Original scientific paper

MINIMUM-TIME PATH PLANNING FOR ROBOT MANIPULATORS USING PATH PARAMETER OPTIMIZATION WITH EXTERNAL FORCE AND FRICTIONS

Habib GHANBARPOUR ASL

Abstract: This paper presents a new minimum-time trajectory planning method which consists of a desired path in the Cartesian space to a manipulator under external forces subject to the input voltage of the actuators. Firstly, the path is parametrized with an unknown parameter called a path parameter. This parameter is considered a function of time and an unknown parameter vector for optimization. Secondly, the optimization problem is converted into a regular parameter optimization problem, subject to the equations of motion and limitations in angular velocity, angular acceleration, angular jerk, input torques of actuators', input voltage and final time, respectively. In the presented algorithm, the final time of the task is divided into known partitions, and the final time is an additional unknown variable in the optimization problem. The algorithm attempts to minimize the final time by optimizing the path parameter, thus it is parametrized as a polynomial of time with some unknown parameters. The algorithm can have a smooth input voltage in an allowable range; then all motion parameters and the jerk will remain smooth. Finally, the simulation study shows that the presented approach is efficient in the trajectory planning for a manipulator that wants to follow a Cartesian path. In simulations, the constraints are respected, and all motion variables and path parameters remain smooth.

Keywords: constrained optimization; dynamic constraints; minimum-time robot path planning; path parameter optimization; trajectory tracking

1 INTRODUCTION

In the field of robotics, there is a need for the tracking of a known path for robot manipulators in the Cartesian space with maximum velocity or minimum-time algorithms for their economic benefits. Minimum-time path planning problems for robotic manipulators have widely been studied in the past, especially for industrial applications. Many techniques have been proposed in the past for this problem; however, due to factors such as non-linearity, coupling dynamics, and torque limitation, the task of finding a minimum-time path for robot manipulators is quite complicated. Therefore, an efficient motion planning technique for robot manipulators to move along a pre-defined trajectory is required. The most notable earliest studies in the domain of minimum-time path planning along a specified path were [1-3]. Additionally, ref. [4] is also included in one of the earliest studies in this domain, which presents an analysis of time-optimal trajectories in the case of fixed initial and final positions. In another study of the same era, the authors present a method to compute robotic paths in cases of closed kinematic chain mechanisms [5]. The earliest works in this domain also included solutions to the problem of minimum-time robotic manipulator motion along a specified geometric path, considering constraints such as force and torque [6]. That was the first attempt of its kind to address the minimum-time problem involving constraints. The minimum-time trajectory planning was similarly discussed in [7, 8]. Bobrow took the research further and devised a technique to find a collision-free path to obtain a minimum-time motion of a robotic manipulator [9]. Bobrow used B-spline polynomials and nonlinear equations of motion to produce optimal trajectories in the Cartesian space of the manipulator. These early studies paved the way for progress in robotics research and gave rise to further advancements in this domain. Bobrow's uses of B-spline polynomials to produce optimal trajectories were further investigated by researchers [10], most recently in [11]. Furthermore, minimization of the spline curve path was studied in [12-14]. More notable studies which contributed to further advancements in the research include techniques for making the robot manipulator learn from the previous path devised in [15], where waypoints are determined with the help of a robot manipulator tracking the path. This procedure of specifying a set of waypoints for path planning for robot manipulators is further elaborated in [16, 17], where previous path velocities are estimated for trajectory generation, and several continuous trajectories are considered between two points in a path so that they are blended. That is also referred to as point-to-point trajectory motion in other studies, such as in [18, 19], where a method for generating a smooth, time-optimal trajectory is presented. In [19], the authors used the third derivative of the path parameter with respect to time as an input, and it limits the torque rate in order to achieve the smoothness of the path. The point-to-point trajectory motion was investigated in [20], where the authors generated trajectories in joint space for the point-to-point motion. In this study, the authors also considered the constraints of the actuators' velocity, acceleration and jerk limits while calculating the minimum-time path [21, 22]. In more recent studies, the type of path that a robot manipulator needs to follow between given waypoints, such as a straight-line path, cubic spline or a circular path, are included in the path planning techniques. This consideration has been taken into account in [22]. In addition to considering the path type, the point-to-point motion planning also includes the polynomial coefficients of the constraints such as the position, velocity, acceleration and jerk constraints, as presented in [23-26]. Moreover, recent studies, namely [27, 28], focus on the user-defined trajectories for path planning, as well as on the development of a commercial robotics software. Many previous

researchers have also focused on iterative and geometric methods as well as on optimal switching structures in accordance with Pontryagin's Maximum Principle, as mentioned in [8, 29]. Other approaches have applied B-spline cubical methods for trajectory planning as mentioned in [30-32]. The optimization of the point-to-point minimum-time path problem has also been analyzed using the Sequential Quadratic Programming (SQP) method, as mentioned in [33]. For the purpose of optimization, the above-mentioned method involves the determination of minimum transmission time with electromagnetic constraints such as kinematics. Building up to that, [34] uses a similar approach with a dynamic programming algorithm in order to solve the minimum-time optimization problem. The dynamic programming approach has also been applied in [35]. After the early advancements in this domain, an ample amount of research has been undertaken on the constrained motion of robots. For example, [36, 37] propose a method to find the minimum-time considering the constraints of a jerk or higher order derivatives of position. Other studies presenting an approach for minimum-time calculation subject to kinematic constraints can be found in [38-40]. Another approach for a robot arm with path planning is the connection of straight-lines with circular arcs, perturbations about a straight-line with a Fourier series and cubic Bezier splines, as suggested in [41]. Furthermore, solutions to the path planning problem with end-effector constraints for robotic manipulators have also been studied, as presented in [42]. Another approach to path planning with torque constraints includes either the bang-bang trajectory or the bang-singular-bang trajectory as presented in [43]. Other interesting approaches for solving the path planning problem include a continuous genetic algorithm for path generation of robotic manipulators in a Cartesian space as presented in [44]. More recently, the robotic methodologies of a point-to-point trajectory planning have also been applied in other applications, as mentioned in [45]. Other examples using trajectory planning include agricultural field machines [46] and trajectory generation for animal movement [47]. Most significant studies on robotic manipulators include those, which estimate the minimum path and generate trajectories while handling any kinematic constraints on the velocity and acceleration [48-50]. This study proposes a path planning mechanism, whereby trajectories are generated in the operational space subject to certain dynamic constraints. The studies, which have been hitherto mentioned, do not usually consider the factors of external force and friction while calculating the minimum path for robotic manipulators. However, there is some significant research focusing on external forces as well as friction as factors influencing the calculation of the minimum path for a robotic manipulator. For example, [51, 52] present studies on robotic manipulators being controlled by DC motors, and considering the forces and friction generated by the motors that are dependent upon the kinematic constraints of velocity and acceleration.

A few of these studies have presented an approach for trajectory tracking control with taking into account the actuator dynamics [53]. However, most recent studies have found it necessary to include actuator dynamics and constraints as a part of the trajectory tracking control. For example, [54] proposed a trajectory tracking mechanism where geometrical constraints, impulsive force constraints, torque constraints, maximum acceleration and velocity constraints are considered. Another interesting study in this regard [55] compares different frameworks of trajectory tracking controllers for unmanned vehicles, where nonlinearity of the dynamics, uncertainties, noise, disturbances and several constraints are considered. The actuating machines usually used for robotic manipulators are DC motors. Hence, the actuator dynamics mostly consist of the force, energy and friction required for the DC motors which are attached to the manipulators [56]. However, there is some significant research focusing on external forces as well as friction as factors influencing the calculation of the minimum path for a robotic manipulator.

