Transaction B: Mechanical Engineering
Vol. 17, No. 3, pp. 229–239

Research Note

Synthesis of Cable Driven Robots' Dynamic Motion with Maximum Load Carrying Capacities: Iterative Linear Programming Approach

M.H. Korayem[1,*], Kh. Najafi[2] and M. Bamdad[1]

Abstract. *In this paper, the general dynamic equation of motion of Cable Driven Robots (CDRs) is obtained from Lagrangian formulation. A computational technique is developed for obtaining an optimal trajectory to maximize the dynamic load carrying capacity for a given point-to-point task. Dynamic equations are organized in a closed form and are formulated in the state space form. In order to find the Dynamic Load Carrying Capacity (DLCC) of CDRs, joint actuators torque, and robot workspace constraints for obtaining the positive tension in cables are considered. The problem is formulated as a trajectory optimization problem, which fundamentally is a constrained nonlinear optimization problem. Then, the Iterative Linear Programming (ILP) method is used to solve the optimization problem. Finally, a numerical example involving a 6 d.o.f CDR is presented and, due to validation, the results of the ILP method are compared with the optimal control method.*

Keywords: *Cable driven robot; Dynamic load; Optimal trajectory; Linear programming.*

INTRODUCTION

Cable driven robots are a special form of parallel robot, in which the rigid links are replaced by the cables. Cable Driven Robots (CDRs) have some advantages over conventional serial and parallel robots. They have a rather large workspace, low inertia properties and a high payload to weight ratio. On the other hand, the main disadvantage of CDRs is that the cables are only capable of pulling, which can cause instabilities in their motion.

One of the early works in Robocrane was developed by NIST (National Institute of Standard Technology) in order to automate a crane for lifting operations. It is particularly a cable-driven manipulator based on a modification of the 6 degrees of freedom (d.o.f) Gough-Stewart platform, where the linear actuators have been replaced by cables [1].

Alp and Agrawal determined the statically reachable workspace for a 6 d.o.f spatial cable CDR, which has been built and tested [2]. With the ever-growing application of these robots, one of the questions produced is: "what is the optimal trajectory with regard to maximum allowable payload?" The main aim of this paper is to find a proper answer to the above question.

The Dynamic Load Carrying Capacity (DLCC) of a manipulator is defined as the maximum payload that the manipulator can repeatedly carry in a defined trajectory. However, to determine the maximum allowable load of a robot, the inertia effect of the load along a desired trajectory, as well as the manipulator dynamics, must be taken into account. The literature on determining DLCC on different types of robotic system is fairly rich. Wang and Ravani offered a method for determining the maximum load capacity of fixed base robots, and treated the problem as the optimization of trajectory. In this method, the torque capacity of actuators

1. *School of Mechanical Engineering, Iran University of Science and Technology, Tehran, P.O. Box 13114-16846, Iran.*
2. *Colleges of Mechanical Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran.*
*. *Corresponding author. E-mail: hkorayem@iust.ac.ir*

Received 4 September 2009; received in revised form 12 January 2010; accepted 8 February 2010

is considered as the main constraint [3]. Korayem and Gariblu acquired the maximum load capacity of manipulators for two points at a certain time by considering the joint actuator torque, kinematical redundancy and non-holonomic constraints. They have dealt with the problem using iterative linear programming [4].

Korayem and Nikoobin employed an indirect approach, based on the open loop optimal control for obtaining the optimal trajectory of robot manipulators, to maximize the load carrying capacity for a given point-to-point task [5]. Korayem and Bamdad determined the dynamic load carrying capacity of CDRs, regarding the tensile capacity of cables and the actuator torque capacity for a given trajectory in a specified time [6]. The finite element method is used for describing the dynamics of the system and the maximum payload of kinematically redundant flexible manipulators was computed [7-8]. Korayem and Shokri determined the maximum payload capacity for a 6UPS-Stewart platform, by considering the joint actuator torque capacity and the motion accuracy [9].

In this paper, a method is developed for determining the maximum dynamic payload of CDRs between two given end points of their workspace. The load carrying capacity of a robotic system between two points at a specified time does not have a unique value; it depends directly on the selected trajectory between two points. Therefore, the main aim of this paper is to find an optimum trajectory, such that the maximum load can be carried between two end points at a specified time, by considering the actuators torque and robot workspace constraints. An objective function is defined and the nonlinear state space dynamic equations are linearized. Then, the iterative linear programming method is used to numerically solve the linearized trajectory optimization problem. Finally, a numerical example involving a 6 d.o.f CDR is presented and the results are discussed.

MATHEMATICAL MODELING OF CDRS

In this study, the well-known NIST robocrane, as a 6 d.o.f cable driven robot, was considered. This robot is an inverted Stewart platform, in which rigid legs are replaced by cables. Its suspended movable platform (or end-effector) and fixed support are two equilateral triangles, as shown in Figure 1.

The end-effector is kinematically constrained by maintaining tension in all six cables, which terminate in pairs at the vertices of the fixed support. The orientation and position of the end-effector are determined by a six-actuator system, in which each cable is controlled separately. In order to study the kinematics and dynamics of a robocrane, two frames are used:

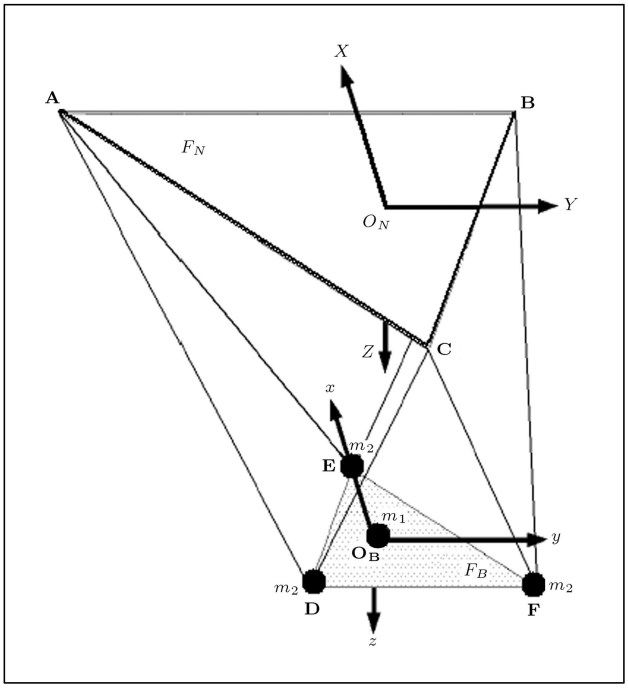

Figure 1. Free-body diagram for the robocrane with point masses.

1. The inertial coordinate, F_N, with its origin in the center of the fixed support (with vertices A, B and C).
2. The body coordinate, F_B, which is similarly connected to the center of mass of the end-effector (with vertices D, E and F).

