



Optimization the Trajectories of Robot Manipulators Along Specified Task

Bendali Nadir^{1,2*} Ouali Mohammed¹

¹*Structural Mechanics Research Laboratory, Department of Mechanical Engineering,
University Blida-1, Algeria*

²*Faculty of Science and Technology, University of Khemis Miliana,
Road of Theniet El Had, Khemis Miliana, 44225, Algeria*

* Corresponding author's Email: nadir_b1102@yahoo.com

Abstract: This paper proposes a procedure for obtaining an optimal trajectory for robot manipulators along specified tasks (spot welding, drilling ...). The purpose is to obtain smooth trajectories with minimum time using spline cubic functions under a various kinematic and/or dynamic constraints, taking into account a full dynamic model of robot manipulator and with a large number of via points. This procedure is based on the optimization by a genetic algorithm of a vector which represents only a time intervals in order to find the best objective function. Two examples illustrated with numerical details for a two-link planar arm and six-link manipulators, for the former, comparisons with an alternative optimization solver are exposed.

Keywords: Splines cubic, Imposed task, Trajectory planning, Genetic algorithms.

1. Introduction

The use of robot manipulators in the industry is often devoted to repetitive tasks that execute at a sustained pace. It is, therefore, natural that the productivity of industrial processes closely dependent on the way to use these robots. In this case, the rapidity of the robot can favour an efficiency factor and considerable economic gain.

The choice of the quality of the robot's movements appears as a necessity, particularly during rapid transfer of objects, so for example, sorting or packing operations on production lines. Moreover, there are several possibilities for the implementation of the same task; it would be wise to take advantage of the multitude of choices to opt for the least expensive solution in terms of a predefined criterion. One way to achieve this is to formulate the problem as an optimization problem.

The work presented in this paper is interested in imposed task, in this kind of trajectory the end of the chain tool influences on its environment without interruption and on a desired track. It is necessary in this case to specify the trajectory of the effector in

the operational space. This is the case, for example, cutting work or the continuous welding.

Several research projects were dedicated to the study the trajectory planning problems of robot manipulators in the imposed tasks, we cite in this context the papers of [1] where the authors proposed a fast and unified approach based on particle swarm optimization (PSO) with K-means clustering to solve the near optimal solution of a minimum-jerk joint trajectory constrained by a fixed traverse time of robot manipulator, the cubic splines were used to interpolate between the nodes of the trajectory in an imposed tasks. In [2] used sequential unconstrained minimization techniques (SUMT) to do optimum trajectory planning of a STANFORD manipulator, the cost functions used in optimal trajectory planning are minimum traveling time, the minimum mechanical energy of the actuators and minimum penalty for obstacle avoidance. The SQP method and initial value finding algorithm are used in the works of [3, 4]. The jerk minimization is applied when the objective function of the optimization problem and the traveling time are fixed, their results show the main disadvantage of SQP

technique as it finds a global minimum only if the suitable initial value is set. In [5] proposes a minimum-time trajectory of robot arm which uses a clamped cubic splines for modelling the trajectory and Harmony Search (HS) method for solving the optimization problem, minimum traveling time trajectory planning and considering joint kinematic constraints are essential in optimal trajectory planning problems, this is because these constraints are typically concerned about industrial productivity. In [6, 7] the authors has been described an experimental validation of the minimum time-jerk trajectory planning algorithm, the trajectories have been implemented on Cartesian 3-axes manipulator equipped with a piezoelectric accelerometer, the obtained experimental results have been discussed by considering the measure of the acceleration (directly related to the vibration induced on the mechanism) as the comparison parameter. In [8] a semi-infinite optimization is proposed; the minimum-time trajectory is planned including the kinodynamic constraints where a hybrid technique using genetic algorithms is put forward. In [9] the authors developed an approach based on fuzzy genetic algorithm using real coding and elitism approach to treat the problem of trajectory planning of robotics arm along specified tasks to minimize time-jerk by considering the kinematic constraints, this same problem is treated by the authors of [10] where an approach based on vector of time intervals is developed, the results of minimum time-jerk trajectory are satisfactory to solve the problem between high production efficiency and low structure vibration of robotic arms.

The aim of this study is to determine a simple and efficient approach to solve the problem of the trajectories planning for robot manipulators along the imposed tasks the advantages of this approaches are: First its flexibility to deal with the problems of different robot manipulators with a large number of degrees of freedom and complex imposed spots (many intermediate points). The treatment of the constraints (geometric, kinematic and dynamic) is done by order of difficulty and with a systematic way to reduce the proportion of the rejects during the random selection. By using parametric functions, it is possible to stabilize the trajectory (stress mechanical stresses) and at the same time to guarantee continuity up to third order (Jerk). The different problems studied have shown, on the one hand, the versatility of this approach and, on the other hand, its effectiveness in terms of the quality of results by comparing it with other techniques proposed in [4, 8, 11].

