Robotica (2000) volume 18, pp. 611–631. Printed in the United Kingdom

Trajectory planning of robot manipulators by using algebraic and trigonometric splines*

Antonio Visioli

Dipartimento di Elettronica per l'Automazione, University of Brescia, Via Branze 38-25123, Brescia (Italy)
e-mail: visioli@bsing.ing.unibs.it

(Received in Final Form: March 18, 2000)

SUMMARY
In this paper, the use of algebraic and trigonometric splines for the trajectory planning of robot manipulators is discussed. First, the two methods are analyzed and compared in detail; then, a strategy, which involves a combined use of the two schemes to perform sudden changes in a predefined trajectory (e.g. in case of obstacle avoidance) is proposed. Results show that the main interest in using trigonometric splines lies especially in the task of connecting two separate pieces of cubic splines, as overshoots are significantly reduced, although the continuity of velocity, acceleration and (in case) jerk is guaranteed.

KEYWORDS: Trajectory planning; Trigonometric splines; Algebraic splines; Real-time obstacle avoidance.

1. INTRODUCTION

The motion of an industrial robot manipulator is generally specified in terms of the motion of the end-effector in the Cartesian space. However, from a technological point of view, the implementation of a motion control system in the Cartesian space is very difficult, mainly because obtaining an accurate direct measurement of the end-effector's position is a very complex task. Thus, in practical cases the end-effector's motion is converted into the joints' motion by applying inverse kinematics, and then the control task is performed in the joint space.[1]

In many situations, the end-effector's motion is actually specified as a sequence of via points, whose number is determined by taking into account the trade-off between exactness and computational expense. Then, these via points are mapped into a set of joint angles/offset (knots) that have to be subsequently interpolated in the joint space by means of a selected function. Between the interpolating functions, (algebraic) cubic splines are widely adopted because they can assure the continuity of the position, velocity and acceleration commands for each joint. Moreover, some optimization procedures can be accomplished in this framework.[2–4]

Recently, the use of the trigonometric splines, first introduced by Schoenberg[5] in 1964, have also been proposed in trajectory planning of robot manipulators by Simon and Isik.[6–10] In their very interesting works, Simon and Isik have stressed the fact that joint trajectories with continuous velocity, acceleration and jerk and low overshoot can be generated. Besides, the computational effort is very low, and an optimization procedure can be adopted for the selection of the spline parameters in order to minimize an objective function, (minimum jerk or minimum energy). Hence, the use of trigonometric splines in the trajectory planning in the joint space seems to be very promising, especially to implement a real-time obstacle avoidance functionality. In any case, there remain some unclear points that deserve further investigation. For example, the influence of the order of the trigonometric splines on the overall result has to be analyzed. In fact, in the Simon and Isik's results, only the fourth order splines are considered, i.e. a continuity of the jerk function is imposed. But in practical cases the jerk might be discontinuous, although its limitation generally guarantees lower tracking errors and the excitation of resonances is somewhat prevented. Besides, in the proposed schemes, the spline intervals are assumed to be the same, which is a strong assumption in real industrial automated cells.

In this paper a more detailed analysis of trigonometric splines from the design point of view is provided, together with a comparison with algebraic splines. Furthermore, the main contribution of the paper is the exposition of a new method which involves both cubic splines and trigonometric splines and which seems particularly appropriate for real-time obstacle avoidance. Specifically, trigonometric splines are employed to connect pieces of cubic splines when the original cubic splines trajectory is interrupted by a obstacle.

The paper is organized as follows: In Section 2 the use of both algebraic and trigonometric splines for the trajectory planning problem is described. In Section 3 some examples are presented in order to discuss and compare the two approaches. The combined use of cubic and trigonometric splines is proposed in Section 4 and illustrated by an example in Section 5. Conclusions are drawn in the final section.

2. TRAJECTORY PLANNING WITH SPLINES

2.1. Generalities

The trajectory planning task of a robot manipulator is generally specified in terms of the motion of the end-effector in the Cartesian space. In order to implement a

* This work was supported in part by MURST scientific research funds.

https://doi.org/10.1017/S0263574700002721 Published online by Cambridge University Press

typical motion controller in the joint space, the trajectory in the Cartesian space has to be mapped in the joint space by applying the inverse kinematics. A way to specify a trajectory is to give a sequence of intermediate points in the Cartesian space to be passed through by the end-effector and therefore a corresponding sequence of angles/offsets to be assumed by each joint at given time instants.

In this context, if it is not strictly necessary that the manipulator assumes exactly the desired poses, a simple technique is to connect the via points with linear functions and then add parabolic blend regions around each via point. Conversely, if it is required that the robot passes exactly through the intermediate points, a suitable solution is to use splines. By choosing an appropriate order of the spline, the user can guarantee the continuity of velocity, acceleration and jerk signals along the whole trajectory.

2.2. Algebraic splines

Algebraic splines are widely known and adopted in the robotics field. In particular, cubic splines are often employed, as they assure the continuity of velocity and acceleration signals along the planned motion. Besides, the parameters are easy to calculate and large oscillations of the position function and its time derivatives are prevented. Suppose to have a sequence $\mathbf{q}=[q_0, q_1, \ldots, q_n]$ of intermediate positions that the joint has to pass through at time $\mathbf{t}=[t_0, t_1, \ldots, t_n]$, respectively. Denote the spline times as $h_i := t_i - t_{i-1}$, $i=1, \ldots, n$. Consider also that velocity at time t_0 is v_0 and at time t_n is v_n. Then, we can write a set of polynomial functions:

$$Q_i(t)=a_it^3+b_it^2+c_it+d_i \quad i=1,\ldots,n, \tag{1}$$

that represents the position function linking knot q_{i-1} and q_i. The values of the $4n$ coefficients can be determined considering the following initial and final conditions:

$$\begin{cases} Q_1(t_0)=q_0 \\ \dot{Q}_1(t_0)=v_0 \\ Q_n(t_n)=q_n \\ \dot{Q}_n(t_n)=v_n \end{cases} \tag{2}$$

and for any intermediate point ($i=1, \ldots, n-1$):

$$\begin{cases} Q_i(t_i)=q_i \\ Q_{i+1}(t_i)=q_i \\ \dot{Q}_i(t_i)=\dot{Q}_{i+1}(t_i) \\ \ddot{Q}_i(t_i)=\ddot{Q}_{i+1}(t_i) \end{cases} \tag{3}$$

It appears that in this framework, the initial and final accelerations a_0 and a_n cannot be fixed a priori. In order to do that, a quintic polynomial for the first and last splines is required, with the obvious drawback to allow larger overshoots in these parts of the trajectory and to slightly increase the number of computations in the control system. An alternative method is to add two “free” extra-knots in second and penultimate positions.[2]

2.3. Trigonometric splines

One of the main advantages claimed for trigonometric splines is the prevention of large overshoots despite the fact that the continuity of the function can be imposed until a high order (for example, by using a fourth-order trigonometric spline the continuity of the jerk function is guaranteed and the overshoots are significantly reduced with respect to the quartic algebraic splines that satisfy the same requirement). In general, for a m-order spline, the position function between two knots can be determined fixing the values of the derivatives until the m-1 order at the knots. A clear theoretical advantage of the trigonometric splines with respect to the algebraic ones is the opportunity to fix these values (depending on the motion task) without increasing the overshoots of the trajectory too much.[6,7,9] Thus, this seems to be a significant feature, worth to be exploited for real-time obstacle avoidance algorithms.

The formal definition of a trigonometric spline is the following:

Definition 1 *An m-th order trigonometric spline function y(t) with a total of 2m constraints in each of the n closed arcs* $[t_{i-1}, t_i]$ ($i=1, \ldots, n$) *is defined as*

$$y(t)=y_i(t) \quad t\in[t_{i-1}, t_i] \tag{4}$$

where $y_i(t)$ *is given by*

$$y_i(t)=a_{i,0}+\sum_{k=1}^{m-1}(a_{i,k}\cos kt+b_{i,k}\sin kt) + a_{i,m}\sin m(t-\gamma_i) \tag{5}$$

$$\gamma_i=\sum_{j=0}^{2m-1}\frac{\tau_{i,j}}{2m} \tag{6}$$

and $\tau_{i,j}$ *are the values of t where* $y_i(t)$ *has a constraint applied.*

The existence and uniqueness of these functions are guaranteed provided that, for each i and j, $y_i^{(r)}(\tau_{i,j})$ is not constrained unless $y_i^{(r-1)}(\tau_{i,j})$ is also constrained ($r=1, 2, \ldots$), where $y^{(r)}$ denotes the r-th order time derivative of y. From (5) it appears that there are $2m$ coefficients for each segment of the trigonometric spline, so that $2m$ costraints on each segment have to be satisfied. They can be chosen to be $y^{(r)}(t_i)=y_i^{(r)}$, $r=0, \ldots, m-1$, $i=0, \ldots, n$. Of course, it has also to be $y_i^{(r)}(t_i^-)=y_i^{(r)}(t_i^+)$ if the trigonometric spline and its first $(m-1)$ derivatives have to be continuous. However, by constraining the values of $y_i^{(r)}(t_i)$ rather than simply requiring continuity, it results that the determination of the coefficients is decoupled for each spline segment.

In general, each trigonometric polynomial is normalized, that is, the spline times $\theta_i := t_i - t_{i-1}$ are expressed in radians according to the following expression

$$\theta_i=\frac{n\frac{\pi}{m}h_i}{T_{tot}} \quad i=1,\ldots,n \tag{7}$$

where h_i is the time interval of the i-th polynomial (in seconds) and $T_{tot} := \sum_{i=1}^{n} h_i$ is the motion time (in seconds)

https://doi.org/10.1017/S0263574700002721 Published online by Cambridge University Press

of the whole trajectory. Note also that for each polynomial we can easily impose $t_{i-1}=0$, and hence we have $\theta_i=t_i$. It appears that if the spline time intervals are assumed to be equal to each other and we fix $m=4$, then $\theta_i=t_i=\pi/4$ and $\gamma_i=\pi/8$, i=1, . . . , n. Again, it is worth stressing that once the spine intervals have been fixed, the determination of the spline coefficients is easily performed by multiplying a constant matrix with the vector of the knot angles and derivatives.[9]

The setting of the constraints $y^{(r)}(t_i)=y_i^{(r)}$, $r=0, \ldots, m-1$, $i=0, \ldots, n$ might not be intuitive to the user, but a useful optimization procedure can be exploited in order to determine the values that minimize an objective function, such as the integral of the squared jerk function over the whole trajectory.[6] In this case the optimization problem has a closed-form solution.

3. COMPARATIVE EXAMPLES

It has been shown in the previous section that the main advantage of the trigonometric splines over the algebraic ones is that they allow to fix constraints on the position function and its derivatives at the knots, rather than requiring their continuity. In this way, a single segment can be modified during the motion without recalculating the whole trajectory, which is a highly desirable feature in real-time applications (for example in avoiding unexpected obstacles). If a similar approach is adopted for algebraic splines then a high-order polynomial is to be employed; this would clearly result in large overshoots between the knots, which is unacceptable for robot trajectory planning, especially if an obstacle needs to be avoided. In this context, the superiority of the fourth-order trigonometric splines over the quartic algebraic splines has already been shown in the works of Simon and Isik.[6,9] Therefore, the purpose of this section is to analyze the use of different order trigonometric splines determined with different methods for the choice of the constraints values and to compare the results with the ones obtained by using the typical cubic splines. As a first illustrative example, a single link trajectory is considered; it has to connect the following angles (in degrees): **q**=[120, 60, 80, 120, 0]. The employed time intervals are 2s for each polynomial (this case will hereinafter be denoted as example I) and therefore we are in the situation assumed in the previous works, where the spline times are assumed to be the equal to each others. The values of velocity, acceleration and jerk for the fourth-order trigonometric splines have been determined according to the optimization procedure described in [6], which aims at minimizing the integral of the squared jerk function over the whole trajectory. The same results, except for the jerk value, have been also applied to the third-order trigonometric splines constraints. Results are plotted in Figures 1-4, for the position (in [deg]), velocity (in [deg/s]), acceleration (in [deg/s^2]) and jerk (in [deg/s^3]) profiles, respectively. Note that for the cubic splines a jerk impulse is actually present at the beginning and at the end of the trajectory. It can be noted that the only draw-back of the cubic splines is just the presence of the discontinuity in the acceleration function at these points. This can be avoided, as already mentioned, by adding two free extra-knots in the second and penultimate

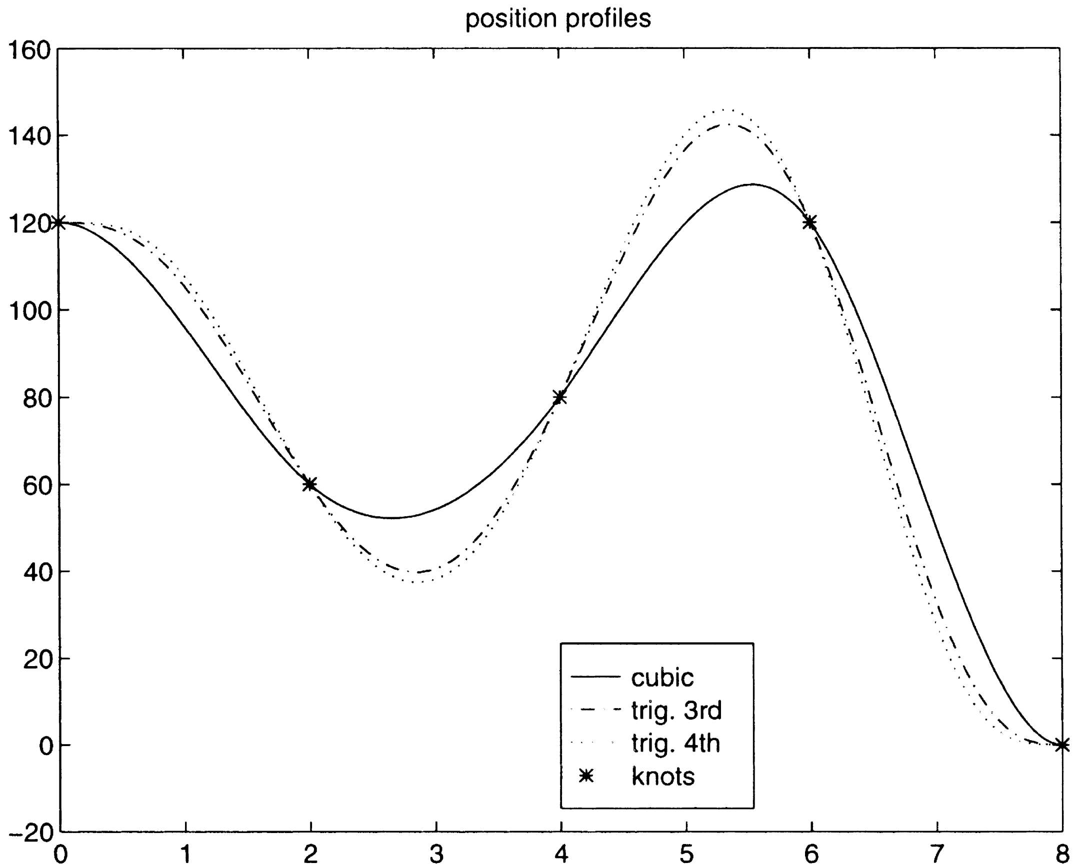

Fig. 1. Position functions for the cubic and trigonometric splines. Example I.