Here, $\{q\}$ is introduced as follows:

$$\{q\} = [x_D, y_D, z_D, x_E, y_E, z_E, x_F, y_F, z_F]. \quad (1)$$

The elements of generalized coordinates $\{q\}$ are the Cartesian coordinate of vertices of the end-effector, as written in the reference coordinate. These generalized coordinates are not independent and, since the robot has six degrees of freedom, three constraint equations are necessary. These constraint equations are as follows:

$$\begin{cases} (x_D - x_E)^2 + (y_D - y_E)^2 + (z_D - z_E)^2 = (2b)^2 \\ (x_E - x_F)^2 + (y_E - y_F)^2 + (z_E - z_F)^2 = (2b)^2 \\ (x_D - x_F)^2 + (y_D - y_F)^2 + (z_D - z_F)^2 = (2b)^2 \end{cases} \quad (2)$$

In order to derive kinetic and potential energies, in terms of the discussed generalized coordinates, it will be easy to use point masses instead of distributed mass. Otherwise, complex terms will appear in the rotational part of the kinetic energy. As shown in Figure 1, four point masses are located instead of a distributed mass in the end-effector: a single point mass (m_1) at the

center of the mass and three identical masses (m_2) at the vertices of the end-effector. By considering the inertia specification of these systems, (m_1) and (m_2) can be determined as follows:

$$\begin{cases} m_1 + 3m_2 = m \\ 2m_2 b^2 = I_{xx} \end{cases} \Rightarrow m_1 = 9m_2 = \frac{3}{4}m. \tag{3}$$

The kinetic and potential energies of the robot are derived in terms of the discussed generalized coordinates. The general dynamic equations of motion can be obtained from Lagrangian formulation [10]:

$$\frac{d}{dt}\left(\frac{\partial K(q,\dot{q})}{\partial \dot{q}}\right) - \frac{\partial K(q,\dot{q})}{\partial q} + \frac{\partial V(q)}{\partial q} = Q_i + \sum_{j=1}^{3} \lambda J_{cij},$$

$$i = 1, 2, \cdots, 9. \tag{4}$$

It leads to;

$$[D(q)]_{9\times 9}\{\ddot{q}\}_{9\times 1} + \{G\}_{9\times 1} = \{Q\}_{9\times 1} - [p]^T_{9\times 6}\{T\}_{6\times 1} - [J_c]^T_{9\times 3}\{\lambda\}_{3\times 1}. \tag{5}$$

Equation 5 is written in the following form:

$$[D(q)]\{\ddot{q}\} + \{G\} - \{Q\} = -[J]\left\{\frac{T}{\lambda}\right\}, \tag{6}$$

where $[D(q)]$ is the inertia matrix, $\{G(q)\}$ is the gravity vector, $[J(q)]$ is the Jacobian matrix, $\{T\}$ is the vector of cables tension and $\{\lambda\}$ is the Lagrange multiplier vector.

Since the dynamic modeling of CDRs is concerned with relating the motion end-effector to the required active actuator torque, the forces in the cables are derived using the dynamic equations of the end-effector and actuators. In this paper, the dynamic behavior of the lumped actuators (each actuator includes a motor and pulley system) is also considered. The free body diagram for the ith actuator is shown in Figure 2.

The combined motor and cable pulley dynamics equations can be expressed as [11]:

$$\tau - \tau_{\text{evl}} = r.T, \tag{7}$$

$$\tau_{\text{evl}} = J_a.\ddot{\beta} + C_a.\dot{\beta}, \tag{8}$$

where r is the identical cable pulley radius for each actuator. The lumped actuator rotational inertias for each actuator and the viscous damping coefficients at each motor shaft are also included to provide a linear model for the system friction with:

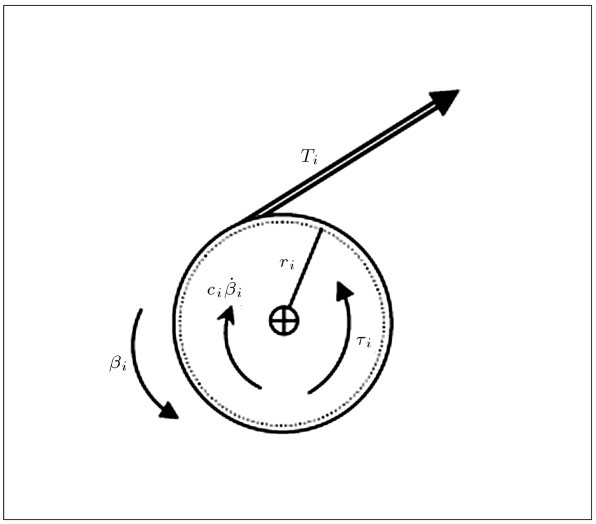

Figure 2. Free-body diagram for the ith actuator [11].

$$J_a = \text{diag}(J_1, \cdots, J_m), \tag{9}$$

$$C_a = \text{diag}(C_1, \cdots, C_m). \tag{10}$$

In this case study, the dynamic equation is given as follows:

$$\left\{\frac{\tau}{r\lambda}\right\} = [J^T].r.\left([D]\{\ddot{q}\} + \{G\} + \left(\frac{1}{r}\right)[p^T]\{J_a\ddot{\beta} + c_a\dot{\beta}\}\right). \tag{11}$$

LINEARIZATION OF THE STATE SPACE DYNAMIC EQUATION

In order to obtain the numerical solution of the nonlinear constrained trajectory optimization problem, in order to increase the load carrying capacity, dynamic Equation 11 can be rewritten as:

$$\ddot{q} = \left(r.[D(q)] + [p]^T\left\{J_a.\left(\frac{\partial\beta}{\partial q}\right)\right\}\right)^{-1}.\left(J\left\{\frac{\tau}{r\lambda}\right\} - [p]^T\left\{J_a\frac{d}{dt}\left(\frac{\partial\beta}{\partial q}\right)\dot{q} + C_a\dot{\beta}\right\} - r\{G\}\right) = f(\vec{q},\vec{\dot{q}},\tau,m_L). \tag{12}$$