In this paper, the minimum-time path planning problem for a parametric approach to the solution of path planning in robotic manipulators uses a manipulator's path description, dynamic relations, and other defined constraints. Thus, an optimization strategy has been devised for path planning considering the mentioned constraints. The paper is presented in the following manner: Section 2 describes and illustrates the dynamics of the robotic manipulator. The problem is formulated and presented in Section 3, and the approach to solving the problem of optimization of a robotic manipulator is presented in Section 4. Afterward, the results of the simulation and path planning are presented and discussed in Section 5. Finally, a summary and conclusion of the paper are presented in Section 6.

2   DYNAMIC MODELLING OF THE MANIPULATOR

If we suppose that the manipulator's joint angles are represented by a vector q, the joint angle velocity vector will be $\dot{q}$, and $\ddot{q}$ will be the angular acceleration vector. The dynamic equation of the motion of the robot manipulator can be written as [49, 50]:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = T \qquad (1.1)$$

The total torque T is then given by:

$$T = T_{ext} + T_c + T_f \qquad (1.2)$$

where $M(q) \in R^{n \times n}$ is the inertia matrix and $C(q,\dot{q}) \in R^{n \times n}$ is a matrix that contains the information of centrifugal and Coriolis torques. Here, C is not a unique matrix, but $C(q,\dot{q})\dot{q}$ is a unique vector. $G(q) \in R^{n \times n}$ is the gravity torque, T is the vector of total torques, and n is the number of joints angles. T_{ext}, T_c, and T_f are the external force, control and friction torques, respectively. The external torque T_{ext} can be modelled as [49]:

$$T_{ext} = J^T F_{ext} \qquad (2)$$

where F_{ext} is the external force at the robot's end-effector, and J^T is the transpose of the Jacobian matrix. We consider

that the external force is generated by friction between the end-effector and task plane and its direction is along the opposite direction of the end-effector's velocity. We can consider a model such as:

$$F_{ext} = -\mu v f_n \hat{n}_v \qquad (3)$$

where v is the norm of the velocity vector of the end-effector, f_n is a normal force which is perpendicular to the surface which contains the velocity vector and the end-effector's link, μ is the friction coefficient, and $\hat{n}_v$ is a unit vector along the velocity vector tangent to the path. The joint angle friction torque is thus modeled as [49, 50]:

$$T_f = K_c sign(\dot{q}) + K_v \dot{q} \qquad (4)$$

where K_c is a diagonal matrix which contains the coulomb friction coefficients for any joint, and K_v is another diagonal matrix that contains the viscous friction coefficients of the joint angles. There are a number of constraints due to the limitations of the robot manipulator. These constraints can be divided into two groups, the first of which is related to kinematics constraints, such as the constraints in joint angles, angular velocity, angular acceleration, jerk and higher time derivatives of the joint angles. We can consider limitations for the higher time derivatives of the joint angles. The second group is related to the constraints on the actuators. Many researchers consider $\dot{q}$ and $\ddot{q}$ to have constraints, but when the actuator is able to generate more torque, it will be able to generate higher jerk and acceleration, so the limitations on the time derivatives of the joint angles come from the actuators. Consequently, the added limitations on the time derivatives of the joint angles will cause more limitation on the robot, which is why we will not use the actual potential of the robot. Here, we will consider the transmission of rotation between the actuators and the arm of the robot as being guaranteed by the mechanical transmission system of the gears. Although this mechanism reduces the angular velocity of the motor, it increases the generated torque of the motor, thus [49]:

$$T_c = NT_m, N = diag([N_1, ..., N_n]) \qquad (5)$$
$$\dot{q} = N^{-1}\dot{q}_m$$

where N is the diagonal matrix of the transmission gear system, T_m the vector of motor's torque, and $\dot{q}_m$ is the speed of the motor. When the joint angle motors are DC motors, and using Kirchhoff's voltage law for armature windings as represented in Fig. 1, the equations of the DC motor become [52]:

$$\dot{I} = L^{-1}\left(-RI - K_{bemf}\dot{q}_m + U\right) \qquad (6)$$
$$T_m = K_m I$$

where $L = diag([L_1, L_2, ..., L_n])$ is a diagonal matrix which contains the element of motor inductances and $R = diag([R_1, R_2, ..., R_n])$ is a matrix which contains armature resistances, $K_{bemf} = diag([K_{bemf_1}, K_{bemf_2}, ..., K_{bemf_n}])$ is

the back electromotive force constant matrix, $U = (U_1, U_2, ..., U_n)$ is the input voltage vector, $I = (I_1, I_2, ..., I_n)$ is the armature current of each DC motor, and $K_m = diag([K_1, K_2, ..., K_n])$ is the motor torque constant matrix. By substituting Eq. (6) into Eq. (5), we find that:

$$\dot{T}_c = AT_c + B\dot{q} + DU$$
$$A = diag\left(\left[-\frac{R_1}{L_1}, -\frac{R_2}{L_2}, ..., -\frac{R_n}{L_n}\right]\right)$$
$$B = diag\left(\left[-\frac{K_{m_1}K_{bemf_1}N_1{}^2}{L_1}, -\frac{K_{m_2}K_{bemf_2}N_2{}^2}{L_2}, ..., -\frac{K_{m_n}K_{bemf_n}N_n{}^2}{L_n}\right]\right) \qquad (7)$$
$$D = diag\left(\left[\frac{K_{m_1}N_1}{L_1}, \frac{K_{m_2}N_2}{L_2}, ..., \frac{K_{m_n}N_n}{L_n}\right]\right)$$
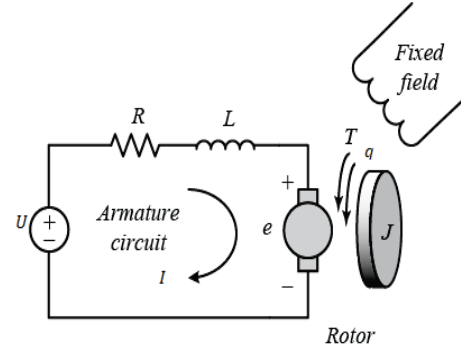

Figure 1 DC motor equivalent circuit.

As a result, the augmented equations of the motion of the manipulator can be summarized as:

$$T_c = M\ddot{q} + C\dot{q} + G(q) - T_{ext} - T_f \qquad (8)$$

By calculating the time derivative of Eq. (8), we have:

$$\dot{T}_c = \dot{M}\ddot{q} + M\dddot{q} + \dot{C}\dot{q} + C\ddot{q} + \frac{\partial G}{\partial q}\dot{q} - \dot{T}_{ext} - \dot{T}_f \qquad (9)$$

Eq. (7) is rearranged as:

$$U = D^{-1}(\dot{T}_c - AT_c - B\dot{q}) \qquad (10)$$

Here, $\dot{T}_{ext}$ and $\dot{T}_f$ are calculated using Eq. (2), thus:

$$\dot{T}_{ext} = \dot{J}^T F_{ext} + J^T \dot{F}_{ext} \qquad (11)$$

When the normal force is constant, by calculating the time derivative of F_{ext} using Eq. (3), we obtain:

$$\dot{F}_{ext} = -\mu\dot{v}f_n\hat{n}_v - \mu v f_n\dot{\hat{n}}_v \qquad (12)$$

where $\hat{n}_v$ and $\dot{\hat{n}}_v$ are the unit vector of velocity and its time derivative, respectively. Consequently, when the desired joint angles and their first, second and third time derivatives are known, to calculate U, we need to calculate T_c using the desired path, after which Eq. (9) is used to calculate $\dot{T}_c$. Finally, by using Eq. (10), U is calculated. The constraints of the motors are as follows [52]:

$$|\dot{q}_{m_i}| \le \bar{\bar{q}}_{m_i} \qquad (13.1)$$

$$|I_i| \le \bar{I}_i \qquad (13.2)$$

$$|U_i| \le \bar{U}_i \qquad (13.3)$$

$$|\dot{I}_i| \le \bar{\dot{I}}_i \qquad (13.4)$$

$$\sqrt{\frac{1}{t_f} \int_0^{t_f} I_i^2(t)\,dt} \le \bar{I}_{c_i}, \; (i = 1, 2, \dots, n) \qquad (13.5)$$

