2018 Latin American Robotic Symposium, 2018 Brazilian Symposium on Robotics (SBR) and 2018 Workshop on Robotics in Education (WRE)

A Novel Tuning Method for PD Control of Robotic Manipulators Based on Minimum Jerk Principle

Phelipe W. Oliveira, Guilherme A. Barreto and George T. P. Thé

Abstract— **In this paper we introduce a novel technique for optimal tuning of PD controllers engaged in tracking minimum-jerk (MJ) trajectories. The proposed approach is an attempt to bridge the gap between the MJ principle for trajectory planning, which is based solely on the robot's kinematics, and the optimal estimation of the gains of the joint controllers, which depends on the robot dynamics. For this purpose we define an objective function that combines kinematic and dynamic-based performance indices and which is minimized via a genetic algorithm that searches for optimal gains for the joint controllers. The proposed approach is shown to perform consistently better than the standard PD control for tracking MJ trajectories.**

I. INTRODUCTION

Since the early 1990's, the principle of minimum-jerk (MJ) has been widely used for trajectory planning purposes in robotics [1], [2], [3], [4], [5], [6], [7]. As the jerk is the time-derivative of acceleration, MJ trajectories are desirable for limiting excessive wear on the robot and the excitation of resonances so that the robot life-span is extended [5]. Moreover, MJ principle is also important for neuroscience studies, since it has been hypothesized that the movements of human joints tend to follow MJ paths [8], [9], a feature that has been used e.g. for the purpose of motor rehabilitation [10].

It should be noted, however, that the generation of MJ trajectories are based solely on the robot's kinematics. Despite this seems to be a rather convenient property from the point of view of trajectory planning, for the optimal design of the controllers responsible for the effective tracking of the desired trajectory the robot's dynamics must be taken into account. Indeed, it has been pointed out in [6] and [7] that the jerk of the desired trajectory adversely affects the performance of the tracking control algorithms for robotic manipulators. This happens because PID-like controllers [11], [12] at the robot's joints, while doing their best in tracking the planned trajectories as close as possible, cannot guarantee that this goal will be achieved since the robot's dynamics is not taken into account in the formulation of the MJ principle.

Motivated by this mismatch between MJ-based trajectory planning and the effective tracking of the desired trajectories, we introduce a novel technique for optimal tuning of PD-like controllers engaged in tracking minimum-jerk (MJ) trajectories. For this purpose we define an objective function in which we aggregate kinematic and dynamic-based performance indices and which is minimized by means of a genetic algorithm whose goal is to find optimal gains for the joint controllers. A comprehensive performance evaluation of the proposed approach reveals that it consistently outperforms the standard PD-like control for tracking MJ trajectories.

The remainder of the paper is organized as follows. The fundamentals of the MJ principle for trajectory planning is presented in Section II, while the robot dynamics and control are discussed in Section III. In Section IV a new approach is introduced for tuning the gains of the joint controllers for effective tracking of the planned MJ trajectories. In Section V we finally report the results of our computer experiments and discuss them. The paper is concluded in Section VI.

II. BASICS OF THE MINIMUM JERK PRINCIPLE

Let us assume a single joint. If the angular position of this joint is defined by a function of time $q(t)$, then the jerk J of the system, i.e. the rate in which the acceleration varies, is defined as the third derivative of the position:

$$J(t) = \frac{d^3 q(t)}{dt^3}. \tag{1}$$

The jerk plays an important role in robotic system because it is widely known since the work of Kyriakopoulos and Saridis [7] that joint position errors increase in the presence of jerky movements. Hence, to increase the accuracy of position control in robotic systems, an interesting approach is to minimize the jerk. In this regard, the objective of the minimum jerk (MJ) principle is to find a function q that minimizes the integral of the squared jerk over time:

$$L_J(q) = \frac{1}{2}\int_0^T J^2(t)dt = \frac{1}{2}\int_0^T \left(\frac{d^3 q(t)}{dt^3}\right)^2 dt. \tag{2}$$

Thus, by choosing q as a 5th-order polynomial we have the guarantee that the trajectory will be of minimum jerk. This choice leads to the following equation for the joint position:

$$q(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5, \tag{3}$$

where $\{a_k\}_{k=0}^5$ corresponds to the set of coefficients that must be estimated to satisfy the MJ principle. By differentiating Eq. 3 with respect to time, we obtain the corresponding expressions for the velocity ($\dot{q}$), acceleration ($\ddot{q}$) and jerk

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001. This work was also partially supported by the project INCT (National Institute of Science and Technology) under the grant CNPq (Brazilian National Research Council) 465755/2014-3, FAPESP (São Paulo Research Foundation) 2014/50851-0.