https://doi.org/10.1017/S0263574700002721 Published online by Cambridge University Press

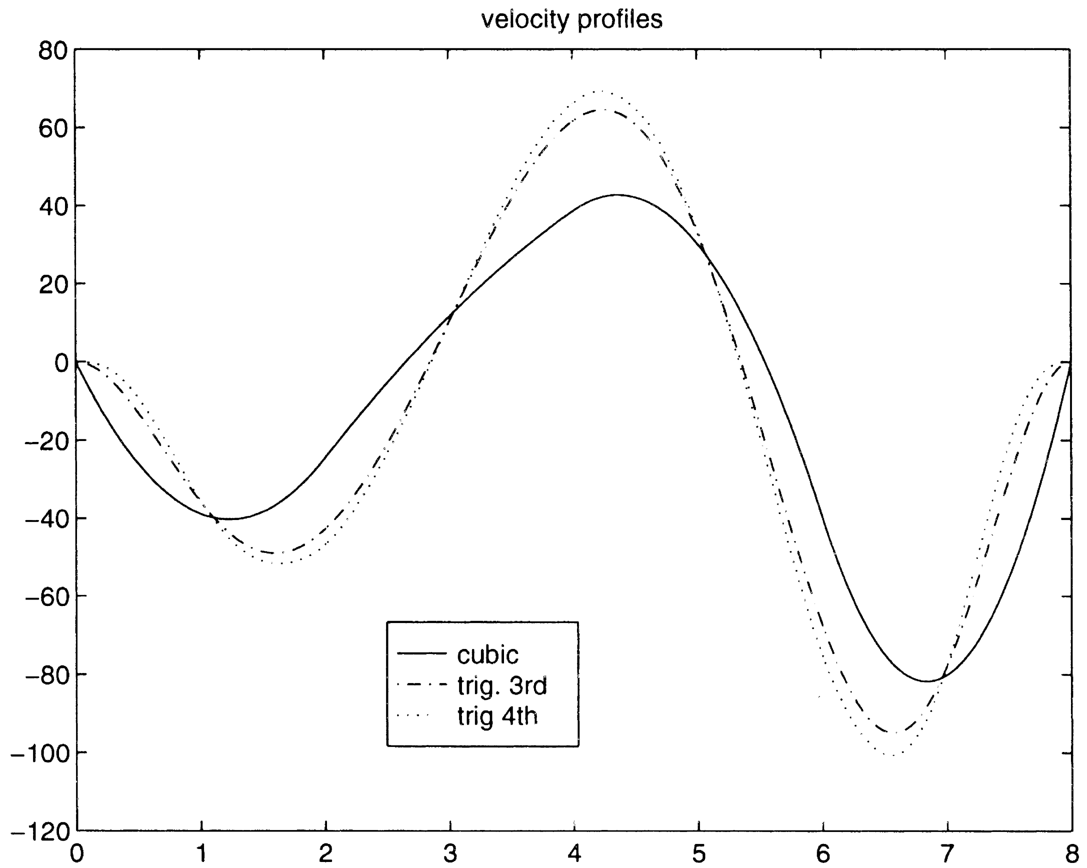

Fig. 2. Velocity functions for the cubic and trigonometric splines. Example I.

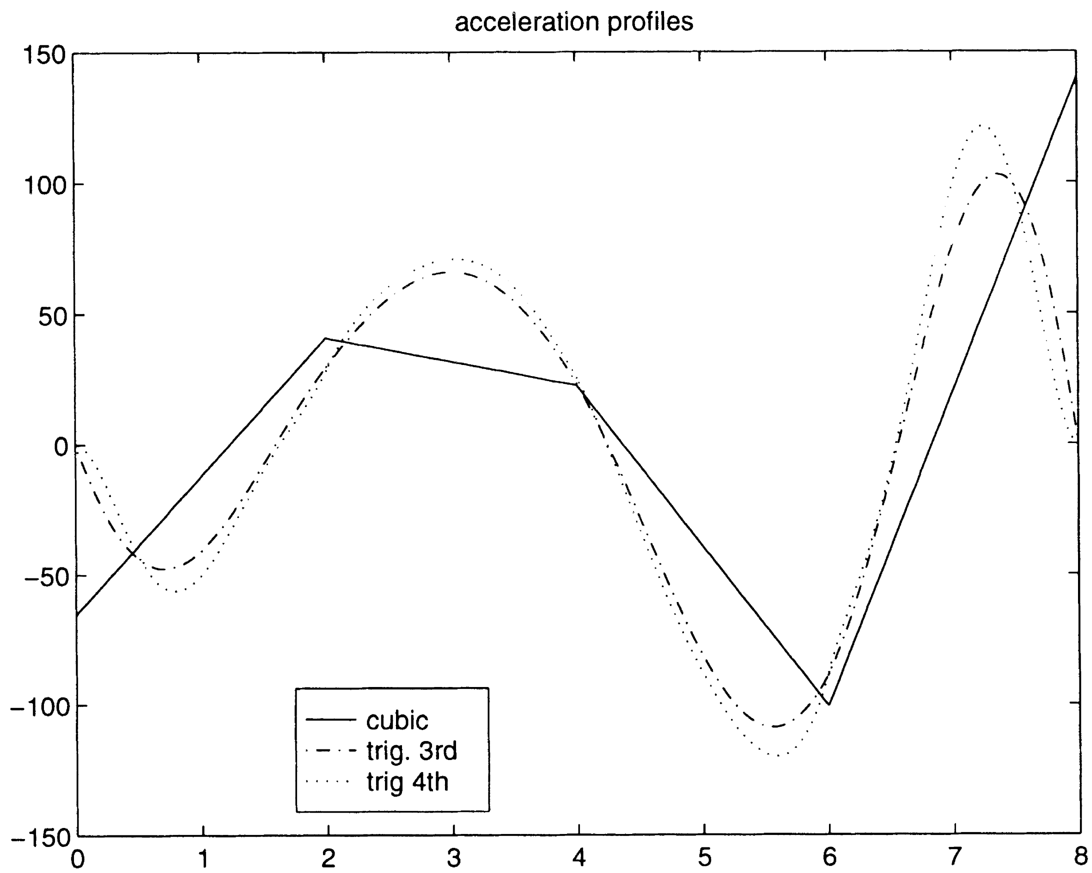

Fig. 3. Acceleration functions for the cubic and trigonometric splines. Example I.

https://doi.org/10.1017/S0263574700002721 Published online by Cambridge University Press

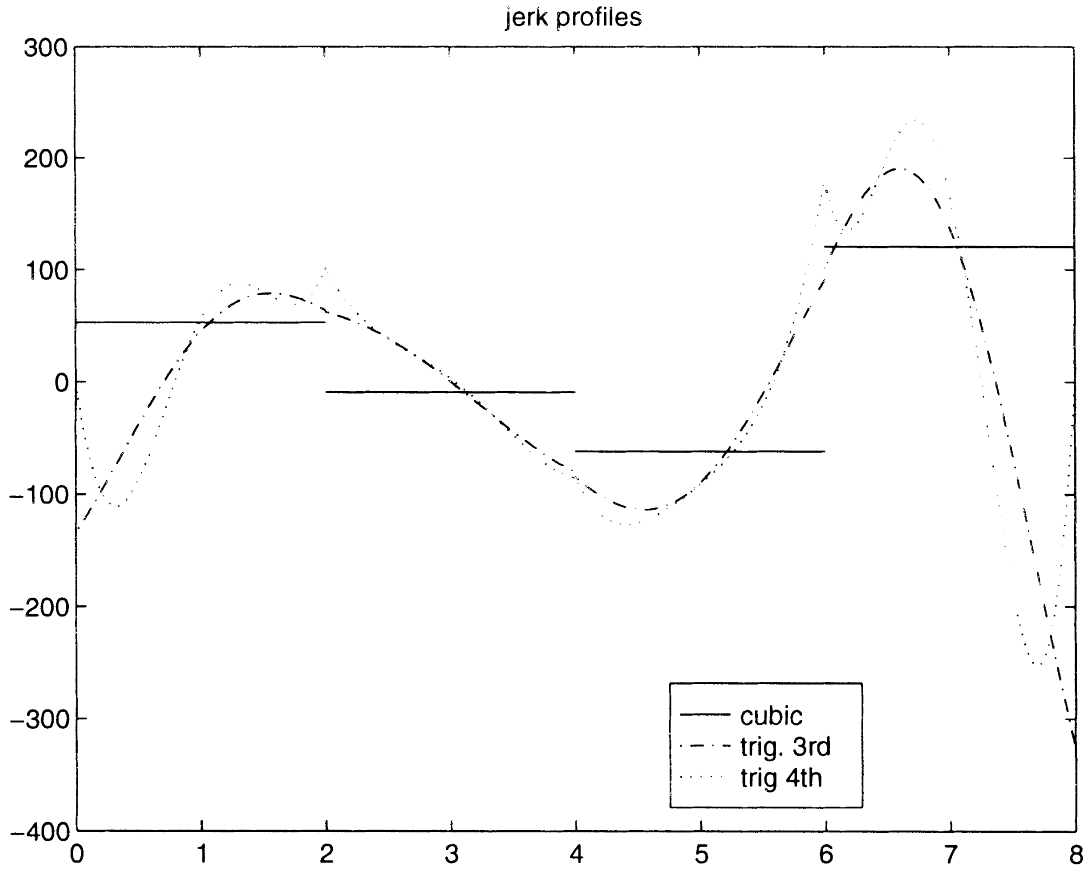

Fig. 4. Jerk functions for the cubic and trigonometric splines. Example I.

position or, otherwise, by considering a fifth-order polynomial for the first and last interval. Especially in the case of many intermediate knots, this method does not yield particular problems. From other points of view, cubic splices still assure better performances, as the overshoots are less significant with respect to the trigonometric ones (both of third and fourth order) and the position derivatives functions are not significantly different.

Another interesting method that has to be considered in the analysis is to employ different time intervals between two subsequent knots. With the same knots as before, we considered the following time intervals: $\mathbf{h} = [1, 3, 5, 1]$. Two cases emerges for the trigonometric splines: in the first (example II), the constraints at the knots are evaluated by considering the spline times equal to each other (i.e. by considering each time interval equal to $T_{tot}/(n-1)$. In the second (example III), the optimization procedure for the minimization of the jerk has been adapted to consider different time intervals and the resulting constraint value have been employed. Results related to the first and second case are reported in Figures 5–8 and 9–12, respectively. It is clear that it is not useful to consider the different time intervals as they are, since in this case the overshoot are huge and surely not tolerable by the system. If the spline times are treated as being of the same value, much better results are obtained, even better than the ones obtained by using cubic splines. It can be noted in Figure 5 that the best position function is the one obtained by using the fourth-order trigonometric spline. However, the superiority of the trigonometric splines in the position function is paid by having significantly higher values in the acceleration and jerk functions (again disregarding the impulses at the beginning and at the end of the trajectory that are present in the jerk profile for the cubic spline).

Another significant example is to use time intervals proportional to the angular displacement between the knots. In this case the spline times have been selected as $\mathbf{h} = [3, 1, 2, 6]$ (example IV). Results are plotted in Figures 13–16 and show that in this case cubic splines outperform the trigonometric ones.

In view of the above illustrative examples and of many others not reported for brevity, the following conclusions can be drawn:

- for the trigonometric splines, the optimization procedure (minimum jerk) to determine the constraint values of the position derivatives at the knots has to be applied, considering the spline times as equal to each others and therefore disregarding their actual values;
- if there is some sort of ‘regularity’ in the spline times (i.e. they are equal to each other or they are proportional to the angular displacements, then it seems that from a practical point of view, in general, it is not worth leaving the cubic splines framework just to impose the continuity of the jerk function;
- when the splines times are not related to the intermediate points of the trajectory, then it might be worth employing the trigonometric splines, taking into account that they give smaller values of the overshoots but also bigger values of the acceleration and jerk functions;

https://doi.org/10.1017/S0263574700002721 Published online by Cambridge University Press

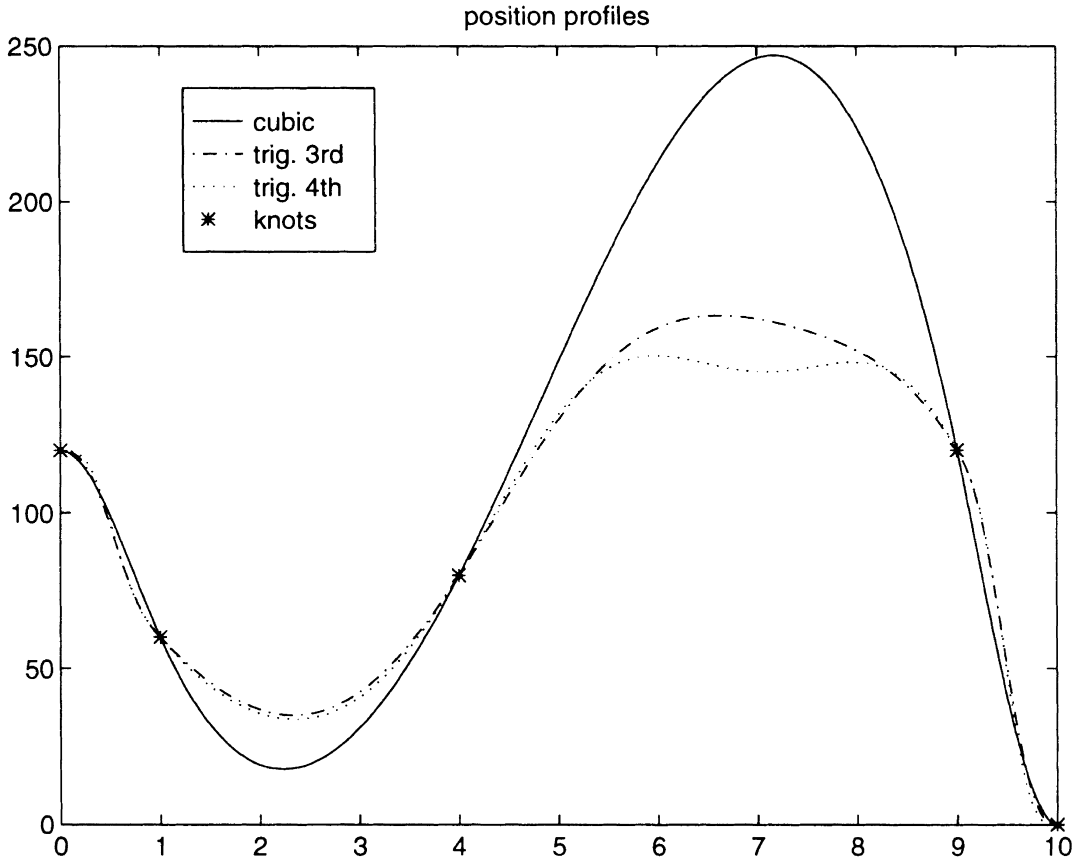

Fig. 5. Position functions for the cubic and trigonometric splines. Example II.

