



Science Arts & Métiers (SAM)

is an open access repository that collects the work of Arts et Métiers Institute of Technology researchers and makes it freely available over the web where possible.

This is an author-deposited version published in: <https://sam.ensam.eu>
Handle ID: [.http://hdl.handle.net/10985/11402](http://hdl.handle.net/10985/11402)

To cite this version :

Pierre BESSET, Richard BEAREE, Olivier GIBARU - FIR Filter-Based Online Jerk-Controlled Trajectory Generation - In: IEEE International Conference on Industrial Technology (ICIT), Taiwan, 2016 - 2016 IEEE International Conference on Industrial Technology (ICIT) - 2016

Any correspondence concerning this service should be sent to the repository

Administrator : scienceouverte@ensam.eu



FIR Filter-Based Online Jerk-Controlled Trajectory Generation

Pierre Besset

Arts et Metiers ParisTech / LSIS
8 Bd Louis XIV, 59000 France
Email: pierre.besset@ensam.eu

Richard Béarée

Arts et Metiers ParisTech / LSIS
Email: richard.bearee@ensam.eu

Olivier Gibaru

Arts et Metiers ParisTech / LSIS
Email: olivier.gibaru@ensam.eu

Abstract—This paper presents a novel approach to generate online jerk-limited trajectories for multi-DOF robotic systems. Finite Impulse Response filters are used to efficiently turn low computational cost acceleration-limited profiles into jerk-limited profiles. Starting from a new setpoint, e.g. an event given by external sensors, and an arbitrary state of motion, i.e. with non-zero initial velocity and acceleration values, the proposed method can generate different shapes of jerk profile, including time-optimal and fixed-time jerk-limited trajectories. A new definition of the velocity, acceleration and jerk kinematic limits can be instantaneously taken into account during the motion. Moreover, the very low calculation time (less than 1 microsecond) makes it possible to easily control a multi-DOF system during one control cycle (classically about 1 millisecond), while preserving time for other computer processing. The algorithm is experimentally tested on the new 7-DOF industrial robot KUKA LWR iiwa.

I. INTRODUCTION

Modern industrial robotic applications often require smooth trajectory generation. Robotic manipulation classically uses smooth profiles, such as jerk-limited trajectories, to avoid acceleration discontinuities, which can induce detrimental vibrations. A significant amount of work have been done to find efficient and elegant solutions to this problem of trajectory planning [1], [2], [3]. As the previous robotic field, service robotics (collaborative) and mobile robotics have to solve the same problem but with online capability. Indeed, in this recent context, robots attempt to instantaneously react to unpredicted events originating from the environment and potentially from the physical human-robot interaction. Therefore, according to the context, the trajectory generation has to verify all or part of the following points:

- Trajectories should be modified during execution time (online capability) depending on external measurements or human commands;
- Trajectories should be time-optimal for productivity reasons;
- The smoothness of the trajectories, at least jerk-limited, should depend on the expected behavior (e.g. human motion reproduction or end-point accuracy).

Existing solutions to the problem of time-optimal online trajectory generation can be found in the literature. In [4], authors propose an efficient real-time planner in the operational space, but only the acceleration-limited trajectory (trapezoidal velocity) is considered. In [5], [6], Kröger *et al.* introduce an elegant framework based on the current state of the system and

the final desired state, that provide time-optimal acceleration-limited trajectories. The jerk-limited solution is presented, but not detailed, in their recent works [7]. One notes that this algorithm belonging to the well-known Reflexxes Motion Library [8] is no longer freely available for robotics research. Algorithms based on the analytical recalculation of the time-optimal third order trajectory parameters can be found in [9], [10]. The computing time of such solutions is about 350 microseconds for one joint trajectory, which can be a significant limitation for multi-axis systems during one control cycle (typically 1 millisecond).

In this work, we propose a faster and more versatile methodology that exploits the property of FIR (Finite Impulse Response) filtering to generate on the fly jerk-limited trajectories, based on initial acceleration-limited trajectories. In addition to the real-time ability, the FIR-based method can be used to generate different jerk pattern (sine squared jerk, damped-jerk [11]) or smoother trajectories (snap-limited, crackle-limited,...) by a simple modification of the filter coefficients or combination of filters.

The next Section presents the methodology adopted for the generation of jerk-limited trajectories. Section III details the procedure to generate online time-optimal jerk-limited trajectories. Section IV shows how the kinematic limits can be changed online. In section V, a fixed-time solution for multi-axis motion synchronization is proposed. Experimental results obtained with a collaborative 7 axis robot are presented in Section VI. Conclusion are drawn in section VII.

