International Journal of Advanced Manufacturing Technology manuscript No.
(will be inserted by the editor)

A new path-constrained trajectory planning strategy for spray painting robots - Rev.1

Giulio Trigatti · Paolo Boscariol · Lorenzo Scalera · Daniele Pillan · Alessandro Gasparetto

Preprint

Abstract In this paper a novel method for planning spray painting trajectories for industrial robots is presented. The proposed method takes as input an arbitrary parametric description of the end-effector path in the operative space. red The method is aimed at providing feasible motion profiles without resorting to optimization routines and without the need for a dynamic description of the painting robot. The motion law is the defined by the algorithm to achieve end-effector speed limitation, in order to comply with the constraints imposed by the spray painting process and by the manipulator specifications. Subsequently a sequence of look-ahead filtering operations on the speed profiles ensures joint acceleration limitation. red The proposed methods has been tested on an industrial painting robot, showing its effectiveness, which is experimentally evaluated against the results obtained with the original manufacturer's proprietary planning method. The improvements include, other then the required end-effector speed and joint acceleration limitation, a sensible reduction of the cycle time and of the torque effort requirement. The method is of simple implementation and can be useful for other robot-oriented industrial tasks.

Keywords Trajectory planning · Painting robot · B-spline parametrization · Spray painting · Motion planning

G. Trigatti, D. Pillan
CMA Robotics
via Pasolini 35, 33040 Pradamano, Udine, Italy

P. Boscariol
DTG, Universitá degli Studi di Padova
Stradella S. Nicola 3, 36100 Vicenza, Italy
E-mail: paolo.boscariol@unipd.it

L. Scalera, A. Gasparetto
DPIA, Universitá degli Studi di Udine
via delle Scienze 206, 33100 Udine, Italy

1 Introduction

Spray painting is a task commonly performed by robots in industry. In spray painting the object surface is covered by means of a fluid (paint, ink, varnish, etc.) for protection or aesthetics. The painting process uses a compressed-air driven spray gun to atomize and direct the fluid particles onto the surface. Other treatments, such as coating, can be made by means of a spray process as well [1, 2, 3]. Choosing a robot in place of an operator for performing such tasks generally offers several advantages. The first and most evident one is the consistency of the results, as it is evident that robots can easily outperform the repeatability of results obtained by even the most skilled operator. The quality of the paint job is heavily affected by paint thickness, which should be as uniform as possible [4]. Not less important is the possibility of reducing the exposition of an operator to hazardous environments [5], as well as reducing the amount of gaseous polluting emission [6]. The uniformity of paint thickness is mainly affected by the trajectory followed by the paint gun, since an optimized trajectory should include a correctly defined path that covers all the areas that need to be painted, as well as a correct speed profile, to ensure optimal paint distribution [7]. A non-optimal path can, for example, introduce a too large or a too big overlap between two passes of the spray gun, resulting in non-optimal paint use and limited uniformity of the results.

The planning of the tool path for the spray operation, i.e. the definition of the sequence of positions and orientations of the robot during the task can be performed in several ways, which range from completely manual to a full automated procedures [8]. red According to a straightforward approach, the sequence of operations to be performed can be recorded by the

robot according to the actions performed by a skilled operator that simply moves a custom teaching support along the workspace, using an approach suggest by [9]. Alternatively, if the robot is equipped with additional devices which allows for it, the 'teaching by showing', also known as 'robot guiding' can be effective since it allows to integrate the teaching support within the robot [10, 11]. However such operation can be very time consuming, and during their execution the normal production operations are made impossible. Moreover, the quality of the result is inevitably affected by the experience and skill of the operator, as well as by the complexity of the part that needs to be painted.

Painting path and trajectories are usually generated using high level computer-based approaches, such as the so-called Computer-Aided Toolpath Planning (CATP) [8]. Such software tools usually work by processing a CAD model of a part to be painted and, using the specifications of the painting device and the process requirements, generates the trajectory that will be executed by the painting robot. These methodologies necessarily make use of mathematical models of the paint deposition process, through flux flow rate functions and spray cone geometry [12, 13]. The scientific literature has, over the years, proposed several methods aimed at the automatic or semi-automatic definition of a robot-assisted painting procedure. In this context, Suh et al. presented in two seminal works an automatic trajectory planning system for spray painting robots [14, 15] making use of geometrical modeling, painting mechanics and robot dynamics, whereas Asakawa et al. developed an automatic spray painting system that can generate paths starting from the CAD description of the work-pieces [16]. Other examples are given by Chen et al., who proposed a model-based automated robot path planning for spray painting of free-form surfaces [17] and by Li et at., who studied an automatic path generation system capable of building regular painting diagrams on the surface of 3D models, such as car bodies [18]. In addition, Gasparetto et al. presented a CAD-guided optimization of path planning based on the Chinese postman algorithm [19], whereas a review of CAD-based robot path planning for spray painting can be found in the work of Chen et al. [20]. Along the same line, Hausler et al. implemented an approach that uses image data from laser sensors in order to obtain the geometry of an unknown part and to automatically generate the robot spray painting paths [21], while Chen and Zhao [22] provided an optimization algorithm for spray painting time and paint thickness based on a paint deposition rate function that achieves both a more even paint thickness and a execution time reduction. The complexity of the problem of designing effective painting trajectories can also been tackled by splitting the problem into two parts, i.e. by dealing with the path definition and the speed profile definition as separate tasks. Within this approach, a path planner is used to define the sequence of points that the tool of the robot should follow to completely cover the areas to be painted. The speed profile that the robot should follow is designed in a subsequent stage, as an independent problem. The two results, i.e. the path and the speed profile, concur to the actual definition of the trajectory of the painting tool. Within this decoupled approach, path planning is usually set as a coverage path planning problem [23], in which specific procedures are used to define a proper sequence of points to ensure that the robot end-effector covers a pre-defined target area. Countless examples can be found in literature, given also the fact that this problem red is encountered in several applications, including robotic de-miniming, exploration, lawn moving and of course spray painting [24, 25, 26].