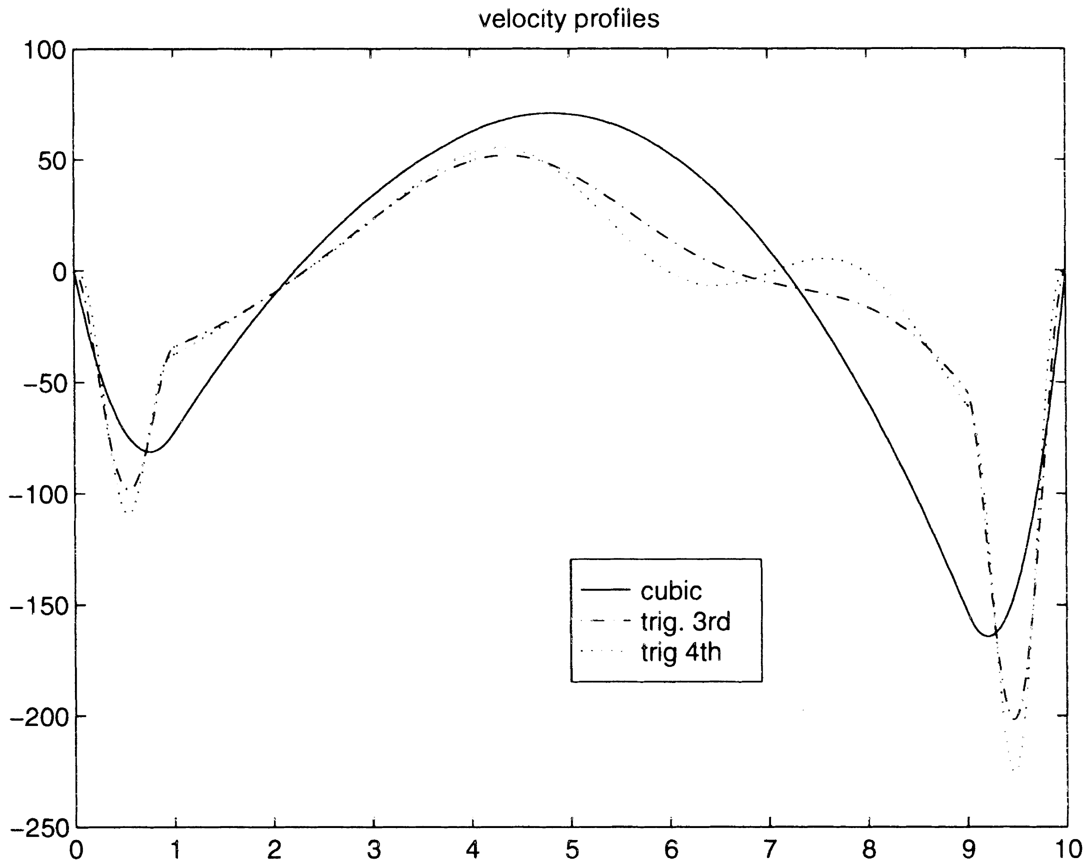

Fig. 6. Velocity functions for the cubic and trigonometric splines. Example II.

https://doi.org/10.1017/S0263574700002721 Published online by Cambridge University Press

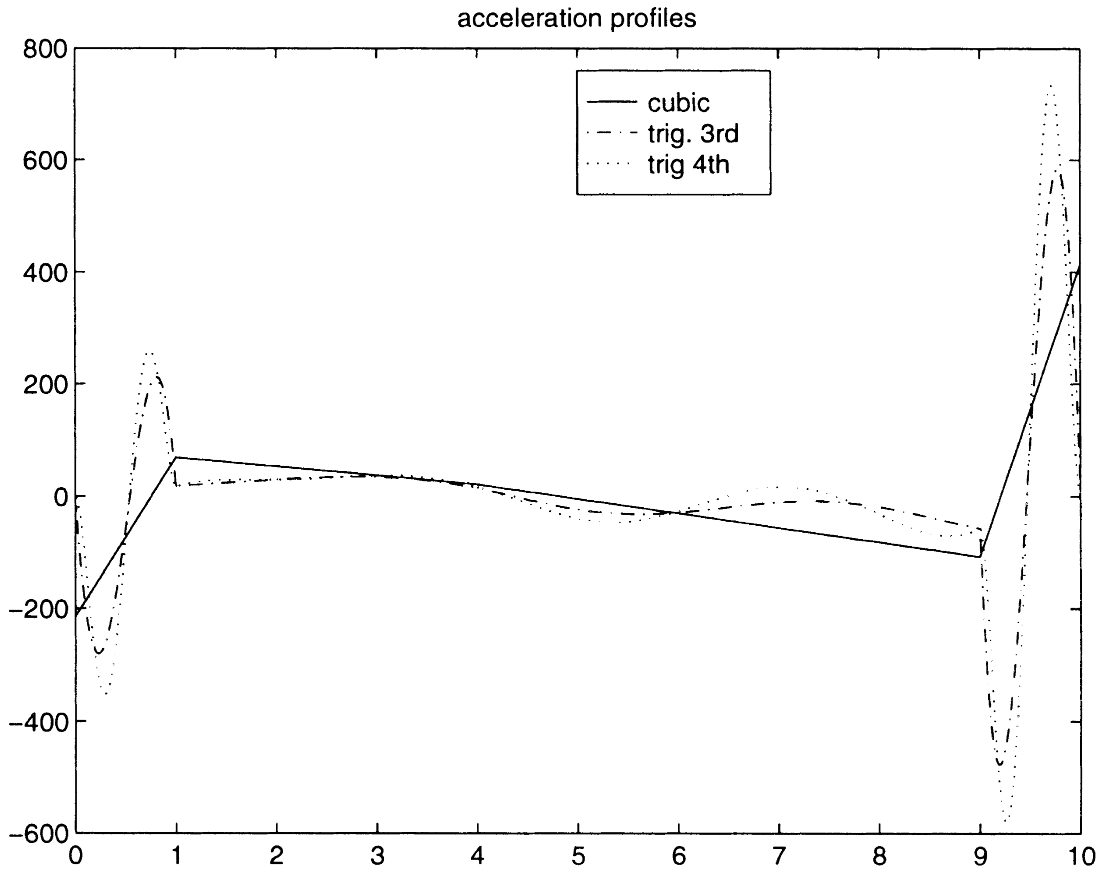

Fig. 7. Acceleration functions for the cubic and trigonometric splines. Example II.

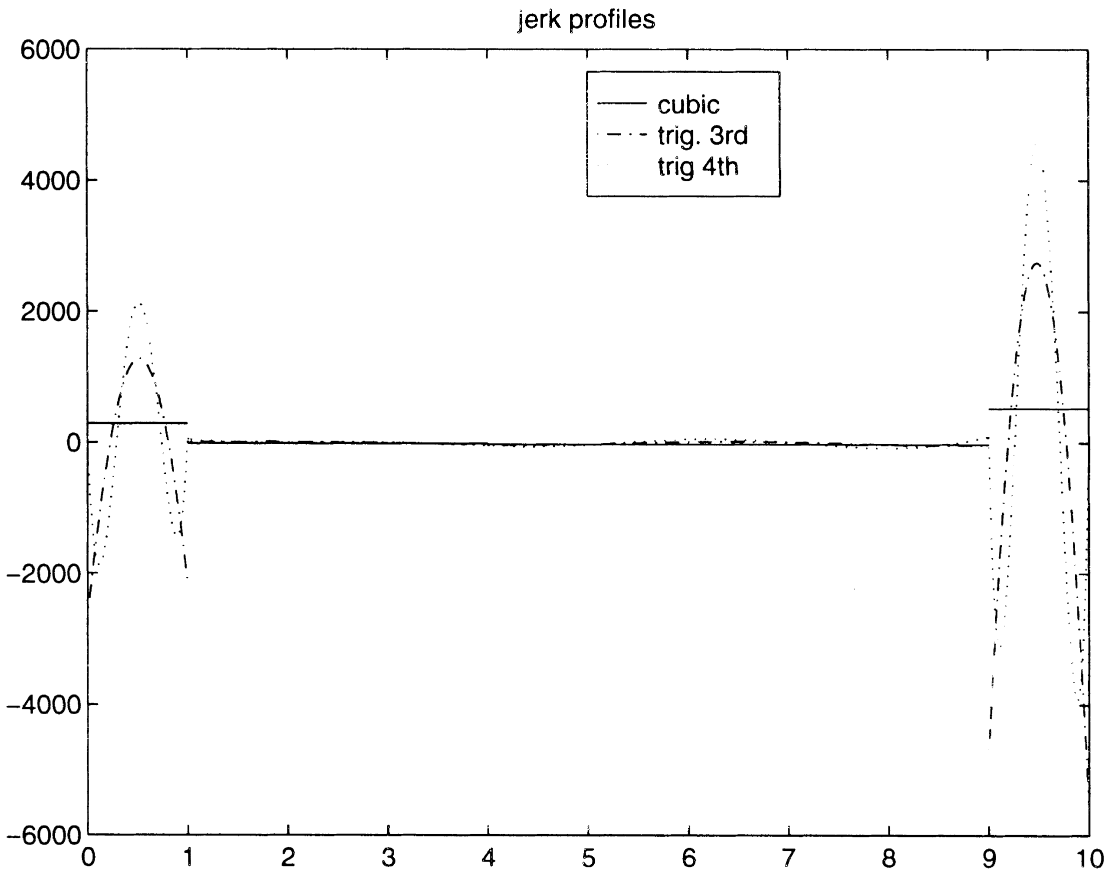

Fig. 8. Jerk functions for the cubic and trigonometric splines. Example II.

https://doi.org/10.1017/S0263574700002721 Published online by Cambridge University Press

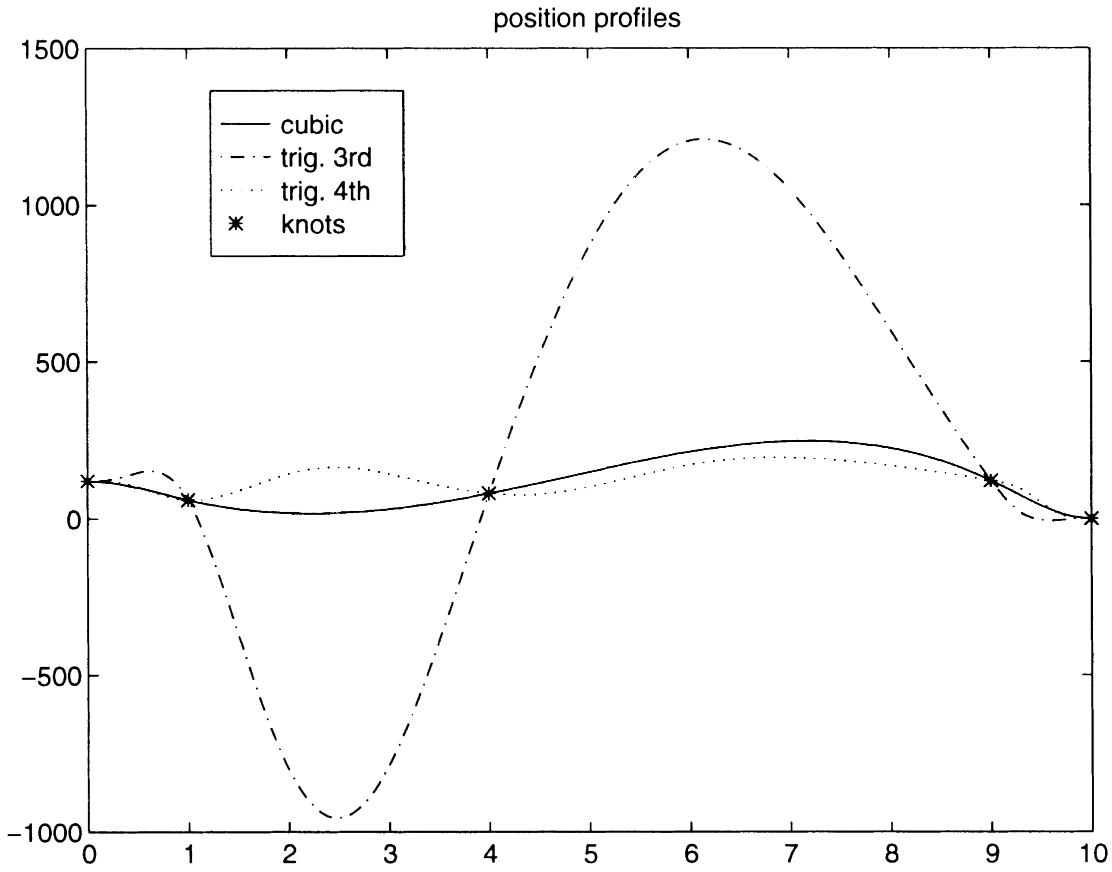

Fig. 9. Position functions for the cubic and trigonometric splines. Example III.

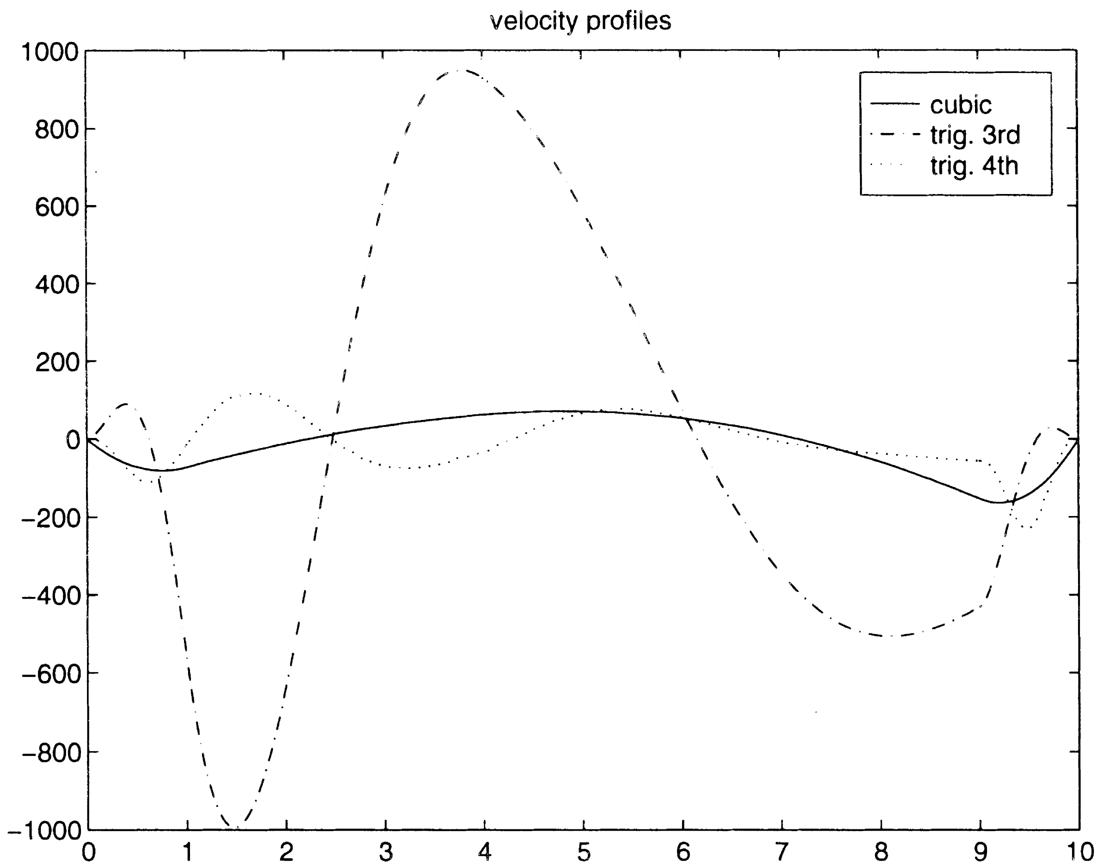

Fig. 10. Velocity functions for the cubic and trigonometric splines. Example III.

