International Journal of Industrial Engineering & Production Research September 2023 Vol. 34, No. 3: 1-21
DOI: 10.22068/ijiepr.34.3.2

RESEARCH PAPER

Trajectory optimization of Biped Robots with Maximum Load Carrying Capacities: Iterative Linear Programming Approach

Ammar Fadhil Hussein AL-Maliki[1] & Moharam Habibnejad Korayem[*2]

Received 15 May 2023; Revised 22 July 2023; Accepted 26 July 2023;

ABSTRACT
A computational approach is presented to obtain the optimal path of the end-effector for the 10 DOF bipedal robot to increase its load carrying capacity for a given task from point to point. The synthesizing optimal trajectories problem of a robot is formulated as a problem of trajectory optimization. An Iterative Linear Programming method (ILP) is developed for finding a numerical solution for this nonlinear trajectory. This method is used for determining the maximum dynamic load carrying capacity of bipedal robot walking subjected to torque actuators, stability and jerk limits constraints. First, the Lagrangian dynamic equation should be written to be suitable for the load dynamics which together with kinematic equations are substantial for determining the optimal trajectory. After that, a representation of the state space of the dynamic equations is introduced also the linearized dynamic equations are needed to obtain the numerical solution of the trajectory optimization followed by formulation for the optimal trajectory problem with a maximum load. Finally, the method of ILP and the computational aspect is applied to solve the problem of trajectory synthesis and determine the dynamic load carrying capacity (DLCC) to the bipedal robot for each of the linear and circular path. By implementing on an experimental biped robot, the simulation results were validated.

KEYWORDS: *Dynamic load; Biped robots; Optimal trajectory; Actuator constraint.*

1. Introduction

In this work, problem of calculating the maximum load carrying capacity in optimal trajectory is explained. Specific two end-position for the end effector, the matter is to synthesize the dynamic path of the end effector, thus allowing carrying the maximum load between these two end-position. By using a state representation for dynamic equations, at first, the formulation of state space related to biped robot dynamic equations is presented, followed by the thorough problem formulation of the track optimization problem. In other words, the matter of dynamic trajectories synthesizing of robot when carrying the maximum load can be formulated as a problem of trajectory optimization by the state representation of the equations of motion. The method of Iterative Linear Programming (ILP) is developed to find solutions for the problem of nonlinear trajectory optimization. As a general concept, this method coordination point to point robot movements, which are load optimal with cycle time constraint. The formulation takes into account both joint variable and actuator constraints. ILP technique depends on the “Method of Approximate Programming (MAP)” which is essentially the revision of the class of general cutting plane methods. The basic concept behind the method of cutting plane is straightforward. Reality the problem of nonlinear programming (NLP), the algorithms include repeated updating of the solution of the linear programming (LP) problem which is found by the linearization of the initial NLP problem about the prior solution of the LP. If the solution converges to a predefined little toleration, the iterations will be stopped. The method of ILP offered in general, is suitable to employ and can handle a set of constraints. The

[*] **Corresponding author:** ***Moharam Habibnejad Korayem***
hkorayem@iust.ac.ir

1. *Robotics Research Laboratory, Center of Excellence in Experimental Solid Mechanics and Dynamics, School of Mechanical Engineering, Iran University of Science and Technology, Tehran, Iran.*
2. *Robotics Research Laboratory, Center of Excellence in Experimental Solid Mechanics and Dynamics, School of Mechanical Engineering, Iran University of Science and Technology, Tehran, Iran.*

process of linearization and its convergence with the optimal path are complicated tasks when the systems have a large degree of freedom. That means even when the method converges with the solution, cannot warranty that it is the right optimal solution to the problem. In reality, it may not even be an optimal solution. Often procedure to solve the difficulty is to work the program for a set of different elementary guesses. When the numerical approach converges with the same trajectory and control for a variety of elementary guesses, this reinforces that there is some emphasis, that a truly optimal solution to the main problem has been found [1, 2].

[3] presents a formulation and numerical solution to the problem of determining a point-to-point path for flexible manipulators with a maximum load. Two constraints were imposed actuator torque and the deformation on the end effector to find the maximum allowable load. For calculating the maximum load of the elastic robot in the optimal path subject to the constraints above, an ILP method is used, whilst a computational approach for the case of multiple-link with arbitrary paths is introduced in detail. A computational procedure for finding optimal path of flexible joints and links mobile manipulators to rise their ability to carry the load for a given point-to-point task were developed by [4]. In this study, similar to classical rigid manipulator equations, the dynamic equations are organized in a closed form. After that, the problem of DLCC on flexible mobile manipulators is converted into the problem of a trajectory optimization. Formulation of dynamic equations at first in the state space form then linearized. After that, the ILP method is developed to calculate the DLCC of these types of manipulators. Also, the study included a simulation using two numerical examples with a discussed the results. The main purpose of the study carried out by [5]. is to find the DLCC for the flexible link manipulators in point to-point motion and was formulated as an optimal control problem. The finite element method was utilized for modeling the dynamic equations of motion and deriving these equations. The work appointed an indirect solution for optimum control of the planning of system motion. The implementation of Pontryagin's minimum principle to problem resulted in the standard two-point boundary value problem (TPBVP), found in a numerical way. Then the formulation was developed to obtain the maximum load and optimal trajectory. The proposed method has the major feature that several optimal paths can be found beside different maximum payloads and many characteristics. Thus, the specialist can choose the proper trajectory between the many optimal trajectories. To investigate the effectiveness of the method, a simulation on a two-link flexible manipulator was introduced.

[6] studied a general formulation to determine the DLCC of a mechanical manipulator has elastic links. Optimal control method was applied to form the Hamiltonian function, through selecting suitable objective function, and pick the needful conditions from the Pontryagin minimum principle PMP. This study aims to find the maximum load that a manipulator can carry with optimal trajectory, the simulation and experimental data confirm the trustiness of the technique in calculating the DLCC for two links flexible manipulator. The objective of the work carried out by [7] was to calculate the dynamic load carrying capacity of a manipulator with a specific trajectory. The method of a closed-loop optimal control was used. The solution methods depend on indirect methods for designing optimal controllers in a closed-loop form commonly, while the presented method is a mixing of direct as well as indirect methods. The theorem of the optimum control is offered from solve the Hamilton Jacobi Bellman (HJB) partial differential equation. Also, the equation is difficult to solve properly for complex dynamics, thus can be solved numerically by the Galerkin technique together with the algorithm of nonlinear optimization. By using a fixed manipulator, the simulation was carried out. The results illustrated the effectiveness of the method for tracking the predetermined trajectory and calculating DLCC. By using a two-link manipulator an experimental test has been achieved. The major innovation of the study by [8]. is to calculate the DLCC of a flexible joint manipulator using the approach of nonlinear optimal control. The presented approach was compared with closed-loop nonlinear methods. The method is compared with robust sliding mode control (SMC) and feedback linearization (FL) methods to introduce a better behavior of the approach of the nonlinear optimal control. From the Hamilton Jacobi Bellman (HJB) equation an optimal controller was designed. By using the FL method, angular position, velocity, acceleration and jerk of links are new states to linearize the dynamic equations. In the situation of SMC, the equations of motion for the manipulator are replaced with the standard form. Finally, the Slotine method can be used to design the sliding mode controller. In this study, both simulation and experimental tests were performed to validate the offered method. [9] used the method of optimal

sliding mode control which is extracted from the state-dependent Riccati equation (SDRE) and sliding mode control (SMC) technique. The proposed technique is utilized in a type of nonlinear closed-loop system. The percentage of load distribution between every manipulator can be derived depending on the cost function, to raise the value DLCC, and by comparison for SDRE way, DLCC increased about 15% when applying the method of optimum sliding mode control. The problem in this method is the retard, for real-time performance, so power series approximation PSA is utilized for achievement. A new method is presented by [10]. to solve the problem of optimum control and also to find the DLCC for mobile and fixed manipulators in point-to-point movement. The new method used includes a direct and indirect method. The law of optimum control with the form of a state feedback, the systems of nonlinear dynamics are specified by the solution to the equation of the nonlinear Hamilton Jacobi Bellman (HJB). The Galerkin way and nonlinear optimization algorithm are utilized to solve the equation in a numerical way. The second innovation is the planning of the optimal path, which is carried out in parallel with the method of controller design. A new algorithm is presented to determine the DLCC for manipulators and the linked optimal path using the presented technique. Simulation and experimental tests were performed to validate these proposed methods. This study aims to calculate DLCC for a biped robot for optimal trajectory within torque of DC motor, stability and jerk limits constraints. After the DLCC for a specific path was obtained according to the above constraints in different ways as used in works [11, 12], then can be introduced the state space representation of dynamic equations. And the nonlinear state space dynamic equations for biped robot are linearized. Then the synthesis problem of dynamic trajectories of robot with maximum load carrying capacities can be formulated as a problem of path optimization by the state space formulation of the dynamic equations. Finally, the ILP method and the computational technique to compute the optimum path are developed and validated with experimental results. It should be noted that we have chosen to publish this study in the IJIEPR due to presenting new ideas and results of modern research, where this study provides a computational approach to find the optimal path for the movement of the end-effector for the 10 DOF bipedal robot, as this type of robots has a wide scope of applications and uses in various fields of industry and modern production systems, which this journal is concerned with providing new research that have a close relationship in these vital areas.

2. Modeling

2.1. State space representation for dynamic equation of a biped robot

When the arm of the biped robot carries the load, thus the load can be supposed as available. This load modeled as a point mass (m_l), thus the general Lagrangian dynamic equation for n degree of freedom bipedal robot:

$$\{[M\ (q, m_l)][\ \ddot{q}\ ]+[\ C(q, \dot{q}, m_l)][\ \dot{q}]+[\ G(q,\ m_l)]\}=[\ D(q,\ m_l)][\tau] \tag{1}$$

$$\tau = [\ D(q, m_l)]^{-1}\ \{[M\ (q, m_l)]\ \ddot{q} + [C\ (q, \dot{q},\ m_l)]\ \dot{q}\ + [G\ (q, m_l)]\} \tag{2}$$

$$\tau = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \vdots \\ \tau_m \end{bmatrix} \tag{3}$$

Where τ is an (n × 1) the actuator torques vector. M (q,m_l) is an (n×n) matrix of mass, the matrix is a function of the joint position (q), the mass and inertia of links and the mass of the load (m_l). In general, the form can be defined as:

$$M\ (q, m_l) = \begin{bmatrix} M_1(q_1, m_l) & 0_{n1\times n2} & \cdots & 0_{n1\times nm} \\ 0_{n2\times n1} & M_2(q_2, m_l) & \cdots & 0_{n2\times nm} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{nm\times n1} & 0_{nm\times n2} & \cdots & M_m(q_m, m_l) \end{bmatrix} \tag{4}$$

C (q,q̇, m_l) is an (n × 1) vector, including the centripetal as well as Coriolis terms and the mass of load (m_l). In the general form can be defined as:

$$C\,(q,\dot{q},m_l) = \begin{bmatrix} C_1(q_1,\dot{q}_1,m_l) & 0_{n1\times n2} & \cdots & 0_{n1\times nm} \\ 0_{n2\times n1} & C_2(q_2,\dot{q}_2,m_l) & \dots & 0_{n2\times nm} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{nm\times n1} & 0_{nm\times n2} & \cdots & C_m(q_m,\dot{q}_m,m_l) \end{bmatrix} \tag{5}$$

G (q, m_l): is the gravity influence vector with mass of the load (m_l). Where:

$$G\,(q,m_l) = \begin{bmatrix} G_1(q_1,m_l) \\ G_2(q_2,m_l) \\ \vdots \\ G_m(q_m,m_l) \end{bmatrix} \tag{6}$$

D (q, m_l): represents the matrix of torque distribution containing mass of the load(m_l), in the general form we can get:

$$D\,(q,m_l) = \begin{bmatrix} D_1(q_1,m_l) & 0_{n1\times n2} & \cdots & 0_{n1\times nm} \\ 0_{n2\times n1} & D_2(q_2,m_l) & \dots & 0_{n2\times nm} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{nm\times n1} & 0_{nm\times n2} & \cdots & D_m(q_m,m_l) \end{bmatrix} \tag{7}$$

q, $\dot{q}$, $\ddot{q}$ are (n×1) vectors of position, velocity, and acceleration for the joints. So we can express as follows:
$\ddot{q}$= [M (q,m_l)]$^{-1}$ [(D (q, m_l))τ - C (q,$\dot{q}$,m_l) [$\dot{q}$] - G (q, m_l)]

= f (q, $\dot{q}$,τ, m_l) (8)

The state vectors is :

$$X = [x_1, x_2]^T \tag{9}$$

Where:

$$x_1=[q_1, q_2,\dots q_n]^T \tag{10}$$

$$x_2=[\dot{q}_1, \dot{q}_2,\dots\dot{q}_n]^T \tag{11}$$

The state space representation of the dynamic Eq. (1) is represented as:

$$\dot{X} = \left|\frac{\dot{x}_1}{\dot{x}_2}\right| = \left|\frac{x_2}{f(X,\tau,m_l)}\right| \tag{12}$$

Where: X is a (2n×1) vector and$(X,\tau,\ m_l\ )$: (n) non-linear functions.

