IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 8 December 2022, accepted 27 December 2022, date of publication 5 January 2023,
date of current version 10 January 2023.

*Digital Object Identifier 10.1109/ACCESS.2023.3234315*

RESEARCH ARTICLE

# A Novel General Inverse Kinematics Optimization-Based Solution for Legged Robots in Dynamic Walking by a Heuristic Approach

**JACOBO TORRES-FIGUEROA[1], EDGAR A. PORTILLA-FLORES[2], JOSÉ A. VÁSQUEZ-SANTACRUZ[3], EDUARDO VEGA-ALVARADO[1], AND LUIS F. MARÍN-URÍAS[3]**

[1]Centro de Inovación y Desarrollo Técnologico en Cómputo (CIDETEC), Group of Research and Innovation in Mechatronics (GRIM), Instituto Politécnico Nacional, Ciudad de Mexico 07700, Mexico
[2]Instituto Politécnico Nacional, UPIIT, Tlaxcala 90000, Mexico
[3]FIEE, Universidad Veracruzana, Veracruz 94294, Mexico

Corresponding author: Edgar A. Portilla-Flores (aportilla@ipn.mx)

This work was supported in part by the Instituto Politécnico Nacional of México via Secretaría de Investigación y Posgrado (SIP) under Project 20221928 and Project 20221960, and in part by the Comisión de Operación y Fomento de Actividades Académicas (COFAA). The work of Jacobo Torres-Figueroa was supported by Consejo Nacional de Ciencia y Tecnología (CONACYT) of México through the Doctorate Grant.

**ABSTRACT** This work presents a new optimization-based solution to the Inverse Kinematics Problem (IKP) of legged robots, including a modified Walking Pattern Generator that automatically avoids singularity configurations regarding position. The approach uses a numeric constrained problem solved with heuristic algorithms, where the joint vector is calculated to minimize the position errors, while orientation errors are handled as equality constraints, and threshold values are used to set the maximum error for each foot along the trajectories. The optimization model can generalize the IKP for robots with any number of legs and any number of poses within the physically consistent trajectories for the Center of Mass and the feet, considering the dynamics of the robot for a stable walking. The case study was a 12-DOF biped robot, and the resulting joint trajectories were validated by a dynamic simulation, using a Gazebo-ROS platform, where the walking was successfully performed without requiring a feedback control for correcting the torso tilt, showing the solution quality.

**INDEX TERMS** Inverse kinematics, legged robots, optimization, metaheuristics, dynamic walking.

## I. INTRODUCTION

The solution of the Inverse Kinematics Problem (IKP) is mandatory when it is required to control the position of an articulated robot. For this reason, diverse analytical and numerical methods have been developed for solving it [1]. Analytical methods require to derive closed equations, by means of algebraic and geometric techniques. However, since these equations are highly non-linear, finding a solution is a non-trivial task [2]. In some cases, the mechanical structure of the robots makes it very difficult to obtain the closed equations. On the other side, numerical methods can generate multiple solutions for the IKP of complex robots, but their main drawback is the computational cost, because of factors such as singularity neighborhoods, Jacobian calculations, and initial conditions, among others.

The solution approaches based on optimization consider the IKP as a non-linear numeric optimization problem that is solved by using search algorithms such as gradient-based mathematical programming or the heuristic techniques, which have shown a good performance in the solution of the IKP for manipulator robots. In [3], an improvement to the Particle Swarm Optimization (PSO) algorithm is presented for solving the IKP of a 6-DOF robot. The objective function (OF) is defined considering the position and orientation errors, and multiple swarms are implemented to avoid local minimum. Similarly, a PSO version was developed in [4] for two manipulators, with an OF that includes both position and orientation. The T-IK algorithm based in NSGA-II was implemented in [5], considering a population

The associate editor coordinating the review of this manuscript and approving it for publication was Guilin Yang.

migration strategy to solve the IKP of an 8-DOF manipulator. In [6], the Differential Evolution (DE) algorithm is applied to the solution of the IKP of a redundant manipulator robot, to find optimal configurations while considering the range of the robot joints.

A comparative study of diverse metaheuristics including DE, Artificial Bee Colony (ABC), and Adaptive Differential Evolution With Optional External Archive (JADE), among others, was developed in [7] for solving the IKP of a 5-DOF manipulator. Another comparative between the Elitist Non-Dominated Sorting Genetic Algorithm NSGA-II and the Bacterial Chemotaxis Multi-Objective Algorithm (BCMOA) is presented in [8] for an industrial robot. In [9], the Chaos-based Vortex Search algorithm was implemented to solve the IKP, considering the trajectory tracking of a 6-DOF serial robot manipulator with an offset wrist, using an OF for minimizing both the end effector position error and the joint displacement. A modified version of the Gray Wolf Optimization (GWO) algorithm is presented in [10], named Fast Parabolic Descending GWO, to solve the IKP for a 7-DOF robot manipulator, only considering the position error. The algorithm showed a better performance compared to the original GWO, avoiding getting stuck in local optima. Similarly, in [11], the Artificial Bee Colony algorithm was implemented to solve the IKP considering 100 different points, using the same 7-DOF robot. The results were compared with PSO, showing that ABC produces effective results. As can be seen from these works, the use of metaheuristics is a clear trend for solving complex engineering problems as is the IKP.

The use of metaheuristics has not been limited to optimization problems in robotics. For example, in [12], the authors used the binary Artificial Bee Colony algorithm to ensure the correct distribution of negative literals in the Discrete Hopfield Neural Network logical structure, using a new variant of satisfiability logical rule called Weighted Random 2 Satisfiability. Similarly, a non-systematic Satisfiability model is proposed in [13] to avoid overfitting global minima solution in a Discrete Hopfield Neural Network. The authors formulated a Weighted Random k Satisfiability, considering in-the-logic randomized literals. A logic phase was introduced to optimally generate its right structure by using the Genetic Algorithm (GA), with respect to the desired ratio of negative literals. A comparative study was carried out, including other metaheuristics, where GA showed a better performance in both the logic and training phases.

The solution of the IKP for legged robots presents a higher difficult grade even for numerical methods, since the hyper redundancy in those robots make singularities highly probable because of the multiple kinematic chains and/or the use of more than one end effector. Recent proposals, as in [8] and [14] try to solve the IKP of multiple robots with different morphologies, including legged robots. The solutions are generally obtained from static scenarios [15] or for animation purposes only [16], without considering dynamic constraints. However, most of the heuristic-based works reported in literature for solving the IKP of legged robots do not consider crucial stability constraints to get dynamic walking of the real robots, where inertial and dynamic parameters are extremely important. To fill this gap, in this work, a new optimization-based approach for solving the IKP of legged robots is proposed, based on a numeric constrained optimization approach. It can be applied not only to a single static point in the robot workspace, but to every pose of the time-variable physically consistent trajectories for the Center of Mass (CoM) and the feet, considering the robot dynamic behavior to ensure a stable dynamic walking. This approach automatically avoids singularity configurations on position, and threshold values can be used to set the maximum orientation error for each end effector (foot) along the trajectories, making it possible to modify the error precision as required.

The main contributions of this work are:

- The proposal of a general Walking Pattern Generator for legged robots, which is capable of generating the joint trajectories by considering the physically consistent trajectories to develop dynamic walking, avoiding the use of the closed equations to solve the IKP in legged robots.
- A novel general optimization approach to solve the IKP in legged robots, defined by a new optimization model which consists in a constrained optimization problem, valid for every pose of the desired trajectories.
- The solution of the optimization problem by means of heuristic algorithms with constraint handling capabilities, only requiring the DKM of the legged robot.

This paper is organized as follows: Section II presents the proposed approach for solving the IKP of legged robots, whereas Section III describes the case study. In Section IV, the experimental implementation and its results are analyzed, and in Section V the final considerations are presented, including both conclusions and future work.

## II. PROPOSED SOLUTION APPROACH

In this section, the IKP of legged robots is first presented in a general way, considering each foot of the robot as an end effector. Then, the Walking Pattern Generator (WPG) required to obtain the joint trajectories is described, taking into account the dynamic constraints to develop the walking. These constraints correspond to the physically consistent trajectories of the CoM and the feet. Subsequently, a new solution approach based on a constraint numerical optimization is proposed, considering both the OF and the constraints to solve the IKP as a part of a heuristic-based IK-solver integrated in the WPG. Finally, the optimization model and its solution are presented, applying heuristic algorithms in combination with a constraint handler.

### A. INITIAL CONSIDERATIONS

The mechanical structure of a legged robot can be described by its kinematic chains made up of the links that transmit movement through the kinematic pairs associated with the robot joints. This is parameterized by a $n$-dimensional vector

$\theta$ expressed in (1), where $n$ is the number of kinematic pairs,

$$\theta = [\theta_1, \theta_2, \ldots, \theta_{n-1}, \theta_n] \tag{1}$$

Each foot of the robot is considered as an end effector of those chains, and its position and orientation are described by $A^c$. The set of all possible poses avoiding any robot singularity is known as the reachable workspace. The poses that the end effector can achieve with arbitrary orientations are a subset of the reachable workspace, and it is called the dexterous workspace [17].

Two methods, Direct Kinematics (DK) and Inverse Kinematics (IK), are applied to solve the Direct Kinematic Problem (DKP) and the IKP by means of the equations that make up the Direct Kinematic Model (DKM) and the Inverse Kinematic Model (IKM), respectively. The solution of the DKP consists on determining the position and orientation of the end effector $A^c$ by terms of its dexterous workspace, for a value set $\theta$ defined in the joint space, as expressed in (2). The DKP can be solved using recursive algorithms to multiply the transformation matrices $T$ which characterize the relative movements and translations between links.

$$A^c = \mathbf{DKM}(\theta) \tag{2}$$

On the other hand, the IKP consists in determining the set of joint variables $\theta$ expressed in (3) that correspond to a desired pose $A^d$ of the end effector, defined in the dexterous workspace:

$$\theta = \mathbf{IKM}(A^d) \tag{3}$$

The complexity of the IKP of legged robots increases because of their multiple kinematic chains and end effectors, as indicated in (4), where $r$ is the number of end effectors,

$$\theta = \mathbf{IKM}(A_1^d, A_2^d, \ldots, A_{r-1}^d, A_r^d) \tag{4}$$

The use of closed equations to solve the IKP of legged robots is not the best option, since those expressions are related to the mechanical characteristics of the specific robot, so the method can not be generalized. On the other side, Jacobian-based numerical methods can present failures when handling configurations with singularities. For that reason, an approach based on modeling the IKP as a general Constrained Numerical Optimization Problem (CNOP) is proposed in this work. The CNOP is solved by means of heuristic algorithms, with an approximation of the IKM using only the DKM, for tracking trajectories defined in the dexterous workspace. The solution considers both the position and the orientation, which allow to perform dynamic walking in robots whose legs are constituted by serial open kinematic chains and rotational joints.

Diverse criteria have been considered to develop stable walking [18]. A typical condition requires a joint performance where the CoM of the robot describes a trajectory into a stable configuration with projection on the walking surface within the support polygon [19]. The reference trajectories for that performance are derived from a Walking Pattern Generator (WPG), as depicted in Fig. 1. It includes three stages: a generator for the CoM trajectory $C_t$ to ensure a dynamically stable walking; a generator of the trajectories $P_i(t)$ for the feet, where $i = 1$ to $r$; and the block that solves the IKP using both $C(t)$ and $P_i(t)$.

As can be seen in Fig. 1, $C(t)$ and $P_i(t)$ are described in terms of the workspace or inertial frame, since the information that encodes the foot placements as the input for the WPG is established in that system. For solving the IKP, it is necessary to transform these trajectories according to the reference system of the CoM, since the IK of each end effector should be calculated with respect to the free floating base of the robot. Therefore, it is necessary to include a transformation step within the block of the Inverse Kinematics.

If $C(t)$ and $P_i(t)$ are discretized using a uniform sample time $S$ with $m$ samples along the trajectory, then matrices $C^d \in \mathbb{R}^{3\times m}$ and $P_i^d \in \mathbb{R}^{3\times m}$ can be defined by (5) and (6), respectively, where $i = 1, .., r$ and $j = 1, .., m$.

$$C^d = \begin{bmatrix} c_{x_1}^d, & c_{x_2}^d, & \ldots, & c_{x_{j-1}}^d, & c_{x_j}^d \\ c_{y_1}^d, & c_{y_2}^d, & \ldots, & c_{y_{j-1}}^d, & c_{y_j}^d \\ c_{z_1}^d, & c_{z_2}^d, & \ldots, & c_{z_{j-1}}^d, & c_{z_j}^d \end{bmatrix} \tag{5}$$

$$P_i^d = \begin{bmatrix} p_{i,x_1}^d, & p_{i,x_2}^d, & \ldots, & p_{i,x_{j-1}}^d, & p_{i,x_j}^d \\ p_{i,y_1}^d, & p_{i,y_2}^d, & \ldots, & p_{i,y_{j-1}}^d, & p_{i,y_j}^d \\ p_{i,z_1}^d, & p_{i,z_2}^d, & \ldots, & p_{i,z_{j-1}}^d, & p_{i,z_j}^d \end{bmatrix} \tag{6}$$

The orientation of the CoM and the feet depends on the rotation matrices $\mathbf{Rot}_{CoM}^d(x_\gamma, y_\beta, z_\alpha)$ and $\mathbf{Rot}_i^d(x_\gamma, y_\beta, z_\alpha)$ respectively, from the roll ($\gamma$), pitch ($\beta$) and yaw ($\alpha$) rotation angles [2]. In this proposal, the global pose at a specific time is obtained from the global transformation multi-matrices $A_{CoM}^d$ and $A_i^d$, corresponding to expressions (7) and (8), respectively. This matrix representation contains all the information of the trajectories defined in the dexterous workspace, allowing the synchronization of the trajectory tracking of all the end effectors by means of the $j$ index. Fig. 2 presents a graphic representation of the multi-matrices $A_{CoM}^d$ and $A_i^d$, where the information of each element of the trajectories includes its Cartesian coordinate system (frame), in order to define the desired pose for each end effector (foot attitude) with respect to the inertial frame, in this case the dexterous workspace of the robot.

$$\begin{aligned} A_{CoM}^d &= \begin{bmatrix} \mathbf{Rot}_{CoM,j}^d(x_\gamma, y_\beta, z_\alpha) & C_j^d \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} n_{x_j}^d & o_{x_j}^d & a_{x_j}^d & c_{x_j}^d \\ n_{y_j}^d & o_{y_j}^d & a_{y_j}^d & c_{y_j}^d \\ n_{z_j}^d & o_{z_j}^d & a_{z_j}^d & c_{z_j}^d \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ A_i^d &= \begin{bmatrix} \mathbf{Rot}_{i,j}^d(x_\gamma, y_\beta, z_\alpha) & P_{i,j}^d \\ 0 & 1 \end{bmatrix} \end{aligned} \tag{7}$$

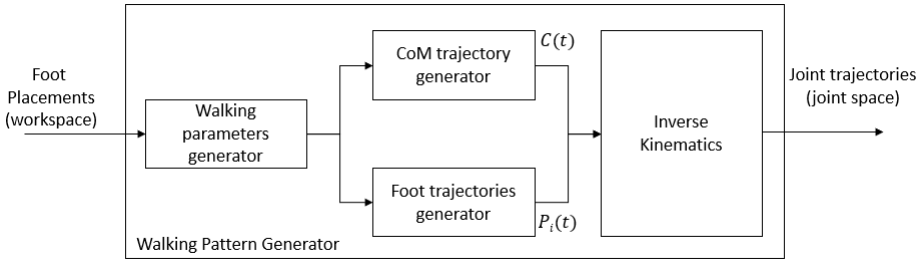


**FIGURE 1.** Structure for a walking pattern generator.

$$= \begin{bmatrix} n^d_{i,x_j} & o^d_{i,x_j} & a^d_{i,x_j} & p^d_{i,x_j} \\ n^d_{i,y_j} & o^d_{i,y_j} & a^d_{i,y_j} & p^d_{i,y_j} \\ n^d_{i,z_j} & o^d_{i,z_j} & a^d_{i,z_j} & p^d_{i,z_j} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{8}$$

By using (7) and (8), it is possible to establish a solution for the IKP along the trajectories, applying heuristic algorithms and an appropriate objective function to approximate the IKM using only the DKM, obtaining the pose $A^c_i$ of each foot of the legged robot. It depends on the joint vector $\theta$, as indicated in (9) and (10), with $j = 0, \ldots, m$.

$$\mathbf{Rot}^c_i(\theta) = \begin{bmatrix} n^c_{i,x_j} & o^c_{i,x_j} & a^c_{i,x_j} \\ n^c_{i,y_j} & o^c_{i,y_j} & a^c_{i,y_j} \\ n^c_{i,z_j} & o^c_{i,z_j} & a^c_{i,z_j} \end{bmatrix} \tag{9}$$

$$A^c_i(\theta) = \begin{bmatrix} \mathbf{Rot}^c_{i,j}(\theta) & P^c_{i,j}(\theta) \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} n^c_{i,x_j} & o^c_{i,x_j} & a^c_{i,x_j} & p^c_{i,x_j} \\ n^c_{i,y_j} & o^c_{i,y_j} & a^c_{i,y_j} & p^c_{i,y_j} \\ n^c_{i,z_j} & o^c_{i,z_j} & a^c_{i,z_j} & p^c_{i,z_j} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{10}$$

In most of the heuristic based developments in the related literature, the IKP of legged robots has been solved for static configurations without considering physically consistent trajectories, commonly implementing an approach where the OF is based on the sum of the position and orientation errors of the end effectors [20]. It is indicated in (11), where $\theta = [\theta_1, \theta_2, \ldots, \theta_{n-1}, \theta_n]$ is the joint vector, $n$ is the number of degrees of freedom, $r$ is the number of end effectors, and $\omega_i$ are the weights for a pondered sum.

$$f(\theta) = \sqrt{\frac{1}{r}\sum_{i=1}^{r} \omega_i L_i^2(\theta)} \tag{11}$$

The term $L(\theta)$ in the objective function is defined as expressed in (12), where $E^p_i(\theta)$ and $E^o_i(\theta)$ are the position and orientation errors. Since these errors have different units the normalization constant $\phi > 0$ must be included [1]. Additionally, the solution can fall into local optima because the possible difference in magnitude between the errors of the end effectors is not considered [15]. In order to avoid this problem, it has been proposed the use of randomly generated constants $\omega_i$ [20].

$$L(\theta) = \phi E^p_i(\theta) + E^o_i(\theta) \tag{12}$$

### B. OPTIMIZATION APPROACH

In this development, a new optimization approach is proposed for solving the IKP of legged robots not only for a point in the robotic workspace, but also considering each pose of the physically consistent trajectories for the CoM and the feet, described in the dexterous workspace, to ensure a stable dynamic walking. One of the advantages of this approach is that none randomly generated $\omega_i$ values are required to avoid bias while searching the solution neither normalizing the OF. Only the position errors $E^p_i(\theta)$ of all the end effectors are considered in the OF, whereas the orientation errors $E^o_i(\theta)$ are handled as equality constraints to guarantee IK compliance.