When referring to the decoupled problem, since the path is constrained and dependent on both the geometry of the objects to be painted and on the spray forming process, speed profile planning should employ specific strategies to meet the desired quality of the results. This approach is used, among many others, by [27], which assumes a predefined path and optimizes the speed profile to improve execution time and paint thickness variation.

An effective trajectory planning for painting robots must also ensure sufficient smoothness, given that high or discontinuous accelerations red affect both the quality of the painting results, both the accuracy of motion [28]. Conversely, if the total execution time of the manufacturing task is a critical parameter, the operation of the robot should be planned to ensure high speed, but without violating the robot specifications in terms of maximum joint speed and acceleration. The fulfillment of this request is the main target of the trajectory planning strategy proposed in this paper.

red The proposed method differs from the majority of the works available in literature since it does not rely on optimization routine for planning the painting operation: among the cited references, the works [1, 3, 4, 14, 17, 18, 19, 22, 27, 28] include numerical optimization routines as the core of the computation of perfected painting trajectories. Engineering researchers often resort to numerical optimization methods, but such occurrence is less frequent among industrial practitioners, which simply might not have the availability of sophisticated, and sometimes expensive, software optimization programs. Avoiding the use of optimiza-

tion, can therefore boost the field of application if the method proposed and tested in this work.

In particular, the proposed methodology takes as input a parametric description, such as a sequence of B-splines, of the path of the spray gun, red assuming that it fulfills a sufficiently accurate coverage. Such path is then used to generate a first-trial speed profile. The latter is then modified to achieve end-effector speed and acceleration limitations, chosen according to the requirements imposed by the painting process and the robot dynamic properties. red All the processing associated with the trajectory computation is performed offline, so that a fine-tuning of the results can be performed independently from the manufacturing process also using simulation data. red The proposed method requires just the availability of a numerical routine for the solution of the inverse kinematic problem, a feature which widens the application to any serial structure robot, thus including manipulators for which the inverse kinematic equation do not admit analytic solutions, such as some robots equipped with non-spherical wrists [29].

red The resulting motion profiles, defined for each individual axis of the robot, can be fed directly to the position-controlled manipulator which will execute the painting operation. The outcome of this methodology has been experimentally tested on a GR 680 industrial robot by CMA Robotics, showing substantial improvements red over the method used by the manufacturer. Such improvements include speed and acceleration limitations, as prescribed by the algorithm, as well as a reduction of the motor effort and a process speed-up.

2 The trajectory planning algorithm

The proposed trajectory planning method is based on the use of a generic description of the end-effector path in the operative space defined as $s = \mathbf{B}(u)$, in which $\mathbf{B}$ can be any convenient parametrized form of the end-effector path of the spray gun. red It is therefore assumed that the path definition problem has already been solved by decoupling it from the motion planning, in accordance with the choice operated, as a matter of example, in the works [27, 30]. The path of the tip of the spray gun is defined in order to produce a sufficiently uniform coverage of the surface to be painted, according to the workpiece shape and according to the geometry of the spray cone. Such functionality is provided by an offline software tool that might be either custom-made, either a commercially available software [8]. The trajectory of the robot is then completely defined only when the motion law $u(t)$ is fixed. If a constant sampling time T is chosen, the motion law can be sampled as $u_k = u(kT)$: the latter can be then used to evaluate the sampled trajectory by direct substitution of the u_k values in $\mathbf{B}(u)$.

2.1 Definition of the motion profile

The methodology introduced here starts by defining a sequence of u_k to ensure constant speed operation in the operative space, red or in other words, a constant translation speed of the spray gun. A constant speed operation of the spray gun would represent the ideal operative condition, if uniform paint thickness is the main priority of the finishing process. Painting practitioners choose to move the spray gun at constant speed, whenever possible [4]: the same work, after a detailed analysis of the path definition problem and the motion planning problem, also concludes that end-effector speed uniformity is the key factor when striving for paint uniformity, provided that the path planning results in a sufficient overspray. Considering that the spray gun is held, whenever possible, normal to the surface to be painted to obtain uniform electrostatic effects [4], the ideal situation is the one in which the end-effector of the robot can be moved at a fixed distance from the surface with a constant speed. Optimal speed and distances can be chosen according to the paint gun in use for the paint operation.

In the cases in which it is possible to plan the motion law of the robot so that constant tangential speed of the spray gun using a reasonable cycle time, the motion planning problem is easily solved. This however does not happen in the most general case, given the impossibility of obtaining reasonable end-effector speeds while respecting the obvious and inevitable limitation of the values of joint acceleration that need to be produced by the robot actuators. Using well-known formulas [31, 32, 33], the sampled motion law that ensures the execution of the trajectory at the desired red constant speed v_c can be evaluated iteratively as:

$$u_{k+1} = u_k + \frac{v_c T}{\left| \dfrac{d\mathbf{B}(u)}{du} \right|_{u_k}} \tag{1}$$

red The value of the desired tangential velocity v_c is usually chosen according to the chosen painting process when defining the painting path. Once the sequence of the u_k has been obtained red according to eq. (1), the end-effector trajectory is completely defined. The latter is in general either unsuitable or non-optimal, since the absence of bounds on joint velocities and accelerations might result in an infeasible trajectory. A specific strategy to enforce maximum joint and speed acceleration is