2.2. Linearized of the state space representation for dynamic equation of a biped robot

In this step, the form of the state space dynamic equation above, can be rewriting so to find the solution for the problem of nonlinear constrained trajectory optimization. This procedure is one of the requirements for describing the dynamic motion equation in accordance with the parameters of the iterative method to obtain the optimal path. In the other word, state space dynamic equations are linearized:

$$\frac{[X(j+1)-X(j)]}{p} = f\,(X\,(j), \tau(j),\ m_l) \tag{13}$$

Where p = $\frac{(t_{final}-t_{initial})}{m}$, (m) represents the overall number of set points utilized to discretize the path of the end- effector. When Equation (13) substituting into Equation (12), this give:

$$F\,(X\,(j), \tau(j), m_l) = \left[\frac{(x_2)}{(\,f\,(X\,(j),\tau\,(j), m_l)\,)}\right] \tag{14}$$

The nonlinear function above in Equation (14) should be expanded in the (Taylor series form) about the trajectory of(k^{th}). With the need to neglect the terms of higher order, can get:

$$f^{k+1}(\mathrm{j}) = f^{k}(\mathrm{j}) + \frac{\partial f}{\partial x1}(\mathrm{j})\,(x_1^{k+1}(\mathrm{j}) - x_1^{k}(\mathrm{j})) + \frac{\partial f}{\partial x2}(\mathrm{j})\,(x_2^{k+1}(\mathrm{j}) - x_2^{k}(\mathrm{j})) + \frac{\partial f}{\partial \tau}(\mathrm{j})\,(\tau^{k+1}(\mathrm{j}) - \tau^{k}(\mathrm{j})) + \frac{\partial f}{\partial\, m_l}(\mathrm{j})\,(m_l^{k+1} - m_l^{k}) \quad (15)$$

Now, with more compactly, Equation (15) can be rewritten as:

$$f^{k+1}(\mathrm{j}) = \frac{\partial f}{\partial x1}(\mathrm{j})\, x_1^{k+1}(\mathrm{j}) + \frac{\partial f}{\partial x2}(\mathrm{j})\, x_2^{k+1}(\mathrm{j}) + \frac{\partial f}{\partial \tau}(\mathrm{j})\, \tau^{k+1}(\mathrm{j}) + \frac{\partial f}{\partial\, m_l}(\mathrm{j})\, m_l^{k+1} + X^K(j) \quad (16)$$

And:

$$X^K(j) = f^{k}(\mathrm{j}) - \frac{\partial f}{\partial x1}(\mathrm{j})\, x_1^{k}(\mathrm{j}) - \frac{\partial f}{\partial x2}(\mathrm{j})\, x_2^{k}(\mathrm{j}) - \frac{\partial f}{\partial \tau}(\mathrm{j})\, \tau^{k}(\mathrm{j}) - \frac{\partial f}{\partial\, m_l}(\mathrm{j})\, m_l^{k} \quad (17)$$

The superscript (k + 1) can be neglected for the sake of simplicity, substituting Eq. (16) into Eq. (13):

$$x_2(\mathrm{j+1}) = [\frac{\partial f}{\partial x1}(\mathrm{j})]\, x_1(\mathrm{j}) + [\, p\,\frac{\partial f}{\partial x2}(\mathrm{j}) + \mathrm{I}]\, x_2(\mathrm{j}) + p\,[\frac{\partial f}{\partial \tau}(\mathrm{j})]\, \tau(\mathrm{j}) + p\,[\frac{\partial f}{\partial\, m_l}(\mathrm{j})\,]m_l + pX^K(j) \quad (18)$$

And

$$x_1(\mathrm{j+1}) = x_1(\mathrm{j}) + p\, x_2(\mathrm{j}) \quad (19)$$

Equations (17) and (18) will be rewritten in the form of a matrix:

$$\mathrm{X}(\mathrm{j+1}) = [E_x]_\mathrm{j}\, \mathrm{X}(\mathrm{j}) + [E_\tau]_\mathrm{j}\, \tau(\mathrm{j}) + [E_m]_\mathrm{j}\, m_l + E_\mathrm{j} \quad (20)$$

All the matrices $[E_x]_\mathrm{j}$, $[E_\tau]_\mathrm{j}$, $[E_m]_\mathrm{j}$ and E_j as below:

$$[\mathrm{E_x}]_\mathrm{j} = \left[\begin{array}{c|c} [I] & p[I] \\ \hline \frac{\partial f}{\partial x1}(j)p] & \frac{\partial f}{\partial x2}(j)p + I] \end{array}\right] \quad (2\mathrm{n}\times 2\mathrm{n}\ \text{matrix}) \quad (21)$$

$$[\mathrm{E_\tau}]_\mathrm{j} = \left[\begin{array}{c} [0] \\ \hline \frac{\partial f}{\partial \tau}(j)p] \end{array}\right] \quad (2\mathrm{n}\times \mathrm{n}\ \text{matrix}) \quad (22)$$

$$[\mathrm{E}_m]_\mathrm{j} = \left[\begin{array}{c} 0 \\ \hline \frac{\partial f}{\partial\, m_l}(j)p \end{array}\right], \quad \mathrm{E_j} = \left[\begin{array}{c} 0 \\ \hline X^K(j)\, p \end{array}\right] \quad (2\mathrm{n}\times 1\text{vector}) \quad (23)$$

[I]: (n×n) identity matrix, [0] : (n×n) null matrix, and 0 : (n×1) null vector. The procedures to derive the matrices $[E_x]_\mathrm{j}$, $[E_\tau]_\mathrm{j}$, $[E_m]_\mathrm{j}$ and E_j will be explained in the Appendix (A). X (j+1) introduced as a linear combination of (m_l) and τ (j). Equation (20) will be:

$$\mathrm{X}(\mathrm{j+1}) = X_p(\mathrm{j}+1) + \lambda_\mathrm{j}\, m_l + \sum_{i=1}^{j}[\mu_\mathrm{ji}]\, \tau(\mathrm{j}) \qquad \mathrm{j}=1, 2,.., \mathrm{m} \quad (24)$$

It can be considered that Eq. (24) is the linearized dynamic equation, where:

$$X_p(t_{initial}) = X_p(1) \quad (25)$$

$$X_p(\mathrm{j+1}) = [\mathrm{E}_x]_\mathrm{j}\, X_p(\mathrm{j}) + E_\mathrm{j}\ ,\ \lambda_1 = [\mathrm{E}_m]_1 \quad (26)$$

$$[\lambda_\mathrm{j}] = [E_x]_\mathrm{j}\lambda_{\mathrm{j}-1} + [\mathrm{E_m}]_\mathrm{j} \quad (27)$$

$$[\mu_\mathrm{ji}] = [E_x]_\mathrm{j}\, [\mu_{\mathrm{j-1,i}}] \quad (\mathrm{i} < \mathrm{j})\ ,\ [\mu_\mathrm{ji}] = [E_\tau]_\mathrm{j}\ (\mathrm{i} = \mathrm{j}) \quad (28)$$

2.3. Formulation of the problem for optimal trajectory with maximum load carried by a biped robot

The synthesizing dynamic motion trajectories problem with maximum dynamic load is possible formulated as the trajectory optimization problem via the formulation of the state space of dynamic equations. From the point to point movement with constraints of joint variable and actuator during the path, the formulation written for maximize the (m_{load}), whilst ensuring that the Equation (12) is satisfied, the torques of joints are bounded by:

$$\tau_{max}(X(t)) \geq \tau(t)\ ; \quad \tau_{min}(X(t)) \leq \tau(t) \qquad (29)$$

τ_{max} (X(t)) and τ_{min} (X (t)) are arbitrary known functions of the joint angles and velocities. As well, the initial and final states should be fulfilled, therefore the requirements below in Equation (30) should be satisfied:

$$X_1(t_{initial}) = q\ (t_{initial}) = X_{1)initial}\ , \text{ with } X_2(t_{initial}) = 0$$
$$X_1(t_{final}) = q\ (t_{final}) = X_{1)\mathrm{final}}\ , \text{ with } X_2\ (t_{final}) = 0 \qquad (30)$$

The joint displacements should be bounded during movement as follow:

$$(X_1) \leq X_1^{+} \text{ and } (X_1) \geq X_1^{-} \qquad (31)$$

Where X_1^{+} and X_1^{-} : upper and lower bounds of the joint variables, respectively. Also another constraint is the upper bound of payload (m_l): is the least value of the static load carrying capacity at the two end positions of path. It must be taken into account that the optimal trajectory of the load must be subject to the constraints of stability and limitations of a jerk. The stability criterion is given by the following equation:

$$L_{min} \leq X_{ZMP} \leq L_{max} \qquad (32)$$

Where, L_{max}(length from the ankle to the toe) and L_{min} (length from the ankle to the heel) are determined by the size of the designed robot's foot. While, the bounded jerk may be represented by the followig equation:

$$\left|\dddot{\theta}_{(t)}\right| \leq \max(\dddot{\theta}) \qquad (33)$$

The above two constraints are verified after determining the optimal trajectory. If they are not achieved, this trajectory is canceled and another trajectory is generated that fulfills all the imposed restrictions. Thus the path between the two points be implemented without any violation of the imposed constraints.