where n is the number of joint angles, and $\bar{\bar{q}}_{m_i}, \bar{I}_i, \bar{U}_i, \bar{\dot{I}}_i, \bar{I}_{c_i}$, are the maximum admissible motor speed, current, feeding voltage, the time derivative of current and braked motor current, respectively. Finally, we have to convert these constraints into robot manipulator constraints:

$$|\dot{q}_i| \le N^{-1}\bar{\bar{q}}_{m_i} \qquad (14.1)$$

$$|T_{c_i}| \le NK_m \bar{I}_i \qquad (14.2)$$

$$|\dot{T}_{c_i}| \le NK_m \bar{\dot{I}}_i \qquad (14.3)$$

$$\sqrt{\frac{1}{t_f} \int_0^{t_f} T_{c_i}^2(t)\,dt} \le N^2 K_m^2 \bar{I}_{c_i}^2, (i = 1, 2, \dots, n) \qquad (14.4)$$

where $\dot{q}_i, T_{c_i}, \dot{T}_{c_i}$ and $\sqrt{\frac{1}{t_f} \int_0^{t_f} T_{c_i}^2(t)\,dt}$ are the angular velocity of joint angles, the control torque, the time derivative of the control torque, and the guaranteed term for harmless overtaking of the permanent operating range [33].

3 DEFINITION OF THE PARAMETRIC TRAJECTORY OPTIMIZATION PROBLEM

We are interested in performing a task with the manipulator in the minimum-time. Therefore, the cost function can be expressed as:
Cost function:

$$J = \int_0^{t_f} dt = t_f \qquad (15.1)$$

Subject to the constraints:

$$T_c = M\ddot{q} + C\dot{q} + G(q) - T_{ext} - T_f$$

$$\dot{T}_c = \dot{M}\ddot{q} + M\dddot{q} + \dot{C}\dot{q} + C\ddot{q} + \frac{\partial G}{\partial q}\dot{q} - \dot{T}_{ext} - \dot{T}_f$$

$$\qquad (15.2)$$

$$U = D^{-1}(\dot{T}_c - AT_c - B\dot{q})$$

$$|\dot{q}_i| \le N^{-1}\bar{\bar{q}}_{m_i}$$

$$|T_{c_i}| \le NK_m \bar{I}_i$$

$$|\dot{T}_{c_i}| \le NK_m \bar{\dot{I}}_i$$

$$\sqrt{\frac{1}{t_f} \int_0^{t_f} T_{c_i}^2(t)\,dt} \le N^2 K_m^2, \qquad (i = 1, 2, \dots, n)$$

where M and C are functions of q and $\dot{q}$, respectively. Therefore, the time derivatives of these matrices are:

$$\dot{M} = \sum_{i=1}^n \frac{\partial M}{\partial q_i} \dot{q}_i$$

$$\dot{C} = \sum_{i=1}^n \frac{\partial C}{\partial q_i} \dot{q}_i + \sum_{i=1}^n \frac{\partial C}{\partial \dot{q}_i} \ddot{q}_i$$

$$\qquad (16)$$

Let us suppose that the purpose of path planning is to track a desired path in the Cartesian space that is parameterized by γ. The position of end-effector r is the function of the path parameter (γ).
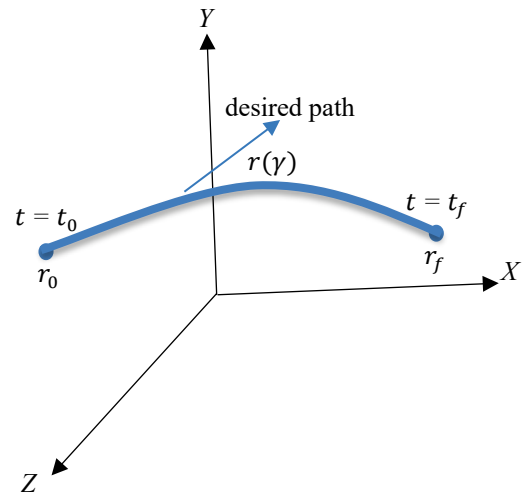
Figure 2 Trajectory planning along a parameterized path.

In Fig. 2, the coordinates of the start and stop points are r_0 and r_f. When the robot is at the starting point, for simplicity γ is $\gamma(t_0) = 0$, and at the stopping point, it is $\gamma(t_f) = \gamma_f$. The aim of an optimization problem is to find $\gamma(t)$ to minimize the cost function equation Eq. (15.1) by considering the constraints equation Eq. (15.2). In this paper, a polynomial approach is taken to convert the function optimization problem into a parameter optimization problem. Therefore, the approach will be a suboptimal solution. However, $\gamma(t)$ has a number of constraints due to the limitations of a manipulator. Firstly, we know that γ has two constraints due to the definition of the path at the initial and the final times as:

$$\gamma(t_0) = \gamma(0) = 0$$
$$\gamma(t_f) = \gamma_f \qquad (17)$$

To find other constraints, we have to find the relation between the kinematic constraints in the Cartesian space and the joint space. Therefore, we start by finding the relation between the velocity, acceleration, and jerk as functions of time in the Cartesian space, thus:

$$\dot{r} = v = r'\dot{\gamma}$$

$$\ddot{r} = a = r''\dot{\gamma}^2 + r'\ddot{\gamma} \tag{18}$$

$$\dddot{r} = \mathcal{J} = r'''\dot{\gamma}^2 + 2r''\ddot{\gamma} + r'\dddot{\gamma}$$

Here, $v, a,$ and $\mathcal{J}$ are the velocity, acceleration and jerk of the end-effector, respectively. The relation between kinematic parameters in the Cartesian space and the joint space are:

$$v = J\dot{q}$$

$$a = \dot{J}\dot{q} + J\ddot{q} \tag{19}$$

$$\mathcal{J} = \ddot{J}\dot{q} + 2\dot{J}\ddot{q} + J\dddot{q}$$

where J, $\dot{J}$, and $\ddot{J}$ are the Jacobian, first and second time derivatives of the Jacobian matrix at any time, respectively. It is desirable for the velocity, acceleration and jerk to be zero at the initial time. Therefore, by using Eq. (18), $\dot{q}$, $\ddot{q}$, $\dddot{q}$, will be zero. These conditions are simple in the view of the actuators for starting because the commands of the actuator will not jump to the maximum. Hence, if at the starting point the velocity, acceleration and jerk in the Cartesian space are zero, then the time derivatives of the path parameter $\dot{\gamma}(0)$, $\ddot{\gamma}(0)$ and $\dddot{\gamma}(0)$ will be zero. Therefore, the initial conditions for $\gamma(t)$ are:

$$\gamma(0) = 0$$

$$\dot{\gamma}(0) = 0 \tag{20}$$

$$\ddot{\gamma}(0) = 0$$

$$\dddot{\gamma}(0) = 0$$

It is necessary to define the final conditions for the path parameter. We know that γ_f is known by the definition of the path, hence again, in order to have zero velocity, zero acceleration, and zero jerk in the Cartesian and joint spaces, the final condition for $\gamma(t)$ will be:

$$\gamma(t_f) = \gamma_f$$

$$\dot{\gamma}(t_f) = 0$$

$$\ddot{\gamma}(t_f) = 0 \tag{21}$$

$$\dddot{\gamma}(t_f) = 0$$

For a suboptimal solution, let us suppose that the function $\gamma(t)$ is approximated by a time series as:

$$\gamma(t) = \sum_{i=4}^{n} a_i t^i \tag{22}$$

In the above format, all initial conditions will satisfy, but the final conditions will not satisfy. Because of the simplicity, for the satisfaction of the final conditions, we will consider a model for $\gamma(t)$ to be:

$$\gamma(t) = A\gamma_1(t) + B\gamma_2(t) + C\gamma_3(t) + D\gamma_4(t) \tag{23}$$