The authors are with the Graduate Program in Teleinformatics Engineering, Federal University of Ceará, Campus of Pici, Fortaleza, Ceará, Brazil. Emails: phewesley@gmail.com, gbarreto@ufc.br and george.the@ufc.br

978-1-5386-7761-2/18/$31.00 ©2018 IEEE
DOI 10.1109/LARS/SBR/WRE.2018.00077

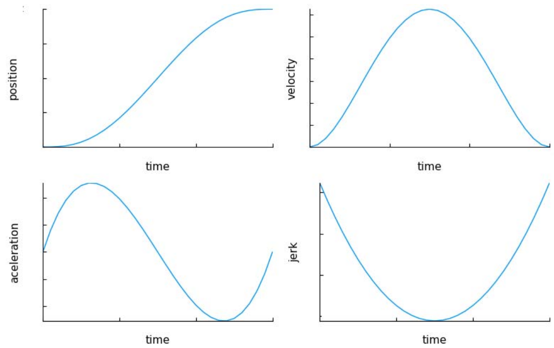


Fig. 1. Minimum jerk trajectories obtained when the function $q(t)$ is defined as in Eq. (3).

(J), which are given by

$$\begin{aligned} \dot{q}(t) &= a_1 + 2a_2t + 3a_3t^2 + 4a_4t^3 + 5a_5t^4, & (4)\\ \ddot{q}(t) &= 2a_2 + 6a_3t + 12a_4t^2 + 20a_5t^3, & (5)\\ J(t) &= 6a_3 + 24a_4t + 60a_5t^2. & (6) \end{aligned}$$

Typical curves for $q(t)$ and its derivatives for certain values of the coefficients are shown in Figure 1. It should be noted that for obtaining these equations we assumed that trajectory planning will be carried out in the joint space of the robot. Similar procedure can be used to generate trajectories in the cartesian domain.

To compute the optimal coefficients according to MJ principle, it is necessary to specify the positions, velocities and accelerations at the initial instant t_0 (q_0, $\dot{q}_0$, $\ddot{q}_0$) and at the final instant t_f (q_f, $\dot{q}_f$, $\ddot{q}_f$). Thus, let $\mathbf{a}$ be the vector of coefficients and $\mathbf{s}$ be the vector of initial and final states of the system, defined respectively as

$$\mathbf{a} = [a_0 \;\; a_1 \;\; a_2 \;\; a_3 \;\; a_4 \;\; a_5]^T, \tag{7}$$

and

$$\mathbf{s} = [q_0 \;\; \dot{q}_0 \;\; \ddot{q}_0 \;\; q_f \;\; \dot{q}_f \;\; \ddot{q}_f]^T, \tag{8}$$

where the superscript T denotes the transpose of a vector/matrix. Furthermore, let $\mathbf{D}$ be a matrix whose entries depend on the initial and final time instants of the movement:

$$\mathbf{D} = \begin{bmatrix} 1 & t_0 & (t_0)^2 & (t_0)^3 & (t_0)^4 & (t_0)^5 \\ 0 & 1 & 2t_0 & 3(t_0)^2 & 4(t_0)^3 & 5(t_0)^4 \\ 0 & 0 & 2 & 6t_0 & 12(t_0)^2 & 20(t_0)^3 \\ 1 & t_f & (t_f)^2 & (t_f)^3 & (t_f)^4 & (t_f)^5 \\ 0 & 1 & 2t_f & 3(t_f)^2 & 4(t_f)^3 & 5(t_f)^4 \\ 0 & 0 & 2 & 6t_f & 12(t_f)^2 & 20(t_f)^3 \end{bmatrix}; \tag{9}$$

Finally, if we formulate the problem as a linear system $\mathbf{s} = \mathbf{D}\mathbf{a}$, then the coefficients are computed as

$$\mathbf{a} = \mathbf{D}^{-1}\mathbf{s}, \tag{10}$$

assuming that the matrix $\mathbf{D}$ is invertible. It is worth mentioning that the procedure for computing the coefficients $\{a_k\}_{k=0}^5$ and the corresponding functions should be repeated for each joint of the robot. Thus, for a robot with n DOF, we will have to compute $6 \times n$ coefficients.