This paper is organized as follows: section 2 presents the problem of the trajectory planning. Section 3 presents a kinematic and dynamic equation of robot manipulators used in trajectory planning. Section 4 presents the parametric functions used to model the trajectories. To make this method complete and usable, we can see in section 5 an appropriate optimization technique. In Section 6, two numerical examples i.e. the two-link planar robot SCARA IBM7535 B04 and robot with the six degrees of freedom (6R) are presented to prove the effectiveness of the proposed approach. Finally, the work is concluded in Section 7.

2. Problem statements

The planar robot with two degrees of freedom rotoïde and the robot manipulator with six degrees of freedom are considered, the task is to move the robots in the workspace along a specified path while minimizing an objective function. The dynamic model of the robots is derived using Lagrange's energy function and Euler-Lagrange's equations. The equations of motion include the inertia terms of the actuators and friction forces.

The objective function presents the time interval h_i between two via-knots of the trajectory and the optimization problem is defined as follows:

Minimize:

$$F_{obj} = \sum_{i=1}^{n-1} h_i \quad (1)$$

The subjects are:

- Joint velocities: $|V_{ji}(t)| \leq V_j^{max}$ for $j=1, \dots, n$ and $i=1, \dots, m-1$
- Joint accelerations: $|W_{ji}(t)| \leq W_j^{max}$ for $j=1, \dots, n$ and $i=1, \dots, m-1$
- Joint jerks: $|J_{ji}(t)| \leq J_j^{max}$ for $j=1, \dots, n$ and $i=1, \dots, m-1$
- Joint Cartesian velocity: $|cv_{ji}(t)| \leq cv_j^{max}$ for $j=1, \dots, n$ and $i=1, \dots, m-1$
- Joint angular velocity: $|av_{ji}(t)| \leq av_j^{max}$ for $j=1, \dots, n$ and $i=1, \dots, m-1$

Where V_j^{max} , W_j^{max} , J_j^{max} , cv_j^{max} and av_j^{max} denote the limit kinematics performances of velocity, acceleration, jerk, Cartesian velocity and angular velocity respectively of the j -th joint deduced from technological and design data.

3. Dynamic model

The dynamic equations for conservative systems in terms of the Lagrangian function are given by:

$$\tau_i = \frac{d}{dt} \left(\frac{\delta L}{\delta \dot{\phi}_i} \right) - \frac{\delta L}{\delta q_i} \quad (2)$$

The Lagrangian function L is defined as the difference between the kinetic energy K and the potential energy P of the system:

$$L = K - P \quad (3)$$

Where:

q_i : generalizes coordinates of the system (Q_i for rotational joints and d_i for prismatic joints).

\dot{q}_i : generalized velocities (angular velocity $\dot{\theta}_i$ for rotational joints and linear velocity \dot{d}_i for prismatic joints).

τ_i : generalized forces.

The generalized forces τ_i are given by [2] as:

$$\tau_i = \sum_{j=1}^n D_{ij} \ddot{q}_j + \sum_{i=1}^n \sum_{k=1}^j C_{ijk} \dot{q}_j \dot{q}_k + G_i \quad (4)$$

Where D_{ij} is the inertial system matrix, C_{ijk} is the Coriolis and centripetal forces matrix, G_i is the gravity loading vector and \ddot{q}_i is generalized accelerations.

4. Formulation of the trajectory with Cubic Splines

The cubic spline is a piecewise 3rd degree polynomial function. The velocities and accelerations of the initial and terminal conditions (v_1, v_m, a_1 and a_m) are specified to be zero. These conditions cause two equations of the spline algorithm becoming zero and the path pattern cannot be solved. Therefore, two extra knots (position values at time t_2 and t_m) are added and their position values are not specified. Let $Q_i(t)$ be the cubic polynomial for the i -joint in the interval time $[t_i, t_{i+1}]$. The second derivative of $Q_i(t)$ is a linear interpolation and can be written as [2]:

$$W_i(t) = \frac{t_{i+1}-t}{h_i} W_i(t_i) + \frac{t-t_i}{h_i} W_i(t_{i+1}) \quad (5)$$

Where: $h_i = t_{i+1} - t_i$

Integrating Eq. (5) for the given points $Q_i(t_i) = q_i$ and $Q_i(t_{i+1}) = q_{i+1}$, the following interpolation functions are obtained:

$$V_i(t) = -\frac{W_i(t_i)}{2h_i} (t_{i+1}-t)^2 + \frac{W_i(t_{i+1})}{2h_i} (t-t_i)^2 + \left[\frac{q_{i+1}}{h_i} - \frac{h_i W_i(t_{i+1})}{6} \right] - \left[\frac{q_i}{h_i} - \frac{h_i W_i(t_i)}{6} \right] \quad (6)$$

And:

$$Q_i(t) = \frac{W_i(t_i)}{6h_i} (t_{i+1}-t)^3 + \frac{W_i(t_{i+1})}{6h_i} (t-t_i)^3 + \left[\frac{q_{i+1}}{h_i} - \frac{h_i W_i(t_{i+1})}{6} \right] (t-t_i) + \left[\frac{q_i}{h_i} - \frac{h_i W_i(t_i)}{6} \right] (t_{i+1}-t) \quad (7)$$