The proposed OF is expressed in (13) for $j = 0$ to $m$, where $P^d_{i,j}$ and $P^c_{i,j}$ are the desired and current $j$-th position for each foot $i$ along a trajectory, respectively, with $P^c_{i,j}$ calculated from the DKM.

$$f(\mathbf{q}) = \sum_{i=1}^{r} \| P^d_{i,j} - P^c_{i,j}(\mathbf{q}) \| \tag{13}$$

Additionally, the errors $E^o_i(\mathbf{q})$ are defined by the subtraction of the rotation matrices $\mathbf{Rot}^d_{i,j}(x_\gamma, y_\beta, z_\alpha)$ and $\mathbf{Rot}^c_{i,j}(\mathbf{q})$, as expressed in (14), where $\mathbf{q}$ is the joint vector including $n$ elements.

$$E^o_i(\mathbf{q}) = \mathbf{abs}(\mathbf{Rot}^d_{i,j}(x_\gamma, y_\beta, z_\alpha) - \mathbf{Rot}^c_{i,j}(\mathbf{q})) \tag{14}$$

It is possible to obtain the $n$ joint trajectories required to execute the stable dynamic walking, by solving the $m$ restricted numerical optimization problems for each pose defined within the trajectories $A^d_{CoM}$ and $A^d_i$, applying (13). Therefore, the angles that describe the joint space of the robot conform the vector of design variables, indicated in (15).

$$\mathbf{q} = [q_1, q_2, \ldots, q_{n-1}, q_n] = [\theta_1, \theta_2, \ldots, \theta_{n-1}, \theta_n] \tag{15}$$

The proposed WPG is shown in Fig. 3. As can be seen from the internal conformation of the heuristic-based IK solver, it includes a sampler/transformer element to translate the CoM and feet trajectories to the floating frame in the torso link [19]. It also contains both a heuristic algorithm that

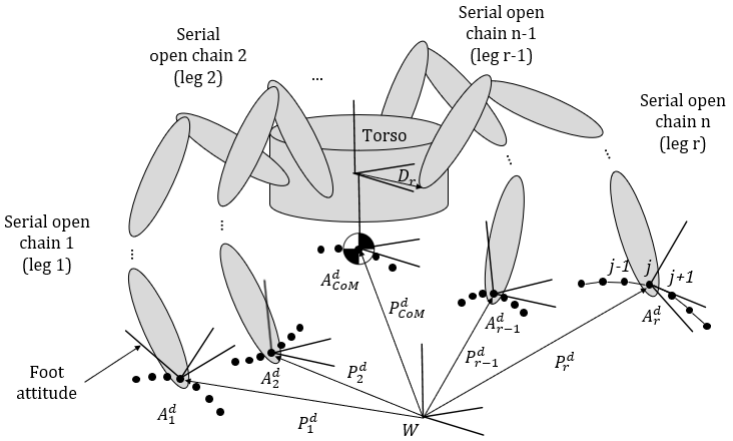


**FIGURE 2.** Global transformation multi-matrices $A^d_{CoM}$ and $A^d_i$ for a general legged robot.

calculates the joint vector $\mathbf{q}_i$ along the trajectories by using information from the DKM, and a constraint handler for the equality constraints required to meet the IK solution.

### C. OPTIMIZATION MODEL

In general, a constrained optimization problem can be described with expressions (16)-(18) [21], where $\mathbf{x}$ is the vector of design variables, $f(\mathbf{x})$ is the OF, $h_u(\mathbf{x})$ is the $u$-th equality constraint, and $g_v(\mathbf{x})$ is the $v$-th inequality constraint:

$$\min_{\mathbf{x}\in\mathbb{R}^n} \quad f(\mathbf{x}) \tag{16}$$

subject to:

$$h_u(\mathbf{x}) = 0, \quad u < n \tag{17}$$

$$g_v(\mathbf{x}) \leq 0 \tag{18}$$

The optimization approach proposed to solved the IKP by using the DKM for each defined pose within the trajectories of the legged robot is described by expressions (19)-(26). As can be seen in (19), the objective function considers the position error of each foot, and so it has to be minimized being zero the ideal optimal value.

$$\begin{aligned} \min_{\mathbf{q}\in\mathbb{R}^n} \quad f(\mathbf{q}) &= \sum_{i=1}^{r} \|P^d_{i,j}(\mathbf{q}) - P^c_{i,j}(\mathbf{q})\| \\ &\forall j \in \mathbb{Z}, 1 \leq j \leq m, \\ &\forall i \in \mathbb{Z}, 1 \leq i \leq r \end{aligned} \tag{19}$$

subject to:

$$\begin{aligned} h_i = &\mid n^d_{i,x_j} - n^c_{i,x_j} \mid + \mid n^d_{i,y_j} - n^c_{i,y_j} \mid + \mid n^d_{i,z_j} - n^c_{i,z_j} \mid \\ &+ \mid o^d_{i,x_j} - o^c_{i,x_j} \mid + \mid o^d_{i,y_j} - o^c_{i,y_j} \mid + \mid o^d_{i,z_j} - o^c_{i,z_j} \mid \\ &+ \mid a^d_{i,x_j} - a^c_{i,x_j} \mid + \mid a^d_{i,y_j} - a^c_{i,y_j} \mid \\ &+ \mid a^d_{i,z_j} - a^c_{i,z_j} \mid = 0 \end{aligned} \tag{20}$$

$$\begin{aligned} A^c_i(\mathbf{q}) &= \begin{bmatrix} \mathbf{Rot}^c_{i,j}(\mathbf{q}) & P^c_{i,j}(\mathbf{q}) \\ 0 & 1 \end{bmatrix} = {}^0T_{l_i} \\ &= {}^0T_{l_{i-1}+1}\, {}^{l_{i-1}+1}T_{l_{i-1}+2} \ \ldots\ {}^{l_i-2}T_{l_i-1}\, {}^{l_i-1}T_{l_i} \\ &= \begin{bmatrix} n^c_{i,x_j} & o^c_{i,x_j} & a^c_{i,x_j} & p^c_{i,x_j} \\ n^c_{i,y_j} & o^c_{i,y_j} & a^c_{i,y_j} & p^c_{i,y_j} \\ n^c_{i,z_j} & o^c_{i,z_j} & a^c_{i,z_j} & p^c_{i,z_j} \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{21}$$

$$q_{1_{min}} \leq q_1 \leq q_{1_{max}} \tag{22}$$

$$q_{2_{min}} \leq q_2 \leq q_{2_{max}} \tag{23}$$

$$\vdots$$

$$q_{n-1_{min}} \leq q_{n-1} \leq q_{n-1_{max}} \tag{24}$$

$$q_{n_{min}} \leq q_n \leq q_{n_{max}} \tag{25}$$

where $l_i$ is the $i$-th DOF for the kinematic chain associated to each end effector, as indicated in (26):

$$n = \sum_{i=1}^{r} l_i \tag{26}$$

The intermediate matrices ${}^{l_{i-l}}T_i$ in (27) describe the rotation $\mathbf{Rot}^c_{i,j}(x_\gamma, y_\beta, z_\alpha)$ and translation $P^c_{i,j}$ between two consecutive links $i-1$ and $i$ in each kinematic chain [2],

$${}^{l_i-1}T_i = \begin{bmatrix} C_\alpha C_\beta & C_\alpha S_\beta S_\gamma - S_\alpha C_\gamma & C_\alpha S_\beta C_\gamma + S_\alpha S_\gamma & p^c_{i,x_j} \\ S_\alpha C_\beta & S_\alpha S_\beta S_\gamma + C_\alpha C_\gamma & S_\alpha S_\beta C_\gamma - C_\alpha S_\gamma & p^c_{i,y_j} \\ -S_\beta & C_\beta S_\gamma & C_\beta C_\gamma & p^c_{i,z_j} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{27}$$

### D. SOLUTION OF THE PROPOSED OPTIMIZATION PROBLEM TO OBTAIN THE IK APPLYING THE DE ALGORITHM

Although in this proposal it is possible to use different heuristic algorithms as solver, in this particular case the Differential Evolution [22] algorithm has been implemented, since it has been applied to the solution of diverse optimization problems in engineering, such as synthesis of mechanisms [23],

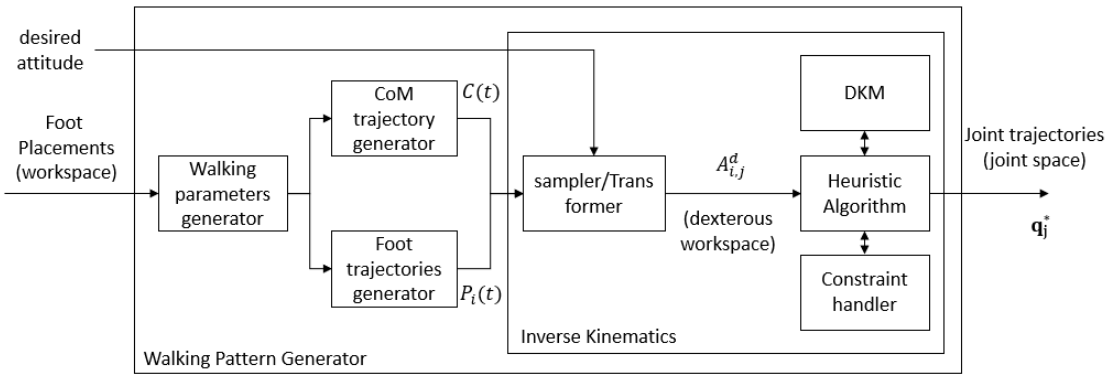


**FIGURE 3.** Proposed internal composition of the IK heuristic-based solver block.

[24], robotics [25] and renewable energies [26], among others. Its popularity is mainly due to its good performance and simplicity. It can handle non-differentiable, multi-modal and non-linear objective functions, and requires few tuning parameters [27]. Differential Evolution is a population-based heuristic algorithm proposed by Storn and Price in 1995. In its rand/1/bin version, the central mechanism is the differential mutation, which consists of adding to a vector of the population, the scaled difference between two vectors taken from the same population, in order to direct the search. Although originally DE was designed for unconstrained optimization problems, it is possible to incorporate a constraint handler for solving CNOP cases [28].

Figure 4 shows the flowchart of the proposed approach for solving the IKP at each point in the discretized trajectories of the robot feet. As a first step, the diagram considers the generation of the trajectories $P_i^d(t)$ for the feet, taking into account the walking parameters that encode the foot placements. It also considers the generation of the trajectory $C^d(t)$ of the CoM, that the torso must follow to ensure a stable dynamic walking. Subsequently, a process of discretization of the trajectories is carried out to obtain $m$ points in $\mathbb{R}^3$ and construct the multi-matrices $A^d_{CoM}$ and $A^d_i$, which specify the orientation and desired position (pose) at each point $m$ of the trajectories for both the CoM and the feet with respect to the dexterous workspace coordinate system (inertial frame). This is in turn transformed to the floating base in the torso in order to state the IKP from this reference.

The IKP is solved by means of DE/rand/1/bin, considering equations (19)-(26) and keeping the best solution (joint vector) for each of the $m$ points. In the last stage, the $n$ joint trajectories are constructed from these solutions within their own space. It is important to notice that these are safe trajectories for the robot in terms of evasion of position singularities and joint range, allowing their direct implementation in the legged robot.

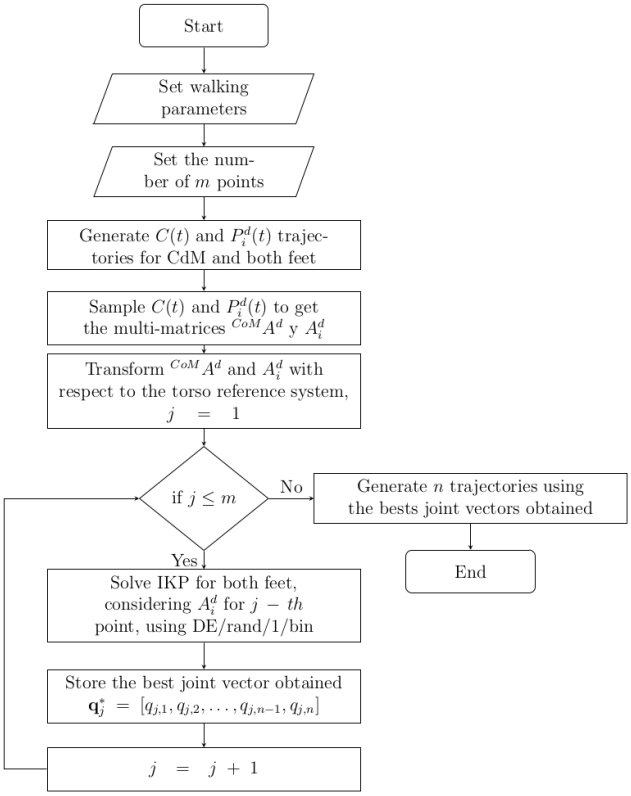


**FIGURE 4.** Flowchart for the proposed solution approach.

## III. CASE STUDY: A 12-DOF BIPED ROBOT

A 12-DOF biped robot is analyzed as a case study, to solve its IKP with the proposed approach for determining the joint trajectories to generate a stable dynamic walking. It is a symmetric robot with 6-DOF in each leg. This was selected because is considered as one of the most complex types between the legged robots since it presents an intrinsic instability [19], [29]. Figure 5 includes a graphical description of the selected robot [30].

The physically consistent trajectories $C^d_{CoM}$ and $P^d_i$ are generated from a 3D Linear Inverted Pendulum Model (3D-LIPM) [19] and cubic Bezier curves [2], respectively. For the CoM, equations (28) and (29) describe the 3D pendulum dynamics [18], considering the parameters in Tables 1 and 2,

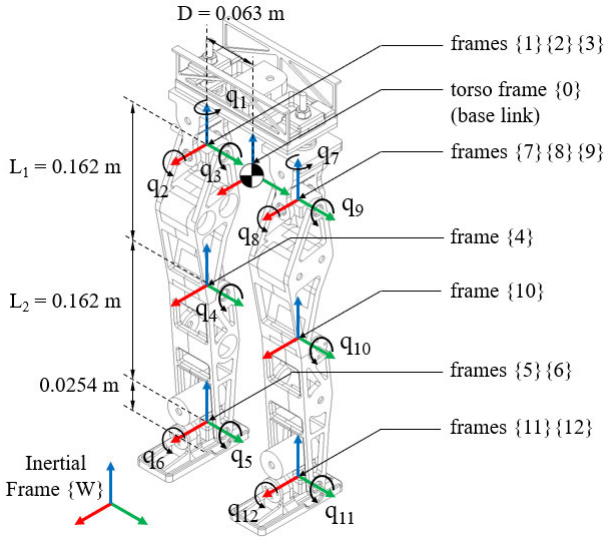


**FIGURE 5.** Biped robot kinematic structure.

**TABLE 1.** Parameters of the foot trajectories.

| Parameter | Magnitude |
|---|---|
| step time $T_{sub}$ | 1 s |
| step height $r_h$, $l_h$ | 0.05 m |
| foot placements $S_x^{n_s}$, $S_y^{n_s}$ | [m] |

where $z_c$ is a distance along the z-axis with respect to the coordinate system of the dexterous workspace representing the height of the constraint plane for CoM; and $p_x^*$, $p_y^*$ are the Cartesian coordinates that modify the foot placements to change the CoM velocity of the robot when it is required. The coordinates $p_x^*$ and $p_y^*$ are calculated by applying the optimization method proposed in [18], with the constant values $a = 5$ and $b = 1$ for the minimum foot-placement position error.

$$\ddot{x}(t) = \frac{g}{z_c}(x - p_x^*) \tag{28}$$

$$\ddot{y}(t) = \frac{g}{z_c}(y - p_y^*), \tag{29}$$

The foot trajectories $P_i^d$ are defined by the function in (30), where $P_0, P_1, P_2, P_3$ are control points to define the curve, and $t \in [0, 1]$. Table 3 contains the coded information of the foot placements, for the five steps considered for the case study.

$$\vec{B}(t) = (1-t)^3\vec{P}_0 + 3t(1-t)^2\vec{P}_1 + 3t(1-t)^2\vec{P}_2 + t^3\vec{P}_3, \tag{30}$$

The reference trajectories were sampled with $m = 350$ points for covering 7s of simulation with a sampling time $T = 20$ms. These trajectories are presented in Fig. 6, where the torso describes the dynamics of the 3D-LIPM consistently synchronized with both foot trajectories, which is required to generate a stable walking. The model 3D-LIPM was solved considering seven segments, in such a way that the final conditions of one pendulum correspond to the initial conditions of the next, ensuring a continuous dynamic behavior of the robot along the walking.

**TABLE 2.** Parameters of the CoM trajectory.

| Parameter | Magnitude |
|---|---|
| step time $T_{sub}$ | 1 s |
| CoM constraint plane height $z_c$ | 0.32 m |
| CoM height $z_r$ | 0.349 m |
| number of steps $n_s$ | 5 |
| foot placements $S_x^{n_s}$, $S_y^{n_s}$ | [m] |

**TABLE 3.** Coding of the foot placements.

| | Number of step $n_s$ | | | | |
|---|---|---|---|---|---|
| Foot placement | 1 | 2 | 3 | 4 | 5 |
| $S_x^{n_s}$ [m] | 0.0 | 0.1 | 0.1 | 0.1 | 0.0 |
| $S_y^{n_s}$ [m] | 0.126 | 0.126 | 0.126 | 0.126 | 0.126 |

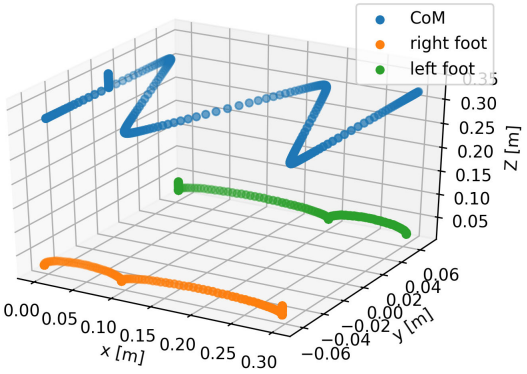


**FIGURE 6.** Sampled $C_{CdM}^d$ and $P_i^d$ trajectories.

### A. DEVELOPMENT OF THE DIRECT KINEMATICS MODEL (DKM) FOR THE CASE STUDY

The Cartesian positions of each end effector from the joint vector can be calculated with the DKM, by means of homogeneous transformations of the consecutive frames (coordinate systems) defined in Fig. 4 with respect to the floating frame of the torso (base link), identified as {0}. Since the legs are assumed to be identical, the consecutive transformations to each foot with frames {7} and {12} for the right and left legs respectively, are described in Table 4, where $S_\delta = sin(q_\delta)$, $C_\delta = cos(q_\delta)$ for $\delta = 1$ to 12, and $D$ is the distance between the torso frame and the origin of the hip frame.

