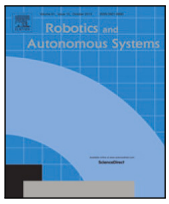
Robotics and Autonomous Systems 192 (2025) 105049

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot



# Planning optimal minimum-jerk trajectories for redundant robots

Federico Lozer [a], Lorenzo Scalera [a,*], Paolo Boscariol [b], Alessandro Gasparetto [a]

[a] Polytechnic Department of Engineering and Architecture, University of Udine, Udine, 33100, Italy
[b] Department of Management and Engineering, University of Padova, Vicenza, 36100, Italy

ARTICLE INFO

*Keywords:*
Robotics
Trajectory planning
Optimization
Mechanical vibration
Redundancy

ABSTRACT

In this paper, a minimum-jerk trajectory planning approach for redundant manipulators is presented. The proposed approach leverages not only the optimization of the time intervals between each of the way points of the assigned path, but also the optimal positions of a selected robot joint to reduce the jerk of the robot end-effector. This multi-stage optimization strategy is validated through extensive numerical simulations and experimental tests on a robot with seven degrees of freedom performing a pick-and-place motion. The results of the tests, supported by accelerometer measurements of the vibrations of the robot end-effector, prove the performance of the proposed approach in reducing both the acceleration and the jerk levels of the redundant manipulator in comparison with a state-of-the-art trajectory planning technique.

## 1. Introduction

Nowadays, trajectory planning is a fundamental problem in robotics, where it plays an essential role in several industrial processes. The purpose of trajectory planning is to establish a correlation between a geometric path and a timed sequence of positions, velocities, and accelerations that a robotic manipulator has to execute [1]. A carefully planned trajectory can lead to several advantages, one of which is the reduction of the operational time [2]. By wisely designing the path and movements of robotic systems, manufacturers can minimize the duration required to complete specific tasks, leading to higher throughput, lower costs, and enhanced productivity [3].

Beyond time minimization, trajectory planning can also contribute to improve energy efficiency [4,5]. In manufacturing industries, energy consumption is a critical factor that directly impacts operational costs and sustainability [6]. Optimizing the trajectories of machines and robots can reduce unnecessary movements and energy expenditure, leading to lower power consumption and a more sustainable production [7].

Another key benefit that can be achieved through optimal trajectory planning is the reduction of vibrations and jerk during operations [8]. This aspect is increasingly important as manufacturing systems evolve also including collaborative robotics, where robots work alongside human operators [9–11]. In such settings, robots are often lightweight and flexible to ensure safe and effective interaction with humans, and these characteristics make them more susceptible to vibrations and mechanical shocks. Furthermore, emerging human–robot collaborative scenarios demand smooth and minimum-jerk trajectories for the robots not only for safety [12,13], but also to account for the cognitive workload and stress of the human operator [14]. By minimizing the unwanted dynamic effects of mechanical vibrations and jerk, trajectory planning can enhance the reliability and lifespan of the robot actuators and mechanical components, while also ensuring a safer and less stressful interaction between humans and robots.

Several approaches have been proposed in the literature to achieve smooth and minimum-jerk trajectories for robotic and mechatronic systems [15,16]. In [17], Piazzi and Visioli presented a method for the planning of minimum-jerk trajectories based on a global constrained minimax optimization. In the problem they formulated, the interpolation is based on a sequence of cubic polynomial functions. In [18,19], Gasparetto and Zanotto proposed a possible solution to the problem of jerk continuity by using five-degree B-spline curves together with a cost function composed of both time and jerk contributions. Another popular trajectory planning technique is based on a simpler interpolation function, the 434 approach developed by Ho and Cook in [20]. Moreover, the authors in [21] illustrated a technique capable of obtaining continuous-jerk trajectories with high smoothness and minimum excitation of vibrations based on the so-called 545 and 5455 interpolations.

More recently, the minimum time-jerk trajectory planning for robotic manipulators is solved in [22] using a cost function proportional to the total execution time and to the integral of the squared jerk, and applying the augmented Lagrange constrained particle swarm optimization technique. Furthermore, a time-optimal and smooth S-curve

* Corresponding author.
*E-mail address:* lorenzo.scalera@uniud.it (L. Scalera).

https://doi.org/10.1016/j.robot.2025.105049
Received 2 October 2024; Received in revised form 18 March 2025; Accepted 1 May 2025
Available online 19 May 2025

trajectory planning approach is presented in [23] for achieving high-speed and precise operations for robotic manipulators. The method uses the piecewise sigmoid function to create a jerk profile with appropriately specified phase durations, resulting in infinitely continuously differentiable trajectories under the provided velocity, acceleration, and jerk limits. An improved sinusoidal jerk model is proposed in [24] to generate online smooth joint trajectories with short execution time and continuity up to the jerk level.

Additional examples of minimum-time trajectory planning include the work [25], where a piecewise quintic polynomial interpolation is adopted, and the work [26], where the robot joint-space trajectory is interpolated using a fifth-order B-spline curve, and then optimized with the elitist non-dominated sorting genetic algorithm (NSGA-II) for two objectives: traveling time and root mean square of jerk. Furthermore, the authors in [27] proposed a spline-based strategy for generating smooth and time-optimal trajectories, while meeting kinematic constraints. In [28], a high-order smooth trajectory planning approach for robot surface machining is proposed, considering both tool pose constraints and robot singularities. Fifth-order B-spline curves together with a hybrid whale optimization and genetic algorithm are applied in [29] for the time-optimal and jerk-continuous trajectory planning of industrial manipulators.

All the abovementioned work is developed for traditional industrial robots without redundancy, i.e., robots with a number of degrees of freedom equal to the number of variables needed to perform the desired task [30]. However, redundancy can be exploited to further improve the performance of trajectory planning and achieve better overall efficiency of robotic operations [31,32]. For instance, a motion planning strategy for vibration reduction of a redundant robot is described in [33]. The time-optimal path following along a fixed end-effector path is addressed in [34] for kinematically redundant robots. Time-jerk optimal trajectory planning of a 7-degree-of-freedom (DOF) redundant robot is addressed in [35], combining the constrained particle swarm optimization technique with the augmented Lagrange multiplier approach. The problem of reducing jerk in redundant robots is also investigated in [36], where the authors presented a multilevel simultaneous minimization strategy to remedy the joint-angle drift and non-zero final joint-velocity phenomena, as well as to prevent the occurrence of excessive joint variables of redundant manipulators. Finally, a method for effectively planning the motion trajectory of robots in manufacturing tasks is presented in [37], and validated on a challenging robot-assisted 3D printing task.

In this paper, we present an approach for the minimum-jerk trajectory planning for redundant robots. The proposed approach is based on a multi-stage optimization strategy: a time-jerk optimization is first performed to reduce both the total time and the jerk of the robot end-effector, as in [18]. Then, the position of one selected joint of the redundant manipulator is optimized for each of the considered way points of the robot path. Finally, a second time-jerk optimization is run to further improve the results of the two previous stages. The effectiveness of proposed approach is validated with both extensive numerical simulations and experimental tests on a Franka Emika Panda arm with seven degrees of freedom performing a pick-and-place trajectory. The results show the feasibility of the proposed approach and the benefits in reducing the jerk levels and the measured mechanical vibrations of the robot end-effector accordingly. Furthermore, the effectiveness of the approach in improving the overall smoothness of robot trajectories is also proved with respect to a notable trajectory planning technique [18]. To summarize, the main contributions of this work include:

- an approach for minimum-jerk trajectory planning of redundant manipulators based on a multi-stage optimization of time intervals and positions of a selected joint;
- the results of extensive numerical and experimental tests on a redundant manipulator, also including accelerometer measurements of the end-effector vibrations;
- a comparison of the proposed approach with the minimum-jerk trajectory planning technique in [18].

The paper is organized as follows: Section 2 describes the approach proposed in this paper. Section 3 illustrates the experimental setup, whereas the results are analyzed in Section 4. More in detail, Section 4.1 reports the results of the extensive numerical simulations, whereas the outcomes of the experiments are illustrated in Section 4.2. Finally, the conclusions of this work and possible future developments are summarized in Section 5.

## 2. Optimal minimum-jerk trajectory planning

The approach proposed in this paper for the minimum-jerk trajectory planning for redundant robots is based on the parameterization of the trajectory by means of time intervals between consecutive way points to be visited by the manipulator in a fixed sequence. The main goal of this work is to reduce the mechanical vibrations induced by the motion of the robot. This objective is pursued through the reduction of the jerk of the end-effector, a common approach in the literature [15,18,19,38]. The proposed motion planning procedure leverages the optimization of both the vector of time intervals $\boldsymbol{h}$ between consecutive way points, and the vector of positions $\boldsymbol{q}^*$ of a selected redundant joint of the robot along the way points. The proposed approach is structured in the following three stages of optimization:

- Stage 1: the first optimization phase is formulated to reduce both the total time, and the jerk of the end-effector, by varying the vector of time intervals $\boldsymbol{h}$, as in [18];
- Stage 2: the second optimization phase further reduces the jerk of the end-effector by optimizing the position of a selected joint $q^*$ at each of the way points of the robot path;
- Stage 3: the third optimization stage varies the vector of time intervals $\boldsymbol{h}$ to reduce a weighted sum of the total task time and the jerk at the end-effector, as in Stage 1.