Using the continuity conditions on velocities and accelerations, a system of $m-2$ linear equations solving for $m-2$ unknowns $W_i(t_i), i=2,3, \dots, m-1$ is obtained as:

$$A [W_2(t_2) \quad W_3(t_3) \dots W_{m-1}(t_{m-1})]^T = B \quad (8)$$

In (8), the matrix A is non-singular matrix and entries of the matrix B are changed for each joint. Then, the extra knots position values can obtain from:

$$q_2 = q_1 + h_1 v_1 + \frac{h_1^2}{3} a_1 + \frac{h_1^2}{6} W_2(t_2) \quad (9)$$

$$q_{m-1} = q_m - h_{m-1} v_m + \frac{h_{m-1}^2}{3} a_m + \frac{h_{m-1}^2}{6} W_{m-1}(t_{m-1}) \quad (10)$$

5. Optimization procedures

5.1 Initialization and route of generation

Let h be defined as the vector of design variables $h_i = [h_1, h_2, \dots, h_{m-1}]$. To initialize the optimization process it is considered that:

$$h^{(0)} = \max_{j=1, \dots, m-1} \left[\left(\frac{|q_{j2} - q_{j1}|}{V_j^{max}} \right), \left(\frac{|V_{j2} - V_{j1}|}{W_j^{max}} \right), \dots, \left(\frac{|W_{jm} - W_{j,m-1}|}{J_j^{max}} \right) \right] \quad (11)$$

As two extra knots are needed they are initially taken as:

$$q_{j2} = (q_{j1} + q_{j3}) / 2 \text{ and } q_{j,m-1} = (q_{j,m-2} + q_{jm}) / 2 \quad (12)$$

During the optimization process the intermediate knots and the end point will generate only horizontally as seen in Fig.1, consequently the

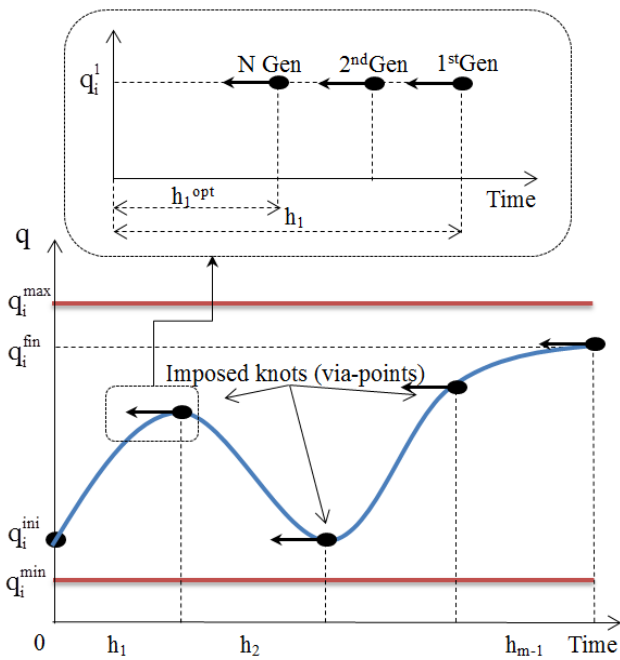


Figure. 1 Representation of the Cubic Spline Trajectory with the horizontal movement of the intermediate knots and the end point after generation

trajectory changes and moves also horizontally by minimizing the objective function and obtaining the best vector h^{opt} .

5.2 Genetic algorithms

In order to make this approach usable, it is required to use an appropriate optimization technique to solve the objective function presented previously in Eq. (1), we propose to use a genetic algorithms GAs (Fig. 2).

GAs are stochastic optimization algorithms based on the mechanisms of natural selection and genetics. Their operation is extremely simple. We start with a population of arbitrarily chosen potential solutions (chromosomes). Their relative fitness (fitness) is evaluated. On the basis of these performances, a new population of potential solutions is created using simple evolutionary operators: Selection rules select the individuals, called parents, which contribute to the population at the next generation. Crossover rules the creation of new individuals or children according to a very simple process for the next generation. Mutation rule is to randomly alter with a certain probability the value of a component of the individual parent to form children.