Here, $A, B, C,$ and D are constant and unknown parameters and γ_1, γ_2, γ_3 and γ_4 have a format as in Eq. (22); therefore, they will satisfy the initial conditions:

$$\gamma_1(t) = \sum_{i=4}^{n} a_i t^i$$

$$\gamma_2(t) = \sum_{i=4}^{n} b_i t^i$$

$$\gamma_3(t) = \sum_{i=4}^{n} c_i t^i \tag{24}$$

$$\gamma_4(t) = \sum_{i=4}^{n} d_i t^i$$

Therefore, when a_i, b_i, $c_i,$ and $d_i, (i = 4, ..., n)$, and t_f are known, we are able to calculate $A, B, C,$ and D as:

$$A\gamma_1(t_f) + B\gamma_2(t_f) + C\gamma_3(t_f) + D\gamma_4(t_f) = \gamma_f$$

$$A\dot{\gamma}_1(t_f) + B\dot{\gamma}_2(t_f) + C\dot{\gamma}_3(t_f) + D\dot{\gamma}_4(t_f) = 0$$

$$A\ddot{\gamma}_1(t_f) + B\ddot{\gamma}_2(t_f) + C\ddot{\gamma}_3(t_f) + D\ddot{\gamma}_4(t_f) = 0 \tag{25}$$

$$A\dddot{\gamma}_1(t_f) + B\dddot{\gamma}_2(t_f) + C\dddot{\gamma}_3(t_f) + D\dddot{\gamma}_4(t_f) = 0$$

In Eq. (25), there are four unknown parameters $A, B, C,$ and D, when $\gamma_1, \gamma_2, \gamma_3$ and γ_4 at the final time are known. Therefore, for the satisfaction of the final conditions, we can then write:

$$\begin{bmatrix} A \\ B \\ C \\ D \end{bmatrix} = \begin{bmatrix} \gamma_1(t_f) & \gamma_2(t_f) & \gamma_3(t_f) & \gamma_4(t_f) \\ \dot{\gamma}_1(t_f) & \dot{\gamma}_2(t_f) & \dot{\gamma}_3(t_f) & \dot{\gamma}_4(t_f) \\ \ddot{\gamma}_1(t_f) & \ddot{\gamma}_2(t_f) & \ddot{\gamma}_3(t_f) & \ddot{\gamma}_4(t_f) \\ \dddot{\gamma}_1(t_f) & \dddot{\gamma}_2(t_f) & \dddot{\gamma}_3(t_f) & \dddot{\gamma}_4(t_f) \end{bmatrix}^{-1} \begin{bmatrix} \gamma_f \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{26}$$

Hence, when $A, B, C,$ and D are selected from Eq. (26), the initial and final conditions will always be satisfied. Consequently, $\gamma(t)$ will be a function of the unknown vector as:

$$\gamma(t) = \gamma(x, t)$$
$$x = [a_4, \dots, a_n, b_4, \dots, b_n, c_4, \dots, c_n, d_4, \dots, d_n, t_f]^T \qquad (27)$$

Here, x is an unknown vector that has to be found by the minimization of the cost function in Eq. (15.1) subjected to the constraint Eq. (15.2). Therefore, the objective function Eq. (15.1) and constraints Eq. (15.2) will updated as:
Cost function:

$$J(x) = [0_{1 \times 4(n-4)} \ 1] x$$

The constraint equations are:

$$T_c = M\ddot{q} + C\dot{q} + G(q) - T_{ext} - T_f$$

$$\dot{T}_c = \dot{M}\ddot{q} + M\dddot{q} + \dot{C}\dot{q} + C\ddot{q} + \frac{\partial G}{\partial q}\dot{q} - \dot{T}_{ext} - \dot{T}_f$$

$$U = D^{-1}(\dot{T}_c - AT_c - B\dot{q})$$

$$|\dot{q}_i| \le N^{-1}\bar{\dot{q}}_{m_i} \qquad (28)$$

$$|T_{c_i}| \le N K_m \bar{I}_i$$

$$|\dot{T}_{c_i}| \le N K_m \bar{\dot{I}}_i$$

$$\sqrt{\frac{1}{t_f}\int_0^{t_f} T_{c_i}{}^2(t)\,dt} \le N^2 K_m{}^2 \bar{I}_{c_i}{}^2, \ (i = 1, 2, \dots, n)$$

4 MANAGING THE CONSTRAINTS OF THE OPTIMIZATION PROBLEM

The purpose of the optimization problem is to minimize the final time subjected to the dynamic of the manipulator. As mentioned previously, parameters are unknown.
Here they are collected in the vector x. Therefore, the objective function can be written as:

$$J(x) = x_{\widehat{N}+1} = t_f \qquad (29)$$

Here, $\widehat{N}$ is the number of unknown parameters for the modelling of $\gamma(t)$; therefore, $4(n-4) = \widehat{N}$. However, there are a number of constraints due to the limitations related to the angular velocity and the motor input voltages, and the torques of the motors. These constraints may appear at any time. We can manage the constraints of the problem at any time by adding the previous constraints. A simple method of doing this is to divide the time between the initial and final time with known and constant m incremental times. The constraints listed in Table 1 are divided into two types. The first type consists of differential equations and the second type consists of normal nonlinear or linear inequality equations.
In this paper, we suggest rewriting every constraint without solving the differential equations. We know that some constraints may appear at all times, hence we can manage the constraints of the problem at any time by adding to the previous constraints. A simple method for doing so is to generate new constraints for any time interval. Therefore, when the vectors x and t_f are known, we can divide the t_f to m incremental times as:

$$\Delta t = \frac{t_f}{m}, \ t_k = \Delta t k, \ (k = 0, \dots, m) \qquad (30)$$

To prepare the constraints as a function of the unknown vector x, the initial time x_0 is first generated as a random vector. This random vector will guarantee the initial and final condition of the path parameter, but it is necessary for it to be tuned for other constraints. Tab. 1 shows the processes of the generating of constraints for each time as listed in Tab. 1.

Table 1 Managing constraints for the optimization problem at time t_k

Initial time	- Initial estimate of the final time $t_f(0)$
	- Generating a random vector x
For $(k = 1, \dots, m)$	- using of x, calculate $\gamma_k, \dot{\gamma}_k, \ddot{\gamma}_k, \dddot{\gamma}_k$
	- using Eq. (18), calculate $r_k, \dot{r}_k, \ddot{r}_k, \dddot{r}_k$
	- Calculate: $q_k = q(r_k)$
	J_k and $\dot{J}_k$
	$\dot{q}_k = J_k{}^{-1} r_k{}' \dot{\gamma}_k J_k$
	$v = J_k \dot{q}_k$
	- Calculate: $M_k(q_k), C_k(q_k, \dot{q}_k), G_k, T_{ext_k}, T_{f_k}$
	$\ddot{q}_k = J_k{}^{-1}(-\dot{J}_k \dot{q}_k + r_k{}'' \dot{\gamma}_k{}^2 + r_k{}' \ddot{\gamma}_k)$
	$T_{c_k} = M_k \ddot{q}_k + C_k \dot{q}_k + G_k q_k - T_{ext_k} - T_{f_k}$
	- Calculate: $\dot{M}_k(q_k, \dot{q}_k), \dot{C}_k, \dot{G}_k, \dot{T}_{ext_k}, \dot{T}_{f_k}$
	$\dddot{q}_k = J_k{}^{-1}(-J_k{}^{-1}\dot{q}_k - 2\dot{J}_k \ddot{q}_k + r_k{}''' \dot{\gamma}_k{}^2 + 2r_k{}''\ddot{\gamma}_k + r_k{}'\dddot{\gamma}_k)$
	$\dot{T}_{c_k} = \dot{M}\ddot{q} + M\dddot{q} + \dot{C}\dot{q} + C\ddot{q} + \frac{\partial G}{\partial q}\dot{q} - \dot{T}_{ext}$
	$U_k = D^{-1}(\dot{T}_{c_k} - AT_{c_k} - B\dot{q}_k)$
	Constraints are:
	$C_k = \begin{cases} U_k - \bar{U} \\ \underline{U} - U_k \\ T_{c_k} - \bar{T}_c \\ \underline{T}_c - T_{c_k} \\ \dot{q}_k - \bar{\dot{q}} \\ \underline{\dot{q}} - \dot{q}_k \\ \dot{T}_{c_k} - \bar{\dot{T}}_c \\ \underline{\dot{T}}_c - \dot{T}_{c_k} \\ \gamma_k \ge 0 \\ \dot{\gamma}_k \ge 0 \end{cases} \le 0$

In Tab. 1, $\bar{a}$ and $\underline{a}$ are respectively used to represent the maximum and minimum of the parameter a. Therefore, the size of constraints at each step will increase. There are ten constraints in each t_k, hence the size of the constraint vector at the end will be $10(m+1)$. In Eq. (28), there is an integral equation for any motor. This integral can be approximated as:

$$\Delta t \sum_{i=1}^{N} T_{c_i}{}^2 = N\Delta t \bar{T}_{c_i}{}^2 \qquad (31)$$

Therefore, at the end of each iteration, another constraint will be added to the previous constraints.

