iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 6 **Issue:** III **Month of publication:** March 2018

DOI: http://doi.org/10.22214/ijraset.2018.3722

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 6 Issue III, March 2018- Available at www.ijraset.com

Optimization of Joint Space Trajectory for Minimization of Traversal Time with Specified Actuator Constraints

Adwait Verulkar[1], Mayur Gangwal[2], Santosh Joshi[3]
[1, 2, 3] Department of Mechanical Engineering, Vishwakarma Institute of Technology, Pune, India

***Abstract:** This paper deals with optimization of the joint space trajectory for any general manipulator. The optimization has been done with the primary design objective of minimization of traversal time. Multi-segment blended trajectory for point to point motion with via points was used for calculation of path. The time required for the manipulator to complete a particular segment was calculated based on the actuator inputs and path constraints specified by the user. There is a possibility that the user may specify actuator parameters such that the maximum possible acceleration cannot attain the maximum possible velocity within the constraints of the path. For such a case, a bisection method algorithm was applied to find the best possible and attainable velocity. Also the possibility of successively increasing or decreasing via points has been covered. Inherently a multi-segment linear trajectory with parabolic blends is an approximate technique. Therefore a correlation between the positional error and specified velocity and acceleration has also been established. A MATLAB code has been written for implementation of the above methodology. Plots of joint displacements, velocities and accelerations have been plotted against time. The program also calculates the minimum possible execution time for the calculated path.*
***Keywords:** Optimization, Joint Space, Trajectory Planning, Manipulator, Traversal Time, Blended Trajectory, Via Points*

I. INTRODUCTION

Optimization of manipulator trajectory in industrial automation is one of the primary concerns for minimizing cycle time. Luh, J. Y. S., Walker, M. W [1] have shown that minimization of traversal time can be achieved by interpolating the set of via points using straight line segments connected by smooth arcs. This problem can be expressed as a set of inequalities quantifying the physical constraints imposed by manipulator hardware. Linear programming can then be used to solve such a set of inequalities. These physical constraints can be in the form of a certain torque that produced at a particular joint [2], [3] or it can be expressed as a limit on the acceleration and velocity that can be achieved [4]. However, owing to the complexity of the task and manual computational limitations, research in this direction has been towards developing algorithms and computational techniques which optimize trajectory based on certain constraints. Manipulator dynamics is also a crucial deciding parameter which restricts the link acceleration and velocity [5], [6]. Zvi Shiller et. al. [5] have proven that the optimal trajectory is extremal in the acceleration along the path at all times except at singular points and arcs where it maximizes the feasible velocity along the path. This algorithm is robust to path variations and does not fail at singular points. The feasible range for the acceleration and velocity was found using actuator constraints in the dynamic model of the manipulator. J. Y. S. Luh and C. S. Lin [7] have presented a method of obtaining a time history of velocities and accelerations along the specified manipulator path to obtain a minimum traveling time, under the constraints on linear and angular velocities and accelerations. To solve the myriad of nonlinear inequality constraints from physical limitations, the "method of approximate programming (MAP)" is applied. Also, a "direct approximate programming algorithm (DAPA)" was developed and found to converge to optimum feasible solution for the trajectory planning problem. Apart from the actual optimization of, the constraint of “jerk” or a sudden acceleration must also be considered [8], [9]. For the successful execution of any task, the manipulator is required to move along a particular pre-defined geometric path [10]. This geometric path is in the so called “Cartesian Space” of the robot. This geometric path can be converted to a suitable set of joint variables in the “Joint Space” of manipulator, using the inverse kinematic model of that particular manipulator. The acceleration and velocities obtained such that it satisfies the aforementioned conditions can be converted into the joint acceleration and velocities using the dynamic model. The prime goal of trajectory planning is to construct the required motion in the form of a time sequence of locations where the joints or end effector should be located along with the calculation of velocities and accelerations at every point in the time sequence. The former approach is known as ‘joint space trajectory planning’ and is the subject of this work. This set of data is the input to the control system of the manipulator. The trajectory is usually specified by a number of points known as ‘via points’ and the path is constructed by either exact or approximate interpolation through these via points. It is very crucial from a practical standpoint that the path generated has at least tangent continuity throughout. This prevents sudden jerks, mechanical vibrations, wear and tear of mechanical systems and ensures practically achievable gradual starts and stops.