By defining the state vector as $q = [x_1, x_2]^T$, where $x_1 = (q_1, q_2, \cdots, q_n)^T$ and $x_2 = (\dot{q}_1, \dot{q}_2, \cdots, \dot{q}_n)^T$, Equation 12 becomes:

$$\vec{\dot{q}} = \begin{bmatrix} \vec{\dot{x}_1} \\ \vec{\dot{x}_2} \end{bmatrix} = \begin{bmatrix} \vec{x_2} \\ f(\vec{X}(j), \vec{\tau}(j), m_L) \end{bmatrix}$$

$$= \vec{F}(\vec{X}(j), \vec{\tau}(j), m_L). \quad (13)$$

Equation 13 is the state space representation of the dynamic Equation 12, where $\vec{X}$ is a $2n \times 1$ vector and $f(X, \tau, m_L)$ consists of n nonlinear functions. The discretized form of the state space dynamic Equation 13 is:

$$\frac{\vec{X}(j+1) - \vec{X}(j)}{h} = \vec{F}(\vec{X}(j), \vec{\tau}(j), m_L), \quad (14)$$

where $h = \frac{(t_f - t_i)}{m}$, t_f, t_i are the start and end time of the robot motion and m is the number of set points used to discretize the end-effector trajectory. The nonlinear function, $f(X, \tau, m_L)$, at the $(k+1)$th trajectory, is expanded in a Taylor series about the (k)th trajectory. After neglecting the higher order (nonlinear) terms, the following equation is obtained:

$$\vec{X}(j+1) = [G_j]\vec{X}(j) + [H_j]\tau(j) + \vec{B}_j m_L + D_j, \quad (15)$$

where the matrices $[G_j]$, $[H_j]$ and β_j, D_j vectors are given in [13].

$X(j+1)$ can be written as a linear combination of the load m_L and the torque $\vec{\tau}(i)$, $i = 1, 2, 3, \cdots, j$. Equation 15 then becomes:

$$\vec{X}(j+1) = \vec{X}_h(j+1) + \vec{\beta}_j m_L + \sum_{i=1}^{j} [\alpha_{ji}]\tau(i),$$

$$j = 1, 2, \cdots, m. \quad (16)$$

This equation is the basic linearized dynamic equation, where:

$$\vec{X}_h(1) = \vec{X}(t_1), \quad (17)$$

$$\vec{X}_h(j+1) = [G_j]\vec{X}_h(j) + \vec{D}_j, \quad (18)$$

$$\vec{\beta}_1 = \vec{B}_1, \quad (19)$$

$$\vec{\beta}_1 = [G_j]\vec{\beta}_{j-1} + \vec{\beta}_1, \quad (20)$$

$$[a_{ji}] = [G_j][a_{j-1,i}],$$

$$\text{for} \quad i < j, \quad (21)$$

$$[a_{ji}] = [H_j],$$

$$\text{for} \quad i = j. \quad (22)$$

FORMULATION OF THE OPTIMIZATION PROBLEM

In this section, the complete problem formulation is presented for synthesizing dynamic robot motion with maximum load carrying capacities. The complete formulation of such a problem is given below:

1. The robot geometric parameters and the mass of the end-effector;
2. The initial and final states:

$$\vec{x}_1(t_i) = \vec{q}(t_i) = \vec{x}_{1i},$$

$$x_2(t_i) = \vec{o}, \quad (23)$$

$$\vec{x}_1(t_f) = \vec{q}(t_f) = \vec{x}_{1f},$$

$$x_2(t_f) = \vec{o}, \quad (24)$$

   where o is an $n \times 1$ null vector;
3. Total cycle time, $\nabla T = t_f - t_i$.

Constraints of problem:

1. The state space Equation 13 should be satisfied;
2. The joint actuator torques constraint:

$$\vec{\tau}_{\min}(\vec{X}(j)) \leq \vec{\tau}(j) \leq \vec{\tau}_{\max}(\vec{X}(j)), \quad (25)$$

   where $\vec{\tau}_{\min}(\vec{X}(j))$ and $\vec{\tau}_{\max}(\vec{X}(j))$ are in general nonlinear functions representing the torque-speed characteristics of the actuators;
3. The workspace constraint:

$$\vec{x}_1^- \leq \vec{x}_1(t) \leq \vec{x}_1^+. \quad (26)$$

   At the moment of moving of the end-effector, it is bounded to be within the limitation of the robot workspace. The generalized coordinate, assumed to be chosen from inside the 3D volume [10], is the workspace limitation. x_1^+ and x_1^- are the upper and lower bound of the generalized coordinates, respectively. The obtained optimal trajectory should not exceed this limitation, because this causes cables to lose their positive tensions;
4. The upper bound of payload m_L^+: That is the smaller value of the static load carrying capacity calculated at the two end positions.

The objective function is maximizing the dynamic load carrying capacities, m_L, along with finding the optimal trajectory, $x^*(t)$.

The trajectory synthesis formulation is a nonlinear optimization problem. It is written in a form slightly different from a general optimal control problem, where the objective function has an integral form. In the above formulation, the objective function consists of a single variable, m_L, which is not a function of time. This variable is also a single valued quantity

for the entire trajectory. It should be pointed out, however, that m_L is implicitly included in the nonlinear state space equation.

ITERATIVE LINEAR PROGRAMMING (ILP) FORMULATION

The ILP method uses linear programming to solve the linearized equations for each iteration. The linear programming problem can be formulated as follows.

At first, the final state reaching the condition from Equation 16 can be obtained as:

$$\vec{X}(m+1) = \vec{X}_h(m+1) + \vec{\beta}_m m_L + \sum_{i=1}^{m} [a_{mi}]\vec{\tau}(i) = X(t_f). \quad (27)$$

Equation 27 can be written as:

$$\vec{\beta}_m m_L + [E]\vec{\tau} = \vec{X}(t_f) - \vec{X}_h(m+1), \quad (28)$$

where:

$$[E] = \begin{bmatrix}[\alpha_{m1}] & [\alpha_{m2}] & \cdots & [\alpha_{mm}]\end{bmatrix} \in R^{(2n \times nm)}, \quad (29)$$

$$\vec{\tau} = \begin{bmatrix}\vec{\tau}(1), \vec{\tau}(2), \cdots, \vec{\tau}(m)\end{bmatrix} \in R^{(nm \times 1)}. \quad (30)$$

It should be noted that $[E]$ and $X_h(m+1)$ in the above equation are computed based on the values of the state and control variables of the previous iteration. Since $X(t_f)$ is also given, the only unknowns in Equation 28 are m_L and $\vec{\tau}$ vectors. In order to facilitate the LP solution, Equation 28 can be written by two sets of inequalities.

