An Experimental Setup to Test Time-Jerk Optimal Trajectories for Robotic Manipulators

Publisher:
Springer Science and Business Media B.V.

Published
DOI:10.1007/978-3-031-32606-6_36

An experimental setup to test time-jerk optimal trajectories for robotic manipulators

Federico Lozer[1], Lorenzo Scalera[1],
Paolo Boscariol[2], and Alessandro Gasparetto[1]

[1] University of Udine, 33100 Udine, Italy
{federico.lozer,lorenzo.scalera,alessandro.gasparetto}@uniud.it
[2] University of Padova, 36100 Vicenza, Italy
paolo.boscariol@unipd.it

Abstract. An experimental setup to test optimal time-jerk trajectories for robotic manipulators is presented in this paper. The setup is used, in this work, to test the execution of smooth motion profiles passing through a sequence of via-points, designed by means of the optimization of a mixed time-jerk cost function. Experimental tests are performed on a Franka Emika robot with seven degrees of freedom equipped with accelerometers to measure the motion-induced oscillations of the end-effector. The experimental results show a good agreement with the numerical tests and demonstrate the feasibility of the approach chosen for optimizing smooth trajectories for robotic manipulators.

Keywords: robotics · trajectory planning · vibrations · splines · jerk · Franka Emika.

1 Introduction

Trajectory planning is a fundamental problem in robotics. Its purpose is to correlate a geometric path with a time sequence of position, velocity and acceleration to be executed by a robotic manipulator. Several approaches for trajectory planning have been investigated in the literature with the aim of enhancing productivity or other performance figures. One possible choice, which has been largely investigated in recent years, is to optimize the motion law with respect to the energy consumption of a robotic system [1].

Another interesting field of application of trajectory planning is the reduction of motion-induced oscillations. Indeed, robotic applications usually require smooth and jerk-limited trajectories during the prescribed task [2], to extend actuators lifetime, prevent mechanical damages and improve motion accuracy.

Splines are often chosen for representing an effective basis for the planning of robot trajectories with high order of continuity, sporting at the same time flexibility and a concise mathematical representation. In [3], Cook and Ho elaborated a trajectory planning method based on the “434” polynomial spline approach achieving acceleration continuity. That method was extended by Boscariol et al. [4], who presented the smoother "545" and the "5455" polynomial methods, which provide continuity up to jerk.

Smoothness, or lack thereof, is commonly regarded as the main cause of motion-induced oscillations, so its minimization is often the main goal of the motion planning. For example in [5], Fang et al. proposed a method based on S-curve point-to-point trajectories to solve a minimum-jerk optimization problem. In [6], Dai et al. experimented a genetic algorithm for minimum-jerk trajectories calculated in the configuration space of a UR3 robot. In [7], Wu et al. solved a constrained optimization problem for the minimum-jerk trajectories calculated with NURBS curves. In that case, jerk is reduced by using a cost function that is equal to the sum of the integrals of squared jerk of each robot joint.

The minimum-time calculation is also a focal point of trajectory planning, since a fast trajectory can increase the amount of tasks performed over a fixed amount of time. In this context, Abu-Dakka et al. proposed a method based on the PPGA1/PPGA2 genetic algorithm to calculate minimum-time trajectories using cubic splines [8]. Furthermore, in [9], Palleschi et al. used minimax method to calculate minimum-time 6^{th}-order B-spline trajectories.

Most other works, like [10] by Gasparetto et al., investigated the optimization of a cost function that considers contributions which are proportional to both time and squared jerk, due to the need to have at the same time a fast and a smooth trajectory, proposing a mixed cost function that sets a trade-off between those two terms. Furthermore, Zanotto et al. in [11] have shown, through extensive experiments, the effectiveness of that approach by means of a comparison with the outcomes of a standard spline algorithm with imposed timing. The same optimum problem was tested on a SCARA robot by Huang et al. [12], solving it with a genetic algorithm (`NSGA-II`) and comparing their results to those of Gasparetto. However, to the best of our knowledge, that method has not been yet tested on a manipulator with 6 or 7 degrees of freedom (DOFs).