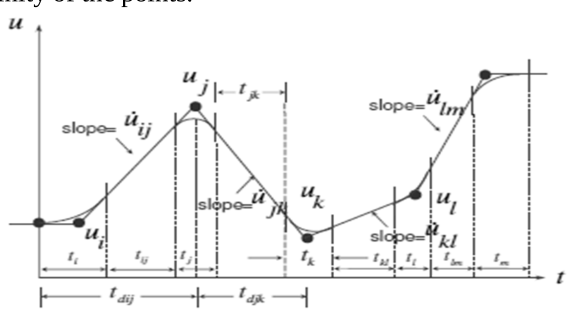
II. APPROACH AND ANAYTICAL TREATMENT

A. *Selection of Joint Space Technique*

There are several techniques that can be used to construct a path through the specified via points. However, as the fastest possible path between any two points is the straight line joining them, a linear trajectory with parabolic blends becomes the obvious choice satisfying the tangent continuity condition [11], [12]. The only limitation of this approach is that, this technique gives an approximate interpolation through via points rather than an exact one. Only the initial and final positions are successfully met, whereas for other positions the end effector motion achieves the closest possible position, for the specified acceleration and velocity, in the vicinity of via point. This limitation however can be easily circumvented by giving a position with the error in position taken into consideration. There exists a definite correlation between the specified velocity, acceleration and the error associated with the position which has been stated in the subsequent sections.

B. *Analytical Treatment*

As shown in Fig. 1, a first order differentiable path for joint variable, ‘q’, is computed with a linear segment connecting via points and parabolic blends are added in the vicinity of the points.



Fig, 1: Blended trajectory for several segments

To maintain the tangency condition at intermediate points, a continuous parabola for the two-blended segments at via point has been used. This is the reason why the polynomial *q(t)* does pass exactly through the via point and renders the point as a virtual or pseudo via point. Naturally, the path points, travel time between successive path points and constant blend acceleration are the prerequisites for computation of trajectory. This leaves only the blend duration at each path point as a parameter to be calculated.

The following nomenclature followed in [13] has been used in this analysis

q^j– value of joint variable q corresponding to path point j

$\ddot{q}$–Magnitude of constant blend acceleration at point j

T_{jl}– total travel time between points j and l

t_{jl}– travel time for linear segment between j and l

t_j– duration of blend around path point j

$\dot{q}^{jl}$ – constant joint velocity between points j and l

For the via points, the blend duration t_{jl} near via point l is computed from specified acceleration at the via point q^l and the constant linear velocities in two segments $\dot{q}^{jl}$ and $\dot{q}^{lm}$.

$$t_l = \frac{\dot{q}^{jl} - \dot{q}^{lm}}{\ddot{q}^l} \tag{1}$$

where,

$$\ddot{q}^l = sign(\dot{q}^{jl} - \dot{q}^{jl})\,|\,\ddot{q}^l\,| \tag{2}$$

The linear velocity in segment *jl* is given by

$$\dot{q}^{jl} = \frac{q^l - q^j}{T_{jl}} \tag{3}$$

he duration of the linear segment t_{jl}, assuming parabolic blend symmetric around the path point, is

$$t_{jl} = T_{j;} - \tfrac{1}{2}t_j - \tfrac{1}{2}t_l \tag{4}$$

To compute the blend duration at the initial point $j = 1$, the fact that the joint velocity at the end of the blend is same as the velocity of the linear segment 12 (t_{12}), is used. This gives

$$\ddot{q}^1 t_1 = \frac{q^2 - q^1}{T_{12} - \tfrac{1}{2}t_1} \tag{5}$$

where,
$$\ddot{q}^1 = sign(\dot{q}^2 - \dot{q}^1)\,|\,\ddot{q}^1\,| \tag{6}$$

From eq. (5) the blend time t_1 at the initial point and using t_1, the linear segment velocity and duration, $\dot{q}^{12}$ and t_{12} are computed as