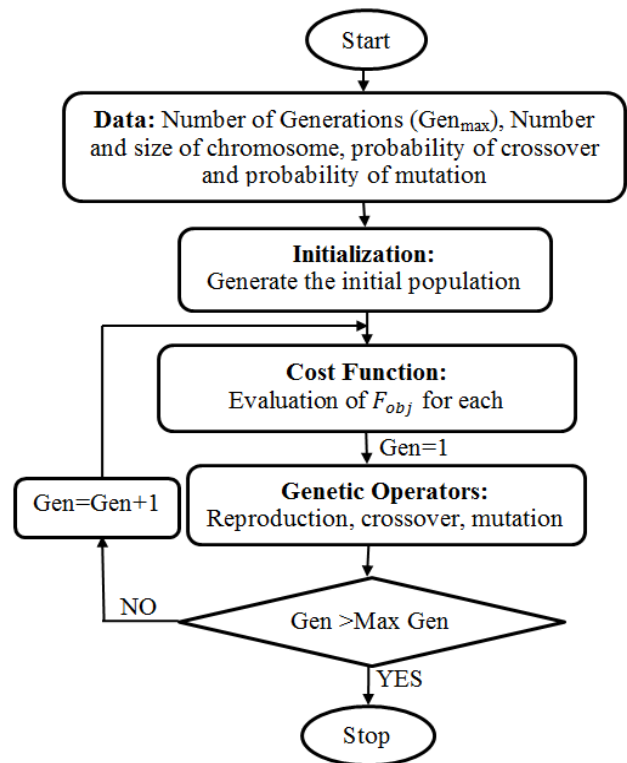


Figure. 2 Flowchart of the genetic algorithms

This cycle is repeated until a satisfactory solution is found. There are four main points that make the fundamental difference between these algorithms and other methods:

- GAs Use a coding of the parameters, not the parameters themselves.
- GAs Work on a population of points, instead of a single point.
- GAs Use only the values of the objective function studied, not its derivative, or another auxiliary knowledge.
- GAs select the next population Using probabilistic computations, not deterministic ones.

6 Simulation results and comparison

The simulation results by cubic spline compare our procedure with two different techniques. For this, two examples are presented in this section; the first one concerns a planar 2R robot, it has been mainly suggested to compare the results obtained with our procedure with those obtained with the genetic /interval algorithm used in [8]. The second example considers an actual manipulator 6R degrees of freedom, the results obtained are compared with those obtained with HHSA method used in [10]. The objective function taken in these simulations is

presented by the Eq. (1). The purpose is to prove that the proposed procedure can be fruitfully used with non-trivial problems. Noting that, during the optimization process, the probability of crossover is fixed to 65% and the probability of mutation to 4%.

6.1 Example 1: SCARA Two d.o.f planar robot

The robot considered in this example is planar with two degrees of freedom, the task is to plan under torque and velocities constraints a trajectory in Cartesian path; the parameters inertial and dynamic constraints are summarized in Table 1 and the limit kinematics performance are the linear and angular velocities of the end-effector, which are defined in (13) and (14) successively. The robot is asked to move from a given via points reported in Table 2. Moreover, we consider the arm at rest in the initial and final positions ($\phi_1^{init} = \phi_1^{fin} = 0$; $\phi_2^{init} = \phi_2^{fin} = 0$).

$$\|cv\| = \sqrt{l_1^2 \dot{\phi}_1^2 + l_2^2 (\dot{\phi}_1 + \dot{\phi}_2)^2 + 2l_1 l_2 \dot{\phi}_1 (\dot{\phi}_1 + \dot{\phi}_2) \cos(q_2)} \leq 0.7 \text{ m/s} \quad (13)$$

$$av_z = \dot{\phi}_1 + \dot{\phi}_2 \leq 1.5 \text{ rd/s} \quad (14)$$

Table 1. Parameters inertial and dynamic constraints of 2R robot.

	Link 1	Link 2
M (kg)	15.0	7.0
L (m)	1.0	0.50
Torques (N. m)	260.0	50.0

Table2. Values of the via-point of the trajectory of 2R robot

Joint	1	2
1	0.000	-1.5708
2	Extra knots	
3	0.1253	-1.6804
4	0.2517	-1.7594
5	0.3789	-1.8074
6	0.5054	-1.8235
7	0.5837	-1.7087
8	0.6119	-1.4581
9	0.4263	-1.1040
10	0.3903	-1.1124
11	Extra knots	
12	0.3526	-1.1152