$$\vec{\beta}_m m_L + [E]\vec{\tau}_m - \vec{e} \leq \vec{X}(t_f) - \vec{X}_h(m+1), \quad (31)$$

$$\vec{\beta}_m m_L + [E]\vec{\tau}_m + \vec{e} \geq \vec{X}(t_f) - \vec{X}_h(m+1), \quad (32)$$

where $\vec{e} = [e_{\text{pos}1}, e_{\text{pos}2}, \cdots, e_{\text{vel}1}, e_{\text{vel}2}, \cdots]^T$ is a $2n$ vector, and the first n elements (e_{pos}) represent the final position error tolerances, and the last n elements (e_{vel}) represent the velocity error tolerances. This modification introduces two more variables ($e_{\text{pos}}, e_{\text{vel}}$) and $2n$ inequality constraints. If actuators are permanent magnet DC motors, the torque-speed characteristic function, $\vec{\tau}_{\min}(\vec{X}(j))$ and $\vec{\tau}_{\max}(\vec{X}(j))$, can be approximated by the following equations:

$$\vec{\tau}(j) \leq \vec{\tau}_{\max}(\vec{X}(j)) = \vec{K}_1 - [K_2]\dot{\beta}(j), \quad j = 1, 2, \cdots, m, \quad (33)$$

$$\vec{\tau}(j) \geq \vec{\tau}_{\min}(\vec{X}(j)) = -\vec{K}_1 - [K_2]\dot{\beta}(j), \quad j = 1, 2, \cdots, m, \quad (34)$$

where $k_1 = \tau_{\text{stall}}$ is an $n \times 1$ constant vector, $k_2 = \frac{\tau_{stall}}{\omega_0}$ is an $n \times n$ diagonal constant matrix obtained from the equivalent motor constants and ω_0 is the maximum no-load speed of the actuator. The "max" or "min" in the superscripts indicates whether the actuator is saturated at its upper or lower bound. Note that both stall torques and the maximum no-load speed of the actuators must be specified in order to determine an actuator torque bound.

The main drawback of CDRs is the unilateral actuation imposed by the cables. Since cables can only apply tensile force, this limitation can cause excessive deviation from the prescribed trajectory, even if the joint torque constraint is not violated. A more successful approach should maximize the load-carrying capacity and allowable cable tension bound attained for the end-effector trajectory. The desired trajectory is characterized as the set of points that the centroid of the end-effector can reach with tensions in all suspension cables. The following assumption is used for the points on the trajectory:

1. The maximum tension is considered for all cables.
2. The cables must be capable of exerting a positive wrench on the platform. All cable tensions must be non-negative to equilibrate the end effector for an applied force.
3. All active cables must remain in tension to be effective for the equilibrium or dynamic motions. The feasible points are specified by imposing the following inequalities [12]:

$$0 \leq T_i \leq T_{\max}, \qquad i = 1, ..., m. \quad (35)$$

In a pseudostatic condition, the tension in the cables is equal or greater than $\vec{\tau}_{\min}(\vec{X}(j))/r$. When the dynamic effects are considered, one or more cable can become slack, despite a positive $\vec{\tau}_{\min}(\vec{X}(j))$, and online cable tension estimation is needed. Referring to the dynamic term, Equation 7 clarified that higher minimum torques are needed. If the bias term, τ_{vel}, is positive, all cable force components are forced to be zero at the minimum. It can be shown that, for each actuator, $\vec{\tau}_{\min}(\vec{X}(j)) = \tau_{\text{vel}}$. If τ_{vel} were negative, the minimum torque required to ensure that the corresponding cable was in tension could be negative for

one or more actuators, since all torque actuators were forced to be positive. Therefore, constraints are needed for the online actuator torques control, which can be written as follows:

$$\vec{\tau}_{\min}(\vec{X}(j)) = \max\{\tau_{\text{vel}}, 0\}. \tag{36}$$

Writing these constraints in matrix form leads to:

$$\vec{\tau} \le \vec{b}_u = \begin{vmatrix} \vec{K}_1 - [K_2]\vec{\beta}^k(1) \\ \vec{K}_1 - [K_2]\vec{\beta}^k(2) \\ \vdots \\ \vec{K}_1 - [K_2]\vec{\beta}^k(m) \end{vmatrix}, \tag{37}$$

$$-\vec{\tau} \le \vec{b}_l = \max\{\tau_{\text{vel}}, 0\}, \tag{38}$$

where $\vec{b}_l$ and $\vec{b}_u$ are, respectively, the lower and upper bound vectors of the actuators. By a change of variables, as below, the problem can be converted to a standard linear programming problem:

$$\vec{Y} = \vec{b}_u - \vec{\tau}, \quad \text{or} \quad \vec{\tau} = \vec{b}_u - \vec{Y}, \qquad \vec{Y} \ge 0. \tag{39}$$

Substituting Equation 39 into Equation 37 leads to:

$$\vec{Y} \le \vec{b}_u - \vec{b}_l. \tag{40}$$

Using Equation 16, the workspace constraint is given by Inequality 26 written as:

$$\begin{aligned} \vec{x}_1^{\,-} - \vec{X}_{1h}(j+1) &\le \vec{\beta}_{1j} m_L + \sum_{i=1}^{j} [a_{1ji}]\vec{\tau}(i) \\ &\le \vec{x}_1^{\,+} - \vec{X}_{1h}(j+1), \\ & j = 1, 2, \cdots, m, \end{aligned} \tag{41}$$

where β_{1j}, $X_{1h}(j+1)$ are the upper $n \times 1$ vectors of β_j and $X_h(j+1)$, respectively, and $[\alpha_{1ji}]$ is the upper $n \times n$ sub-matrix of $[\alpha_{ji}]$. Equation 41 can be written in the following form by letting $[A_j] = [\alpha_{1j1}, \alpha_{1j2}, \cdots, \alpha_{1jj}, 0, 0, \cdots, 0]$ and rearranging terms which can be written as follows:

$$\begin{aligned} &\vec{\beta}_{1j} m_L - [A_j]\vec{Y} \le ([\vec{x}_1^{\,+} - \vec{X}_{1h}(j+1)] - [A_j]\vec{b}_u), \\ &j = 1, 2, \cdots, m, \end{aligned} \tag{42}$$

$$\begin{aligned} &-\vec{\beta}_{1j} m_L + [A_j]\vec{Y} \le ([\vec{X}_{1h}(j+1) - \vec{x}_1^{\,-}] + [A_j]\vec{b}_u), \\ &j = 1, 2, \cdots, m, \end{aligned} \tag{43}$$

where $[A_j]$ is a $n \times nm$ matrix.