The DKM can be described for the complete trajectory by the multi-matrices $A_1^c$ and $A_2^c$ in (31) and (32), for the right ($i = 1$) and left ($i = 2$) legs, respectively, for $j = 0$ to $m$. The

**TABLE 4.** Homogeneous transformation matrices for each leg.

**Right leg**

$$^{0}T_{1} = \begin{bmatrix} C_1 & -S_1 & 0 & 0 \\ S_1 & C_1 & 0 & -D \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad ^{1}T_{2} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C_2 & -S_2 & 0 \\ 0 & S_2 & C_2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{2}T_{3} = \begin{bmatrix} C_3 & 0 & S_3 & 0 \\ 0 & 1 & 0 & 0 \\ -S_3 & 0 & C_3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad ^{3}T_{4} = \begin{bmatrix} C_4 & 0 & S_4 & 0 \\ 0 & 1 & 0 & 0 \\ -S_4 & 0 & C_4 & -L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{4}T_{5} = \begin{bmatrix} C_5 & 0 & S_5 & 0 \\ 0 & 1 & 0 & 0 \\ -S_5 & 0 & C_5 & -L_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad ^{5}T_{6} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C_6 & -S_6 & 0 \\ 0 & S_6 & C_6 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

**Left leg**

$$^{0}T_{7} = \begin{bmatrix} C_7 & -S_7 & 0 & 0 \\ S_7 & C_7 & 0 & D \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad ^{7}T_{8} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C_8 & -S_8 & 0 \\ 0 & S_8 & C_8 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{8}T_{9} = \begin{bmatrix} C_9 & 0 & S_9 & 0 \\ 0 & 1 & 0 & 0 \\ -S_9 & 0 & C_9 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad ^{9}T_{10} = \begin{bmatrix} C_{10} & 0 & S_{10} & 0 \\ 0 & 1 & 0 & 0 \\ -S_{10} & 0 & C_{10} & -L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{10}T_{11} = \begin{bmatrix} C_{11} & 0 & S_{11} & 0 \\ 0 & 1 & 0 & 0 \\ -S_{11} & 0 & C_{11} & -L_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad ^{11}T_{12} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C_{12} & -S_{12} & 0 \\ 0 & S_{12} & C_{12} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

elements of the multi-matrices are defined by (33) to (56),

$$A_1^c(\Theta) = \begin{bmatrix} \mathbf{Rot}_1^c(\Theta) & P_1^c(\Theta) \\ 0 & 1 \end{bmatrix} = {}^{0}T_1^1 T_2^2 T_3^3 T_4^4 T_5^5 T_6 = \begin{bmatrix} n_{1,x_j}^c & o_{1,x_j}^c & a_{1,x_j}^c & p_{1,x_j}^c \\ n_{1,y_j}^c & o_{1,y_j}^c & a_{1,y_j}^c & p_{1,y_j}^c \\ n_{1,z_j}^c & o_{1,z_j}^c & a_{1,z_j}^c & p_{1,z_j}^c \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{31}$$

$$A_2^c(\Theta) = \begin{bmatrix} \mathbf{Rot}_2^c(\Theta) & P_2^c(\Theta) \\ 0 & 1 \end{bmatrix} = {}^{0}T_7^7 T_8^8 T_9^9 T_10^{10} T_{11}^{11} T_{12} = \begin{bmatrix} n_{2,x_j}^c & o_{2,x_j}^c & a_{2,x_j}^c & p_{2,x_j}^c \\ n_{2,y_j}^c & o_{2,y_j}^c & a_{2,y_j}^c & p_{2,y_j}^c \\ n_{2,z_j}^c & o_{2,z_j}^c & a_{2,z_j}^c & p_{2,z_j}^c \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{32}$$

$$p_{1,x_j}^c = -L_1(S_1S_2C_3 + S_3C_1) - L_2((-S_1S_2S_3 + C_1C_3)S_4 + (S_1S_2C_3 + S_3C_1)C_4) \tag{33}$$

$$p_{1,y_j}^c = -D - L_1(S_1S_3 - S_2C_1C_3) - L_2((S_1S_3 - S_2C_1C_3)C_4 + (S_1C_3 + S_2S_3C_1)S_4) \tag{34}$$

$$p_{1,z_j}^c = -L_1C_2C_3 - L_2(-S_3S_4C_2 + C_2C_3C_4) \tag{35}$$

$$n_{1,x_j}^c = -((-S_1S_2S_3 + C_1C_3)S_4 + (S_1S_2C_3 + S_3C_1)C_4)S_5 + ((-S_1S_2S_3 + C_1C_3)C_4 - (S_1S_2C_3 + S_3C_1)S_4)C_5 \tag{36}$$

$$n_{1,y_j}^c = (-(S_1S_3 - S_2C_1C_3)S_4 + (S_1C_3 + S_2S_3C_1)C_4)C_5 - ((S_1S_3 - S_2C_1C_3)C_4 + (S_1C_3 + S_2S_3C_1)S_4)S_5 \tag{37}$$

$$n_{1,z_j}^c = -(-S_3S_4C_2 + C_2C_3C_4)S_5 + (-S_3C_2C_4 - S_4C_2C_3)C_5 \tag{38}$$

$$o_{1,x_j}^c = (((-S_1S_2S_3 + C_1C_3)S_4 + (S_1S_2C_3 + S_3C_1)C_4)C_5 + ((-S_1S_2S_3 + C_1C_3)C_4 - (S_1S_2C_3 + S_3C_1)S_4)S_5)S_6 - S_1C_2C_6 \tag{39}$$

$$o_{1,y_j}^c = ((-(S_1S_3 - S_2C_1C_3)S_4 + (S_1C_3 + S_2S_3C_1)C_4)S_5 + ((S_1S_3 - S_2C_1C_3)C_4 + (S_1C_3 + S_2S_3C_1)S_4)C_5)S_6 + C_1C_2C_6 \tag{40}$$

$$o_{1,z_j}^c = ((-S_3S_4C_2 + C_2C_3C_4)C_5 + (-S_3C_2C_4 - S_4C_2C_3)S_5)S_6 + S_2C_6 \tag{41}$$

$$a_{1,x_j}^c = (((-S_1S_2S_3 + C_1C_3)S_4 + (S_1S_2C_3 + S_3C_1)C_4)C_5 + ((-S_1S_2S_3 + C_1C_3)C_4 - (S_1S_2C_3 + S_3C_1)S_4)S_5)C_6 + S_1S_6C_2 \tag{42}$$

$$a_{1,y_j}^c = ((-(S_1S_3 - S_2C_1C_3)S_4 + (S_1C_3 + S_2S_3C_1)C_4)S_5 + ((S_1S_3 - S_2C_1C_3)C_4 + (S_1C_3 + S_2S_3C_1)S_4)C_5)C_6 - S_6C_1C_2 \tag{43}$$

$$a_{1,z_j}^c = ((-S_3S_4C_2 + C_2C_3C_4)C_5 + (-S_3C_2C_4 - S_4C_2C_3)S_5)C_6 - S_2S_6 \tag{44}$$

$$p_{2,x_j}^c = -L_3(S_7S_8C_9 + S_9C_7) - L_4((-S_7S_8S_9 + C_7C_9)S_{10} + (S_7S_8C_9 + S_9C_7)C_{10}) \tag{45}$$

$$p_{2,y_j}^c = D - L_3(S_7S_9 - S_8C_7C_9) - L_4((S_7S_9 - S_8C_7C_9)C_{10} + (S_7C_9 + S_8S_9C_7)S_{10}) \tag{46}$$

$$p_{2,z_j}^c = -L_3C_8C_9 - L_4(-S_{10}S_9C_8 + C_{10}C_8C_9) \tag{47}$$

$$n_{2,x_j}^c = -((-S_7S_8S_9 + C_7C_9)S_{10} + (S_7S_8C_9 + S_9C_7)C_{10})S_{11} + ((-S_7S_8S_9 + C_7C_9)C_{10} - (S_7S_8C_9 + S_9C_7)S_{10})C_{11} \tag{48}$$

$$n_{2,y_j}^c = (-(S_7S_9 - S_8C_7C_9)S_{10} + (S_7C_9 + S_8S_9C_7)C_{10})C_{11} - ((S_7S_9 - S_8C_7C_9)C_{10} + (S_7C_9 + S_8S_9C_7)S_{10})S_{11} \tag{49}$$

$$n_{2,z_j}^c = -(-S_{10}S_9C_8 + C_{10}C_8C_9)S_{11} + (-S_{10}C_8C_9 - S_9C_{10}C_8)C_{11} \tag{50}$$

$$o_{2,x_j}^c = (((-S_7S_8S_9 + C_7C_9)S_{10} + (S_7S_8C_9 + S_9C_7)C_{10})C_{11} + ((-S_7S_8S_9 + C_7C_9)C_{10} - (S_7S_8C_9 + S_9C_7)S_{10})S_{11})S_{12} - S_7C_{12}C_8 \tag{51}$$

$$o_{2,y_j}^c = ((-(S_7S_9 - S_8C_7C_9)S_{10} + (S_7C_9 + S_8S_9C_7)C_{10})S_{11} + ((S_7S_9 - S_8C_7C_9)C_{10} + (S_7C_9 + S_8S_8C_7)S_{10})C_{11})C_{12}S_{12} + C_7C_8 \tag{52}$$

$$o_{2,z_j}^c = ((-S_{10}S_9C_8 + C_{10}C_8C_9)C_{11} + (-S_{10}C_8C_9 - S_9C_{10}C_8)S_{11})S_{12} + S_8C_{12} \tag{53}$$

$$a_{2,x_j}^c = (((-S_1S_2S_3 + C_1C_3)S_4 + (S_1S_2C_3 + S_3C_1)C_4)C_5 + ((-S_1S_2S_3 + C_1C_3)C_4 - (S_1S_2C_3 + S_3C_1)S_4)S_5)C_6 + S_1S_6C_2 \tag{54}$$

$$a_{2,y_j}^c = ((-(S_7S_9 - S_8C_7C_9)S_{10} + (S_7C_9 + S_8S_9C_7)C_{10})S_{11} + ((S_7S_9 - S_8C_7C_9)C_{10} + (S_7C_9 + S_8S_9C_7)S_{10})C_{11})C_{12} - S_{12}C_7C_8 \tag{55}$$

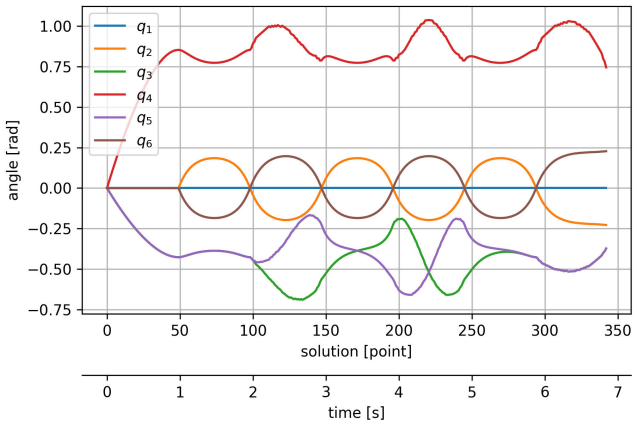


**FIGURE 7.** Joint trajectories of the right leg, obtained using analytical equations.

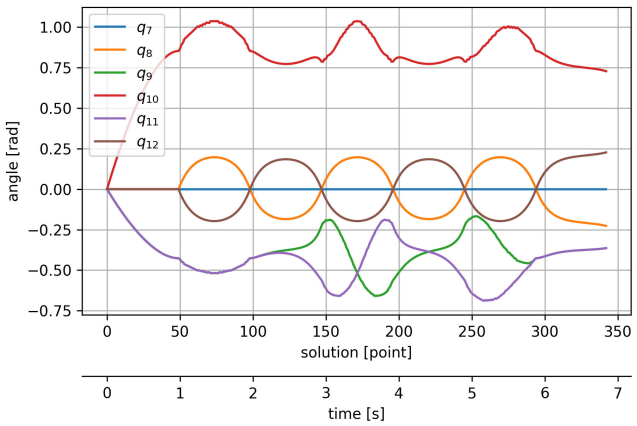


**FIGURE 8.** Joint trajectories of the left leg, obtained using analytical equations.

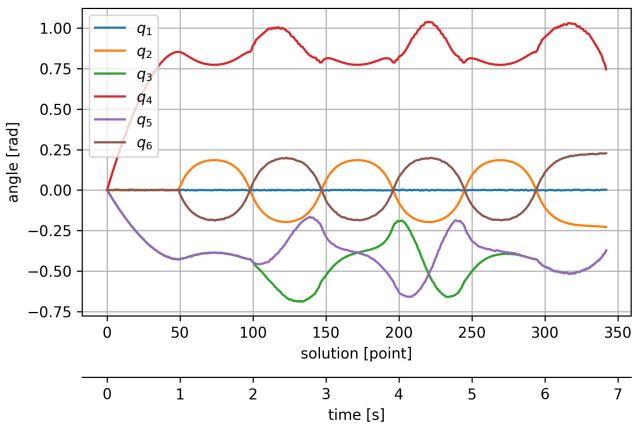


**FIGURE 9.** Joint trajectories of the right leg, obtained using DE.

$$a^c_{2,z_j} = ((-S_{10}S_9C_8 + C_{10}C_8C_9)C_{11} + (-S_{10}C_8C_9 - S_9C_{10}C_8)S_{11})C_{12} - S_{12}S_8 \quad (56)$$

### B. PARTICULAR OPTIMIZATION PROBLEM

The general optimization problem for legged robots proposed in Section II-C is particularized for the biped robot of the case study, in (57)-(64). Since trajectories are defined with $m = 350$ sampled points, the IKP must be solved

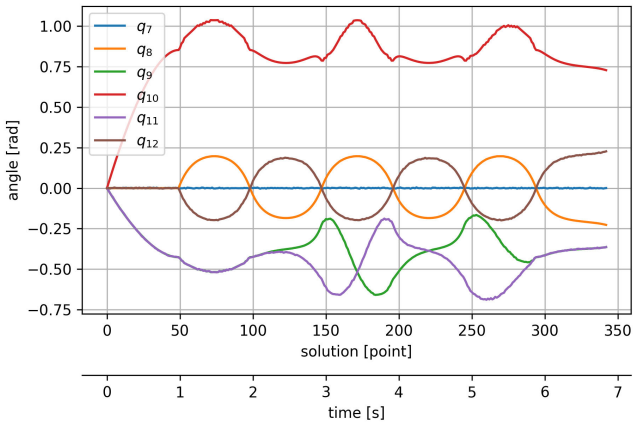


**FIGURE 10.** Joint trajectories of the left leg, obtained using DE.

350 times, considering simultaneously the CoM and the foot transformed trajectories. The mechanical joint limits of the biped robot considered in the case study [30] and shown in Table 5 define the inequality constraints in (61)-(64), for an optimal configuration within the joint ranges to avoid damages in a real implementation. The design vector in (58) is the joint coordinate vector, where the first six variables describe the right leg and the remaining variables correspond to the left one.

$$\min_{\mathbf{q}\in\mathbb{R}^{12}} \quad f_j(\mathbf{q}) = \sum_{i=1}^{2} \|P^d_{i,j}(\mathbf{q}) - P^c_{i,j}(\mathbf{q})\| \quad \forall j \in \mathbb{Z}, \quad 1 \le j \le 350 \quad (57)$$

where

$$\mathbf{q} = [q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9, q_{10}, q_{11}, q_{12}] \quad (58)$$

subject to:

$$\begin{aligned} h_1 = {} & |\, n^d_{1,x_j} - n^c_{1,x_j} \,| + |\, n^d_{1,y_j} - n^c_{1,y_j} \,| + |\, n^d_{1,z_j} - n^c_{1,z_j} \,| \\ & + |\, o^d_{1,x_j} - o^c_{1,x_j} \,| + |\, o^d_{1,y_j} - o^c_{1,y_j} \,| + |o^d_{1,z_j} - o^c_{1,z_j} \,| \\ & + |\, a^d_{1,x_j} - a^c_{1,x_j} \,| + |\, a^d_{1,y_j} - a^c_{1,y_j} \,| \\ & + |\, a^d_{1,z_j} - a^c_{1,z_j} \,| = 0 \quad (59) \\ h_2 = {} & |\, n^d_{2,x_j} - n^c_{2,x_j} \,| + |\, n^d_{2,y_j} - n^c_{2,y_j} \,| + |\, n^d_{2,z_j} - n^c_{2,z_j} \,| \\ & + |\, o^d_{2,x_j} - o^c_{2,x_j} \,| + |\, o^d_{2,y_j} - o^c_{2,y_j} \,| + |\, o^d_{2,z_j} - o^c_{2,z_j} \,| \\ & + |\, a^d_{2,x_j} - a^c_{2,x_j} \,| + |\, a^d_{2,y_j} - a^c_{2,y_j} \,| \\ & + |\, a^d_{2,z_j} - a^c_{2,z_j} \,| = 0 \quad (60) \end{aligned}$$

$$-0.7 \le q_1, q_5, q_6, q_7, q_{11}, q_{12} \le 0.7 \quad (61)$$

$$-0.5 \le q_2, q_8 \le 0.5 \quad (62)$$

$$-1.5707 \le q_3, q_9 \le 0.5 \quad (63)$$

$$0 \le q_4, q_{10} \le 1.5707 \quad (64)$$

## IV. IMPLEMENTATION AND RESULTS

### A. COMPUTATIONAL ISSUES

The DE/rand/1/bin algorithm for solving the IKP of the biped robot was implemented in a Phyton-based ROS node

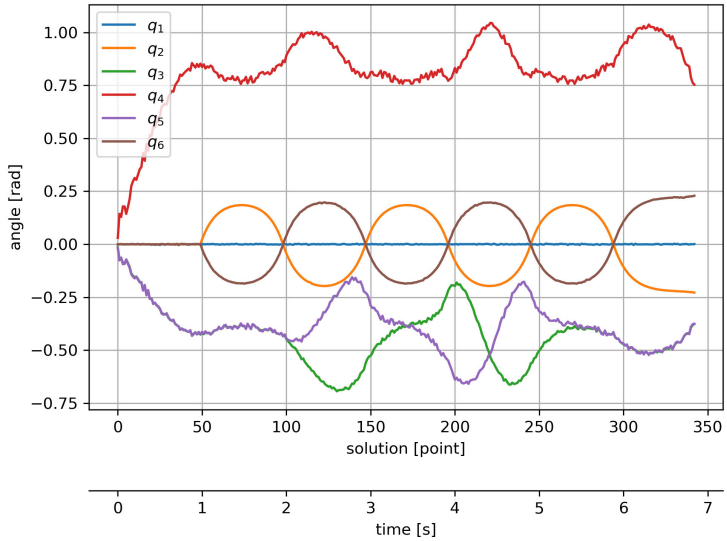


**FIGURE 11.** Joint trajectories of the right leg, obtained using HS.

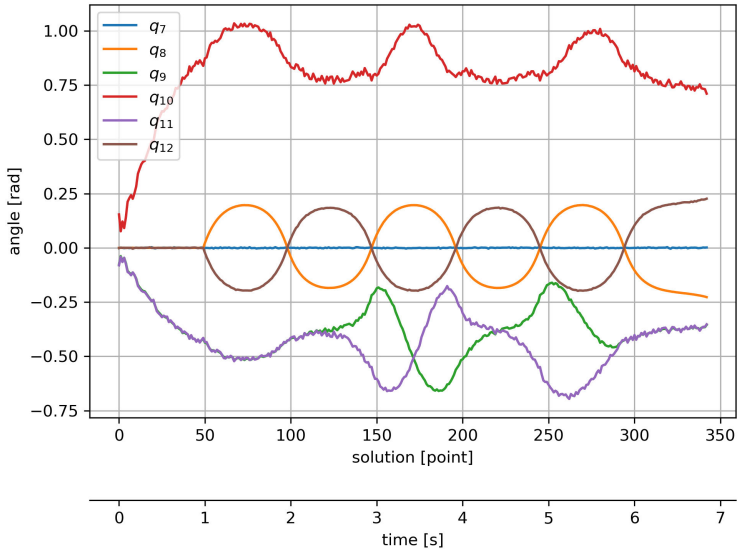


**FIGURE 12.** Joint trajectories of the left leg, obtained using HS.

**TABLE 5.** Biped robot mechanical joint limits.

| **Joint** | $q_1, q_7$ | $q_2, q_8$ | $q_3, q_9$ |
|---|---|---|---|
| Range [**rad**] | $[-0.7, 0.7]$ | $[-0.5, 0.5]$ | $[-1.5707, 0.5]$ |
| **Joint** | $q_4, q_{10}$ | $q_5, q_{11}$ | $q_6, q_{12}$ |
| Range [**rad**] | $[0, 1.5707]$ | $[-0.7, 0.7]$ | $[-0.7, 0.7]$ |

(melodic distro), using a NVIDIA Jetson Nano developer device with a quad-core CPU ARM A57@1.43GHz, a 128-core Maxwell CPU, 4GB RAMLPDDR4, and a Ubuntu 18.04 operating system. Table 6 shows the algorithm tuning. The algorithm uses two parameters for controlling the evolution of the individuals along the generations. The cross factor $CR$ determines the mixture grade between the individuals that will produce a new solution, while the mutation factor $F$ corresponds to the mutation grade from one generation to another. The values were selected using a trial and error approach, in order to privilege exploration. The experiments showed that a population with less than 30 individuals and low values for $CR$ and $F$, conduce to local optimal solutions, mainly in the first part of the trajectories, since the configuration of the robot is close to a singularity. Also, values of $\epsilon_1$ and $\epsilon_2$ lower than 0.01 radians can cause slow convergence.

Two stop criteria were applied to the solution of each point: the algorithm halts after 15,000 iterations or if the difference between the OF value of the best and worse individuals at the end of an iteration is lower than $1 \times 10^{-6}$, and both of them are feasible. The implementation of the DE/rand/1/bin algorithm was complemented with the feasibility rules of Deb [31] for handling the equality constraints, incorporating tolerances for each foot orientation error $E_i^o$, defined by $\epsilon_1$ and $\epsilon_2$ in order to prioritize, if required, the maximum orientation error per foot.