Remark 1: It is worth recalling that the MJ principle, as a trajectory planning method, is based solely on the robot's kinematics. As a consequence, there is no guarantee at all that the joint controllers will accurately follow the desired acceleration and jerk trajectories since the robot's dynamics is not taken into account in the formulation of the MJ principle.

III. Robot Dynamics and Control

Robot dynamics is concerned with the relationship between the forces and torques acting on a robotic structure and the accelerations they produce on it. In this paper, a rigid-body robotic manipulator is modeled as a kinematic chain with n degrees of freedom, whose dynamics is given by [13], [14], [15]

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \mathbf{Q}, \tag{11}$$

where $\mathbf{q}$ is the vector of joint positions, $\dot{\mathbf{q}}$ is the vector of joint velocities, $\ddot{\mathbf{q}}$ is the vector of accelerations, $\mathbf{M}(\mathbf{q})$ denotes the joint-space inertia matrix, and it is an $n \times n$ symmetric, positive-definite matrix. The term $\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}})$ accounts for the effects of Coriolis and centripetal forces, while $\mathbf{g}(\mathbf{q})$ is the term that accounts for the effects of gravity. The term $\mathbf{Q}$ represents the action of all external forces acting on the system, such as friction and torques due to joint movement.

A. Low-Level Control of Manipulators

Proportional-integral-derivative (PID) controllers and variants, such as the proportional-derivative controller (PD), are the most commonly used control strategies in industry [11].

The PD controller is a variant of the PID controller commonly used in Robotics [15], [12], whose control law can be written as

$$u = K_p e + K_d \dot{e}, \tag{12}$$

where K_p and K_d are, respectively, the controller's proportional and derivative gains, e is the error signal and $\dot{e}$ is its derivative. The term u is the control action. This equation has to be modified in order to adapt to the dynamics of a robotic manipulator.

Since the robotic system of interest can be understood as a multi-input/multi-output (MIMO) system, the control law in (12) must be rewritten in vector-matrix format as

$$\mathbf{K}_p\mathbf{e} + \mathbf{K}_d\dot{\mathbf{e}} = \tau, \tag{13}$$

where the gain terms $\mathbf{K}_p$ and $\mathbf{K}_d$ are now positive-definite diagonal matrices, while the error terms $\mathbf{e}$, $\dot{\mathbf{e}}$ and τ are vectors. It should be noted that the variable associated with the control action has been replaced by τ (i.e. the torque vector), since it is the torque that effectively generates the motion of the joint.

A crucial issue in the design of the PD controller is the choice of the gain matrices, which is usually performed by means of the heuristic method by Ziegler & Nichols [16]. While this widely used method provides gain values which are good enough for simple end-effector positioning tasks, this is not the case for the more complex robotic tasks we are interested in.

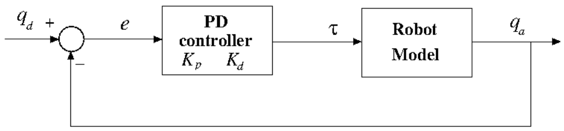


Fig. 2. Standard PD joint control scheme that must be executed for each chromosome of the population at a given generation.

The MJ principle for trajectory planning generates trajectories for position, velocity, acceleration and jerk. In our studies, we have observed that by using the standard Ziegler-Nichols method for tuning the PD controller, there was absolutely no guarantee that the combined motions of the joints would effectively follow (i.e. track) the generated trajectories, especially the acceleration and jerk trajectories.

In order to reduce these effects, we developed a novel approach to search for the gains of the PD controllers of each joint so that the minimum-jerk requirements could be realized at the controller level. The proposed approach is based on a metaheuristic optimization method and is described in detail in the next section.