The optimization problem in Stage 1 considers the following cost function composed of two contributions: the sum $T$ of each of the time intervals $h_i$, with $1 \leq i \leq m$ (being $m$ the number of way points of the robot path); and the sum of the six values (three translational and three rotational) of the integral of the squared jerk of the end-effector $\dddot{X}_n$, with $1 \leq n \leq 6$, along the robot trajectory $(q, \dot{q}, \ddot{q}, \dddot{q})$:

$$\min_{\boldsymbol{h}} \quad \alpha \sum_{i=1}^{m-1} h_i + (1-\alpha) \sum_{n=1}^{6} \int_0^T \dddot{X}_n^2(q, \dot{q}, \ddot{q}, \dddot{q}) dt \tag{1}$$

subject to:

$$\begin{cases} \boldsymbol{q}_{min} < \boldsymbol{q} < \boldsymbol{q}_{max} \\ |\dot{\boldsymbol{q}}| < \dot{\boldsymbol{q}}_{max} \\ |\ddot{\boldsymbol{q}}| < \ddot{\boldsymbol{q}}_{max} \\ |\dddot{\boldsymbol{q}}| < \dddot{\boldsymbol{q}}_{max} \\ |\boldsymbol{\tau}| < \boldsymbol{\tau}_{max} \\ |\dot{\boldsymbol{\tau}}| < \dot{\boldsymbol{\tau}}_{max} \end{cases} \quad \text{with} \quad \begin{cases} \boldsymbol{q}_{min}, \boldsymbol{q}_{max} \text{ joint position limits} \\ \dot{\boldsymbol{q}}_{max} \text{ joint velocity limit} \\ \ddot{\boldsymbol{q}}_{max} \text{ joint acceleration limit} \\ \dddot{\boldsymbol{q}}_{max} \text{ joint jerk limit} \\ \boldsymbol{\tau}_{max} \text{ joint torque limit} \\ \dot{\boldsymbol{\tau}}_{max} \text{ joint torque rate limit} \end{cases} \tag{2}$$

The cost function in (1) corresponds to the one proposed by Gasparetto and Zanotto in [18]. To solve the optimization problem (1), the centripetal distribution [1] of the intermediate time intervals $\boldsymbol{h}$ is used as initial condition. The parameter $\alpha$ is not used as an optimization variable, but only as a weight that balances the two contributions of the cost function in (1): the term proportional to the execution time and the term proportional to the integral of the squared jerk. The parameter $\alpha$, which varies between 0 and 1, weights the cost function between the two cases of minimum jerk ($\alpha = 0$), and minimum time ($\alpha = 1$). The optimization problem in (1) with (2) can be solved for any value of $\alpha$ within its bounds for the desired time/smoothness tradeoff, or by imposing the total execution time $T$. In case the total time of the robot

trajectory is fixed or given a priori, the cost function in (1) can be expressed as follows:

$$\min_{\boldsymbol{h}} \quad J = \sum_{n=1}^{6} \int_0^T \dddot{X}_n^2(q, \dot{q}, \ddot{q}, \dddot{q}) dt \tag{3}$$

where $J$ is defined as the integral of the squared jerk of the robot end-effector. The vector $\dddot{\boldsymbol{X}}$ of the six components $\dddot{X}_n$ of the jerk of the end-effector is obtained with the following equation:

$$\dddot{\boldsymbol{X}} = \boldsymbol{J}\dddot{\boldsymbol{q}} + 2\dot{\boldsymbol{J}}\ddot{\boldsymbol{q}} + \ddot{\boldsymbol{J}}\dot{\boldsymbol{q}} \tag{4}$$

where $\boldsymbol{J}$ is the Jacobian matrix of the considered redundant manipulator. Please see Appendix for the complete formulation of the first and second time derivatives of $\boldsymbol{J}$. The cost function $J$ in (3) does not depend on the measured acceleration, but only on the values of the squared jerk of the robot end-effector $\dddot{\boldsymbol{X}}$ computed as in (4) through the Jacobian matrix $\boldsymbol{J}$, its derivaties $\dot{\boldsymbol{J}}$ and $\ddot{\boldsymbol{J}}$, and the desired joint-space trajectory of the robot defined by ($\boldsymbol{q}$, $\dot{\boldsymbol{q}}$, $\ddot{\boldsymbol{q}}$, $\dddot{\boldsymbol{q}}$).

To verify the joint torque values during the robot trajectory, the dynamics of the robot is computed as:

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{F}_v\dot{\boldsymbol{q}} + \boldsymbol{f}_c\,\mathrm{sign}(\dot{\boldsymbol{q}}) + \boldsymbol{g}(\boldsymbol{q}) \tag{5}$$

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 $\boldsymbol{f}_c$ represent the viscous and Coulomb friction terms, respectively, whereas $\boldsymbol{g}(\boldsymbol{q})$ accounts for the gravitational contribution. The torque rate $\dot{\boldsymbol{\tau}}$ is computed as the incremental ratio of joint torque $\boldsymbol{\tau}$ over time. In this work, the simple Coulomb model is selected to describe joint friction effects. However, alternative friction models are available in the literature, as described in [39]. Furthermore, we consider the dynamic parameters of the robot to be well known. However, if the available robot dynamics is imprecise, an alternative formulation to compute torques bounds from uncertain dynamics can be implemented, as for instance using a recursive Newton-Euler algorithm based on interval arithmetic [40,41].

In this work, we consider the trajectory for the robot planned in the joint space as a 434 spline curve, i.e., a third-degree polynomial is used to interpolate all the way points, except in the first and last segments, where the trajectory is described using four-degree polynomial functions [20]. However, the proposed optimization approach can be applied to alternative motion primitives as well, as long as they comply with a similar parameterization.

The 434 spline function used in this work requires as input the specification of $m$ way points in the joint space through which the trajectory must pass at the time instants $\boldsymbol{h}$, together with the initial and final joint velocities that are usually set equal to zero. Therefore, the parameterization of the trajectory is given by the time intervals. In this way, a complex trajectory planning problem is managed as a simple problem that only requires the definition of a low number of parameters (e.g., the $m-1$ time intervals). If, for the sake of simplicity, we consider a single axis of the robot, in the 434 spline interpolation, the trajectory at time $t$ between two adjacent way points $P_k$ and $P_{k+1}$ with $2 \le k \le m-2$ can be computed as follows:

$$F_k(t) = B_{k,1} + B_{k,2}\,t + B_{k,3}\,t^2 + B_{k,4}\,t^3 \tag{6}$$

where the $B_k$ are the coefficients of the polynomial equation. The values of the four coefficients $B_k$ are univocally defined through the boundary positions and velocities. The motion law between the first two way points $P_1$ and $P_2$, and between the last two way points $P_{m-1}$ and $P_m$ (i.e., $k = 1$ and $k = m-1$) is defined as:

$$F_k(t) = B_{k,1} + B_{k,2}\,t + B_{k,3}\,t^2 + B_{k,4}\,t^3 + B_{k,5}\,t^4 \tag{7}$$

The $B_k$ polynomial coefficients are computed to ensure continuity in position, velocity and acceleration. However, jerk continuity is not guaranteed, as the jerk varies linearly or is constant during each trajectory segment.

Once the time-jerk optimization in (1) with (2) is solved and the optimal vector of time intervals $\boldsymbol{h}$ is found, the optimization problem of Stage 2 leverages the position $q^*$ of a selected robot joint at each of the way points of the robot path to further reduce the jerk of the end-effector. $q^*$ serves as the parameter that changes the configuration of the robot without altering the position and orientation of the end-effector. More in detail, the following optimization problem is defined:

$$\min_{q^*} \quad \sum_{n=1}^{6} \int_0^T \dddot{X}_n^2(q, \dot{q}, \ddot{q}, \dddot{q}) dt \tag{8}$$

subject to the constraints in (2). In (8), the vector $\boldsymbol{q}^*$ includes the positions of a selected joint in each of the way points of the robot path. For instance, $\boldsymbol{q}^* = [q^*_{wp1}, q^*_{wp2}, q^*_{wp3}, q^*_{wp4}]$, in case the robot path is defined by four way points.

The positions of the selected redundant joint $q^*$ in the way points are considered as optimization variables. The proposed strategy can be applied to any of the joints of a reduntant manipulator, as described below. However, establishing a criterion for selecting the best joint whose positions are to be optimized is beyond the scope of this paper. At each step of the optimization, the robot configuration in each of the way points is found via inverse kinematics after the position of the selected joint $q^*$ is defined. Then, the 434 spline trajectory is planned, and the jerk at the robot end-effector is computed. This process runs until the optimal solution is found. The theoretical basis of the proposed approach is the use of the redundancy and the solution of the inverse kinematics of the redundant manipulator to choose the configuration of the robot in each way point that minimizes the jerk of the trajectory, and, in turn, the induced mechanical vibrations. By optimizing in Stage 2 the position of one joint at each of the way points and then solving the inverse kinematics of the manipulator, the configuration of the robot is not left to chance, but it is optimized for a specific purpose, i.e., to obtain a smoother trajectory.

Finally, Stage 3 consists of a further refinement of the jerk minimization procedure, obtained with another time-jerk optimization equal to the one in Stage 1 (i.e., (1) with (2)). This third optimization problem starts from the results of Stage 2, and looks for a new optimal time vector that best fits the optimal configuration of the robot at each way point found in the previous stage.

The proposed optimization problem is designed to be applied to redundant manipulators, i.e., robots that possess more degrees of freedom $N$ than the number $R$ of operational space variables necessary to specify a given task [30]. In this work, we consider intrinsically redundant robots with $N - R = 1$. However, the approach can be extended to robotic systems with a higher degree of redundancy. Furthermore, the proposed approach allows us to easily consider the kinematics and dynamics limits of the robot in (2) as constraints of the optimization problem.

To implement the proposed approach, the nominal kinematics and dynamics parameters of the robot must be known. In this work, we do not consider kinematics and dynamics uncertainties, and the robot is assumed to be kinematically calibrated. Finally, to solve the optimization problems in (1) and (8) the kinematics and dynamics limits of the robot reported in (2) must be available.

## 3. Experimental setup

The proposed approach is implemented on a Franka Emika Panda arm with 7 DOFs, available at the Mechatronics and Robotics Lab of University of Udine (Italy), and shown in Fig. 1(a). This intrinsically redundant manipulator has a reachability of 855 mm and a maximum payload of 3 kg. The robot is controlled via ROS Melodic Morenia and Python 3.6 using a workstation running Ubuntu 18.04 LTS Bionic Beaver with an Intel i5-10600k and 32 GB of RAM. The trajectories are sent to the robot by specifying the sequence of joint positions, velocities, and accelerations using a custom Python script. The Franka Emika arm can serve as a good experimental setup not only for research

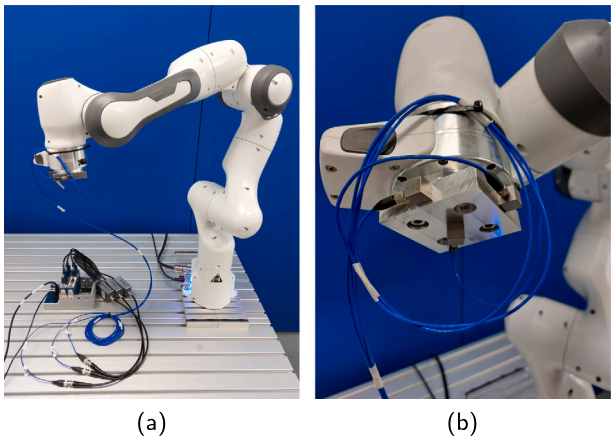



**Fig. 1.** Franka Emika Panda arm and data acquisition devices used in the experimental tests (a); close view of the custom aluminum flange and PCB accelerometers (b).