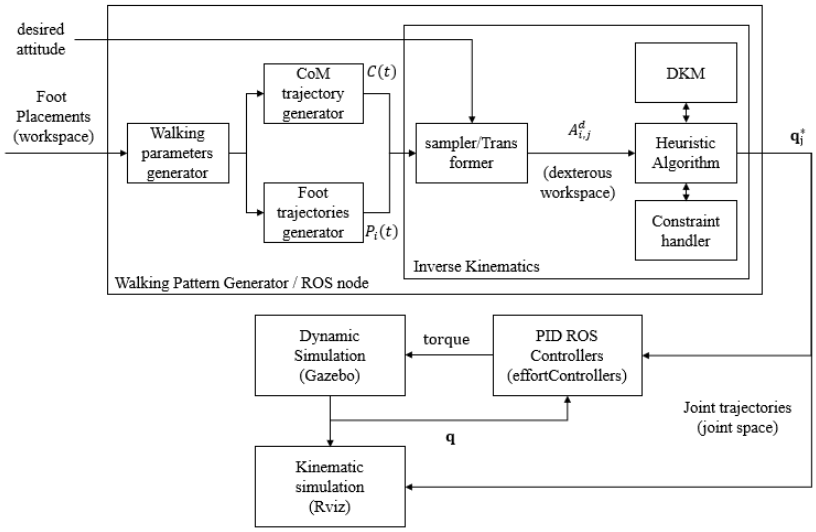


**FIGURE 13.** Dynamic walking control scheme implemented in ROS.

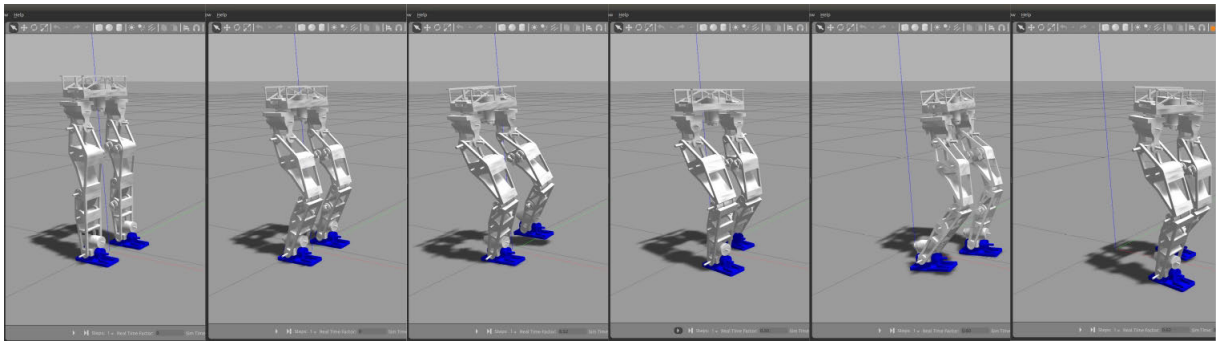

**FIGURE 14.** Simulation in Gazebo of the dynamic walking performed by the biped robot, using the best solution generated by the proposed optimization approach.

**TABLE 6.** ED/rand/1/bin implemented parameters.

| Parameter | Value |
|---|---|
| Population | 300 |
| $CR$ | 0.9 |
| $F$ | 0.8-0.9 |
| Max. Generations. | 15000 |
| Tolerance [rad] | $\epsilon_1 = 0.01$ $\epsilon_2 = 0.01$ |

**TABLE 7.** Harmony Search implemented parameters.

| Parameter | Value |
|---|---|
| Harmonies | 10 |
| $r_{accept}$ | 0.95 |
| $r_{pa}$ | 0.85 |
| Max. Cycles. | 15000 |
| Tolerance [rad] | $\epsilon_1 = 0.01$ $\epsilon_2 = 0.01$ |

The complexity of a numerical optimization problem can be evaluated by means of the $\rho$ parameter [32], which establishes a relation between the feasible region derived from a set of feasible solutions $F$ and the search space defined by at least 1,000,000 randomly generated individuals $S$, as indicated in (65),

$$\rho = \frac{F}{S} \times 100\% \tag{65}$$

As can be seen, the complexity of the problem increases as the value of $\rho$ diminishes. For the IKP case study, 350 millions of random solutions were generated along the trajectories and none feasible solution was found, producing $\rho = 0$. This value indicates that the case study is a very complex optimization problem, with a reduced feasible zone.

### B. ANALYSIS OF RESULTS

For comparison purposes, the IKP was solved $m = 350$ times for the CoM and foot trajectories considering both analytical approach and the proposed approach. Thus, it was necessary to derive the analytical equations that describe the Inverse Kinematics for this particular biped robot, Figs. 7 and 8 shows the 12 joint trajectories obtained. For the proposed approach, the DE implementation was able to solve the problem, resulting in 12 joint trajectories for a successful dynamic walking of the biped robot, six for each leg. The corresponding joint

**TABLE 8.** Best joint vectors obtained.

| $q_1$ | $q_2$ | $q_3$ | $q_4$ | $q_5$ | $q_6$ | $q_7$ | $q_8$ | $q_9$ | $q_{10}$ | $q_{11}$ | $q_{12}$ | $OF[m]$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| -0.000718 | -0. | -0.000104 | 0.000209 | 0.000691 | 0.001931 | 0.001913 | -0. | -0.000567 | 0.001134 | -0.001783 | -0.001571 | 0. |
| 0.00211 | 0. | -0.014845 | 0.029691 | -0.013584 | -0.000065 | -0.001747 | -0. | -0.014851 | 0.029701 | -0.015223 | 0.00163 | 0. |
| -0.000826 | 0. | -0.029497 | 0.058994 | -0.026323 | 0.000593 | 0.001031 | -0. | -0.029501 | 0.059002 | -0.030446 | -0.002948 | 0. |
| -0.000719 | -0. | -0.043941 | 0.087883 | -0.043762 | -0.000295 | -0.000935 | 0. | -0.043945 | 0.08789 | -0.045022 | 0.000221 | 0. |
| 0.000333 | 0. | -0.05818 | 0.11636 | -0.059096 | -0.00051 | 0.001029 | 0. | -0.05818 | 0.116361 | -0.055283 | 0.000813 | 0. |
| 0.002191 | -0. | -0.072205 | 0.144409 | -0.071506 | 0.000082 | -0.000408 | 0. | -0.072207 | 0.144414 | -0.072169 | 0.003884 | 0. |
| -0.000957 | 0. | -0.086017 | 0.172035 | -0.085078 | 0.002968 | 0.001319 | 0. | -0.086018 | 0.172037 | -0.086569 | 0.001557 | 0. |
| 0.000461 | -0. | -0.09962 | 0.19924 | -0.09973 | 0.002778 | -0.000647 | 0. | -0.099618 | 0.199235 | -0.099533 | 0.001104 | 0. |
| 0.001483 | -0. | -0.113005 | 0.226011 | -0.113647 | 0.002216 | 0.000611 | 0. | -0.113007 | 0.226014 | -0.113689 | 0.001 | 0. |
| 0.000628 | -0. | -0.126172 | 0.252344 | -0.128839 | -0.00071 | -0.000219 | 0. | -0.126176 | 0.252353 | -0.128559 | 0.001181 | 0. |
| 0.000594 | 0. | -0.139122 | 0.278244 | -0.136969 | -0.000245 | -0.000042 | 0. | -0.139122 | 0.278245 | -0.138342 | -0.001041 | 0. |
| 0.000242 | -0. | -0.151847 | 0.303693 | -0.15219 | -0.000034 | 0.002423 | 0. | -0.151849 | 0.303699 | -0.150012 | 0.000004 | 0. |
| -0.000089 | -0. | -0.164352 | 0.328704 | -0.166806 | -0.001151 | -0.000375 | 0. | -0.16435 | 0.328699 | -0.161875 | 0.001809 | 0. |
| -0.000292 | -0. | -0.176623 | 0.353246 | -0.177403 | 0.000536 | 0.002865 | -0. | -0.176623 | 0.353246 | -0.176279 | -0.001249 | 0. |
| -0.000176 | -0. | -0.188665 | 0.377331 | -0.190236 | 0.000239 | 0.000114 | -0. | -0.188667 | 0.377333 | -0.188467 | 0.0009 | 0. |
| -0.001925 | 0. | -0.200477 | 0.400953 | -0.200017 | 0.000987 | -0.000523 | 0. | -0.200474 | 0.400949 | -0.20086 | 0.001263 | 0. |
| 0.004356 | 0. | -0.212047 | 0.424093 | -0.212169 | 0.000383 | 0.001564 | -0. | -0.212046 | 0.424093 | -0.212066 | -0.000481 | 0. |
| 0.000316 | -0. | -0.223378 | 0.446756 | -0.225571 | 0.000601 | 0.001937 | -0. | -0.223377 | 0.446755 | -0.223286 | -0.000242 | 0. |
| 0.000983 | 0. | -0.234464 | 0.468928 | -0.233881 | 0.00289 | -0.000403 | -0. | -0.234464 | 0.468928 | -0.233642 | -0.00225 | 0. |
| 0.001591 | -0. | -0.245303 | 0.490607 | -0.244079 | -0.001956 | -0.001384 | -0. | -0.245303 | 0.490606 | -0.244687 | 0.001441 | 0. |
| 0.000168 | -0. | -0.255891 | 0.511781 | -0.256247 | -0.001453 | 0.000872 | -0. | -0.255891 | 0.511782 | -0.256317 | 0.002784 | 0. |
| 0.001404 | 0. | -0.266222 | 0.532445 | -0.267328 | 0.001173 | -0.000982 | 0. | -0.266221 | 0.532443 | -0.266795 | -0.001205 | 0. |
| 0.001833 | 0. | -0.276293 | 0.552587 | -0.274698 | -0.001489 | -0.000645 | -0. | -0.276294 | 0.552588 | -0.275338 | 0.003167 | 0. |
| 0.000052 | -0. | -0.2861 | 0.5722 | -0.287969 | -0.000139 | 0.000932 | -0. | -0.2861 | 0.5722 | -0.287796 | 0.000318 | 0. |
| -0.000043 | -0. | -0.295637 | 0.591274 | -0.294196 | 0.000037 | -0.00007 | 0. | -0.295638 | 0.591275 | -0.298765 | -0.000692 | 0. |
| 0.00027 | -0. | -0.304899 | 0.609797 | -0.30235 | -0.001002 | 0.002655 | 0. | -0.3049 | 0.6098 | -0.304781 | 0.00174 | 0. |
| -0.002802 | -0. | -0.313882 | 0.627765 | -0.314504 | 0.000324 | -0.001525 | 0. | -0.313882 | 0.627765 | -0.315786 | -0.00011 | 0. |
| -0.000307 | 0. | -0.32258 | 0.645161 | -0.323613 | 0.001983 | -0.001107 | -0. | -0.322581 | 0.645163 | -0.324073 | 0.000382 | 0. |
| -0.001258 | -0. | -0.330987 | 0.661973 | -0.331868 | -0.000491 | 0.003207 | -0. | -0.330987 | 0.661974 | -0.330882 | -0.000322 | 0. |
| 0.000776 | 0. | -0.339096 | 0.678193 | -0.339984 | 0.000849 | -0.001373 | 0. | -0.339097 | 0.678193 | -0.33931 | 0.003274 | 0. |
| -0.000461 | 0. | -0.346903 | 0.693806 | -0.349101 | 0.001404 | -0.001772 | 0. | -0.346903 | 0.693807 | -0.346914 | 0.002624 | 0. |
| 0.001157 | 0. | -0.3544 | 0.708801 | -0.354588 | -0.000416 | 0.001269 | 0. | -0.3544 | 0.7088 | -0.355309 | -0.000074 | 0. |
| 0.001139 | -0. | -0.361581 | 0.723163 | -0.360957 | -0.000258 | 0.002975 | 0. | -0.36158 | 0.72316 | -0.361168 | -0.000384 | 0. |
| -0.000616 | -0. | -0.368437 | 0.736874 | -0.368593 | 0.000202 | -0.000533 | 0. | -0.368438 | 0.736875 | -0.371136 | -0.000224 | 0. |
| -0.000589 | 0. | -0.374963 | 0.749926 | -0.372355 | 0.001481 | 0.003132 | 0. | -0.374962 | 0.749923 | -0.374605 | 0.000087 | 0. |
| 0.000623 | -0. | -0.381147 | 0.762295 | -0.381724 | 0.000132 | -0.0002 | -0. | -0.381147 | 0.762294 | -0.383601 | -0.001089 | 0. |
| 0.001488 | -0. | -0.386984 | 0.77397 | -0.388642 | 0.000297 | -0.001956 | 0. | -0.386985 | 0.773971 | -0.385994 | -0.001882 | 0. |
| 0.000235 | -0. | -0.392465 | 0.784931 | -0.391544 | 0.001763 | 0.000043 | 0. | -0.392465 | 0.78493 | -0.391983 | -0.000996 | 0. |
| 0.000118 | 0. | -0.39758 | 0.79516 | -0.397077 | 0.003034 | -0.000934 | -0. | -0.39758 | 0.795159 | -0.397158 | -0.00313 | 0. |
| 0.000589 | 0. | -0.402318 | 0.804636 | -0.404859 | 0.001362 | 0.000384 | -0. | -0.402319 | 0.804637 | -0.401174 | -0.001786 | 0. |
| 0.000237 | -0. | -0.40667 | 0.81334 | -0.406157 | -0.00188 | 0.001624 | -0.000001 | -0.406669 | 0.813339 | -0.404862 | -0.000152 | 0. |
| 0.001962 | -0. | -0.410623 | 0.821247 | -0.411223 | -0.001952 | 0.000717 | -0. | -0.410623 | 0.821246 | -0.412375 | -0.000073 | 0. |
| 0.001059 | 0. | -0.414167 | 0.828333 | -0.410681 | -0.000219 | -0.000162 | 0. | -0.414166 | 0.828331 | -0.41696 | 0.001937 | 0. |
| -0.002094 | -0. | -0.417289 | 0.834578 | -0.416748 | -0.002019 | 0.001836 | 0. | -0.417287 | 0.834575 | -0.419625 | -0.000181 | 0. |
| 0.001234 | 0. | -0.419974 | 0.839947 | -0.420616 | -0.000466 | 0.000257 | -0. | -0.419974 | 0.839948 | -0.419529 | -0.002195 | 0. |
| 0.001097 | 0. | -0.42221 | 0.844419 | -0.424799 | 0.000563 | 0.001572 | -0. | -0.422209 | 0.844417 | -0.420703 | -0.000823 | 0. |
| -0.001201 | -0. | -0.423978 | 0.847956 | -0.425511 | 0.000508 | 0.001645 | -0. | -0.423978 | 0.847955 | -0.423542 | 0.000139 | 0. |

**TABLE 8.** *(Continued.)* **Best joint vectors obtained.**

| $q_1$ | $q_2$ | $q_3$ | $q_4$ | $q_5$ | $q_6$ | $q_7$ | $q_8$ | $q_9$ | $q_{10}$ | $q_{11}$ | $q_{12}$ | $OF[m]$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| -0.001225 | -0. | -0.425266 | 0.850532 | -0.424722 | -0.000669 | 0.000456 | 0. | -0.425266 | 0.850532 | -0.423348 | 0.00231 | 0. |
| 0.002582 | -0. | -0.426053 | 0.852106 | -0.427681 | -0.000754 | 0.000888 | -0.000001 | -0.426053 | 0.852106 | -0.42483 | 0.000673 | 0. |
| 0.001289 | -0. | -0.426319 | 0.852639 | -0.427115 | 0.002808 | -0.002339 | 0. | -0.42632 | 0.852639 | -0.427287 | -0.00096 | 0. |
| -0.000996 | 0.022624 | -0.425733 | 0.851512 | -0.426728 | -0.024725 | -0.002235 | 0.022702 | -0.433115 | 0.866331 | -0.43451 | -0.021878 | 0. |
| 0.000933 | 0.042789 | -0.424338 | 0.848596 | -0.421669 | -0.042866 | -0.000594 | 0.043229 | -0.446212 | 0.892475 | -0.444701 | -0.042849 | 0. |
| 0.002364 | 0.060746 | -0.422376 | 0.844465 | -0.421565 | -0.062079 | -0.000535 | 0.061579 | -0.45133 | 0.902726 | -0.452536 | -0.064166 | 0. |
| 0.001394 | 0.076723 | -0.419883 | 0.839553 | -0.420307 | -0.076445 | 0.001517 | 0.07831 | -0.463117 | 0.925997 | -0.462514 | -0.076619 | 0. |
| -0.002874 | 0.090929 | -0.416831 | 0.834184 | -0.417314 | -0.091351 | -0.002503 | 0.093126 | -0.467134 | 0.934735 | -0.466509 | -0.093892 | 0. |
| 0.001573 | 0.10355 | -0.414466 | 0.828608 | -0.415388 | -0.101743 | 0.003107 | 0.106415 | -0.471931 | 0.943203 | -0.472198 | -0.106987 | 0. |
| 0.000068 | 0.114756 | -0.411507 | 0.822998 | -0.409782 | -0.116013 | 0.002044 | 0.118334 | -0.476027 | 0.951571 | -0.47495 | -0.117246 | 0. |
| -0.001703 | 0.124696 | -0.408538 | 0.817499 | -0.407663 | -0.124613 | 0.000919 | 0.129024 | -0.480101 | 0.959966 | -0.480134 | -0.129817 | 0. |
| -0.000088 | 0.133504 | -0.406088 | 0.812199 | -0.410155 | -0.133717 | 0.000039 | 0.138611 | -0.48424 | 0.968468 | -0.484906 | -0.13522 | 0. |
| -0.001555 | 0.141299 | -0.403365 | 0.807167 | -0.404219 | -0.141052 | -0.002199 | 0.147205 | -0.488244 | 0.977132 | -0.489254 | -0.147578 | 0. |
| 0.000905 | 0.148184 | -0.40136 | 0.802453 | -0.399065 | -0.148107 | -0.001847 | 0.154909 | -0.492712 | 0.985993 | -0.492044 | -0.153873 | 0. |
| -0.003207 | 0.154253 | -0.398547 | 0.798079 | -0.399654 | -0.154438 | 0.001188 | 0.161809 | -0.497724 | 0.995065 | -0.496179 | -0.160247 | 0. |
| 0.000086 | 0.159588 | -0.397046 | 0.794065 | -0.398938 | -0.159324 | 0.000725 | 0.167982 | -0.502301 | 1.00436 | -0.502044 | -0.168648 | 0. |
| 0.000249 | 0.16426 | -0.395249 | 0.790417 | -0.396094 | -0.163868 | -0.00081 | 0.17289 | -0.500499 | 1.001277 | -0.498975 | -0.173383 | 0. |
| 0.000015 | 0.16833 | -0.393569 | 0.787132 | -0.391685 | -0.168877 | 0.000077 | 0.177786 | -0.505577 | 1.011126 | -0.503611 | -0.177373 | 0. |
| 0.000734 | 0.171853 | -0.392231 | 0.784212 | -0.391605 | -0.171355 | 0.003508 | 0.181498 | -0.504978 | 1.008689 | -0.503961 | -0.181445 | 0. |
| 0.000064 | 0.174876 | -0.390834 | 0.781646 | -0.390014 | -0.173087 | 0.002599 | 0.185333 | -0.510007 | 1.019056 | -0.510458 | -0.185354 | 0. |
| -0.00048 | 0.17744 | -0.389629 | 0.779427 | -0.388306 | -0.178265 | -0.001532 | 0.188044 | -0.508325 | 1.017222 | -0.509235 | -0.188009 | 0. |
| -0.00034 | 0.179577 | -0.388713 | 0.777548 | -0.389495 | -0.182045 | 0.00088 | 0.190973 | -0.5142 | 1.028066 | -0.513868 | -0.191684 | 0. |
| -0.001017 | 0.181317 | -0.387815 | 0.775996 | -0.390236 | -0.182044 | 0.001134 | 0.192818 | -0.513615 | 1.026794 | -0.512533 | -0.195625 | 0. |
| -0.000405 | 0.182682 | -0.387309 | 0.774766 | -0.387072 | -0.185042 | 0.000176 | 0.194266 | -0.512928 | 1.025788 | -0.512208 | -0.193969 | 0. |
| -0.000398 | 0.183691 | -0.386852 | 0.773849 | -0.386693 | -0.182265 | -0.000583 | 0.196025 | -0.518545 | 1.037318 | -0.516661 | -0.196711 | 0. |
| -0.00235 | 0.184355 | -0.38619 | 0.773241 | -0.386177 | -0.183211 | 0.00308 | 0.196731 | -0.519015 | 1.036825 | -0.519146 | -0.196858 | 0. |
| -0.000285 | 0.184686 | -0.386416 | 0.772937 | -0.386055 | -0.185736 | -0.00197 | 0.197083 | -0.517904 | 1.03658 | -0.518298 | -0.197513 | 0. |
| 0.001854 | 0.184686 | -0.38681 | 0.772939 | -0.386004 | -0.187556 | -0.002381 | 0.197083 | -0.517824 | 1.03658 | -0.518715 | -0.197377 | 0. |
| -0.000517 | 0.184356 | -0.386525 | 0.773241 | -0.386484 | -0.182956 | -0.000932 | 0.196732 | -0.518231 | 1.036826 | -0.519741 | -0.196034 | 0. |
| 0.001187 | 0.183691 | -0.387142 | 0.77385 | -0.38922 | -0.18381 | -0.000673 | 0.196024 | -0.518528 | 1.037318 | -0.518522 | -0.198639 | 0. |
| 0.00065 | 0.182682 | -0.387502 | 0.774768 | -0.386933 | -0.181103 | 0.002641 | 0.194265 | -0.513404 | 1.025788 | -0.511551 | -0.193023 | 0. |
| 0.000758 | 0.181317 | -0.388135 | 0.775996 | -0.386648 | -0.180858 | -0.000931 | 0.192818 | -0.513219 | 1.026795 | -0.515433 | -0.191319 | 0. |
| 0.001241 | 0.179577 | -0.388995 | 0.777547 | -0.387292 | -0.177324 | 0.000549 | 0.190973 | -0.514137 | 1.028065 | -0.514521 | -0.1891 | 0. |
| -0.00128 | 0.17744 | -0.389488 | 0.779429 | -0.390081 | -0.180487 | -0.002372 | 0.188043 | -0.508168 | 1.017223 | -0.508697 | -0.187058 | 0. |
| 0.000843 | 0.174876 | -0.39097 | 0.781647 | -0.39129 | -0.171283 | -0.000587 | 0.185334 | -0.50942 | 1.019056 | -0.510779 | -0.18386 | 0. |
| -0.001871 | 0.171853 | -0.391786 | 0.784212 | -0.393635 | -0.172027 | -0.000063 | 0.181499 | -0.504333 | 1.008688 | -0.502954 | -0.180477 | 0. |
| -0.000336 | 0.16833 | -0.39351 | 0.787131 | -0.394655 | -0.168313 | 0.000401 | 0.177786 | -0.505635 | 1.011127 | -0.506687 | -0.177038 | 0. |
| 0.001213 | 0.16426 | -0.395406 | 0.790416 | -0.39565 | -0.164226 | -0.002007 | 0.172889 | -0.500294 | 1.001278 | -0.499516 | -0.174615 | 0. |
| -0.001337 | 0.159588 | -0.39682 | 0.794064 | -0.397189 | -0.15896 | -0.000048 | 0.167982 | -0.502173 | 1.004361 | -0.501245 | -0.167053 | 0. |
| -0.000916 | 0.154254 | -0.398899 | 0.798079 | -0.40012 | -0.155067 | -0.0003 | 0.161809 | -0.497484 | 0.995066 | -0.499759 | -0.162346 | 0. |
| 0.001712 | 0.148184 | -0.401478 | 0.802451 | -0.401644 | -0.149047 | -0.00033 | 0.154909 | -0.492945 | 0.985992 | -0.492966 | -0.154387 | 0. |
| -0.001018 | 0.141299 | -0.403441 | 0.807168 | -0.403723 | -0.140648 | 0.000097 | 0.147205 | -0.48858 | 0.977132 | -0.489106 | -0.146121 | 0. |
| 0.000488 | 0.133504 | -0.406164 | 0.812199 | -0.404176 | -0.132869 | -0.000308 | 0.138611 | -0.484192 | 0.968468 | -0.485003 | -0.142217 | 0. |
| -0.000268 | 0.124696 | -0.408716 | 0.817498 | -0.405403 | -0.124161 | -0.000736 | 0.129024 | -0.47989 | 0.959969 | -0.480923 | -0.13116 | 0. |
| 0.000542 | 0.114756 | -0.411561 | 0.822999 | -0.411507 | -0.11395 | -0.000591 | 0.118334 | -0.475716 | 0.951573 | -0.475635 | -0.118748 | 0. |
| 0.000521 | 0.103551 | -0.414357 | 0.828605 | -0.413254 | -0.104282 | -0.000057 | 0.106416 | -0.471595 | 0.943203 | -0.470709 | -0.102624 | 0. |
| 0.003168 | 0.090928 | -0.417379 | 0.834184 | -0.418085 | -0.091441 | -0.00076 | 0.093126 | -0.467297 | 0.934735 | -0.467678 | -0.092939 | 0. |
| -0.000578 | 0.076724 | -0.419732 | 0.839552 | -0.418633 | -0.074676 | 0.001384 | 0.07831 | -0.463106 | 0.925997 | -0.463774 | -0.07555 | 0. |