IV. The Proposed Approach

Aiming at reducing undesirable effects due to the robot dynamics while tracking MJ trajectories the following objective function is proposed:

$$L(K_p, K_d) = \alpha \int_{t_0}^{t_f} e dt + \beta \int_{t_0}^{t_f} J dt + \gamma \int_{t_0}^{t_f} \tau dt + \int_{t_0}^{t_f} \Delta\tau dt, \quad (14)$$

where we have included the jerk (J) and the torque (τ) of the system in the equation so that the controller, besides minimizing the error (e), also causes reduction of jerk and reduction of torque in the system, thus making wear on the joints of the kinematic chain smaller. Furthermore, a term involving torque change ($\Delta\tau$) is included to ensure that there will be no abrupt variations of the torque values. The constants α, β and γ are used to normalize the terms to roughly the same order of magnitude, and can be chosen based on the reading time of the position in the controller.

Due to the complexity of the cost function shown in (14), we decided to follow a metaheuristic based optimization approach and chose a simple genetic algorithm (GA) [17] for this purpose. The optimization process is executed for the PD controller of each joint independently. Hence, for the j-th joint of the robot the i-th chromosome in a population is defined as

$$\mathbf{x}(i|j) = [K_p(i|j) \;\; K_d(i|j)], \quad (15)$$

where $K_p, K_d \in \mathbb{R}$, with $0 < K_p < 10^4$ and $0 < K_d < 10^3$. Optimization process is executed sequentially, starting at the end effector joint and ending at the joint closest to the manipulator base. The PD controller tuning technique based on the minimization of the proposed objective function in (14) will be henceforth be referred to as the *minimum jerk optimal PD control* (MJ-OPD).

The design parameters of the GA used in this paper are the following: (*i*) size of the population ($M = 20$); (*ii*) number of generations ($N = 15$); (*iii*) selection method: roulette; (*iv*) crossover type: intermediate recombination; (*v*) mutation type: random Gaussian number with standard deviation $\sigma_{GA} = 0.1$; (*vi*) crossover probability ($p_c = 80\%$); (*vii*) mutation probability ($p_m = 10\%$), and (*viii*) elitism: yes. In case of violation of the search space constraints, the gain value is clipped to the limit of the corresponding interval. The search intervals for the gains were selected based on the values obtained for them by the Ziegler-Nichols (ZN) method. The choice of GA parameters required some initial experimentation, but nothing beyond the expectation for this type of optimization method.

Fig. 3. Two-dof robot introduced in [18] whose model was used in the computer experiments reported in this paper.

Remark 2: Since the search for the optimal gains is executed independently, we have to execute two GAs, one for each joint. We decided for this approach because we observed in the experiments that it converged much faster than a single GA with the i-th chromosome being defined as

$$\mathbf{x}(i) = [K_p(i|1) \;\; K_d(i|1) \;\; K_p(i|2) \;\; K_d(i|2)], \quad (16)$$

where $K_p(i|1)$ and $K_d(i|1)$ are the gains for the Joint 1 controller, while $K_p(i|2)$ and $K_d(i|2)$ are the gains for the Joint 2 controller.

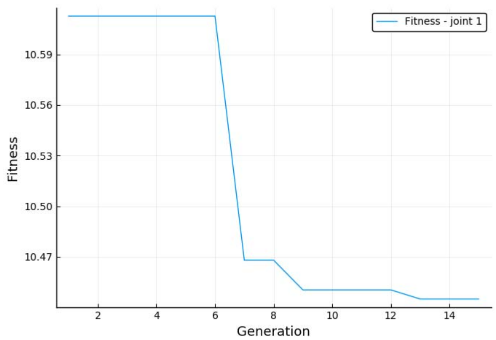
Remark 3: The objective function defined in Eq. (14) is used as fitness function for the GAs. For each chromosome $\mathbf{x}(i|j)$, the associated values of the gains (i.e. $K_p(i|j)$ and $K_d(i|j)$) are used in a standard closed-loop joint control scheme (see Fig. 2 to position the arm at the desired joint positions (q_d), which corresponds to points of the planned trajectory sampled at certain time instants, t_k, $k = 0, \ldots, f$. The actual joint position is denoted as q_a.