https://doi.org/10.1017/S0263574700002721 Published online by Cambridge University Press

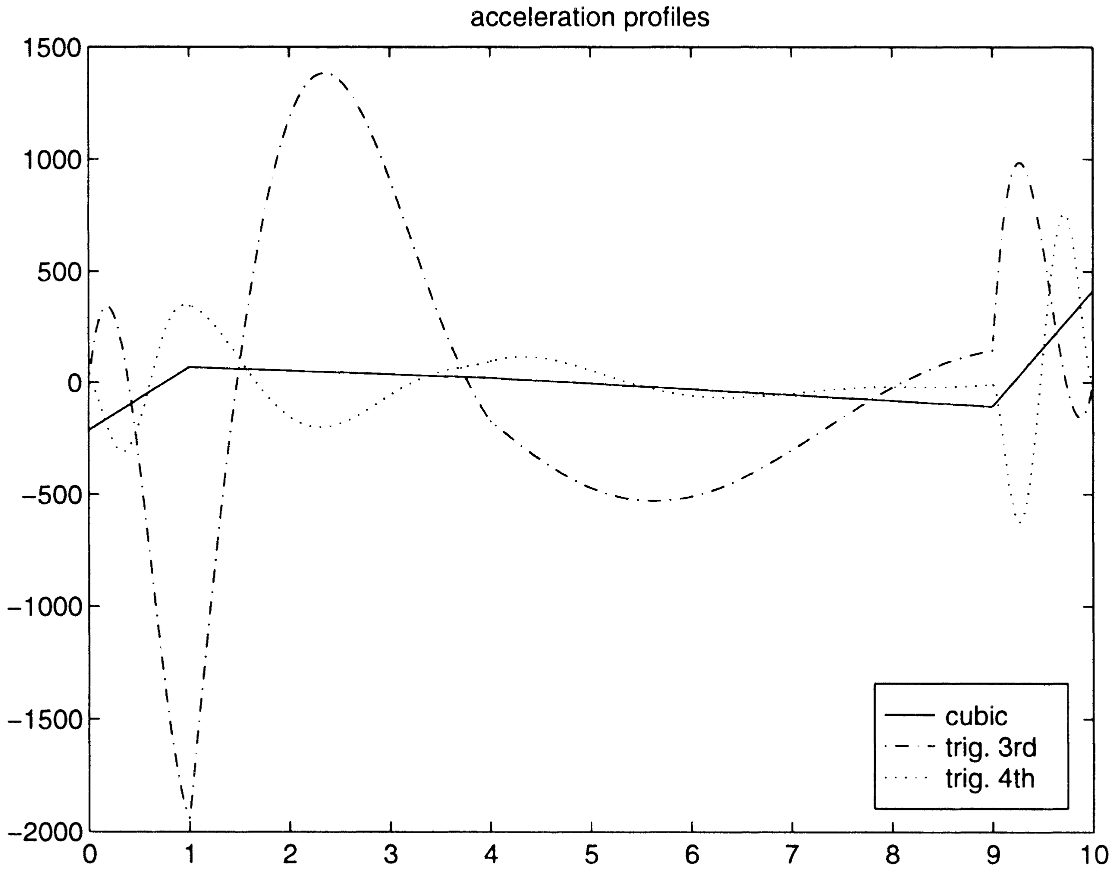

Fig. 11. Acceleration functions for the cubic and trigonometric splines. Example III.

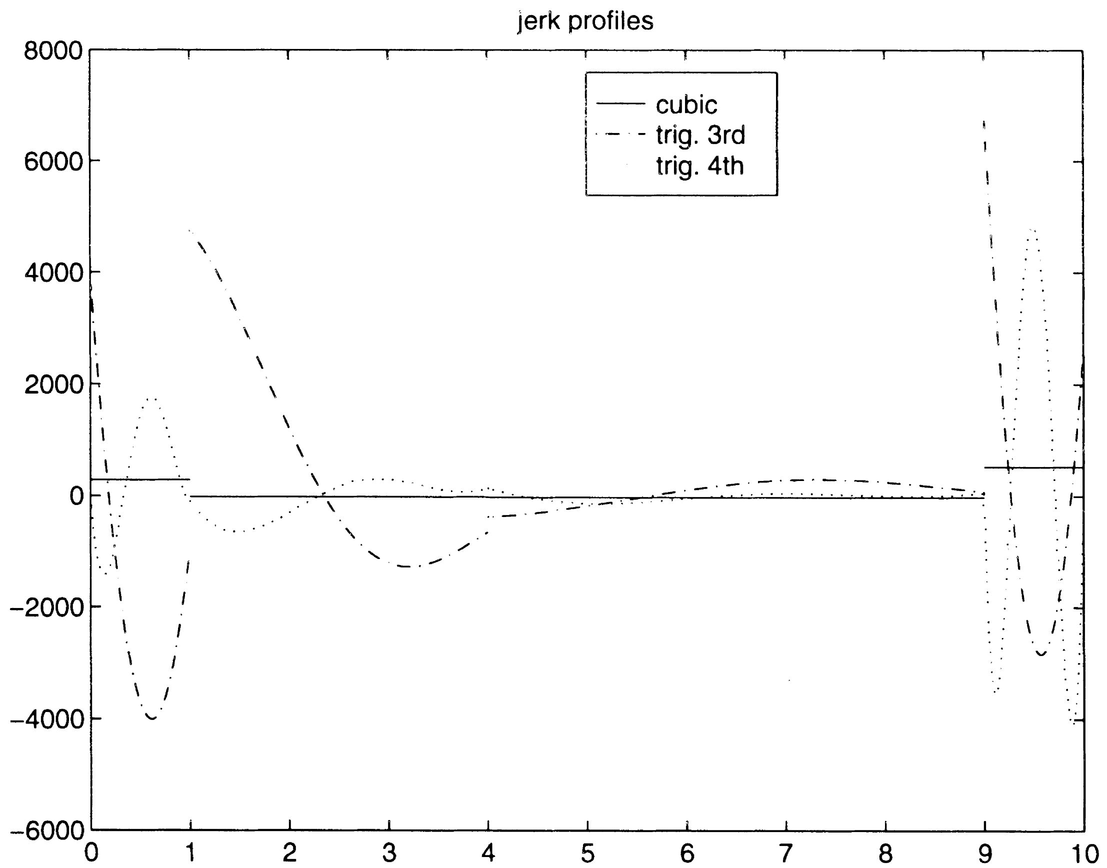

Fig. 12. Jerk functions for the cubic and trigonometric splines. Example III.

https://doi.org/10.1017/S0263574700002721 Published online by Cambridge University Press

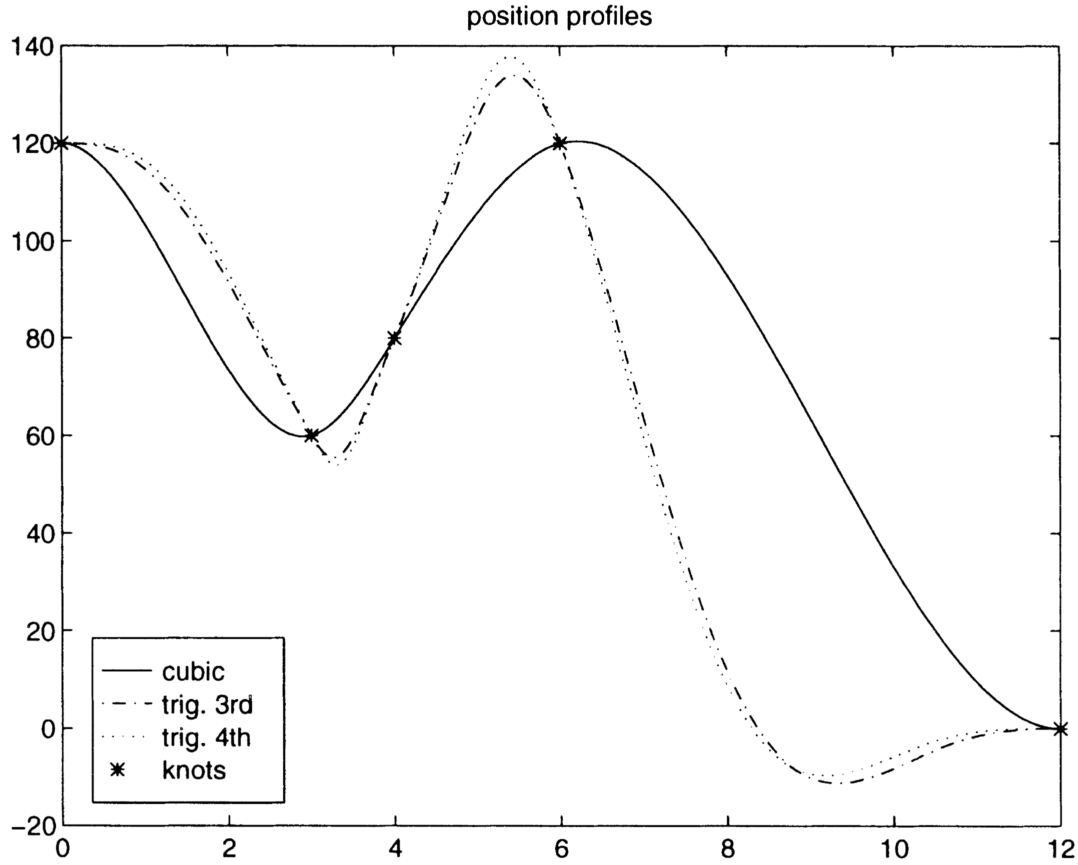

Fig. 13. Position functions for the cubic and trigonometric splines. Example IV.

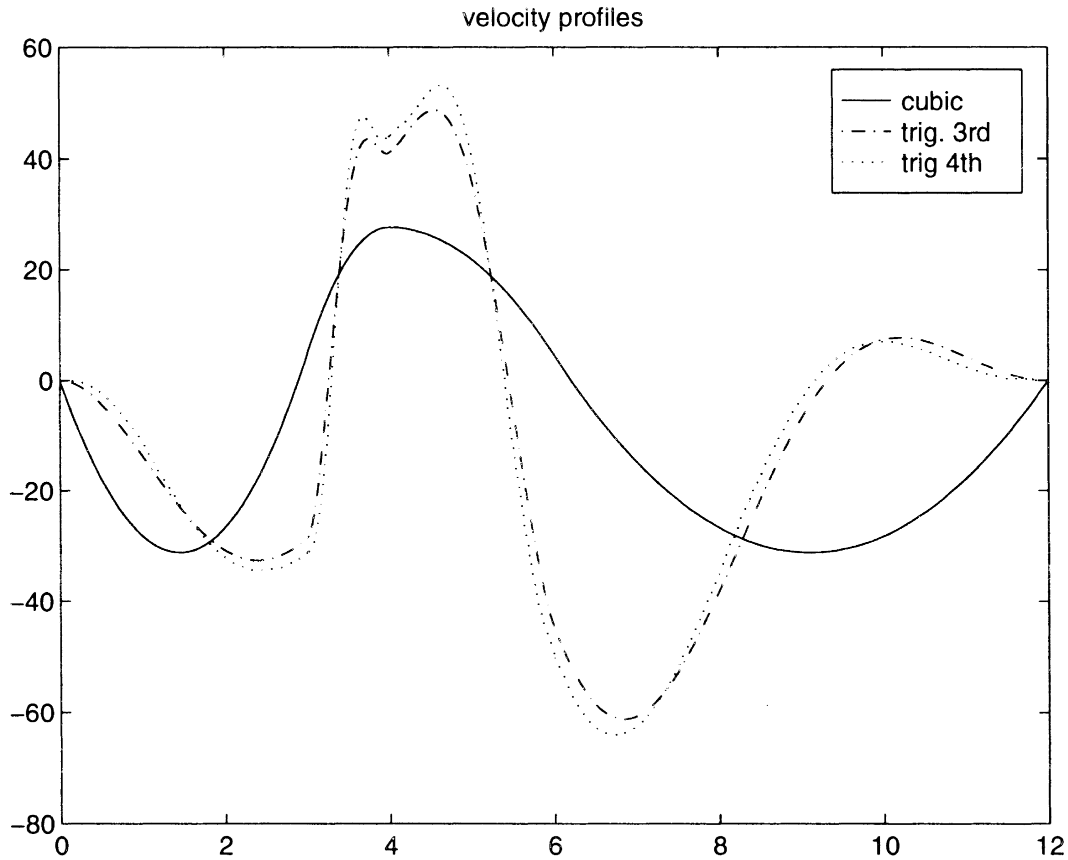

Fig. 14. Velocity functions for the cubic and trigonometric splines. Example IV.

https://doi.org/10.1017/S0263574700002721 Published online by Cambridge University Press

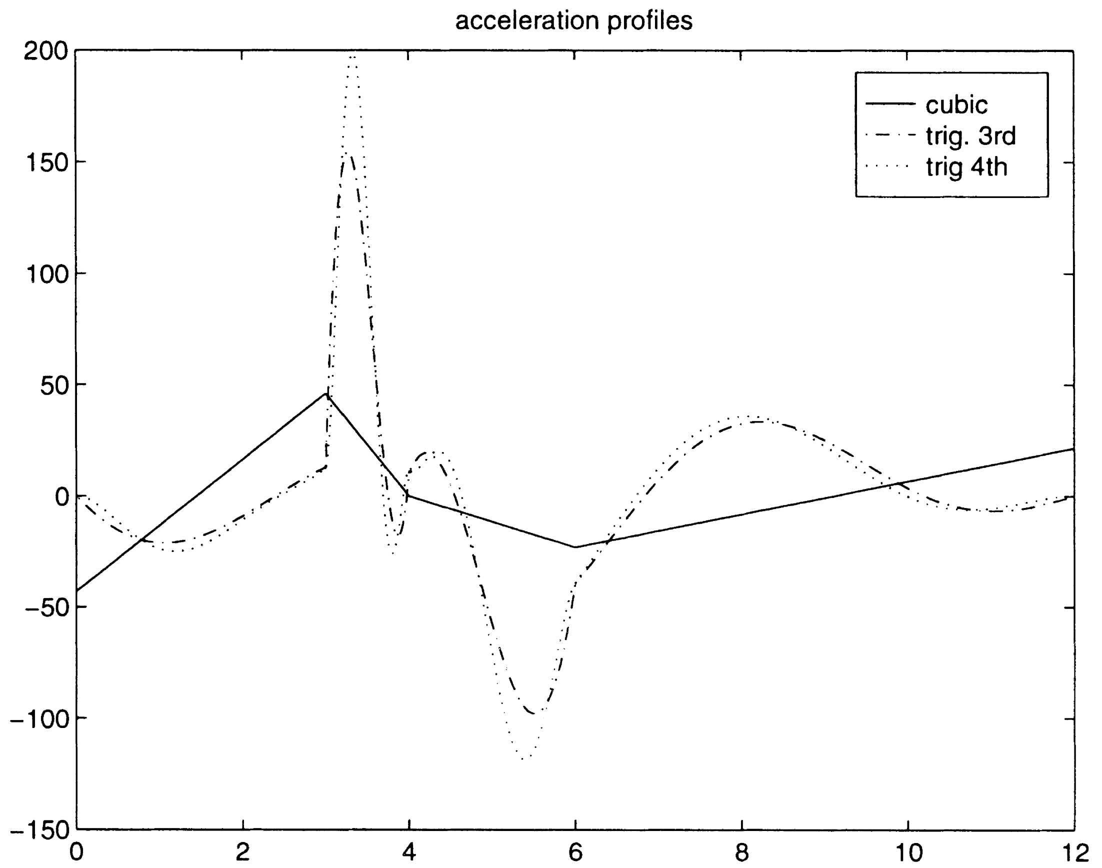

Fig. 15. Acceleration functions for the cubic and trigonometric splines. Example IV.