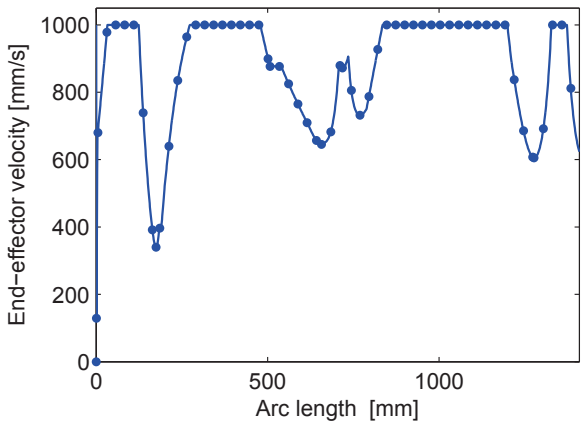

Fig. 1 Definition of first motion profile and its sampling.

therefore here introduced. The value of the $i-$th joint acceleration can be approximated from the sequence of sampled joint positions q_k^i using finite differences as:

$$\ddot{q}_{k+1}^i = \frac{q_{k+1}^i - 2q_k^i + q_{k-1}^i}{T^2} \tag{2}$$

From the last equation, knowing the maximum admissible absolute value of the acceleration for each joint $\ddot{q}_{lim}^i$, the time interval $\widehat{T}^i$, which represents the minimum time to go from q_{k-1}^i to q_{k+1}^i at constant speed for the i-th joint, can be determined as:

$$\widehat{T}^i = \sqrt{\frac{q_{k+1}^i - 2q_k^i + q_{k-1}^i}{\ddot{q}_{lim}^i}} \tag{3}$$

The maximum value of the $\widehat{T}^i$'s among all joints is then used to evaluate the tangential velocity $v_{t,k}$ at k-th time instant. In other words, a time scaling is applied to the trajectory so that the joint with maximum acceleration complies with the acceleration limit, resulting in a tangential velocity whose k-th sample is:

$$v_{t,k} = v_c \frac{T}{\max\limits_i \widehat{T}^i} \tag{4}$$

However, the velocity profile thus obtained cannot still be used as it is. The reduced speed values are computed under the assumption of constant velocity but, since the velocity along the path is not constant, it is necessary to modify the profile in order to compensate the effects of acceleration and deceleration along the whole trajectory. Such operation is performed by the look-ahead algorithm described in the following.

2.2 Updating of the motion profile with a look-ahead strategy

A re-sampling of the motion profile obtained so far is now performed to reduce the computational effort required to run the algorithm. The re-sampling procedure defines a set of uniformly spaced samples, plus a sample located at each minimum of the speed profile. An example of a speed profile, together with an adequate sampling, is shown in Fig. 1. The continuous line refers to a finely sampled speed profile, while the dots represent the reduced set of samples.

Starting from the samples of velocity just extracted, the velocity profile is updated, so as to take into account the joint accelerations. Starting from every point corresponding to a local minimum of the velocity, new velocity values for the adjacent points are computed according to the following four-step procedure:

1. compute the time interval between two consecutive points, based on the average speed $\overline{v}_{k,k+1}$ and travel time Δt_k:

$$\overline{v}_{k,k+1} = \frac{v_{t,k+1} + v_{t,k}}{2} \tag{5}$$

$$\Delta t_k = \frac{s_{k+1} - s_k}{\overline{v}_{k,k+1}} \tag{6}$$

where $v_{t,k}$ and s_k are the values of the end-effector velocity and arc length of the path at the k-th time instant, respectively.

2. compute the average joint acceleration $\ddot{q}_k$ over the same interval, whose time duration is Δt_k:

$$\ddot{q}_k^i = \frac{\dot{q}_{k+1}^i - \dot{q}_k^i}{\Delta t_k} \tag{7}$$

3. redefine the tangential velocity at the $(k+1)$-th time step as:

$$v_{t,k+1} = \frac{v_{t,k+1} - v_{t,k}}{\gamma_{acc}} + v_{t,k} \tag{8}$$

where:

$$\gamma_{acc} = \max \left| \frac{\ddot{q}_k}{\ddot{q}_{lim}} \right| \tag{9}$$

The new velocity at the $(k+1)$-th instant is reduced of a quantity proportional to the ratio γ_{acc} between the current joint acceleration and the corresponding user-defined acceleration limit $\ddot{q}_{lim}$.

4. recalculate the end-effector velocity $\mathbf{v}_{k+1}$ and the joint velocities $\dot{\mathbf{q}}_{k+1}$ at the $(k+1)$-th instant as:

$$\hat{\mathbf{v}}_{k+1} = \mathbf{v}_{k+1} \frac{v_{t,k+1}}{v_c} \tag{10}$$

$$\dot{\mathbf{q}}_{k+1} = \mathbf{J}^{-1}(\mathbf{q}_{k+1}) \hat{\mathbf{v}}_{k+1} \tag{11}$$

where $\mathbf{J}$ is the Jacobian matrix of the robot.

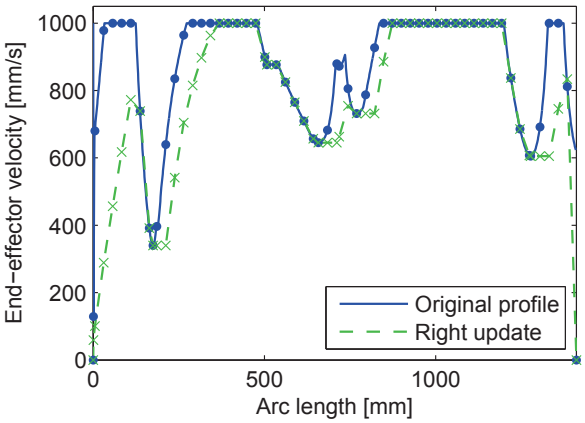

Fig. 2 Update in right direction.