As mentioned above, the upper bound of load m_L^+ is determined from SLCC (Static Load Carrying Capacity) at the two end points. Since the robot must completely stop at the two end points, the maximum DLCC for the trajectory cannot be greater than the SLCC at either one of the end positions. This means that:

$$m_L^+ = \min\{m_{SLi}, m_{SLf}\}, \tag{44}$$

and:

$$m_L \le m_L^+, \tag{45}$$

where m_{SLi} and m_{SLf} are the SLCC at the initial and final positions, respectively. Combining all the constraints and writing the result in a matrix form gives:

$$\begin{bmatrix} 1 & 0 & 0 \\ [0] & [1]_{(nm\times nm)} & [0] \\ \pm\beta_{1j_{(nm\times 1)}} & \mp[A_j]_{(nm\times nm)} & [0] \\ \beta_{m_{(2n\times 1)}} & -[E]_{(2n\times nm)} & -[I]_{(2n\times 2n)} \\ -\beta_{m_{(2n\times 1)}} & [E]_{(2n\times nm)} & [I]_{(2n\times 2n)} \end{bmatrix} \times \begin{Bmatrix} m_L \\ \vec{Y}_{(nm\times 1)} \\ \vec{e}_{(2n\times 1)} \end{Bmatrix} \le \begin{Bmatrix} m_L^+ \\ \vec{Y}_{\lim it(nm\times 1)} \\ \vec{X}_{\lim it(nm\times 1)} \\ \vec{X}_f^+ \\ \vec{X}_f^- \end{Bmatrix}. \tag{46}$$

And the dimensions of the matrices and vectors are as follows:

$$\begin{aligned} &[(2nm + 4n + 1) \times (2n + nm + 1)] \\ &\quad \times \{(2n + nm + 1) \times 1\} \\ &\quad \le \{(2nm + 4n + 1) \times 1\}, \end{aligned} \tag{47}$$

where β_{1j}, $[A_j]$ and $X_{\lim it}$ can be obtained from Inequalities 42 and 43 using the method described above, and:

$$\vec{Y}_{\lim it} = \vec{b}_u - \vec{b}_l,$$

$$\vec{X}_f^+ = [X(t_f) - X_h(m+1)] - [E]b_u,$$

$$\vec{X}_f^- = [X_h(m+1) - X(t_f)] + [E]b_u.$$

The objective function of this LP problem is defined as:

$$Z = \max\{C^T V\}, \tag{48}$$

where $C = [1, 0, 0, \cdots, -W_{\text{pos}}, -W_{\text{vel}}]$ with $W_{\text{pos}}, W_{\text{vel}} > 0$ (weighting factors) and $\vec{V} = [m_L, \vec{Y}, \vec{e}_{\text{pos}}, \vec{e}_{\text{vel}}]$ with $\vec{Y} \ge 0$.

The load carrying capacity, m_L, can be maximized by this objective function, and simultaneously the position and velocity errors at the end points of the trajectory are minimized. Since the objective function (Equation 48) and the constraints equation (Equation 46) are both linear, we have a standard linear programming problem.

COMPUTATIONAL ALGORITHM

The computing method for the optimal trajectory problem is formulated, as shown in Figure 3. First, initialize the trajectory. This step involves guessing an initial control and state variable trajectory. A good initial guess may be obtained by using polynomial trajectories to connect the two end points. By discretizing the initial trajectory into m set points, the corresponding linearized constraint coefficients are computed and, then, an iterative linear programming subroutine is invoked to update m_L, $\vec{Y}$ and $\vec{e}$. Using these updated variables, the new trajectory, $X^{k+1}(j)$, is synthesized. Then, the termination conditions are checked:

$$\max\{e_{\text{pos}}, e_{\text{vel}}\} \leq \varepsilon_e, \tag{49}$$

$$\max\{|\vec{X}^{k+1}(j) - \vec{X}^{k}(j)|, j = 2, \cdots, m\} \leq \varepsilon_x, \tag{50}$$

$$\left|m_L^{k+1} - m_L^k\right| \leq \varepsilon_m, \tag{51}$$

where $\varepsilon_e, \varepsilon_x$ and ε_m are predefined small positive constants. If the termination conditions are satisfied, then, the updated trajectory is optimal, and the corresponding value of m_L is the maximum allowable load that can be carried by the cable driven robot. Otherwise, the program jumps to step 2 in ILP method flowchart shown in Figure 3. Also, satisfying the termination criterion means that linearization errors are eliminated (or significantly reduced) when the ILP method converges to the optimal solution.

In general, because of the discretization (truncation) error of the difference equation, the continuous state space equation will be satisfied only if the time interval is "sufficiently small".

SIMULATION

A simulation study is presented to investigate the application and efficiency of the proposed algorithm. A cable driven robot with 6 d.o.f is considered, as shown in Figure 1 [13].

For simulation, a cable driven robot must carry a load from an initial point with coordinate $X0 = [-0.1, -0.1, 1.5, 0, 0, 0]$ to the final point in the robot workspace with coordinate $XF = [0.1, 0.1, 1.8, 0, 0, 0]$, during overall time $t_f = 1$ s. The velocities and accelerations at the start and end points are zero. The path parameter unknowns are determined, so that the initial and final conditions are satisfied. The trajectory used for an initial guess in simulation is a polynomial of the fifth order. In this simulation, it is assumed that there is no exerted external wrench on the robot.

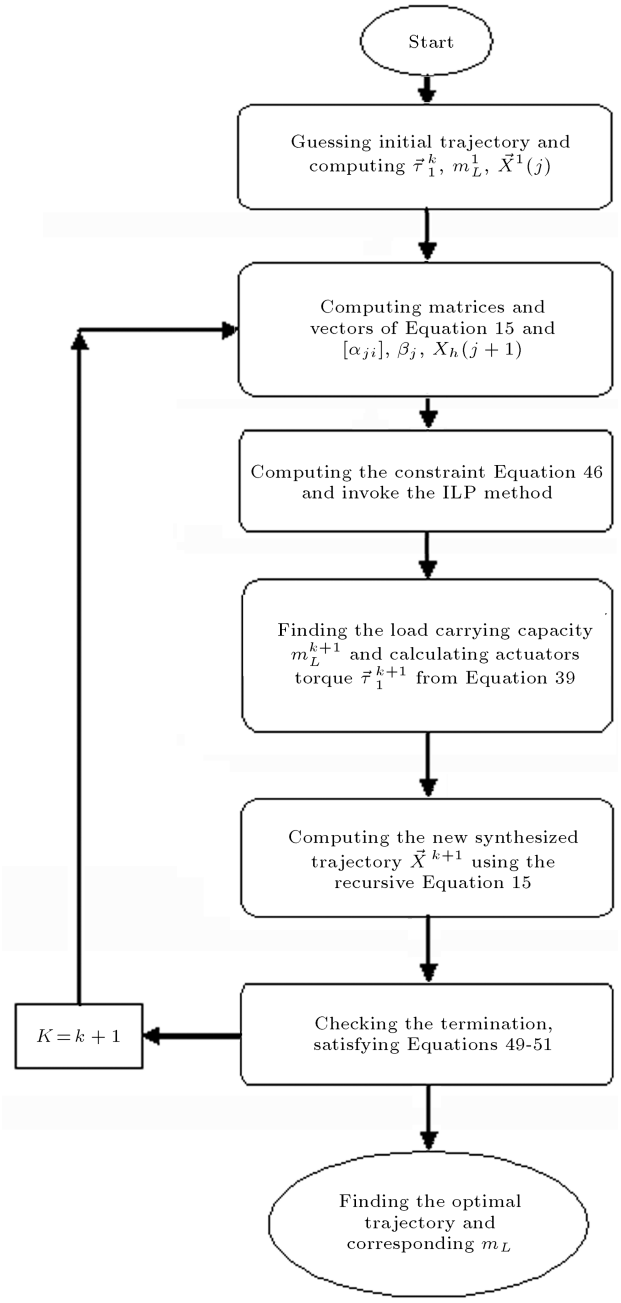

Figure 3. ILP method flowchart for computing optimal trajectory.