2.4. Solve the trajectory synthesis problem of a biped robot.

To solve the problem of trajectory synthesis, the ILP method is used. Taking into account the following condition [2, 3]:

$$\{\, m_l \leq m_l{}^{+} \,\} \qquad (34)$$

This condition above is very important as it sets the highest value of the dynamic load that the robot can raise within the specified path. This value will then be adopted as an initial value to proceed with the iterative method to calculate the greatest dynamic load in the optimal path. The actuator constraints are described as:

$$\tau(j) \leq \tau_{max}\ (x\ (j)) \ \text{ and } \ \tau(j) \geq \tau_{min}\ (x(j)) \qquad (j=1, 2, \ldots, m\ ) \qquad (35)$$

Using the characteristics of typical torque-speed of DC motors, where this leads to determining the maximum allowable torque depending on these relationships:

$$k_1 - (k_2)\ x_2(j) \text{ should be equivalent to } \longrightarrow \tau(j) \leq \tau_{max}\ (x\ (j))$$
$$k_1 - (k_2)\ x_2(j) \text{ should be equivalent to } \longrightarrow \tau(j) \geq \tau_{min}\ (x\ (j)) \qquad (36)$$

Where k_1 and k_2 : $(n \times 1)$ constant vector and $(n \times n)$ diagonal constant matrix, respectively, determined by the equivalent actuator constants. Now these constraints can be written as:

$$\tau \leq b^{+} = \begin{bmatrix} k_1 - [k_2]\ x_2(1) \\ k_1 - [k_2]\ x_2(2) \\ \ldots\ldots \\ k_1 - [k_2]\ x_2(m) \end{bmatrix} \qquad (37)$$

$$-\tau \le -b^{-} = \begin{bmatrix} -\,k_1 - [k_2]\, x_2(1) \\ -\,k_1 - [k_2]\, x_2(2) \\ \dots\dots \\ -\,k_1 - [k_2]\, x_2(m) \end{bmatrix} \qquad (38)$$

Where b^{+} : the upper bound vector of joint actuator torques and b^{-} : the lower bound vector of joint actuator torques. In other words, b^{+} and b^{-} are functions of the positions and velocities of joints. τ = [τ (1), τ (2), …, τ (m)] : (n x m) vector including the control pattern set point (1 to m). The problem is converted to a linear programming problem if the change of variable as:

$$\tau_{final} = b^{+} - \tau \quad \tau_{final} \ge 0 \ \text{ and it is possible too } \ \tau = b^{+} - \tau_{final} \qquad (39)$$

Also we can get the following Eq. (40) by substituting Eq. (39) into Eq. (38):

$$\tau_{final} \le b^{+} - b^{-} \qquad (40)$$

By using Equation (24), can be written the constraints of joint variable as:

$$x_1^{-} - x_{1p}\,(j+1) \le \lambda_{1j}\, m_l + \textstyle\sum_{i=1}^{j}\, [\mu_{1ji}]\, \tau\,(i) \le x_1^{+} - x_{1p}(j+1) \qquad (j=1, ..., m) \qquad (41)$$

The upper vectors of λ_j and X_p(j+1) are λ_{1j}, x_{1p} (j+1) respectively. μ_{1ji}: upper (n x n) sub matrix of μ_{ji}. Eq. (41) described in the form: [Yj] = [μ_{1j1}, μ_{1j2}, μ_{1jj},0,….,0]

$$\lambda_{1j}\, m_l - [Yj]\, \tau_{final} \le [x_1^{+} - x_{1p}(j+1)] - [Yj]\, b^{+} \qquad (j=1,2,\dots, m) \qquad (42)$$

$$-\lambda_{1j}\, m_l + [Yj]\, \tau_{final} \le [x_{1p}(j+1)] - x_1^{-}] + [Yj]\, b^{+} \qquad (j=1,2,\dots, m\ ) \qquad (43)$$

Where [Yj]: (n × nm) matrix. By Eq. (24) the final state X (m+1) be found:

$$\begin{aligned} X\,(m+1) &= X_p(m+1) + \lambda_m m_l + \textstyle\sum_{i=1}^{m} [\mu_{mi}]\, \tau\,(i) \\ &= X_p(m+1) + [\delta]\, \tau_m = X\,(t_{final}) \end{aligned} \qquad (44)$$

Where [δ] = [μ_{m1} , μ_{m2} , …, μ_{mm}]

Thus, all the constraints can be combined and writing in the form of a matrix as:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \lambda_{1j} & -[Y_j] & 0 \\ -\lambda_{1j} & [Y_j] & 0 \\ \lambda_m & -\delta & -1 \\ -\lambda_m & \delta & -1 \end{bmatrix} \begin{bmatrix} m_l \\ \tau_{final} \\ X \end{bmatrix} \le \begin{bmatrix} m_{max} \\ b^{+} - b^{-} \\ [x_1^{+} - x_{1p}(j+1)] - [Yj]\, b^{+} \\ [x_{1p}(j+1) - x_1^{-}] + [Yj]\, b^{+} \\ X\left(t_{final}\right) - X_p(m+1) - [\delta] b^{+} \\ X_p(m+1) - X\left(t_{final}\right) + [\delta] b^{+} \end{bmatrix} \qquad (45)$$

Each of the constraints and topical functions are linear, so it is a linear programming problem that can increase the dynamic load carrying capacity, decidedly an important requirement.

2.5. Method of the computing

The computing technique of the optimum path problem is formulated in details in Fig. 1. This computational technique includes guessing the elementary control and state variable path. From discretizing the initially specified trajectory into (m) points. m_{load} was obtained to be the maximum load and can be carried in the elementary path with not skip any of the constraints. We must point out that this formulation presented in the current study is unique because it took into consideration additional constraints in calculating the maximum dynamic load, and these constraints are (the stability constraint and the jerk limits) compared to the studies [1-3] that based on actuators torque as major constraints imposed in calculating the dynamic load carrying capacity.

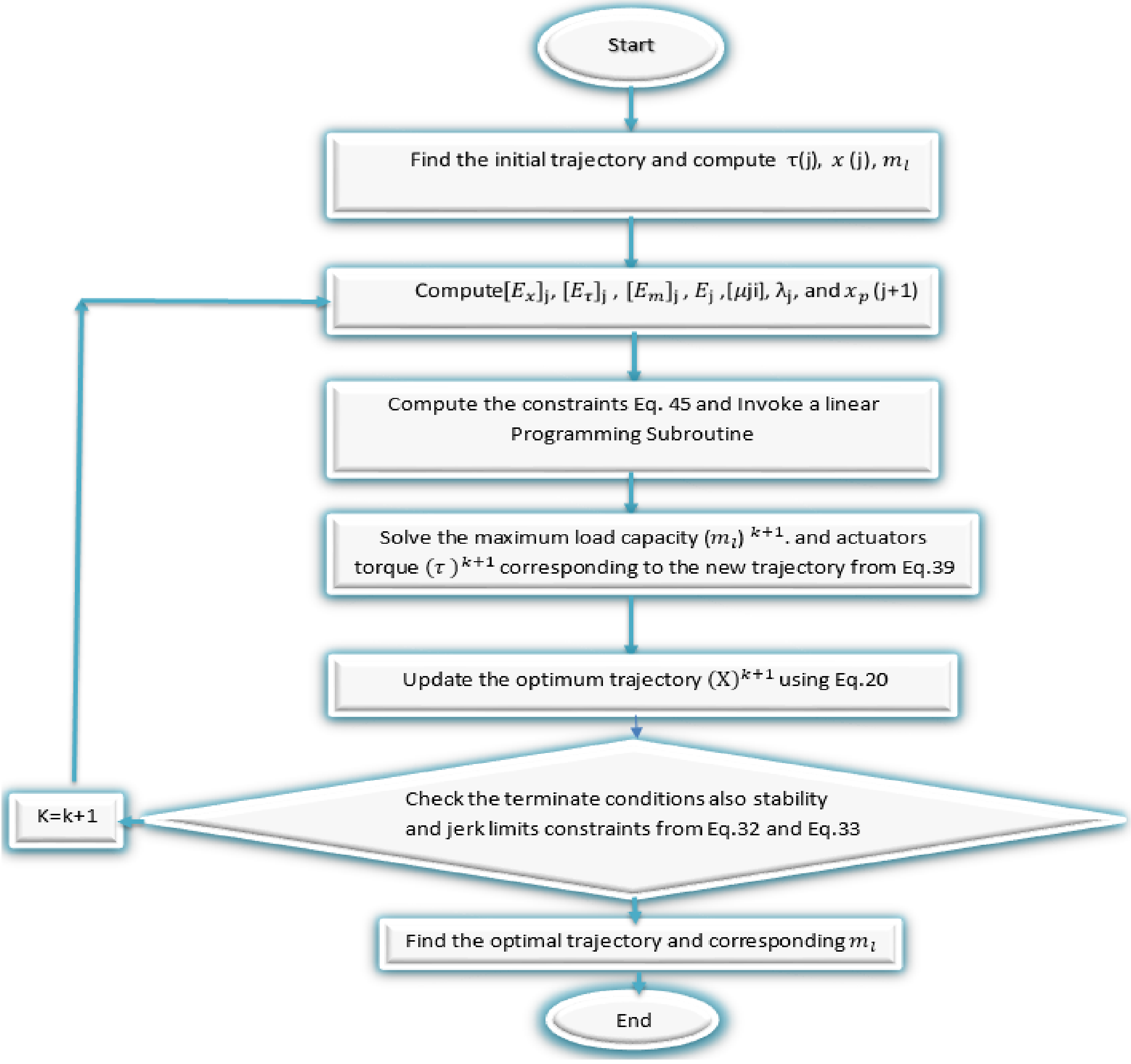

Start
Find the initial trajectory and compute τ(j), x (j), m_l
Compute$[E_x]_j$, $[E_\tau]_j$, $[E_m]_j$, E_j ,[μji], λ_j, and x_p (j+1)
Compute the constraints Eq. 45 and Invoke a linear Programming Subroutine
Solve the maximum load capacity $(m_l)^{k+1}$. and actuators torque $(\tau)^{k+1}$ corresponding to the new trajectory from Eq.39
Update the optimum trajectory $(X)^{k+1}$ using Eq.20
Check the terminate conditions also stability and jerk limits constraints from Eq.32 and Eq.33
K=k+1
Find the optimal trajectory and corresponding m_l
End

Fig. 1. Flow chart of computing procedure for bipedal robot

3. Simulation

At this stage of the study, the maximum DLCC will be found within the optimal trajectory for the end-effector. The synthesizing point to point dynamic motions problem with optimal load carrying capacities is explained in details. An ILP approach is introduced to obtain numerical solutions for the problem of nonlinear trajectory optimization. By using ILP method, the optimal trajectory and maximum DLCC they will be found. The computational technique related to the trajectory optimization problem formulated in terms of the ILP problem is introduced as:

1. Beginning the path, in this stage guess state variable path and an initial control. At the first the initial path discretize via "m" set points, after that calculation τ(j), x (j), m_l for j= 1, …, m, also calculation of the upper bound for the load m_l^+. Then we check if $m_l^+ \leq 0$, then: stop (this indicates the trajectory is impenetrable and the biped robot cannot carry the load into the two end-positions), else: select a possible τ(j), and put the iteration counter k=1.
2. Computing the coefficients $[E_x]_j$, $[E_\tau]_j$, $[E_m]_j$, E_j of the linearized state Equation (20) and repeatedly compute [μji], [λ_j], and x_p (j+1) → {j=1, 2,…,m, i=1, 2,…m}.
3. Using the Equations (42) to (44) to calculation the constraints then conjure a linear programming subroutine. After that, we check if no favorable solution for the LP is available, setting k = k+1 and going back to the second step. Else: solving the maximum load $(m_l)^{k+1}$. and actuators torque $(\tau)^{k+1}$ corresponding to the new trajectory from Equation (39).
4. Updating the optimal path $(X)^{k+1}$(j) → {j=1, 2, …, m+1}, using Equation (20).
5. Checking the end conditions, including the stabilization and jerk conditions. If it satisfies, after that obtaining the optimal path and corresponding m_l, afterward stopping.

Based on the above, all necessary calculations are performed to obtain the optimal DLCC with optimal path. Where the simulation was performed in two straight line and circular paths, on the other hand, actuators constraint, stability criteria, and jerk limitations were imposed as it is in finding dynamic load carrying capacity in given trajectory. The ILP method can be classified as an iterative and approximate numerical method. Undoubtedly, this numerical method allows the user to obtain results with acceptable accuracy and within a scientific method that enjoys sobriety, although it is not devoid of complexity during implementation.

4. Experimental Setup

It is very important to verify the simulation results to see how effective the presented method for calculating DLCC for optimal trajectory. In the current study the suggested method has been applied to the 10 DOF biped robot walking, [3 DOF for each leg, 2 DOF for each arm and 1 DOF for spinal region (head and neck)], which is illustrated in Fig. 2. The parameters of robot shown in the Table 1.