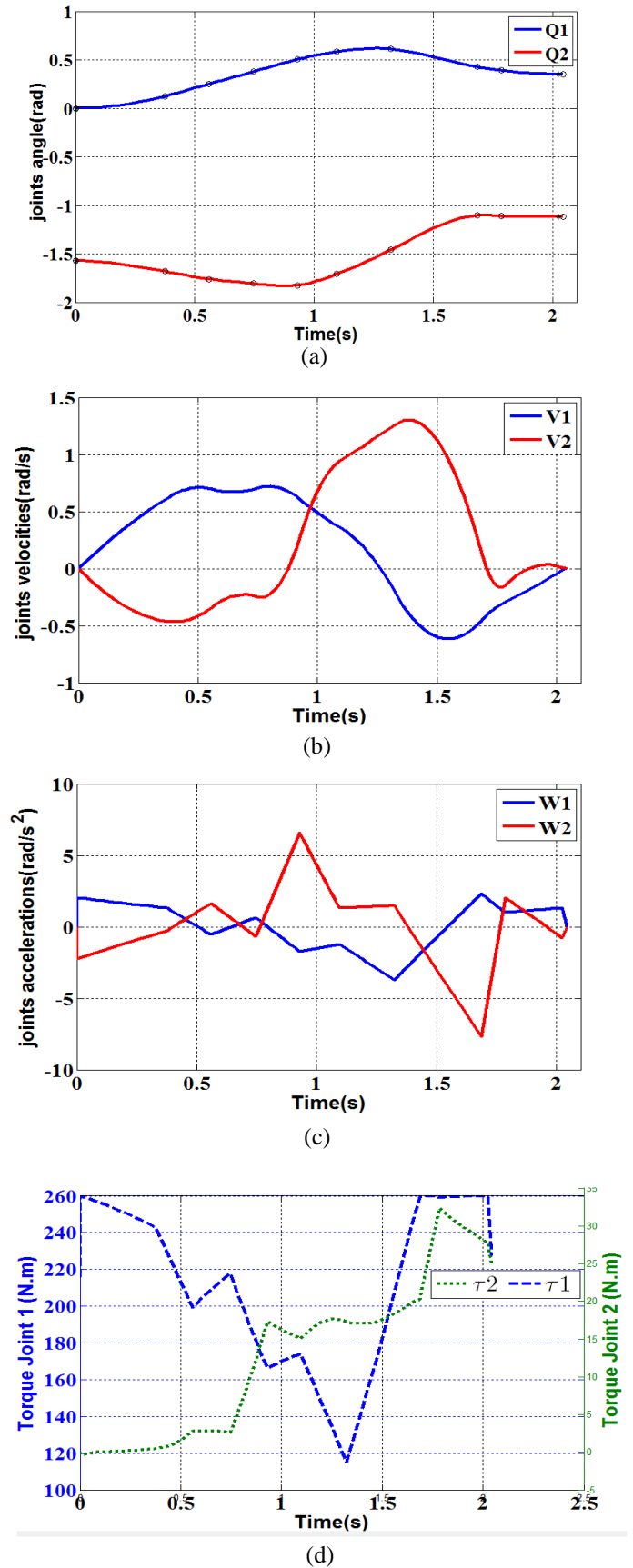
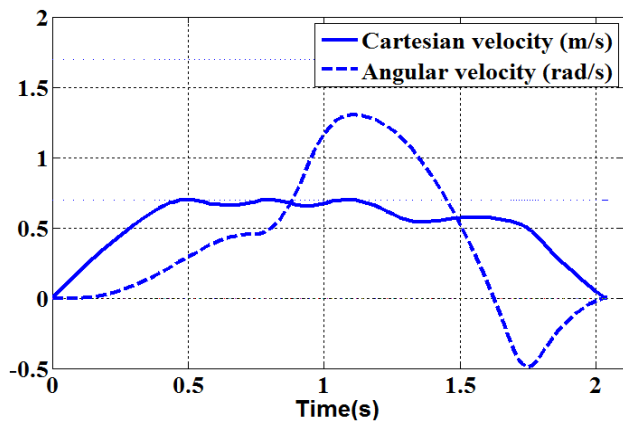
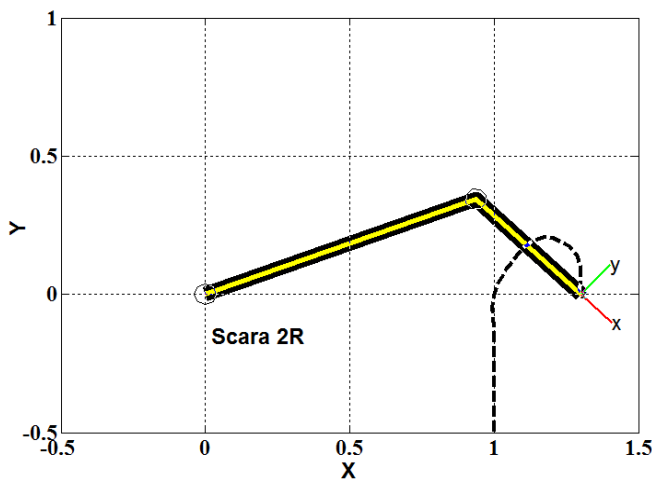


Figure. 3 Results of SCARA 2R robot: (a) joints angles, (b) joints velocities, (c) joints accelerations and (d) torques joints



(a)



(b)

Figure. 4 Cartesian and Angular velocities of End-effector (a) and optimal trajectory in Cartesian space (b) of SCARA 2R robot

The profiles results, of each joint minimum time of smooth trajectory modelled with the cubic spline function and their derivatives, joints velocities, joints accelerations and torques are shown in Fig. 3, the trajectory was given by ten knots positions and two extra knots, to represent the trajectory and solve these extra knots we use both Eq. (9) and Eq. (10), the circles and the crosses presented in joints angle indicate the knot positions and the two dummy knots respectively, the dynamic constraints have been respected and the joint 1 torque presents a segment of broad time saturation.

In addition Fig. 4-a shows the Cartesian and Angular velocities of End-effector, the kinematic constraints applied in the end-effector were respected. It should be mentioned that this approach treats all the constraints in a sequential way and by order of difficulty; the kinematics and dynamics constraints by the inclusion of a full manipulator dynamic model and any trajectory which would

violate one of the kinodynamic constraints will be automatically rejected. The Fig. 4-b shows the optimal trajectory in Cartesian space (XY) of Scara 2R robot which performs an imposed task.