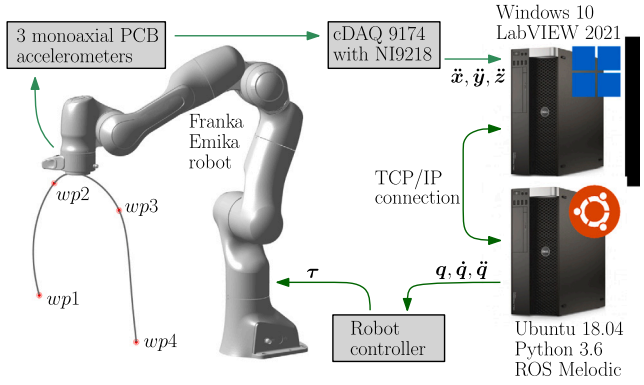



**Fig. 2.** Overview on the experimental setup and data acquisition architecture.

in pure trajectory planning, but also for collaborative robotics applications, since it is lightweight, has a payload to weight ratio comparable with similar manipulators, and is capable of providing torque measures at joints [42].

To measure the vibrations of the robot end-effector during the experimental tests, three PCB piezoelectric mono-axial accelerometers are mounted on the robot tool by means of a custom aluminum flange (Fig. 1(b)), as in [43]. The three accelerometers are placed on three orthogonal planes so as to measure the vibrations along the $x$, $y$, and $z$ axis of the robot end-effector. The signal from the accelerometers is acquired by means of two NI9218 C Series modules by National Instruments connected to a cDAQ 9174 chassis, and sent to a second workstation with the same technical specifications, running Windows 10. Data acquisition is performed in LabVIEW 2021 through a Virtual Instrument based on a state-machine architecture that communicates with the workstation running Ubuntu through a TCP/IP connection. The state machine waits a trigger signal to start the recording so as to be synchronized with the trajectory execution of the robot. The vibration signal from the accelerometers is acquired at 10 kHz. Fig. 2 shows an overview on the experimental setup and data acquisition architecture.

The measured accelerations are only used to validate the proposed approach, showing that the induced vibrations obtained with the proposed approach are lower in terms of both root mean square and absolute values with respect to those obtained in the reference case. A direct measure of the jerk is extremely challenging to perform, and the numerical differentiation of the noisy signals of the accelerometers is also critical [38].

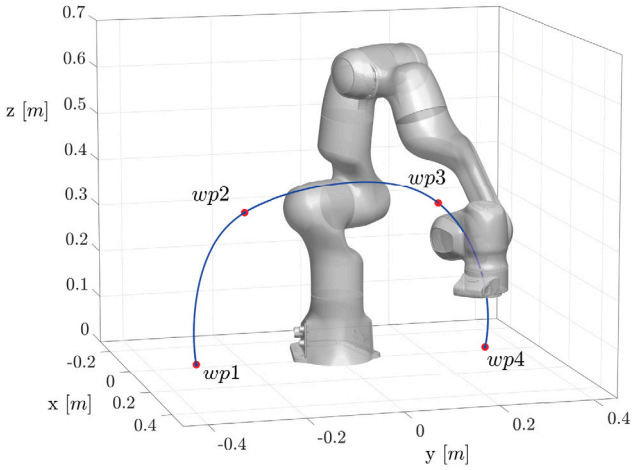



**Fig. 3.** Way points of the pick-and-place task (red markers) and example of the 3D path of the robot (blue curve).

The optimization problems in (1) with (2) is solved in Matlab using *fmincon* and the sequential quadratic programming (SQP) algorithm. The maximum number of iterations is set equal to 500. The optimization problem in (8) with (2) is solved through the Matlab genetic algorithm to better explore the domain of the optimization variables. In this case, the population size is set equal to ten times the number of the considered way points, whereas the maximum number of generations is equal to 50. Furthermore, the inverse kinematics of the redundant robot is solved using the *ikcon* function of the Matlab Robotics Toolbox. *ikcon* searches for the optimal configuration of the robot to satisfy the end-effector position and orientation, using *fmincon* and the *active-set* algorithm. The dynamic model of the Franka Emika robot identified in [44] is used to verify the torque values during the robot trajectories as in (5), whereas the kinematics and dynamics limits of the robot are taken from [45].

The trajectory chosen to validate the proposed approach is a pick-and-place motion defined by means of four way points in the Cartesian space of the robot (Fig. 3), imposing initial and final joint velocities equal to zero. The desired pose of the robot $[x, y, z, \varphi, \vartheta, \psi]$ for each of the way points of the pick-and-place task is reported in Table 1. The Roll-Pitch-Yaw angles are used for the representation of the end-effector orientation, and the origin of the reference frame is considered to be placed at the center of the robot base. We chose to keep the orientation of the end-effector constant at all the path way points (as in a common pick-and-place operation), since we are measuring only linear accelerations of the end-effector and, therefore, the readability and analysis of the experimental data is facilitated in such conditions. However, the proposed approach is general, and can be applied to more complex robotics tasks that also include changes in the end-effector orientation among way points. Furthermore, in the proposed approach, the orientation of the robot end-effector is defined only at the predefined way points, and the robot is not constrained to maintain the same orientation while traveling between consecutive way points. The inverse kinematics works regardless of the orientation that is set at the way point (provided that the way point is within the robot workspace). Therefore, the proposed strategy does not depend on the orientation imposed on the robot end-effector at each specific way point.

## 4. Results

In this section, the practical benefits of the proposed approach are demonstrated through numerical and experimental results. The performance of the approach is first evaluated by means of 100 numerical tests in which the parameter $\alpha$ is increased with constant steps from

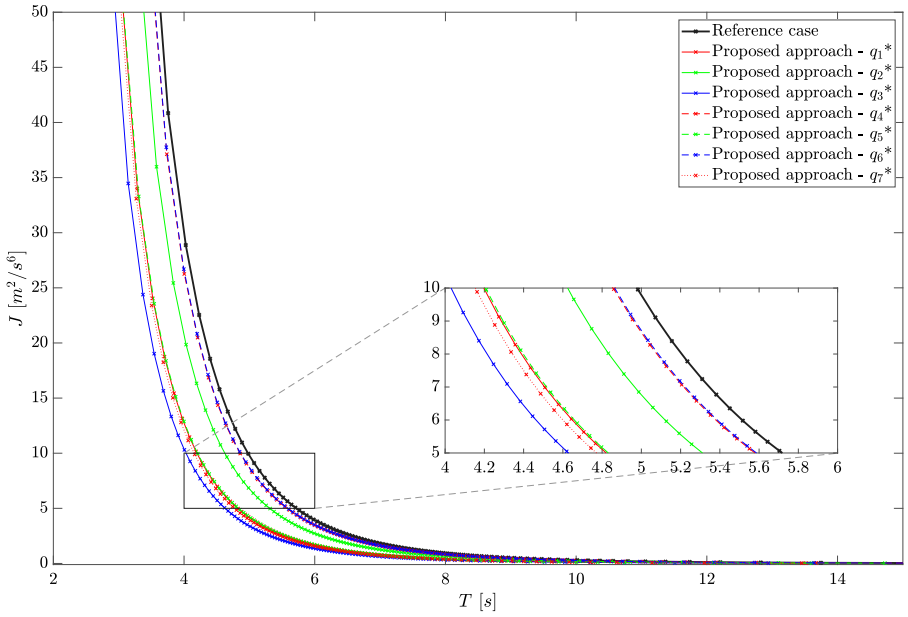



**Fig. 4.** Numerical results: time-jerk Pareto front for the optimal trajectory planning.

**Table 1**
Desired pose of the robot for each of the way points of the pick-and-place trajectory.

| Way point | $x$ [m] | $y$ [m] | $z$ [m] | $\varphi$ [rad] | $\vartheta$ [rad] | $\psi$ [rad] |
|---|---|---|---|---|---|---|
| 1 | 0.40 | −0.40 | 0.10 | $\pi$ | 0 | 0 |
| 2 | 0.40 | −0.30 | 0.40 | $\pi$ | 0 | 0 |
| 3 | 0.40 | 0.10 | 0.40 | $\pi$ | 0 | 0 |
| 4 | 0.40 | 0.20 | 0.10 | $\pi$ | 0 | 0 |

0.01 to 1, to explore a wide and comprehensive range of possible trajectories for the robot without imposing the total execution time. For each of the tests, all three stages of the optimization are performed, as described in Section 2. These tests are repeated seven times: one for each of the seven joints of the robot, whose positions are to be optimized in the optimization problem (8).

Then, a different test is performed by imposing the total time of the pick-and-place task equal to $T = 5$ s. In this case, in Stage 1 and in Stage 3, the optimization problem in (3) is considered. Also in the tests with fixed total time, the minimum-jerk three-stage optimization is repeated seven times, to evaluate the performance of the choice of the selected robot joint $q^*$ to be optimized in Stage 2.

To verify the effectiveness of the proposed approach on a real test case, 100 experimental tests as $\alpha$ changes are performed on the Franka Emika Panda arm equipped with accelerometers, as described in Section 3. Extensive experiments with imposed value of the total time of the pick-and-place task are also performed. To limit the number of trials and the amount of data acquired during the experiments, only the trajectories corresponding to the selected joints that had performed best in the numerical simulations are tested on the real robot.

For both the numerical and the experimental tests, we consider the results of Stage 1 (i.e., minimum time-jerk optimization) as the reference case for comparison with the proposed approach (i.e., the whole three-stage optimization strategy based not only on the definition of the optimal time intervals $\boldsymbol{h}$, but also of the optimal positions $\boldsymbol{q}^*$ of a selected joint of the robot). As stated above, the optimization problem, composed of both time and jerk contributions in (1) and here adopted as the reference case, corresponds to the one proposed by Gasparetto and Zanotto in [18].

### 4.1. Numerical results

Fig. 4 reports the Pareto front for the time-jerk optimal trajectory planning obtained with the 100 numerical simulations by changing the value of $\alpha$. Each point in the plot represents a single test, which is identified by a pair of total time and jerk values (i.e., $(T, J)$). The black curve indicates the reference case (Stage 1, as in [18]), whereas each of the colored curves describes the results obtained with the proposed approach (three-stage optimization), by selecting and optimizing the positions of a different robot joint in Stage 2 of the proposed approach. As it can be seen from Fig. 4, all the colored curves are lower than the reference case (black curve), indicating that the proposed three-stage optimization holds to better results than the single time-jerk optimization of the reference case. More in detail, when selecting joint 4 or joint 6 (i.e., $q_4^*$ or $q_6^*$ indicated by the red and blue dashed curves, respectively), the decreasing of jerk values is only limited. Conversely, optimizing the positions of joint 3 of the robot (blue continuous curve) holds to better results in terms of time-jerk.

Table 2 reports the numerical results of the root-mean-square (RMS) and maximum (MAX) values of the accelerations of the robot end-effector, and the integral of the squared jerk $J$ for exemplary values of $\alpha$ with the proposed approach by optimizing the positions of joint 3. From the values reported in the table it can be seen that the values of acceleration and jerk increase by increasing $\alpha$, i.e., by moving from a minimum-jerk optimization to a minimum-time scenario.

