11th International Research/Expert Conference
"Trends in the Development of Machinery and Associated Technology"
TMT 2007, Hammamet, Tunisia, 05-09 September, 2007.

COMPARISSON BETWEEN INTERPOLATING POLYNOMIALS WITH VELOCITY CONSTRAINTS AND CONTINUOUS ACCELERATION AT PATH POINTS OF END EFFECTOR

Ph.D. Vlatko Doleček, professor
Mašinski fakultet Sarajevo
ul. Vilsonovo šetalište 9, 71000 Sarajevo, Bosna i Hercegovina
e-mail: vldolecek@bih.net.ba

Armin Hadžić, dipl. ing. mašinstva
Kakanjcement
ul. Selima ef. Merdanovića 146, Kakanj, Bosna i Hercegovina
e-mail: tvcemk06@bih.net.ba,
tel: +387 33 221 757

ABSTRACT
The main goal of trajectory planning of robot manipulator is to generate of referal inputs to the motion control system, which assures manipulator to execute the planned trajectories. It is usual that user specifies a number of parameters which describe planned trajectory. Planning consists of generating the time sequence of values attained by a polinomial function which interpolates planned trajectory. This paper analises the influence of virtual points in interpolating polynomials of third order with continous acceleration in path points on calculated values of acceleration and compares use of interpolation polynomials with velocity constraints and continuous acceleration at path points. We will show that interpolation polynomials with continuous acceleration at path points is preferable, as interpolation polynomials with velocity constraints does not assure continuity of acceleration at path points.
Keywords : virtual points, trajectory, path.

1. PATH MOTION
In several applications, the path is described in terms of a number of points greater than two. For instance, even for the simple point-to-point motion of a pick and place task, it may be worth assigning two intermediate points between the initial point and the final point; suitable positions can be set for lifting off and setting down the object. For more complex applications, it may be convenient to assign a *sequence of points* so as to guarantee better monitoring on the executed trajectories; the points are to be specified more densely in those segments of the path where obstacles have to be avoided or a high path curvature is expected. It should not be forgotten that the corresponding joint variables have to be computed from the operational space locations.
Therefore, the problem is to generate a trajectory when N points, termed *path points*, are specified and have to be reached by the manipulator at certain instants of time. For each joint variable there are N constraints, and then one might want to use an (N-1)-order polynomial. This choice, however, has the following disadvantages:
- It is not possible to assign the initial and final velocities,
- As the order of the polynomial increases, its oscillatory behavior increases, and this may lead to trajectories which are not natural for the manipulator.
- Numerical accuracy for computation of polynomial coefficients decreases as order increases,
- The resulting system of constraint equations is heavy to solve,

- Polynomial coefficients depend on all the assigned points; thus, if it is desired to change a point, all of them have to be recomputed.

These drawbacks can be overcome if a suitable number of law-order interpolating polynomials, continuous at the path points, are considered in place of a single high-order polynomial.

According to the previous section, the interpolating polynomial of lowest order is the cubic polynomial, since it allows imposing continuity of velocities at the path points. With reference to the single joint variable, a function $q(t)$ is sought, formed by a sequence of N-1 cubic polynomials $\Pi_k(t)$, for k=1,…,N-1, continuous with continuous first derivatives. The function $q(t)$ attains the values q_k for $t = t_k (k = 1,..., N)$, and $q_1 = q_i, t_1 = 0, q_N = q_f, t_N = t_f$; the q_k's represent the path points describing the desired trajectory at $t = t_k$ -Figure 1. The following situations can be considered:

- Arbitrary values of $\dot{q}(t)$ are imposed at the path points,
- The values of $\dot{q}(t)$ at the path points are assigned according to a certain criterion,
- The acceleration $\ddot{q}(t)$ shall be continuous at the path points.

To simplify the problem, it is also possible to find interpolating polynomials of order less than three which determine trajectory passing nearby the path points at the given instant of time.