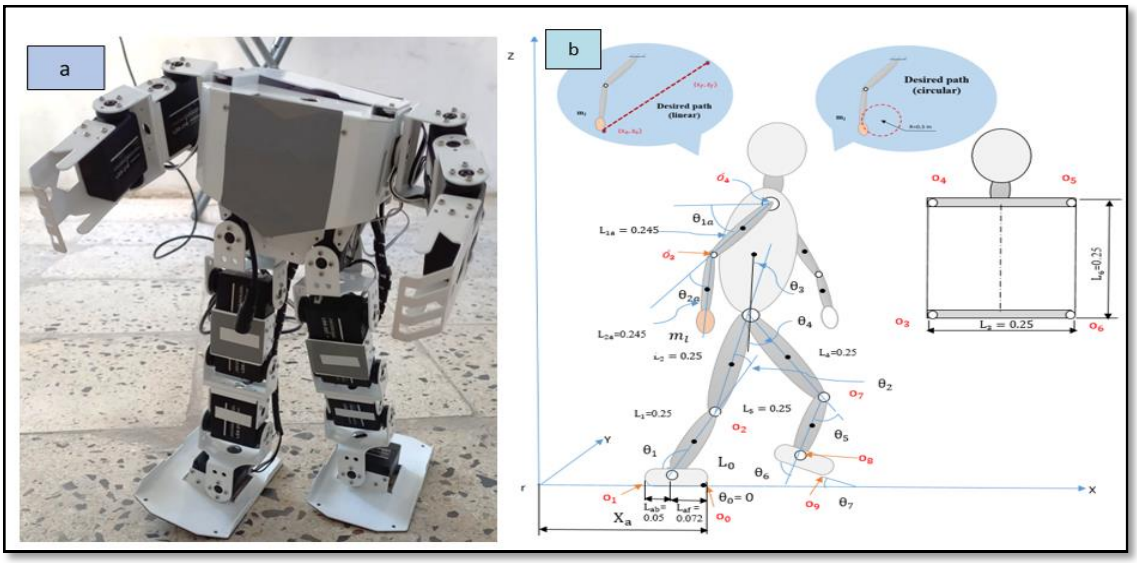


Fig. 2. a) 10 DOF Bipedal Robot. b) The schematic of robot

Tab. 1. The parameters of the bipedal robot

Length of links(m)	L_1	L_2	L_3	L_4	L_5	L_6	L_{1a}	L_{2a}
Values	0.25	0.25	0.25	0.25	0.25	0.25	0.245	0.245
Mass(kg)	m_1	m_2	m_3	m_4	m_5	m_6	m_{1a}	m_{2a}
Values	0.32	0.32	0.32	0.32	0.32	0.32	0.135	0.135
Mass of end effector (kg)	m_e							
Value	0.11							
Moment of inertia(kgm^2)	I_1	I_2	I_3	I_4	I_5	I_6	I_{1a}	I_{2a}
Values	0.0983	0.0983	0.0983	0.0983	0.0983	0.0983	0.054	0.054
Max. no load speed(Rad/sec)	ω_1	ω_2	ω_3	ω_4	ω_5	ω_{1a}	ω_{2a}	
Values	2.5	2.5	2.5	2.5	2.5	2.5	2.5	
Actuator stall torque(Nm)	τ_{s1}	τ_{s2}	τ_{s3}	τ_{s4}	τ_{s5}	τ_{s1a}	τ_{s2a}	
Values	30	30	30	30	30	17	17	
Length from the ankle to the toe(m)	L_{af}							
Value	0.072							
Length from the ankle to the heel(m)	L_{ab}							
Value	0.05							

The following procedure was used to carry out the experimental work:

- Starting the Lobot servo control software with the servo controller of the bipedal robot, then we make sure that all options are working and fully activated and viewing through the interface of the program on the computer. To ensure valid results, the parts of the bipedal robot are calibrated, also checking all the robot components of the external structure, then servomotors, sensors and the rest of the auxiliary parts.
- Angular positions and velocities can be set and adjusted by servomotor slider and deviation of the servo which is appears clearly on the program interface. All the values of the angular positions and angular velocities of the end effector of the bipedal robot when it moves in the optimal trajectory that was generated from the proposed iterative attempts by using the ILP method can be used in order to program and move the arm within the optimal path the of the robot experimentally. In addition, auxiliary measuring devices were also used to verify the correctness of the obtained values of positions, velocities.
- At this stage, the bipedal robot is allowed to move forward within the path specified for it, while the command is given to the (arm or end-effector) of the bipedal robot to move within the optimal path that was generated through the ILP method, to carry out the path from the first point to the final point of the load. Taking into account the test conditions and not exceeding the constraints imposed. Where the values of the angular positions and velocities are checked, also the angular acceleration along the period of the load path, using auxiliary devices and equipment, and then recording all the readings to converge with the simulation results. On the other hand, the calculations of torques for joints are closely related to the angular velocities values recorded during the test. Thus, we obtain a complete estimate of the maximum dynamic load that the robot can carry between two known points within the optimal trajectory. The experimental test of the bipedal robot is illustrated in Fig.3.
- The above steps are repeated for two linear and circular paths. In the current study, a comparison is made between the two highest load values obtained from these two paths. The method of verifying the results of the simulation gives us complete reliability of the results. This is on the one hand. On the other hand, a simulation and an experimental test within two different paths, indicate to a distinct procedure that enhances the study and gives an impression on all test particles. It should be noted that the test is accompanied by many effects on the parts of the robot, particularly the joints, as a result of the dynamic motion. The monitoring of the test is important during the implementation of the trajectory and to ensure that the load reaches the final point without any violation of the imposed constraints. In the stages of the experimental tests that were carried out within this study, there is a lot of keenness to ensure that the motion of the joints is carried out accurately within the specified velocities and acceleration values, and when any change occurs in the angular positions, the test is repeated, and the error is identified and corrected. It is very expected that a sudden change in the angular positions will occur, which will lead to changes in the velocities values, and thus a change in the torques of the joints directly, and this causes confusion in the work of the actuators of joints, and inevitably leads to instability or unbalance for the entire robot. The above-mentioned should be taken into account, especially in the single support stage, as well as the double support for the bipedal walking robot. In addition, maintaining the path of the end effector with the load ensuring that it does not deviate from the trajectory.
- Finally, the data obtained during the test are compiled after fulfilling all the necessary requirements to find the DLCC within the optimal trajectory to verify the simulation results of the numerical analysis approach used in the current study. Where it is possible to view all the data simultaneously through the interface of the operating program of the robot in an updated and continuous manner. In addition to the angular positions of all joints, optimal control over them, and exporting all data to a personal computer in order to schedule it and employ it to the MATLAB.

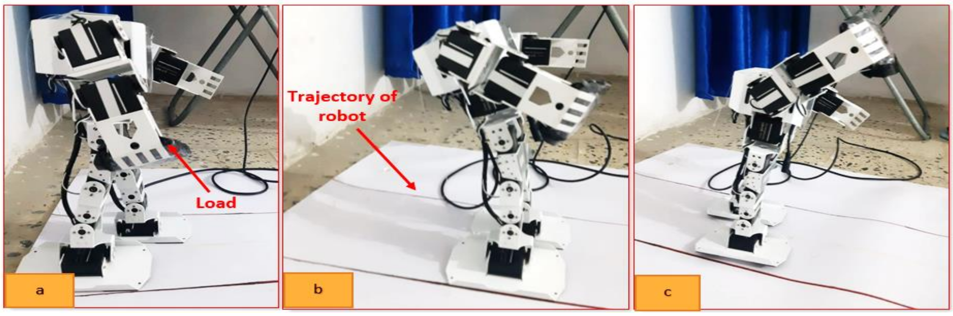


Fig. 3. The experimental test of the bipedal robot a. The load at the beginning of the path b. The load at the middle of the path c. The load at the end of the path

5. Comparison Between Simulation and Experimental Results

Two tests were carried out to find the maximum load for the optimal path of end effector of the bipedal robot with 10 DOF. The first test involves moving the end effector of biped robot in the straight path. The start position of the end effector is (0.23m, 0.345m) while the final position at time (t) =1sec is (1m, 1m). The second test includes the movement of the end effector in a circular path as well at time (t) = 1 sec. Where the circle centered at (x = 0.53 m, y = 0, z = 0.645 m, relative to the point of initiation of movement of the robot, with a radius equal to 0.3 m). In both tests, the robot moves forward carrying the load over the same period of time (1 sec).The simulation and experimental results associated with the optimum trajectory for both tests are presented in Fig. 4.

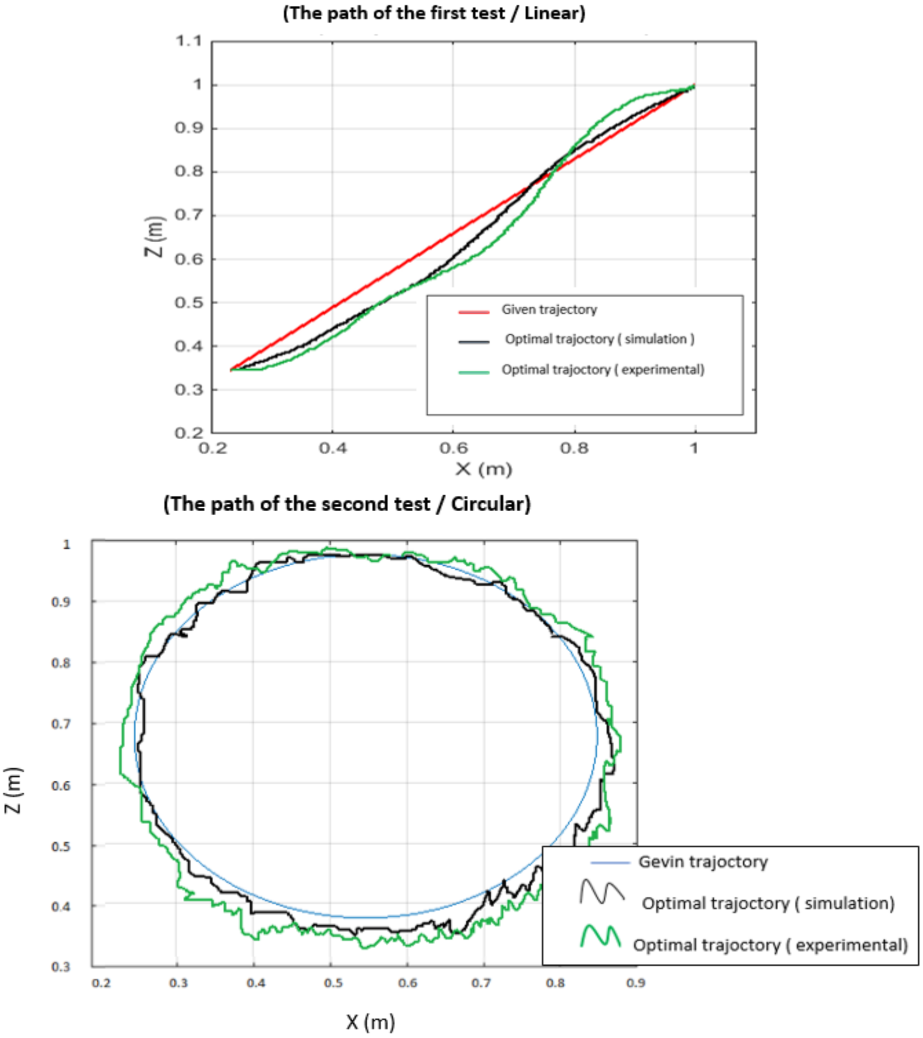


Fig. 4. Optimal trajectories

From the relationships of finding the dynamic load for a specific path, the results showed that the value of the maximum dynamic load for a specific path in the linear path is 0.837 kg. While, for the second test, the maximum allowable load was found 0.51 kg in which the end effector should move in a circular path. In the current study, the optimal trajectory converged after (10) iterations for the first test (linear path), while after (8) iterations for the second test (circular path). Also the value of maximum dynamic load for optimum trajectory DLCC II for both paths (linear and circular) is 1.52 kg and 0.901 kg, respectively. Fig. 5 shows the ILP solution for the maximum dynamic load carrying capacity in each iterations. It is worth mentioning, in the above two cases, in the case of the given trajectory and the optimum trajectory, the maximum dynamic load was determined when imposing the three constraints (actuator torque, stability and jerk limits). In other words, the results showed that the value of the maximum dynamic load when imposing just the major actuators constraint is 1.233 kg and 0.814 kg in the case of the given linear and circular path trajectory, respectively. But when imposing stability and jerk limits constraints together, experimentally the robot will not be able to implement the trajectory and carry the load to the final point because it overturned and did not maintain its stability. Therefore, it is important to add the stability and jerk limits constraints to calculate the maximum dynamic load, which gives an accurate and scientific impression of the ability of the robot to carry the maximum dynamic load. Therefore, the values of DLCC determined within the three constraints were adopted as initial values that were entered into the computational process of ILP method.