**TABLE 8.** *(Continued.)* **Best joint vectors obtained.**

| $q_1$ | $q_2$ | $q_3$ | $q_4$ | $q_5$ | $q_6$ | $q_7$ | $q_8$ | $q_9$ | $q_{10}$ | $q_{11}$ | $q_{12}$ | $OF[m]$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0.001074 | 0.060747 | -0.422297 | 0.844465 | -0.42083 | -0.061804 | 0.000512 | 0.06158 | -0.451394 | 0.902724 | -0.453389 | -0.063559 | 0. |
| -0.001673 | 0.04279 | -0.424227 | 0.848597 | -0.423717 | -0.041061 | 0.000628 | 0.043229 | -0.446265 | 0.892475 | -0.44609 | -0.043396 | 0. |
| -0.001909 | 0.022624 | -0.425713 | 0.851511 | -0.426955 | -0.021453 | -0.00111 | 0.022701 | -0.433141 | 0.866332 | -0.432191 | -0.021096 | 0. |
| -0.00048 | -0. | -0.426319 | 0.852638 | -0.424613 | -0.002039 | -0.000564 | 0. | -0.42632 | 0.852641 | -0.42867 | 0.000625 | 0. |
| -0.001059 | -0.022701 | -0.433182 | 0.866333 | -0.43535 | 0.022525 | 0.002211 | -0.022624 | -0.425697 | 0.851511 | -0.426617 | 0.020715 | 0. |
| 0.001765 | -0.043228 | -0.446126 | 0.892475 | -0.446368 | 0.044806 | -0.000373 | -0.04279 | -0.42428 | 0.848597 | -0.424667 | 0.044687 | 0. |
| -0.002639 | -0.06157 | -0.454864 | 0.902702 | -0.448296 | 0.062579 | 0.002573 | -0.060746 | -0.421998 | 0.844464 | -0.422389 | 0.062645 | 0. |
| 0.000175 | -0.078311 | -0.469696 | 0.925905 | -0.455549 | 0.075747 | 0.001334 | -0.076723 | -0.419535 | 0.839552 | -0.416994 | 0.076473 | 0. |
| 0.000728 | -0.093134 | -0.477345 | 0.934533 | -0.455219 | 0.092211 | 0.003523 | -0.090928 | -0.416552 | 0.834183 | -0.417538 | 0.091067 | 0. |
| 0.00008 | -0.106416 | -0.484943 | 0.942844 | -0.457984 | 0.108111 | 0.000251 | -0.10355 | -0.413957 | 0.828605 | -0.415653 | 0.106763 | 0. |
| -0.002443 | -0.118285 | -0.496047 | 0.950767 | -0.456018 | 0.118957 | -0.003468 | -0.114757 | -0.411456 | 0.822998 | -0.411467 | 0.115375 | 0. |
| -0.00029 | -0.129017 | -0.503228 | 0.958883 | -0.454083 | 0.131343 | 0.000174 | -0.124696 | -0.408144 | 0.817499 | -0.411205 | 0.124297 | 0. |
| -0.0002 | -0.138605 | -0.513968 | 0.966686 | -0.45383 | 0.136062 | -0.000933 | -0.133505 | -0.405472 | 0.812198 | -0.406178 | 0.131205 | 0. |
| 0.001606 | -0.147265 | -0.524458 | 0.974492 | -0.448376 | 0.146148 | 0.000471 | -0.141298 | -0.402572 | 0.807167 | -0.403244 | 0.140756 | 0. |
| -0.001923 | -0.154825 | -0.535759 | 0.982329 | -0.447317 | 0.154852 | -0.002774 | -0.148187 | -0.400467 | 0.802451 | -0.400803 | 0.148328 | 0. |
| -0.001935 | -0.161711 | -0.546576 | 0.990231 | -0.444823 | 0.162327 | 0.001216 | -0.154252 | -0.397429 | 0.798075 | -0.398937 | 0.152233 | 0. |
| 0.000786 | -0.168027 | -0.556968 | 0.998209 | -0.442083 | 0.169184 | -0.000463 | -0.159589 | -0.395392 | 0.794057 | -0.400351 | 0.162011 | 0. |
| -0.002713 | -0.172706 | -0.564955 | 0.992748 | -0.427255 | 0.172841 | 0.001137 | -0.164257 | -0.392979 | 0.790407 | -0.398475 | 0.162808 | 0. |
| 0.001767 | -0.177917 | -0.575072 | 1.000914 | -0.42614 | 0.17866 | 0.001424 | -0.168326 | -0.390913 | 0.787118 | -0.395358 | 0.165789 | 0. |
| 0.000493 | -0.181541 | -0.582605 | 0.995496 | -0.414038 | 0.180349 | -0.000452 | -0.171855 | -0.389348 | 0.784192 | -0.398564 | 0.17239 | 0. |
| -0.002616 | -0.185088 | -0.597021 | 1.002648 | -0.406074 | 0.185375 | 0.001438 | -0.174872 | -0.387263 | 0.781619 | -0.396524 | 0.175876 | 0. |
| 0.001782 | -0.188228 | -0.60334 | 0.997136 | -0.393785 | 0.189497 | -0.000376 | -0.177441 | -0.385938 | 0.779394 | -0.394755 | 0.178378 | 0. |
| -0.001217 | -0.190839 | -0.614828 | 1.005589 | -0.390967 | 0.189647 | 0.001007 | -0.179573 | -0.384151 | 0.777502 | -0.394038 | 0.180467 | 0. |
| 0.001176 | -0.192959 | -0.621378 | 1.000113 | -0.378977 | 0.193217 | -0.000082 | -0.181318 | -0.382894 | 0.775933 | -0.392347 | 0.181149 | 0. |
| 0.000732 | -0.19436 | -0.628331 | 0.994575 | -0.365104 | 0.195831 | 0.000442 | -0.18268 | -0.381423 | 0.774683 | -0.394845 | 0.185184 | 0. |
| -0.001609 | -0.1958 | -0.642274 | 1.001537 | -0.358552 | 0.196638 | 0.00103 | -0.183684 | -0.380003 | 0.77374 | -0.393362 | 0.182388 | 0. |
| -0.000223 | -0.196699 | -0.648635 | 0.995941 | -0.345322 | 0.195478 | -0.000738 | -0.184361 | -0.379062 | 0.7731 | -0.398767 | 0.184472 | 0. |
| -0.001673 | -0.196819 | -0.655397 | 0.990314 | -0.334837 | 0.196131 | -0.000479 | -0.18469 | -0.377783 | 0.772753 | -0.395741 | 0.187991 | 0. |
| -0.000611 | -0.196979 | -0.663851 | 0.982454 | -0.319033 | 0.19957 | 0.001293 | -0.184673 | -0.376244 | 0.772699 | -0.394796 | 0.186329 | 0. |
| 0.001218 | -0.196951 | -0.669583 | 0.9767 | -0.306915 | 0.194184 | -0.001058 | -0.184367 | -0.375461 | 0.772933 | -0.397287 | 0.185923 | 0. |
| 0.000001 | -0.196025 | -0.675753 | 0.97099 | -0.293392 | 0.195525 | -0.000375 | -0.183696 | -0.374106 | 0.773455 | -0.397981 | 0.182697 | 0. |
| -0.000612 | -0.194145 | -0.674538 | 0.952487 | -0.278847 | 0.194046 | -0.001595 | -0.182705 | -0.373058 | 0.77426 | -0.400983 | 0.182501 | 0. |
| -0.00068 | -0.192678 | -0.680118 | 0.94693 | -0.268397 | 0.193202 | -0.002094 | -0.18135 | -0.371818 | 0.77535 | -0.403978 | 0.18148 | 0. |
| -0.000303 | -0.190909 | -0.685464 | 0.941544 | -0.252476 | 0.190669 | 0.002529 | -0.179531 | -0.369575 | 0.776726 | -0.407814 | 0.179171 | 0. |
| -0.002163 | -0.187571 | -0.681884 | 0.926216 | -0.244267 | 0.18617 | -0.000001 | -0.17744 | -0.368502 | 0.778383 | -0.413675 | 0.177521 | 0. |
| -0.001116 | -0.185082 | -0.686837 | 0.921421 | -0.23487 | 0.181678 | 0.001369 | -0.174845 | -0.366596 | 0.780325 | -0.411877 | 0.174368 | 0. |
| -0.000822 | -0.181308 | -0.6843 | 0.903554 | -0.221624 | 0.180313 | 0.001931 | -0.171802 | -0.364653 | 0.782539 | -0.41595 | 0.170733 | 0. |
| -0.001525 | -0.177421 | -0.689268 | 0.899528 | -0.210306 | 0.176543 | -0.000245 | -0.168337 | -0.362948 | 0.785018 | -0.42072 | 0.168322 | 0. |
| 0.001186 | -0.173177 | -0.684273 | 0.885801 | -0.200338 | 0.174396 | 0.001066 | -0.164225 | -0.360381 | 0.787749 | -0.426395 | 0.16567 | 0. |
| 0.001812 | -0.168433 | -0.688811 | 0.883008 | -0.193498 | 0.170075 | -0.001379 | -0.159639 | -0.358085 | 0.790704 | -0.432348 | 0.162524 | 0. |
| -0.00208 | -0.161286 | -0.684688 | 0.870645 | -0.187027 | 0.163253 | -0.002365 | -0.154352 | -0.355132 | 0.793851 | -0.439411 | 0.153399 | 0. |
| 0.001145 | -0.155198 | -0.679264 | 0.85905 | -0.178942 | 0.15612 | -0.000906 | -0.148227 | -0.351309 | 0.797137 | -0.447113 | 0.149796 | 0. |
| 0.000344 | -0.147293 | -0.674311 | 0.848314 | -0.175098 | 0.148653 | 0.000276 | -0.141284 | -0.346944 | 0.800491 | -0.452964 | 0.143854 | 0. |
| 0.001792 | -0.139063 | -0.668847 | 0.838533 | -0.170427 | 0.138148 | -0.000936 | -0.133559 | -0.342193 | 0.803818 | -0.464655 | 0.134023 | 0. |
| 0.001359 | -0.129366 | -0.663422 | 0.82978 | -0.166158 | 0.127265 | 0.000759 | -0.124646 | -0.336184 | 0.806984 | -0.468882 | 0.124952 | 0. |
| -0.002062 | -0.117825 | -0.656653 | 0.825859 | -0.170611 | 0.116404 | 0.0024 | -0.114575 | -0.329161 | 0.809816 | -0.47953 | 0.113503 | 0. |
| 0.001492 | -0.10678 | -0.650077 | 0.819301 | -0.169225 | 0.106837 | 0.002122 | -0.103371 | -0.321098 | 0.812078 | -0.490028 | 0.101547 | 0. |
| 0.002435 | -0.093702 | -0.641852 | 0.817445 | -0.176982 | 0.093465 | -0.000333 | -0.090961 | -0.311689 | 0.813464 | -0.50118 | 0.088883 | 0. |

