-3/8 + \sqrt{3}/4)\sqrt{3} + \frac{363\sqrt{3}}{4} \\ \frac{(\sqrt{3}/16 - 3/8)\sqrt{3}}{2} + \frac{3}{16} - \frac{\sqrt{3}}{8} - \frac{\sqrt{3}}{32} + \frac{3}{16} + \frac{(3/8 - \sqrt{3}/4)\sqrt{3}}{2} & \frac{(-\sqrt{3}/8 - 3/4)\sqrt{3}}{2} + \frac{1}{8} & \frac{363}{4} - 40\left(-\frac{\sqrt{3}}{8} - \frac{3}{4}\right)\sqrt{3} \\ \frac{5\sqrt{3}}{16} & \frac{13}{16} & \frac{\sqrt{3}}{8} & 708 + \frac{343\sqrt{3}}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (47)$$

From this point, equations (14) to (25) are rewritten by substituting $n_x, n_y, n_z, o_x, o_y, o_z, a_x, a_y, a_z, x, y$, and z by the respective values of obtained numerical matrix, which exactly equals to matrix (47). All these 12 polynomials, added to the seven polynomials obtained from the fundamental trigonometric relation for each joint variable, are used to

generate a Gröbner basis. Precisely, as in the previous configuration, the lexicographic order $c_7 > s_7 > c_6 > s_6 > c_5 > s_5 > c_4 > s_4 > c_3 > s_3 > c_2 > s_2 > c_1 > s_1$ was also selected for this robot.

Performing 5753 floating-point operations (408 additions, 2161 multiplications, and 398 functions), we obtain the following Gröbner basis consisting of 26 polynomials:

$$g_1 = c_1^2 + s_1^2 - 1, \quad (48)$$

$$g_2 = k_1 s_1^4 s_2^2 + k_2 c_1 s_1^2 s_2 + k_3 s_1^3 s_2 + k_4 s_1^2 s_2^2 + k_5 c_1 s_1^2 s_2 + k_6 s_1^3 s_2 + k_7 s_2^2 s_1^2 + k_8 c_1 s_2 + k_9 s_1 s_2 + k_{10} s_2^2 + k_{11} c_1 s_2 + k_{12} s_1 s_2 + k_{13} s_2^2, \quad (49)$$

$$g_3 = k_{14} s_1^3 s_2^2 + k_{15} s_1^3 s_2^2 + k_{16} c_1 s_1 s_2 + k_{17} s_1^2 s_2 + k_{18} s_1 s_2^2 + k_{19} c_1 s_1 s_2 + k_{20} c_1 s_2^2 + k_{21} s_1^2 s_2 + k_{22} s_2^2 s_1 + k_{23} s_2 + k_{24} s_2, \quad (50)$$

$$g_4 = k_{25} c_1 s_2 + k_{26} s_1 s_2 + k_{27} c_1 s_2 + k_{28} s_1 s_2 + k_{29} c_2 + k_{30}, \quad (51)$$

$$g_5 = k_{31} c_1 s_1 - 2s_1^2 + 4s_3^2 - 1, \quad (52)$$

$$g_6 = k_{32} s_1^2 s_3 c_1 s_2 + k_{33} s_1^3 s_3 s_2 + k_{34} s_1^2 s_3 c_1 s_2 + k_{35} s_1^3 s_2 s_3 + k_{36} s_3 c_1 s_2 + k_{37} s_1 s_2 s_3 + k_{38} c_1 s_1 s_3 + k_{39} s_3 c_1 s_2 + k_{40} c_3 s_1^2 + k_{41} s_1 s_2 s_3 + k_{42} s_3 + k_{43} c_3, \quad (53)$$

$$g_7 = k_{44} s_1 s_3 c_1 s_2 + k_{45} s_3 s_1^2 s_2 + k_{46} c_1 s_1 s_2 s_3 + k_{47} s_3 s_1^2 s_2 + k_{48} s_3 c_1 + k_{49} c_3 s_1 + k_{50} s_2 s_3 + k_{51} c_1 c_3 + k_{52} s_1 s_3 + k_{53} s_2 s_3, \quad (54)$$