In this work, we present an experimental setup to test time-jerk optimal trajectories for robotic manipulators. We implement an optimization method for the planning of smooth trajectories based on a cost function composed of a term proportional to the total execution time, and one term proportional to the squared jerk along the trajectory. The setup comprises a Franka Emika robot with 7 DOFs equipped with three accelerometers to measure the motion-induced oscillations of the end-effector. The experimental results show a good agreement with the numerical tests and demonstrate the feasibility of the approach in optimizing smooth trajectories for robotic manipulators.

The paper is organized as follows: in Sect. 2 the trajectory planning approach is presented. Section 3 describes experimental setup, whereas Sect. 4 reports the experimental results. Finally, the conclusions are given in Sect. 5.

2 Time-jerk optimal trajectories

The planning of time-jerk optimal trajectories aims at defining motion laws as smooth as possible and, at the same time, as fast as possible. Limiting jerk values produces smooth movements but leads to a slow trajectory. On the other hand, decreasing the duration of the trajectory produces higher levels of jerk. The key point is to find a suitable trade-off between a smooth and a fast trajectory.

The proposed algorithm is based on a parametrization of the motion profile by means of a set of time intervals h_i: each one of them representing the duration of the i-th segment of the trajectory, which connects the via-point i to the via-point $i+1$. The set of h_i with $1 \leq i \leq w_p$ (number of via points), i.e., the vector $\mathbf{h}$, is the decision variable vector, whose values are set by means of the minimization of the cost function in Eq. (1). The latter weighs both the sum of h_i, which equals the total duration T, and the sum of the N values of the integral of joint jerks, being N the number of robot joints. The trade off between the two elements of the cost function is set by α, which is assumed to vary between 0 and 1.

$$\min_{\mathbf{h}} \alpha \sum_{i=1}^{w_p-1} h_i + (1-\alpha) \sum_{j=1}^{N} \int_0^T (\dddot{q}_j(t))^2 dt \tag{1}$$

The feasibility of the resulting trajectory $(\boldsymbol{q}(t), \ddot{\boldsymbol{q}}(t), \dddot{\boldsymbol{q}}(t))$ is ensured by the following constraints:

$$\begin{cases} \boldsymbol{q}_{min} < \boldsymbol{q} < \boldsymbol{q}_{max} & \boldsymbol{q}_{min}, \boldsymbol{q}_{max} \text{ lower and upper position bounds} \\ |\dot{\boldsymbol{q}}| < \dot{\boldsymbol{q}}_{max} & \dot{\boldsymbol{q}}_{max} \text{ velocity bound} \\ |\ddot{\boldsymbol{q}}| < \ddot{\boldsymbol{q}}_{max} & \ddot{\boldsymbol{q}}_{max} \text{ acceleration bound} \\ |\dddot{\boldsymbol{q}}| < \dddot{\boldsymbol{q}}_{max} & \dddot{\boldsymbol{q}}_{max} \text{ jerk bound} \\ |\boldsymbol{\tau}| < \boldsymbol{\tau}_{max} & \boldsymbol{\tau}_{max} \text{ torque bound} \\ |\dot{\boldsymbol{\tau}}| < \dot{\boldsymbol{\tau}}_{max} & \dot{\boldsymbol{\tau}}_{max} \text{ torque derivative bound} \end{cases} \tag{2}$$

The constrains in Eq. (2) include kinematic quantities, as well as joint torques, the latter being computed on the basis of the inverse dynamic model of the manipulator. In particular, the torques $\boldsymbol{\tau}$ are computed as:

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{F}_v\dot{\boldsymbol{q}} + \boldsymbol{f}_c \text{sign}(\dot{\boldsymbol{q}}) + \boldsymbol{g}(\boldsymbol{q}) \tag{3}$$

where $\boldsymbol{M}(\boldsymbol{q})$ is the mass matrix of the robot, and $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}$ considers the Coriolis and centrifugal terms. $\boldsymbol{F}_v$ and $\boldsymbol{f}_c$ are the viscous and Coulomb friction terms, respectively, whereas $\boldsymbol{g}(\boldsymbol{q})$ accounts for gravity. The optimization algorithm has been tested with a "434" spline [3,4], in which a 3^{rd} order polynomial function is used to describe all the segments between the via-points, with the exception of the first and the last one, where a 4^{th} polynomial is adopted, but the method can be extended to more elaborated spline functions as well. The trajectory between two adjacent via-points P_k and P_{k+1} with $2 \leq k \leq N-2$ can be expressed with the following function of time:

$$F_k(t) = B_{k,1} + B_{k,2}t + B_{k,3}t^2 + B_{k,4}t^3 \tag{4}$$

where $B_{k,n}$ are the polynomial coefficients of the polynomial equation. Instead, the trajectory between the first two via-points P_1 and P_2 and the last two way points P_{N-1} and P_N (i.e., $k = 1, N-1$) is given by:

$$F_k(t) = B_{k,1} + B_{k,2}t + B_{k,3}t^2 + B_{k,4}t^3 + B_{k,5}t^4 \tag{5}$$

The polynomial coefficients $B_{k,i}$ are computed to ensure the continuity of positions, speeds and accelerations. Jerk continuity is not however achieved, as jerk is either linearly changing or constant during each segment [3].

3 Experimental setup

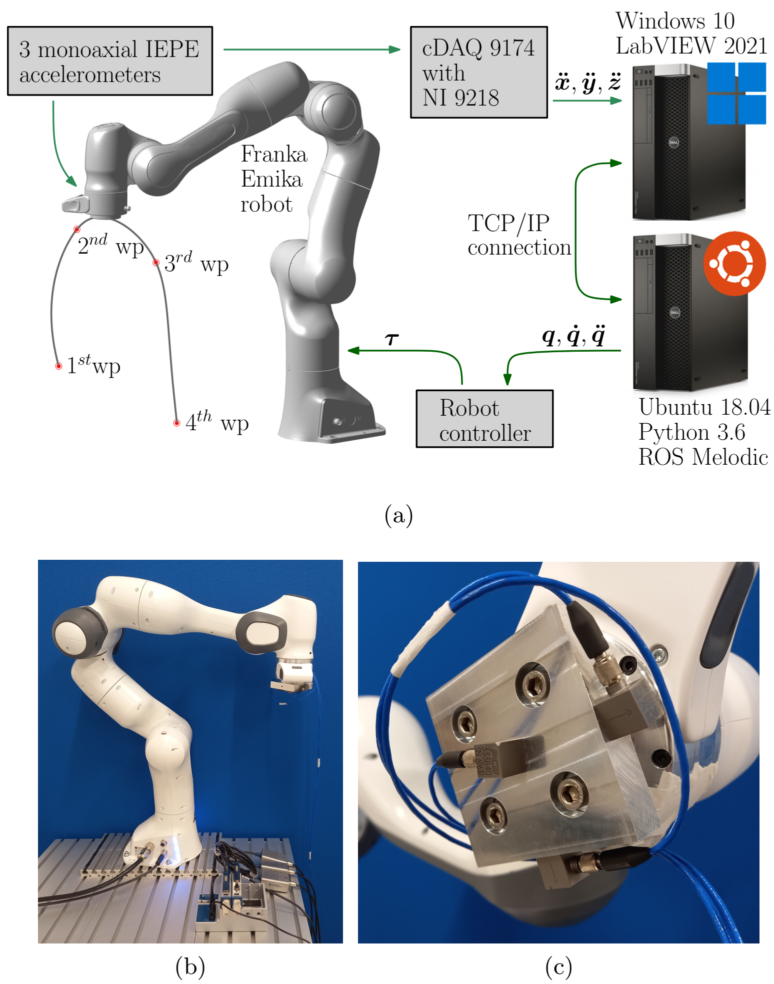

Fig. 1: (a) Overview of the experimental setup; (b) Franka Emika robot and data acquisition system; (c) aluminium flange and accelerometers on the end-effector.

$\boldsymbol{\alpha}$	$\boldsymbol{T}$ $[s]$	$\boldsymbol{J}$ $[rad^2/s^5]$	$\ddot{\boldsymbol{x}}_{\boldsymbol{rms}}$ $[m/s^2]$	$\ddot{\boldsymbol{y}}_{\boldsymbol{rms}}$ $[m/s^2]$	$\ddot{\boldsymbol{z}}_{\boldsymbol{rms}}$ $[m/s^2]$	$\ddot{\boldsymbol{X}}_{\boldsymbol{rms}}$ $[m/s^2]$
0.20	12.07	0.60	0.23	0.28	0.18	0.40
0.40	10.25	1.37	0.27	0.32	0.20	0.46
0.60	8.95	2.69	0.31	0.36	0.21	0.52
0.80	7.60	6.09	0.34	0.43	0.25	0.56
1.00	2.17	$3.51 \cdot 10^3$	1.14	1.76	1.19	2.40

Table 1: Results of Test 1 for some values of α.