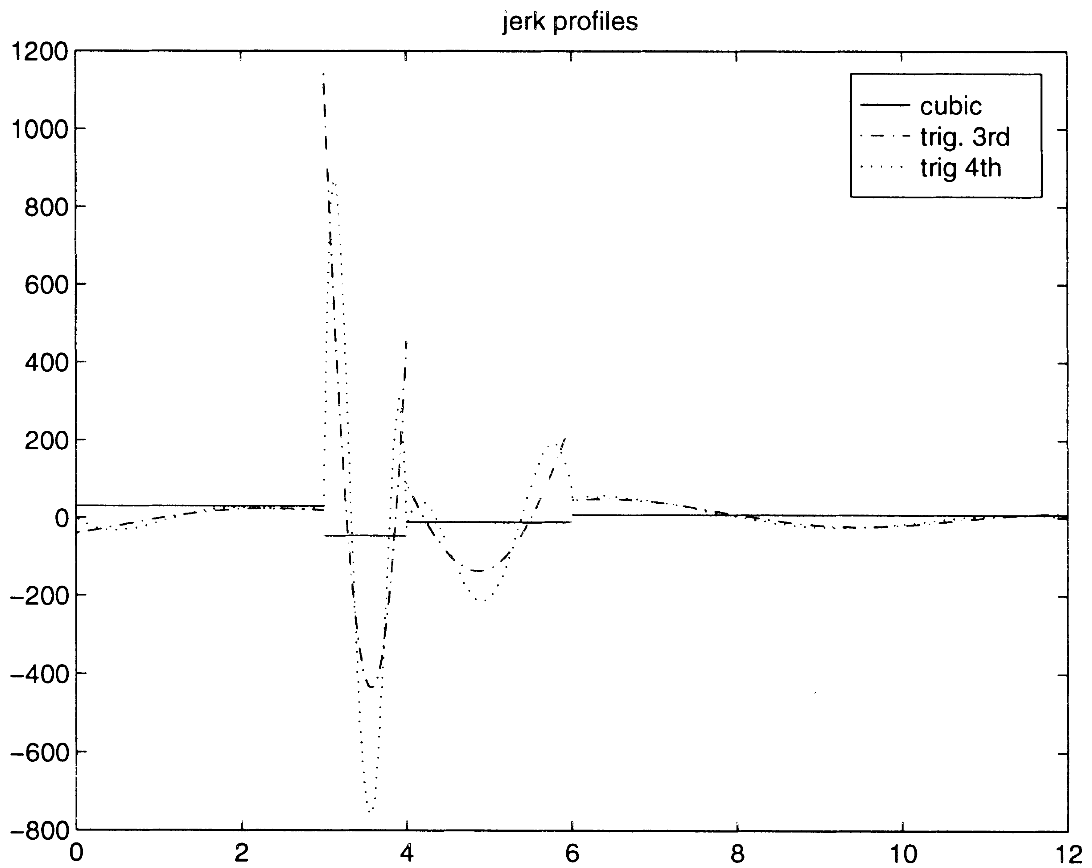

Fig. 16. Jerk functions for the cubic and trigonometric splines. Example IV.

https://doi.org/10.1017/S0263574700002721 Published online by Cambridge University Press

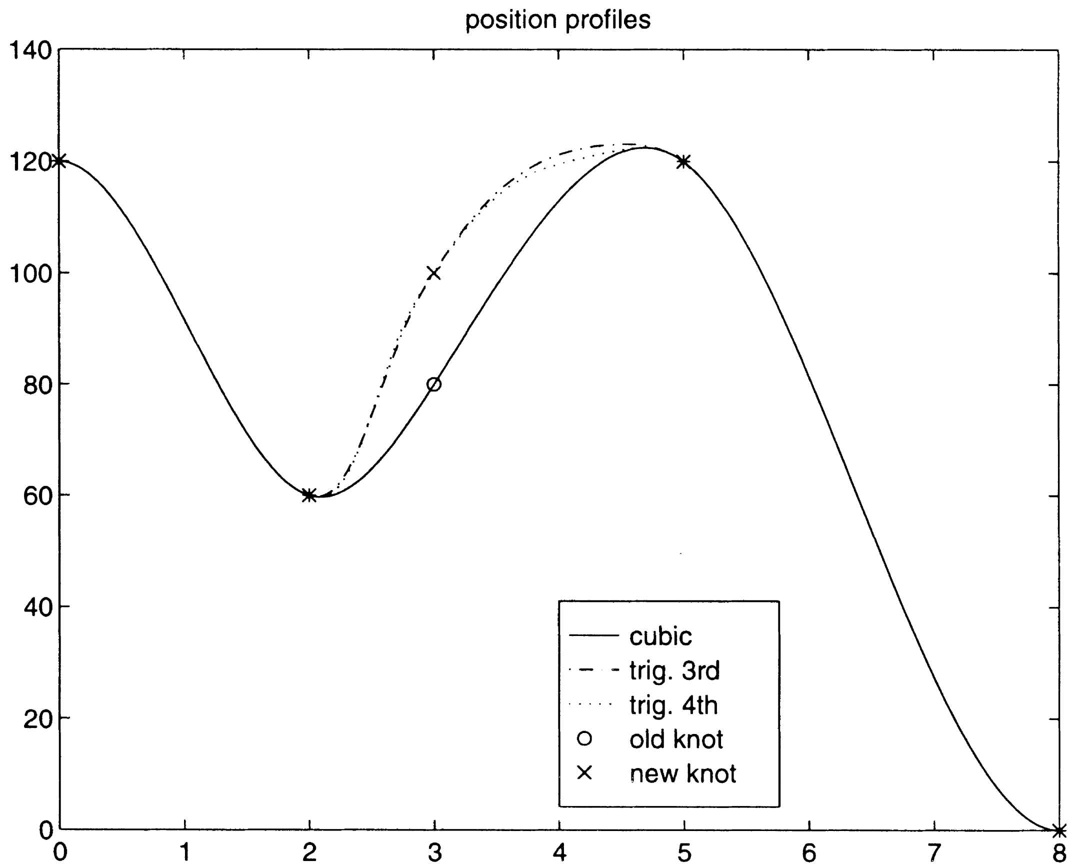

Fig. 17. Position functions when the original trajectory is modified. Example V.

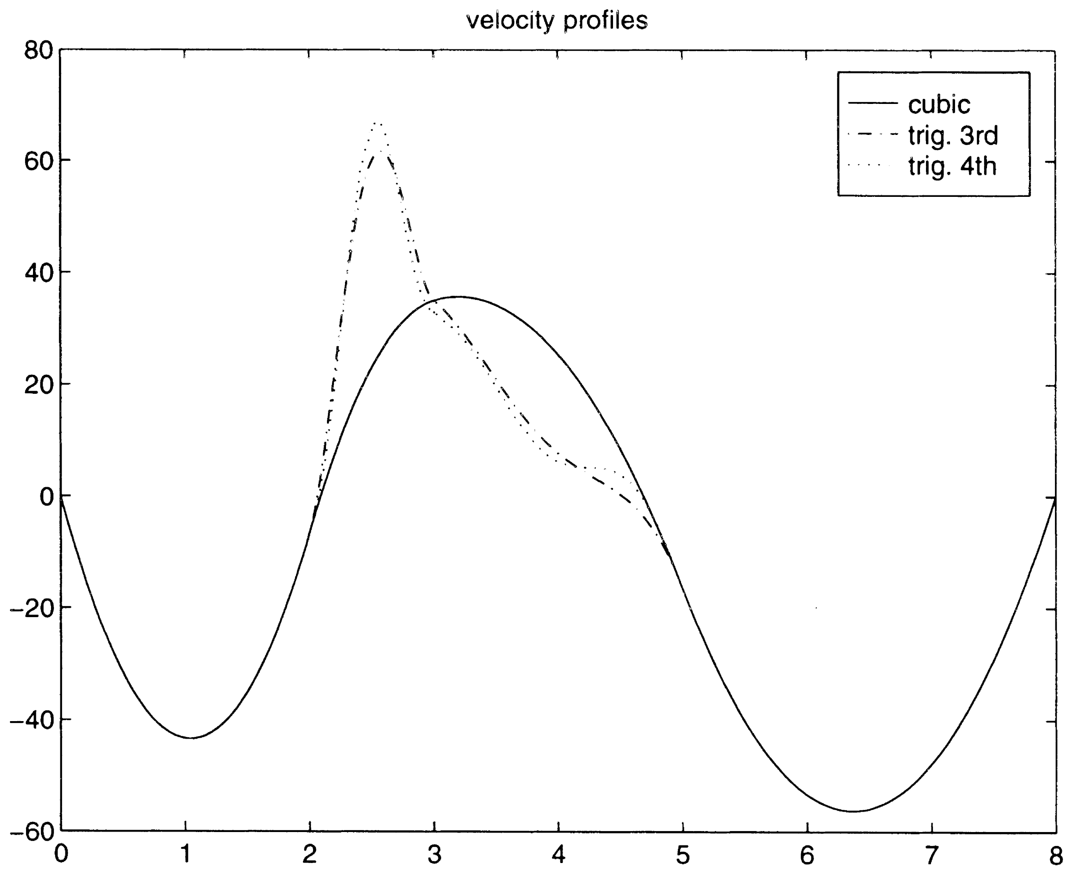

Fig. 18. Velocity functions in case the original trajectory is modified. Example V.

https://doi.org/10.1017/S0263574700002721 Published online by Cambridge University Press

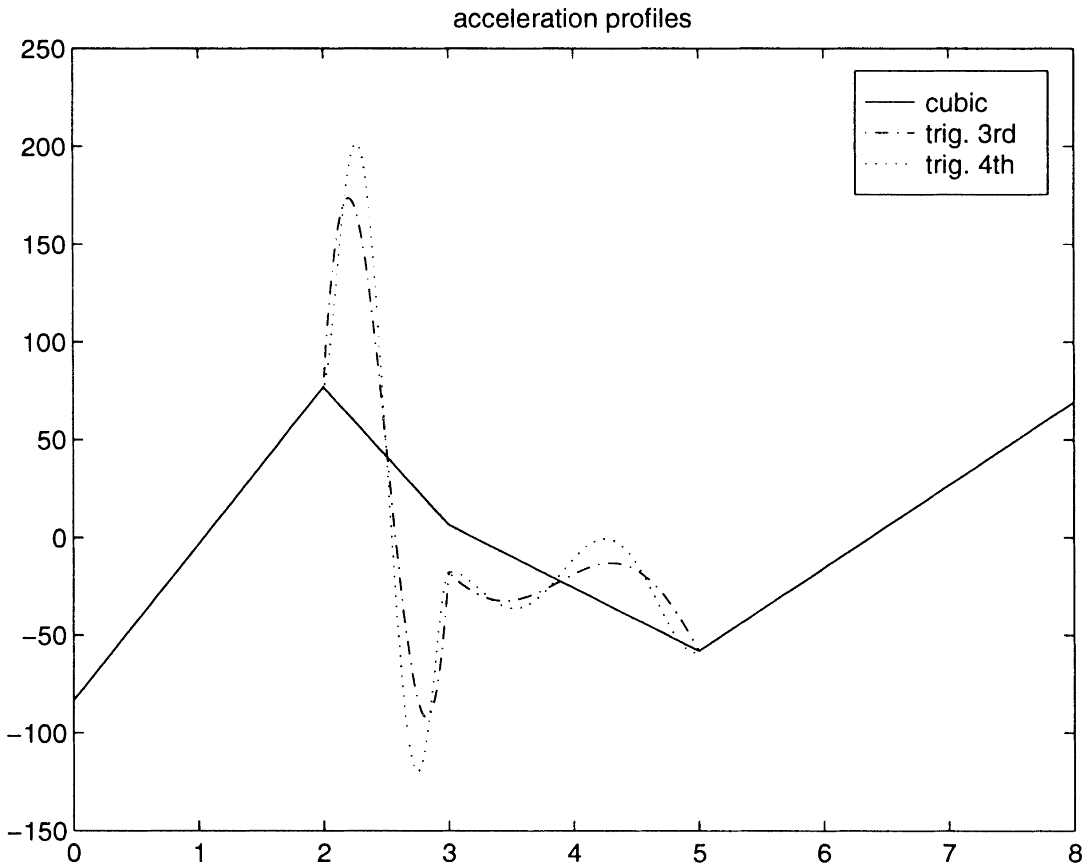

Fig. 19. Acceleration functions when the original trajectory is modified. Example V.