V. Results and Discussion

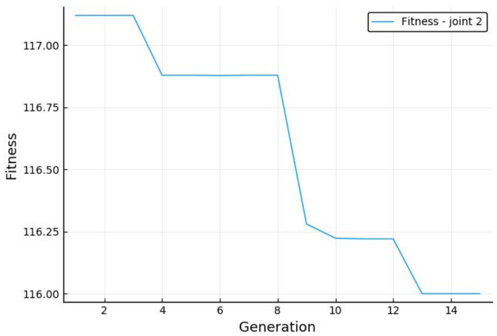
In order to compare the performance of the MJ-OPD controller with that of the standard PD controller, simulations were performed in the Julia language[1]. The robot model used in the computer experiments was obtained from [18] and consists of a 2-dof planar robot whose movements are constrained to the vertical plane (see Fig. 3), so that the effects of gravity must be taken into account.

The constant parameters of the objective function in Eq. (14) were set to $\alpha = 10$, $\beta = 0.01$ and $\gamma = 0.1$. For

[1] https://julialang.org/

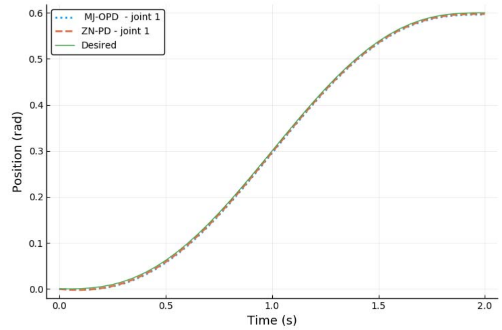


(a) GA for Joint 1

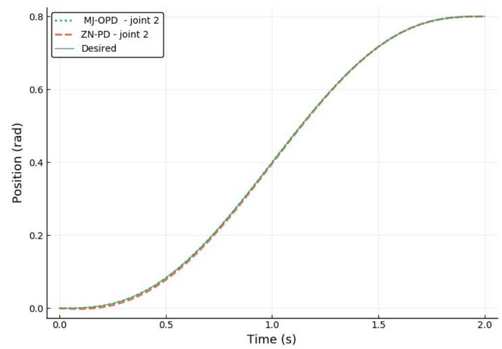


(b) GA for Joint 2

Fig. 4. Best fitness values at each generation for the two GAs.



(a) Joint 1



(b) Joint 2

Fig. 5. Position results for joints 1 and 2.

the sake of comparison, in addition to estimating the gains of the MJ-OPD controller, we also tuned the gains of the PD controller using the classical Ziegler-Nichols (ZN) method. The values obtained by the two approaches are presented in Table I. The evolution through generations of the fitness of the best individual in the population for the two GAs (one for each joint) are shown in Figs. 4(a) and 4(b).

TABLE I
ESTIMATED GAINS OF THE JOINT CONTROLLERS.

Gains		**ZN-PD**	**MJ-OPD**
K_p	Joint 1	8550.00	5611.43
	Joint 2	160.00	1495.46
K_d	Joint 1	415.00	969.85
	Joint 2	15.00	424.93

The performances of the classical ZN-PD and MJ-OPD controllers introduced earlier are compared on a trajectory tracking task. The duration of each trajectory is set to 2 seconds ($t_0 = 0$ and $t_f = 2$). Trajectory planning and generation based on minimum jerk rely not only on specification for position, but it takes into account velocity and acceleration profiles, as well as the jerk itself. Initial results are plotted in Figure 5, revealing that both controllers perform well for the basic positioning task.

Similarly, both controllers performed well in tracking the planned velocity profile, as can be seen in Figure 6 for joints 1 and 2, respectively. The accuracy in tracking position and velocity trajectories was already expected for both controllers because, although they present different gains, the PD control law explicitly takes into account position and velocity errors.