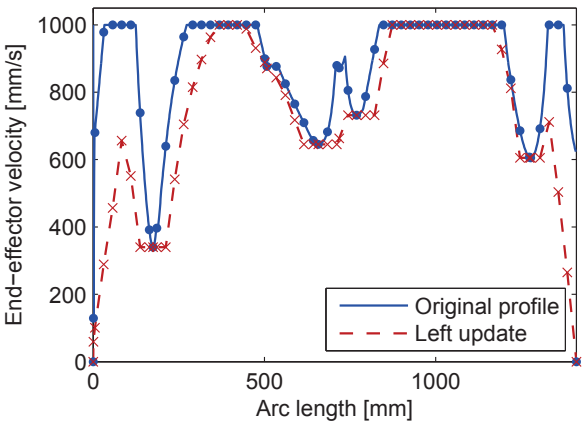

Fig. 3 Update in left direction.

This algorithm is first run 'rightwards', beginning from the first minimum point of the speed profile, by taking into consideration only the speed ramps for acceleration (Fig.2). Once the entire profile has been updated in the right direction, the algorithm is then run 'leftwards', so as to impose the constraints on the deceleration ramps (Fig.3). The use of a rightwards run followed by a leftwards run is necessary because the only points of the original velocity profile that respect the acceleration constraints are the local minima of the velocity. Starting from these points, the velocity profile is updated, by means of the look-ahead strategy, in order to ensure that the acceleration limits are respected.

2.3 Filtering of the updated velocity profile

Filtering in the space domain The updated velocity profile guarantees the respect of the acceleration limits at the joints but might present irregularities that could make the resulting trajectory not sufficiently smooth. In order to reduce these irregularities, a moving average filter with a centered window is used:

$$v_{f,s}(k) = \frac{1}{L} \sum_{l=-L/2}^{L/2} v(k-l) \tag{12}$$

where $v_{f,s}(k)$ is the sample of the filtered velocity at the k-th instant, $v(k)$ is the sequence of samples of velocity and L is the buffer length. This type of filter does not add any delay red or phase error to the signal.

Filtering in the time domain The velocity profile after the application of the space domain filtering is defined in the space domain. It can be converted to the time domain by plotting the velocity as a function of time.

Knowing the distance between two consecutive points along the path, and the velocities in correspondence of these points, the time required to travel the distance between the samples can be computed as:

$$t_{k+1} = t_k + \frac{s_{k+1} - s_k}{\overline{v}_{(k,k+1)}} \tag{13}$$

where $\overline{v}_{(k,k+1)}$ is the average of the corresponding velocities: $\overline{v}_{(k,k+1)} = (v_{k+1} + v_k)/2$. The domain conversion is done in order to easily apply the velocity profile in the robot controller, which works in the time domain. Thus, the velocity profile can be interpolated red in the time domain by a function such as:

$$v(t) = \frac{t - t_k}{t_{k+1} - t_k}(v_{k+1} - v_k) + v_k \qquad t \in [t_k, t_{k+1}] \tag{14}$$

At this stage, the velocity profile consists of a succession of ramps: hence, the acceleration profile is piecewise constant and the jerk results unbound. In order to limit jerk, a moving average filter is applied to the velocity profile. This filter is similar to the one used in the previous step, with the only difference that, instead of

centering the buffer of samples at the output sample, as in Eq. (12), in this case the buffer collects only the previous velocity samples. This different type of buffer, though introducing a phase lag to the signal, allows to obtain a more regular signal. However, in filters with a limited number of samples this delay is very small and does not affect significantly the accuracy of the profile. The profile before and after the application of the time-domain filter is plotted in Figure 5 versus time.

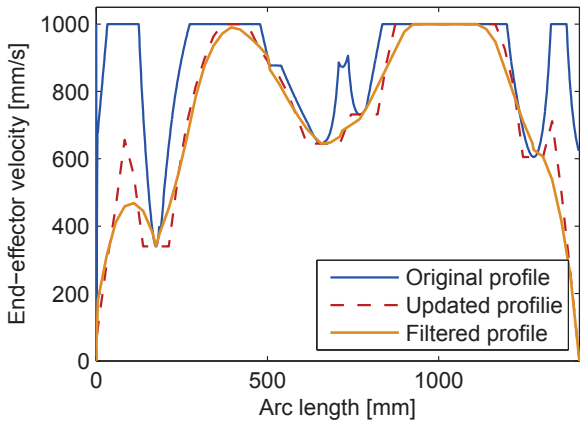

Fig. 4 Filtering in space domain.

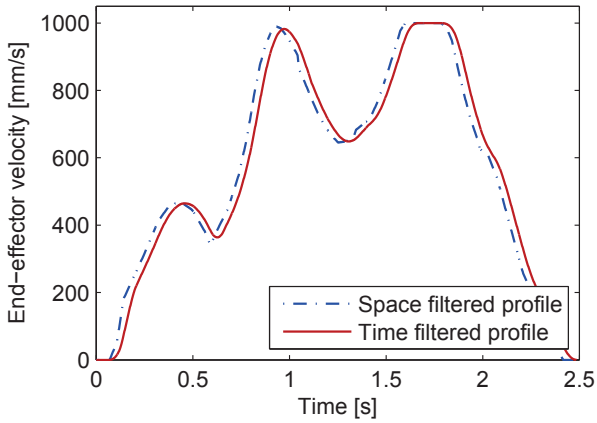

Fig. 5 Filtering in time domain.

red It must be mentioned that the last filtering operation, i.e. the filtering in the time domain, results in a slight distortion of the trajectory from the initially prescribed path, given that up to the last filtering operation the path and the motion profiles are related by the parametric description of eq. (1). At the same time, the actual accuracy in the reproduction of the spray gun path is affected also by the limited accuracy of common painting robots, which are designed for long reach and dexterity rather than for pronounced accuracy. It must be however recalled that an exact reproduction of the prescribed path is not strictly needed, given that a few millimeters deviation does not significantly affect the outcome of the painting procedure as far as painting thickness uniformity is concerned.