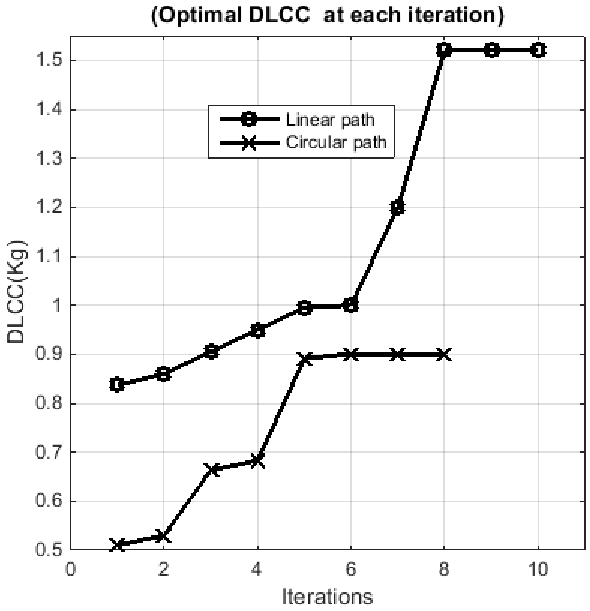


Fig. 5. Optimal dynamic load carrying capacity

The outcome of the comparison between the values of the maximum dynamic load for the linear and circular path showed that the values of DLCC increased by approximately 82% and 78%, respectively after applying the ILP approach, as shown in Figure 6. This indicates the effectiveness of this approach, especially since the amount of increase in the maximum load of the robot increased at an acceptable rate. Predicting the optimal path taken by the robotic arm has taken up a wide space in many researches and has implications for the reality of work in robotic applications, as it gives more accuracy in the implementation of robotic tasks and the speed of implementation of those tasks. However, the use of the presented algorithm in this study is coupled with the fulfillment of all the constraints imposed to find the maximum dynamic load. This was fulfilled in this study, where the above results were drawn in light of the imposed limitations (motor torques, stability criterion and jerk limitations), which will discuss successively.

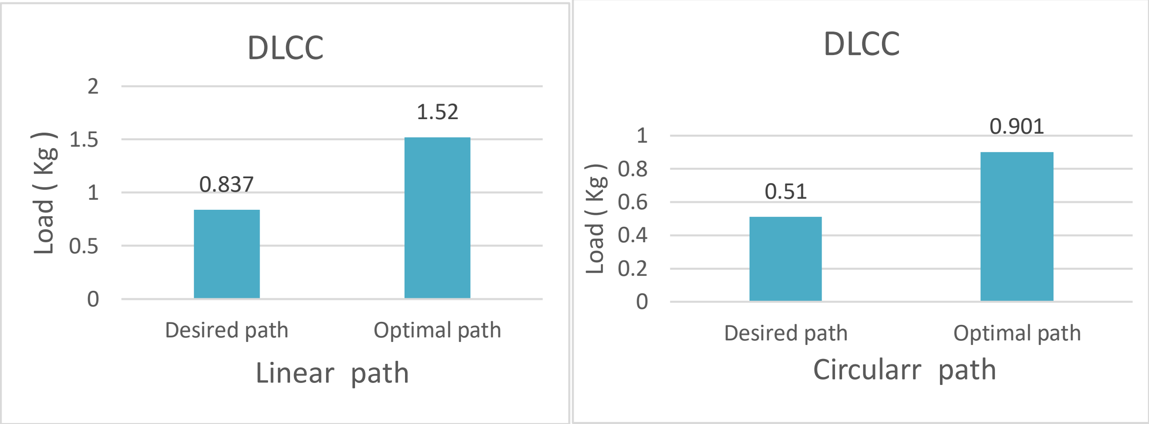


Fig. 6. The values of DLCC

Figures 7 and 8 show the optimal position and velocities for the first and second joint of the right arm of the bipedal robot for both linear and circular trajectory of the end-effector, respectively. As mentioned earlier, the algorithm presented can be used for different robot manipulators with any degree of freedom because of the possibility the solution of inverse kinematics is determined for the trajectory. Anyway, the results presented here are for the first and second joint of the end effector to clarify the activity of the algorithm. Both the first and second joint were assumed to have a value of initial velocity equal to zero. In general, in finding the optimum trajectories for maximum DLCC, the end of the second link should be restricted at the first to move along a given path. Due to the multiplicity of trajectories generated by this approach, angular positions, velocities and accelerations are determined by recognizing the trajectory. This, in fact, distinguishes the iterative method, where several attempts are made to determine the optimal path, and this is accompanied by sequential procedures that the algorithm presented here shows. It must be noted that the results show that the simulated angular positions and velocities are compatible to an acceptable extent with the experimental results. Some deviations may be noticed in the values due to the influences of the experimental test environment, as they often occur.

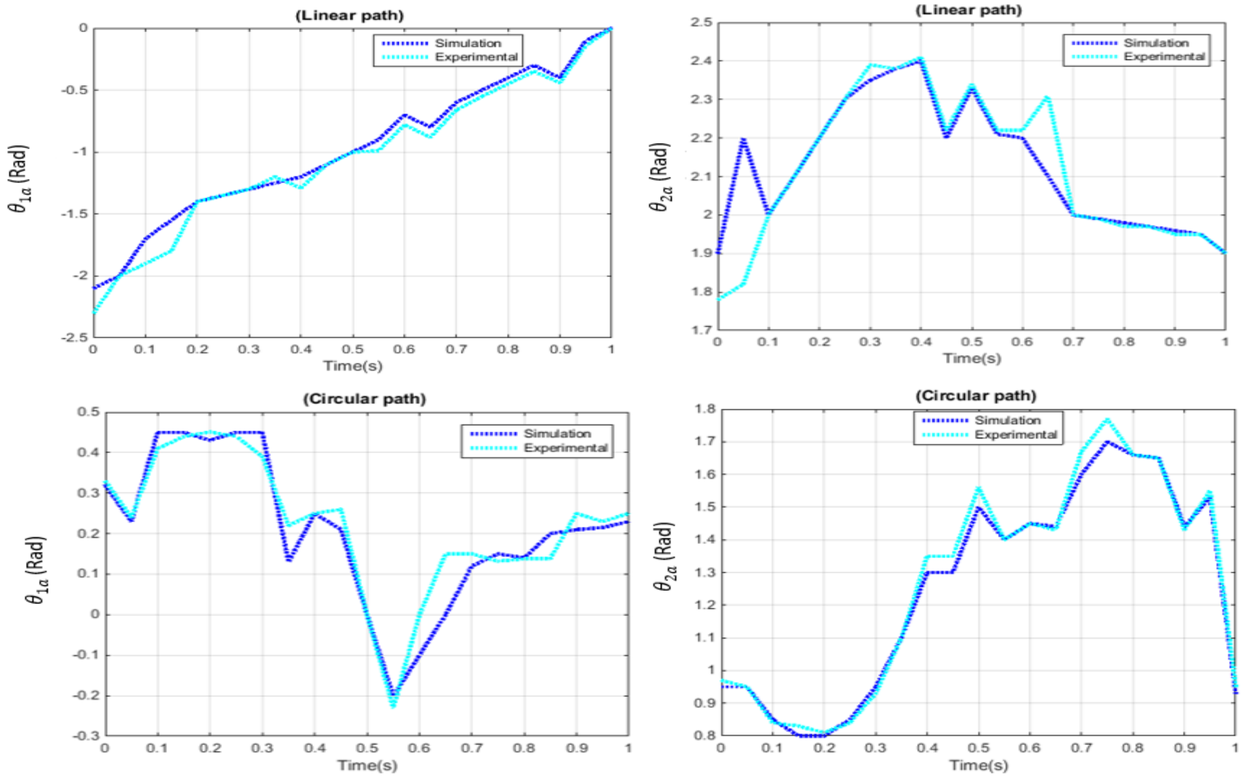


Fig. 7. Optimal angular position for first and second joint of right arm (θ_{1a} and θ_{2a})

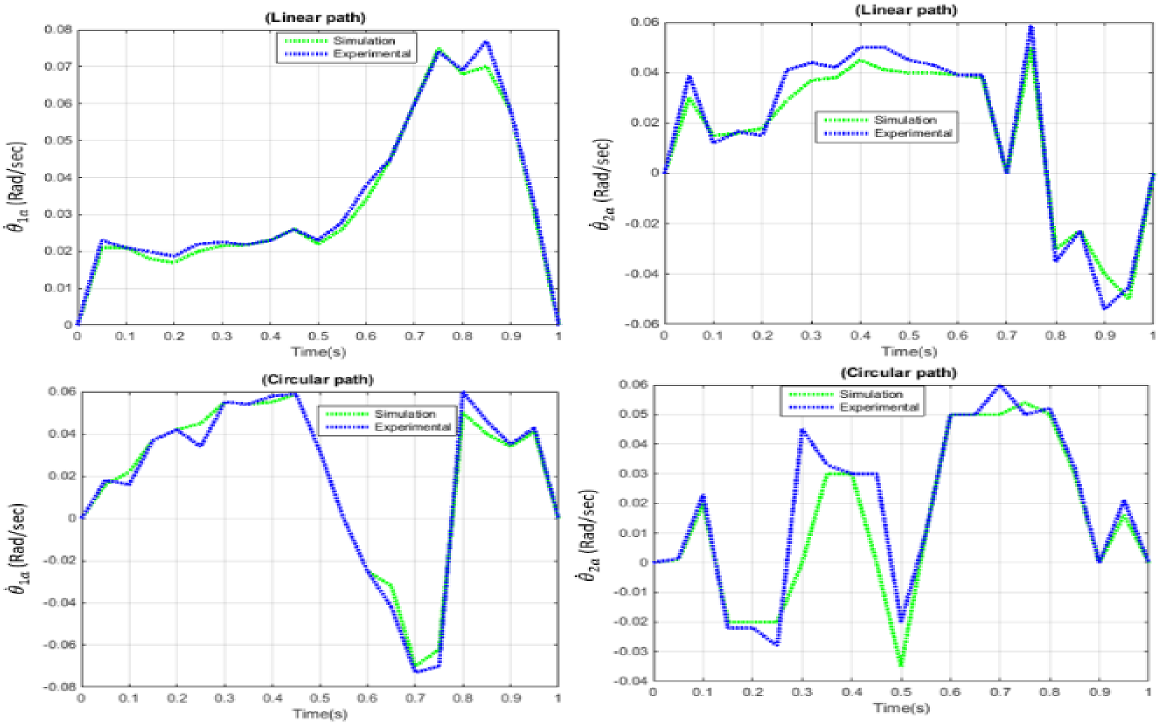


Fig. 8. Optimal angular velocities for first and second joint of right arm ($\dot{\theta}_{1a}$ and $\dot{\theta}_{2a}$)

In the current work the constraints of stability and jerk were taken into account. The stability criterion is important and should be investigated to ensure that it is met. When the robot moves with the load along its path, it may be subject to overturning if the stability condition is not satisfied. The calculation of zero moment point (ZMP) depends entirely on values of angular positions as well as the angular velocities. Taking into account the mass of the links with a load, plus the mass of the end effector. Based on the above, since the iteration method (ILP) is based on the idea of changing trajectories and suggesting sequential trajectories to reach the optimal trajectory, this will inevitably be accompanied by continuous changes in angular positions and angular velocities as well as mass values of links, so the value of ZMP was calculated for the optimal linear and circular paths. The zero moment point ZMP trajectory illustrated in Figure 9. As shown in this Figure, the biped robot is within the conditions of stability, because the ZMP criterion is within the support polygon for all the positions for both linear and circular optimal trajectory. It can also be seen that there is a convergence of the ZMP trajectory with the upper bound in the case of the circular path, this is due to the sudden and continuous changes in angular positions and angular velocities of joints within the optimal trajectory.