$$g_8 = k_{54} s_3 s_1^2 s_2^2 + k_{55} s_3 s_2^2 s_1^2 + k_{56} s_3 c_1 s_2 + k_{57} c_3 s_1 s_2 + k_{58} s_1 s_2 s_3 + k_{59} s_2^2 s_3 + k_{60} s_3 c_1 s_2 + k_{61} c_3 s_1 s_2 + k_{62} c_3 s_2^2 + k_{63} s_1 s_2 s_3 + k_{64} s_2^2 s_3, \quad (55)$$

$$g_9 = k_{75} c_1 s_1^2 s_2 + k_{76} c_1 s_1^2 s_2 + k_{77} c_1 s_2 + k_{78} s_1^2 + k_{79} s_1 s_2 + k_{80} c_1 s_1 + k_{81} c_1 s_2 + k_{82} c_3 s_3 + k_{83} s_1 s_2 + k_{84}, \quad (56)$$

$$g_{10} = k_{85} c_1 s_1 + 4c_3^2 + 2s_1^2 + k_{86}, \quad (57)$$

$$g_{11} = k_{87}s_1s_3c_1s_2 + k_{88}s_3s_1^2s_2 + k_{89}c_1s_1s_2s_3 + k_{90}s_3s_1^2s_2 + k_{91}c_3s_2 + k_{92}s_2s_3 + k_{93}c_1s_3 + k_{94}c_3s_1 + k_{95}c_3s_2 + k_{96}s_2s_3 + k_{97}s_4, \quad (58)$$

$$g_{12} = k_{98}c_4 + k_{99}, \quad (59)$$

$$\begin{aligned} g_{13} = & k_{100}s_1^4c_1s_2 + k_{101}c_1s_1^8s_2 + k_{102}s_1s_2 + k_{103}c_1s_2 + k_{104}c_1s_1^6s_2 + k_{105}s_1^7s_2 + k_{106}s_1^2s_5^2 + k_{107}s_1^8s_5^2 + k_{108}s_1^6s_5^2 + k_{109}s_1^2s_5^2 \\ & + k_{110}s_1^4 + k_{111}s_1^6 + k_{112}s_1^2s_5^2 + k_{113}s_1^5s_2 + k_{114}s_1^9s_2 + k_{115}s_1^6s_5^2 + k_{116}s_1^7s_2 + k_{117}s_5^2 + k_{118}s_1^8 + k_{119}s_1^6 + k_{120}s_1^2 + k_{121}s_5^2 \\ & + k_{122}c_1s_1^6s_2 + k_{123}c_1s_1s_2 + k_{124}s_1s_2 + k_{125}s_1^3s_2 + k_{126}c_1s_1^8s_2 + k_{127}s_1^2 + k_{128}c_1s_1^2s_2 + k_{129}s_1^4s_5^2 + k_{130}s_1^5s_2 + k_{131}c_1s_1^4s_2 \\ & + k_{132}c_1s_1^2s_2 + k_{133}s_1^4 + k_{134} + k_{135}s_1^3s_2 + k_{136}s_1^9s_2 + k_{137}, \end{aligned} \quad (60)$$

$$\begin{aligned} g_{14} = & k_{138}s_1 + k_{139}s_1^7s_5^2 + k_{140}s_1^5 + k_{141}s_1^7 + k_{142}s_2 + k_{143}c_1 + k_{144}s_1^4s_2 + k_{145}s_1^5s_5^2 + k_{146}s_1^6s_2 + k_{147}s_1^5s_5^2 + k_{148}s_1^7s_5^2 + k_{149}s_1^8s_2 \\ & + k_{150}s_1^7 + k_{151}s_1s_5^2 + k_{152}s_1^3s_5^2 + k_{153}s_1^7 + k_{154}c_1s_1s_2 + k_{155}c_1s_1^7s_2 + k_{156}c_1s_1^5s_2 + k_{157}s_1^3c_1s_2 + k_{158}c_1s_5^2 \\ & + k_{159}s_1s_5^2 + k_{160}s_1^2s_2 + k_{161}s_1^8s_2 + k_{162}s_2 + k_{163}s_1 + k_{164}s_1^3 + k_{165}s_5^2s_1^3 + k_{166}s_1^3 + k_{167}c_1s_1s_2 + k_{168}s_1^2s_2 + k_{169}s_1^6s_2 \\ & + k_{170}c_1s_1^7s_2 + k_{171}c_1s_1^5s_2 + k_{172}s_1^4s_2 + k_{173}s_1^3c_1s_2, \end{aligned} \quad (61)$$