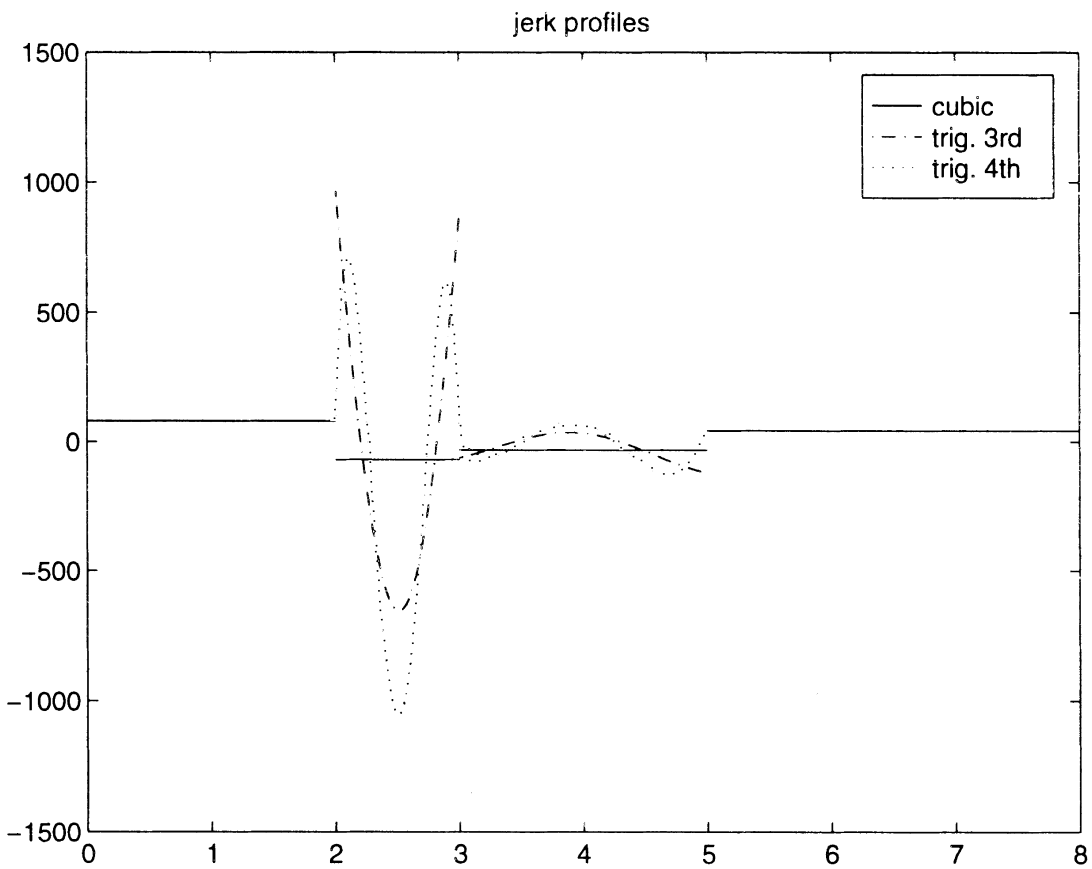

Fig. 20. Jerk functions in case the original trajectory is modified. Example V.

https://doi.org/10.1017/S0263574700002721 Published online by Cambridge University Press

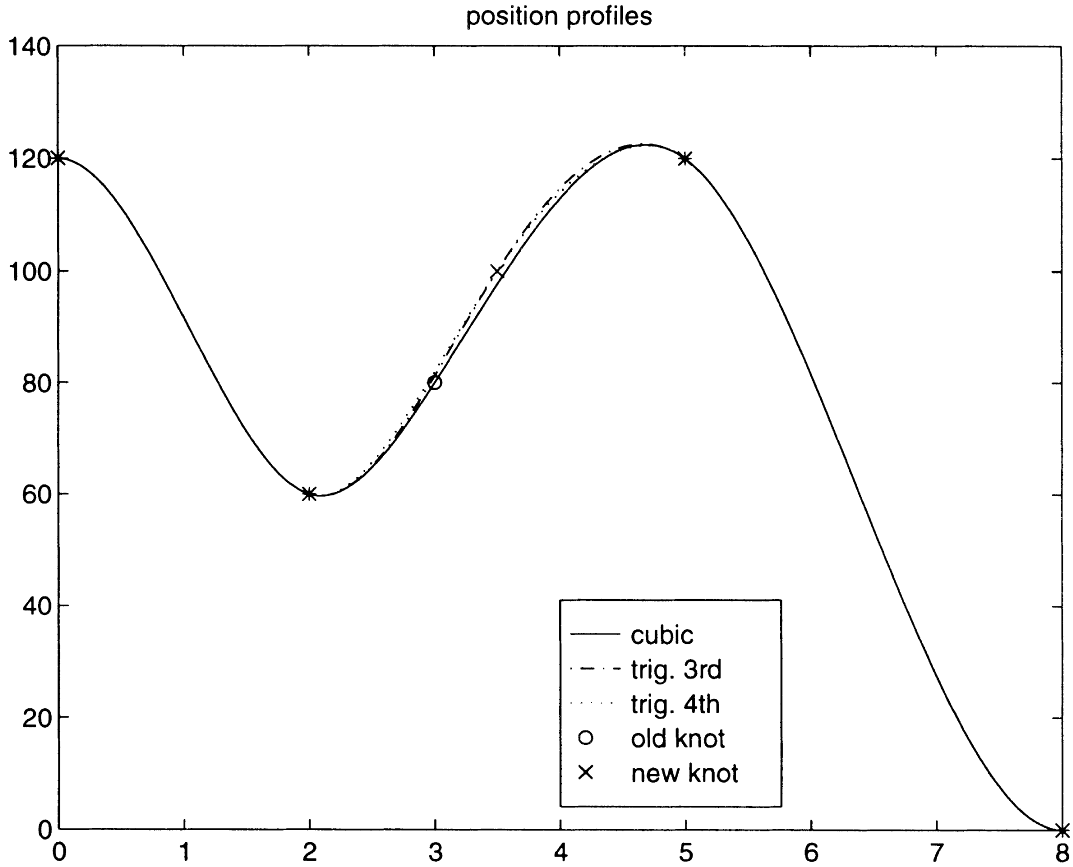

Fig. 21. Position functions in case the original trajectory is modified. Example VI.

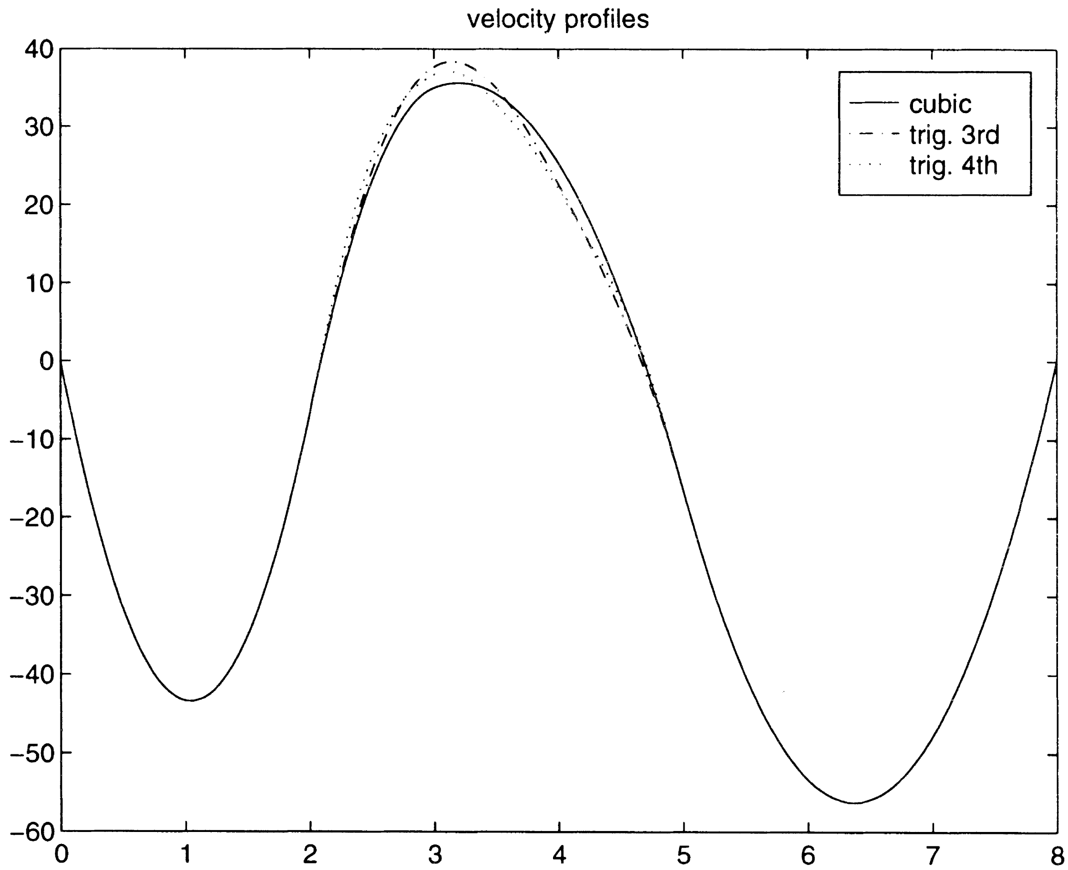

Fig. 22. Velocity functions in case the original trajectory is modified. Example VI.

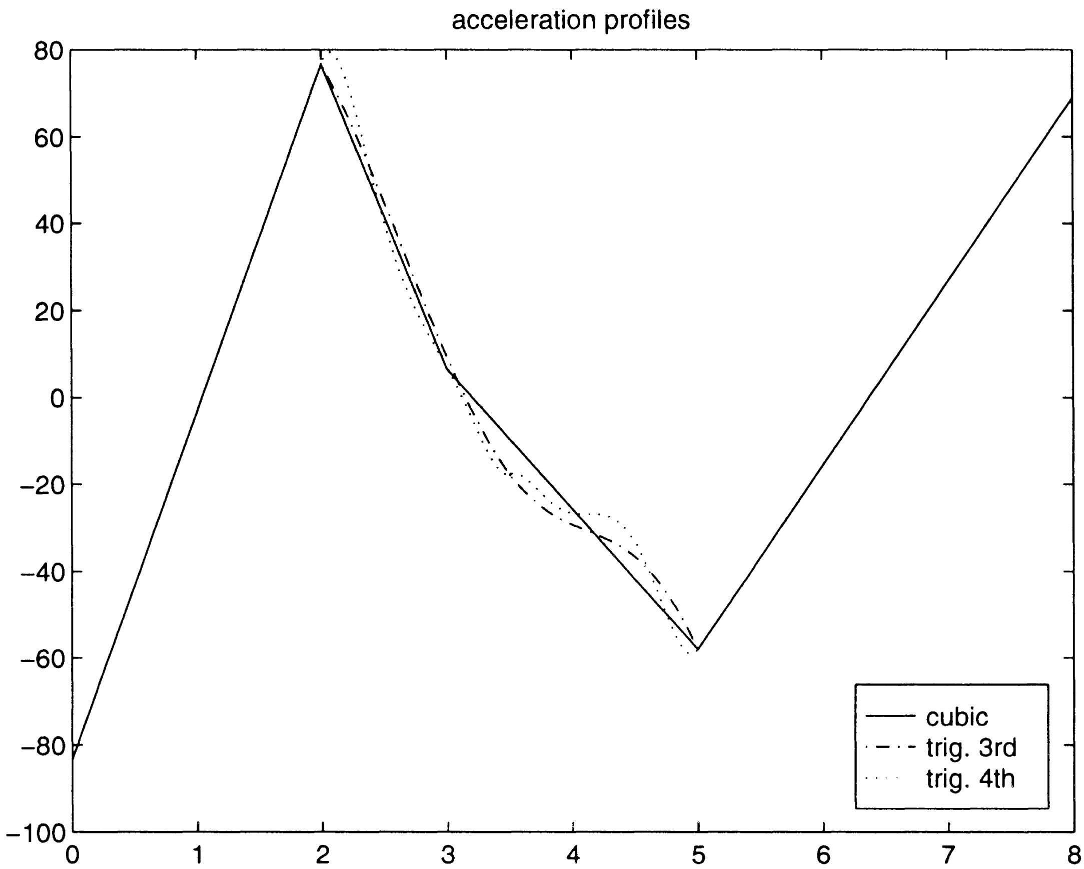

Fig. 23. Acceleration functions when the original trajectory is modified. Example VI.

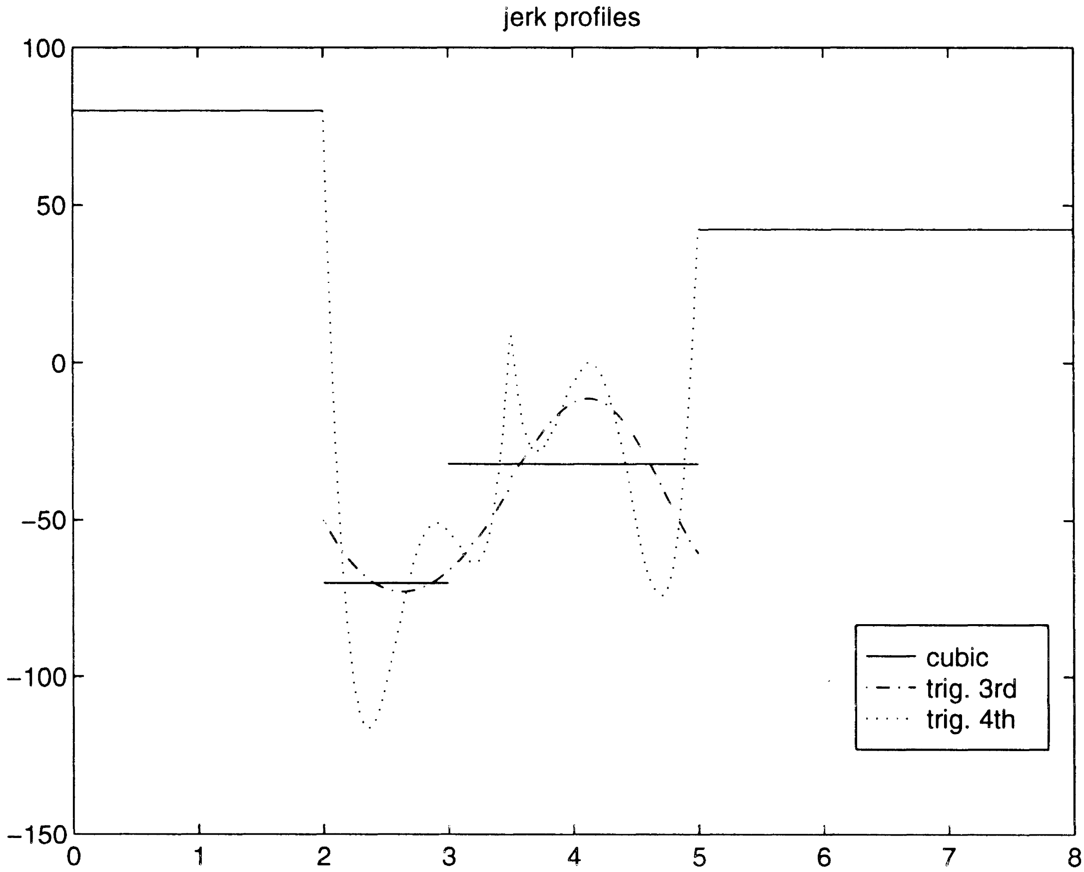

Fig. 24. Jerk functions in case the original trajectory is modified. Example VI.

https://doi.org/10.1017/S0263574700002721 Published online by Cambridge University Press

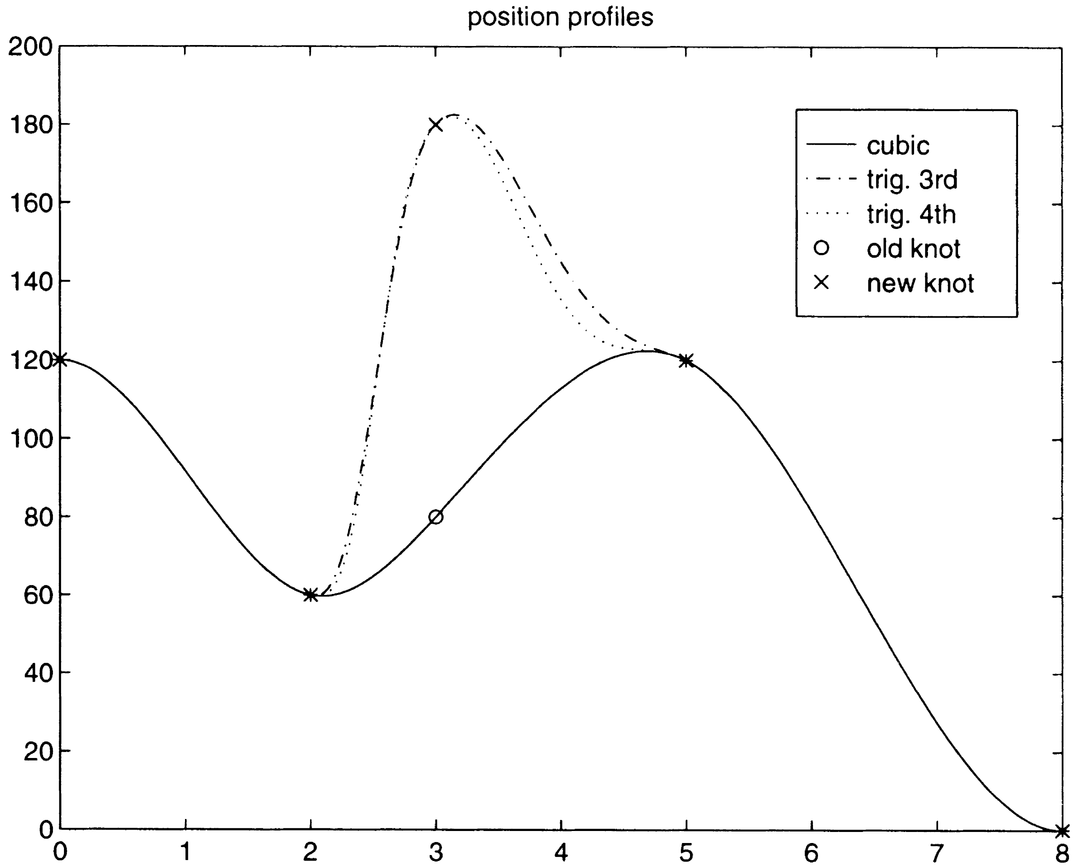

Fig. 25. Position functions in case the original trajectory is modified. Example VII.