The time domain filtering is the last step of the proposed planning procedure. The resulting trajectory can be fed to robot controller to perform the spray-painting task.

3 Experimental tests and results

The trajectory planning method explained in Section 2 has been implemented in Matlab^{TM} and the resulting trajectories have been tested on an industrial painting robot. The robot of choice, designed and manufactured by CMA Robotics, is a 6 d.o.f. manipulator which comprises an anthropomorphic robot equipped with a non-spherical wrist, as shown in Figs. 6-7. red The robot is operated by a position control loop which is embedded in the robot control unit. Motion is therefore specified as a sequence uniformly sampled joint positions.

Fig. 6 Robot GR 680, photo courtesy of CMA Robotics.

Fig. 7 The non-spherical wrist mounted on the robot.

The path chosen for the experimental tests is design as a single B-spline curve, built using 17 via-points located on a vertical plane placed at 2 m distance from the origin of the reference frame located at the base of the robot. The resulting path, whose overall length is equal to 9.16 m, is shown in Figure 8.

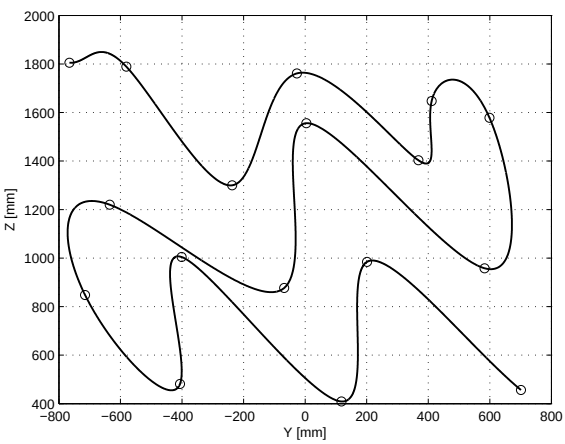

Fig. 8 End-effector path.

A first motion profile, used here as a benchmark to show the improvements brought by the application of the proposed method, is designed according to the proprietary method implemented by the robot manufacturer. The corresponding tangential speed of the end-effector is shown in Figure 9.

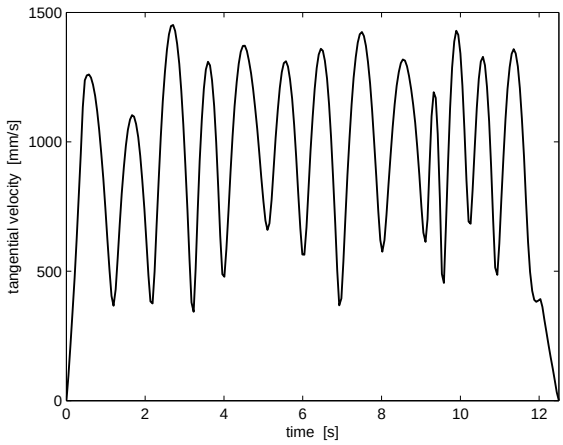

Fig. 9 Original velocity profile (Test I).

A new trajectory has then been designed using the proposed algorithm, by choosing a speed limit, red i.e. v_c as appearing eq. (1) and (4), equal to 1000 mm/s for the end-effector. The maximum allowed value for the joint accelerations are set to 540 deg/s^2 for the first three joints, to 1600 deg/s^2 for joints 4 and 5, and to 3000 rad/s^2 for the last joint.

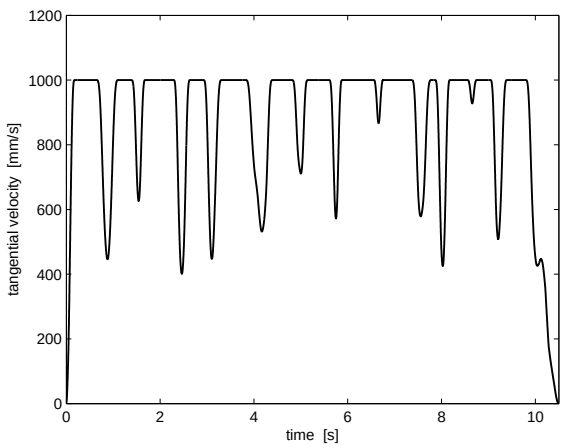

Fig. 10 Velocity profile obtained with the proposed algorithm (Test II).

red Figure 10 shows the end-effector speed after the sequence of filtering operations described in equations (1)-(14), i.e. after the completion of the proposed planning procedure. A direct comparison between Fig. 10 and Fig. 9 shows that the proposed algorithm allows to effectively red and precisely reduce the speed of the end-effector of the robot to the imposed limit, and that a significant time speed-up is achieved as well. The latter is, in this case, equal to a 16% reduction of the

overall total execution time, red since the cycle time is reduced from 12.52 seconds to 10.72 seconds.

As far as the joint accelerations are concerned, they have been estimated, in the absence of a direct measurement, by twice differentiating the data red collected at runtime from the encoders mounted on the robot joints. The estimated joint accelerations are reported in Fig.s 11-13, which show also the prescribed acceleration limits using gray lines.