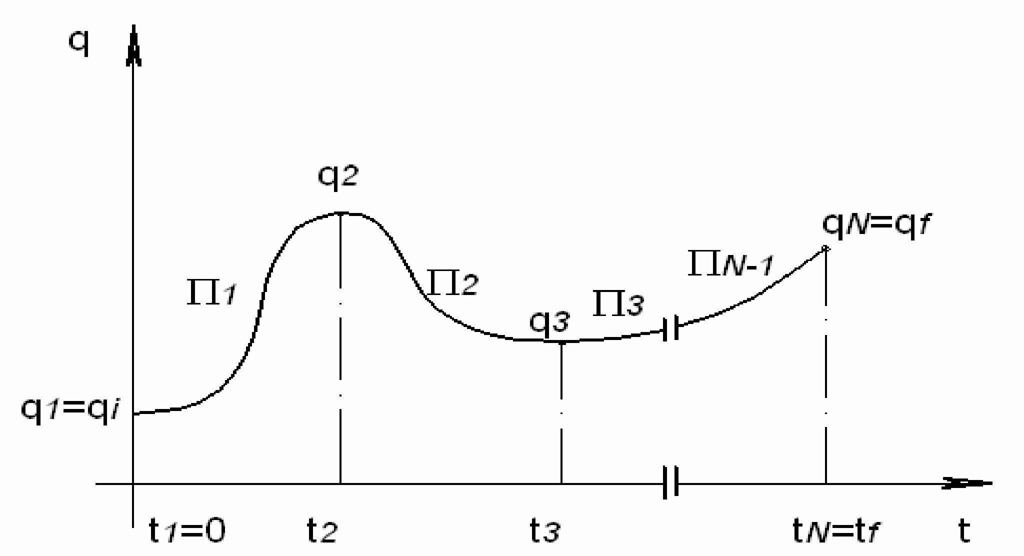


Figure 1. Characterization of a trajectory on a given path obtained through interpolating polynomials.

1.1. Interpolating Polynomials with Velocity Constraints at Path Points.

This solution requires the user to be able to specify the desired velocity at each path point. The system of equations allowing computation of the coefficients of the N-1 cubic polynomials interpolating the N path points is obtained by imposing the following conditions on the generic polynomial $\Pi_k(t)$ interpolating q_k and q_{k+1}, for $(k = 1,..., N-1)$:

$$\Pi_k(t_k) = q_k\ ;\ \Pi_k(t_{k+1}) = q_{k+1}\ ;\ \dot{\Pi}_k(t_k) = \dot{q}_k\ ;\ \dot{\Pi}_k(t_{k+1}) = \dot{q}_{k+1}. \quad (1)$$

The result is N-1 systems of four equations in the four unknown coefficients of the generic polynomial; these can be solved one independently of the other. The initial and final velocities of the trajectory are typically set to zero ($\dot{q}_1 = \dot{q}_N = 0$) and continuity of velocity at the path points is ensured by setting $\dot{\Pi}_k(t_{k+1}) = \dot{\Pi}_{k+1}(t_{k+1})$ for $k = 1,..., N-2$.

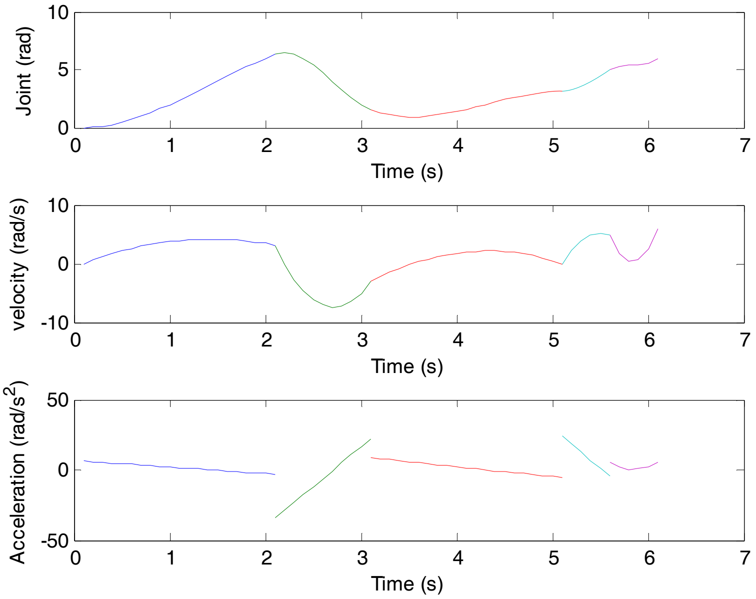


Figure 2. Time history of position, velocity and acceleration with a time law of interpolating polynomials with velocity constraints at path points.