$$\begin{aligned} g_{15} = & k_{174}s_1 + k_{175}s_1^7s_5^2 + k_{176}s_2s_5^2 + k_{177}s_1^5 + k_{178}s_1^7 + k_{179}s_2 + k_{180}s_1^4s_2 + k_{181}s_1^5s_5^2 + k_{182}s_1^6s_2 + k_{183}s_1^5s_5^2 + k_{184}s_1^7s_5^2 + k_{185}s_1^8s_2 \\ & + k_{186}s_1^5 + k_{187}s_1s_5^2 + k_{188}s_1^3s_5^2 + k_{189}s_1^7 + k_{190}c_1s_1s_2 + k_{191}c_1s_1^7s_2 + k_{192}c_1s_1^5s_2 + k_{193}s_1^3c_1s_2 + k_{194}s_1s_5^2 \\ & + k_{195}s_1^2s_2 + k_{196}s_1^8s_2 + k_{197}s_2 + k_{198}s_1 + k_{199}s_1^3 + k_{200}s_5^2s_1^3 + k_{201}s_1^3 + k_{202}c_1s_1s_2 + k_{203}s_1^2s_2 + k_{204}s_1^6s_2 + k_{205}c_1s_1^7s_2 \\ & + k_{206}c_1s_1^5s_2 + k_{207}s_1^4s_2 + k_{208}s_1^3c_1s_2, \end{aligned} \quad (62)$$

$$\begin{aligned} g_{16} = & k_{209}s_3c_1s_1^6s_2 + k_{210}s_3s_5^2 + k_{211}s_3 + k_{212}c_3 + k_{213}s_1s_2s_3 + k_{214}c_3s_1s_2 + k_{215}s_3s_5^2 + k_{216}s_1^2s_3 + k_{217}s_3s_1^6 + k_{218}s_3s_1^4 + k_{219}c_3s_5^2 \\ & + k_{220}s_1^2s_3s_5^2 + k_{221}s_1^2s_3 + k_{222}s_3 + k_{223}s_3s_1^6 + k_{224}c_1s_1^4s_2s_3 + k_{225}s_1^2s_3c_1s_2 + k_{226}s_1^5s_2s_3 + k_{227}s_1^4s_3s_5^2 + k_{228}s_3c_1s_2 + k_{229}s_1s_2s_3 \\ & + k_{230}s_1^3s_3s_2 + k_{231}s_3s_1^7s_2 + k_{232}s_3s_1^6s_5^2 + k_{233}c_3s_1s_2 + k_{234}s_1^5s_2s_3 + k_{235}s_3s_1^4s_5^2 + k_{236}s_3c_1s_1^6s_2 + k_{237}s_1^2s_3s_5^2 + k_{238}s_3c_1s_2 \\ & + k_{239}s_3s_1^7s_2 + k_{240}s_3s_1^6s_5^2 + k_{241}c_1s_1^4s_2s_3 + k_{242}s_1^4s_3 + k_{243}s_1^3s_2s_3 + k_{244}s_1^2s_3c_1s_2, \end{aligned} \quad (63)$$