$$t_1 = T_{12} - \sqrt{T_{12}^2 - \frac{2(q^2 - q^1)}{\ddot{q}^1}} \tag{7}$$

$$t_{12} = T_{12} - \tfrac{1}{2}t_2 - t_1 \tag{8}$$

and,

$$\dot{q}^{12} = \frac{q^2 - q^1}{T_{12} - \tfrac{1}{2}t_1} \tag{9}$$

Finally, the blend duration at the goal point is determined. Similar to the starting point, the velocity continuity constraint, in the middle segments connecting the path points (k – 1) and k, gives

$$\ddot{q}^k t_k = \frac{q^k - q^{k-1}}{T_{(k-1)k} - \tfrac{1}{2}t_1} \tag{10}$$

where,

$$\ddot{q}^k = sign(\dot{q}^{k-1} - \dot{q}^k)\,|\,\ddot{q}^k\,| \tag{11}$$

The solution for blend duration, linear velocity, and its duration are

$$t_k = T_{(k-1)k} - \sqrt{T_{(k-1)k}^2 - \frac{2(q^k - q^{k-1})}{\ddot{q}^k}} \tag{12}$$

$$t_{(k-1)k} = T_{(k-1)k} - \tfrac{1}{2}t_{k-1} - t_k \tag{13}$$

$$\dot{q}^{(k-1)k} = \frac{q^k - q^{k-}}{T_{(k-1)k} - \tfrac{1}{2}t_k} \tag{14}$$

Thus, all the requisite parameters to determine the time history of joint position, velocity and acceleration are obtained using eq. stated above. It is important to note that at each via point the acceleration must be sufficiently large and should satisfy the below mentioned condition.

$$|\,\ddot{q}^c\,| \geq \frac{4\,|\,q^g - q^s\,|}{t_g^2}; \qquad and \qquad \ddot{q}^c \neq 0 \tag{15}$$

The solution will not exist if eq. (15) is not satisfied. In case of the equality the two blends meet at t_m and there is no linear or constant velocity segment and only acceleration and deceleration segments are present giving a triangular profile rather than a trapezoidal one. On the other hand, as the acceleration becomes larger and larger, the blend diminishes and the trajectory tends to linear interpolation with acceleration tending to infinity or t_b tending to zero. In our case the same value of blend acceleration has been used to make computations simpler. Also, this condition has been used as an optimizing parameter.

III. OPTIMIZATION METHODOLOGY AND ALGORITHM OF THE CODE

A. Modifications Required in The Approach

Now, with the analytical approach discussed, the methodology for optimization may be formulated. The time required to cover the distance between any two points is taken as an input in the above technique. This facilitates the calculation of t_b through the use of eq. (12). However, the problem at hand, requires this time to be calculated based on the actuator specifications. Hence we modify and rearrange eq. (14) as follows to suit our requirements.

$$T_{(k-1)k} = \frac{q^k - q^{k-1}}{\dot{q}^{(k-1)k}} \tag{16}$$

Also, rather than calculating the blend time from the total time, the blend time is independently calculated.

1) For the start and end blends the blend time is calculated by fitting a parabola with 3 boundary conditions of 2 slopes at ends of the parabola and a point continuity at either the start or end points. Thus the blend time t_b, can be calculated by the following formula.

$$t_k = \frac{\dot{q}_{kl}}{\ddot{q}_k} \tag{17}$$

2) *For intermediate blends there are 2 cases*

3) The first case is of successively increasing or decreasing path points. In this case there is no need for any blend as the path may continue with the same velocity. In this case the via points will be reached exactly.

4) The second case is of via points initially increasing then decreasing or vice versa. In this case a blend is required and the blend time is calculated as follows.

$$t_k = \frac{2\dot{q}^{kl}}{\ddot{q}^k} \tag{18}$$

Now, for calculating the straight line time history, the time during which the acceleration of the joint is zero is calculated either from eq. (4), eq. (8) or eq. (13), depending on the case.