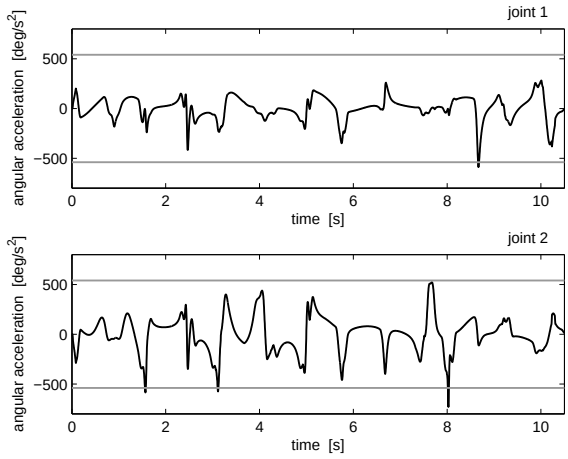

Fig. 11 Test II: angular acceleration for joints 1 and 2.

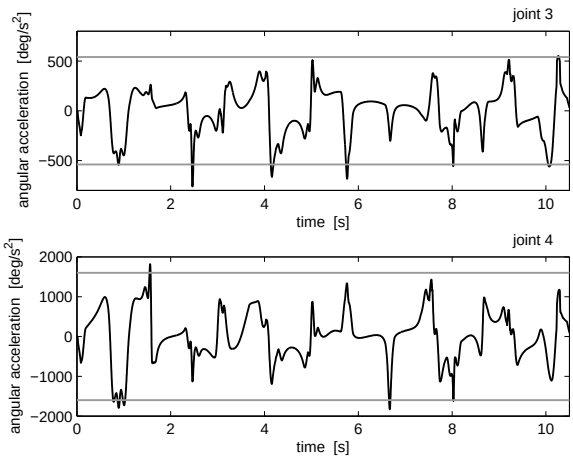

Fig. 12 Test II: angular acceleration for joints 3 and 4.

The joint accelerations are generally confined within the prescribed limits, however some brief violations of the constraints can be detected. red The main source of the inaccuracy of the acceleration limitation is due to the non precise estimation of the extreme valued of joint acceleration by resorting to the average values of joint speed as operated in eq. (7). According to eq. (7)-(9) the

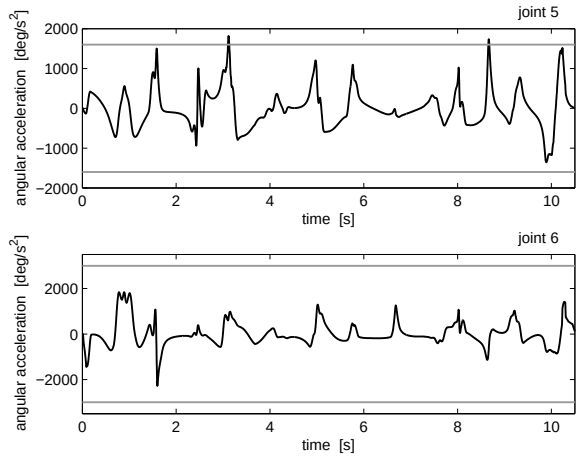

Fig. 13 Test II: angular acceleration for joints 5 and 6.

time scaling that should ensure the joint acceleration limitation is based on an average rather than an actual peak value. In other words, the peak joint acceleration estimation is performed by a first-order approximation which is of limited accuracy if the sampling time Δt_k is not sufficiently small.

This problem can be partially circumvented by a better choice of the downsampling operated prior to the two look-ahead filtering phases, for example by increasing the number of samples in the proximity of high joint acceleration values.

The proposed method shows also an improvement in terms of motor effort, as shown in Fig. 14 and 15, which report the measurements of the torque provided by the first three motors when testing the original and the improved trajectories. The results in Fig. 14 correspond to the same test already analyzed in Fig. 11-13, while Fig. 15 reports the results of a further test, which has been performed by setting the maximum angular acceleration for joints 1, 2 and 3 to 240 deg/s^2. Such lower threshold ensures that the execution time of the original and the modified trajectory are very similar to each other.

The comparison shown in Fig. 14 shows that, despite the sensible reduction in the total execution time, the peak values of the motor torque needed to perform the tasks are very similar among the two test. The results of test III, which are reported in Fig. 15 shows, on the other hand, a significant reduction of the peak value of motor torque when comparing two trajectories with similar total execution time, one obtained with the proprietary method, and the other obtained with the proposed method.

red A further comparison is set by evaluating the RMS value of the motor torques evaluated from the data recorded during the execution of tests I, II and

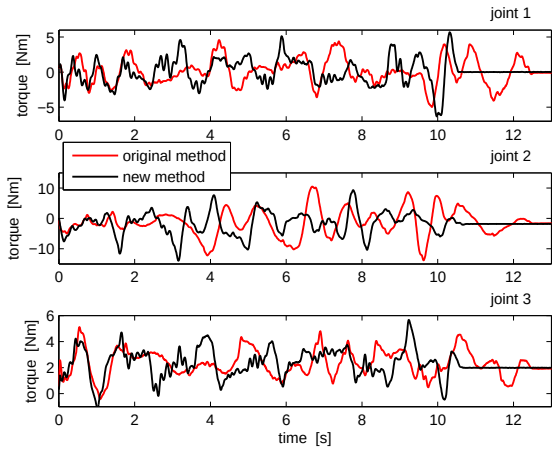

Fig. 14 Motor torques with angular acceleration limit of $540\,deg/s^2$ (Comparison between test I and II).