Fig. 5 shows the numerical results obtained by imposing the total time of the pick-and-place task equal to $T = 5$ s. Fig. 5(a) reports the intermediate results obtained with Stage 2 of the proposed approach (optimization of joint positions). With respect to the reference case (Stage 1, as in [18]), the optimization of joint positions holds to a notable decreasing of the jerk of the end-effector, when the positions of one joint of the redundant robot is optimized (this does not apply to joints 4 and 6, whose optimizations do not hold to significant reductions of $J$). For instance, optimizing the positions of joint 3 results in a decreasing of the squared sum of jerk of 56.05% with respect to the reference case. This value reaches the percentage of 58.24% when the positions of joint 2 are optimized for each of the four way points.

The optimization of Stage 3 of the proposed approach further reduces the jerk values of the robot end-effector with respect to Stage 2.

**Table 2**
Numerical results: RMS and maximum values of the accelerations of the robot end-effector, and integral of the squared jerk for different values of $\alpha$ with the proposed approach by optimizing the positions of joint 3.

| $\alpha$ | $\ddot{x}^{RMS}$ [m/s$^2$] | $\ddot{y}^{RMS}$ [m/s$^2$] | $\ddot{z}^{RMS}$ [m/s$^2$] | $\ddot{X}^{RMS}$ [m/s$^2$] | $\ddot{x}^{MAX}$ [m/s$^2$] | $\ddot{y}^{MAX}$ [m/s$^2$] | $\ddot{z}^{MAX}$ [m/s$^2$] | $\ddot{X}^{MAX}$ [m/s$^2$] | $J$ [m$^2$/s$^6$] |
|---|---|---|---|---|---|---|---|---|---|
| 0.01 | 0.0193 | 0.0177 | 0.0311 | 0.0406 | 0.0653 | 0.0431 | 0.0563 | 0.0808 | 0.0293 |
| 0.2 | 0.0958 | 0.0880 | 0.1544 | 0.2019 | 0.3247 | 0.2143 | 0.2799 | 0.4015 | 0.4242 |
| 0.4 | 0.1565 | 0.1437 | 0.2524 | 0.3299 | 0.5313 | 0.3499 | 0.4579 | 0.6568 | 0.9614 |
| 0.6 | 0.2346 | 0.2156 | 0.3784 | 0.4946 | 0.7956 | 0.5248 | 0.6860 | 0.9838 | 1.8880 |
| 0.8 | 0.3833 | 0.3520 | 0.6183 | 0.8082 | 1.2997 | 0.8570 | 1.1208 | 1.6071 | 4.2788 |
| 1 | 5.2518 | 5.5037 | 7.3332 | 10.566 | 12.874 | 12.363 | 15.8821 | 18.425 | 317.17 |

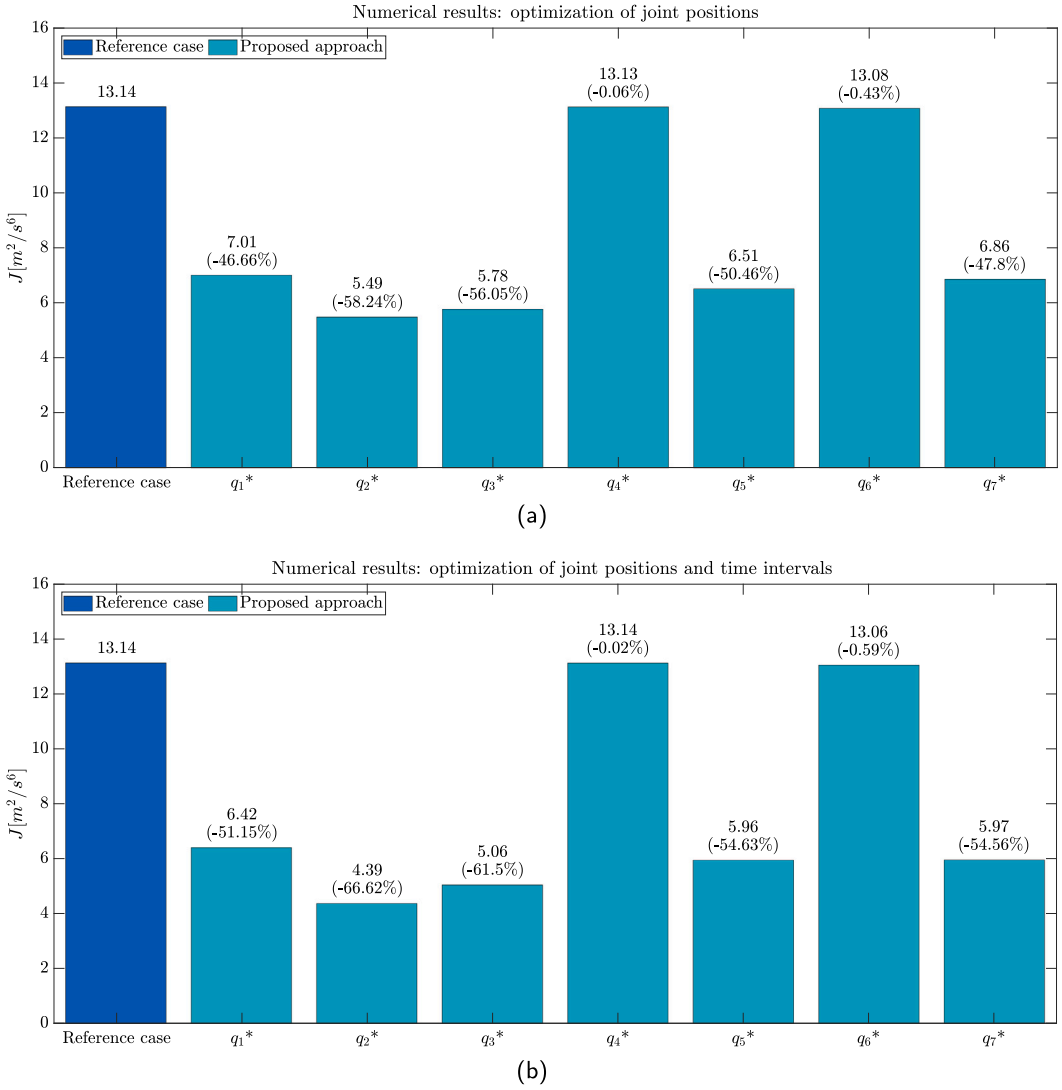



**Fig. 5.** Numerical results: RMS of the integral of squared jerk for Stage 2 of the proposed approach (optimization of joint positions) with respect to the reference case (a); RMS of the integral of squared jerk for Stage 3 of the proposed approach (optimization of joint positions and time intervals) with respect to the reference case (b).

The numerical results of Stage 3 of the proposed approach (optimization of joint positions and time intervals) with respect to the reference case are shown in Fig. 5(b). As it can be seen from the figure, the jerk is reduced with a percentage of 61.5% and 66.62% when $q_3^*$ and $q_2^*$ are considered, respectively.

Table 3 reports the RMS and maximum values of the accelerations of the robot end-effector, and the integral of the squared jerk obtained with the proposed approach by optimizing the positions of the different robot joints. The values in Table 3 show that joints 2 and 3 are those that provide better results in terms of acceleration and jerk reduction with respect to the other robot joints.

### 4.2. Experimental results

Fig. 6 illustrates the time-acceleration Pareto front obtained with the 100 experimental tests carried out by changing the value of $\alpha$ with the proposed optimal trajectory planning approach, and repeated for both the reference case and five of the robot joints (1, 2, 3, 5 and 7). No experimental tests are conducted for the trajectories with optimal positions of $q_4^*$ and $q_6^*$, since no valuable results were obtained in the corresponding numerical simulations (see Figs. 4 and 5). Each point in the plot of Fig. 6 represents the absolute acceleration of the robot end-effector measured with the PCB accelerometers. As it can be seen from

**Table 3**
Numerical results: RMS and maximum values of the accelerations of the robot end-effector, and integral of the squared jerk obtained with the proposed approach by optimizing the positions of the different robot joints.

| | $\ddot{x}^{RMS}$ [m/s$^2$] | $\ddot{y}^{RMS}$ [m/s$^2$] | $\ddot{z}^{RMS}$ [m/s$^2$] | $\ddot{X}^{RMS}$ [m/s$^2$] | $\ddot{x}^{MAX}$ [m/s$^2$] | $\ddot{y}^{MAX}$ [m/s$^2$] | $\ddot{z}^{MAX}$ [m/s$^2$] | $\ddot{X}^{MAX}$ [m/s$^2$] | $J$ [m$^2$/s$^6$] |
|---|---|---|---|---|---|---|---|---|---|
| Reference case | 0.5069 | 0.5239 | 0.6586 | 0.9824 | 1.5727 | 1.1049 | 1.2567 | 2.0861 | 13.14 |
| Proposed - $q_1^*$ | 0.4625 | 0.4181 | 0.6167 | 0.8769 | 1.5454 | 0.8901 | 1.2784 | 1.9862 | 6.42 |
| Proposed - $q_2^*$ | 0.3315 | 0.4485 | 0.4638 | 0.7254 | 1.0408 | 1.1129 | 0.7705 | 1.5238 | 4.39 |
| Proposed - $q_3^*$ | 0.4100 | 0.4094 | 0.5469 | 0.7967 | 1.3199 | 0.8851 | 1.1219 | 1.7125 | 5.06 |
| Proposed - $q_4^*$ | 0.5070 | 0.5238 | 0.6587 | 0.9825 | 1.5742 | 1.1043 | 1.2577 | 2.0869 | 13.14 |
| Proposed - $q_5^*$ | 0.4323 | 0.4449 | 0.5312 | 0.8167 | 1.3205 | 0.9932 | 1.0774 | 1.7772 | 5.96 |
| Proposed - $q_6^*$ | 0.5054 | 0.5226 | 0.6571 | 0.9799 | 1.5707 | 1.1028 | 1.2527 | 2.0837 | 13.06 |
| Proposed - $q_7^*$ | 0.4245 | 0.4376 | 0.5182 | 0.8001 | 1.2774 | 0.9783 | 1.0387 | 1.7325 | 5.97 |