The optimization problem in Eq. (1) with (2) is solved by a sequential quadratic programming algorithm using the `fmincon` Matlab function. To evaluate the method, 100 solutions have been run by varying α between 0 and 1 in equal steps. Each resulting trajectory has then been executed three times each on a Franka Emika robot with 7 DOFs (Fig. 1). The end-effector of the manipulator is equipped with a custom aluminium flange on which three monoaxial piezoelectric accelerometers are rigidly attached to. Data from accelerometers is acquired with a sampling frequency of 10 kHz by means of two a National Instruments NI9218 C Series modules, and a cDAQ 9174 chassis. Before post processing, raw acceleration data are filtered in Matlab with a moving-average filter that averages data over 5 samples.

The tests are carried out using two computers, one with Windows 10 operating system and one with Ubuntu 18.04: the two can exchange data using the TCP/IP protocol (Fig. 1). The workstation running Windows 10 is used for generating trajectories, for the data acquisition in LabVIEW 2021 and for the data analysis and post-processing in Matlab. The workstation running Ubuntu is used to control the robot by means of ROS Melodic Morenia and Python 3.6. The trajectories are executed on the Franka arm with a Python program that builds a message for the robot controller, specifying the sequence of joint position, velocity and acceleration which are sampled at 100 Hz.

4 Experimental results

Figure 2 shows an example of trajectory, which has been evaluated for $\alpha = 0.5$. The figure reports the joint positions, velocities, and absolute acceleration over time. The overall acceleration is computed by the vector sum of the acceleration signals measured by each mono-axial accelerometer. Vertical solid lines in Fig. 2 indicate the passing on via-points.

Examples of measured data obtained by varying the weight α are reported in Tab. 1. In particular, total time T, sum of integral of squared joint jerks J, root mean square Cartesian accelerations ($\ddot{\boldsymbol{x}}_{\boldsymbol{rms}}$,$\ddot{\boldsymbol{y}}_{\boldsymbol{rms}}$,$\ddot{\boldsymbol{z}}_{\boldsymbol{rms}}$) and absolute root mean square acceleration $\ddot{\boldsymbol{X}}_{\boldsymbol{rms}}$ are reported for some values of α. The data in Tab. 1 clearly shows that the relative weight α allows the user to set the desired

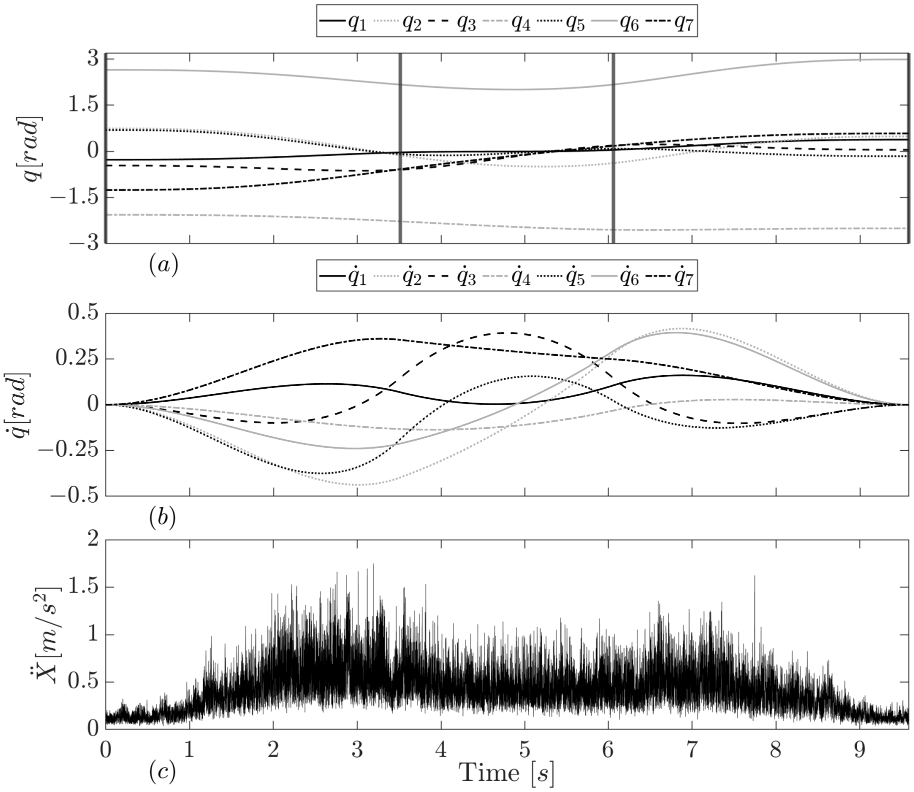

Fig. 2: Example trajectory: joint position, velocity and absolute acceleration signal for $\alpha = 0.5$. Vertical solid lines indicate target via-points.