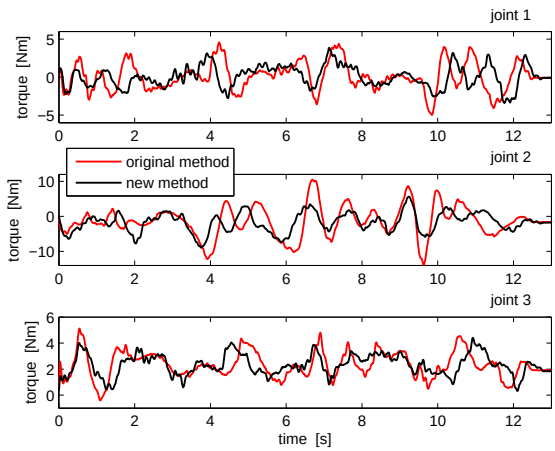

Fig. 15 Motor torques with angular acceleration limit of $240\,deg/s^2$ (Comparison between test I and III).

joint #	test I	test II	test III
1	1.9154	2.0290 (+5.93%)	1.4576 (-23.90%)
2	4.6417	4.1090 (-11.47%)	3.1592 (-31.93%)
3	2.5545	2.6131 (+2.29%)	2.4687 (-3.36%)
4	0.2918	0.2775 (-4.90%)	0.2898 (-0.68%)
5	0.3283	0.3056 (-6.91%)	0.3201 (-2.49%)
6	0.3315	0.3132 (-5.52&)	0.3278 (-0.84%)

Table 1 Experimental values of RMS motor torques, $[Nm]$.

III. Table 1 highlights that the proposed method can lead to significant improvements in terms of motor effort, which are a direct result of the joint acceleration reduction and of the correlated reduced inertial effects. When comparing test I and test III, i.e. two trajectories with the same total execution time, it is evident that a noticeable improvement is detected, especially for the first three joints. Improvements up to 31.93 % can be achieved by using the proposed method, as measured for the second joint. The comparison of the RMS torques between test I and test II shows that the 16 % speed-up brought by the proposed algorithm does result in an increase of the RMS torque equal to just 5.93 % for joint 1 and to 2.29 % for the third joint. The effort required to the other four joints is reduced, with particular reference to the second joint, for which an improvement up to 11.47 % is achieved. Such result is particularly important if it is also considered that the second joint motor is the one that provides the largest effort.

Conclusion

In this paper, a novel path-constrained trajectory planning algorithm for industrial spray painting robots has been introduced.red The main goal of the work is to provide an effective and easy to implement method, whose field of application is widened by the choice of avoiding the use of numerical optimization routines. Starting from a parametrization of the end-effector path in the operative space red that ensures an accurate paint coverage, the velocity profile red of the spray gun is at first defined by the algorithm in order to ensure the limitation of the tangential end-effector velocity red to enhance paint thickness uniformity. In a later stage a sequence of filtering operations are performed in order to bound the joint accelerations and to improve smoothness. The proposed methodology has been experimentally tested on a 6 d.o.f. GR 680 industrial robot by CMA Robotics. red which has been used to assess the performance resulting from the manufacturer's proprietary motion planning method. After that, such results are used to quantify the performance improvement brought by the application of the proposed method. When reproducing the same painting operation, specified by the same end-effector path, the novel method is, unlike the original one, capable of enforcing precise bounds on end-effector speed and approximate bounds on joint accelerations. Two different scenarios are used to evaluate the overall performance of the new method, chosen by setting different values of the joint acceleration bounds. By comparing the two tests it is highlighted that in one case a shorter cycle time is achieved while requesting a similar motor torque effort in comparison with the traditional method, while in the other case a sensibly smaller motor effort is achieved for a comparable cycle time. The limitation of the tangential velocity and of motor joints accelerations ensures that the proposed method can be applied to industrial painting applications for improved performances over simpler techniques, red while the offline nature of the method make it suitable for the integration with several

software applications dedicated to the operation painting robots. The authors are confident that, thanks to the simple implementation and to the interesting improvements obtained, the proposed algorithm can be applied also to other industrial applications.