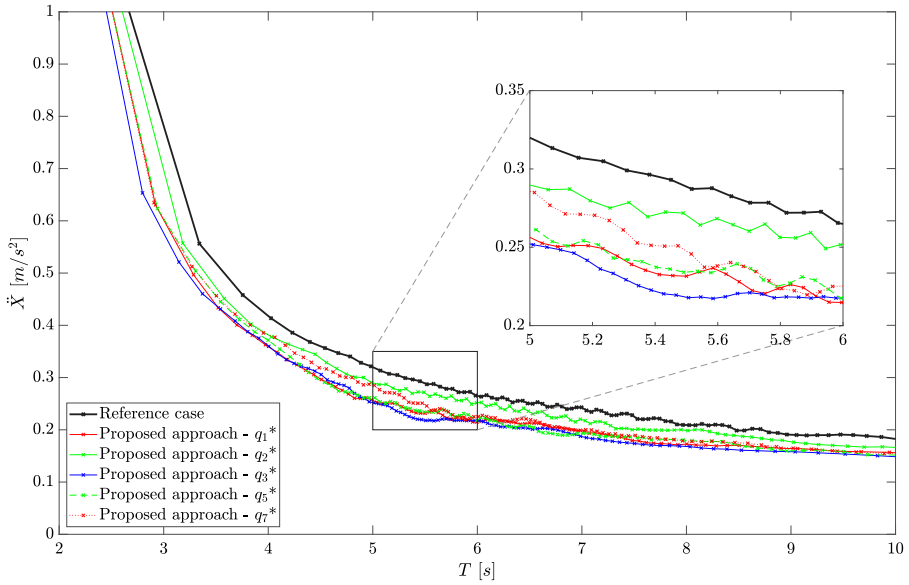



**Fig. 6.** Experimental results: time-acceleration Pareto front for the optimal trajectory planning.

**Table 4**
Experimental results: RMS and maximum values of the measured accelerations of the robot end-effector for different values of $\alpha$ with the proposed approach by optimizing the positions of joint 3.

| $\alpha$ | $\ddot{x}^{RMS}$ [m/s$^2$] | $\ddot{y}^{RMS}$ [m/s$^2$] | $\ddot{z}^{RMS}$ [m/s$^2$] | $\ddot{X}^{RMS}$ [m/s$^2$] | $\ddot{x}^{MAX}$ [m/s$^2$] | $\ddot{y}^{MAX}$ [m/s$^2$] | $\ddot{z}^{MAX}$ [m/s$^2$] | $\ddot{X}^{MAX}$ [m/s$^2$] |
|---|---|---|---|---|---|---|---|---|
| 0.01 | 0.0661 | 0.0852 | 0.0771 | 0.1326 | 0.3038 | 0.3719 | 0.2685 | 0.4354 |
| 0.2 | 0.0964 | 0.1100 | 0.0906 | 0.1720 | 0.4660 | 0.4997 | 0.3373 | 0.5328 |
| 0.4 | 0.1066 | 0.1388 | 0.1047 | 0.2039 | 0.4645 | 0.6514 | 0.3241 | 0.6667 |
| 0.6 | 0.1112 | 0.1455 | 0.1191 | 0.2184 | 0.4701 | 0.5280 | 0.3706 | 0.5978 |
| 0.8 | 0.1374 | 0.1928 | 0.1489 | 0.2797 | 0.4765 | 0.7169 | 0.4295 | 0.7944 |
| 1 | 0.4592 | 0.8442 | 0.9696 | 1.3651 | 1.9922 | 2.4814 | 2.5289 | 3.1041 |

the Pareto plot, all the curves corresponding to the proposed approach and different optimized joints are lower than the reference case. This testifies that the proposed approach has good performance in reducing the measured accelerations, with respect to the compared reference case in [18], not only in the numerical simulations but also in the experimental trials.

The experimental results in terms of RMS and maximum values of the measured accelerations of the robot end-effector for different values of $\alpha$ with the proposed approach by optimizing the positions of joint 3 are shown in Table 4. The upward trend of both RMS and maximum measured accelerations by increasing $\alpha$ can be clearly appreciated from the experimental results.

Fig. 7 reports the results of the measured accelerations obtained with the proposed optimization strategy by optimizing different joint positions in Stage 2 by imposing the total execution time equal to $T = 5$ s. More in detail, the measured accelerations along the $x, y, z$ axis of the robot end-effector, and the absolute acceleration are illustrated in Figs. 7(a), 7(b), 7(c), and 7(d), respectively. The lowest acceleration levels along the $x$ axis can be found by optimizing joint 3 (21.86% of acceleration reduction with respect to the reference case). The best results are found with $q_3^*$ also along the $y$ axis, where a reduction of 30.3% of the measured acceleration is recorded. The second joint is the one that results in the lowest acceleration values along the $z$ axis, with a percentage reduction of 13.4% with respect to the reference case. Finally, the lowest value of the measured absolute accelerations of the robot end-effector are found by optimizing the positions of joint 3 (−23.14% with respect to the compared approach).

Table 5 reports the experimental results in terms of RMS and maximum values of the measured accelerations of the robot end-effector during the tests obtained with the proposed approach by optimizing the positions of the different robot joints. Also in this case, the results show that joints 2 and 3 are the most promising in reducing the vibrations of the robot during the pick-and-place trajectory.

For a comparison with the reference case (i.e., the minimum-jerk trajectory planning strategy in [18]), Fig. 8 illustrates the joint positions, velocities, accelerations and jerks in the considered pick-and-place trajectory for the reference case and the proposed approach by optimizing the positions of joint 3 (one of the most promising from the results of Fig. 7 and Table 5). Furthermore, the end-effector positions, velocities, accelerations and jerks in the considered pick-and-place trajectory for the reference case and the proposed approach by optimizing the positions of the same joint are shown in Fig. 9. The joint positions of the robot for the four way points in the reference case and in the proposed approach (with optimal $q_3^*$) can be found in Tables 6 and 7, respectively. Furthermore, the time intervals $\boldsymbol{h}$ obtained with the

**Table 5**
Experimental results: RMS and maximum values of the measured accelerations of the robot end-effector during the tests obtained with the proposed approach by optimizing the positions of the different robot joints.

| | $\ddot{x}^{RMS}$ [m/s$^2$] | $\ddot{y}^{RMS}$ [m/s$^2$] | $\ddot{z}^{RMS}$ [m/s$^2$] | $\ddot{X}^{RMS}$ [m/s$^2$] | $\ddot{x}^{MAX}$ [m/s$^2$] | $\ddot{y}^{MAX}$ [m/s$^2$] | $\ddot{z}^{MAX}$ [m/s$^2$] | $\ddot{X}^{MAX}$ [m/s$^2$] |
|---|---|---|---|---|---|---|---|---|
| Reference case | 0.1776 | 0.2551 | 0.1521 | 0.3461 | 0.6746 | 1.0002 | 0.4685 | 1.0836 |
| Proposed - $q_1^*$ | 0.1460 | 0.2070 | 0.1477 | 0.2932 | 0.7021 | 0.7882 | 0.4819 | 0.8624 |
| Proposed - $q_2^*$ | 0.1416 | 0.2134 | 0.1317 | 0.2879 | 0.4997 | 0.9040 | 0.3661 | 0.9622 |
| Proposed - $q_3^*$ | 0.1388 | 0.1778 | 0.1410 | 0.2660 | 0.5854 | 0.7499 | 0.4513 | 0.7648 |
| Proposed - $q_5^*$ | 0.1450 | 0.1904 | 0.1384 | 0.2764 | 0.7290 | 0.8148 | 0.4400 | 0.8346 |
| Proposed - $q_7^*$ | 0.1439 | 0.1918 | 0.1360 | 0.2757 | 0.6715 | 0.7250 | 0.4255 | 0.7957 |

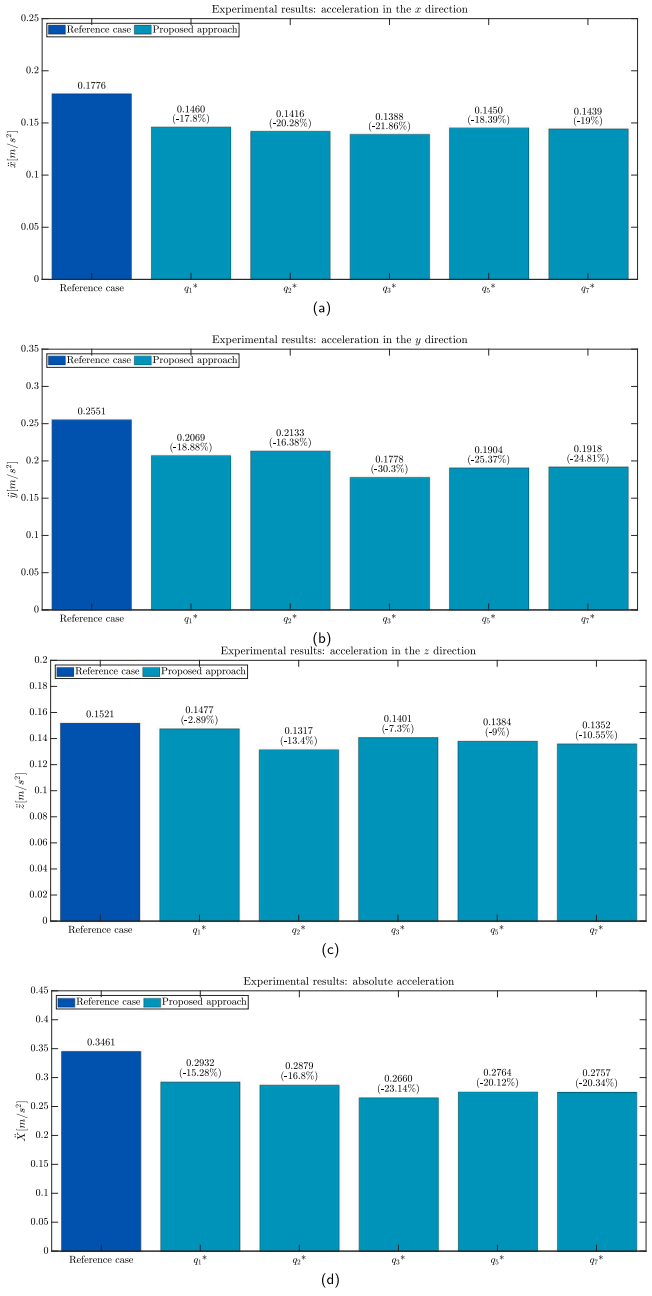



**Fig. 7.** Experimental results: measured accelerations in the $x$ (a); $y$ (b); $z$ directions (c); and absolute acceleration (d).

reference case and the proposed approach by optimizing the positions of joint 3 are shown in Table 8.