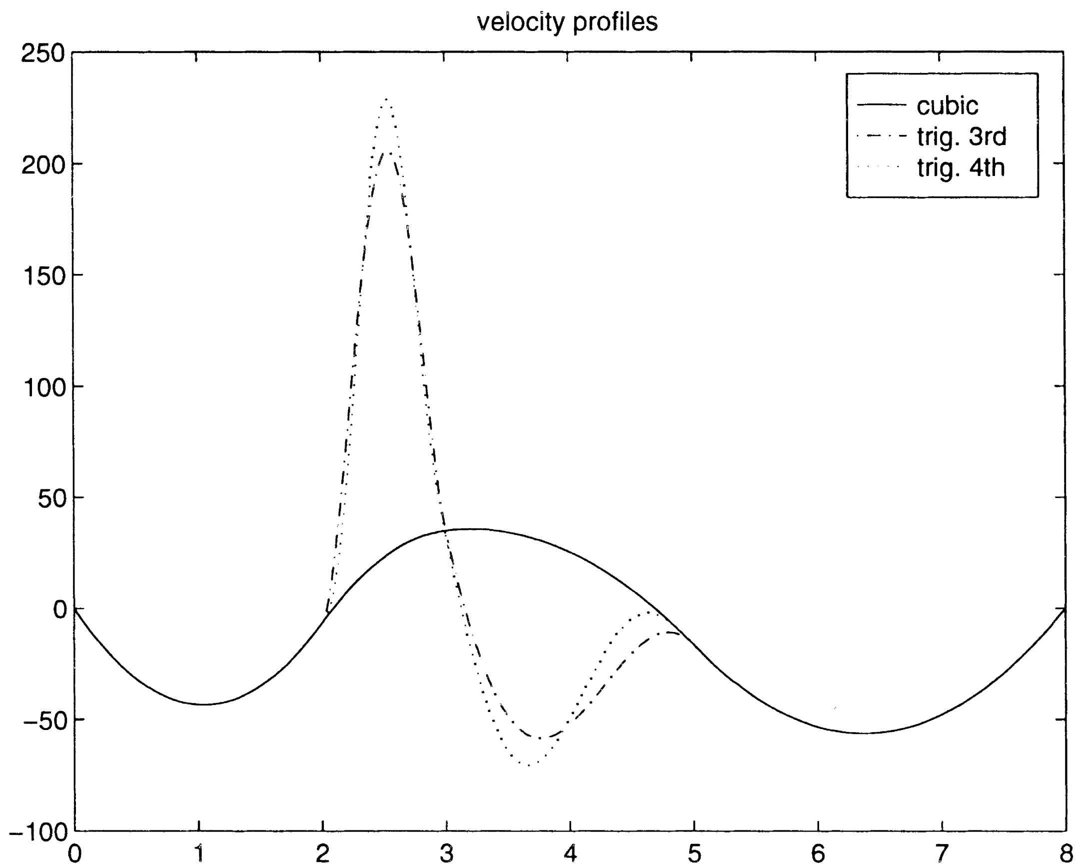

Fig. 26. Velocity functions in case the original trajectory is modified. Example VII.

https://doi.org/10.1017/S0263574700002721 Published online by Cambridge University Press

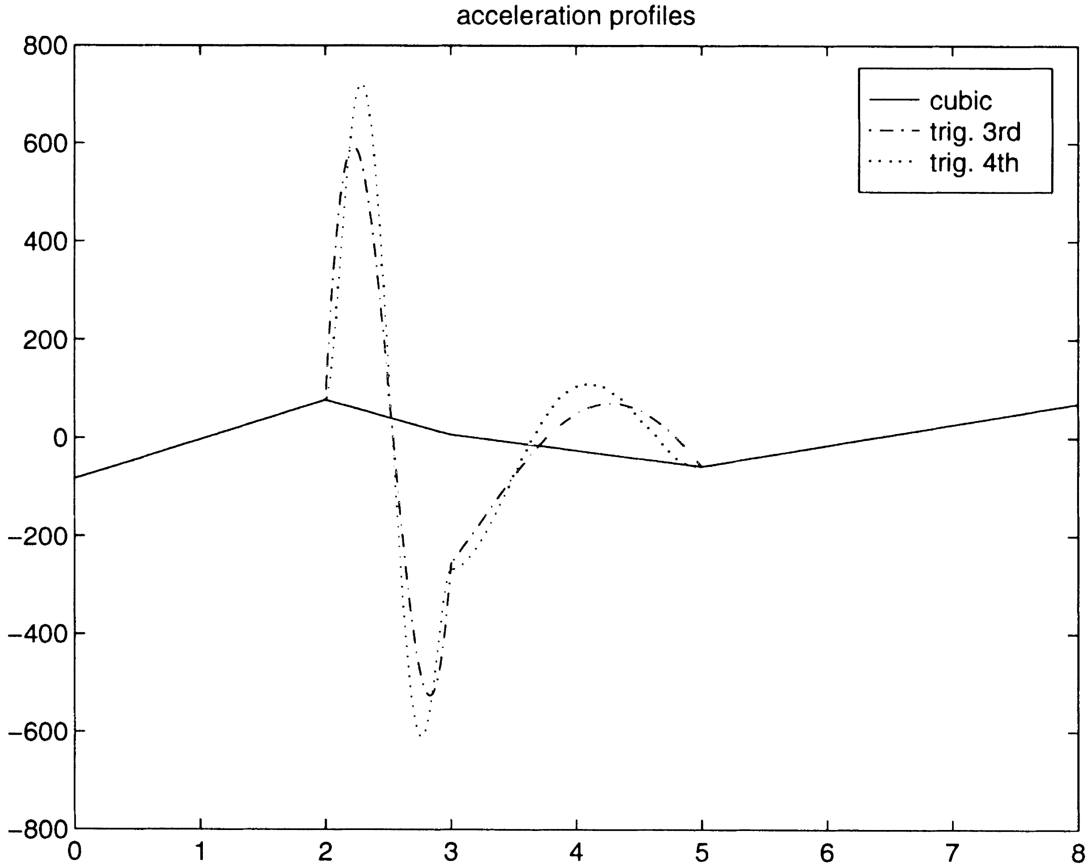

Fig. 27. Acceleration functions when the original trajectory is modified. Example VII.

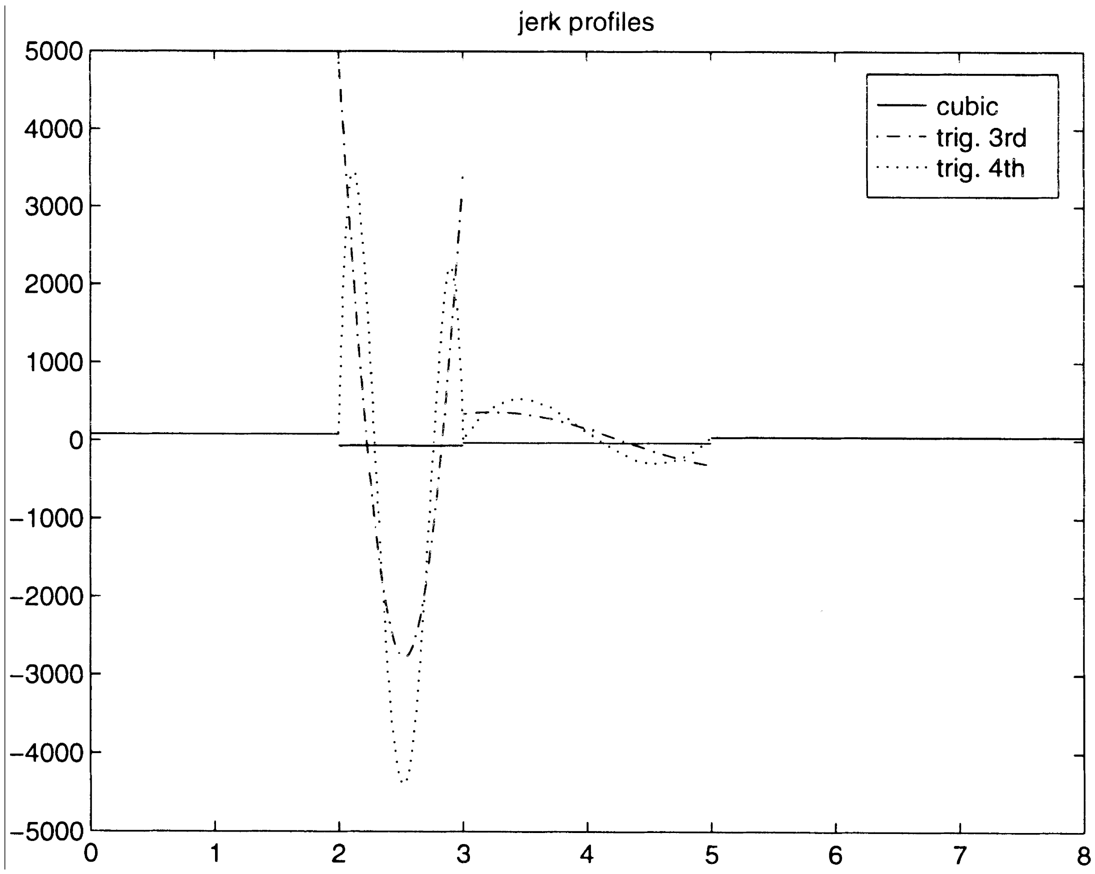

Fig. 28. Jerk functions in case the original trajectory is modified. Example VII.

https://doi.org/10.1017/S0263574700002721 Published online by Cambridge University Press

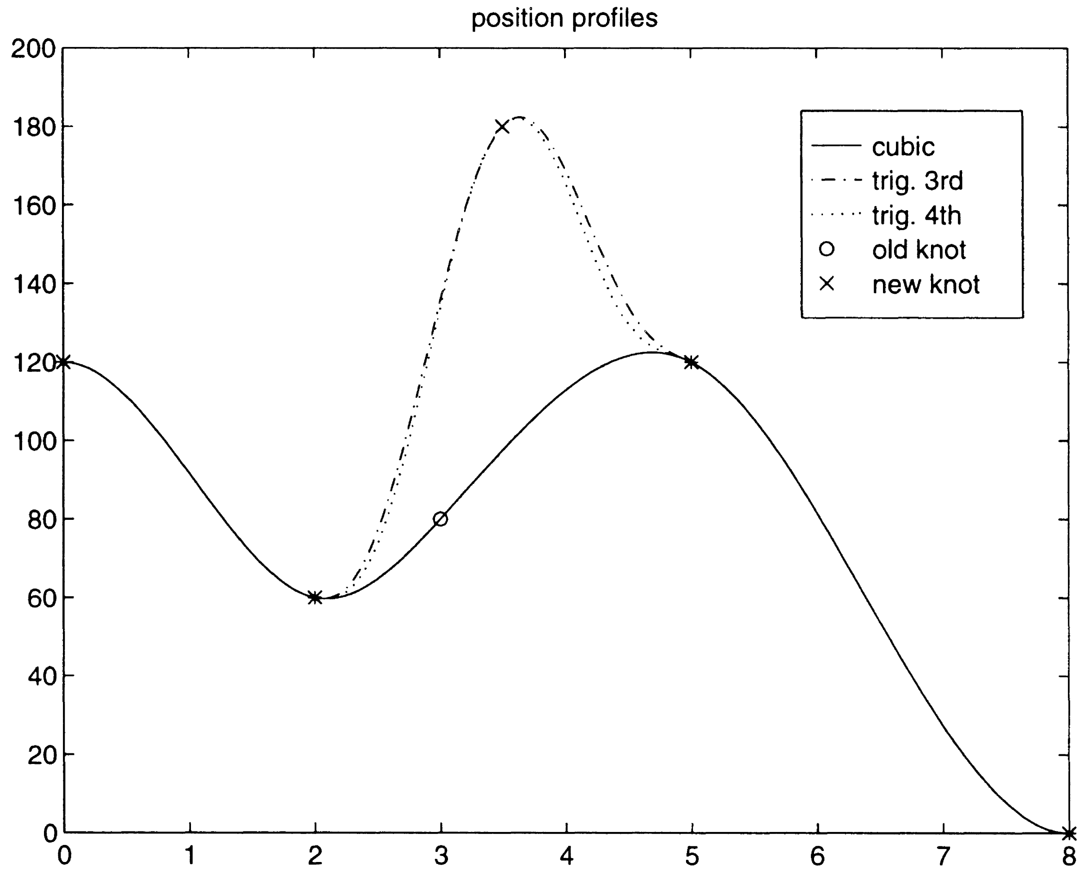

Fig. 29. Position functions in case the original trajectory is modified. Example VIII.