II. TRAJECTORY GENERATION WITH JERK LIMITATION

The methodology used in this paper is based on the convolution of acceleration-limited trajectories with a FIR filter inducing the final jerk pattern. For instance, convolving a step function of value e with a simple sliding average FIR filter produces a ramp function with ending value e and a slope value of e/T (with T the FIR filter time). Therefore, as illustrated in Figure 1, an acceleration-limited profile may be turned into a jerk-limited profile by FIR filtering. In [12], Biagiotti *et al.* generalize this principle by combining FIR average filters in a cascade configuration to obtain smooth trajectories from simple rough step position commands. However, the resulting trajectories cannot be modified online (only zero initial velocity and acceleration conditions could be handled).

In this paper, the subscript f stands for "filtered", i.e.

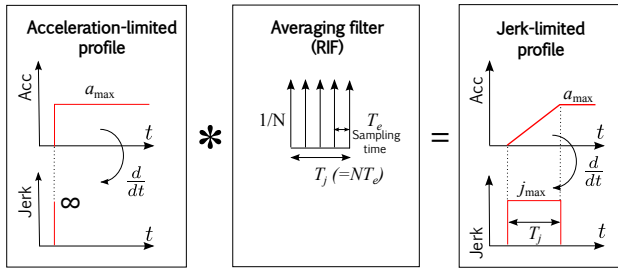


Fig. 1. FIR filtering principle for jerk-limited trajectory synthesis.

denotes the variables of the jerk-limited profile, which is the output of the algorithm. Moreover, for the online trajectory generation, it is assumed that the final velocity and acceleration are zeros, since if no new set-point is given, the goal is reached and the motion stops.

The inputs of the algorithm, presented in Figure 2 are the set-point q_e , the initial state of the system (q_{0f}, v_{0f}, a_{0f}) , and the kinematics limits $(v_{max}, a_{max}, d_{max}, j_{max})$. The algorithm is divided into two stages. The first one consists in generating an acceleration-limited (AL) trajectory, with is adapted for the following FIR filtering stage. The output of the filtering stage is a jerk-limited (JL) trajectory, which fulfills the requirements in terms of final position, initial state and kinematic limitations on the velocity, acceleration and jerk. As detailed in the remainder of this paper, the adaptation of the initial AL trajectory is based on simple analytical relations. One notes that for the time-optimal solution, this adaptation process can be repeated iteratively according to a decision tree.

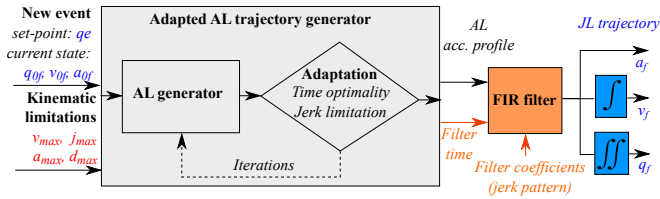


Fig. 2. Proposed online trajectory generation algorithm based on FIR filtering.

A. AL Generator

The first stage of our methodology consists in generating the time-optimal AL trajectory, which respects the following kinematic constraints and boundary conditions :

$$\begin{aligned} |v(t)| &\leq v_{max} \\ \left\{ \begin{array}{ll} \text{if } t < T_a + T_v & |a(t)| \leq a_{max} \\ \text{else} & |a(t)| \leq d_{max} \end{array} \right. & (1) \end{aligned}$$

The acceleration-limited profile, also known as bang-bang profile, can be build in three phases: first an acceleration phase during which the acceleration is constant: $|a(t)| = a_{max}$. During the second phase the acceleration is zero, i.e. the maximal velocity has been reached and is constant: $|v(t)| = v_{max}$. Finally in the third phase comes the deceleration, constant as well: $|a(t)| = d_{max}$. The durations of these three phases are respectively noted T_a , T_v and T_d , as illustrated in Figure 3. The displacement Δq is given by relation 2:

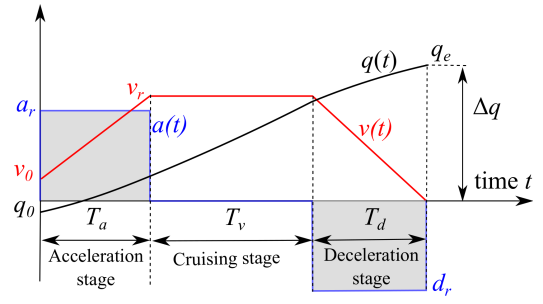


Fig. 3. A classic acceleration-limited profile. a_r , d_r and v_r are the reached acceleration, deceleration and velocity values.