Figure 2 illustrates the time history of position, velocity, and acceleration obtained with the data: $q_1 = 0, q_2 = 2\pi, q_3 = \pi/2, q_4 = \pi, t_1 = 0, t_2 = 2, t_3 = 3, t_4 = 5, \dot{q}_1 = 0, \dot{q}_2 = \pi, \dot{q}_3 = -\pi, \dot{q}_4 = 0.$ Resulting discontinuity on the acceleration, since only continuity of velocity is guaranteed.

1.2. Interpolating Polynomials with Continuous Accelerations at Path Points (Splines). Above solution does not ensure continuity of accelerations at the path points. Given a sequence of N path points, also the acceleration is continuous at each t_k if four constraints are imposed; two positions constraints for each of the adjacent cubics and two constraints guaranteeing continuity of velocity and acceleration. The following equations have then to be satisfied:

$$\Pi_{k-1}(t_k) = q_k\,;\ \Pi_{k-1}(t_k) = \Pi_k(t_k)\,;\ \dot{\Pi}_{k-1}(t_k) = \dot{\Pi}_k(t_k)\,;\ \ddot{\Pi}_{k-1}(t_k) = \ddot{\Pi}_k(t_k). \qquad (2)$$

The system can be solved only if one eliminates the two equations which allow arbitrarily assigning the initial and final acceleration values. Fourth-order polynomials should be used to include this possibility for the first and last segment.

On the other hand, if only third-order polynomials are to be used, we could proceed as follows: Two virtual points are introduced for which continuity constraints on position, velocity, and acceleration can be imposed, without specifying the actual positions, though. The effective location of these points is irrelevant, since their position constraints regard continuity only. The introduction of two virtual points implies the determination of N+1 cubic polynomials.

Consider N+2 time instants t_k, where t_2 and t_{N+1} conventionally refer to the virtual points. The system of equations for determining the N+1 cubic polynomials can be found by taking the 4(N-2) equations:

$$\Pi_{k-1}(t_k) = q_k\,;\ \Pi_{k-1}(t_k) = \Pi_k(t_k)\,;\ \dot{\Pi}_{k-1}(t_k) = \dot{\Pi}_k(t_k)\,;\ \ddot{\Pi}_{k-1}(t_k) = \ddot{\Pi}_k(t_k) \qquad (3)$$

for $k = 3,\ldots,N$, written for the N-2 intermediate path points, the 6 equations:

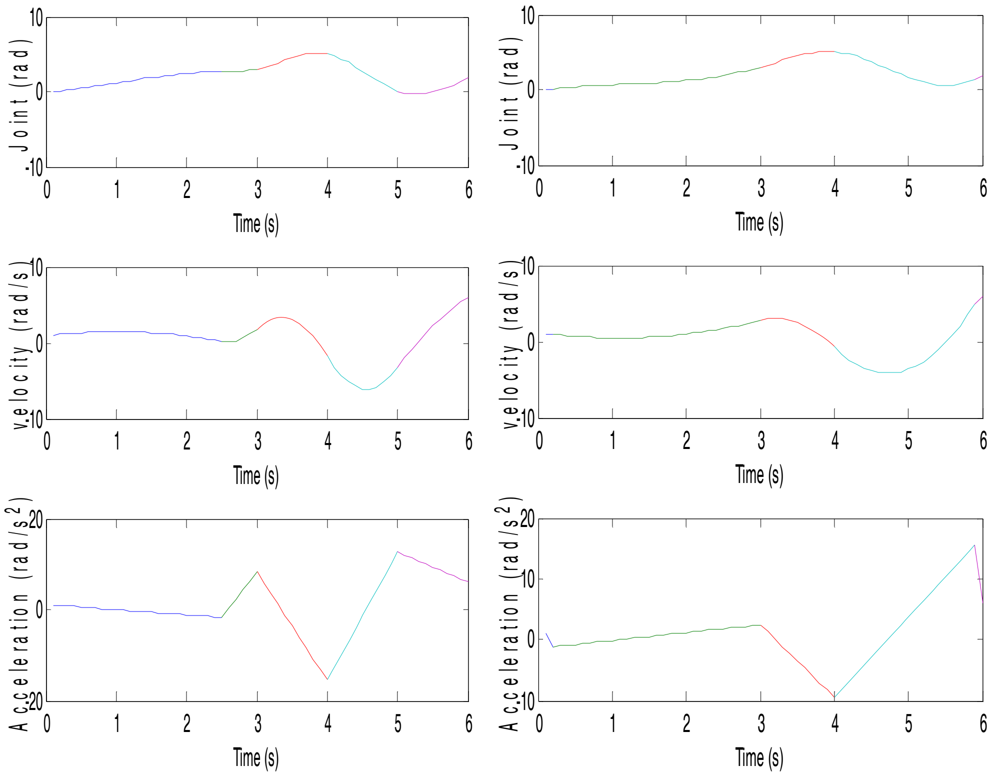
$$\Pi_1(t_1) = q_i\,;\ \dot{\Pi}_1(t_1) = \dot{q}_i\,;\ \ddot{\Pi}_1(t_1) = \ddot{q}_i\,;\ \Pi_{N+1}(t_{N+2}) = q_f\,;\ \dot{\Pi}_{N+1}(t_{N+2}) = \dot{q}_f\,;\ddot{\Pi}_{N+1}(t_{N+2}) = \ddot{q}_f \qquad (4)$$

written for then initial and final points, and the 6 equations:

$$\Pi_{k-1}(t_k) = \Pi_k(t_k)\,;\ \dot{\Pi}_{k-1}(t_k) = \dot{\Pi}_k(t_k)\,;\ \ddot{\Pi}_{k-1}(t_k) = \ddot{\Pi}_k(t_k) \qquad (5)$$

for $k = 2, N + 1$, written for the two virtual points. The resulting system has 4(N+1) equations in 4(N+1) unknowns, that are the coefficients of the N+1 cubic polynomials. The solution to the system is computationally demanding, even for low values of N.

The above sequence of cubic polynomials is termed *spline* to indicate smooth functions that interpolate a sequence of given points ensuring continuity of the function and its derivatives.



(a) (b)

Fig. 3. Time history of position, velocity, and acceleration with a time law of cubic splines for two different pairs of virtual points.

(a) Time history of position, velocity, and acceleration for t2=2.5, t5=5, a55=-1,09923

(b) Time history of position, velocity, and acceleration for t2=0.2, t5=5.9, a55=-15,7996

Fig. 3 illustrates the time history of position, velocity, and acceleration obtained with the data: q_1=0, q_3=2π, q_4=π/2, q_6=π, t_1=0, t_3=2, t_4=3, t_6=5, q'_1=0, q'_6=0. Two different pairs of virtual points were considered at the time instants: t_2=0,2, t_5=5,9 (b), and t_2=2,5, t_5=5,0 (a). For the second pair, larger values of acceleration are obtained, since the relative time instants are closer to those of the two intermediate points.

We showed that, as virtual points are closer to the defined internal points of the trajectory, greater values of acceleration are calculated in internal points of trajectory.

2. REFERENCES

[1] Sciavicco L., Siciliano B.: Modeling and Control of Robot Manipulators, McGraw-Hill International Editions, 1996.,
[2] Spong M., Hutchinson S., Vidyasagar M.: Robot Modeling and Control Wiley, 2006,
[3] McKerrow P. J.: Introduction to Robotics, Addison Wesley, 1998,
[4] Craig J. J.: Introduction to Robotics, Mechanics and Control, Pearson Prentice Hall, 2005,
[5] Dolecek V., Karabegovic I.: Robotika, Grafičar Bihac, 2002,
[6] Murray R., Li Z., Sastry S.S.: A Mathematical Introduction to Robotic Manipulation., CRC Press, 1994,
[7] Tsai L.W.: Robot Analysis, John Wiley & sons, inc, 1999,
[8] International Journal of Robotics and Automation, 2006.
[9] IEEE Robotics and Automation Magazine, 2006.