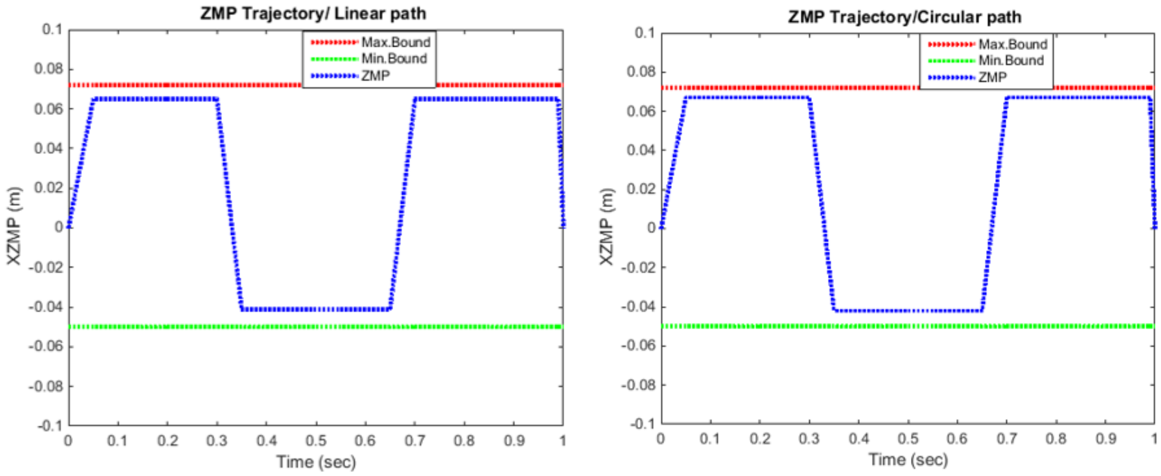


Fig. 9. ZMP trajectory

Figure 10 show the jerk in first and second joint of arm of bipedal robot in the linear and circular optimal trajectory. It can be seen from this figure all curves of (jerk-time) are not exceeded the upper and lower jerk bound. Thus, it can be said that the biped robot can move carrying the load with it within the optimal path without any violation of the jerk limits. As mentioned above, the importance of this limitation is clearly evident when using the iterative method (ILP), because the sudden changes in the positions for the joints stimulate the occurrence of vibrations in the joints. This problem is very important during experimental applications and must be reduced it.

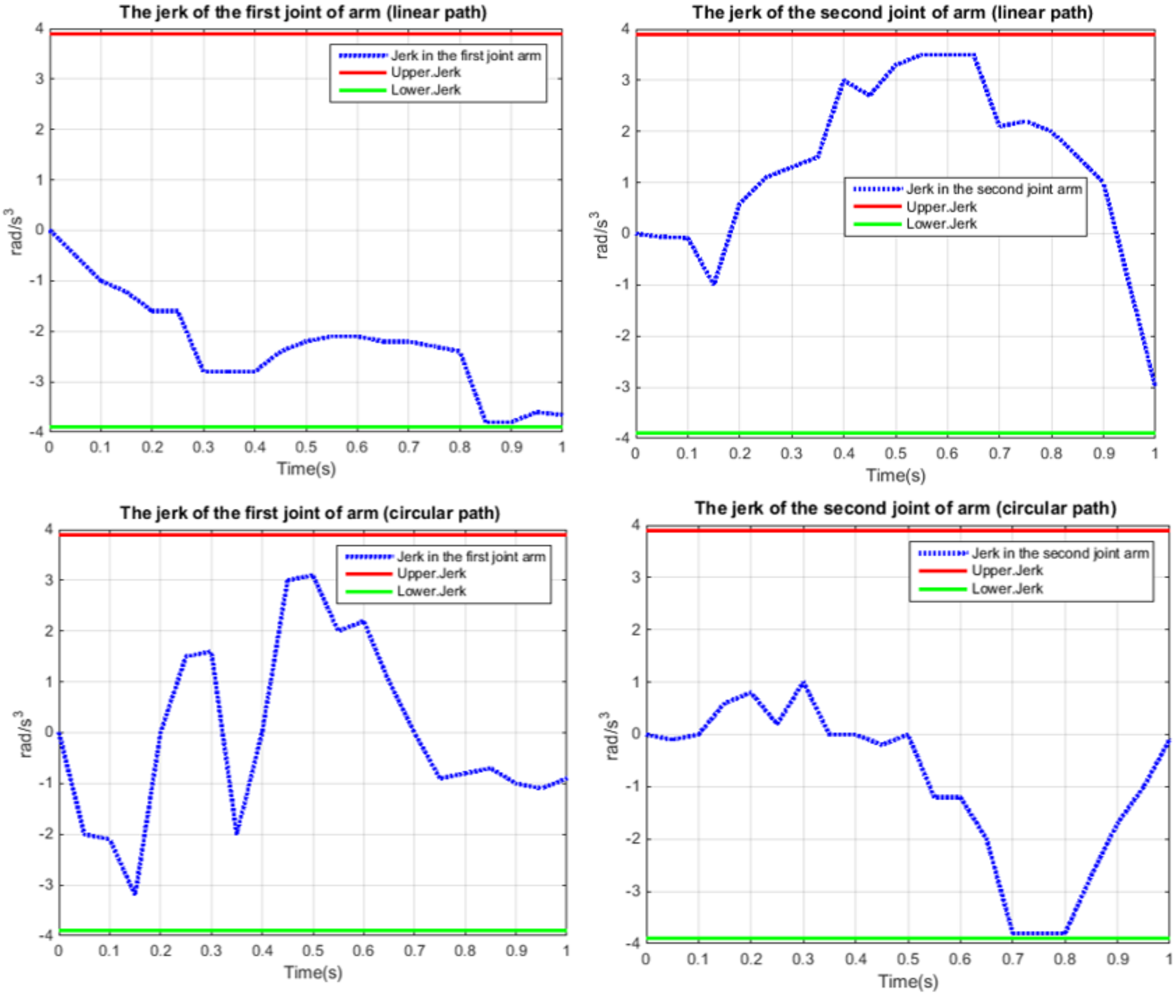
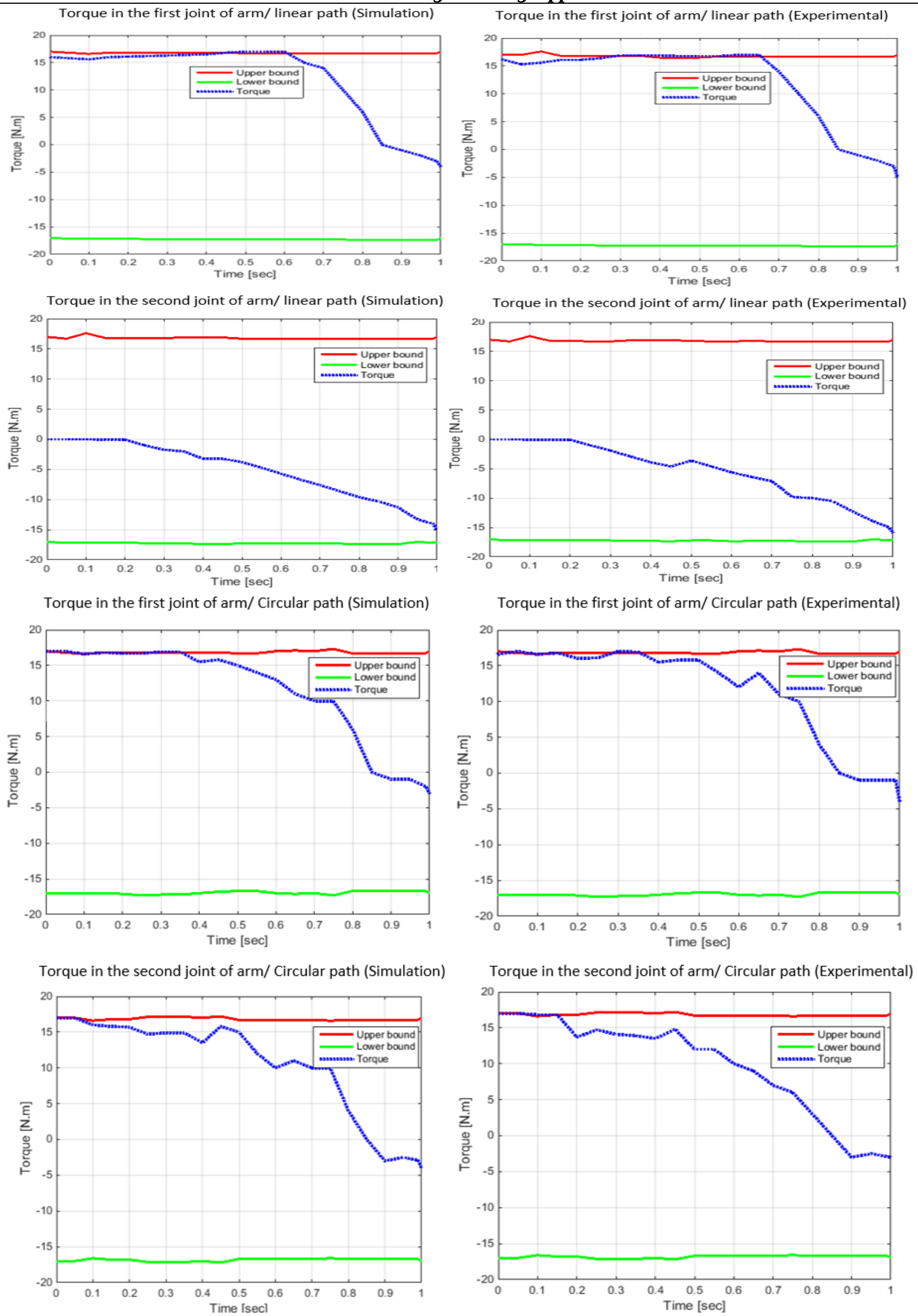


Fig. 10. Jerk in the first and second joint of arm in optimal trajectories

The optimal torques against torques bounds for the maximum allowable load for first and second joints of right arm can be seen in Figure 11.


Torque in the first joint of arm/ linear path (Simulation)

Torque in the first joint of arm/ linear path (Experimental)

Torque in the second joint of arm/ linear path (Simulation)

Torque in the second joint of arm/ linear path (Experimental)

Torque in the first joint of arm/ Circular path (Simulation)

Torque in the first joint of arm/ Circular path (Experimental)

Torque in the second joint of arm/ Circular path (Simulation)

Torque in the second joint of arm/ Circular path (Experimental)

Fig. 11. Optimal torque applied on joints

The torque-time curves above show an initial linear behavior at the beginning of test time besides a non-linear behavior from the middle of the time to the end of time. This behavior refers to the rapid and abrupt changes that occur in angular positions and angular velocities during the optimal trajectory. Besides the above, the gravity and the mass of the end-effector, as well as the vibrations that occur during the dynamic motion, have an impact on describing the behavior of the torque-time curves experimentally. From the results of the optimal torque-time curve for the first joint of the arm linear path, it exhibited the highest value of optimal torque of (17.5 N.m) followed by the first and second joint of the arm circular path with (16.88 N.m). On the other hand, it is very clear that both behaviors of torque-time curves for simulation and experimental tests are near to each other. Anyway, the value of the maximum dynamic load based mainly on the values of the torques, while the torques depend on the trajectories that the load moves with different positions and velocities until the best path is reached. From an applied point of view, the optimal path for the load is chosen depending on the design of the robotic arm and its principle of operation, as well as the energy consumption rates in the actuators of the joints, which should be as low as possible. Also, we do not condone the limitations of stability and jerk, the high stability during the dynamic motion, in addition to the lowest limits of a jerk, which would give the best dynamic performance of the robotic system.