**TABLE 8.** *(Continued.)* **Best joint vectors obtained.**

| $q_1$ | $q_2$ | $q_3$ | $q_4$ | $q_5$ | $q_6$ | $q_7$ | $q_8$ | $q_9$ | $q_{10}$ | $q_{11}$ | $q_{12}$ | $OF[m]$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| -0.000055 | -0.078298 | -0.63313 | 0.8164 | -0.185475 | 0.080108 | 0.000473 | -0.076673 | -0.300098 | 0.813567 | -0.512946 | 0.072966 | 0. |
| -0.000777 | -0.061411 | -0.614767 | 0.800609 | -0.186947 | 0.060425 | 0.00075 | -0.060656 | -0.286299 | 0.811851 | -0.523601 | 0.058599 | 0. |
| -0.000279 | -0.043171 | -0.603255 | 0.800264 | -0.201182 | 0.043141 | 0.000475 | -0.042726 | -0.269781 | 0.807613 | -0.539059 | 0.044303 | 0. |
| 0.001181 | -0.022923 | -0.579562 | 0.78687 | -0.207938 | 0.023275 | -0.001775 | -0.022892 | -0.249933 | 0.799911 | -0.549302 | 0.021988 | 0. |
| -0.000657 | 0.000111 | -0.561639 | 0.787487 | -0.226692 | -0.001929 | -0.001064 | -0.00018 | -0.225848 | 0.787487 | -0.559375 | 0.00106 | 0. |
| -0.000318 | 0.022673 | -0.550006 | 0.799917 | -0.24811 | -0.020844 | -0.002301 | 0.022268 | -0.207215 | 0.786857 | -0.580375 | -0.023021 | 0. |
| -0.00242 | 0.043116 | -0.537684 | 0.807633 | -0.269497 | -0.041059 | 0.000647 | 0.04336 | -0.201866 | 0.803396 | -0.601334 | -0.041724 | 0. |
| -0.001589 | 0.060937 | -0.525354 | 0.811893 | -0.286658 | -0.063725 | -0.000361 | 0.061502 | -0.1907 | 0.803872 | -0.609937 | -0.062021 | 0. |
| 0.000409 | 0.076679 | -0.513361 | 0.813635 | -0.301362 | -0.080176 | 0.002269 | 0.078812 | -0.193219 | 0.823072 | -0.629567 | -0.078523 | 0. |
| -0.002833 | 0.091197 | -0.501378 | 0.81356 | -0.31223 | -0.091631 | 0.001139 | 0.093384 | -0.190231 | 0.827714 | -0.63775 | -0.091268 | 0. |
| -0.000234 | 0.10357 | -0.490482 | 0.812203 | -0.322058 | -0.102809 | -0.001546 | 0.106059 | -0.18867 | 0.833234 | -0.644718 | -0.107934 | 0. |
| 0.000809 | 0.114696 | -0.480113 | 0.809971 | -0.330481 | -0.114149 | 0.002236 | 0.118848 | -0.194352 | 0.843166 | -0.650575 | -0.119077 | 0. |
| 0.000345 | 0.124674 | -0.470259 | 0.807167 | -0.33429 | -0.123734 | 0.001375 | 0.129338 | -0.200929 | 0.854043 | -0.653979 | -0.13148 | 0. |
| 0.001408 | 0.133422 | -0.461294 | 0.804028 | -0.340639 | -0.132821 | 0.001064 | 0.138847 | -0.213782 | 0.869049 | -0.65468 | -0.139975 | 0. |
| 0.000126 | 0.141292 | -0.452702 | 0.800727 | -0.348547 | -0.137306 | -0.000226 | 0.147157 | -0.227549 | 0.884631 | -0.656501 | -0.149724 | 0. |
| 0.000696 | 0.148152 | -0.445029 | 0.797396 | -0.352266 | -0.150192 | 0.000445 | 0.155002 | -0.242525 | 0.900669 | -0.656557 | -0.156473 | 0. |
| 0.000777 | 0.154223 | -0.437923 | 0.794132 | -0.354923 | -0.153529 | 0.002335 | 0.162276 | -0.25851 | 0.917047 | -0.658634 | -0.162964 | 0. |
| 0.001118 | 0.159549 | -0.431457 | 0.791005 | -0.360365 | -0.161817 | 0.003375 | 0.168617 | -0.279752 | 0.936246 | -0.657464 | -0.168483 | 0. |
| -0.000738 | 0.164282 | -0.425196 | 0.788066 | -0.363553 | -0.167394 | 0.000852 | 0.173039 | -0.294879 | 0.942102 | -0.650392 | -0.172748 | 0. |
| -0.000814 | 0.168352 | -0.419735 | 0.78535 | -0.364532 | -0.167574 | 0.001376 | 0.17801 | -0.31689 | 0.96095 | -0.643817 | -0.177957 | 0. |
| 0.001883 | 0.171809 | -0.415224 | 0.782879 | -0.368291 | -0.169243 | 0.001148 | 0.18167 | -0.33312 | 0.966533 | -0.634307 | -0.184498 | 0. |
| -0.000757 | 0.174891 | -0.410238 | 0.780671 | -0.367875 | -0.174264 | 0.003181 | 0.185753 | -0.360257 | 0.986423 | -0.625089 | -0.186229 | 0. |
| 0.000997 | 0.177423 | -0.40641 | 0.778735 | -0.370526 | -0.177315 | 0.000461 | 0.188098 | -0.376309 | 0.991205 | -0.614563 | -0.18622 | 0. |
| 0.001556 | 0.179556 | -0.402732 | 0.777074 | -0.375021 | -0.180143 | 0.001568 | 0.19113 | -0.40321 | 1.009471 | -0.607486 | -0.190937 | 0. |
| -0.001781 | 0.181336 | -0.398673 | 0.775691 | -0.377723 | -0.179719 | -0.00174 | 0.192675 | -0.423187 | 1.014287 | -0.591663 | -0.192812 | 0. |
| 0.000058 | 0.182682 | -0.395829 | 0.774588 | -0.37751 | -0.183799 | -0.000681 | 0.194222 | -0.44373 | 1.018191 | -0.571855 | -0.192434 | 0. |
| 0.001698 | 0.18368 | -0.393202 | 0.773761 | -0.383219 | -0.183359 | -0.000265 | 0.196013 | -0.469816 | 1.033461 | -0.566641 | -0.195473 | 0. |
| -0.000631 | 0.184358 | -0.390065 | 0.77321 | -0.381926 | -0.185307 | 0.00256 | 0.196802 | -0.490059 | 1.035434 | -0.546557 | -0.197204 | 0. |
| -0.000392 | 0.184687 | -0.387583 | 0.772936 | -0.383769 | -0.183364 | 0.001242 | 0.197095 | -0.509062 | 1.036425 | -0.528982 | -0.196545 | 0. |
| 0.000497 | 0.184687 | -0.385371 | 0.772933 | -0.385364 | -0.184174 | -0.001636 | 0.197098 | -0.527285 | 1.036424 | -0.506588 | -0.197814 | 0. |
| -0.001087 | 0.184352 | -0.38283 | 0.77321 | -0.390561 | -0.186753 | -0.003724 | 0.196834 | -0.545148 | 1.035434 | -0.490351 | -0.197613 | 0. |
| 0.001459 | 0.183699 | -0.381136 | 0.773762 | -0.391869 | -0.184138 | 0.002659 | 0.195902 | -0.56411 | 1.03346 | -0.468192 | -0.195391 | 0. |
| -0.000004 | 0.182682 | -0.378768 | 0.774587 | -0.396821 | -0.184898 | 0.000345 | 0.194243 | -0.574397 | 1.018192 | -0.443716 | -0.1911 | 0. |
| -0.00155 | 0.1813 | -0.376418 | 0.775692 | -0.397393 | -0.180228 | 0.001739 | 0.192675 | -0.591099 | 1.014286 | -0.425739 | -0.192715 | 0. |
| -0.00095 | 0.179564 | -0.37445 | 0.777075 | -0.402402 | -0.180994 | -0.002226 | 0.191196 | -0.606136 | 1.00947 | -0.403463 | -0.18987 | 0. |
| -0.000513 | 0.177431 | -0.372411 | 0.778736 | -0.403203 | -0.175653 | -0.000656 | 0.188121 | -0.61486 | 0.991205 | -0.376346 | -0.185035 | 0. |
| -0.001002 | 0.174857 | -0.370127 | 0.780671 | -0.407218 | -0.175203 | 0.002223 | 0.18504 | -0.627162 | 0.986423 | -0.358686 | -0.185791 | 0. |
| 0.001378 | 0.171885 | -0.368213 | 0.782879 | -0.414077 | -0.169982 | 0.001139 | 0.181329 | -0.633826 | 0.966533 | -0.332809 | -0.180917 | 0. |
| -0.001104 | 0.1683 | -0.365295 | 0.785352 | -0.420442 | -0.165902 | -0.000996 | 0.177949 | -0.644127 | 0.96095 | -0.315864 | -0.177977 | 0. |
| -0.000341 | 0.164249 | -0.362695 | 0.788067 | -0.424749 | -0.160372 | 0.001423 | 0.172639 | -0.647615 | 0.942103 | -0.29607 | -0.173612 | 0. |
| 0.000035 | 0.15959 | -0.35973 | 0.791005 | -0.428837 | -0.157837 | -0.000003 | 0.167983 | -0.657059 | 0.936247 | -0.279491 | -0.166552 | 0. |
| 0.000018 | 0.154255 | -0.356331 | 0.794133 | -0.438141 | -0.153327 | 0.001105 | 0.161587 | -0.659093 | 0.917048 | -0.260074 | -0.161549 | 0. |
| -0.002306 | 0.148078 | -0.35213 | 0.797396 | -0.445084 | -0.146802 | -0.000062 | 0.154922 | -0.658203 | 0.900669 | -0.240176 | -0.154581 | 0. |
| -0.000451 | 0.141275 | -0.34798 | 0.800727 | -0.453048 | -0.138205 | 0.0023 | 0.146709 | -0.657385 | 0.88463 | -0.22827 | -0.146916 | 0. |
| -0.001492 | 0.133417 | -0.342722 | 0.804028 | -0.461842 | -0.131564 | -0.001294 | 0.138898 | -0.655234 | 0.869048 | -0.213136 | -0.137128 | 0. |
| -0.001565 | 0.124593 | -0.336757 | 0.807168 | -0.471869 | -0.123417 | 0.000755 | 0.128852 | -0.653388 | 0.854043 | -0.198432 | -0.129109 | 0. |
| -0.003038 | 0.114529 | -0.329603 | 0.809971 | -0.480835 | -0.115278 | 0.002839 | 0.117681 | -0.649413 | 0.843166 | -0.195555 | -0.118108 | 0. |
| -0.000828 | 0.103481 | -0.321612 | 0.812204 | -0.490113 | -0.102486 | 0.002825 | 0.105764 | -0.6447 | 0.833234 | -0.188192 | -0.105139 | 0. |

**TABLE 8.** *(Continued.)* **Best joint vectors obtained.**

| $q_1$ | $q_2$ | $q_3$ | $q_4$ | $q_5$ | $q_6$ | $q_7$ | $q_8$ | $q_9$ | $q_{10}$ | $q_{11}$ | $q_{12}$ | $OF[m]$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| -0.00182 | 0.090757 | -0.311758 | 0.813558 | -0.500412 | -0.091824 | -0.00072 | 0.093289 | -0.637522 | 0.827714 | -0.194068 | -0.093323 | 0. |
| -0.0024 | 0.076467 | -0.300123 | 0.813636 | -0.511335 | -0.076599 | 0.000727 | 0.078149 | -0.630088 | 0.823072 | -0.192668 | -0.078635 | 0. |
| 0.000106 | 0.060759 | -0.286449 | 0.811894 | -0.523286 | -0.05858 | -0.001359 | 0.06187 | -0.613064 | 0.80387 | -0.191422 | -0.062392 | 0. |
| -0.002104 | 0.042506 | -0.269756 | 0.807634 | -0.5369 | -0.043119 | -0.00304 | 0.043844 | -0.601426 | 0.803397 | -0.202655 | -0.0445 | 0. |
| -0.000522 | 0.022545 | -0.24989 | 0.799915 | -0.549165 | -0.024536 | 0.00039 | 0.022628 | -0.579601 | 0.786861 | -0.206092 | -0.021487 | 0. |
| -0.003279 | -0.000556 | -0.225849 | 0.787489 | -0.562189 | -0.000447 | 0.000725 | -0.000123 | -0.56164 | 0.787488 | -0.225615 | -0.002929 | 0. |
| 0.001738 | -0.022374 | -0.207229 | 0.786861 | -0.581281 | 0.021479 | -0.001979 | -0.022325 | -0.550059 | 0.799918 | -0.249728 | 0.022792 | 0. |
| 0.001544 | -0.042916 | -0.201772 | 0.803396 | -0.601582 | 0.045009 | -0.002461 | -0.042458 | -0.537893 | 0.807634 | -0.270573 | 0.043346 | 0. |
| 0.000198 | -0.061537 | -0.19071 | 0.803871 | -0.611199 | 0.061729 | 0.001629 | -0.060941 | -0.525353 | 0.811896 | -0.286151 | 0.061333 | 0. |
| -0.001293 | -0.078596 | -0.193143 | 0.823073 | -0.630439 | 0.08122 | 0.000372 | -0.076763 | -0.513301 | 0.813635 | -0.300082 | 0.074441 | 0. |
| -0.000692 | -0.093283 | -0.19019 | 0.827714 | -0.64098 | 0.093061 | 0.004174 | -0.091323 | -0.501256 | 0.813561 | -0.312295 | 0.09198 | 0. |
| 0.00357 | -0.105592 | -0.188456 | 0.833235 | -0.643852 | 0.105194 | 0.000816 | -0.103619 | -0.490423 | 0.812205 | -0.319801 | 0.103632 | 0. |
| 0.001482 | -0.117993 | -0.193912 | 0.843166 | -0.649231 | 0.115424 | -0.000667 | -0.114706 | -0.480096 | 0.809969 | -0.330699 | 0.117718 | 0. |
| 0.00028 | -0.12896 | -0.200716 | 0.854044 | -0.651129 | 0.129329 | -0.001793 | -0.124577 | -0.47044 | 0.807169 | -0.336814 | 0.121711 | 0. |
| -0.000972 | -0.138827 | -0.213769 | 0.869048 | -0.656108 | 0.138226 | 0.000108 | -0.133511 | -0.461091 | 0.804026 | -0.342758 | 0.133554 | 0. |
| -0.002102 | -0.147659 | -0.227891 | 0.88463 | -0.657646 | 0.14645 | 0.000196 | -0.141309 | -0.452657 | 0.800728 | -0.351362 | 0.141251 | 0. |
| 0.001501 | -0.154596 | -0.242225 | 0.900669 | -0.658137 | 0.156697 | 0.000321 | -0.148199 | -0.444879 | 0.797397 | -0.352548 | 0.151495 | 0. |
| 0.001006 | -0.161607 | -0.257971 | 0.917047 | -0.656949 | 0.162485 | -0.003459 | -0.154114 | -0.438335 | 0.794132 | -0.355933 | 0.15512 | 0. |
| -0.002008 | -0.16836 | -0.279524 | 0.936248 | -0.65682 | 0.167044 | 0.001128 | -0.159628 | -0.4311 | 0.791004 | -0.358466 | 0.158647 | 0. |
| 0.001359 | -0.172651 | -0.294499 | 0.942103 | -0.648445 | 0.173454 | 0.001707 | -0.164312 | -0.425036 | 0.788064 | -0.362448 | 0.16243 | 0. |
| 0.000743 | -0.177665 | -0.316516 | 0.960951 | -0.645073 | 0.180412 | -0.000893 | -0.168306 | -0.42002 | 0.785348 | -0.367797 | 0.16822 | 0. |
| 0.000059 | -0.18149 | -0.332901 | 0.966533 | -0.635061 | 0.183625 | 0.004608 | -0.171958 | -0.414114 | 0.78288 | -0.368819 | 0.172108 | 0. |
| 0.001043 | -0.185197 | -0.359478 | 0.986423 | -0.626938 | 0.186486 | 0.000615 | -0.174888 | -0.410263 | 0.780672 | -0.370809 | 0.17547 | 0. |
| -0.000196 | -0.188067 | -0.376259 | 0.991204 | -0.615971 | 0.185602 | -0.000559 | -0.177431 | -0.406333 | 0.778735 | -0.370595 | 0.177406 | 0. |
| -0.004434 | -0.191416 | -0.403754 | 1.009471 | -0.6059 | 0.191662 | 0.001126 | -0.179593 | -0.402253 | 0.777074 | -0.375756 | 0.176974 | 0. |
| 0.001544 | -0.192691 | -0.423224 | 1.014286 | -0.591042 | 0.193208 | -0.000739 | -0.181309 | -0.399127 | 0.775691 | -0.377068 | 0.181226 | 0. |
| -0.002633 | -0.194434 | -0.44437 | 1.018192 | -0.575381 | 0.19334 | -0.002612 | -0.18266 | -0.396293 | 0.774587 | -0.378279 | 0.182093 | 0. |
| 0.001086 | -0.195975 | -0.469657 | 1.033461 | -0.563296 | 0.194471 | 0.000138 | -0.183692 | -0.392867 | 0.773763 | -0.380102 | 0.187098 | 0. |
| 0.001807 | -0.196682 | -0.489204 | 1.035434 | -0.544481 | 0.195901 | -0.000847 | -0.184352 | -0.390338 | 0.773213 | -0.385216 | 0.182303 | 0. |
| -0.001467 | -0.197097 | -0.509106 | 1.036424 | -0.527194 | 0.194216 | -0.001191 | -0.184685 | -0.387873 | 0.772935 | -0.385242 | 0.184245 | 0. |
| -0.003871 | -0.197046 | -0.528363 | 1.036424 | -0.508337 | 0.197553 | -0.001011 | -0.184687 | -0.385466 | 0.772935 | -0.38732 | 0.184623 | 0. |
| 0.000806 | -0.196754 | -0.545719 | 1.035434 | -0.487703 | 0.198478 | -0.002337 | -0.184364 | -0.383458 | 0.773211 | -0.390183 | 0.183261 | 0. |
| 0.000279 | -0.196037 | -0.563538 | 1.033461 | -0.472255 | 0.197233 | 0.003016 | -0.183672 | -0.380319 | 0.773762 | -0.394676 | 0.183553 | 0. |
| 0.000908 | -0.194323 | -0.574155 | 1.018191 | -0.446448 | 0.194741 | 0.002635 | -0.18266 | -0.37829 | 0.774587 | -0.397656 | 0.182924 | 0. |
| -0.002595 | -0.192604 | -0.591264 | 1.014287 | -0.423298 | 0.193247 | 0.002491 | -0.18129 | -0.376249 | 0.775694 | -0.399487 | 0.182433 | 0. |
| 0.001664 | -0.19114 | -0.606243 | 1.00947 | -0.402553 | 0.193121 | -0.00052 | -0.179584 | -0.374713 | 0.777075 | -0.403005 | 0.182414 | 0. |
| 0.001079 | -0.188171 | -0.614781 | 0.991205 | -0.376237 | 0.188516 | 0.000857 | -0.177426 | -0.372349 | 0.778734 | -0.40614 | 0.178267 | 0. |
| 0.000236 | -0.185365 | -0.62671 | 0.986424 | -0.358691 | 0.187897 | -0.000694 | -0.17489 | -0.370422 | 0.780671 | -0.410711 | 0.171768 | 0. |
| 0.000777 | -0.181615 | -0.63348 | 0.966532 | -0.331578 | 0.180229 | 0.001619 | -0.171815 | -0.3677 | 0.782879 | -0.41341 | 0.171132 | 0. |
| -0.000912 | -0.177638 | -0.644465 | 0.96095 | -0.316062 | 0.175303 | 0.000706 | -0.168311 | -0.36536 | 0.785349 | -0.419914 | 0.167024 | 0. |
| -0.000867 | -0.172737 | -0.64752 | 0.942102 | -0.292832 | 0.17474 | -0.000955 | -0.164289 | -0.362905 | 0.788065 | -0.42724 | 0.163847 | 0. |
| -0.00068 | -0.167854 | -0.657173 | 0.936247 | -0.281373 | 0.167438 | 0.001964 | -0.159519 | -0.359412 | 0.791005 | -0.432325 | 0.159842 | 0. |
| 0.000785 | -0.161966 | -0.658788 | 0.917048 | -0.258979 | 0.164255 | 0.000703 | -0.154226 | -0.35622 | 0.794131 | -0.436452 | 0.154769 | 0. |
| -0.000224 | -0.154862 | -0.658247 | 0.900669 | -0.242294 | 0.153309 | 0.00242 | -0.148074 | -0.352113 | 0.797395 | -0.445432 | 0.149487 | 0. |
| -0.002171 | -0.146737 | -0.657366 | 0.88463 | -0.229375 | 0.147215 | -0.000953 | -0.141348 | -0.348177 | 0.800726 | -0.452591 | 0.139152 | 0. |
| -0.000206 | -0.138565 | -0.655441 | 0.869048 | -0.213746 | 0.139163 | -0.000031 | -0.133506 | -0.342925 | 0.804028 | -0.462078 | 0.133637 | 0. |
| 0.000004 | -0.129025 | -0.653291 | 0.854044 | -0.203193 | 0.12896 | 0.00126 | -0.124613 | -0.336795 | 0.807168 | -0.472929 | 0.124598 | 0. |
| -0.000734 | -0.118166 | -0.649166 | 0.843166 | -0.191377 | 0.117603 | 0.002017 | -0.114605 | -0.32972 | 0.809971 | -0.480683 | 0.115721 | 0. |