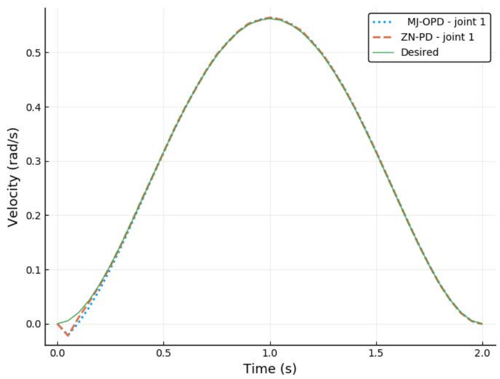
However, in what concerns the acceleration profile, the performance of the controllers starts to differ considerably. From Figure 7, which report the desired and the actually executed acceleration trajectories for joints 1 and 2, one can easily see that the proposed MJ-OPD controller performed considerably better than the classical ZN-based PD controller, which suffers from severe limitations leading to oscillatory behavior along the executed trajectory. As a consequence the MJ-OPD controller was able to track the desired trajectory with a squared error lower than the other controller, as reported in Table II.

TABLE II
ACCELERATION SQUARED ERRORS.

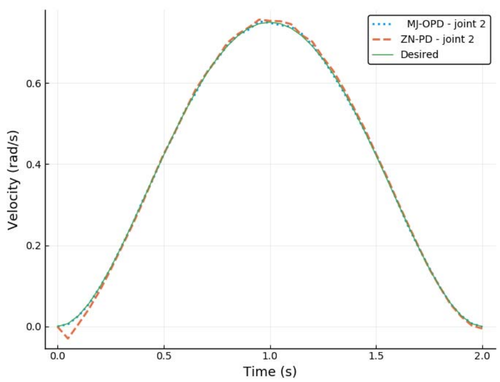
Squared Error $(rad/s^2)^2$	**ZN-PD**	**MJ-OPD**
Joint 1	0.0136	**0.0092**
Joint 2	0.0247	**0.0091**

Tracking the trajectory profile of the jerk was challenging for classical ZN-PD controller once again. In Figure 8 one can observe abrupt oscillations (in red dashed lines) resulting from the action of the classical PD controller. The MJ-OPD controller also faced some minor oscillations, specially for Joint 2, but in a much lower degree in comparison to the classical PD controller.

In order to assess both controllers quantitatively, in Table



(a) Joint 1



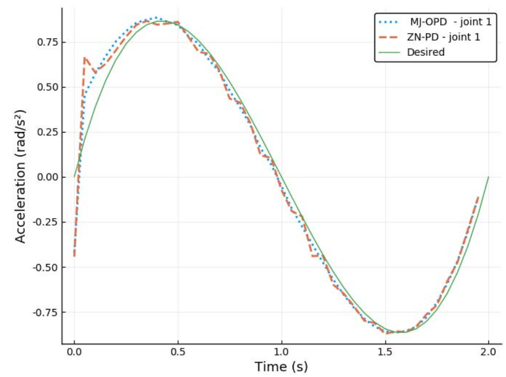
(b) Joint 2

Fig. 6. Velocity results for joints 1 and 2.

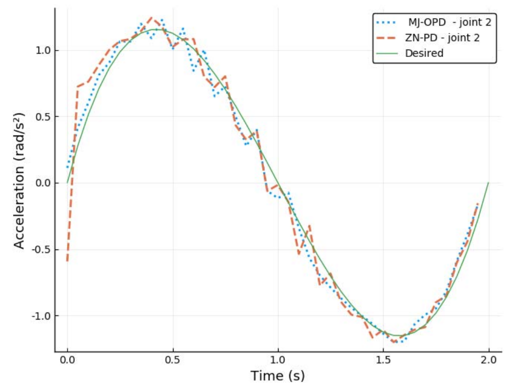
TABLE III

COMPARISON OF THE PRODUCED JERK VALUES.

Jerk	(rad/s^3)	**ZN-PD**	**MJ-OPD**
Maximum	Joint 1	22.19	**17.51**
	Joint 2	26.34	**9.19**
Total	Joint 1	80.55	**76.21**
	Joint 2	131.14	**118.16**



(a) Joint 1



(b) Joint 2

Fig. 7. Acceleration results for joints 1 and 2.

TABLE IV

SQUARED ERROR VALUES FOR THE JERK TRAJECTORIES.