6. Conclusions

The maximum optimal DLCC is the maximum allowable load that can be carried by the robot in an optimal trajectory to implement tasks. It is obvious that manipulation of a load is one of the most momentous tasks of robots and has a myriad of applications in actuality. The main reason that led to the conduct of the current study, is that the optimal DLCC describes the ability to repeatedly lift and carry the load in maximum value by the arm of the bipedal robot 10 DOF in optimal trajectory. The current study includes finding the optimum trajectory and the maximum DLCC in this trajectory using the iterative linear programming ILP method. The ILP method is very accurate, especially when calculating the dynamic load for complex and multiple degrees of freedom systems. The effectiveness of the presented method has been verified, simulations of the first and second joint of an arm of a bipedal robot were performed and all the obtained results were discussed in detail. The first test of the simulation involved a linear trajectory. The first position of the end effector at time = 0 is (0.23m, 0.345m) while the final position at time =1sec is (1m, 1m). The second test includes the movement of the end effector in a circular trajectory, also at time = 1 sec. Where the circle centered (x = 0.53 m, y = 0, z = 0.645 m, relative to the point of initiation of movement of the robot, with a radius equal to 0.3 m). In both tests, the robot moves forward carrying the load over the same period of time (1 sec). The optimal trajectory converged after (10) iterations for the first test (linear path), while after (8) iterations for the second test (circular path). The value of maximum dynamic load for optimal trajectory DLCC for both paths (linear and circular) is (1.52 kg) and (0.901 kg), respectively. An experimental test was performed to validate the results by using 10 DOF bipedal robot. Where all the results of the optimal trajectories, angular positions, and angular velocities were investigated, and it was found that there is a convergence between the results of the simulation and the experimental tests. Thus, the results showed that the technique presented in this study was implemented correctly and its reliability is high. It should be noted that all calculations were performed within constraints (actuators torque, stability conditions and jerk limits), where all constraints were fully met without violating any constraint.

References

[1] L. T. Wang and J. A. B. Ravani," Dynamic load carrying capacity of mechanical manipulators- part I: problem formulation," Journal of dynamic systems, Measurement, and Control, Vol. 110, (1988).

[2] L. T. Wang and J. A. B. Ravani," Dynamic Load Carrying Capacity of Mechanical Manipulators-part II, " Journal of dynamic systems, Measurement, and Control, Vol. 110, No.1, (1988), pp. 53-61.

[3] M. H. Korayem and A. Basu," Formulation and numerical solution of elastic robot dynamic motion with maximum load carrying capacities," Robotica, Vol. 12, No. 3, (1994), pp. 253-261.

[4] M. H. Korayem and, H. Ghariblu," Analysis of wheeled mobile flexible manipulator dynamic motions with maximum load carrying capacities," Robotics and autonomous systems, Vol. 48, Nso. 2-3, (2004), pp. 63-76.

[5] M. H. Korayem, M. Nazemizadeh and H. Rahimi nahooji, "Dynamic load carrying

capacity of flexible manipulators using finite element method and pontryagin’s minimum principle," Journal of optimization in industrial engineering, Vol. 12, (2013), pp.17-24.

[6] A. M. Shafei and M. H. Korayem, "Theoretical and experimental study of dynamic load-carrying capacity for flexible robotic arms in point-to- point motion," Optim Control Appl Meth, Vol. 38, No. 6, (2017), pp.1-10.

[7] M. Irani Rahaghi and F. Barat, "Solving nonlinear optimal path tracking problem using a new closed loop direct–indirect optimization method: application on mechanical manipulators,"Robotica, Vol. 37, No. 1, (2019), pp. 1-23.

[8] A. H. Korayem, M. Irani Rahagi, H. Babaee and M. H. Korayem, "Maximum load of flexible joint manipulators using nonlinear controllers," Robotica, Vol. 35, (2015), pp. 119-142.

[9] A. H. Korayem, S. R. Nekoo and M. H. Korayem,"Optimal sliding mode control design based on the state-dependent Riccati equation for cooperative manipulators to increase dynamic load carrying capacity, "Robotica, Vol. 37, No. 2, (2018), pp. 1-17.

[10] M. H. Korayem and M. Irani," New optimization method to solve motion planning of dynamic systems: application on mechanical manipulators," Multibody system dynamics Vol. 31, No. 2, (2012), pp. 2-21.

[11] H. R. Shafei, M. Bahrami, A. Kamali, and A. M. Shafei, "Determining robot’s maximum dynamic load carrying capacity in point-to-point motion by applying limitation of joints’ torque," *Glob. J. Sci. Res.*, Vol. 3, No. 3, (2015), pp. 12-24.

[12] M.H. Korayem, M. Irani, and S. Rafee Nekoo, "Motion control and dynamic load carrying capacity of mobile robot via nonlinear optimal feedback," *AMAE Int. J. on Manufacturing and Material Science*, (2011).

Appendix (A)

The torque in the first and second joint of the arm (end-effector) of bipedal robot:

$$\tau_{1a} = (\mathrm{A}_1)\,\ddot{\theta}_{1a} + (\mathrm{B}_1)\,\ddot{\theta}_{2a} + \mathrm{g}_1\,(\theta_{1a},\ \theta_{2a}\,,\ \dot{\theta}_{1a}, \dot{\theta}_{2a}\,, m_l) \tag{A.1}$$
$$\tau_{2a} = (\mathrm{A}_2)\,\ddot{\theta}_{1a} + (\mathrm{B}_2)\,\ddot{\theta}_{2a} + \mathrm{g}_2\,(\theta_{1a},\ \theta_{1a}, \dot{\theta}_{1a}, \dot{\theta}_{2a}, m_l)$$

Thus:

$$\begin{bmatrix} \ddot{\theta}_{1a} \\ \ddot{\theta}_{2a} \end{bmatrix} = \begin{bmatrix} \mathrm{A}_1 & \mathrm{B}_1 \\ \mathrm{A}_2 & \mathrm{B}_2 \end{bmatrix}^{-1} \begin{bmatrix} \mathrm{C}_1 \\ \mathrm{C}_2 \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} \tag{A.2}$$

Where:

$\mathrm{C}_1 = \tau_{1a} - \mathrm{g}_1$, $\mathrm{C}_2 = \tau_{2a} - \mathrm{g}_2$, $\mathrm{F} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}$

1) $\frac{\partial f}{\partial x1} \Rightarrow \{\frac{\partial f}{\partial\,\theta_{1a}}, \frac{\partial f}{\partial\,\theta_{2a}}\}$

$$f_{q_{1a}} = \begin{bmatrix} \frac{\partial f_1}{\partial\,\theta_{1a}} \\ \frac{\partial f_2}{\partial\,\theta_{1a}} \end{bmatrix} \tag{A.3}$$

$$f_{q_{2a}} = \begin{bmatrix} \frac{\partial f_1}{\partial\,\theta_{2a}} \\ \frac{\partial f_2}{\partial\,\theta_{2a}} \end{bmatrix} \tag{A.4}$$

2) $\frac{\partial f}{\partial x2} \Rightarrow \{\frac{\partial f}{\partial \dot{\Theta}_{1a}}, \frac{\partial f}{\partial \dot{\Theta}_{2a}}\}$

$$f_{\dot{q}_1} = \begin{bmatrix} \frac{\partial f_1}{\partial \dot{\Theta}_{1a}} \\ \frac{\partial f_2}{\partial \dot{\Theta}_{1a}} \end{bmatrix} \tag{A.5}$$

$$f_{\dot{q}_2} = \begin{bmatrix} \frac{\partial f_1}{\partial \dot{\Theta}_{2a}} \\ \frac{\partial f_2}{\partial \dot{\Theta}_{2a}} \end{bmatrix} \tag{A.6}$$

3) $\frac{\partial f}{\partial \tau} \Rightarrow \{\frac{\partial f}{\partial \tau 1a}, \frac{\partial f}{\partial \tau 2a}\}$

$$f_{\tau 1a} = \begin{bmatrix} \frac{\partial f_1}{\partial \tau_{1a}} \\ \frac{\partial f_2}{\partial \tau_{1a}} \end{bmatrix} \tag{A. 7}$$

$$f_{\tau 2a} = \begin{bmatrix} \frac{\partial f_1}{\partial \tau_{2a}} \\ \frac{\partial f_2}{\partial \tau_{2a}} \end{bmatrix} \tag{A. 8}$$

4) $\frac{\partial f}{\partial m} \quad \{\frac{\partial f_1}{\partial m}, \frac{\partial f_2}{\partial m}\} \Rightarrow f_m = \begin{bmatrix} \frac{\partial f_1}{\partial m} \\ \frac{\partial f_2}{\partial m} \end{bmatrix}$ (A.9)

Expanding the nonlinear function in the Taylor series. With the need to remove the higher-order terms:

$$f_1^{k+1}(j) = f^k\,(\mathrm{j}) + \begin{bmatrix} \frac{\partial f_1}{\partial\, \Theta_{1a}} \\ \frac{\partial f_2}{\partial\, \Theta_{1a}} \end{bmatrix} (\Theta_{1a}^{k+1}\,(\mathrm{j}) - \Theta_{1a}^{k}(\mathrm{j})) + \begin{bmatrix} \frac{\partial f_1}{\partial \dot{\Theta}_{1a}} \\ \frac{\partial f_2}{\partial \dot{\Theta}_{1a}} \end{bmatrix} ({\dot{\Theta}_{1a}}^{k+1}(\mathrm{j}) - {\dot{\Theta}_{1a}}^{k}\,(\mathrm{j})\,) + \begin{bmatrix} \frac{\partial f_1}{\partial \tau 1a} \\ \frac{\partial f_2}{\partial \tau 1a} \end{bmatrix} (\tau_{1a}^{k+1}\,(\mathrm{j}) - \tau_{1a}^{k}\,(\mathrm{j})\,) + \begin{bmatrix} \frac{\partial f_1}{\partial m} \\ \frac{\partial f_2}{\partial m} \end{bmatrix} (m_l^{k+1} - m_l^{k}) \tag{A.10}$$

The above Equation (A.10) for fist joint of the arm (end-effector) of bipedal robot, now for second joint:

$$f_2^{k+1}\,(\mathrm{j}) = f^k\,(\mathrm{j}) + \begin{bmatrix} \frac{\partial f_1}{\partial\, \Theta_{2a}} \\ \frac{\partial f_2}{\partial\, \Theta_{2a}} \end{bmatrix} (\Theta_{2a}^{k+1}\,(\mathrm{j}) - \Theta_{2a}^{k}(\mathrm{j})) + \begin{bmatrix} \frac{\partial f_1}{\partial \dot{\Theta}_{2a}} \\ \frac{\partial f_2}{\partial \dot{\Theta}_{2a}} \end{bmatrix} ({\dot{\Theta}_{2a}}^{k+1}\,(\mathrm{j}) - {\dot{\Theta}_{2a}}^{k}\,(\mathrm{j})\,) + \begin{bmatrix} \frac{\partial f_1}{\partial \tau 2a} \\ \frac{\partial f_2}{\partial \tau 2a} \end{bmatrix} (\tau_{2a}^{k+1}\,(\mathrm{j}) - \tau_{2a}^{k}\,(\mathrm{j})\,) + \begin{bmatrix} \frac{\partial f_1}{\partial m} \\ \frac{\partial f_2}{\partial m} \end{bmatrix} (m_l^{k+1} - m_l^{k}) \tag{A.11}$$