**TABLE 8.** *(Continued.)* **Best joint vectors obtained.**

| $q_1$ | $q_2$ | $q_3$ | $q_4$ | $q_5$ | $q_6$ | $q_7$ | $q_8$ | $q_9$ | $q_{10}$ | $q_{11}$ | $q_{12}$ | $OF[m]$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| -0.000828 | -0.106224 | -0.644489 | 0.833235 | -0.186597 | 0.104618 | 0.000787 | -0.103484 | -0.321616 | 0.812204 | -0.493044 | 0.104649 | 0. |
| 0.002331 | -0.093654 | -0.637371 | 0.827713 | -0.190934 | 0.093329 | 0.001409 | -0.090795 | -0.311798 | 0.813562 | -0.502107 | 0.090587 | 0. |
| 0.001273 | -0.078592 | -0.62993 | 0.82307 | -0.192981 | 0.080356 | -0.000821 | -0.076811 | -0.30037 | 0.813637 | -0.512874 | 0.076703 | 0. |
| -0.001242 | -0.061314 | -0.613225 | 0.803872 | -0.189581 | 0.062992 | 0.003702 | -0.060302 | -0.28622 | 0.811895 | -0.526159 | 0.060761 | 0. |
| -0.001513 | -0.042923 | -0.601623 | 0.803397 | -0.20165 | 0.044601 | 0.002187 | -0.042495 | -0.269753 | 0.807634 | -0.537107 | 0.040516 | 0. |
| -0.002483 | -0.022233 | -0.579649 | 0.786861 | -0.208461 | 0.023312 | 0.000423 | -0.02256 | -0.249894 | 0.799918 | -0.546162 | 0.021883 | 0. |
| 0.002257 | -0.000382 | -0.561639 | 0.787488 | -0.225151 | 0.001108 | 0.00078 | 0.000132 | -0.225848 | 0.787487 | -0.561922 | -0.00177 | 0. |
| 0.001338 | 0.022422 | -0.550049 | 0.799911 | -0.251818 | -0.021332 | 0.002157 | 0.023107 | -0.207329 | 0.786866 | -0.580926 | -0.024384 | 0. |
| -0.00211 | 0.043074 | -0.537721 | 0.807612 | -0.269038 | -0.042519 | 0.001197 | 0.043476 | -0.197073 | 0.800263 | -0.603255 | -0.042803 | 0. |
| 0.000096 | 0.060735 | -0.525512 | 0.811852 | -0.282301 | -0.060467 | 0.000858 | 0.061766 | -0.185943 | 0.80061 | -0.615685 | -0.061079 | 0. |
| 0.000237 | 0.076699 | -0.513451 | 0.813567 | -0.300497 | -0.076908 | -0.003109 | 0.0776 | -0.183033 | 0.8164 | -0.634594 | -0.077164 | 0. |
| 0.00006 | 0.090923 | -0.501811 | 0.813463 | -0.312795 | -0.090627 | -0.000924 | 0.092908 | -0.17528 | 0.817446 | -0.641974 | -0.094115 | 0. |
| 0.001479 | 0.103425 | -0.490914 | 0.812079 | -0.320868 | -0.104598 | -0.001875 | 0.105958 | -0.168868 | 0.819302 | -0.651525 | -0.105573 | 0. |
| -0.000312 | 0.11478 | -0.480345 | 0.809816 | -0.327597 | -0.112152 | 0.001391 | 0.118677 | -0.169613 | 0.825859 | -0.655244 | -0.117087 | 0. |
| -0.000852 | 0.124753 | -0.4706 | 0.806985 | -0.333902 | -0.124209 | -0.003263 | 0.128201 | -0.165765 | 0.829781 | -0.664105 | -0.129021 | 0. |
| -0.00048 | 0.133533 | -0.461687 | 0.80382 | -0.343768 | -0.132646 | 0.002699 | 0.139292 | -0.169812 | 0.838532 | -0.67009 | -0.13871 | 0. |
| 0.000009 | 0.141299 | -0.45351 | 0.800492 | -0.347875 | -0.139928 | -0.000246 | 0.147144 | -0.173917 | 0.848316 | -0.67068 | -0.14648 | 0. |
| 0.000458 | 0.148163 | -0.44603 | 0.797139 | -0.351818 | -0.147358 | -0.000816 | 0.154703 | -0.179483 | 0.85905 | -0.682274 | -0.154906 | 0. |
| 0.00066 | 0.154227 | -0.439184 | 0.793851 | -0.353535 | -0.15608 | 0.000265 | 0.161875 | -0.186335 | 0.870646 | -0.684254 | -0.161538 | 0. |
| 0.002772 | 0.159485 | -0.433279 | 0.790705 | -0.357772 | -0.160065 | -0.00103 | 0.167725 | -0.193721 | 0.883007 | -0.686648 | -0.1682 | 0. |
| 0.000483 | 0.164244 | -0.427273 | 0.787749 | -0.358903 | -0.164895 | 0.000271 | 0.172955 | -0.201369 | 0.885799 | -0.685093 | -0.172441 | 0. |
| -0.000292 | 0.168338 | -0.422063 | 0.785019 | -0.364196 | -0.166998 | 0.002441 | 0.178372 | -0.210962 | 0.899528 | -0.689466 | -0.177398 | 0. |
| -0.000426 | 0.171864 | -0.417484 | 0.78254 | -0.363439 | -0.172646 | 0.001369 | 0.181818 | -0.21965 | 0.903554 | -0.682106 | -0.18046 | 0. |
| -0.000411 | 0.174886 | -0.41342 | 0.780324 | -0.36534 | -0.175446 | -0.000811 | 0.18515 | -0.23464 | 0.92142 | -0.687341 | -0.185794 | 0. |
| -0.001454 | 0.177469 | -0.409625 | 0.778383 | -0.367201 | -0.17927 | -0.000047 | 0.188034 | -0.244726 | 0.926214 | -0.679515 | -0.188934 | 0.000001 |
| -0.000014 | 0.179578 | -0.406697 | 0.776726 | -0.371367 | -0.182476 | -0.002347 | 0.19047 | -0.255693 | 0.941544 | -0.686647 | -0.191886 | 0. |
| 0.000685 | 0.181306 | -0.404034 | 0.775351 | -0.374135 | -0.182521 | -0.00014 | 0.192789 | -0.266915 | 0.946931 | -0.681732 | -0.190498 | 0. |
| 0.00123 | 0.182665 | -0.401715 | 0.774259 | -0.375963 | -0.182844 | 0.000431 | 0.194351 | -0.278151 | 0.952489 | -0.671192 | -0.193658 | 0. |
| -0.000209 | 0.183693 | -0.399379 | 0.773455 | -0.370259 | -0.183153 | 0.002341 | 0.196466 | -0.295693 | 0.970991 | -0.676293 | -0.19664 | 0. |
| 0.000438 | 0.184351 | -0.397746 | 0.772934 | -0.373809 | -0.182671 | 0.000083 | 0.196747 | -0.306895 | 0.976701 | -0.670935 | -0.197768 | 0. |
| 0.002758 | 0.184659 | -0.396725 | 0.7727 | -0.374717 | -0.184652 | -0.001114 | 0.196893 | -0.318505 | 0.982454 | -0.663742 | -0.194178 | 0. |
| 0.000343 | 0.184683 | -0.395121 | 0.772754 | -0.378003 | -0.185483 | -0.000781 | 0.19696 | -0.335093 | 0.990315 | -0.653719 | -0.19913 | 0. |
| 0.000008 | 0.184356 | -0.394173 | 0.773097 | -0.378746 | -0.184848 | 0.000248 | 0.196769 | -0.347398 | 0.995941 | -0.649511 | -0.197805 | 0. |
| 0.001832 | 0.183678 | -0.393883 | 0.773739 | -0.381815 | -0.183393 | 0.001724 | 0.196265 | -0.359914 | 1.001539 | -0.645621 | -0.196392 | 0. |
| -0.002432 | 0.182696 | -0.392737 | 0.774682 | -0.383547 | -0.182016 | -0.001636 | 0.194054 | -0.365787 | 0.994575 | -0.630394 | -0.193792 | 0. |
| -0.001529 | 0.181325 | -0.392778 | 0.775931 | -0.384216 | -0.181453 | -0.001049 | 0.192692 | -0.378309 | 1.000113 | -0.621577 | -0.192131 | 0. |
| 0.000083 | 0.179577 | -0.393185 | 0.7775 | -0.38378 | -0.182617 | -0.000935 | 0.190871 | -0.390814 | 1.005588 | -0.61235 | -0.192479 | 0. |
| 0.001014 | 0.177436 | -0.393701 | 0.779394 | -0.388592 | -0.179001 | 0.001382 | 0.188187 | -0.393721 | 0.997136 | -0.606508 | -0.187107 | 0. |
| 0.002563 | 0.174867 | -0.394552 | 0.781621 | -0.386194 | -0.176192 | -0.000808 | 0.185258 | -0.405961 | 1.00265 | -0.596868 | -0.189017 | 0. |
| 0.001765 | 0.171848 | -0.395224 | 0.784193 | -0.388593 | -0.170143 | 0.002079 | 0.181673 | -0.413177 | 0.995496 | -0.582548 | -0.181439 | 0. |
| -0.001323 | 0.168333 | -0.395746 | 0.787119 | -0.388442 | -0.168781 | 0.000582 | 0.177829 | -0.425633 | 1.000914 | -0.575007 | -0.180112 | 0. |
| -0.001341 | 0.164262 | -0.397022 | 0.790405 | -0.39086 | -0.164881 | 0.00091 | 0.172951 | -0.428417 | 0.992749 | -0.563854 | -0.171724 | 0. |
| 0.000316 | 0.159588 | -0.39879 | 0.794058 | -0.397756 | -0.161051 | -0.00267 | 0.167829 | -0.440663 | 0.998208 | -0.555869 | -0.167625 | 0. |
| -0.001543 | 0.154256 | -0.400222 | 0.798074 | -0.395529 | -0.153024 | 0.000257 | 0.161822 | -0.44401 | 0.990232 | -0.54485 | -0.160213 | 0. |
| -0.00036 | 0.148185 | -0.402338 | 0.802448 | -0.400816 | -0.148451 | -0.001518 | 0.154843 | -0.446633 | 0.98233 | -0.537843 | -0.154623 | 0. |
| 0.001534 | 0.141297 | -0.404744 | 0.807167 | -0.402752 | -0.14092 | 0.003118 | 0.14732 | -0.450256 | 0.97449 | -0.524203 | -0.148333 | 0. |
| 0.000737 | 0.133504 | -0.406949 | 0.812198 | -0.408045 | -0.134272 | -0.00119 | 0.138574 | -0.452582 | 0.966687 | -0.513346 | -0.136406 | 0. |
| -0.002067 | 0.124697 | -0.409076 | 0.817498 | -0.407281 | -0.122976 | 0.000143 | 0.129027 | -0.455712 | 0.958884 | -0.502999 | -0.131897 | 0. |

**TABLE 8.** *(Continued.)* **Best joint vectors obtained.**

| $q_1$ | $q_2$ | $q_3$ | $q_4$ | $q_5$ | $q_6$ | $q_7$ | $q_8$ | $q_9$ | $q_{10}$ | $q_{11}$ | $q_{12}$ | $OF[m]$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0.003515 | 0.114754 | -0.412342 | 0.822997 | -0.410705 | -0.11568 | 0.000106 | 0.118336 | -0.455021 | 0.950767 | -0.496804 | -0.118105 | 0. |
| -0.002298 | 0.103551 | -0.414384 | 0.828605 | -0.414501 | -0.102173 | -0.001342 | 0.106398 | -0.45775 | 0.942844 | -0.485147 | -0.106362 | 0. |
| -0.003171 | 0.090929 | -0.417024 | 0.834183 | -0.416096 | -0.091353 | 0.00141 | 0.093141 | -0.457251 | 0.934531 | -0.478243 | -0.093701 | 0. |
| -0.000372 | 0.076724 | -0.419886 | 0.839551 | -0.419235 | -0.07422 | -0.001464 | 0.0783 | -0.456082 | 0.925906 | -0.469066 | -0.080203 | 0. |
| -0.000555 | 0.060746 | -0.422276 | 0.844464 | -0.420328 | -0.058839 | 0.001544 | 0.061584 | -0.448095 | 0.902702 | -0.455986 | -0.063586 | 0. |
| 0.000224 | 0.04279 | -0.424341 | 0.848594 | -0.424606 | -0.044351 | 0.000181 | 0.043229 | -0.44628 | 0.892475 | -0.44654 | -0.046011 | 0. |
| -0.001957 | 0.022624 | -0.42572 | 0.851511 | -0.426699 | -0.02146 | 0.000219 | 0.022701 | -0.433181 | 0.866334 | -0.43128 | -0.025515 | 0. |
| -0.000106 | -0. | -0.42632 | 0.85264 | -0.427775 | 0.00011 | 0.001521 | -0. | -0.42632 | 0.852639 | -0.426516 | -0.000477 | 0. |
| 0.002323 | -0.022713 | -0.433113 | 0.866332 | -0.431487 | 0.022907 | 0.001261 | -0.022636 | -0.425727 | 0.85151 | -0.42449 | 0.022196 | 0. |
| -0.002539 | -0.043277 | -0.446343 | 0.892466 | -0.44654 | 0.04199 | -0.0008 | -0.042837 | -0.424327 | 0.848586 | -0.422808 | 0.044017 | 0. |
| -0.000108 | -0.061688 | -0.451355 | 0.902696 | -0.44961 | 0.061987 | 0.00088 | -0.060854 | -0.422165 | 0.844438 | -0.422953 | 0.059245 | 0. |
| 0.000204 | -0.078505 | -0.462952 | 0.925936 | -0.46267 | 0.077908 | 0.000271 | -0.076915 | -0.419722 | 0.839485 | -0.419186 | 0.077251 | 0. |
| 0.000291 | -0.093435 | -0.467283 | 0.934621 | -0.468574 | 0.090278 | 0.002396 | -0.09123 | -0.416812 | 0.834059 | -0.415924 | 0.092405 | 0. |
| -0.001562 | -0.106865 | -0.471674 | 0.943015 | -0.472357 | 0.109245 | 0.001209 | -0.103989 | -0.414074 | 0.828398 | -0.416988 | 0.104585 | 0. |
| -0.003999 | -0.118955 | -0.476117 | 0.951285 | -0.474822 | 0.118421 | -0.001225 | -0.115359 | -0.41148 | 0.82268 | -0.41085 | 0.113838 | 0. |
| 0.000705 | -0.129849 | -0.479686 | 0.959554 | -0.478729 | 0.129927 | -0.002034 | -0.125494 | -0.408772 | 0.817034 | -0.408554 | 0.123336 | 0. |
| 0.000273 | -0.139674 | -0.483913 | 0.967902 | -0.480869 | 0.139209 | 0.000727 | -0.13453 | -0.40568 | 0.811556 | -0.404863 | 0.13458 | 0. |
| 0.000361 | -0.148546 | -0.488137 | 0.976381 | -0.487983 | 0.147778 | 0.000693 | -0.142587 | -0.403055 | 0.806306 | -0.400687 | 0.141708 | 0. |
| -0.000137 | -0.15657 | -0.492531 | 0.98502 | -0.49336 | 0.155652 | -0.000605 | -0.149775 | -0.400753 | 0.801326 | -0.399938 | 0.150548 | 0. |
| -0.000422 | -0.163835 | -0.496989 | 0.993839 | -0.499042 | 0.164175 | 0.000494 | -0.156189 | -0.398245 | 0.796643 | -0.398205 | 0.159028 | 0. |
| -0.002734 | -0.170425 | -0.501882 | 1.002837 | -0.502869 | 0.170741 | 0.000637 | -0.161914 | -0.396029 | 0.792264 | -0.395816 | 0.16294 | 0. |
| 0.000409 | -0.175797 | -0.49963 | 0.999403 | -0.497637 | 0.17855 | 0.002609 | -0.167028 | -0.393663 | 0.788193 | -0.395314 | 0.165985 | 0. |
| -0.000866 | -0.18123 | -0.504589 | 1.008866 | -0.505469 | 0.180855 | 0.002611 | -0.171598 | -0.391767 | 0.784425 | -0.390544 | 0.171151 | 0. |
| 0.000487 | -0.185537 | -0.502895 | 1.005968 | -0.503862 | 0.188088 | -0.002407 | -0.175685 | -0.390895 | 0.78095 | -0.389739 | 0.175247 | 0. |
| -0.002775 | -0.190055 | -0.508444 | 1.015841 | -0.509109 | 0.190418 | -0.0008 | -0.179344 | -0.38902 | 0.777755 | -0.390023 | 0.180881 | 0. |
| -0.000119 | -0.19352 | -0.506733 | 1.01342 | -0.506163 | 0.195565 | -0.00247 | -0.182621 | -0.38786 | 0.774822 | -0.386403 | 0.181387 | 0. |
| -0.002031 | -0.197318 | -0.512215 | 1.023634 | -0.513758 | 0.198138 | -0.000423 | -0.185561 | -0.386144 | 0.772133 | -0.3873 | 0.183339 | 0. |
| -0.000738 | -0.200118 | -0.510959 | 1.021625 | -0.511756 | 0.200413 | 0.001375 | -0.188202 | -0.384578 | 0.769671 | -0.383708 | 0.189646 | 0. |
| 0.000627 | -0.202636 | -0.509768 | 1.019788 | -0.507927 | 0.204215 | -0.000748 | -0.190579 | -0.383851 | 0.767419 | -0.385733 | 0.188962 | 0. |
| 0.000463 | -0.205629 | -0.515126 | 1.03044 | -0.513462 | 0.205235 | 0.002813 | -0.19272 | -0.382139 | 0.765355 | -0.382448 | 0.193748 | 0. |
| -0.000721 | -0.207689 | -0.514605 | 1.028914 | -0.518664 | 0.208127 | -0.001816 | -0.194658 | -0.382084 | 0.763465 | -0.380906 | 0.195805 | 0. |
| -0.002239 | -0.209556 | -0.514221 | 1.027512 | -0.513392 | 0.2122 | -0.00002 | -0.196415 | -0.380868 | 0.761728 | -0.378289 | 0.19479 | 0. |
| 0.000852 | -0.211257 | -0.512934 | 1.026225 | -0.513569 | 0.20864 | -0.002929 | -0.198014 | -0.380641 | 0.760129 | -0.380408 | 0.197358 | 0. |
| 0.001458 | -0.212811 | -0.51221 | 1.025037 | -0.515064 | 0.212697 | 0.001211 | -0.199477 | -0.379088 | 0.758656 | -0.380519 | 0.197623 | 0. |
| 0.002363 | -0.21424 | -0.511465 | 1.023935 | -0.513061 | 0.214561 | -0.000348 | -0.200822 | -0.378711 | 0.757284 | -0.379951 | 0.201421 | 0. |
| 0.002058 | -0.214808 | -0.50482 | 1.010517 | -0.503562 | 0.213665 | 0.000137 | -0.202066 | -0.377976 | 0.756006 | -0.375217 | 0.203128 | 0. |
| 0.001465 | -0.216036 | -0.504458 | 1.009545 | -0.505189 | 0.216184 | 0.000436 | -0.203226 | -0.377314 | 0.754805 | -0.37424 | 0.202082 | 0. |
| 0.002351 | -0.217191 | -0.503807 | 1.008627 | -0.505444 | 0.217704 | 0.001345 | -0.204315 | -0.376561 | 0.753668 | -0.375335 | 0.204445 | 0. |
| -0.002157 | -0.217525 | -0.498062 | 0.995192 | -0.497291 | 0.219997 | -0.000344 | -0.20535 | -0.376361 | 0.752581 | -0.375882 | 0.201912 | 0. |
| -0.001478 | -0.218572 | -0.49749 | 0.994339 | -0.496349 | 0.217861 | 0.000874 | -0.206342 | -0.375587 | 0.751532 | -0.374533 | 0.207291 | 0. |
| 0.001413 | -0.218826 | -0.490088 | 0.980789 | -0.490627 | 0.21903 | -0.001526 | -0.207304 | -0.375568 | 0.750508 | -0.375188 | 0.207335 | 0. |
| -0.000688 | -0.21982 | -0.490127 | 0.979955 | -0.488599 | 0.218828 | -0.002168 | -0.208249 | -0.375195 | 0.749494 | -0.374404 | 0.207659 | 0. |
| 0.002185 | -0.220045 | -0.482641 | 0.966237 | -0.48386 | 0.220463 | -0.001337 | -0.209189 | -0.374519 | 0.748482 | -0.372066 | 0.21056 | 0. |
| 0.002108 | -0.221038 | -0.482231 | 0.965386 | -0.483221 | 0.220433 | 0.00012 | -0.210136 | -0.373703 | 0.747456 | -0.37255 | 0.210548 | 0. |
| 0.000964 | -0.221286 | -0.475514 | 0.951452 | -0.474123 | 0.223836 | 0.003663 | -0.2111 | -0.372433 | 0.746402 | -0.374334 | 0.21034 | 0. |
| 0.000167 | -0.221565 | -0.468611 | 0.937295 | -0.466905 | 0.222578 | -0.000109 | -0.212098 | -0.372677 | 0.745308 | -0.374012 | 0.210759 | 0. |
| -0.000301 | -0.221889 | -0.461513 | 0.922893 | -0.463914 | 0.223803 | -0.000339 | -0.213139 | -0.37215 | 0.744156 | -0.37263 | 0.212979 | 0. |
| -0.002325 | -0.222268 | -0.454626 | 0.908227 | -0.453551 | 0.222117 | 0.00077 | -0.214237 | -0.371305 | 0.742938 | -0.370238 | 0.212324 | 0. |

**TABLE 8.** *(Continued.)* Best joint vectors obtained.

| $q_1$ | $q_2$ | $q_3$ | $q_4$ | $q_5$ | $q_6$ | $q_7$ | $q_8$ | $q_9$ | $q_{10}$ | $q_{11}$ | $q_{12}$ | $OF[m]$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0.000626 | -0.222721 | -0.446495 | 0.893267 | -0.446917 | 0.219893 | -0.001941 | -0.215406 | -0.371227 | 0.741625 | -0.371014 | 0.215199 | 0. |
| 0.001962 | -0.223257 | -0.438556 | 0.87798 | -0.437724 | 0.223165 | 0.0006 | -0.216661 | -0.369974 | 0.740206 | -0.367239 | 0.216336 | 0. |
| -0.00122 | -0.223896 | -0.431439 | 0.862337 | -0.435275 | 0.223957 | -0.001163 | -0.218018 | -0.36958 | 0.738657 | -0.371437 | 0.220032 | 0. |
| 0.000027 | -0.224652 | -0.423141 | 0.846293 | -0.423644 | 0.228076 | -0.000556 | -0.219493 | -0.368602 | 0.736963 | -0.369734 | 0.216953 | 0. |
| 0.002992 | -0.225543 | -0.414236 | 0.82981 | -0.415093 | 0.22535 | -0.000392 | -0.221106 | -0.367631 | 0.735089 | -0.367395 | 0.219752 | 0. |
| 0.001394 | -0.225839 | -0.398432 | 0.797488 | -0.401291 | 0.226349 | -0.002754 | -0.222876 | -0.367114 | 0.733011 | -0.36585 | 0.223422 | 0. |
| 0.001859 | -0.227059 | -0.389404 | 0.779645 | -0.389959 | 0.224968 | -0.000605 | -0.224828 | -0.365479 | 0.730689 | -0.366902 | 0.224253 | 0. |
| 0.002295 | -0.227729 | -0.371883 | 0.744804 | -0.374261 | 0.228062 | 0.000418 | -0.226983 | -0.363954 | 0.728096 | -0.363755 | 0.22716 | 0. |
| -0.000145 | -0. | -0.000275 | 0.00055 | 0.000036 | 0.002796 | 0.00055 | -0. | -0.000531 | 0.001063 | -0.000215 | -0.000895 | 0. |
| -0.002393 | -0. | -0.014817 | 0.029634 | -0.016519 | 0.000579 | 0.000194 | -0. | -0.014859 | 0.029717 | -0.015038 | 0.004613 | 0. |
| 0.00217 | 0. | -0.029499 | 0.058998 | -0.029619 | 0.002126 | 0.000032 | -0. | -0.029497 | 0.058995 | -0.028512 | 0.00002 | 0. |
| -0.002978 | 0. | -0.043943 | 0.087885 | -0.043587 | 0.000598 | -0.000722 | 0. | -0.043944 | 0.087889 | -0.044379 | 0.002671 | 0. |
| -0.002686 | -0. | -0.058181 | 0.116361 | -0.058482 | 0.001769 | 0.001769 | -0. | -0.05818 | 0.116361 | -0.057174 | 0.001802 | 0. |
| -0.000251 | -0.000001 | -0.072207 | 0.144413 | -0.0696 | -0.000295 | -0.00052 | -0. | -0.072206 | 0.144412 | -0.070387 | -0.000698 | 0. |
| 0.000597 | 0. | -0.086017 | 0.172033 | -0.087638 | -0.001279 | 0.001976 | -0. | -0.086017 | 0.172034 | -0.085391 | 0.000466 | 0. |
| 0.000161 | -0. | -0.09962 | 0.199241 | -0.095241 | 0.000337 | 0.002266 | -0. | -0.099621 | 0.199242 | -0.101173 | -0.000754 | 0. |

vector $\theta$, obtained by DE, is shown in Table 8 (Appendix A), where each column represents a joint trajectory considering a sample time of 20 ms. A graphical representation of these trajectories is presented in Figs. 9 and 10, for right and left legs, respectively. Both figures have two horizontal axis, the first one representing the 350 points of the discretized trajectory, from the rows of Table 8 and the second one corresponding to the elapsed time of the dynamic walking, which in this case is set as 7 s. All trajectories are within the joint limits of the robot, ensuring that they can be safely implemented in a real application, even if the position of the initial configuration is defined near to a singularity.