$$\Delta q = q_e - q_0 = \frac{T_a(v_r + v_0)}{2} + T_v v_r + \frac{T_d v_r}{2}, \quad (2)$$

with:

$$T_a = \frac{v_r - v_0}{a_r}; \quad T_d = \frac{-v_r}{d_r}. \quad (3)$$

To ensure that the trajectory is time-optimal, we set the acceleration, deceleration, and cruising velocity values at the maximum values:

$$a_r = s \cdot a_{max}; \quad d_r = -s \cdot d_{max}; \quad v_r = s \cdot v_{max},$$

where a_r , d_r and v_r are the reached acceleration, deceleration and velocity values and s is the sign of the velocity during the cruising phase. In order to take into account a non-zero initial velocity, s can be determined by:

$$s = \text{sign}(q_e - q_{stop}), \quad (4)$$

where q_{stop} represents the position reached after the fastest full stop [9]. If the initial velocity v_0 is greater than the maximum velocity: $s \cdot v_0 > s \cdot v_r$, the velocity has to be reduced to v_r . In this case, $a_r = -s \cdot a_{max}$ and we have a double deceleration profile where a_r and d_r have the same sign [9]. Finally, we determine suitable values for T_a , T_d and T_v . Let us note Δq_{min} the minimum displacement for which v_r is reached, when $T_v = 0$. If $s \cdot \Delta q \geq s \cdot \Delta q_{min}$ then the maximal velocity is reached during the motion. Consequently we have:

$$T_a = \frac{v_r - v_0}{a_r}; \quad T_v = \frac{\Delta q - \Delta q_{min}}{v_r}; \quad T_d = \frac{-v_r}{d_r}. \quad (5)$$

Otherwise, the maximal velocity is not reached and the velocity profile is wedge-shaped. The maximum reached velocity v_r is then updated:

$$v_r = s \cdot \sqrt{\frac{-\frac{v_0^2}{ar} - 2\Delta q}{\frac{1}{dr} - \frac{1}{ar}}}, \quad (6)$$

and

$$T_a = \max(0, \frac{v_r - v_0}{a_r}); \quad T_v = 0; \quad T_d = \max(0, \frac{-v_r}{d_r}).$$

Finally, the triplet (T_a, T_v, T_d) defines the time-optimal acceleration-limited profile which respects the kinematic bounds.

B. Adaptation of the initial AL profile for FIR filtering

The parameters (q_0, v_0) used to generate the initial AL profile have to be adjusted according to the initial state (q_{0f}, v_{0f}, a_{0f}) . The relationships between the parameters of the non-filtered and filtered profiles are given by equations (7) and (8).

$$v_0 = v_{0f} + \frac{T_j a_{0f}}{2}, \quad (7)$$

$$q_0 = q_{0f} + T_j v_{0f} + \frac{T_j^2 a_{0f}}{3} + \frac{T_j}{2} (T_a a_r + T_d d_r), \quad (8)$$

with T_j the filter time, as shown in Figure 4.

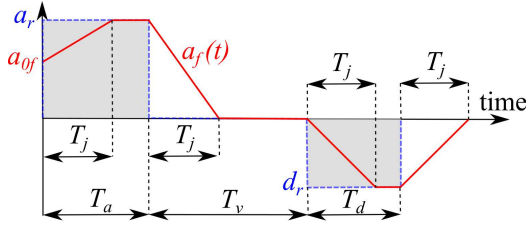


Fig. 4. The initial acceleration-limited trajectory (blue dashed line) is filtered to give a jerk-limited profile (solid red line).

Once the AL profile is generated, the test $T_v > T_j$ should be checked. If T_v is less than T_j , an adaptation of the AL profile is required to keep the jerk of the JL profile from exceeding its limits, as shown on Figure 5. Such an adaptation consists in re-generating the AL profile with a lower value of v_r which satisfies $T_v = T_j$ and (5).

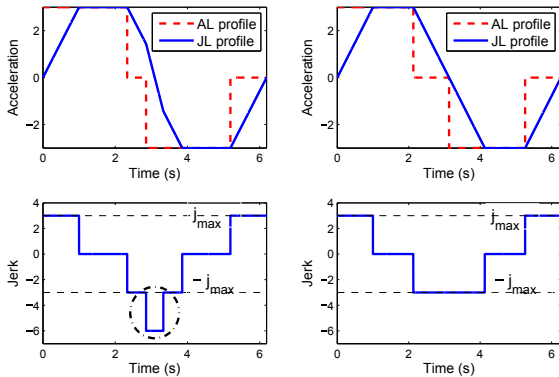


Fig. 5. Example of an adaptation: Type II (see Fig. 11). To the left: non-adapted profile, $T_v < T_j$. To the right, adapted profile, $T_v = T_j$ and $|v_r| < v_{max}$.