B. Algorithm for Optimization

If the user has entered the actuator specifications such that, eq. (15) is being satisfied, then the trajectory can be computed. However, in the case of weak acceleration, the condition is not satisfied. In this case the best possible path would be the one in which the actuator is continuously accelerating and the maximum velocity is achieved at the point of inflection where two consecutive parabolas meet. There are two ways in which this velocity may be calculated, either analytically or numerically. For the analytical solution eq. (15) may be solved for the equality condition to find t_g, for every segment, thereafter one can calculate the velocity. This however becomes complex. Thus we shift towards a numerical approach, the one followed in this work. Bisection method [14] has been implemented to gradually converge onto the best possible velocity. The condition for changing the upper and lower bracketing limits are as follows.

1) If a certain straight line segment is skipped, this means that the velocity is too high. So the upper bracketing limit is changed.

2) Alternatively, if all the straight line paths are achieved, this means that there is further scope for optimization and so the lower limit is changed.

This algorithm is followed for 1000 iterations and the final achievable velocity is displayed. A detailed flowchart of the algorithm has been included in Appendix 'A'.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 6 Issue III, March 2018- Available at www.ijraset.com

IV. RESULTS AND DISCUSSIONS

In this section various cases are discussed and the output of the code has been presented. The path point inputs has been kept constant as shown in Table 1. The only change that has been done is in the velocity and acceleration.

Table. I. Input Path Points

Joint Variable	Value (radians)
q^1	1
q^2	50
q^3	25
q^4	10
q^5	20

A. Case of sufficient acceleration

In this case we take acceleration as 20 rad/s^2 and velocity as 10 rad/s. The output given by the code is as follows.

```
Total no. of points: 5
Path Update Rate (sec): 0.01
Motor Acceleration = 20
Motor Velocity = 10
Q(1)  = 1
Q(2)  = 50
Q(3)  = 25
Q(4)  = 10
Q(5)  = 20
The traversal time is 11.400000
The maximum velocity reached is 10.000000
```

Fig. 2: Output for case 1.

Also the time history plot generated by the code is as follows.
Legend:
Blue- Displacement
Green- Velocity
Red- Acceleration.

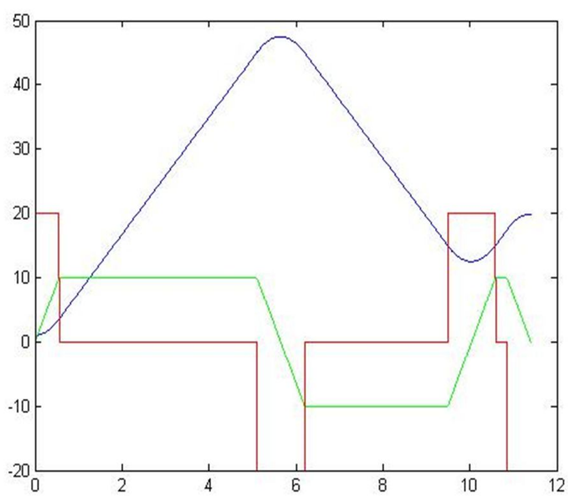


Fig. 3: Time history for case 1.

B. Case of insufficient Acceleration

In this case we take acceleration as 5 rad/s^2 and velocity as 20 rad/s. The output given by the code is as follows.

```
Total no. of points: 5
Path Update Rate (sec): 0.01
Motor Acceleration = 5
Motor Velocity = 20
Q(1) = 1
Q(2) = 50
Q(3) = 25
Q(4) = 10
Q(5) = 20
The traversal time is 22.800000
The maximum velocity reached is 5.000000
```

Fig. 4: Output for case 2.

Also the time history plot generated by the code is as follows.
Legend:
Blue- Displacement
Green- Velocity
Red- Acceleration.