Squared Error	$(rad/s^3)^2$	**ZN-PD**	**MJ-OPD**
	Joint 1	9.9152	**4.63**
	Joint 2	19.2416	**6.68**

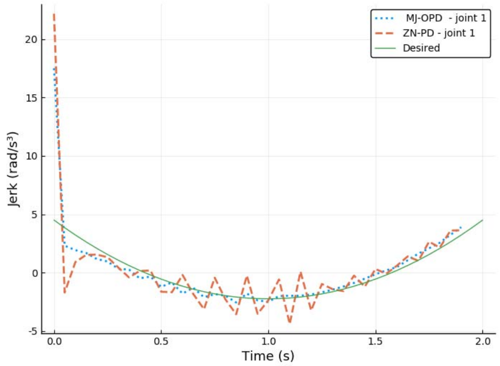
III the peaks and mean values of produced jerk are reported. A careful analysis of this table reveals that the MJ-OPD controller was able to limit significantly the peaks of the jerks in Joints 1 and 2. Significant reduction in the total jerk was also observed. In Table IV we report the resulting squared errors, from which one can infer that the proposed MJ-OPD controller was able to track the desired trajectories consistently better than the classical PD controller.

Finally, a comparative evaluation of the produced torque is carried out for assessing the demanded energy for executing the task. In Figure 9 we report the developed torque along the trajectory tracking task for each joint. An inspection of these figures reveals that the controllers behave similarly, a feature somehow expected because the controller structure is the same, differing only in the tuning method. For the sake of completeness, in Table V we report the maximum (i.e. peak) and total values for the developed torques at both joints.

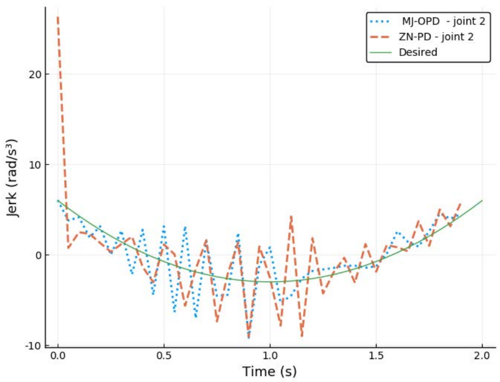
VI. CONCLUSIONS AND FURTHER WORK

In this work the problem of the choice of gains for controllers of robotic manipulators based on trajectories generated by the principle of minimum-jerk was addressed. Simulations were performed to present a comparison between the classical gain selection method by Ziegler-Nichols and the proposed method for optimization based on jerk and torque. The proposed method presents good results in the reduction of the jerk in the system, reducing the difference between the actual values and the planned values in the generation of trajectory, as well as does not present sudden oscillations as it was possible to observe in the classic PD controller in the simulations. The proposed method is simple to implement since the only requirement for its realization is the knowledge of the dynamics of the system and in kinematic chains this can be easily obtained from the knowledge of the lengths and weights of each link.

Currently, we are working in the development of an alternative minimum-jerk-based cost function for optimal tuning



(a) Joint 1



(b) Joint 2

Fig. 8. Jerk results for joints 1 and 2.

TABLE V

COMPARISON OF THE PRODUCED TORQUE VALUES.

Torque	*(Nm)*	**ZN-PD**	**MJ-OPD**
Maximum	Joint 1	29.37	29.38
	Joint 2	0.83	0.82
Total	Joint 1	1022.86	1023.11
	Joint 2	16.98	16.95

of fractional-order PID controllers for robotic manipulators.

REFERENCES

[1] D. Kaserer, H. Gattringer, and A. Mueller, "On-line robot-object synchronization with geometric constraints and limits on velocity, acceleration and jerk," *IEEE Robotics and Automation Letters*, pp. 1–7, 2018.

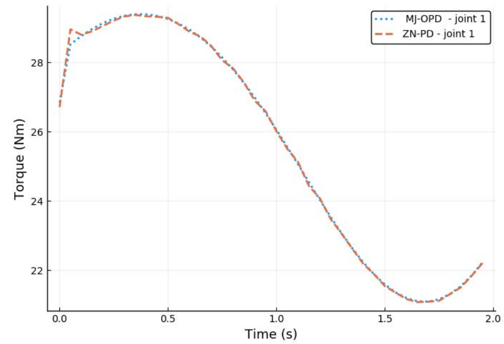
[2] S. L. an J. Zhao, L. Jiang, and H. Liu, "Solving the time-jerk optimal trajectory planning problem of a robot using augmented Lagrange constrained particle swarm optimization," *Mathematical Problems in Engineering*, vol. 2017, no. ID-1921479, pp. 1–10, 2017.