The minimization evaluated by our algorithm gives a vector of time intervals $h_i = [0.0031, 0.3724, 0.1838, 0.1869, 0.1835, 0.1656, 0.2275, 0.3629, 0.0995, 0.2304, 0.0200]$ and the estimated global minimum traveling time is $\sum_{i=1}^{11} h_i = 2.0356 \text{sec}$, the solution found by genetic/interval algorithm used in [8] gives $\sum_{i=1}^{11} h_i = 2.05009 \text{sec}$.

6.2 Example 2: a six d.o.f robot

In this example, we will treat a robot manipulator with six degrees of freedom its structure contains six articulations pivot as studied in [11]. The knot positions (via-points) and kinematics constraints of the 6R robot manipulator joints system are shown in Table 3 and Table 4 respectively.

Each smooth joint of cubic spline trajectories including their derivatives of velocities, accelerations and jerk are illustrated in Fig. 5. The optimal trajectory with minimum time was given by four knots positions and five time intervals for all the six joints, by applying our approach the kinematics limitation constraints considered in the optimization process are satisfied for the 6 DOF and in a sequential way, the velocities constraints, the accelerations constraints and jerk constraints Fig. 6, shows the simulation of robot manipulator 6R in 3D Cartesian coordinates (XYZ), where the optimal trajectory found in this imposed tasks is described between the 1-3-4-6 via-points. Noting that, during the optimization process we have fixed the generation number to 80, the results histories of the generation are shown in the bottom of Fig. 7, these results approve that our approach converges rapidly toward the optimal solution.

Table3. Knot positions of each robot manipulator joint

Joint	Knot (degrees)					
	1	2	3	4	5	6
1	-10	Extra knot	60	20	Extra knot	55
2	20		50	120		35
3	15		100	-10		30
4	150		100	40		10
5	30		110	90		70
6	120		60	100		25