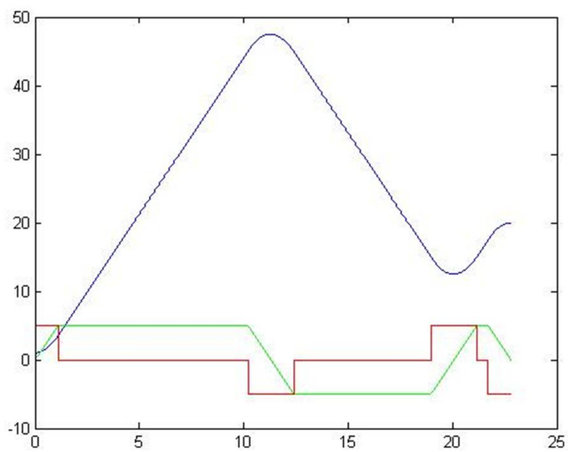


Fig. 4: Time history for case 2.

As it is evident from the graph, the last trapezoidal profile has almost reached a triangular shape. This is the required condition for the most optimized path.

C. Dependancy of positional Error on Acceleration

In this section the dependency of positional error on acceleration is shown. Only 3 path points have been considered for simplicity. This will give only 1 positional error as the start and goal points are exactly met. The path points are:

Table II: Path Points

Joint Variable	Value (radians)
q^1	0
q^2	π
q^3	$\pi/2$

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 6 Issue III, March 2018- Available at www.ijraset.com

Three values of acceleration have been considered, namely 1 rad/s^2, 3 rad/s^2, and 9 rad/s^2. The following plots were obtained.

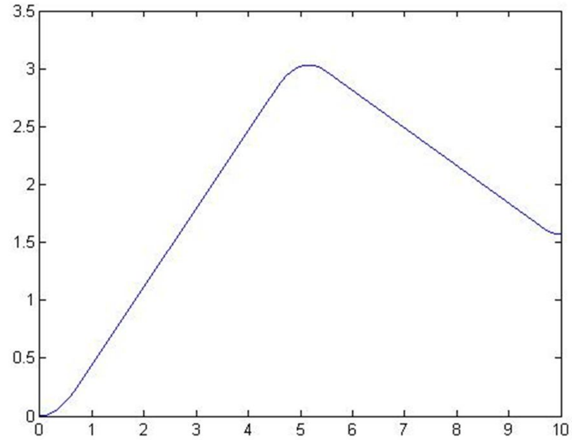


Fig. 5: Acceleration = 1 rad/s^2

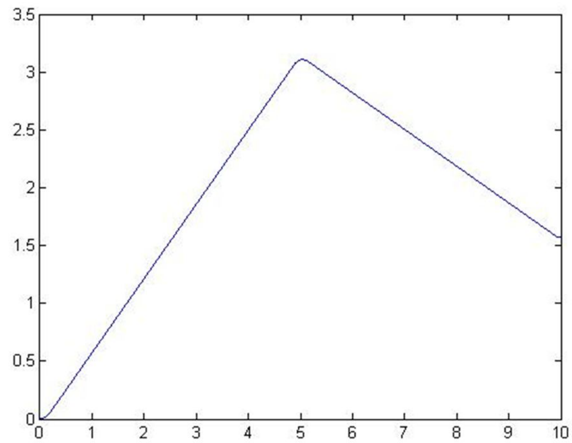


Fig. 6: Acceleration = 3 rad/s^2

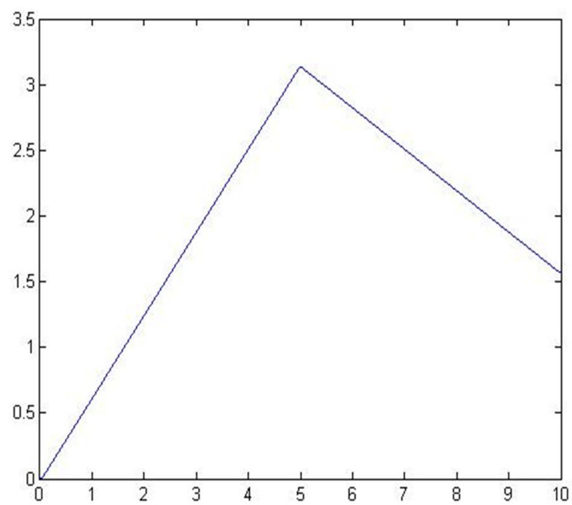


Fig. 7: Acceleration = 9 rad/s^2

The percentage error has been calculated as follows.

$$\%error = \frac{q_{required} - q_{reached}}{q_{required}} \times 100 \quad (19)$$

The results of this analysis have been stated in Table III.