C. FIR Filtering and Jerk pattern

Starting from the previous AL profile, a JL trajectory can be obtained using a FIR moving average filter, as shown on Figure 4. The length of the filter T_j is classically chosen to reach the maximum value of the jerk:

$T_j = \max(|a_r - a_{0f}|, a_{max}, d_{max}) / j_{max}$. In order to respect the desired initial acceleration of the profile, the coefficients of the filter should be initialized at a_{0f} . One notes that the resulting JL trajectory is not time-optimal for all kinds of motion because the jerk time is fixed instead of the maximum

jerk value. Other jerk patterns can be generated by changing the coefficients of the filters. For instance, jerk patterns can be used to reduce residual vibrations in the system as described in [11]. Figure 6 presents two examples of trajectories generated using Damped-Jerk and sine squared Jerk patterns.

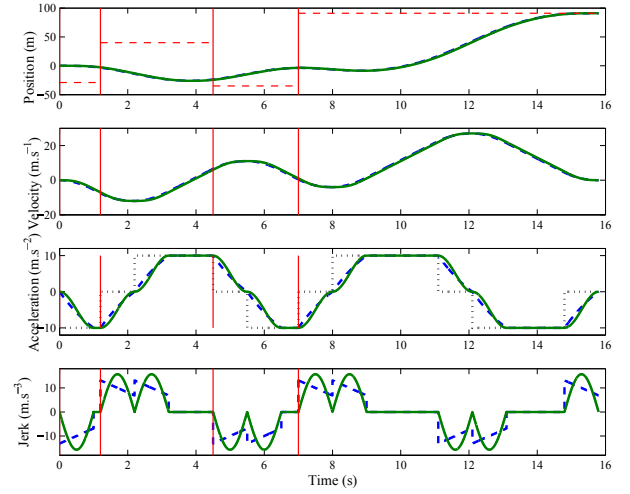


Fig. 6. Example of trajectories with different jerk patterns: Damped-Jerk (dashed blue line) and sine squared Jerk (solid green line).

III. TIME-OPTIMAL JL TRAJECTORY

Many industrial applications require time-optimal trajectories for productivity reasons. To ensure the time-optimality, the maximum jerk value should be fixed instead of the filter time. Therefore the filter time should be changed while filtering to satisfy the following relationships:

$$T_{ja} = \frac{|a_r - a_{0f}|}{j_{max}}; \quad T_{jv} = \frac{|a_r|}{j_{max}}; \quad T_{jd} = \frac{|d_r|}{j_{max}}, \quad (9)$$

where T_{ja} , T_{jv} and T_{jd} are the durations of the constant jerk stages, as shown in Figure 7.

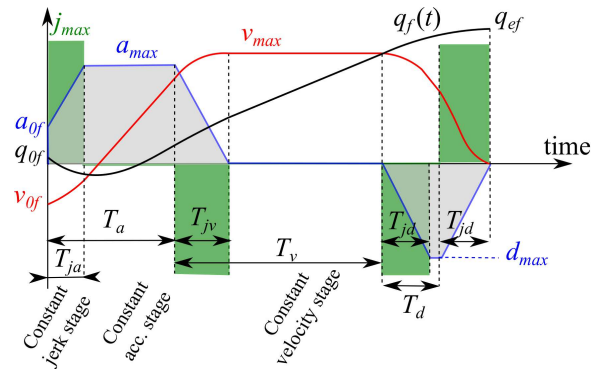


Fig. 7. Time-optimal jerk-limited profile with non-zero initial conditions, each derivative saturates at its bound.

Before generating the AL profile, the sign s has to be determined, i.e. the fastest full-stop position q_{stop} which can be reached under the kinematic bounds, as stated in (4). According to the state of the system, the acceleration profile leading to the fastest full-stop of the system can take different

shapes, as Figure 8 shows. If $q_{stop} = q_e$ then the full-stop profile should be kept as a result with no further calculation. If $sign(a_{0f}) \neq sign(a_r)$, the sign s should be reevaluated at

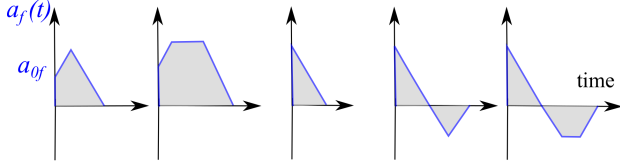


Fig. 8. Examples of fullstops acceleration profiles.

$t = |a_{0f}/j_{max}|$. In the case of a sign change, the trajectory is recalculated from the time T_p where s changes. Figure 9 shows such a trajectory.

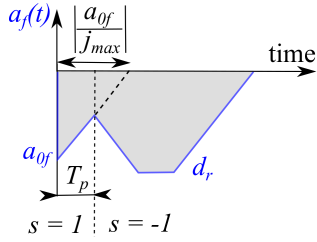


Fig. 9. Example of an acceleration profile with a sign change at $t = T_p$.