**Table 6**
Joint positions of the robot for the four way points in the reference case.

| Way point | $q_1$ [rad] | $q_2$ [rad] | $q_3$ [rad] | $q_4$ [rad] | $q_5$ [rad] | $q_6$ [rad] | $q_7$ [rad] |
|---|---|---|---|---|---|---|---|
| 1 | −0.1610 | 0.7909 | −0.5710 | −2.0460 | 0.8299 | 2.5937 | −1.3361 |
| 2 | 0.2292 | −0.1626 | −0.8511 | −2.2753 | −0.1474 | 2.1626 | −0.5325 |
| 3 | 0.2196 | −0.3740 | 0.0245 | −2.5490 | 0.0111 | 2.1751 | 0.2361 |
| 4 | 0.5452 | 0.4856 | −0.0507 | −2.5047 | 0.1550 | 2.9880 | 0.3472 |

**Table 7**
Joint positions of the robot for the four way points in the proposed approach by optimizing the positions of joint 3.

| Way point | $q_1$ [rad] | $q_2$ [rad] | $q_3^*$ [rad] | $q_4$ [rad] | $q_5$ [rad] | $q_6$ [rad] | $q_7$ [rad] |
|---|---|---|---|---|---|---|---|
| 1 | −0.6102 | 0.6654 | −0.1513 | −2.0987 | 0.2457 | 2.7481 | −0.9572 |
| 2 | −0.4121 | −0.1080 | −0.2272 | −2.2770 | −0.0293 | 2.1717 | −0.6214 |
| 3 | 0.2196 | −0.3737 | 0.0245 | −2.5490 | 0.0111 | 2.1751 | 0.2361 |
| 4 | 0.5452 | 0.4856 | −0.0507 | −2.5047 | 0.1550 | 2.9880 | 0.3472 |

**Table 8**
Time intervals in the reference case and in the proposed approach by optimizing the positions of joint 3.

| | $h_1$ [s] | $h_2$ [s] | $h_3$ [s] |
|---|---|---|---|
| Reference case | 2.0981 | 1.1204 | 1.7815 |
| Proposed approach - $q_3^*$ | 1.9283 | 1.1531 | 1.9186 |

In summary, the proposed approach effectively reduces vibrations in a 7-DOF redundant manipulator through a novel three-stage optimization procedure that considers both time intervals and joint positions for each point along the robot path. Leveraging the redundancy and the inverse kinematics of the redundant manipulator to choose the configuration of the robot at each way point is the key point of the proposed approach to minimize the jerk of the trajectory. By optimizing the position of one joint at each of the way points and then solving the inverse kinematics of the robot, its configuration is not left to chance, but it is optimized to minimize the jerk and the induced mechanical vibrations of the robot. With the proposed approach, redundant robots can be exploited for significant advantages in flexibility and trajectory optimization over conventional anthropomorphic robots with six DOFs.

One critical aspect of this methodology is the selection of one of the joints whose position is to be optimized at Stage 2 of the optimization routine. However, establishing a criterion for selecting the best robot joint whose position is to be optimized is beyond the scope of this paper. Furthermore, even if the proposed approach does not guarantee a global optimum, its performance shows a strong reduction of the jerk and the induced mechanical vibrations with respect to the reference case, as demonstrated by the extensive numerical and experimental results. More in detail, as it is testified by the results of this work that also include the measurement of the induced mechanical vibrations of the robot end-effector, optimizing the positions of the second or third joints of the robot provides better results in terms of acceleration and jerk minimization, compared to a state-of-the-art trajectory planning technique. However, also the optimization of the other joints leads to significant results in both numerical and experimental tests with respect

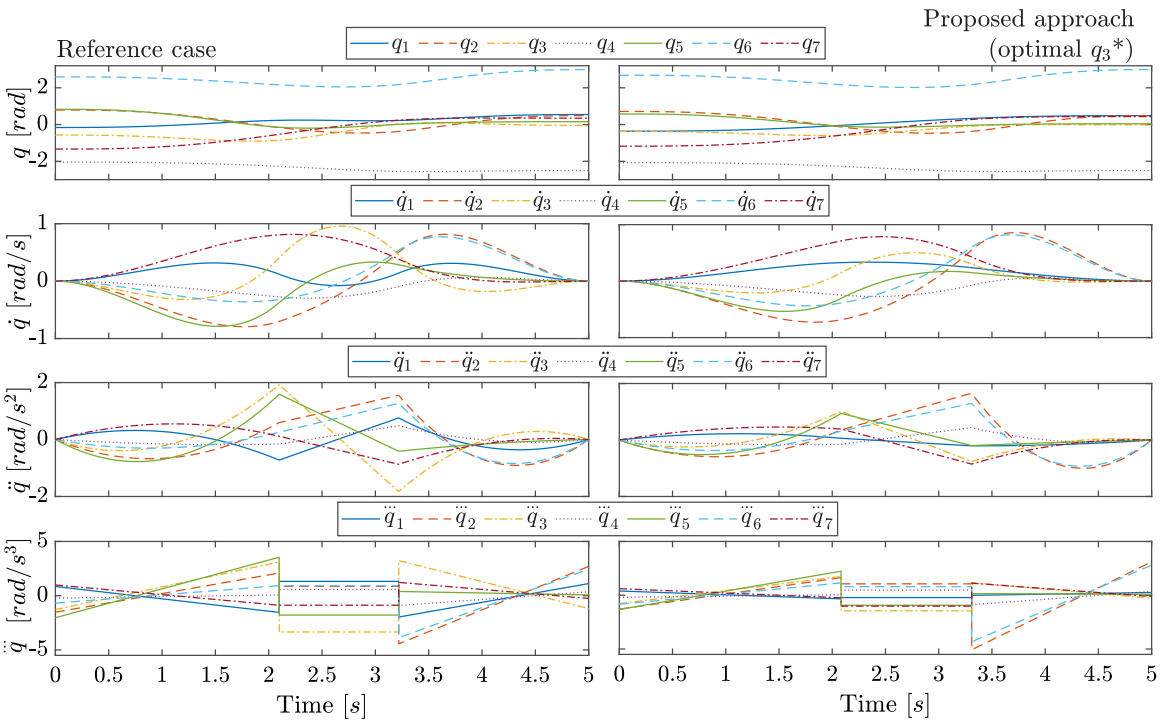



**Fig. 8.** Joint positions, velocities, accelerations and jerks for the considered pick-and-place trajectory for the reference case (left), and the proposed approach by optimizing the positions of joint 3 (right).

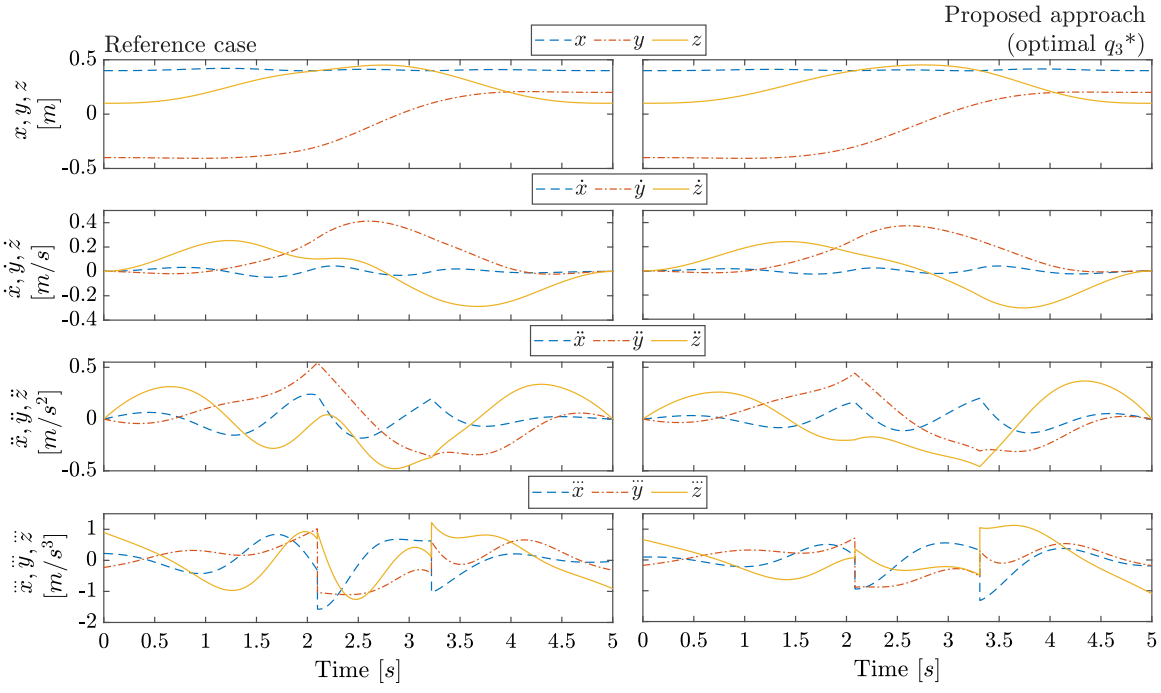



**Fig. 9.** End-effector positions, velocities, accelerations and jerks for the considered pick-and-place trajectory for the reference case (left), and the proposed approach by optimizing the positions of joint 3 (right).

to the reference case. Therefore, the proposed approach represents a significant improvement regardless of the specific joint to be optimized at Stage 2. Furthermore, the proposed approach for planning optimal minimum-jerk trajectories for redundant robots allows us to easily consider the kinematics and dynamics limits of the manipulator as constraints of the optimization problem.

Our procedure enables the optimization of jerk, a key factor in minimizing vibrations, without altering the desired pose of the robot end-effector at each trajectory point. This is achieved while meeting all the kinematics and dynamics limits of the robot, ensuring that the manipulator operates within its physical capabilities and safety limits. Furthermore, by addressing both the timing and positioning aspects of the robot motion, the proposed approach provides a comprehensive solution for enhancing the smoothness and precision of robotic operations, as well for extending the lifespan of the actuators and the mechanical components of the robot. This advancement is particularly

important in applications requiring precision, smoothness, and limited vibrations, highlighting the importance of leveraging the additional degrees of freedom in redundant robots for optimizing the overall performance of the system.

## 5. Conclusions

In this paper, an approach for planning minimum-jerk trajectories for redundant manipulators has been presented. The method takes advantage of optimizing not only the time intervals between each way point of the designated path, but also the positions of a specific robot joint to further minimize the jerk of the robot end-effector. This multi-phase optimization strategy has been validated through comprehensive numerical simulations and experimental trials on a redundant manipulator with seven degrees of freedom. The outcomes of the tests, supported by accelerometer measurements of the induced mechanical vibrations of the robot in motion, demonstrated the effectiveness of the proposed method in reducing both the acceleration and jerk levels of the redundant manipulator, even when compared to a state-of-the-art trajectory planning technique. The proposed approach is general and can be easily applied to other robotic manipulators with seven degrees of freedom, provided that their kinematics and dynamics parameters are known.

Future works will include the extension of the proposed approach to other 7-DOF robots as well as to more complex robotic systems with a different number of degrees of freedom to obtain a reduction of vibrations in more challenging scenarios. In the case of robots with a higher number of degrees of freedom ($N - R > 1$) the approach can be extended by considering the optimization of the position of multiple joints at each way point. On the other hand, in the case of non-redundant robots, the approach can be adapted by leveraging the functional redundancy of a task instead of the structural redundancy of the robot itself. Moreover, we intend to exploit joint positions together with redundant spatial variables in defining specific tasks, such as end-effector orientation, to achieve optimal minimum-jerk trajectory planning for diverse robotic manufacturing applications.