Geometrical and mechanical characteristics used in the simulation are listed in Table 1 and the other used parameters relative to the robot actuators, consisting of motors and pulleys, are given in Table 2.

By discretizing the trajectory to $m = 20$ set points, the procedure for synthesizing the optimum trajectory converged after eleven iterations, and the maximum allowable carrying load without violating either of the constraints is $m_L = 75.52$ kg.

Figure 4 gives the trajectory of the center of mass of the end-effector for the optimal trajectory and initial

Table 1. Geometrical parameters of robot.

Parameter	Value	Units
End-effector mass	$m = 15$	kg
Half of the side of the upper triangle (fixed base)	$a = 0.3\sqrt{3}$	m
Half of the side of the lower triangle (end-effector)	$b = (1/2)a$	m
Moment of inertia (end-effector)	$I_{xx} = I_{yy} = m\frac{b^2}{6}$ $I_{zz} = 2I_{xx}$ $I_{xy} = I_{zx} = I_{yz} = 0$	kg.m^2

Table 2. Simulation parameter for actuators.

Parameter	Value	Units
Pulley radius	$r = 5 \times 10^{-2}$	m
Motor shaft viscous damping coefficient	$C = 0.01$	Nms
Lumped actuator rotational inertia	$J = 8 \times 10^{-4}$	kg.m^2
Stall torque	$\tau_{\text{stall}} = 10$	Nm
Maximum no-load speed	$\omega_0 = 1910$	Rpm

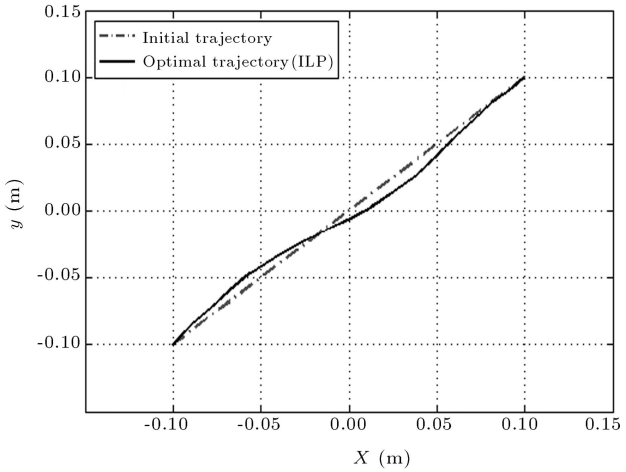

Figure 4. Motion of center of mass of end-effector on $x - y$ coordinates.

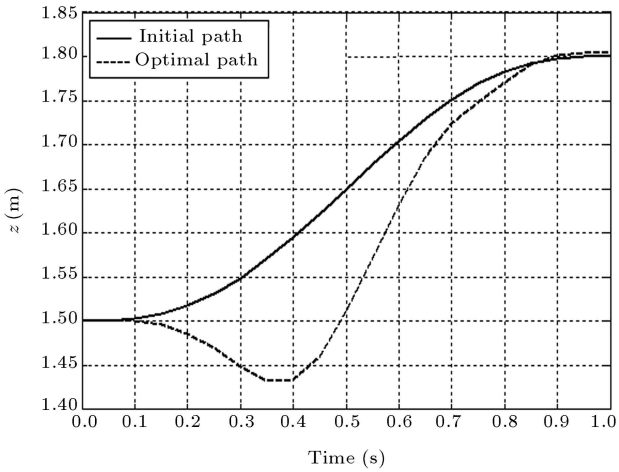

Figure 5. Motion of center of mass of end-effector on z-coordinates.

guess in the xy plane. Figure 5 shows the variation of the z coordinate in time. The initial and final optimal positions and velocities of the vertices of the end-effector that are used as generalized coordinates in the dynamic equations can be depicted. One of them is shown in Figure 6. Figure 7 gives the linear programming solution of m_L at each iteration. The corresponding initial path and the final optimal path in Cartesian 3D space are given in Figure 8. Figure 9 presents the optimal torques related to the optimal trajectory with regard to the upper and lower bounds of the actuator torque.

As mentioned above, in the linearization process of the ILP method, the higher order (nonlinear) terms in the Taylor series are neglected and because CDRs have nonlinear dynamics, errors as a result of neglecting higher order terms will exist in the solution process. Tensions of cables are computed for different masses from an initial guess until optimal trajectory during the process of obtaining the optimal trajectory for ensuring robot stability. Also, as mentioned above, Equation 38 causes the actuators torque to be a non-negative value. Regarding the optimal trajectory and corresponding actuators torque variations, it is realized that, while the end effector moves along the optimal trajectory, the coupling inertia of the load is minimized. Simultaneously, at most parts of the trajectory, actuators work close to their maximum torque capacity, and the actuators torques have a non-negative value in total operating time, resulting in an increase in the conveyable load of the CDR.