A. Adaptation of the AL profile

The relationships between (q_0, v_0) and (q_{0f}, v_{0f}, a_{0f}) given by (7) and (8) cannot be used for time-optimal generation since the length of the filter is not constant in the general case. Equations (10) and (11) are derived to take this difference into account.

$$v_0 = s \cdot \frac{a_r a_{0f}}{j_{max}} - s \cdot \frac{a_{0f}^2}{2j_{max}} + v_{0f}, \quad (10)$$

$$q_0 = a_r \left(\frac{s \cdot v_{0f}}{2j_{max}} + \frac{a_{0f}^2}{4j_{max}^2} \right) + q_{0f} - \frac{a_{0f}^3}{6j_{max}^2} - s(a_r + d_r) \frac{v_r}{2j_{max}}. \quad (11)$$

In addition to the adaptation previously described in II-B, which makes sure that the jerk bound is respected, the AL profile should also be checked to make sure that the solution given after filtering is time-optimal. If $T_{ja} > T_a$ or $T_{jd} > T_d$, then the acceleration profile after filtering saturates at a value less than its bound, respectively a_{max} or d_{max} , as illustrated on Figure 10. In that case, the initial AL profile should be adapted in order to produce the desired time-optimal jerk-limited trajectory. Figure 11 presents the decision table used to select the adaptation to be applied to the AL profile. As an example, if the initial AL profile exhibits $T_d < T_{jd}$ and $T_v > T_{jv}$, a Type III-b adaptation is needed. In that case, the deceleration stage does not saturate, i.e. $|d_r| < d_{max}$. Therefore

$$T_d = T_{jd} \Rightarrow \frac{-v_r}{d_r} = \frac{-s \cdot d_r}{j_{max}},$$

hence:

$$d_r = -s \sqrt{s j_{max} v_r}. \quad (12)$$

Once the AL profile is adapted, it should be re-checked for a possible second adaptation, as shown in Figure 12.

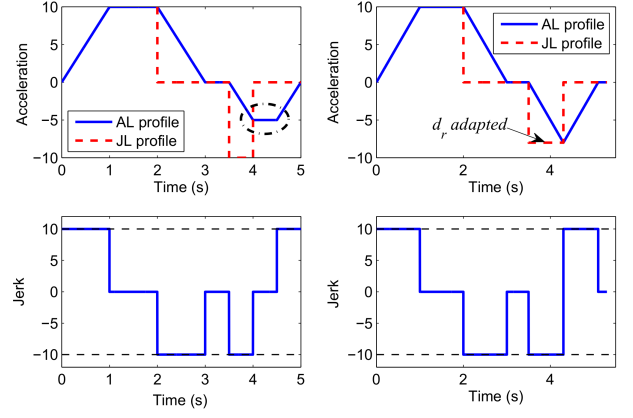


Fig. 10. Example of an adaptation: Type III-b (see Fig. 11). To the left: non-adapted profile, $T_d < T_{jd}$. To the right, adapted profile, $T_d = T_{jd}$ and $|d_r| < d_{max}$.

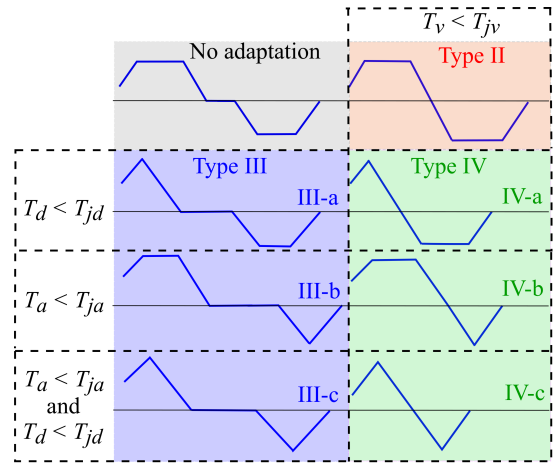


Fig. 11. Decision table for the selection of the adaptation of the AL profile.

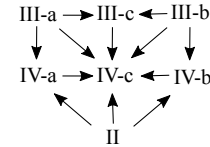


Fig. 12. Decision tree which shows the possible successive adaptations.

B. Filtering

In order to keep the jerk at its maximum value, the length of the filter is dynamically switched according to relation (9). The trajectory starts at $t = 0$ with a filter initialized at a_{0f} with a length T_{ja} . Then the filter is changed at $t = T_a$ to a length T_{jv} , initialized at a_r . Finally at $t = T_a + T_v$ the length of the filter is set to T_{jd} .