[3] W. Suleiman, "On inverse kinematics with inequality constraints: new insights into minimum jerk trajectory generation," *Advanced Robotics*, vol. 30, no. 17–18, pp. 1164–1172, 2016.

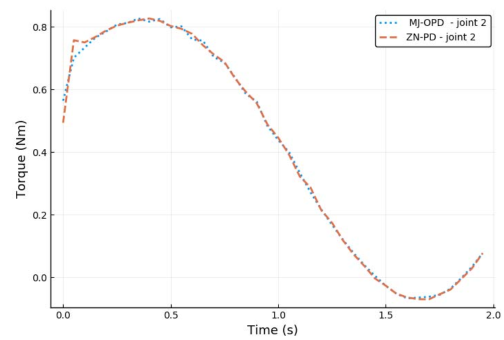
[4] A. Aloulou and O. Boubaker, "Minimum jerk-based control for a three dimensional bipedal robot," in *International Conference on Intelligent Robotics and Applications (ICIRA'2011)*, 2011, pp. 251–262.

[5] A. Piazzi and A. Visioli, "Global minimum-jerk trajectory planning of robot manipulators," *IEEE Transactions on Industrial Electronics*, vol. 47, no. 1, pp. 140–149, 2000.

[6] K. J. Kyriakopoulos and G. Saridis, "Minimum jerk for trajectory planning and control," *Robotica*, vol. 12, no. 2, pp. 109–113, 1994.



(a) Joint 1



(b) Joint 2

Fig. 9. Torque results for joints 1 and 2.

[7] K. J. Kyriakopoulos and G. N. Saridis, "Minimum jerk trajectory planning for robotic manipulators," in *Cooperative Intelligent Robotics in Space*, vol. 1387. International Society for Optics and Photonics, 1991, pp. 159–165.

[8] T. Flash, Y. Meirovitch, and A. Barliya, "Models of human movement: Trajectory planning and inverse kinematics studies," *Robotics and Autonomous Systems*, vol. 61, no. 4, pp. 330–339, 2013.

[9] N. Hogan, "An organizing principle for a class of voluntary movements," *Journal of Neuroscience*, vol. 4, no. 11, pp. 2745–2754, 1984.

[10] B. Rohrer, S. Fasoli, H. I. Krebs, R. Hughes, B. Volpe, W. R. Frontera, J. Stein, and N. Hogan, "Movement smoothness changes during stroke recovery," *Journal of Neuroscience*, vol. 22, no. 18, pp. 8297–8304, 2002.

[11] A. Visioli, "Research trends for PID controllers," *Acta Polytechnica*, vol. 52, no. 5, pp. 133–154, 2012.

[12] R. Lozano, A. Valera, P. Albertos, S.Arimoto, and T. Nakayama, "PD control of robot manipulators with joint flexibility, actuators dynamics and friction," *Automatica*, vol. 35, no. 10, pp. 1697–1700, 1999.

[13] R. Kelly, V. S. Davila, and J. A. L. Perez, *Control of robot manipulators in joint space*. Springer Science & Business Media, 2006.

[14] M. A. Llama, R. Kelly, and V. Santibañez, "A stable motion control system for manipulators via fuzzy self-tuning," *Fuzzy sets and systems*, vol. 124, no. 2, pp. 133–154, 2001.

[15] R. Kelly and R. Carelli, "A class of nonlinear PD-type controllers for robot manipulators," *Journal of Robotic Systems*, vol. 13, no. 12, pp. 793–802, 1996.

[16] J. G. Ziegler and N. B. Nichols, "Optimum settings for automatic controllers," *Transactions of the ASME*, vol. 64, pp. 759–768, 1942.

[17] J. Holland and D. Goldberg, "Genetic algorithms in search, optimization and machine learning," *Massachusetts: Addison-Wesley*, 1989.

[18] J. L. Meza, V. Santibáñez, R. Soto, and M. A. Llama, "Fuzzy self-tuning PID semiglobal regulator for robot manipulators," *IEEE Transactions on industrial electronics*, vol. 59, no. 6, pp. 2709–2717, 2012.