A validation of the resulting joint trajectories obtained by DE, was developed with a dynamic simulation of the biped robot walking, by means of a ROS (Robotic Operating System) based platform that simulates the physical environment in Gazebo [33]. This validation process is depicted in Fig. 13, and was implemented with the best IKP solutions generated by the proposed optimization approach. The joint references are controlled with a simple ROS-based PID control strategy for each joint, which receives a joint position as input and calculates the required torque to develop the dynamic walking, as depicted in the snapshots of Fig. 14.

The robot walking was successfully performed even without including a feedback control for correcting the torso tilt, because the CoM and foot physically consistent trajectories were calculated to develop a stable walking. This is a good indicator of the quality of the IKP solution obtained with the proposed approach. The proposed solution algorithm generated the $n$ joint trajectories, considering the complex dynamic behavior of the biped robot and automatically evading singularity configurations for each desired pose in each end effector. Furthermore, the trajectories were obtained without requiring a normalization of the objective function nor the generation of random $\omega_i$ values to avoid bias in the search for the solution, by considering the orientation errors in each end effector as equality constraints.

It is evident the effectiveness of DE to solve the IKP. However, even due the main purpose of this work is not directly related to the analysis of diverse metaheuristics, the Harmony Search (HS) algorithm was implemented [34], considering the parameters presented in Table 7, and using the previously defined stop criteria. Since HS is inspired by the musical composition process, the algorithm requires two parameters to iterative adjust the tone slightly: pitch adjustment $r_{pa}$ and band width $bw$. The ratio acceptance parameter $r_{accept}$ is used to control the generation of new harmonies based on the previous harmonies or on a randomly manner. Typically, $bw = (U_i - L_i)/1000$, where $U_i$ and $L_i$ are the upper and lower limits respectively, for each design variable. The 12 joint trajectories obtained are presented in Figs. 11 and 12.

However, if it is necessary to compare the performance of different heuristic algorithms, it is possible to perform an inferential statistical test such as the well-known Friedman test, as suggested in [35], [36], and [37].

## V. FINAL DISCUSSION

In this work, a new approach based on constrained numerical optimization was presented to solve the IKP of legged robots in dynamic walking. It applies a heuristic algorithm in conjunction with a constraint handler to obtain the joint trajectories for generating the walking. The approach only uses the DKM of the robot, so it does not require the closed equations that describe the IKM. For this reason, it can be applied to legged robots with different kinematic topologies.

As case study, it was applied to the solution of the IKP of one of the most complex type of legged robots, a biped robot with 12 degrees of freedom, considering its movement in the planes sagital, frontal and lateral. It obtained not only the IK of a series of discrete points in the workspace, but also along the trajectories considering the dynamic and kinematic constraints of the robot (physically consistent trajectories), for the torso and feet, automatically avoiding singularity configurations on position. Furthermore, the optimization model is a generalization of the IKP solution for legged robots with any number of end effectors. The joint trajectories can be directly implemented in both the simulated and in the real robot, since the optimization model considers their joint ranges and finds smooth trajectories in terms of position. However, it is important to note that in real robots it is necessary to correctly set the sampling time, considering the velocity and effort (force or torque) limits of the actuator joints to avoid damage. As future work, full body trajectory planning will be considered of humanoid-type legged robots for grasping and manipulation tasks, with obstacle avoidance in the workspace. Additionally, it is considered to reduce the search time for the solutions, to implement the approach in real-time applications.

At the moment, this approach does not include a direct mechanism to deal with multiple solutions or multi-modal capabilities. This aspect is addressed by the Heuristic Algorithm, since during the search process it will tend to randomly follow only one of the multiple feasible solutions, always considering all of the restrictions to solve the IKP established in the optimization problem.

## APPENDIX A

The table in this appendix contain the best joint vectors obtained with the proposed approach and their corresponding object function, for each of the trajectories considered in the case study.

## REFERENCES

[1] B. Siciliano, O. Khatib, and T. Kröger, *Springer Handbook of Robotics*. Berlin, Germany: Springer, 2008, doi: 10.1007/978-3-540-30301-5.

[2] J. J. Craig, *Robótica*. Mexico City, México: Pearson, 2006.

[3] L. Yiyang, J. Xi, B. Hongfei, W. Zhining, and S. Liangliang, "A general robot inverse kinematics solution method based on improved PSO algorithm," *IEEE Access*, vol. 9, pp. 32341–32350, 2021, doi: 10.1109/ACCESS.2021.3059714.

[4] J. Abdor-Sierra, E. Merchán-Cruz, F. Sánchez-Garfias, R. Rodríguez-Cañizo, E. Portilla-Flores, and V. Vázquez-Castillo, "Particle swarm optimization for inverse kinematics solution and trajectory planning of 7-DOF and 8-DOF robot manipulators based on unit quaternion representation," *J. Appl. Eng. Sci.*, vol. 19, no. 3, pp. 592–599, 2021, doi: 10.5937/jaes0-30557.

[5] D. Wu, G. Hou, W. Qiu, and B. Xie, "T-IK: An efficient multi-objective evolutionary algorithm for analytical inverse kinematics of redundant manipulator," *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 8474–8481, Oct. 2021, doi: 10.1109/LRA.2021.3108550.

[6] L. C. Antonio-Gopar, C. Lopez-Franco, N. Arana-Daniel, E. Gonzalez-Vallejo, and A. Y. Alanis, "Inverse kinematics for a manipulator robot based on differential evolution algorithm," in *Proc. IEEE Latin Amer. Conf. Comput. Intell. (LA-CCI)*, Guadalajara, México, Nov. 2018, pp. 1–5, doi: 10.1109/LA-CCI.2018.8625233.

[7] K. Wichapong, S. Bureerat, and N. Pholdee, "Solving inverse kinematics of robot manipulators by means of meta-heuristic optimisation," *IOP Conf. Ser., Mater. Sci. Eng.*, vol. 370, May 2018, Art. no. 012056, doi: 10.1088/1757-899X/370/1/012056.

[8] C. A. Pena, M. A. Guzman, and P. F. Cardenas, "Inverse kinematics of a 6 DOF industrial robot manipulator based on bio-inspired multi-objective optimization techniques," in *Proc. IEEE Colombian Conf. Robot. Autom. (CCRA)*, Bogotá, Colombia, Sep. 2016, pp. 1–6, doi: 10.1109/CCRA.2016.7811428.

[9] M. Toz, "Chaos-based vortex search algorithm for solving inverse kinematics problem of serial robot manipulators with offset wrist," *Appl. Soft Comput.*, vol. 89, Apr. 2020, Art. no. 106074, doi: 10.1016/j.asoc.2020.106074.

[10] S. Dereli, "A new modified grey wolf optimization algorithm proposal for a fundamental engineering problem in robotics," *Neural Comput. Appl.*, vol. 33, no. 21, pp. 14119–14131, Nov. 2021, doi: 10.1007/s00521-021-06050-2.

[11] S. Dereli and R. Köker, "Simulation based calculation of the inverse kinematics solution of 7-DOF robot manipulator using artificial bee colony algorithm," *Social Netw. Appl. Sci.*, vol. 2, no. 1, pp. 1–11, Jan. 2020, doi: 10.1007/s42452-019-1791-7.

[12] S. S. M. Sidik, N. E. Zamri, M. S. M. Kasihmuddin, H. A. Wahab, Y. Guo, and M. A. Mansor, "Non-systematic weighted satisfiability in discrete Hopfield neural network using binary artificial bee colony optimization," *Mathematics*, vol. 10, no. 7, p. 1129, Apr. 2022, doi: 10.3390/math10071129.

[13] N. E. Zamri, S. A. Azhar, S. S. M. Sidik, M. A. Mansor, M. S. M. Kasihmuddin, S. P. A. Pakruddin, N. A. Pauzi, and S. N. M. Nawi, "Multi-discrete genetic algorithm in Hopfield neural network with weighted random $k$ satisfiability," *Neural Comput. Appl.*, vol. 34, no. 21, pp. 19283–19311, Nov. 2022, doi: 10.1007/s00521-022-07541-6.

[14] S. Starke, N. Hendrich, S. Magg, and J. Zhang, "An efficient hybridization of genetic algorithms and particle swarm optimization for inverse kinematics," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Qingdao, China, Dec. 2016, pp. 1782–1789, doi: 10.1109/ROBIO.2016.7866587.

[15] S. Starke, N. Hendrich, and J. Zhang, "Memetic evolution for generic full-body inverse kinematics in robotics and animation," *IEEE Trans. Evol. Comput.*, vol. 23, no. 3, pp. 406–420, Jun. 2019, doi: 10.1109/TEVC.2018.2867601.

[16] P. Ruppel, N. Hendrich, S. Starke, and J. Zhang, "Cost functions to specify full-body motion and multi-goal manipulation tasks," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Brisbane, QLD, Australia, May 2018, pp. 3152–3159, doi: 10.1109/ICRA.2018.8460799.

[17] M. Toz and S. Kucuk, "Dexterous workspace optimization of an asymmetric six-degree of freedom Stewart–Gough platform type manipulator," *Robot. Auto. Syst.*, vol. 61, no. 12, pp. 1516–1528, Dec. 2013, doi: 10.1016/j.robot.2013.07.004.

[18] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa, "The 3D linear inverted pendulum mode: A simple modeling for a biped walking pattern generation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. Expanding Societal Role Robot. the Next Millennium*, Maui, HI, USA, Oct. 2001, pp. 239–246, doi: 10.1109/IROS.2001.973365.

[19] S. Kajita, H. Hirukawa, K. Harada, and K. Yokoi, *Introduction to Humanoid Robotics*. Heidelberg, Germany: Springer, 2014, doi: 10.1007/978-3-642-54536-8.

[20] S. Starke, N. Hendrich, D. Krupke, and J. Zhang, "Evolutionary multi-objective inverse kinematics on highly articulated and humanoid robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 6959–6966, doi: 10.1109/IROS.2017.8206620.

[21] D. Simon, *Evolutionary Optimization Algorithms*. Hoboken, NJ, USA: Wiley, 2013.

[22] R. Storn and K. Price, "Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optim.*, vol. 11, no. 4, pp. 341–359, Dec. 1997, doi: 10.1023/A:1008202821328.

[23] M. B. Calva-Yáñez, P. A. Niño-Suarez, E. A. Portilla-Flores, J. A. Aponte-Rodríguez, and E. Santiago-Valentín, "Reconfigurable mechanical system design for tracking an ankle trajectory using an evolutionary optimization algorithm," *IEEE Access*, vol. 5, pp. 5480–5493, 2017, doi: 10.1109/ACCESS.2017.2692681.
[24] E. Santiago-Valenten, E. A. Portilla-Flores, E. Mezura-Montes, E. Vega-Alvarado, M. B. Calva-Yanez, and M. Pedroza-Villalba, "A graph-theory-based method for topological and dimensional representation of planar mechanisms as a computational tool for engineering design," *IEEE Access*, vol. 7, pp. 587–596, 2019, doi: 10.1109/ACCESS.2018.2885563.
[25] M. A. Funes, E. A. Portilla, R. Rivera, E. A. Merchan, and M. F. Carbajal, "Metaheuristic techniques comparison to optimize robotic end-effector behavior and its workspace," *Int. J. Adv. Robot. Syst.*, vol. 15, no. 5, 2018, Art. no. 1729881418801132, doi: 10.1177/1729881418801132.
[26] S. Gao, K. Wang, S. Tao, T. Jin, H. Dai, and J. Cheng, "A state-of-the-art differential evolution algorithm for parameter estimation of solar photovoltaic models," *Energy Convers. Manage.*, vol. 230, Feb. 2021, Art. no. 113784, doi: 10.1016/j.enconman.2020.113784.
[27] K. Price, R. M. Storn, and J. A. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*. Berlin, Germany: Springer, 2005, doi: 10.1007/3-540-31306-0.
[28] R. Datta and K. Deb, *Evolutionary Constrained Optimization*. New Delhi, India: Springer, 2014, doi: 10.1007/978-81-322-2184-5.
[29] D. N. Nenchev, A. Konno, and T. Tsujita, *Humanoid Robots: Modeling and Control*. Cambridge, MA, USA: Elsevier, 2019, doi: 10.1016/C2015-0-01877-9.
[30] J. Vazquez-Santacruz, J. Torres-Figueroa, and R. D. J. Portillo-Velez, "Design of a human-like biped locomotion system based on a novel mechatronic methodology," *Concurrent Eng.*, vol. 27, no. 3, pp. 249–267, Sep. 2019, doi: 10.1177/1063293X19857784.
[31] K. Deb, "An efficient constraint handling method for genetic algorithms," *Comput. Methods Appl. Mech. Eng.*, vol. 186, nos. 2–4, pp. 311–338, Jun. 2000, doi: 10.1016/S0045-7825(99)00389-8.
[32] J. J. Liang, T. P. Runarsson, E. Mezura, M. Clerc, P. N. Suganthan, C. A. Coello, and K. Deb, "Problem definitions and evaluation criteria for the CEC 2006 special session on constrained real-parameter optimization," *J. Appl. Mech.*, vol. 41, no. 8, pp. 8–31, 2006.
[33] N. Koenig and A. Howard, "Design and use paradigms for Gazebo, an open-source multi-robot simulator," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sendai, Japan, Sep./Oct. 2004, pp. 2149–2154, doi: 10.1109/IROS.2004.1389727.
[34] Z. W. Geem, J. H. Kim, and G. V. Loganathan, "A new heuristic optimization algorithm: Harmony search," *J. Simul.*, vol. 76, no. 2, pp. 60–68, Feb. 2001, doi: 10.1177/003754970107600201.
[35] M. S. M. Kasihmuddin, S. Z. M. Jamaludin, M. A. Mansor, H. A. Wahab, and S. M. S. Ghadzi, "Supervised learning perspective in logic mining," *Mathematics*, vol. 10, no. 6, p. 915, Mar. 2022, doi: 10.3390/math10060915.
[36] S. Z. M. Jamaludin, N. A. Romli, M. S. M. Kasihmuddin, A. Baharum, M. A. Mansor, and M. F. Marsani, "Novel logic mining incorporating log linear approach," *J. King Saud Univ., Comput. Inf. Sci.*, vol. 34, no. 10, pp. 9011–9027, Nov. 2022, doi: 10.1016/j.jksuci.2022.08.026.
[37] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 3–18, Mar. 2011, doi: 10.1016/j.swevo.2011.02.002.

**JACOBO TORRES-FIGUEROA** received the B.Sc. degree in mechatronics engineering and the master's degree in applied engineering for mechatronics systems and intelligent structures from Universidad Veracruzana, Mexico, in 2015 and 2018, respectively. He is currently pursuing the Ph.D. degree in robotics and mechatronics systems engineering with the Centro de Innovación y Desarrollo Tecnológico en Cómputo (CIDETEC), Instituto Politécnico Nacional, Mexico City. His research interests include deep reinforcement learning, humanoid and legged robots, optimization, heuristic algorithms, robotic operating systems, SysML, and model-based systems engineering (MBSE).

**EDGAR A. PORTILLA-FLORES** was born in Tlaxcala, Mexico, in 1968. He received the B.Sc. degree in electronics engineering from Universidad Autónoma Metropolitana, Mexico, in 1992, the M.Sc. degree in mechanical engineering from the Instituto Tecnológico de Puebla, Mexico, in 2002, and the Ph.D. degree in electrical engineering from the Centro de Investigación y Estudios Avanzados, Mexico, in 2006. He completed the Postdoctoral Residency at the Universidad de Estadual de Campinas, Brazil, in 2012. He is currently a full-time Research Professor with the Unidad Profesional Interdisciplinaria de Ingeniería Campus Tlaxcala, Instituto Politécnico Nacional, Mexico. His research interests include optimum design of mechatronic systems and the application of bio-inspired algorithms to the solution of engineering problems. He is a member of the National System of Researchers of Mexico.

**JOSÉ A. VÁSQUEZ-SANTACRUZ** received the Ph.D. degree from the CINVESTAV, Mexico City. Currently, he is a full-time Professor in mechatronics engineering with Universidad Veracruzana. His research interests include robotics and mechatronics design.

**EDUARDO VEGA-ALVARADO** was born in Mexico City, Mexico, in 1965. He received the B.Sc. degree in communications and electronics engineering from the Escuela Superior de Ingeniería Mecánica y Eléctrica, Mexico, in 1992, the M.Sc. degree in digital systems from the Centro de Investigación Tecnológica en Computación, Mexico, in 1996, and the Ph.D. degree in computational and electronic systems from the Universidad Autónoma de Tlaxcala, Mexico, in 2017. He is currently a full-time Researcher and a Professor with the Centro de Innovación y Desarrollo Tecnológico en Cómputo, Instituto Politécnico Nacional, Mexico City. His research interests include programming, optimization with metaheuristics, the development of hybrid algorithms, and its application on engineering problems.

**LUIS F. MARÍN-URÍAS** received the Graduate degree from the Faculty of Informática, Universidad Veracruzana, the master's degree in artificial intelligence from the Universidad Veracruzana, the Master 2 Recherche degree in critical informatics systems from the University of Toulouse III—Paul Sabatier, and the Doctorate degree (Agreement) in robotics from the LAAS-CNRS, University of Toulouse III—Paul Sabatier. He has worked in the industry in the automation of information processes. He is currently a full-time Professor at Universidad Veracruzana, where he carries out research, technology development, and teaching activities. He returned to do a postdoctoral stay at the Artificial Intelligence Research Center, Universidad Veracruzana. He has participated in European and international research projects, such as COGNIRON, DEXMART, and CHRIS, where artificial intelligence applied to robotics was addressed.