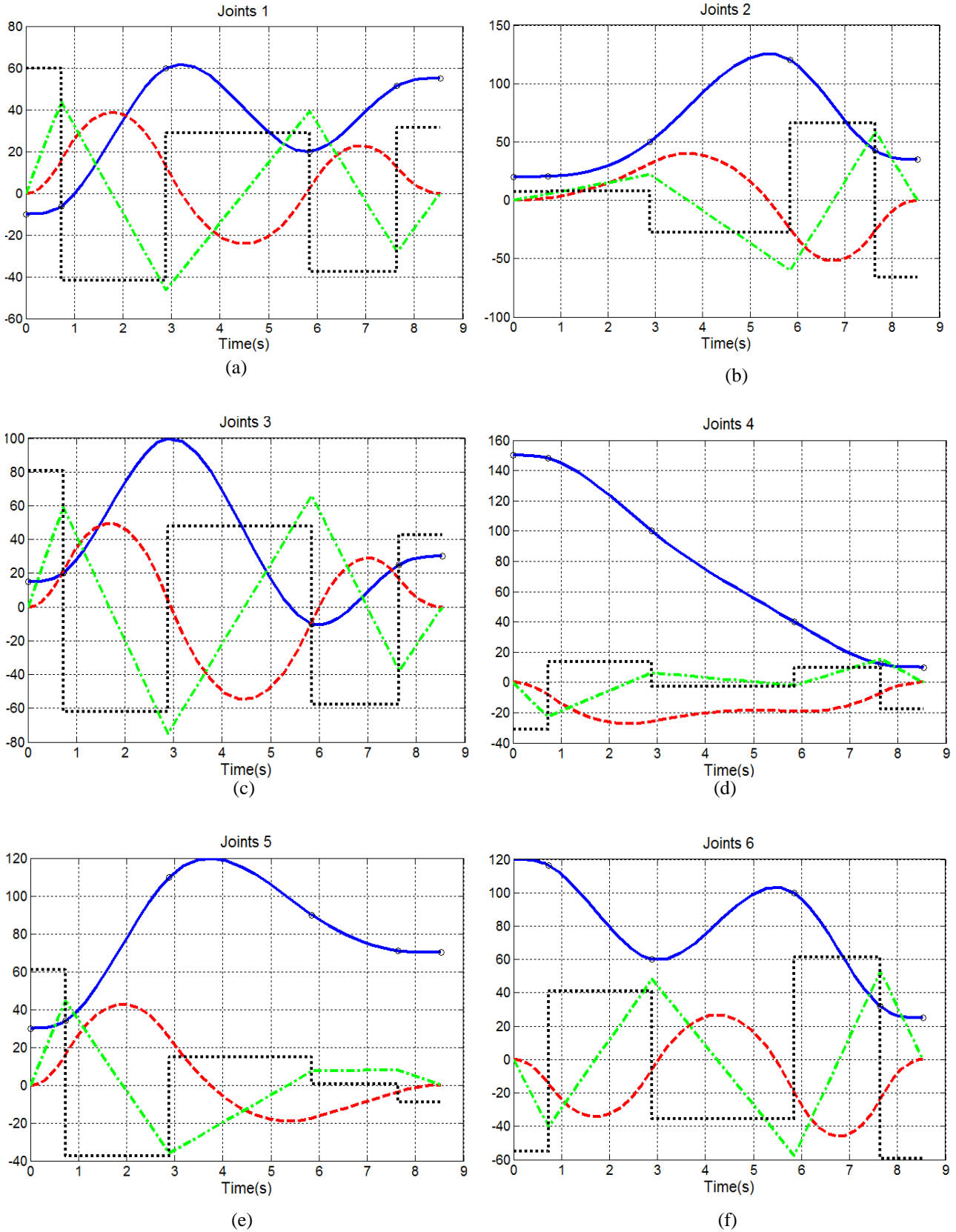


Figure. 5 Results of robot manipulator 6R with (a) to (f) represent respectively joints 1 to 6 under kinematics constraints Where: $Q(\text{deg})$ $V(\text{deg/s})$ $W(\text{deg/s}^2)$ $J(\text{deg/s}^3)$ represent respectively position, velocity, acceleration and jerk. The circles indicate the knot positions and the crosses indicate the two dummy knots

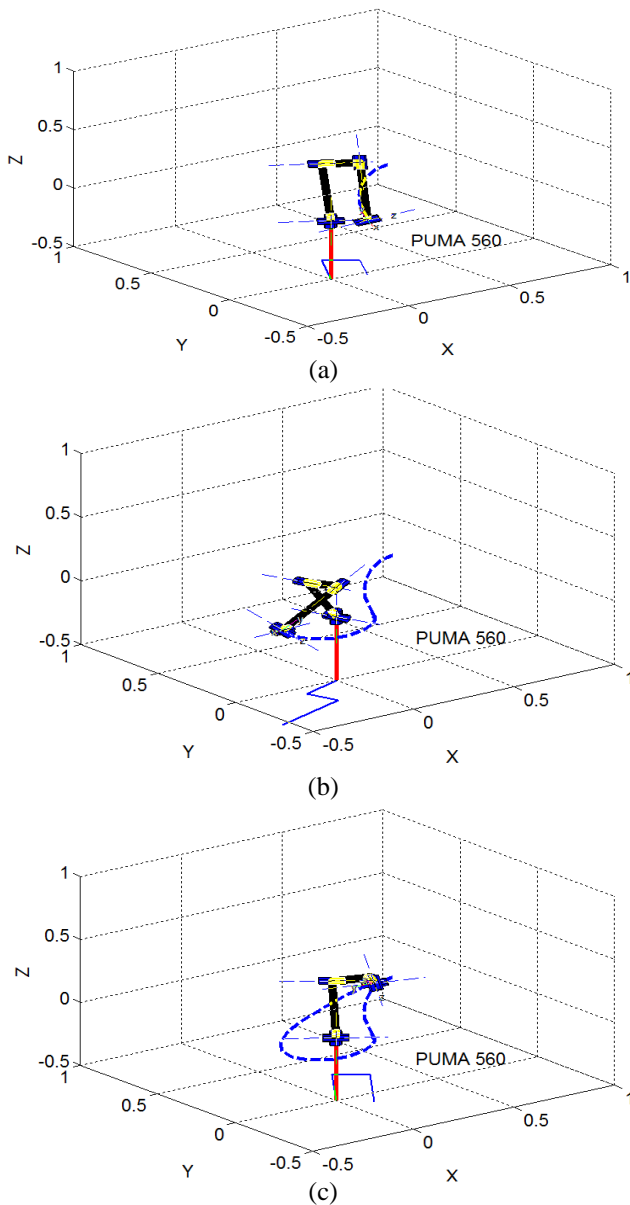


Figure. 6 Trajectory results of robot manipulator 6R: (a) among knot 1 and 3, (b) among knot 1 and 4 and (c) among knot 1 and 6 with kinematics constraints

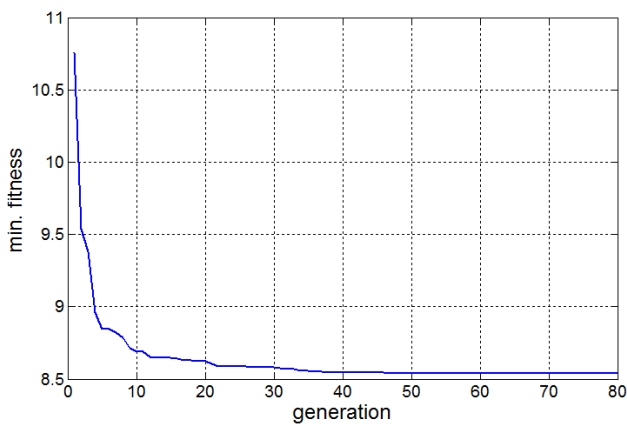


Figure. 7 Histories generation results of robot manipulator 6R

Table4. Kinematics constraints of each of robot joint

Joint	1	2	3	4	5	6
Velocities (deg/s)	100	95	100	150	130	110
Acceleration (deg/s²)	60	60	75	70	90	80
Jerk (deg/s³)	60	66	85	70	75	70

The minimization of objective function Eq. (1) evaluated by our algorithm gives a vector of time intervals $h_i = [0.7289, 2.1588, 2.9611, 1.7954, 0.8856]$ and equal to execution time $\sum_{i=1}^5 h_i = 8.5298 \text{ sec}$.