trade-off between the total execution time and the acceleration measured on the end-effector of the robot.

Figure 3(a) shows the trend of T and J as a function of α. Since α is increasing, T is decreasing, presenting an inflection point approximately for $\alpha = 0.5$. On the contrary, J is increasing over α with a parabolic trend. In fact, starting from low values of α and sweeping to higher values, the algorithm moves from jerk minimization to minimum time trajectory optimization, as in Eq. 1. Fig. 3(b) shows J as a function of T. As previously mentioned, these two quantities are inversely proportional to each other.

Finally, Fig. 3(c) shows $\ddot{X}_{rms}$ as a function of the total execution time T for the three repeated tests. The resulting RMS accelerations are not only inversely proportional to the total execution time, but they are also closely related to the chosen metric, i.e. the integral of squared jerk J. Furthermore, the results of the three repeated tests reported in Fig. 3(c) show very close results throughout the tree repetitions of the same test, testifying a high repeatability of the experimental tests and of the acceleration measurements.

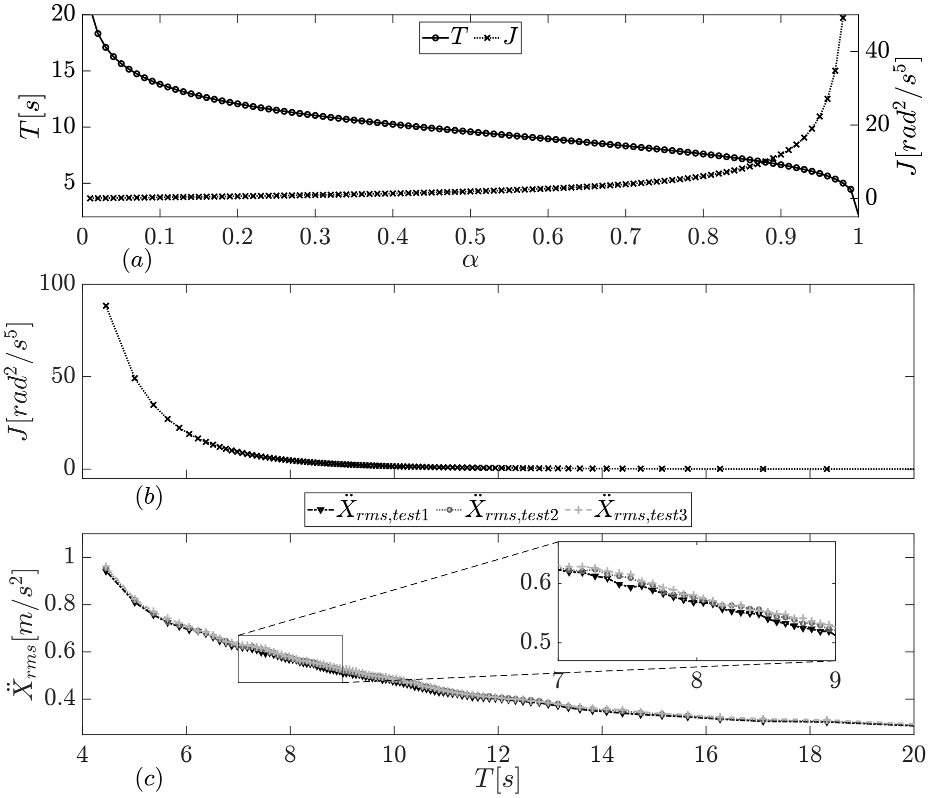

Fig. 3: (a) T and J vs. α; (b) J vs. T; (c) $\ddot{X}_{rms}$ vs. T.