IV. ONLINE CONSTRAINTS CHANGE

Many recent robotic applications require online adaptation of the kinematic bounds of the trajectory. For instance, the detection of human presence in the workspace of the robot can enable a "smooth" mode in which the kinematic limits are lowered, or the jerk pattern modified. In the case where the velocity or acceleration limit is lowered during the motion, a

transition stage may be required. If the initial acceleration and velocity are such that the new velocity limits are or will be exceeded, i.e. if:

$$s \cdot \left(v_{0f} + \frac{a_{0f}}{2} \cdot \left| \frac{a_{0f}}{j_{max}} \right| \right) > s \cdot v_{max}, \quad (13)$$

then the deceleration stage, noted $a_{ds}(t)$, should bring the velocity back into its new bounds as soon as possible. Let us note the duration of this stage T_{ds} , as shown in Figure 13. We want:

$$v_{0f} + \int_{t_0}^{T_{ds}} a_{ds}(t) dt = s \cdot v_{max}.$$

For the fastest solution, we saturate the jerk:

$$v_{ds}(t) = -s \cdot \frac{j_{max} t^2}{2} + a_{0f} t + v_{0f}.$$

We want $v_{ds}(T_{ds}) = s \cdot v_{max}$, hence

$$-s \cdot \frac{j_{max}}{2} T_{ds}^2 + a_{0f} T_{ds} + v_{0f} - v_{max} = 0,$$

if $T_{ds} > \left| \frac{-s \cdot a_{max} - a_{0f}}{j_{max}} \right|$ then a_{ds} should saturate at $-s \cdot a_{max}$.

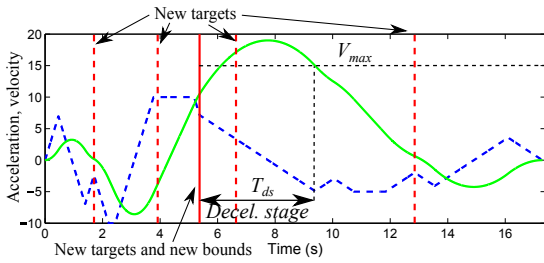


Fig. 13. Example of a trajectory with a deceleration stage. Dashed blue line: acceleration. Full green line: velocity. The kinematic bounds are lowered at $t = 5.3$ s (vertical full line). A deceleration stage ensures that the velocity respects its new limit.

V. MULTI-DIMENSIONAL CASE / AXIS SYNCHRONIZATION

In the general case, several Degrees-Of-Freedom (DOF) are controlled simultaneously to reach a targeted position. Classically, it is wished that the motion of each DOF ends at the same time. Consequently, for a motion involving n DOFs, with T_i the duration of the time-optimal trajectory for the i -th DOF, $n - 1$ trajectories have to be adjusted to the fixed time constraint T_{fix} given by the slowest DOF

$$T_{fix} = \max_i T_i. \quad (14)$$

This adjustment can be done in several, very different ways. Here, we propose to take benefits from the filtering strategy to easily compute the fixed-time solution for the JL profile. Indeed, without consideration of time-optimality, if an AL profile is directly convolved with a FIR filter of time duration T_j , the resulting profile is T_j seconds longer than the initial AL profile.

Noting τ_{fix} the fixed time duration of the AL profile, we have the relation $\tau_{fix} = T_{fix} - T_j$. The AL profile used here

allows arbitrary initial velocities and the fixed-time solution for such a profile can be simply calculated. Considering the rescaling of the whole time evolution with the parameter $\alpha = \tau_{fix}/\tau_i$:

$$\tau_a = \alpha T_a; \quad \tau_v = \alpha T_v; \quad \tau_d = \alpha T_d. \quad (15)$$

The parameters to be determined are a_r and d_r , respectively the reached acceleration and deceleration. From (2) and (3) it is straightforward to derive the relationships (16) and (17):

$$a_r = -\frac{q_0 + \tau_a \cdot v_0 + \tau_d \cdot v_0/2 + \tau_v \cdot v_0 - q_e}{\tau_a^2/2 + \tau_a \cdot \tau_d/2 + \tau_a \cdot \tau_v}, \quad (16)$$

$$d_r = -\frac{v_0 + \tau_a \cdot a_r}{\tau_d}, \quad (17)$$

where q_0 and v_0 are the ones used to generate the initial trajectory, i.e. the ones given by the relationships (7) and (8). Thus the axis synchronization problem for JL trajectories is solved with a set of simple equations. Since no iterative root-finding is required, these equations are almost instantaneous to evaluate. Equations (10) and (11) stand for trajectories with a jerk profile saturated at a given value j_{max} . Since synchronized trajectories do not respect that condition anymore, another method has to be developed. In order to minimize the computation time, it is wished to avoid iterative resolutions of equations. By setting the jerk at its maximum value for the first acceleration phase and preserving the length of the filter for the deceleration phase T_{jd} , as shown on Figure 14, the problem is simplified. T_{ja} and T_{jv} can be expressed as

$$T_{ja} = s \cdot \frac{a_r - a_{0f}}{j_{max}}, \quad T_{jv} = s \cdot \frac{a_r}{j_{max}}. \quad (18)$$