5 SIMULATION STUDY

Simulations in the MATLAB environment were used to test the performance of the proposed algorithm. A model of the SCARA robot (Selective Compliance Assembly Robot Arm) used in the simulation study are given in ref. [33]. However, there are some additional parameters and small changes that we considered in the simulations. The model of the manipulator (represented in Fig. 3) that is considered for the simulation is:

$$\begin{bmatrix} (3.78 + 0.272\cos q_2 + 0.022\sin q_2) + (0.08 + 0.136\cos q_2 + 0.011\sin q_2) & 0 \\ (0.08 + 0.136\cos q_2 + 0.011\sin q_2) & 0.08 \end{bmatrix}\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} +$$
$$\begin{bmatrix} (0.011\cos q_2 - 0.136\sin q_2) + (\dot{q}_2 + \dot{q}_1) + 0.07 & (0.011\cos q_2 - 0.136\sin q_2)\dot{q}_1 \\ (0.011\cos q_2 - 0.136\sin q_2)\dot{q}_1 & 0.013 \end{bmatrix}\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = \begin{bmatrix} T_{c_1} \\ T_{c_2} \end{bmatrix} +$$
$$\begin{bmatrix} 0.62 \text{sign} \dot{q}_1 \\ 0.17 \text{sign} \dot{q}_2 \end{bmatrix} + \begin{bmatrix} K_{v_1} & 0 \\ 0 & K_{v_2} \end{bmatrix}\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + \begin{bmatrix} T_{ext_1} \\ T_{ext_2} \end{bmatrix} \tag{32-1}$$

The position of the end-effector is a function of the length of the robot arm and joint angles, thus:

$$r = \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} L_1 \cos q_1 + L_2 \cos(q_1 + q_2) \\ L_1 \sin q_1 + L_2 \sin(q_1 + q_2) \end{bmatrix} \tag{32-2}$$

By using the above forward kinematic, the inverse kinematic of the robot is:

$$q_1 = \tan^{-1}\left(\frac{y}{x}\right) - \tan^{-1}\left(\frac{K_2}{K_1}\right) \tag{33}$$

$$q_2 = \tan^{-1}\left(\frac{\sin(q_2)}{\cos(q_2)}\right) \tag{34}$$

where K_1, K_2, $\sin(q_2)$ and $\cos(q_2)$ are:

$$K_1 = L_1 + L_2\cos(q_2), \ K_2 = L_2\sin(q_2)$$

$$\sin q_2 = \pm\sqrt{1 - \cos q_2{}^2}$$

$$\cos q_2 = \frac{x^2 + y^2 - L_1{}^2 - L_2{}^2}{2L_1L_2}$$

Figure 3 Photo of the IRCCyN SCARA robot

The parameters of the robot are presented in Tab. 2.

Table 2 Parameters of the IRCCyN SCARA robot.

Task Nº	μ	K_v	K_c	L (m)
1	0.2	0.4	0.01	0.21
2	0.2	0.3	0.01	0.3

The normal force is $f_n = 0.1N$. The electro-mechanical constraints of the actuators are presented in Tab. 3.

Table 3 Electro-mechanical constraints of the SCARA robot.

Axis	$\bar{U}$ (volt)	q (deg)	$\bar{\dot{q}}$ (rad · s^{-1})
1	20	0-270	1.5
2	20	0-180	2

The purpose of the simulations is to plot Fig. 3 with a pencil attached to the end-effector. Hence, there is the desired path, and the end-effector has to track that path. Here, the Cartesian path as a function of the path parameter is considered to be:

$$r = \begin{bmatrix} \rho\sin(\gamma) \\ 0.5\rho\cos(\gamma) \end{bmatrix} \tag{35}$$

Then, the derivative of r with respect to the path parameter (γ) are listed as follows:

$$r' = \begin{bmatrix} \rho'\sin(\gamma) + \rho\cos(\gamma) \\ 0.5\rho'\cos(\gamma) - 0.5\rho\sin(\gamma) \end{bmatrix} \tag{36}$$

$$r'' = \begin{bmatrix} \rho''\sin(\gamma) + 2\rho'\cos(\gamma) - \rho\sin(\gamma) \\ 0.5\rho''\cos(\gamma) - \rho'\sin(\gamma) + 0.5\rho\cos(\gamma) \end{bmatrix} \tag{37}$$

$$r''' = \begin{bmatrix} \rho'''\sin(\gamma) + 3\rho''\cos(\gamma) - 3\rho'\sin(\gamma) + \rho\cos(\gamma) \\ 0.5\rho'''\cos(\gamma) - 1.5\rho''\sin(\gamma) - 0.5\rho'\cos(\gamma) - \rho\sin(\gamma) \end{bmatrix} \tag{38}$$

Here $\rho, \rho', \rho'', \rho'''$, are:

$$\rho = 0.4 - 0.1\gamma\cos(\gamma), \ (0 \le \gamma \le 200^0)$$

$$\rho' = -0.1\cos(\gamma) + 0.1\gamma\sin(\gamma)$$

$$\rho'' = 0.2\sin(\gamma) + 0.1\gamma\cos(\gamma)$$

$$\rho''' = -0.4\sin(\gamma) - 0.1\gamma\cos(\gamma)$$

The desired path is presented in Fig. 4 and the starting and stopping points are:

$$r_0 = \begin{bmatrix} 0 \\ 0.2 \end{bmatrix} m, \ r_f = \begin{bmatrix} -0.1965 \\ -0.2699 \end{bmatrix} m$$

The desired path, start-point and end-point in the Cartesian space are shown in Fig. 4. To provide an approximation for γ as a function of time, n in Eq. (22) is set to be $n = 10$. The Taylor expansion will contain terms up to t^{10} and $a_i = b_i = c_i = d_i = 0, (i = 1, ...,4)$. The number of unknown parameters for the representation of γ is (4×6) and by adding t_f, there will be $(4 \times 6 + 1) = 25$ unknown parameters. To initialize the unknown parameters, we use a

normal distribution to generate the coefficients of the path parameter. Moreover, the initial estimation of the final time is considered to be (17) seconds. The time between (0 and 17) seconds is divided into (5000) points; then, as mentioned in Tab. 1, the number of constraints is (50 000). For any iteration, when the time is increasing, the number of constraints will increase. After running the optimization algorithm, the final time is found to be (14.4296) seconds and the results below are found. The path parameter and its first, second and third time derivatives are presented in Figs. 5 and 6.