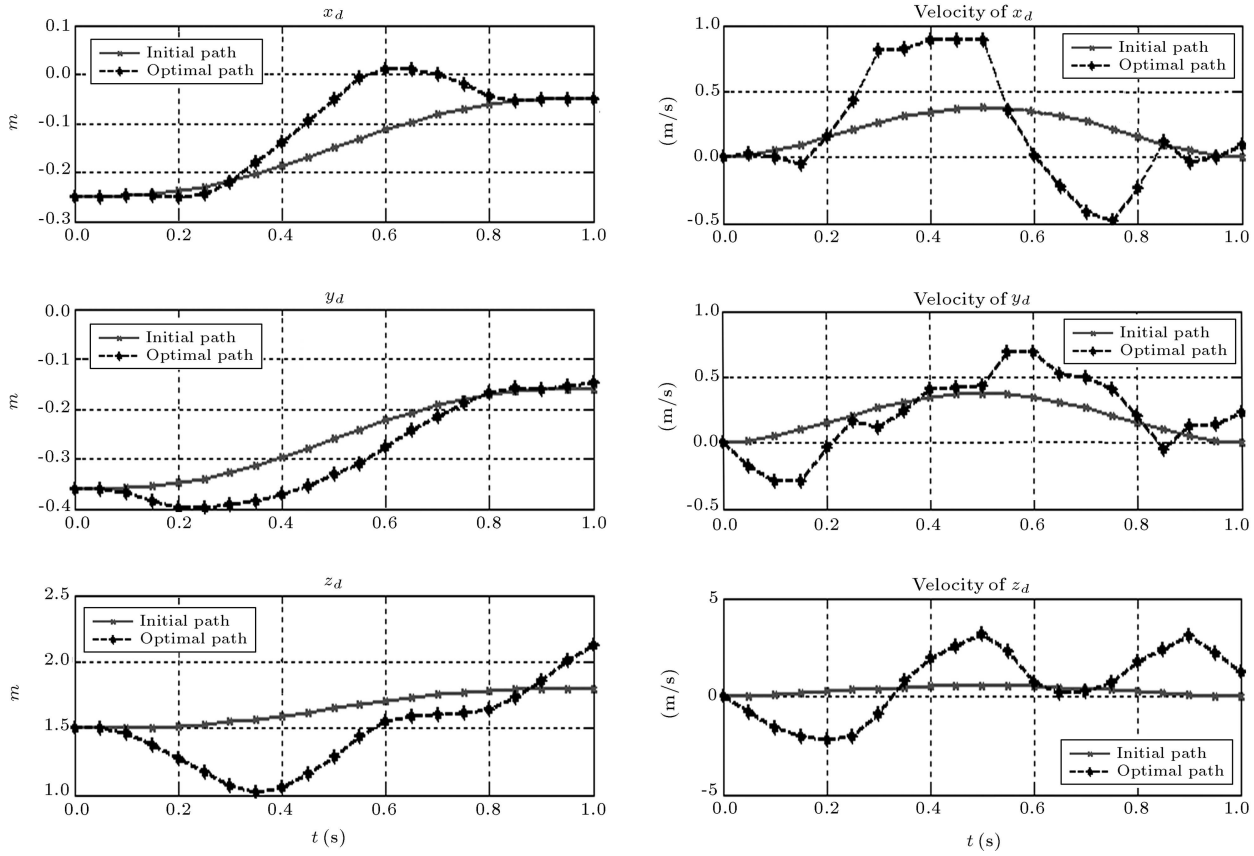

Figure 6. Position and velocity of vertex D.

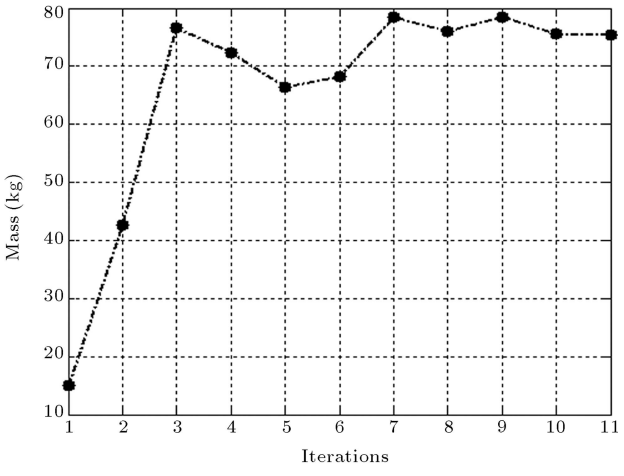

Figure 7. Optimal maximum dynamic payload in each iteration.

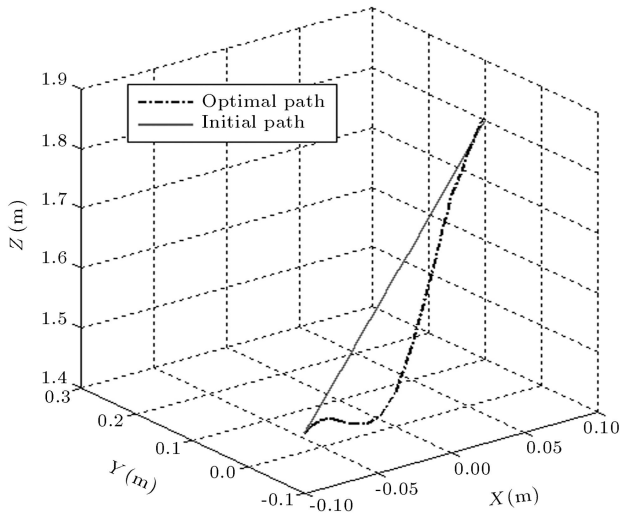

Figure 8. Initial and optimal trajectory in 3D view.

CONCLUSION

In this paper, a computational algorithm is developed to obtain a numerical solution to the optimization problem associated with synthesizing optimal trajectories in CDRs. This was achieved by torque capacity constraints in addition to considering the workspace constraint of a robot. A simulation case study of CDR was presented to investigate the efficiency of the algorithm. It is seen that, while the CDR moves along the optimal trajectory, the coupling inertia of the mass is minimized without violating either of the constraints. Moreover, during the motion, actuators are working with full or near to full capacity. As understood from the simulation, the load carrying capacity is increased from an initial value 15 kg to $m_L = 75.52$ kg.