Establishing a general strategy for selecting in advance the best joint whose positions to optimize and explore mathematically which joint will give the best results in terms of reduction of the jerk and the mechanical vibrations of the robot is an essential part of future development of this work. In this way, we could improve the proposed approach to further reduce the jerk of the robot trajectory and the induced mechanical vibrations, as well as to guarantee a global optimum solution of the optimization problem.

## CRediT authorship contribution statement

**Federico Lozer:** Writing – original draft, Software, Investigation, Data curation. **Lorenzo Scalera:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Paolo Boscariol:** Supervision, Methodology, Investigation, Conceptualization. **Alessandro Gasparetto:** Supervision, Project administration, Funding acquisition.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgments

The first author acknowledges support from the National Doctorate in Robotics and Intelligent Machines of University of Genoa (Italy). The authors thank LAMA FVG (Udine, Italy) for the fabrication of the aluminum flange. This research was partially developed within the Laboratory for Artificial Intelligence for Human-Robot Collaboration funded by Fondazione Friuli (Italy), and the Laboratory for Big Data, IoT, Cyber Security funded by Friuli Venezia Giulia (Italy). This study was partially carried out within the Interconnected Nord-Est Innovation Ecosystem (iNEST) and received funding from the European Union Next-GenerationEU (Piano Nazionale di Ripresa e Resilienza (PNRR), Missione 4, Componente 2, Investimento 1.5, D.D. 1058 23/06/2022, ECS00000043). This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them. We thank the Reviewers for the constructive feedback that helped us to improve the manuscript.

## Appendix. Computation of the Jacobian and its time derivatives

In the following, we briefly recall the expressions for the computation of the $i$th column of the Jacobian matrix $\boldsymbol{J}$, as well as of its first $\dot{\boldsymbol{J}}$ and second time derivatives $\ddot{\boldsymbol{J}}$, according to the Denavit-Hartenberg (DH) convention and with reference to the notation of the classic robotics textbook by Siciliano et al. [30]. For the sake of completeness, the cases of both prismatic ($p$) and revolute ($r$) joints are reported. According to the DH convention as formulated by Siciliano et al. the axis $i$ denotes the axis of the joint connecting link $i-1$ to link $i$. The unit vector of the $z_i$ axis is defined along the axis of joint $i+1$: it is the axis about which there is rotation in the case of a revolute joint, or translation in the case of a prismatic joint. In the following, $\mathbf{p}$ is the position vector of the robot end-effector, $\mathbf{p}_{i-1}$ represents the position vector of the $(i-1)$-th reference frame, $\mathbf{z}_{i-1}$ the unit vector of the $z$ axis of the $(i-1)$-th reference frame, whereas $\boldsymbol{\omega}_{i-1}$ is the contribution to the angular velocity of the $(i-1)$-th joint. The $(i-1)$-th column of the Jacobian for a prismatic and a revolute joint, respectively, is equal to:

$$\boldsymbol{J}_i, p = \begin{bmatrix} \mathbf{z}_{i-1} \\ \mathbf{0} \end{bmatrix} \tag{9}$$

$$\boldsymbol{J}_i, r = \begin{bmatrix} \mathbf{z}_{i-1} \times (\mathbf{p} - \mathbf{p}_{i-1}) \\ \mathbf{z}_{i-1} \end{bmatrix} \tag{10}$$

The first time derivative of (9) and (10) is given by:

$$\dot{\boldsymbol{J}}_i, p = \begin{bmatrix} \boldsymbol{\omega}_{i-1} \times \mathbf{z}_{i-1} \\ \mathbf{0} \end{bmatrix} \tag{11}$$

with

$$\boldsymbol{\omega}_{i-1} = \boldsymbol{\omega}_{i-2} \tag{12}$$

$$\dot{\boldsymbol{J}}_i, r = \begin{bmatrix} (\boldsymbol{\omega}_{i-1} \times \mathbf{z}_{i-1}) \times (\mathbf{p} - \mathbf{p}_{i-1}) + \mathbf{z}_{i-1} \times (\dot{\mathbf{p}} - \dot{\mathbf{p}}_{i-1}) \\ \boldsymbol{\omega}_{i-1} \times \mathbf{z}_{i-1} \end{bmatrix} \tag{13}$$

with

$$\begin{cases} \boldsymbol{\omega}_{i-1} = \mathbf{z}_{i-1} \cdot \dot{\mathbf{q}}_{i-1} + \boldsymbol{\omega}_{i-2} \\ \dot{\mathbf{p}} = \mathbf{J} \cdot \dot{\mathbf{q}} \\ \dot{\mathbf{p}}_{i-1} = \boldsymbol{\omega}_{i-1} \times (\mathbf{p}_{i-1} - \mathbf{p}_{i-2}) + \dot{\mathbf{p}}_{i-2} \end{cases} \tag{14}$$

Finally, the $(i-1)$-th column of the second time derivative of the Jacobian for a prismatic joint can be computed as:

$$\ddot{\boldsymbol{J}}_i, p = \begin{bmatrix} \dot{\boldsymbol{\omega}}_{i-1} \times \mathbf{z}_{i-1} + \boldsymbol{\omega}_{i-1} \times (\boldsymbol{\omega}_{i-1} \times \mathbf{z}_{i-1}) \\ \mathbf{0} \end{bmatrix} \tag{15}$$

with

$$\begin{cases} \boldsymbol{\omega}_{i-1} = \boldsymbol{\omega}_{i-2} \\ \dot{\boldsymbol{\omega}}_{i-1} = \dot{\boldsymbol{\omega}}_{i-2} \end{cases} \tag{16}$$

or, for a revolute joint, as:

$$\ddot{\mathbf{J}}_{i,r} = \begin{bmatrix} (\dot{\boldsymbol{\omega}}_{i-1} \times \mathbf{z}_{i-1} + \boldsymbol{\omega}_{i-1} \times (\boldsymbol{\omega}_{i-1} \times \mathbf{z}_{i-1})) \times (\mathbf{p} - \mathbf{p}_{i-1}) + \cdots \\ +(\boldsymbol{\omega}_{i-1} \times \mathbf{z}_{i-1}) \times (\dot{\mathbf{p}} - \dot{\mathbf{p}}_{i-1})) + \cdots \\ +(\boldsymbol{\omega}_{i-1} \times \mathbf{z}_{i-1}) \times (\dot{\mathbf{p}} - \dot{\mathbf{p}}_{i-1}) + \mathbf{z}_{i-1} \times (\ddot{\mathbf{p}} - \ddot{\mathbf{p}}_{i-1}) \\ \dot{\boldsymbol{\omega}}_{i-1} \times \mathbf{z}_{i-1} + \boldsymbol{\omega}_{i-1} \times (\boldsymbol{\omega}_{i-1} \times \mathbf{z}_{i-1}) \end{bmatrix} \tag{17}$$

with

$$\begin{cases} \boldsymbol{\omega}_{i-1} = \mathbf{z}_{i-1} \cdot \dot{\mathbf{q}}_{i-1} + \boldsymbol{\omega}_{i-2} \\ \dot{\boldsymbol{\omega}}_{i-1} = (\boldsymbol{\omega}_{i-1} \times \mathbf{z}_{i-1}) \cdot \dot{\mathbf{q}}_{i-1} + \mathbf{z}_{i-1} \cdot \ddot{\mathbf{q}}_{i-1} + \boldsymbol{\omega}_{i-2} \\ \dot{\mathbf{p}} = \mathbf{J} \cdot \dot{\mathbf{q}} \\ \dot{\mathbf{p}}_{i-1} = \boldsymbol{\omega}_{i-1} \times (\mathbf{p}_{i-1} - \mathbf{p}_{i-2}) + \dot{\mathbf{p}}_{i-2} \\ \ddot{\mathbf{p}} = \dot{\mathbf{J}} \cdot \dot{\mathbf{q}} + \mathbf{J} \cdot \ddot{\mathbf{q}} \\ \ddot{\mathbf{p}}_{i-1} = \dot{\boldsymbol{\omega}}_{i-1} \times (\mathbf{p}_{i-1} - \mathbf{p}_{i-2}) + \boldsymbol{\omega}_{i-1} \times (\dot{\mathbf{p}}_{i-1} - \dot{\mathbf{p}}_{i-2}) + \ddot{\mathbf{p}}_{i-2} \end{cases} \tag{18}$$

## Data availability

Data will be made available on request.

## References


[1] L. Biagiotti, C. Melchiorri, Trajectory Planning for Automatic Machines and Robots, Springer Science & Business Media, 2008.

[2] A. Piazzi, A. Visioli, Global minimum-time trajectory planning of mechanical manipulators using interval analysis, Internat. J. Control 71 (4) (1998) 631–652.

[3] T. Chettibi, H. Lehtihet, M. Haddad, S. Hanchi, Minimum cost trajectory planning for industrial robots, Eur. J. Mech. A Solids 23 (4) (2004) 703–715.

[4] G. Wu, W. Zhao, X. Zhang, Optimum time-energy-jerk trajectory planning for serial robotic manipulators by reparameterized quintic NURBS curves, Proc. Inst. Mech. Eng. Part C: J. Mech. Eng. Sci. 235 (19) (2021) 4382–4393.

[5] F. Vidussi, P. Boscariol, L. Scalera, A. Gasparetto, Local and trajectory-based indexes for task-related energetic performance optimization of robotic manipulators, J. Mech. Robot. 13 (2) (2021) 021018.

[6] M. Javaid, A. Haleem, R.P. Singh, R. Suman, E.S. Gonzalez, Understanding the adoption of Industry 4.0 technologies in improving environmental sustainability, Sustain. Oper. Comput. 3 (2022) 203–217.

[7] G. Fabris, L. Scalera, A. Gasparetto, Dynamic modelling and energy-efficiency optimization in a 3-DOF parallel robot, Int. J. Adv. Manuf. Technol. 132 (5) (2024) 2677–2699.

[8] P.W. Oliveira, G.A. Barreto, G.A. Thé, A general framework for optimal tuning of PID-like controllers for minimum jerk robotic trajectories, J. Intell. Robot. Syst. 99 (2020) 467–486.

[9] G. Solak, A. Ajoudani, Online learning and suppression of vibration in collaborative robots with power tools, in: 2023 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2023, pp. 12085–12091.

[10] S. Seriani, P. Gallina, L. Scalera, V. Lughi, Development of n-DoF preloaded structures for impact mitigation in cobots, J. Mech. Robot. 10 (5) (2018) 051009.

[11] J. Flowers, G. Wiens, Prediction of human reaching pose sequences in human–robot collaboration, J. Mech. Robot. 16 (11) (2024).