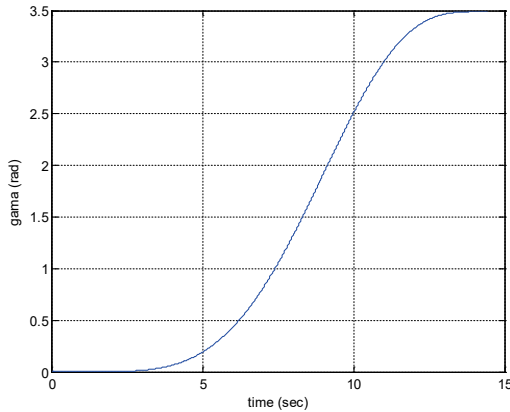Fig. 5 shows that the path parameter starts at zero and stops at (200 deg = 3.4907 rad). The path parameter has a smooth graph and Fig. 6 shows the first three-time derivatives of the path parameter. Those parameters are smooth and have zero values at the start and stop points. In addition to that, the first-time derivative has a positive value. Consequently, γ is an increasing function. Then, when the time increases, the end-effector always goes toward the final point and it does not have any backward motion. However, the other derivatives are sometimes positive and sometimes they are negative because the system changes the acceleration along the Cartesian path.
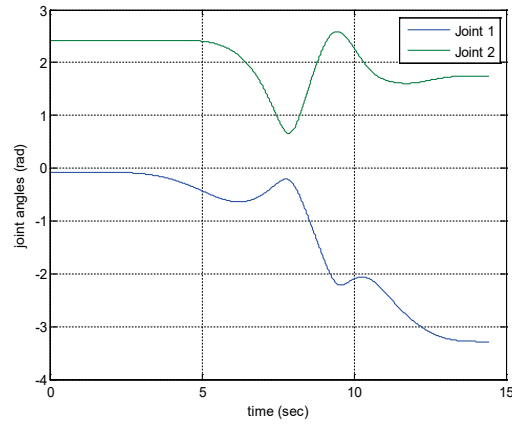

Figure 4 Desired path along the x and y axes

Figure 5 Optimal path parameter
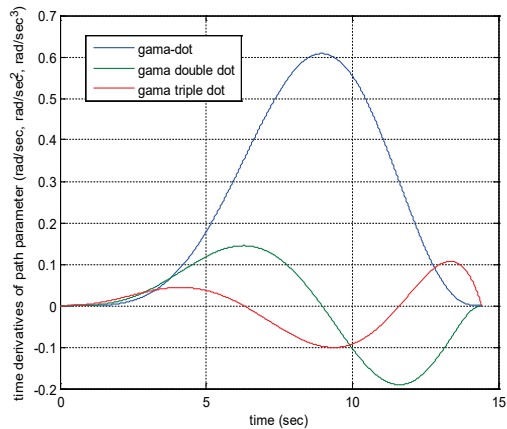
Figure 6 Time derivatives of the path parameter
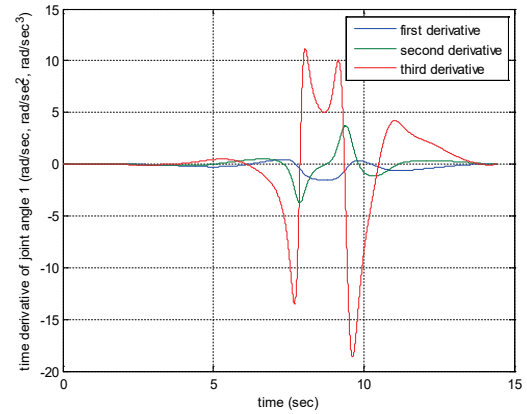

Figure 7 Time history of joint angles

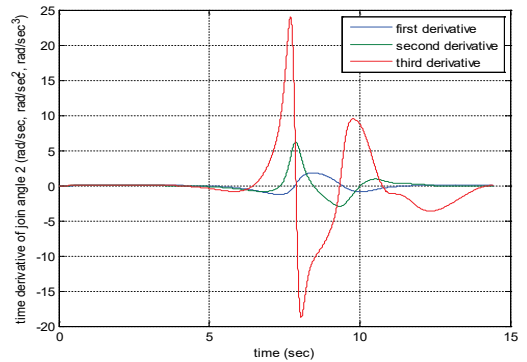**Figure 8** Time derivative of the first joint.

Figure 9 Time derivative of the second joint.

Fig. 7 shows the joint angles and Figs. 8 and 9 show the first, second and third time derivatives of the joint angles. We can see that the time derivatives of the joint angles are smooth and at the initial and final times, they are equal to zero. The figures show that the angular velocity of the joint angles has positive and negative values, which is why the robot changes its angular velocity. The absolute values of the maximum angular velocities are $\begin{bmatrix} 1.5 \\ 1.757 \end{bmatrix}$ rad, and are in the defined range, as presented in Tab. 2.
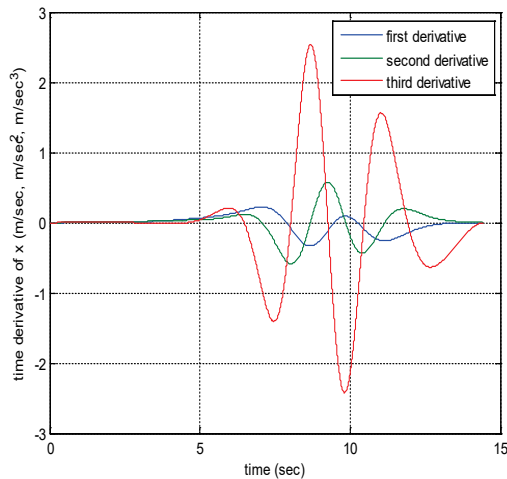

Figure 10 Velocity, acceleration and jerk along the x direction in the Cartesian–space
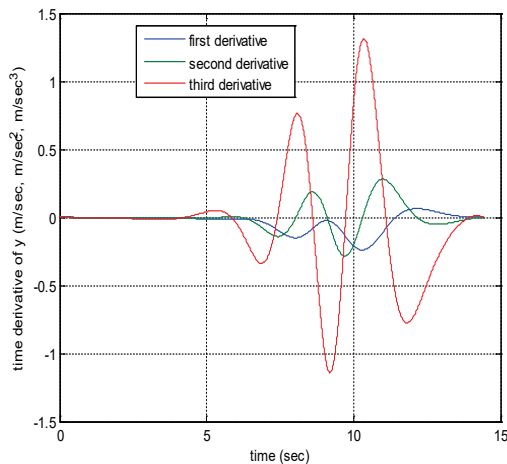

Figure 11 Velocity, acceleration and jerk along the y direction in the Cartesian–space

Figs. 10 and 11 show that the third time derivatives of position have zero values at the initial and the final times and are smooth graphs; and then the end-effector moves smoothly in the Cartesian space.

Fig. 14 shows the feeding voltage of the motors, which are smooth factions and have zero values at the start and stop points. We can see that the robot uses maximum voltages for link one, which is (20) volts and uses (15.161) volts for the second links. Therefore, the feeding voltages are in the range.

6 CONCLUSIONS

In this work, a new technique based on path parameter optimization is used for the path planning problem in the Cartesian space with external forces and frictions. The actuators of the robot are modeled as permanent magnet DC motors with a consideration of their constraints. By dividing the time between the start and stop times to known portions, the path parameter optimization problem is converted to the optimization of a function subject to certain equality and inequality constraints. A MATLAB simulation is used for the path planning of a two-degree robot manipulator with a desired path in the Cartesian space. The path parameter is used in the formulation of the desired path as a function of time. A polynomial model is considered for the path parameter so that it can guarantee constraints at the start and stop points. Consequently, the three first-time derivatives of the joint angles and the position of the end-effector will be zeros. Additionally, the voltages, torques and time derivatives of the torques are zeros at boundary conditions. It is shown that the approach can hold all constraints related to the actuators and other kinematics constraints between the initial and final times. The method was suboptimal due to the consideration of a polynomial model for the path parameter and it will not be a global optimum point for the problem. The best advantages of the method are all the dynamic and kinematic parameters which remain smooth, and it is practically more important that we be able to automatically control the start and stop conditions. Computer simulation results show the satisfactory performance responses of the method on a robotic manipulator path planning.