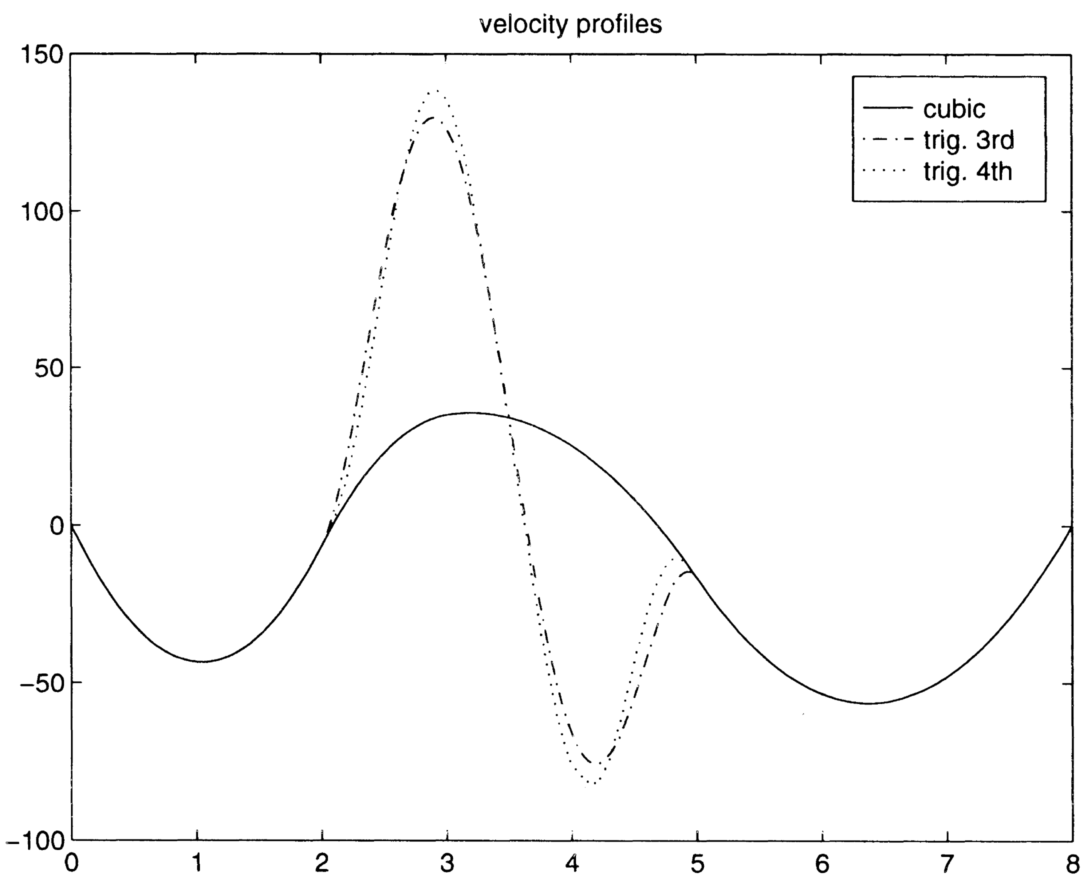

Fig. 30. Velocity functions in case the original trajectory is modified. Example VIII.

https://doi.org/10.1017/S0263574700002721 Published online by Cambridge University Press

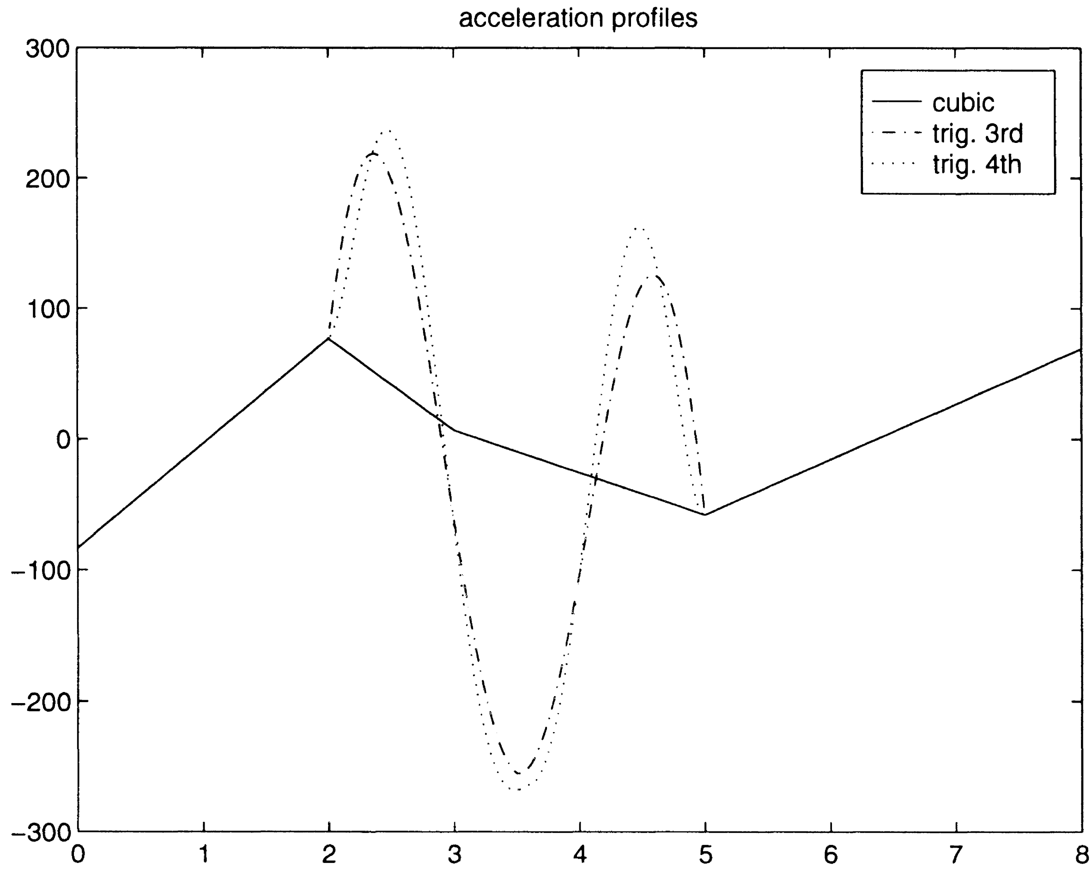

Fig. 31. Acceleration functions when the original trajectory is modified. Example VIII.

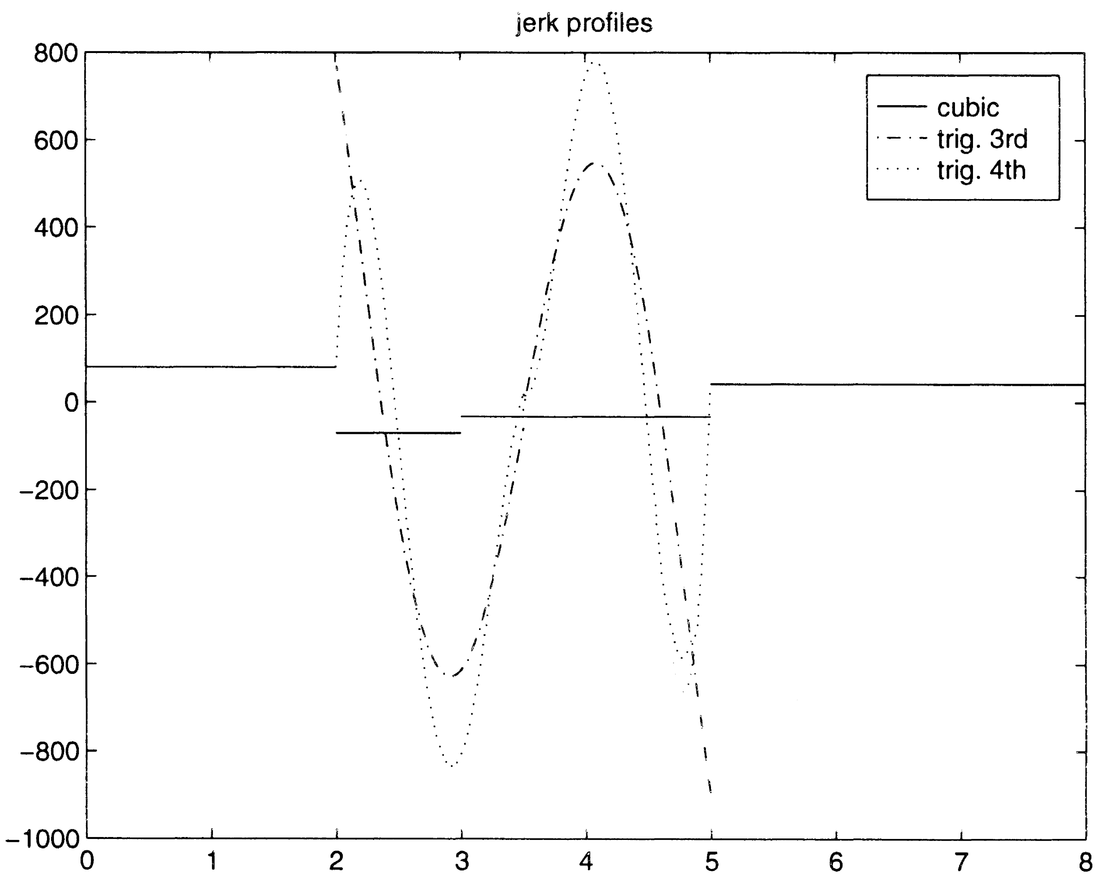

Fig. 32. Jerk functions in case the original trajectory is modified. Example VIII.

https://doi.org/10.1017/S0263574700002721 Published online by Cambridge University Press

- in general, for a given trajectory it cannot be said a priori if third-order trigonometric splines are better than the fouth-order ones. Thus, for each case it is convenient to evaluate both methods before choosing the one to be applied.

4. COMBINED USE OF CUBIC AND TRIGONOMETRIC SPLINES

It has been said that one of the most important features of the trigonometric splines is the practical feasability to fix the constraint values at the knots and that therefore they seem particularly suitable to implement real-time obstacle avoidance algorithms. In this paper, more than the use of a full trigonometric splines framework for this purpose, it is suggested to employ a mixed algebraic/trigonometric approach.

We assume to have a number of via points to pass through and that a trajectory has been already planned using cubic splines. Then, while the robot is moving, the value of a particular knot is changed, before the robot passes through it. In other words, the trajectory planner substitutes one knot with another one, e.g. if an unexpected obstacle is present in the original trajectory and the robot might collide with it. The new knot can be determined by an obstacle-avoidance algorithm. The discussion of these kind of algorithms is beyond the aims of this paper and therefore it will be assumed that the value of the new knot has already been somehow determined.

Specifically, we suppose that the original ith knot value $q_i := q_i^{old}$ is replaced with knot q_i^{new}. In this context, it seems reasonable to adopt trigonometric splines to connect knots q_{i-1}, q_i^{new} and q_{i+1}, in order to preserve the continuity of the acceleration reference function, without providing large overshoots (which would occur if we employ algebraic splines since two polynomials of the 5th order are needed), which, clearly, would very likely cause a collision. The use of a third order trigonometric spline means that the continuity of the jerk function is not obtained. Conversely, this can be imposed by employing fourth order trigonometric splines.

5. ILLUSTRATIVE EXAMPLES

In this section, a few examples are given to illustrate the methodology previously described. The original intermediate points are the same as the ones considered in Section 3. Here we impose different spline times, that is $\mathbf{h} = [2, 1, 2, 3]$. First, we consider a relatively small change that occurs in the third knot, for which the value of 80 degrees is replaced by 100 degrees. Results (example V) are reported in Figures 17–20, where both third and fourth order trigonometric splines have been considered. It has to be stressed that in this case the spline times are left unchanged. It is interesting to compare this case with the one in which, taking into account the results exposed in Section 4, the time intervals related to the trigonometric splines are made equal to each other, i.e. we set $t_2 = 1.5$ and $t_3 = 1.5$ (note that in the original trajectory it is $t_2 + t_3 = 3$). In other words, in avoiding the obstacle, the trajectory is locally modified without modifying anything of the other parts of the motion that are not involved by the presence of the obstacle. Results related to this case (example VI) are plotted in Figures 21–24.

In order to complete the analysis, a case in which a big change in the trajectory occurs is considered. Specifically, the third knot is replaced with $q_2^{new} = 180$ degrees. Results concerning the case in which the spline times are left unaltered (example VII) are shown in Figures 25–28, while the ones in which the time intervals are equalized are exposed in Figures 29–32.

It is worth noting that, in the above figures, the new (trigonometric) functions have not to be compared with the old (algebraic) ones since, of course the position reference signal is different by force of circumstance. In any case, the original trajectory has been left in the figures for the sake of clarity.

From the above results, it comes out that the proposed scheme is effective in each situation, but it is highly preferable to equalize the spline times, as in these cases the required actuators' effort is much lower. No significant differences emerge between third and fourth order trigonometric splines, the latter generally leading to a slightly higher value of the maximum of the acceleration profile.

Finally, it has to be stressed that the limits of the actuators and of the mechanical structure of the manipulator have not been taken into account in the performed analysis. Obviously, if the limits are exceeded in any part of the calculated motion, it is necessary to increase some of the spline times or to select different via points. This task has to be accomplished in general by a higher level of the trajectory planner, which can exploit the combined use of cubic and trigonometric splines in a specific function.

6. CONCLUSIONS

In this paper, the use of algebraic and trigonometric splines for the trajectory planning or robot manipulators have been analyzed. Specifically, the works of Simon and Isik have been significantly improved by making a fair comparison between the two kinds of splines and by discussing the use of trigonometric splines of different order and with different time intervals. As a result, a detailed framework about the use of such methods has emerged. Moreover, a very useful technique, which combines the use of both cubic and trigonometric splines, has been proposed and investigated.

References

1. J.J. Craig, *Introduction to Robotics: Mechanics and Control*, Second edition (Addison-Wesley, New York, 1989).
2. C.-S. Lin, P.-R. Chang and J.Y.S. Luh, "Formulation and optimization of polynomial joint trajectories for industrial robots", *IEEE Trans. on Automatic Control* **28**, No. 12, 1066–1074 (1983).
3. A. Piazzi and A. Visioli, "Global minimum-time trajectory planning of mechanical manipulators using interval analysis", *Int. J. Control* **71**, No. 4, 631–652 (1998).
4. A. Piazzi and A. Visioli, "Global minimum-jerk trajectory planning of robot manipulators", *IEEE Transactions on Industrial Electronics* **47**, No. 1, 140–143 (Feb., 2000).
5. I. Schoenberg, "On trigonometric spline interpolation", *J. Mathematics and Mechanics* **13**, 795–825 (1964).

https://doi.org/10.1017/S0263574700002721 Published online by Cambridge University Press

6. D. Simon and C. Isik, “Optimal trigonometric robot joint trajectories”, *Robotica* **9**, Part 4, 379–386 (1991).
7. D. Simon and C. Isik, “Computational complexity and path error analyses of trigonometric joint trajectories”, *Int. J. Robotics and Automation* **7**, No. 4, 179–185 (1992).
8. D. Simon and C. Isik, “Suboptimal robot joint interpolation within user-specified knot tolerances”, *J. Robotic Systems* **10**, No. 7, 889–911 (1993).
9. D. Simon and C. Isik, A trigonometric trajectory generator for robotic arms”, *Int. J. Control* **57**, No. 3, 505–517 (1993).
10. D. Simon and C. Isik, “Efficient Cartesian path approximation for robots using trigonometric splines”, *Proc. American Control Conference*, Baltimore (ML) (1994) pp. 1752–1756.
11. S. Chand and K.L. Doty, “On-line polynomial trajectories for robot manipulators”, *Int. J. Robotics Research* **4**, No. 2, 38–48 (1985).