By combining terms and rewriting the linearized Equations (A.10 and A.11) more compactly as Equations (A.12 and A.13) below:

$$f_1^{k+1}(j) = \begin{bmatrix} \frac{\partial f_1}{\partial\, \Theta_{1a}} \\ \frac{\partial f_2}{\partial\, \Theta_{1a}} \end{bmatrix} (\Theta_{1a}^{k+1}\,(\mathrm{j})) + \begin{bmatrix} \frac{\partial f_1}{\partial \dot{\Theta}_{1a}} \\ \frac{\partial f_2}{\partial \dot{\Theta}_{1a}} \end{bmatrix} ({\dot{\Theta}_{1a}}^{k+1}\,(\mathrm{j})) + \begin{bmatrix} \frac{\partial f_1}{\partial \tau 1a} \\ \frac{\partial f_2}{\partial \tau 1a} \end{bmatrix} (\tau_{1a}^{k+1}\,(\mathrm{j})) + \begin{bmatrix} \frac{\partial f_1}{\partial m} \\ \frac{\partial f_2}{\partial m} \end{bmatrix} (m_l^{k+1}) + f^k\,(\mathrm{j}) - \begin{bmatrix} \frac{\partial f_1}{\partial\, \Theta_{1a}} \\ \frac{\partial f_2}{\partial\, \Theta_{1a}} \end{bmatrix} (\Theta_{1a}^{k}(\mathrm{j})) - \begin{bmatrix} \frac{\partial f_1}{\partial \dot{\Theta}_{1a}} \\ \frac{\partial f_2}{\partial \dot{\Theta}_{1a}} \end{bmatrix} {\dot{\Theta}_{1a}}^{k}\,(\mathrm{j})\,) - \begin{bmatrix} \frac{\partial f_1}{\partial \tau 1a} \\ \frac{\partial f_2}{\partial \tau 1a} \end{bmatrix} (\tau_{1a}^{k}\,(\mathrm{j})\,) - \begin{bmatrix} \frac{\partial f_1}{\partial m} \\ \frac{\partial f_2}{\partial m} \end{bmatrix} (m_l^{k}) \tag{A.12}$$

$$\mathrm{f}_2^{k+1}\,(\mathrm{j}) = \begin{bmatrix} \frac{\partial f_1}{\partial\, \Theta_{2a}} \\ \frac{\partial f_2}{\partial\, \Theta_{2a}} \end{bmatrix} (\Theta_{2a}^{k+1}\,(\mathrm{j})) + \begin{bmatrix} \frac{\partial f_1}{\partial \dot{\Theta}_{2a}} \\ \frac{\partial f_2}{\partial \dot{\Theta}_{2a}} \end{bmatrix} ({\dot{\Theta}_{2a}}^{k+1}\,(\mathrm{j})) + \begin{bmatrix} \frac{\partial f_1}{\partial \tau 2a} \\ \frac{\partial f_2}{\partial \tau 2a} \end{bmatrix} (\tau_{2a}^{k+1}\,(\mathrm{j})) + \begin{bmatrix} \frac{\partial f_1}{\partial m} \\ \frac{\partial f_2}{\partial m} \end{bmatrix} (m_l^{k+1}) + f^k\,(\mathrm{j}) - \begin{bmatrix} \frac{\partial f_1}{\partial\, \Theta_{2a}} \\ \frac{\partial f_2}{\partial\, \Theta_{2a}} \end{bmatrix}$$

$$(\Theta_{2a}^{k}(\mathrm{j}))- \begin{bmatrix} \frac{\partial f_1}{\partial \dot{\Theta}_{2a}} \\ \frac{\partial f_2}{\partial \dot{\Theta}_{2a}} \end{bmatrix} \dot{\Theta}_{2a}{}^{k}\ (\mathrm{j})\ )- \begin{bmatrix} \frac{\partial f_1}{\partial \tau 2a} \\ \frac{\partial f_2}{\partial \tau 2a} \end{bmatrix} (\tau_{2a}^{k}\ (\mathrm{j})\ )- \begin{bmatrix} \frac{\partial f_1}{\partial m} \\ \frac{\partial f_2}{\partial m} \end{bmatrix} (m_l^k)$$

The above equations for the iteration $(\mathrm{k} + \mathrm{l})^{th}$, it is possible to remove the superscript k+1. By substitution Equations (A.12 and A.13) into Eq. (13), thus lead to:

$$\dot{\Theta}_{1a}(\mathrm{j+1}) = \mathrm{p} \begin{bmatrix} \frac{\partial f_1}{\partial\ \Theta_{1a}} \\ \frac{\partial f_2}{\partial\ \Theta_{1a}} \end{bmatrix} \Theta_{1a}(\mathrm{j}) +[\ \mathrm{p} \begin{bmatrix} \frac{\partial f_1}{\partial \dot{\Theta}_{1a}} \\ \frac{\partial f_2}{\partial \dot{\Theta}_{1a}} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}]\ \dot{\Theta}_{1a}(\mathrm{j}) +[\ \mathrm{p} \begin{bmatrix} \frac{\partial f_1}{\partial \tau 1a} \\ \frac{\partial f_2}{\partial \tau 1a} \end{bmatrix}]\ \tau_{1a}(\mathrm{j}) + \mathrm{p} \begin{bmatrix} \frac{\partial f_1}{\partial m} \\ \frac{\partial f_2}{\partial m} \end{bmatrix} m_l + \mathrm{p}[\ f^k\ (\mathrm{j})\ - \tag{A.13}$$

$$\begin{bmatrix} \frac{\partial f_1}{\partial\ \Theta_{1a}} \\ \frac{\partial f_2}{\partial\ \Theta_{1a}} \end{bmatrix} \Theta_{1a}^{k}(\mathrm{j}) - \begin{bmatrix} \frac{\partial f_1}{\partial \dot{\Theta}_{1a}} \\ \frac{\partial f_2}{\partial \dot{\Theta}_{1a}} \end{bmatrix} \dot{\Theta}_{1a}{}^{k}\ \ (\mathrm{j}) - \begin{bmatrix} \frac{\partial f_1}{\partial \tau 1a} \\ \frac{\partial f_2}{\partial \tau 1a} \end{bmatrix} \tau_1^k\ (\mathrm{j}) - \begin{bmatrix} \frac{\partial f_1}{\partial m} \\ \frac{\partial f_2}{\partial m} \end{bmatrix} (m_l^k)] \tag{A.14}$$

$$\dot{\Theta}_{2a}((\mathrm{j+1}) = \ \mathrm{p} \begin{bmatrix} \frac{\partial f_1}{\partial\ \Theta_{2a}} \\ \frac{\partial f_2}{\partial\ \Theta_{2a}} \end{bmatrix}\ \Theta_{2a}(\mathrm{j})\ +[\ \mathrm{p} \begin{bmatrix} \frac{\partial f_1}{\partial \dot{\Theta}_{2a}} \\ \frac{\partial f_2}{\partial \dot{\Theta}_{2a}} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}]\ \dot{\Theta}_{2a}(\mathrm{j}) +[\ \mathrm{p} \begin{bmatrix} \frac{\partial f_1}{\partial \tau 2a} \\ \frac{\partial f_2}{\partial \tau 2a} \end{bmatrix}]\ \tau_{2a}(\mathrm{j}) +\ \mathrm{p} \begin{bmatrix} \frac{\partial f_1}{\partial m} \\ \frac{\partial f_2}{\partial m} \end{bmatrix} m_l\ +\ \mathrm{p}[\ \ f^k\ (\mathrm{j})\ -$$

$$\begin{bmatrix} \frac{\partial f_1}{\partial\ \Theta_{2a}} \\ \frac{\partial f_2}{\partial\ \Theta_{2a}} \end{bmatrix} \Theta_{1a}^{k}(\mathrm{j}) - \begin{bmatrix} \frac{\partial f_1}{\partial \dot{\Theta}_{2a}} \\ \frac{\partial f_2}{\partial \dot{\Theta}_{2a}} \end{bmatrix} \dot{\Theta}_{2a}{}^{k}\ \ (\mathrm{j}) - \begin{bmatrix} \frac{\partial f_1}{\partial \tau 2} \\ \frac{\partial f_2}{\partial \tau 2} \end{bmatrix} \tau_{1a}^{k}\ (\mathrm{j}) - \begin{bmatrix} \frac{\partial f_1}{\partial m} \\ \frac{\partial f_2}{\partial m} \end{bmatrix} (m_l^k)] \tag{A.15}$$

And:

$$\Theta_{1a}(\mathrm{j+1}) = \ \Theta_{1a}(\mathrm{j}) + \mathrm{p}\ \dot{\Theta}_{1a}(\mathrm{j}) \tag{A.16}$$

$$\Theta_{2a}(\mathrm{j+1}) = \ \Theta_{2a}(\mathrm{j}) + \mathrm{p}\ \dot{\Theta}_{2a}(\mathrm{j}) \tag{A.17}$$

Thus, the above Equations (A.15 to A.17) can be rearranged in matrix form as follows:

$$\mathrm{X}\ (\mathrm{j+1}) = [E_x]_{\mathrm{j}} \mathrm{X}\ (\mathrm{j}) + [E_\tau]_{\mathrm{j}}\ \tau_{\mathrm{j}} + [E_m]_{\mathrm{j}}\ \ m_l + E_{\mathrm{j}} \tag{A.18}$$

Where:

$$[\mathrm{E_x}]_{\mathrm{j}} = \begin{bmatrix} 1 & 0 & \varepsilon & 0 \\ 0 & 1 & 0 & p \\ p\frac{\partial f_1}{\partial q_{1a}} & p\frac{\partial f_1}{\partial q_{2a}} & p\frac{\partial f_1}{\partial \dot{q}_{1a}} + 1 & p\frac{\partial f_1}{\partial \dot{q}_{2a}} \\ p\frac{\partial f_2}{\partial q_{1a}} & p\frac{\partial f_2}{\partial q_{2a}} & p\frac{\partial f2}{\partial \dot{q}_{1a}} & p\frac{\partial f_2}{\partial \dot{q}_{2a}} + 1 \end{bmatrix} \tag{A.19}$$

$$[\mathrm{E_\tau}]_{\mathrm{j}} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ p\frac{\partial f_1}{\partial \tau_{1a}} & p\frac{\partial f_1}{\partial \tau_{2a}} \\ p\frac{\partial f_2}{\partial \tau_{1a}} & p\frac{\partial f_2}{\partial \tau_{2a}} \end{bmatrix} \tag{A.20}$$

$$[E_m]_{\mathrm{j}} = \begin{bmatrix} 0 \\ 0 \\ \frac{\partial f_1}{\partial\ m_l} \\ \frac{\partial f_2}{\partial\ m_l} \end{bmatrix} \quad (\mathrm{A.21}) \qquad , \qquad \mathrm{E_j} = \begin{bmatrix} 0 \\ 0 \\ \mathrm{pX_1^k} \\ \mathrm{pX_2^k} \end{bmatrix} \tag{A.22}$$

Where:

$$X_1^k = f^k\ (\mathrm{j}) - \begin{bmatrix} \frac{\partial f_1}{\partial\ \Theta_{1a}} \\ \frac{\partial f_2}{\partial\ \Theta_{1a}} \end{bmatrix} (\Theta_{1a}^{k}(\mathrm{j}))- \begin{bmatrix} \frac{\partial f_1}{\partial \dot{\Theta}_{1a}} \\ \frac{\partial f_2}{\partial \dot{\Theta}_{1a}} \end{bmatrix} \dot{\Theta}_{1a}{}^{k}\ (\mathrm{j}))- \begin{bmatrix} \frac{\partial f_1}{\partial \tau 1a} \\ \frac{\partial f_2}{\partial \tau 1a} \end{bmatrix} (\tau_{1a}^{k}\ (\mathrm{j}))- \begin{bmatrix} \frac{\partial f_1}{\partial m} \\ \frac{\partial f_2}{\partial m} \end{bmatrix} (m_l^k)$$

$$X_2^k = f^k\,(\mathrm{j}) - \begin{bmatrix} \dfrac{\partial f_1}{\partial\,\Theta_{2a}} \\ \dfrac{\partial f_2}{\partial\,\Theta_{2a}} \end{bmatrix} (\Theta_{2a}^k(\mathrm{j})) - \begin{bmatrix} \dfrac{\partial f_1}{\partial\dot{\Theta}_{2a}} \\ \dfrac{\partial f_2}{\partial\dot{\Theta}_{2a}} \end{bmatrix} \dot{\Theta}_{2a}{}^{k}\,(\mathrm{j})) - \begin{bmatrix} \dfrac{\partial f_1}{\partial\tau 2a} \\ \dfrac{\partial f_2}{\partial\tau 2a} \end{bmatrix} (\tau_{2a}^k\,(\mathrm{j})) - \begin{bmatrix} \dfrac{\partial f_1}{\partial m} \\ \dfrac{\partial f_2}{\partial m} \end{bmatrix} (m_l^k) \tag{A.23}$$

Follow this article at the following site:

Ammar Fadhil Hussein AL-Maliki[1] & Moharam Habibnejad Korayem, Trajectory Optimization of Biped Robots with Maximum Load Carrying Capacities: Iterative Linear Programming Approach. IJIEPR 2023; 34 (3) :1-21
URL: http://ijiepr.iust.ac.ir/article-1-1797-en.html