$$\begin{aligned} g_{17} = & k_{245}c_1s_1^4s_2s_5 + k_{246}s_1^5s_2s_5 + k_{247}c_1s_1^4s_2s_5 + k_{248}s_1^5s_2s_5 + k_{249}s_1^2s_5c_1s_2 + k_{250}s_1^4s_5 + k_{251}s_1^3s_2s_5 + k_{252}c_1s_1^2s_2s_5 + k_{253}c_5s_1^4 \\ & + k_{254}s_1^3s_2s_5 + k_{255}s_5c_1s_2 + k_{256}c_5s_1^2 + k_{257}s_1^2s_5 + k_{258}s_1s_2s_5 + k_{259}c_1s_2s_5 + k_{260}c_5s_1^2 + k_{261}s_1^2s_5 + k_{262}s_1s_2s_5 + k_{263}c_5 \\ & + k_{264}s_5 + k_{265}c_5 + k_{266}s_5, \end{aligned} \quad (64)$$

$$\begin{aligned} g_{18} = & k_{267}c_1s_1^3s_2s_5 + k_{268}s_1^4s_2s_5 + k_{269}c_1s_1^3s_2s_5 + k_{270}s_1^4s_2s_5 + k_{271}c_1s_1s_2s_5 + k_{272}c_5s_1^3 + k_{273}s_1^3s_5 + k_{274}s_1^2s_2s_5 + k_{275}c_1s_1s_2s_5 \\ & + k_{276}c_5s_1^3 + k_{277}s_1^3s_5 + k_{278}s_1^2s_2s_5 + k_{279}s_5c_1 + k_{280}c_5s_1 + k_{281}s_1s_5 + k_{282}s_2s_5 + k_{283}c_1c_5 + k_{284}c_5s_1 + k_{285}s_1s_5 + k_{286}s_2s_5, \end{aligned} \quad (65)$$

$$\begin{aligned} g_{19} = & k_{287}c_1s_1^3s_2s_5 + k_{288}s_1^4s_2s_5 + k_{289}c_1s_1^3s_2s_5 + k_{290}s_1^4s_2s_5 + k_{291}c_1s_1s_2s_5 + k_{292}c_5s_1^3 + k_{293}s_1^3s_5 + k_{294}s_1^2s_2s_5 + k_{295}c_1s_1s_2s_5 \\ & + k_{296}c_5s_1^3 + k_{297}s_1^3s_5 + k_{298}s_1^2s_2s_5 + k_{299}c_5s_1 + k_{300}s_1s_5 + k_{301}s_2s_5 + k_{302}c_5s_1 + k_{303}c_5s_2 + k_{304}s_1s_5 + k_{305}s_2s_5, \end{aligned} \quad (66)$$

$$\begin{aligned}
g_{20} = & k_{306}c_1s_1^2s_2s_3s_5 + k_{307}s_1^3s_2s_3s_5 + k_{308}c_1s_1^2s_2s_3s_5 + k_{309}s_1^3s_2s_3s_5 + k_{310}s_3c_1s_2s_5 + k_{311}c_3s_1s_2s_5 + k_{312}c_5s_1^2s_3 + k_{313}s_1^2s_3s_5 \\
& + k_{314}s_1s_2s_3s_5 + k_{315}s_3c_1s_2s_5 + k_{316}c_3s_1s_2s_5 + k_{317}c_5s_1^2s_3 + k_{318}s_1^2s_3s_5 + k_{319}s_1s_2s_3s_5 + k_{320}c_3s_5 + k_{321}c_5s_3 + k_{322}s_3s_5 \\
& + k_{323}c_3c_5 + k_{324}c_5s_3 + k_{325}s_3s_5,
\end{aligned} \tag{67}$$