Table III: Positional Error as a Function of Acceleration

Acceleration (rad/s^2)	Percentage Error (%)
1	3.47
3	1.09
9	0.36

As one can clearly observe that there exists an approximate linear decrement correlation between the percentage error and the acceleration.

V. CONCLUSION

Thus in this work, an optimization methodology has been discussed to optimize the joint space trajectory with respect to minimization of traversal time. In the case when the user specifies actuator parameters which are not complacent with positions required and the velocities specified, a bisection method was used to find the optimum achievable velocity. Also a correlation between the acceleration and the positional error has been established.

VI. ACKNOWLEDGMENT

We would like to thank Department of Mechanical Engineering of Vishwakarma Institute of Technology for the constant support and motivation.

REFERENCES

[1] Luh, J. Y. S., Walker, M. W., "Minimum-time along the path for a mechanical arm," Proceedings of the 1977 IEEE Conference on Decision and Control, Vol. 1, 77CH1269-OCS, Dec. 7-9, 1977, New Orleans, pp. 755-759.

[2] O. Dahl and L. Nielsen, “Torque limited path following by on-line trajectory time scaling,” IEEE Trans. on Robotics and Automation, 1989.

[3] Kang. G. Shin, Neil D. McKay, “Minimum time trajectory planning for industrial robots with general torque constraints,” Robotics and Automation, Proceedings, April 1986.

[4] Yasmina Bestaoui, “On line motion generation with velocity and acceleration constraints,” Robotics and Autonomous Systems, Volume 5, Issue 3, November 1989, pp. 279-288.

[5] Zvi Shiller et. al., “Computation of path constrained time optimal motions with dynamic singularities,” in Journal of Dynamic Systems Measurement and Control · March 1992.

[6] Kang. G. Shin, Neil D. McKay, “A dynamic programming approach to trajectory planning of robotic manipulators,” IEEE Transactions on Automatic Control, Vol. AC-31, No. 6, June 1986.

[7] J. Y. S. Luh, C. S. Lin, “Optimum path planning for mechanical manipulators,” Transactions of the ASME, 142/Vol. 102, June 1981.

[8] J. Mattmuller and D. Gisler, “Calculating a near time-optimal jerk-constrained trajectory along a specified smooth path,” Int. J. Adv. Manuf. Technol., 45:1007, December 2009.

[9] M. H. Ghasemi, N. Kashiri, and M. Dardel, “Near time-optimal control of redundant manipulatros along a specified path with jerk constraint,” Advanced Robotics, 25:2319-2339, 2011

[10] Kang. G. Shin, Neil D. McKay, “Minimum-time control of robotic manipulators with geometric path constraints,” IEEE Transactions on Automatic Control, Vol. AC-30, No. 6, June 1985.

[11] Atef A. Ata, “Optimal trajectory planning of manipulators: a review,” Journal of Engineering Science and Technology, Vol. 2, No. 1, 2007, pp. 32-54.

[12] M. E. Khan, B. Roth, “The near-minimum time control of open loop articulated kinematic chains,” Transactions of the ASME, Journal of Dynamic Systems, Measurement and Control, September 1971, pp. 164-172.

[13] R. K. Mittal and I. J. Nagrath, Robotics and Control, Tata McGraw-Hill Education Private Limited, Co. 2003, pp. 247 - 285.

[14] Steven C. Chapra, Raymond P. Canale, Numerical Methods for Engineers, Sixth Edition, McGraw-Hill, Co. 2010, pp.124-131.

APPENDIX 'A'- FLOWCHART OF TRAJECTORY COMPUTATION ALGORITHM