The minimum-time is the most significant data to compare the techniques used to find the optimal smooth trajectory planning of robots manipulator along specified tasks. The results show that the approach described in this paper can obtain the best solution compared with the previous related literatures including [4] which used a Sequential Quadratic Programming (SQP) techniques which obtained the objective function equal to $\sum_{i=1}^5 h_i = 85726 \text{ sec}$, and [11] which used a Hybrid Harmony Search Algorithm (HNSA) which obtained the objective function equal to $\sum_{i=1}^5 h_i = 8.5310 \text{ sec}$.

7. Conclusion

In this paper, a procedure of study the trajectory planning problem is suggested for robot manipulators along specified paths, the trajectories uses cubic splines functions which guarantee the smoothing of the trajectory and at the same time guarantee the continuity of velocities, accelerations and jerk with an interpolation of the given via points (plus two dummy knots).

This procedure is based on the optimization a vector of time intervals and it makes possible to treat all the constraints of this problem in a sequential way and by order of difficulty (the geometrical, kinematics and torque constraints) by the inclusion of a full manipulator dynamic model. Moreover, the time efficiency guaranteed by this approach permits to solve the problems with a relatively large number of via points and degrees of freedom of robot manipulators; as well, our proposed method converges quickly and has a faster calculation speed. In consequence, the proposed optimization strategy of the trajectory along imposed tasks is feasible.

This work opens the door for further investigations such as using the B-Spline functions or Non-Uniform Rational B-Spline (NURBS) functions, taking into account obstacles in a workspace and a minimization of multi-objective functions for robots arms in imposed tasks, so as to evaluate the applicability the proposed method and its results in other situations.

References

- [1] H. Lin, "A fast and unified method to find a minimum-jerk robot joint trajectory using particle swarm optimization", *Journal of Intelligent & Robotic Systems*, Vol.75, pp379-392, 2014.
- [2] S.F.P. Saramago and V. Steffen Jr, "Trajectory modeling of robot manipulators in the presence of obstacles", *Journal of Optimization Theory and Applications*, Vol.110, No.1, pp.17-34, 2001.
- [3] A. Gasparetto and V. Zanotto, "A new method for smooth trajectory planning of robot manipulators", *Mechanism and Machine Theory* Vol.42, pp.455-471, 2007.
- [4] A. Gasparetto and V. Zanotto, "A technique for time-jerk optimal planning of robot trajectories." *Robotics and Computer-Integrated Manufacturing*, Vol. 24, pp.415-426, 2008.
- [5] P. Tangpattanakul and P. Artrit, "Minimum-Time Trajectory of Robot Manipulator Using Harmony Search Algorithm", In: *6th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*, Chonburi, Thailand, 2009.
- [6] V. Zanotto, A. Gasparetto, A. Lanzutti, P. Boscariol, and R. Vidoni, "Experimental validation of minimum time-jerk algorithms for industrial robots", *Journal of Intelligent & Robotic Systems*, Vol.64, pp197-219, 2011.
- [7] A. Gasparetto, A. Lanzutti, R. Vidoni, and V. Zanotto, "Experimental validation and comparative analysts if time-jerk algorithms for trajectory planninh", *Robotics and Computer-Integrated Manufacturing*, Vol.28, pp164-181, 2012.
- [8] C. Guarino Lo Bianco, and A. Piazzzi, "A semi-infinite optimization approach to optimal spline trajectory planning of mechanical manipulators." in: M.A. Goberna, M.A. Lopez (Eds.), *Semi-infinite Programming: Recent Advances*, Springer-Verlag, Berlin, pp.271-297, 2001.
- [9] M. Cong, X. Xu, and P. Xu, "Time-jerk synthetic optimal trajectory planning of robot based on fuzzy genetic algorithm", *Int J Intelligent Systems Technologies and Applications*, Vol.8, pp1-4, 2010.
- [10] N. Bendali and M. Ouali, "On-line trajectory planning of time-jerk optimal for robotic arms", *Mediterranean Journal of Modeling and Simulation*, Vol.6, pp.34-44, 2016.
- [11] P. Tangpattanakul, A. Meesomboon, and P. Artrit, "Optimal Trajectory of Robot Manipulator Using Harmony Search Algorithm, Recent Advances in Harmony Search Algorithm", *the series Studies in Computational Intelligence*, Spriger-Verlag, Berlin, Vol.270, pp.23-36, 2010.