$$\begin{aligned}
g_{21} = & k_{326}s_1^4c_1s_2 + k_{327}s_1s_2 + k_{328}c_1s_2 + k_{329}c_1s_1^6s_2 + k_{330}s_1^7s_2 + k_{331}s_1^2s_5^2 + k_{332}s_1^6s_5^2 + k_{333}s_1^2s_5^2 + k_{334}s_1^4 + k_{335}s_1^6 + k_{336}s_1^4s_5^2 \\
& + k_{337}s_1^5s_2 + k_{338}s_1^6s_5^2 + k_{339}s_1^7s_2 + k_{340}s_5^2 + k_{341}s_1^6 + k_{342}s_1^2 + k_{343}s_5^2 + k_{344}c_5s_5 + k_{345}c_1s_1^6s_2 + k_{346}c_1s_2 + k_{347}s_1s_2 \\
& + k_{348}s_1^3s_2 + k_{349}s_1^2 + k_{350}c_1s_1^2s_2 + k_{351}s_1^4s_5^2 + k_{352}s_1^5s_2 + k_{353}c_1s_1^4s_2 + k_{354}c_1s_1^2s_2 + k_{355}s_1^4 + k_{356} + k_{357}s_1^3s_2 + k_{358},
\end{aligned} \tag{68}$$

$$g_{22} = c_5^2 + s_5^2 - 1, \tag{69}$$

$$\begin{aligned}
g_{23} = & k_{359}c_1s_1^3s_2s_3s_5 + k_{360}s_1^4s_2s_3s_5 + k_{361}c_1s_1^3s_2s_3s_5 + k_{362}s_1^4s_2s_3s_5 + k_{363}c_1s_1s_2s_3s_5 + k_{364}c_5s_1^3s_3 + k_{365}s_1^3s_3s_5 + k_{366}s_1^2s_2s_3s_5 \\
& + k_{367}c_1s_1s_2s_3s_5 + k_{368}c_5s_1^3s_3 + k_{369}s_1^3s_3s_5 + k_{370}s_1^2s_2s_3s_5 + k_{371}c_3s_2s_5 + k_{372}c_5s_1s_3 + k_{373}s_1s_3s_5 + k_{374}s_2s_3s_5 + k_{375}c_1s_3s_5 \\
& + k_{376}c_3s_1s_5 + k_{377}c_3s_2s_5 + k_{378}c_5s_1s_3 + k_{379}s_1s_3s_5 + k_{380}s_2s_3s_5 + k_{381}s_6,
\end{aligned} \tag{70}$$

$$g_{24} = k_{382}c_1s_2 + k_{383}s_1s_2 + k_{384}c_1s_2 + k_{385}s_1s_2 + k_{386}c_6 + k_{387}, \tag{71}$$

$$\begin{aligned}
g_{25} = & k_{388}c_1s_1^3s_2s_3s_5 + k_{389}s_1^4s_2s_3s_5 + k_{390}c_1s_1^3s_2s_3s_5 + k_{391}s_1^4s_2s_3s_5 + k_{392}c_1s_1s_2s_3s_5 + k_{393}c_5s_1^3s_3 + k_{394}s_1^3s_3s_5 + k_{395}s_1^2s_2s_3s_5 \\
& + k_{396}c_1s_1s_2s_3s_5 + k_{397}c_5s_1^3s_3 + k_{398}s_1^3s_3s_5 + k_{399}s_1^2s_2s_3s_5 + k_{400}c_1s_3s_5 + k_{401}c_3s_1s_5 + k_{402}c_3s_2s_5 + k_{403}c_5s_1s_3 + k_{404}s_1s_3s_5 \\
& + k_{405}s_2s_3s_5 + k_{406}c_3s_2s_5 + k_{407}c_5s_1s_3 + k_{408}s_1s_3s_5 + k_{409}s_2s_3s_5 + k_{410}s_7,
\end{aligned} \tag{72}$$

$$\begin{aligned}
g_{26} = & k_{411}c_1s_1^3s_2s_3s_5 + k_{412}s_1^4s_2s_3s_5 + k_{413}c_1s_1^3s_2s_3s_5 + k_{414}s_1^4s_2s_3s_5 + k_{415}c_1s_1s_2s_3s_5 + k_{416}c_5s_1^3s_3 + k_{417}s_1^3s_3s_5 + k_{418}s_1^2s_2s_3s_5 \\
& + k_{419}c_1s_1s_2s_3s_5 + k_{420}c_5s_1^3s_3 + k_{421}s_1^3s_3s_5 + k_{422}s_1^2s_2s_3s_5 + k_{423}c_3s_2s_5 + k_{424}c_5s_1s_3 + k_{425}s_1s_3s_5 + k_{426}s_2s_3s_5 + k_{427}c_1s_3s_5 \\
& + k_{428}c_3s_1s_5 + k_{429}c_3s_2s_5 + k_{430}c_5s_1s_3 + k_{431}s_1s_3s_5 + k_{432}s_2s_3s_5 + k_{433}c_7.
\end{aligned} \tag{73}$$