7 REFERENCES

[1] Bobrow, J. E., Dubowsky, S., & Gibson, J. S. (1985). Time-optimal control of robotic manipulators along specified paths. *The International Journal of Robotics Research*, 4, 3-17. https://doi.org/10.1177%2F027836498500400301

[2] Shin, K. G. & McKay, N. D. (1986). Selection of near-minimum time geometric paths for robotic manipulators. *IEEE Transactions on Automatic Control*, June 1986, 31, 501-511. https://doi.org/10.1109/TAC.1986.1104316

[3] Lin, C. S., Chang, P. R., & Luh, J. Y. S. (1983). Formulation and optimization of cubic polynomial joint trajectories for industrial robots. Automatic Control, IEEE Transactions 1983 on, 28(12), 1066-1074. https://doi.org/10.1109/TAC.1983.1103181

[4] Hans, P. G., Lino, G., Stephan, A. R. H., & Christopher, H. O. (1985). *Time-optimal motions of robots in assembly tasks*. IEEE-Institute of Electrical and Electronics Engineers, Inc., 1985. Fort Lauderdale, FL, USA, USA1 December 1985, 982-989. https://doi.org/10.1109/CDC.1985.268647
Luh, J. & Zheng, Y. F. (1985). Computation of input generalized forces for robots with closed kinematic chain mechanisms. *IEEE Journal of Robotics and Automation, 1*(2), 95-103. https://doi.org/10.1109/JRA.1985.1087008

[5] Shin, K. G. & Mckay, N. D. (1985). Minimum time control of robotic manipulators with geometric path constraints. *IEEE Transactions on Automatic Control, AC* 30, 531-541.

[6] Dubowsky, S., Norris, M. A., & Shiller, Z. (1986). Time optimal trajectory planning for robotic manipulators with

obstacle avoidance: A CAD approach. In IEEE International Conference on Robotics and Automation 1986, San Francisco, California, 1906-1912. https://doi.org/10.1109/ROBOT.1986.1087434

[7] Pfeiffer, F. & Johanni, R. (1986). A concept for manipulator trajectory planning. In *Proceedings of the International Conference on Robotics and Automation*, 3, 1399-1405.

[8] Bobrow, J. E. (1988). Optimal robot plant planning using the minimum-time criterion. *IEEE Journal on Robotics and Automation*, 4, 443-450. https://doi.org/10.1109/56.811

[9] Cao, B., Dodds, G. I., & Irwin, G. W. (1994). Time optimal and smooth constrained path planning for robot manipulators. *Proceedings of the International Conference on Robotics and Automation*, Minneapolis, 43(2), 1853-1858. https://doi.org/10.1109/ROBOT.1994.351191

[10] Gang, L. & Chao, Y. (2016). A fast approach for time-optimal and smooth trajectory planning of robot manipulators. *Journal of Harbin Institute of Technology, (New Series), 23*(5).

[11] Lepetic, M., Klancar, G., Skrjanc, I., & Matko, D. et al. (2003). Time-optimal path planning considering acceleration limits. Robotics and Autonomous Systems, 45, 199-210. https://doi: 10.1016/j.robot.2003.09.007

[12] Cao, B., Dodds, G., & Irwin, G. (1997). Constrained time-efficient and smooth cubic spline trajectory generation for industrial robots. In IEEE Proceedings Control Theory and Applications, 144, 467-475. https://doi.org/10.1049/ip-cta:19971494

[13] Chang, Y. H., Lee, T. T., & Liu, C. H. (1992). On-line approximate cartesian path trajectory planning for robotic manipulators. *IEEE Transactions on Systems, Man, and Cybernetics, 22*(3), 542-547. https://doi.org/10.1109/21.155953

[14] Murray, J. J. & Lovell, G. (1989). Dynamic modelling of closed-chain robotic manipulators and implications for trajectory control. *Robotics and Automation, IEEE Transactions on*, 5(4), 522-528. https://doi.org/10.1109/70.88066

[15] Shiller, Z., Chang, H., & Wong, V. (1996). The practical implementation of time-optimal control for robotic manipulators. *Robotics and Computer-Integrated Manufacturing, 12*(1), 29-39. https://doi.org/10.1016/0736-5845(95)00026-7

[16] Lloyd, J. & Hayward, V. (1993). Trajectory generation for sensor-driven and time-varying tasks. International Journal of Robotics Researches 1993, 12(4), 380-393. https://doi.org/10.1177%2F027836493301200405

[17] Volpe, R. (1993). Task space velocity blending for real-time trajectory generation. Proceedings of IEEE International Conference on Robotics and Automation, Atlanta, GA, 2-6 May 1993, 680-687. https://doi.org/10.1109/ROBOT.1993.291880

[18] Sciavicco, L. & Siciliano, B. (1996). Modelling and control of robot manipulators. The McGraw-Hill Companies Inc. 1996, New York.

[19] Constantinescu, D. & Croft, E. A. (2002). Smooth and time-optimal trajectory planning for industrial manipulators along specified paths. *Journal Robotic System, 17*(5), 233-249. https://doi.org/10.1002/(SICI)10974563(200005)17:5%3C233::AID-ROB1%3E3.0.CO;2-Y

[20] Sandor G. N. & Erdman A. G. (1985). Advanced mechanism design; Analysis and synthesis. Prentice-Hall, *Mechanism and Machine Theory, 2*(1), 81.

[21] Shiller, Z. (1994). On singular time-optimal control along specified paths. *IEEE Transactions on Robotics and Automation, 10*(4), 561-566. https://doi.org/10.1109/70.313107

[22] Kardos, E. S. & Kiss, B. (2009). Continuous-curvature paths for mobile robots. *Periodica Polytechnic Electrical Engineering, 53*(1-2), 63-72.

[23] Bianco, C. G. L. & Piazzi, A. (2010). Minimum-time trajectory planning of mechanical manipulators under dynamic constraints. *International Journal Control, 75*(13), 967-980. https://doi.org/10.1080/00207170210156161

[24] Kim, K. W., Kim, H. S., Choi, Y. K., & Park, J. H. (1997). Optimization of cubic polynomial joint trajectories and sliding mode controllers for robots using evolution strategy. *Proceedings of the 23rd International Conference on Industrial Electronics Control and Instrumentation IECON 1997*, 3, 1444-1447. https://doi.org/10.1109/IECON.1997.668531

[25] Aurelio, P. & Antonio, V. (2000). Global minimum-jerk trajectory planning of robot manipulators. *IEEE Transactions on Industrial Electronics, 47*(1), 140-149. https://doi.org/10.1109/41.824136

[26] Gasparetto, A. & Zanotto, V. (2009). Optimal trajectory planning for industrial robots. *Advances in Engineering Software*, 41, 548-556. https://doi.org/10.1016/j.advengsoft.2009.11.001

[27] Incerti, G. (2015). Motion planning of SCARA robots for trajectory tracking. *Excellence in Research and Innovation for Humanity, Part I.* Rome Italy, 17(5). https://10.13140/RG.2.2.29780.40321

[28] Zhao, Y. & Tsiotras, P. (2013). Speed profile optimization for optimal path tracking. *American Control Conference (ACC)*, 1171-1176. https://doi.org/10.1109/ACC.2013.6579994

[29] Wang, C. H. & Horng, J. G. (1990). Constrained minimum-time path planning for robot manipulators via virtual knots of the cubic B-spline functions. IEEE Transactions on Automatic Control, 35(5), 573-577. https://doi.org/10.1109/9.53526

[30] Sakamoto, K. & Kawamura, A. (1993). Trajectory planning using optimum Solution of variational problem. *Power Conversion Conference*, Yokohama, Japan, 666-671. https://doi.org/10.1109/PCCON.1993.264177

[31] Macfarlane, S. & Croft, E. A. (2003). Jerk-bounded manipulator trajectory planning: Design for real-time applications. *IEEE Transactions on Robotics and Automatic, 19*(1), 42-52. https://doi.org/10.1109/TRA.2002.807548

[32] Chettibi, T. & Lemoine, P. (2007). Generation of point-to-point trajectories for robotic manipulators under electro-mechanical constraints. *International Review of Mechanical Engineering (I.RE.M.E.), 1*(2), 131-143.