The unknown parameters are the reached accelerations and deceleration a_r and d_r , which are the solution of the system (19):

$$\begin{cases} A \cdot a_r^2 + B \cdot a_r + C = 0 \\ d_r = -\frac{v_{0f} + (a_{0f} + a_r) \cdot T_{ja}/2 + a_r \cdot (\tau_a - T_{ja}) + a_r \cdot T_{jv}/2}{\tau_d} \end{cases} \quad (19)$$

with

$$\begin{aligned} A &= -\frac{a_{0f} + j_{max} s \tau_a}{2j_{max}^2} \\ B &= \frac{\tau_a (T_{jd} + \tau_a + \tau_d + 2\tau_v)}{2} + \frac{a_{0f}^2}{2j_{max}^2} + \frac{a_{0f} s (T_{jd} + 2\tau_a + \tau_d + 2\tau_v)}{2j_{max}^2} \\ C &= q_0 f - q_e f + v_{0f} \left(\frac{T_{jd}}{2} + \tau_a + \frac{\tau_d}{2} + \tau_v \right) - \frac{a_{0f}^3}{6j_{max}^2} \\ &\quad - s \cdot \frac{a_{0f}^2 (T_{jd} + 2\tau_a + \tau_d + 2\tau_v)}{4j_{max}^2} \end{aligned}$$

This system gives an exact solution while $s \cdot a_{0f} \leq s \cdot a_r$. Otherwise, $T_{ja} = -s \cdot \frac{a_r - a_{0f}}{j_{max}}$, thus the expression of A , B and C changes. This approach is very fast, since it only requires to find the roots of a second order polynomial function.

Figure 15 shows an example of synchronized jerk-limited trajectories.

VI. RESULTS

The algorithm has been tested for all types of configurations to validate its results. In any case the algorithm generates the solution in less than 1 microsecond on a regular laptop i7 / 2.7GHz running with Windows 7 as operating system. At the best of the authors knowledge this is the fastest time-optimal jerk-limited trajectory generator. A typical control cycle time being 1 ms, our algorithm can easily handle any multi-axis

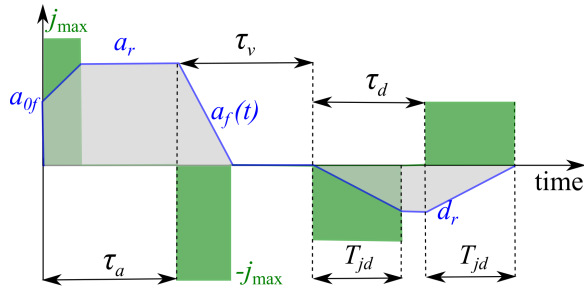


Fig. 14. The approach chosen to deal with joint synchronization is to constrain the value of the jerk for the first acceleration stage and the filter length for the second one.

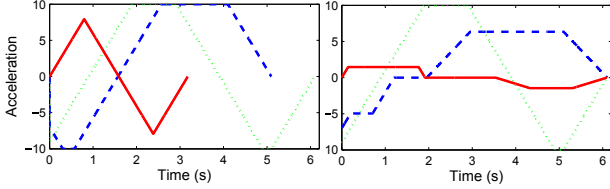


Fig. 15. Synchronization of time-optimal trajectories: initial trajectories to the left, synchronized trajectories to the right.

system and save a significant amount of time to the controller for other tasks, e.g. monitoring, obstacle avoidance or any data processing. Besides, such a small calculation time enable instantaneous reactions to unforeseen events, as introduced in [5].

The algorithm has been tested with a 7 DOF KUKA leightweighth LBR IIWA. The Fast Robot Interface (FRI) of the KUKA *Sunrise.Connectivity* collection of open interfaces was used for real time communication between the robot controller and an external computer, with a rate of 1 ms. Figures 16 and 17 show samples of experimental results with respectively synchronized and not synchronized joints. Dynamic new target references were randomly generated and the new trajectories updated online, with some changes in the kinematic limits.