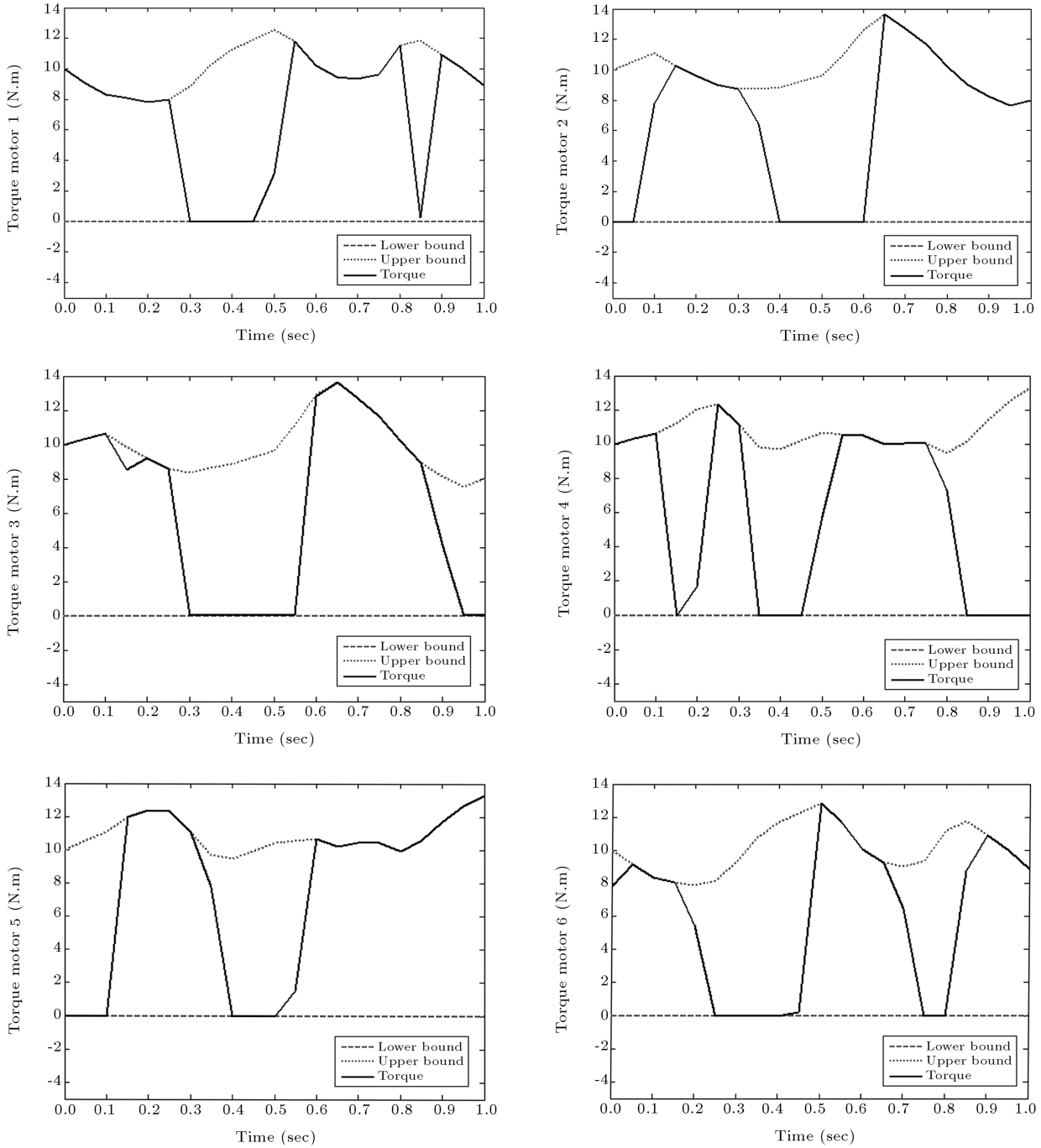

Figure 9. Optimal torques of six motors related to optimal trajectory.

ACKNOWLEDGMENT

The authors gratefully acknowledge the support of the School of Mechanical Engineering at the Iran University of Science & Technology. Support for this research is also provided by the IUST Research Program.

REFERENCES

1. Albus, J., Bostelman, R. and Dagalakis, N. "The NIST robocrane", *J. Robot Syst.*, **10**, pp. 709-724 (1993).

2. Alp, A. and Agrawal, S. "Cable suspended robots: design, planning, and control", *Proceedings of IEEE Int. Conf. on Robotics and Automation*, Washington, pp. 4275-4280 (2002).

3. Wang, L.T. and Ravani, B. "Dynamic load carrying capacity of mechanical manipulators: II. Computational procedure and applications", *J. Dyn. Syst. Meas. Control*, **110**, pp. 53-61 (1988).

4. Korayem, M.H. and Ghariblu, H. "Maximum allowable load of mobile manipulators for two given end points

of end effector", *Int. J. Adv. Manuf. Technol.*, **24**, pp. 743-751 (2004).

5. Korayem, M.H. and Nikoobin, A. "Formulation and numerical solution of robot manipulators in point-to-point motion with maximum load carrying capacity", *Scientia Iranica. Trans. B*, **16**(1), pp. 28-38 (2009).
6. Korayem, M.H. and Bamdad, M. "Dynamic load-carrying capacity of cable-suspended parallel manipulator", *Int. J. Adv Manuf. Technol.*, **44**(7-8), pp. 829-840 (2009).
7. Yue, S., Tso, S.K. and Xu, W.L. "Maximum dynamic payload trajectory for flexible robot manipulators with kinematic redundancy", *Mech. Mach. Theory*, **36**, pp. 785-800 (2001).
8. Korayem, M.H., Haghpanahi, M. and Heidari H.R. "Maximum allowable dynamic load of flexible manipulators undergoing large deformation", *Scientia Iranica, Trans. B*, **17**(1), pp. 61-74 (2010).
9. Korayem, M.H. and Shokri, M. "Maximum dynamic load carrying capacity of a 6UPS-Stewart platform manipulator", *Scientia Iranica*, **15**(1), pp. 131-143 (2008).
10. Afshari, A. and Meghdari, A. "New Jacobian matrix and equations of motion for a 6 d.o.f cable-driven robot", *Int. J. of Advanced Robotic Systems*, **4**(1), pp. 63-68 (2007).
11. Gallina, P., Rossi, A. and Williams, R.L. "Planar cable-direct driven robots: dynamic and control", *Proceedings of the ETC2001/ASME Design Engineering Technical Conference*, Pittsburgh, PA, USA (2001).
12. Fattah, A. and Agrawal, S.K. "On the design of cable-suspended planar parallel robots", *J. of Mech. Des.*, **127**, pp. 1021-1028 (2005).
13. Najafi, Kh. "Dynamic analysis of cable driven robots in the presence of the flexibility and finding dynamic load carry capacity on optimal trajectory", MSc thesis, College of Mechanical Engineering, Islamic Azad University, Science and Research Campus, Tehran (2009).

BIOGRAPHIES

Moharam Habibnejad Korayem was born in Tehran, Iran on April 21, 1961. He received his BS (Hon) and MS in Mechanical Engineering from the Amirkabir University of Technology in 1985 and 1987, respectively, and his PhD degree in Mechanical Engineering from the University of Wollongong, Australia, in 1994. He is a Professor in Mechanical Engineering at the Iran University of Science and Technology and has been involved with teaching and research activities in robotics at the Iran University of Science and Technology for the last 15 years. His research interests include dynamics of elastic mechanical manipulators, trajectory optimization, symbolic modeling, robotic multimedia software, mobile robots, industrial robotics standards, robot vision, soccer robot, and analysis of mechanical manipulators with maximum load carrying capacity. He has published and presented more than 300 papers in international journals and conferences in the field of robotics.

Khaled Najafi was born in Meyaneh, Iran on September 10, 1983. He received his BS and MS (Hon) in Mechanical Engineering from the Islamic Azad University of Tehran in 2006 and 2009, respectively. His research interests include robotic systems, linear programming, parallel manipulators, industrial automation and mechatronic systems.

Mehdi Bamdad was born in Sabzevar, Iran on June 13, 1982. He received his BS in Mechanical Engineering from KNT University of Technology in 2004 and a MS from Semnan University in 2006. He is a PhD candidate in Mechanical Engineering at the Iran University of Science and Technology. His research interests include robotic systems, optimization, parallel manipulators, industrial automation and mechatronic systems.