[12] L. Scalera, R. Vidoni, A. Giusti, Optimal scaling of dynamic safety zones for collaborative robotics, in: 2021 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2021, pp. 3822–3828.

[13] L. Scalera, F. Lozer, A. Giusti, A. Gasparetto, An experimental evaluation of robot-stopping approaches for improving fluency in collaborative robotics, Robotica 42 (5) (2024) 1386–1402.

[14] M. Lagomarsino, M. Lorenzini, E. De Momi, A. Ajoudani, Robot trajectory adaptation to optimise the trade-off between human cognitive ergonomics and workplace productivity in collaborative tasks, in: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2022, pp. 663–669.

[15] A. Gasparetto, A. Lanzutti, R. Vidoni, V. Zanotto, Validation of minimum time-jerk algorithms for trajectory planning of industrial robots, J. Mech. Robot. 3 (3) (2011) 031003, http://dx.doi.org/10.1115/1.4004017.

[16] J. Ni, J. Mei, Y. Ding, D. Yu, Y. Duan, Y. Le, A trajectory planning approach for Delta robots considering both motion smoothness and dynamic stress, J. Mech. Robot. 15 (4) (2023) 041012.

[17] A. Piazzi, A. Visioli, Global minimum-jerk trajectory planning of robot manipulators, IEEE Trans. Ind. Electron. 47 (1) (2000) 140–149.

[18] A. Gasparetto, V. Zanotto, A new method for smooth trajectory planning of robot manipulators, Mech. Mach. Theory 42 (4) (2007) 455–471.

[19] A. Gasparetto, V. Zanotto, A technique for time-jerk optimal planning of robot trajectories, Robot. Comput.-Integr. Manuf. 24 (3) (2008) 415–426.

[20] C. Cook, C. Ho, The application of spline functions to trajectory generation for computer-controlled manipulators, in: Computing Techniques for Robots, Springer, 1984, pp. 101–110.

[21] P. Boscariol, A. Gasparetto, R. Vidoni, Planning continuous-jerk trajectories for industrial manipulators, in: Engineering Systems Design and Analysis, vol. 44861, American Society of Mechanical Engineers, 2012, pp. 127–136.

[22] S. Lu, J. Zhao, L. Jiang, H. Liu, Solving the time-jerk optimal trajectory planning problem of a robot using augmented Lagrange constrained particle swarm optimization, Math. Probl. Eng. 2017 (1) (2017) 1921479.

[23] Y. Fang, J. Hu, W. Liu, Q. Shao, J. Qi, Y. Peng, Smooth and time-optimal S-curve trajectory planning for automated robots and machines, Mech. Mach. Theory 137 (2019) 127–153.

[24] Y. Fang, J. Qi, J. Hu, W. Wang, Y. Peng, An approach for jerk-continuous trajectory generation of robotic manipulators with kinematical constraints, Mech. Mach. Theory 153 (2020) 103957.

[25] S. Lu, B. Ding, Y. Li, Minimum-jerk trajectory planning pertaining to a translational 3-degree-of-freedom parallel manipulator through piecewise quintic polynomials interpolation, Adv. Mech. Eng. 12 (3) (2020) 1687814020913667.

[26] J. Huang, P. Hu, K. Wu, M. Zeng, Optimal time-jerk trajectory planning for industrial robots, Mech. Mach. Theory 121 (2018) 530–544.

[27] M.S. Paing, N. Uchiyama, A spline-based approach to smooth and time-optimal trajectory generation for CNC machines with guaranteed kinematic constraints, Int. J. Adv. Manuf. Technol. 121 (5) (2022) 3385–3398.

[28] L. Lu, L. Zhang, C. Fan, H. Wang, High-order joint-smooth trajectory planning method considering tool-orientation constraints and singularity avoidance for robot surface machining, J. Manuf. Process. 80 (2022) 789–804.

[29] G. Wu, S. Zhang, Real-time jerk-minimization trajectory planning of robotic arm based on polynomial curve optimization, Proc. Inst. Mech. Eng. Part C: J. Mech. Eng. Sci. 236 (21) (2022) 10852–10864.

[30] B. Siciliano, L. Sciavicco, L. Villani, G. Oriolo, Robotics: Modelling, planning and control, Adv. Textb. Control Signal Process. (2009).

[31] E.J. Haug, Redundant serial manipulator inverse position kinematics and dynamics, J. Mech. Robot. 16 (8) (2024) 081008.

[32] X. Liu, B. Wan, Y. Liu, Y. Wang, Y. Zhao, Singularity analysis of actuation coordination and new indices for optimal design of redundantly actuated parallel manipulators, J. Mech. Robot. 16 (10) (2024).

[33] Q. Chang, H. Wang, D. Wang, H. Zhang, K. Li, B. Yu, Motion planning for vibration reduction of a railway bridge maintenance robot with a redundant manipulator, Electronics 10 (22) (2021) 2793.

[34] A. Reiter, A. Müller, H. Gattringer, Inverse kinematics in minimum-time trajectory planning for kinematically redundant manipulators, in: IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society, IEEE, 2016, pp. 6873–6878.

[35] S. Lu, J. Zhao, L. Jiang, H. Liu, Time-jerk optimal trajectory planning of a 7-DOF redundant robot, Turk. J. Electr. Eng. Comput. Sci. 25 (5) (2017) 4211–4222.

[36] D. Chen, S. Li, W. Li, Q. Wu, A multi-level simultaneous minimization scheme applied to jerk-bounded redundant robot manipulators, IEEE Trans. Autom. Sci. Eng. 17 (1) (2019) 463–474.

[37] C. Dai, S. Lefebvre, K.-M. Yu, J.M. Geraedts, C.C. Wang, Planning jerk-optimized trajectory with discrete time constraints for redundant robots, IEEE Trans. Autom. Sci. Eng. 17 (4) (2020) 1711–1724.

[38] P.-J. Barre, R. Bearee, P. Borne, E. Dumetz, Influence of a jerk controlled movement law on the vibratory behaviour of high-dynamics systems, J. Intell. Robot. Syst. 42 (2005) 275–293.

[39] G. Fabris, L. Scalera, P. Boscariol, A. Gasparetto, Experimental analysis and comparison of friction models applied to the UR5e robot, in: 6th IFToMM International Symposium on Mechanism Design for Robotics, MEDER 2024. Mechanisms and Machine Science, vol. 166 MMS, Springer, 2024, pp. 125–133.

[40] A. Giusti, M. Althoff, Efficient computation of interval-arithmetic-based robust controllers for rigid robots, in: 2017 First IEEE International Conference on Robotic Computing, IRC, IEEE, 2017, pp. 129–135.

[41] L. Scalera, C. Nainer, A. Giusti, A. Gasparetto, Robust safety zones for manipulators with uncertain dynamics in collaborative robotics, Int. J. Comput. Integr. Manuf. 37 (7) (2024) 887–899, http://dx.doi.org/10.1080/0951192X.2023.2258111.

[42] S. Haddadin, S. Parusel, L. Johannsmeier, S. Golz, S. Gabl, F. Walch, M. Sabaghian, C. Jähne, L. Hausperger, S. Haddadin, The Franka Emika robot: A reference platform for robotics research and education, IEEE Robot. Autom. Mag. 29 (2) (2022) 46–64.

[43] F. Lozer, L. Scalera, P. Boscariol, A. Gasparetto, An experimental setup to test time-jerk optimal trajectories for robotic manipulators, in: 32nd International Conference on Robotics in Alpe-Adria-Danube Region, RAAD 2023. Mechanisms and Machine Science, vol. 135 MMS, Springer, 2023, pp. 309–316.

[44] C. Gaz, M. Cognetti, A. Oliva, P.R. Giordano, A. De Luca, Dynamic identification of the Franka Emika Panda robot with retrieval of feasible parameters using penalty-based optimization, IEEE Robot. Autom. Lett. 4 (4) (2019) 4147–4154.
[45] Franka Emika GmbH, Robot and interface specifications, 2017, https://frankaemika.github.io/docs/control_parameters.html. (Online; Accessed 2 August 2024).

**Federico Lozer** was born in Tolmezzo in 1996. He studied Mechanical Engineering at the University of Udine (Italy), received his Bachelor's Degree in 2019, and his Master's Degree in 2022. In 2022 he was a research fellow at the Mechatronics and Robotics Lab. Since 2022, he is working at University of Udine (Italy) as a Ph.D. student of the Italian National Doctorate in Robotics and Intelligent Machines coordinated by the University of Genova (Italy). His research interests are in the fields of trajectory planning and collaborative robotics.

**Lorenzo Scalera** achieved the Master's Degree in Mechanical Engineering (cum laude) at University of Trieste in 2015, and the Ph.D. in Industrial and Information Engineering at University of Udine (Italy) in 2019. In 2018 he was a visiting Ph.D. student at the Stevens Institute of Technology in Hoboken (NJ, USA). In 2019 he was a Post Doc Research Fellow at Free University of BozenBolzano. Since 2024, he is Associate Professor of Mechanics of Machines at University of Udine. His research interests are in the fields of dynamic modeling of mechatronic and robotic systems, trajectory planning, collaborative robotics, and mobile robotics. He currently serves in the Editorial Board of IEEE Robotics and Automation Letters.

**Paolo Boscariol** achieved the Master's Degree in Electronic Engineering at the University of Udine in 2008, and the Ph.D. in Industrial and Information Engineering at the DIEGM, University of Udine, in 2021. Since 2016 has been with the Department of Management and Engineering (DTG) at the University of Padova, where he serves as an Associate Professor since 2021. His research interests include the motion planning for energy efficiency in robots and multibody systems, the dynamic modeling of mechatronic systems, the development of robust model-based motion planning algorithms for underactuated systems.

**Alessandro Gasparetto** received the M.Sc. in Electronic Engineering from the University of Padova, Italy, in 1992; the M.Sc. in Mathematics from University of Padova, Italy, in 2003; the Ph.D. in Mechanics of Machines from University of Brescia, Italy, in 1996. He is Full Professor of Mechanics of Machines at the Polytechnic Department of Engineering and Architecture, University of Udine (Udine, Italy), where he is the head of the research group in Mechatronics and Robotics, as well as the Head of the Department (since 2021). He has been included in the ranking of the top 2 quoted and authoritative scientists in the world, published by researchers at Stanford University (2019 and 2021). Since 2017, he is the Chair of IFToMM Italy, the Italian branch of IFToMM (the International Federation for the Promotion of Mechanism and Machine Science). Since 2018, he is the Chair of the IFToMM Permenent Commission for the History of Mechanism and Machine Science. His research interests are in the fields of modeling and control of mechatronic systems, robotics, mechanical design, industrial automation, mechanical vibrations. He is author of more than 200 international publications and has been involved in the scientific and organizing committees of several conferences, as well as in many research projects, at the regional, national and European level.