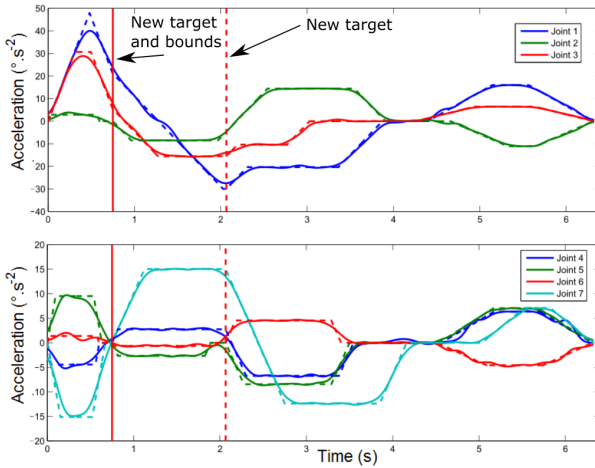


Fig. 16. Experimental results for synchronized trajectories (7 joints). New targets are given on vertical lines. j_{max} is lowered at the full vertical line. Dashed lines: reference acceleration, full lines: measured acceleration. Top graph: joints 1 to 3; bottom graph: joints 4 to 7.

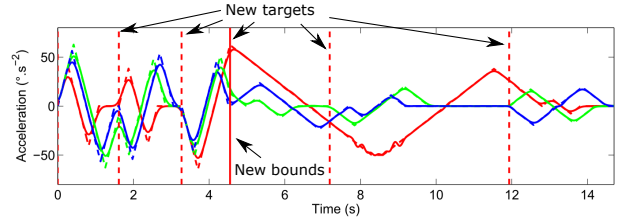


Fig. 17. Experimental results for non-synchronized trajectories (only three joints shown). New targets are given on vertical lines. j_{max} is lowered at the vertical full line. Dashed lines: reference acceleration, full lines: measured acceleration.

VII. CONCLUSION

A method to generate online jerk-limited trajectories based on FIR filtering of adapted acceleration-limited profiles is presented in this paper. This generator handles non-zero velocity and acceleration as well as non-symmetric acceleration bounds. Time-optimal and time-fixed solutions for multi-axis synchronization are also proposed. The proposed method generates time-optimal trajectories in less than one microsecond on a regular laptop.

REFERENCES

- [1] K. Erkorkmaz and Y. Altintas, Highspeed cnc system design. Part I: Jerk limited trajectory generation and quintic spline interpolation, *I.J. Machine Tools & Manufacture*, vol. 41, pp. 1323-1345, 2001.
- [2] S.Y. Jeong, Y.J. Choi, P. Park, and S.G. Choi, Jerk limited velocity profile generation for high speed industrial robot trajectories, in *Proc. of the IFAC World Congress, Praha, Czech Republic*, vol. 16(1), pp. 1362-1368, 2005.
- [3] A. Olabi, R. Béarée, O. Gibaru, and M. Damak, Feedrate planning for machining with industrial six-axis robots, *Control Engineering Practice*, vol.18(5), pp. 471-482, 2010.
- [4] R. Béarée, A. Olabi, Real-Time Planner in the Operational Space for the Automatic Handling of Kinematic Constraints, *IEEE Trans. Robot. Autom.*, 11(3) : 730-739, 2014.
- [5] T. Kröger, A. Tomiczek, and F. M. Wahl, Towards on-line trajectory computation, in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China, 2006*, pp. 736-741.
- [6] T. Kröger and F. M. Wahl, On-line trajectory generation: Basic concepts for instantaneous reactions to unforeseen events, *IEEE Trans. on Robotics*, vol. 26, no. 1, pp. 94-111, 2010.
- [7] T. Kröger, Opening the Door to New Sensor-Based Robot Applications - The Reflexxes Motion Libraries, In *Proc. of the IEEE International Conference on Robotics and Automation, Shanghai, China, May 2011*.
- [8] Reflexxes Motion Library. <http://www.reflexxes.ws>.
- [9] R. Haschke, E. Weitnauer and H. Ritter, On-line planning of time-optimal, jerk-limited trajectories, in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Nice, France, 22-26 Sept. 2008*, pp.3248-3253.
- [10] F. Ramos, M. Gajamohan, N. Huebel and R. D'Andrea, Time-optimal online trajectory generator for robotic manipulators, report of Institute for Dynamic Systems and Control, 2013.
- [11] R. Béarée, New Damped-Jerk trajectory for vibration reduction, *Control Engineering Practice*, vol.28(14), pp. 112-120, 2014.
- [12] L. Biagiotti and C. Melchiorri, FIR filters for online trajectory planning with time and frequency-domain specifications, *Control Engineering Practice*, vol.20(12), pp. 1385-1399, 2012.
- [13] R. Béarée, A. Olabi, Dissociated jerk-limited trajectory applied to time-varying vibration reduction, *Robotics and Computer-Integrated Manufacturing*, 29(3) : 444-453, 2013.