The 433 numerical coefficients k_i can be up to 92 digits, which is why they are not shown in this study. To determine all the values of $\theta = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7]$, which allow the robot to assume the configuration (position + orientation) given by (47), the solve command is used in MAPLE.

Performing 14089 floating-point operations (408 additions, 2161 multiplications, and 1440 functions), twelve different joint configurations are obtained, as can be seen in Table 3.

The set of angles designated can be seen in line 5 of Table 3. The 12th joint configuration does not represent a real solution; thus, there are eleven possible configurations. The corresponding postures for the eleven inverse kinematic solutions are shown in Figure 7.

Adding the 5753 floating-point operations to obtain the Gröbner basis to the number of floating-point operations to obtain the eleven different joint configurations from the generated Gröbner basis (14089 flop), 19842 floating-point operations are obtained.

One notes that the solve command is not used directly to solve the inverse kinematics problem, since it does not involve the Gröbner bases theory in its algorithms. Besides that, the solution is obtained not only by computing the Gröbner basis but also by applying the solve command. For instance, the variables s_k and c_k were assigned for $\sin(\theta_k)$ and for $\cos(\theta_k)$, new polynomials were included in the computation of the basis (related to the fundamental trigonometric relation), and a monomial order was chosen.

TABLE 3: Joints configurations of 7-DoF redundant manipulator obtained from Gröbner bases theory (2nd case).

θ_1 (rad)	θ_2 (rad)	θ_3 (rad)	θ_4 (rad)	θ_5 (rad)	θ_6 (rad)	θ_7 (rad)
2.094395103	0	-1.570796327	0.5235987757	1.047197551	1.047197551	-2.617993878
2.094395103	0	1.570796327	-0.5235987757	1.047197551	-1.047197551	0.5235987755
0.5235987755	0	0	0.5235987758	1.047197551	1.047197551	-2.617993878
-0.5235987755	0	1.047197551	0.5235987758	1.047197551	1.047197551	-2.617993878
0.5235987755	0	3.141592654	-0.5235987758	1.047197551	-1.047197551	0.5235987758
-0.5235987755	0	-2.094395102	-0.5235987758	1.047197551	-1.047197551	0.5235987758
0.1901256033	0	0.3334731723	0.5235987757	1.047197551	1.047197551	-2.617993878
2.094395103	0	-1.570796327	0.5235987757	1.047197551	1.047197551	-2.617993878
2.094395103	0	1.570796327	-0.5235987757	1.047197551	-1.047197551	0.5235987755
0	0	0.5235987757	0.5235987758	1.047197551	1.047197551	-2.617993878
0	0	-2.617993878	-0.5235987758	1.047197551	-1.047197551	0.5235987758
(1.047197537 + 2.035571560i)	0	(2.617993862 - 2.035571554i)	(-0.5235988062 + 4.966308677 $\times 10^{-9}$ i)	1.047197551	(-1.047197582 + 4.931548679 $\times 10^{-9}$ i)	(0.5235987148 + 4.66896827 $\times 10^{-8}$ i)

FIGURE 7: Postures for the 11 inverse kinematic solutions of the Kuka LBR iiwa 7 R800.

5. Conclusions

When using the Gröbner bases theory for the solution of inverse kinematics of 7-degrees-of-freedom serial redundant manipulators, one notes in the tests performed during this research that it is computationally efficient since the equations produced for determination of seven variables are mathematically simpler to be solved. Since through retro-active substitutions, all solutions can be determined to position and orient the robotic manipulator end effector in space.