References

1. Urbano Nunes, António Batista, and João Figueiredo. Spray-painting motion planning and quality analysis in powder coating systems. *IFAC Proceedings Volumes*, 35(1):385–390, 2002.
2. Marcos Ferreira, António Paulo Moreira, and Pedro Neto. A low-cost laser scanning solution for flexible robotic cells: spray coating. *The International Journal of Advanced Manufacturing Technology*, 58(9-12):1031–1041, 2012.
3. N Arunkumar, P Venkatesh, KS Srinivas, and Sanjiv Kaushik. Response surface modeling and optimization of single axis automatic application of automotive polyurethane coatings on plastic components. *The International Journal of Advanced Manufacturing Technology*, 63(9):1065–1072, 2012.
4. Prasad N Atkar, Aaron Greenfield, David C Conner, Howie Choset, and Alfred A Rizzi. Uniform coverage of automotive surface patches. *The International Journal of Robotics Research*, 24(11):883–898, 2005.
5. Scott Baldwin. Robotic paint automation: The pros and cons of using robots in your paint finishing system. *Metal Finishing*, 108(11-12):126–129, 2010.
6. Jiangyao Chen, Ranran Liu, Yanpeng Gao, Guiying Li, and Taicheng An. Preferential purification of oxygenated volatile organic compounds than monoaromatics emitted from paint spray booth and risk attenuation by the integrated decontamination technique. *Journal of Cleaner Production*, 148:268–275, 2017.
7. Yan Chen, Wenzhuo Chen, Bo Li, , Gang Zhang, and Weiming Zhang. Paint thickness simulation for painting robot trajectory planning: a review. *Industrial Robot: An International Journal*, 44(5):629–638, 2017.
8. Heping Chen, Thomas Fuhlbrigge, and Xiongzi Li. Automated industrial robot path planning for spray painting process: a review. In *Automation Science and Engineering, 2008. CASE 2008. IEEE International Conference on*, pages 522–527. IEEE, 2008.
9. S Sugita, T Itaya, and Y Takeuchi. Development of robot teaching support devices to automate deburring and finishing works in casting. *The International Journal of Advanced Manufacturing Technology*, 23(3-4):183–189, 2004.
10. Tomas Lozano-Perez. Robot programming. *Proceedings of the IEEE*, 71(7):821–841, 1983.
11. Geoffrey Biggs and Bruce MacDonald. A survey of robot programming systems. In *Proceedings of the Australasian conference on robotics and automation*, pages 1–3, 2003.
12. Sahir Arıkan and Tuna Balkan. Process modeling, simulation, and paint thickness measurement for robotic spray painting. *Journal of Field Robotics*, 17(9):479–494, 2000.
13. Lorenzo Scalera, Enrico Mazzon, Paolo Gallina, and Alessandro Gasparetto. Airbrush robotic painting system: Experimental validation of a colour spray model. In *International Conference on Robotics in Alpe-Adria Danube Region*, pages 549–556. Springer, 2017.
14. Suk-Hwan Suh, In-Kee Woo, and Sung-Kee Noh. Automatic trajectory planning system (ATPS) for spray painting robots. *Journal of manufacturing Systems*, 10(5):396–406, 1991.
15. S-H Suh, I-K Woo, and S-K Noh. Development of an automatic trajectory planning system (ATPS) for spray painting robots. In *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on*, pages 1948–1955. IEEE, 1991.
16. Naoki Asakawa and Yoshimi Takeuchi. Teachingless spray-painting of sculptured surface by an industrial robot. In *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on*, volume 3, pages 1875–1879. IEEE, 1997.
17. Heping Chen, Weihua Sheng, Ning Xi, Mumin Song, and Yifan Chen. CAD-based automated robot trajectory planning for spray painting of free-form surfaces. *Industrial Robot: An International Journal*, 29(5):426–433, 2002.
18. Xiongzi Li, Oeyvind A Landsnes, Heping Chen, MV Sudarshan, Thomas A Fuhlbrigge, and Mary-Ann Rege. Automatic trajectory generation for robotic painting application. In *Robotics (ISR), 2010 41st International Symposium on and 2010 6th German Conference on Robotics (ROBOTIK)*, pages 1–6. VDE, 2010.
19. A Gasparetto, R Vidoni, E Saccavini, and D Pillan. Optimal path planning for painting robots. In *ASME 2010 10th Biennial Conference on Engineering Systems Design and Analysis*, pages 601–608. American Society of Mechanical Engineers, 2010.
20. Heping Chen, Thomas Fuhlbrigge, and Xiongzi Li. A review of cad-based robot path planning for spray painting. *Industrial Robot: An International Journal*, 36(1):45–50, 2009.
21. Georg Biegelbauer, Andreas Pichler, Markus Vincze, Christian L Nielsen, Henrik John Ander-

sen, and Kurt Häusler. The inverse approach of flexpaint [robotic spray painting]. *IEEE Robotics & Automation Magazine*, 12(3):24–34, 2005.

22. Wei Chen and Dean Zhao. Path planning for spray painting robot of workpiece surfaces. *Mathematical Problems in Engineering*, 2013, 2013.
23. Enric Galceran and Marc Carreras. A survey on coverage path planning for robotics. *Robotics and Autonomous Systems*, 61(12):1258–1276, 2013.
24. Howie Choset. Coverage for robotics–a survey of recent results. *Annals of mathematics and artificial intelligence*, 31(1):113–126, 2001.
25. Wesley H Huang. Optimal line-sweep-based decompositions for coverage algorithms. In *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, volume 1, pages 27–32. IEEE, 2001.
26. Prasad N Atkar, Howie Choset, and Alfred A Rizzi. Towards optimal coverage of 2-dimensional surfaces embedded in ir/sup 3: choice of start curve. In *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, volume 4, pages 3581–3587. IEEE, 2003.
27. Ramanujam Ramabhadran and John K Antonio. Fast solution techniques for a class of optimal trajectory planning problems with applications to automated spray coating. *IEEE Transactions on Robotics and Automation*, 13(4):519–530, 1997.
28. Wei Chen, Yang Tang, and Qiang Zhao. A novel trajectory planning scheme for spray painting robot with bézier curves. In *Control and Decision Conference (CCDC), 2016 Chinese*, pages 6746–6750. IEEE, 2016.
29. Kevin M Lynch and Frank C Park. *Modern Robotics: Mechanics, Planning, and Control*. Cambridge University Press, 2017.
30. Mayur V Andulkar and Shital S Chiddarwar. Automated cad based trajectory for spray painting robot: Variable velocity approach. In *ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pages V05CT08A007–V05CT08A007. American Society of Mechanical Engineers, 2015.
31. James E Bobrow, Steven Dubowsky, and JS Gibson. Time-optimal control of robotic manipulators along specified paths. *The international journal of robotics research*, 4(3):3–17, 1985.
32. Zvi Shiller. Time-energy optimal control of articulated systems with geometric path constraints. *Journal of dynamic systems, measurement, and control*, 118(1):139–143, 1996.
33. Luigi Biagiotti and Claudio Melchiorri. *Trajectory planning for automatic machines and robots*. Springer Science & Business Media, 2008.