5 Conclusions

In this paper an experimental setup to test time-jerk optimal trajectories for robotic manipulators has been presented. The experimental setup comprises a Franka Emika robot with 7 DOFs equipped with three accelerometers which measure the oscillations of the end-effector. The setup has been used for testing the smooth trajectories which were defined upon a "434" planning algorithm, by means of an optimization problem which minimizes a mixed cost function which weighs the total execution time as well as the integral of squared joint jerk. The experimental results shown a good agreement with the numerical results, demonstrating the feasibility of the proposed approach in optimizing smooth trajectories for robotic manipulators as well as proving the accuracy and the repeatability of the proposed experimental setup. Future works will include the exploitation of redundancy in the planning of time-jerk optimal trajectories, and the real-time computation of minimum-jerk trajectories for collaborative robotics, according to the approach presented in [13].

Acknowledgments

This research was supported by the Laboratory for Big Data, IoT, Cyber Security (LABIC) funded by Friuli Venezia Giulia, and the Laboratory for Artificial Intelligence for Human-Robot Collaboration (AI4HRC) funded by Fondazione Friuli. We thank LAMA FVG for the fabrication of the aluminium flange.

References

1. Giovanni Carabin and Lorenzo Scalera. On the trajectory planning for energy efficiency in industrial robotic systems. *Robotics*, 9(4):89, 2020.
2. Giulio Trigatti, Paolo Boscariol, Lorenzo Scalera, Daniele Pillan, and Alessandro Gasparetto. A look-ahead trajectory planning algorithm for spray painting robots with non-spherical wrists. In *IFToMM Symp. on Mech. Design for Robotics*, pages 235–242. Springer, 2018.
3. CC Cook and CY Ho. The application of spline functions to trajectory generation for computer-controlled manipulators. In *Comp. Tech. for Rob.*, pages 101–110. Springer, 1984.
4. Paolo Boscariol, Alessandro Gasparetto, and Renato Vidoni. Planning continuous-jerk trajectories for industrial manipulators. In *Eng. Systems Design and Analysis*, volume 44861, pages 127–136, 2012.
5. Yi Fang, Jie Hu, Wenhai Liu, Quanquan Shao, Jin Qi, and Yinghong Peng. Smooth and time-optimal s-curve trajectory planning for automated robots and machines. *Mechanism and Machine Theory*, 137:127–153, 2019.
6. Chengkai Dai, Sylvain Lefebvre, Kai-Ming Yu, Jo MP Geraedts, and Charlie CL Wang. Planning jerk-optimized trajectory with discrete time constraints for redundant robots. *IEEE Trans. on Autom. Sc. and Eng.*, 17(4):1711–1724, 2020.
7. Guanglei Wu and Sida Zhang. Real-time jerk-minimization trajectory planning of robotic arm based on polynomial curve optimization. *Proc. of the Inst. of Mech. Eng., Part C: J. of Mech. Eng. Sc.*, 2022.
8. Fares J Abu-Dakka, Iyad F Assad, Rasha M Alkhdour, and Mohamed Abderahim. Statistical evaluation of an evolutionary algorithm for minimum time trajectory planning problem for industrial robots. *The Int. J. of Adv. Manuf. Tech.*, 89(1):389–406, 2017.
9. Alessandro Palleschi, Manolo Garabini, Danilo Caporale, and Lucia Pallottino. Time-optimal path tracking for jerk controlled robots. *IEEE Rob. and Autom. Lett.*, 4(4):3932–3939, 2019.
10. Alessandro Gasparetto and Vanni Zanotto. A technique for time-jerk optimal planning of robot trajectories. *Rob. and Comp.-Int. Manuf.*, 24(3):415–426, 2008.
11. Vanni Zanotto, Alessandro Gasparetto, Albano Lanzutti, Paolo Boscariol, and Renato Vidoni. Experimental validation of minimum time-jerk algorithms for industrial robots. *Journal of Intelligent & Robotic Systems*, 64(2):197–219, 2011.
12. Junsen Huang, Pengfei Hu, Kaiyuan Wu, and Min Zeng. Optimal time-jerk trajectory planning for industrial robots. *Mech. and Mach. Theory*, 121:530–544, 2018.
13. Lorenzo Scalera, Andrea Giusti, Renato Vidoni, and Alessandro Gasparetto. Enhancing fluency and productivity in human-robot collaboration through online scaling of dynamic safety zones. *The Int. J. of Adv. Manuf. Tech.*, 121(9):6783–6798, 2022.