The original equations that model the inverse kinematics of these manipulators cannot be solved at once using the solve command in MAPLE. Hence, in order to produce a new set of equations, Gröbner bases theory is studied. The choice of the monomial order in the Gröbner bases enables the solve command to compute the solutions of this problem with these new equations.

In the future work, it is intended to demonstrate that this method can also be computationally efficient when compared to other methods for solving the inverse kinematic problem of other classes of robotic manipulators, e.g., hyper-redundant robots.

All solutions are confirmed with the aid of MAPLE, confirming that this method is accurate and easy to implement.

Data Availability

The MAPLE source code data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior-Brasil (CAPES) [finance code 001].

References

- [1] J. Denavit and R. S. Hartenberg, *Description and Displacement Analysis of Mechanics Based on the 2×2 Dual Matrices*, University of Northwestern, Evanston, IL, USA, 1955.
- [2] R. S. Hartenberg and J. Denavit, *Kinematic Synthesis of Linkages*, McGraw-Hill, New York, NY, USA, 1965.
- [3] R. P. Paul, *Robot Manipulators: Mathematics, Programming, and Control (Artificial Intelligence)*, MIT Press, Cambridge, MA, USA, 1981.

- [4] L. Sciavicco and B. Siciliano, *Modeling and Control of Robot Manipulator*, McGraw-Hill, New York, NY, USA, 1996.
- [5] B. Buchberger, "Bruno Buchberger's PhD thesis 1965: an algorithm for finding the basis elements of the residue class ring of a zero dimensional polynomial ideal," *Journal of Symbolic Computation*, vol. 41, no. 3-4, pp. 475–511, 2006.
- [6] S. R. X. da Silva, L. Schnitman, and V. C. Filho, "Analysis of computational efficiency for the solution of inverse kinematics problem of anthropomorphic robots using Gröbner bases theory," *International Journal of Advanced Robotic Systems*, vol. 16, 2021.
- [7] D. Tolani, A. Goswami, and N. I. Badler, "Real-time inverse kinematics techniques for anthropomorphic limbs," *Graphical Models*, vol. 62, no. 5, pp. 353–388, 2000.
- [8] I. Kuhlemann, A. Schweikard, P. Jaue, and F. Ernst, "Robust inverse kinematics by configuration control for redundant manipulators with seven DoF," in *Proceedings of the International Conference on Control, Automation and Robotics*, IEEE, Hong Kong, China, December 2016.
- [9] D. Cox and J. Little, D. O'Shea, *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra*, Springer, New York, NY, USA, 4th edition, 2015.
- [10] B. Sturmfels, "What is . . . a Gröbner basis?" *Notices of the American Mathematical Society*, vol. 52, no. 10, pp. 1199–1200, 2005.
- [11] S. Stifter, "Algebraic methods for computing inverse kinematics," *Journal of Intelligent & Robotic Systems*, vol. 11, no. 1-2, pp. 79–89, 1994.
- [12] J.-C. Faugère, "A new efficient algorithm for computing Gröbner bases (F4)," *Journal of Pure and Applied Algebra*, vol. 139, no. 1–3, pp. 61–88, 1999.
- [13] C. G. Lee, "Robot arm kinematics, dynamics, and control," *Computer*, vol. 15, no. 12, pp. 62–80, 1982.
- [14] M. Vidyasagar and M. W. Spong, *Robot Dynamics and Control*, Wiley India Pvt. Limited, Bengaluru, India, 2008.
- [15] L. W. Tsai, *Robot Analysis: The Mechanics of Serial and Parallel Manipulators*, John Wiley & Sons, New York, NY, USA, 1999.
- [16] A. Ghosal, *Robotics: Fundamental Concepts and Analysis*, Oxford University Press, New Delhi, India, 2006.
- [17] J. J. Craig, *Introduction to Robotics: Mechanics and Control*, Addison-Wesley, Boston, MA, USA, 4th edition, 2017.