[33] Murray, J. J. & Lovell, G. (1989). Dynamic modelling of closed-chain robotic manipulators and implications for trajectory control. *Robotics and Automation, IEEE Transactions on, 5*(4), 522-528. https://doi.org/10.1109/70.88066

[34] Kröger, T. & Wahl, F. M. (2010). On-line trajectory generation: Basic concepts for instantaneous reactions to unforeseen events. *IEEE Transactions on Robotics, 26*(1), 94-111. https://doi.org/10.1109/TRO.2009.2035744

[35] Sahar, G. & Hollerbach, J. M. (1986). Planning of minimum-time trajectories for robot arms. *The International Journal of Robotic Research*, 5, 90-100. https://doi.org/10.1177%2F027836498600500305

[36] Haschke, R., Weitnauer, E., & Ritter, H. (2008). On-line planning of time-optimal, jerk-limited trajectories. In *Proceedings IEEE/RSJ International Conferences on Intelligent Robots and Systems (IROS) 2008*, Nice, France, 3248-3253. https://doi.org/10.1109/IROS.2008.4650924

[37] Guarino Lo Bianco, C., & Piazzi, A. A. (2001). *Semi-infinite optimization approach to optimal spline trajectory planning of mechanical manipulators, semi-infinite programming: Recent*

advances. Goberna M. A. and Lopez M. A. (Eds.), Springer Verlag 2001, 271-297.

[38] Dijk, van, N. J. M., Wouw, van de, N., Pancras, W. C. M., & Nijmeijer, H. (2007). Path-constrained motion planning for robotics based on kinematic constraints. In *Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE) 2007*, 4-7 September 2007, Nevada, USA,1-10.

[39] Dongmei, X., & Daokui, Q., Fang, X. (2006). Path constrained time-optimal robot control. In *Proceedings of the International Conference on Robotics and Biomimetics*, 1095-1100. https://doi.org/10.1109/ROBIO.2006.340081

[40] Tangpattanakul, P. & Artrit, P. (2009). Minimum time trajectory of robot manipulator using harmony search algorithm. In *Proceedings of the IEEE 6th International Conference on ECTI-CON 2009*, 354-357. https://doi.org/10.1109/ECTICON.2009.5137025

[41] Martin B. J. & Bobrow, J. E. (1995). Determination of minimum-effort motions for general open chains. In *IEEE International Conference on Robotics and Automation*, 1160-1165, Piscataway, New Jersey. https://doi.org/10.1109/ROBOT.1995.525437

[42] Abo-Hammoura, Zaer, S., Nasir M. Mirza., Sikander M. Mirza., & Muhammad, A. (2002). Cartesian path generation of robot manipulators using continuous genetic algorithms. *Robotics and Autonomous Systems*, 41, 179-223.

[43] Yao, Z. & Gupta, K. (2007). Path planning with general end-effector constraints. *Robotics and Autonomous Systems*, 55, 316-327. https://doi.org/10.1109/IROS.2005.1545305

[44] Ardeshiri, T., Norrlof, M., Lofberg, J., & Hansson, A. (2011). Convex optimization approach for time-optimal path tracking of robots with speed dependent constraints. In 18th IFAC World Congress, Milano, Italy. https://10.3182/20110828-6-IT-1002.01136

[45] Oksanen, T. & Visala, A. (2009). Coverage path planning algorithms for agricultural field machines. *Journal of Field Robotics, 26*(8), 651-668. https://doi.org/10.1002/rob.20300

[46] Technitis, G., Othman, W., Safi, K., & Weibel, R. (2015). From A to B, randomly: A point-to-point random trajectory generator for animal movement. *International Journal of Geographical Information Science, 29*(6), 912-934. https://doi.org/10.1080/13658816.2014.999682

[47] Park, F. C., Bobrow J. E., & Ploen, S. R. (1995). A lie group formulation of robot dynamics. *International Journal of Robotics Research, 14*(6), 609-618. https://doi.org/10.1177/027836499501400606

[48] Khalil, W. & Dombre, E. (2002). *Modelling identification and control of robots*. London, U. K. HPS.

[49] Plédel, P. & Bestaoui, Y. (1995). Actuator constraints in point-to-point motion planning of manipulators. *Proceedings of the 34th Conferences on Decision and Control*, New Orleans, LA, 1009-1010. https://doi.org/10.1109/CDC.1995.480220

[50] Bianco, C. & Ghilardelli, F. (2014). Real-time planner in the operational space for the automatic handling of kinematic constraints. *IEEE Transactions on Automatic Sciences and Engineering, 11*(3), 730-739. https://doi.org/10.1109/TASE.2014.2310813

[51] Moojin, K., Wonkyu, M., Daesung, B., & Ilhan, P. (2004). Dynamic simulations of electro-mechanical robotic systems driven by DC motors. *Robotica*, 22, 523-531.

[52] Spong, M. W. & Vidyasagar, M. (2008). Robot dynamics and control. John Wiley and Sons 2008.

[53] Cho, B. H., Choi, B. S., & Lee, J. M. (2006). Time-optimal trajectory planning for a robot system under torque and impulse constraints. *International Journal of Control Automation and Systems, 4*(1), 10-16.

[54] Calzolari, D., Schürmann, B., & Althoff, M. (2017). Comparison of trajectory tracking controllers for autonomous vehicles. In *Intelligent Transportation Systems (ITSC) 2017, IEEE 20th International Conference on*, 1-8.

[55] Leena, N. & Saju, K. K. (2016). Modelling and trajectory tracking of wheeled mobile robots. *Procedia Technology*, 24, 538-545.

LIST OF SYMBOLS

M	Inertia matrix
$q, \dot{q}$	Joints of position and velocity
$\ddot{q}$	Joint of acceleration
$\dddot{q}$	Joint of jerk
C	Coriolis centrifugal torque
G	Gravity torque
n	The number of joints
T_m	$[n \times 1]$ Vector of the motor torque constant
N	$[n \times n]$ Matrix of the gear transmission ratio
T	$[n \times 1]$ Total torques
T_c	Control torque
T_{ext}	External torque
T_f	Friction torque
F_{ext}	External force
J	Jacobian matrix
v	Normal velocity
f_n	Normal force
μ	Friction coefficient
$\hat{n}_v$	Unit vector along the velocity vector
K_c	Coulomb friction coefficient of the joint
K_v	Viscous friction coefficient of the joint
$\dot{q}_m$	Motor speed
I	Armature current in the motor
L	Armature inductance of the motor
R	Armature resistance of the motor
U	Armature voltage of the motor
K_{bemf}	Back electromotive force of the DC motor
K_m	Motor torque constant matrix
$\bar{U}$	Maximum of the motor voltage
$\bar{\dot{q}}_m$	Maximum of speed motor
$\bar{I}$	Maximum current in the motor
$\bar{I}_c$	Maximum armature current
$\underline{T_c}$	Minimum control torque
$\underline{U}$	Minimum motor voltage
$\underline{\dot{q}}$	Minimum joint velocity
J	Cost function in the objective function
v	Velocity of the end-effector
a	Acceleration vector
$\mathcal{J}$	Jerk vector (i.e., derivative of acceleration)
γ_0, γ_f	Path parameter at zero and final times
r_0, r_f	Initial position and final positions
t_0, t_f	Start and final times

Author's contacts:

Habib GHANBARPOUR ASL, Assistant Professor
University of Turkish Aeronautical Association
Department of Mechatronics
Bahçekapı Mahallesi, Okul Sk. No: 11, 06790 Etimesgut/Ankara, Turkey
Tel: +905060214718
E-mail: hghanbarpourasl@thk.edu.tr, habib.ghanbarpour@gmail.com