Online Optimization and Trajectory Planning for Energy Efficiency in a Robotic Linear Axis

Online optimization and trajectory planning for energy efficiency in a robotic linear axis

Giuliano Fabris, Lorenzo Scalera, and Alessandro Gasparetto

University of Udine, 33100 Udine, Italy
{giuliano.fabris, lorenzo.scalera, alessandro.gasparetto}@uniud.it

Abstract. Energy efficiency is a challenging and relevant research field in modern manufacturing industries, where robotic systems play an essential role in the automation of several industrial operations. In this paper we present the dynamic modelling of a robotic linear axis and a method for the online optimization of point-to-point motions of the system. Exploiting the verified dynamic model of the system, the trajectories are optimized online for two different cases, namely minimum-time and minimum-time-energy, allowing the execution of sequential optimal motions without idle times. The experimental results shows the feasibility and the energy saving capabilities of the proposed optimization approach, allowing to flexibly select the best trade-off between execution time and energy consumption.

Keywords: energy efficiency, dynamic modelling, linear axis, robotics, trajectory planning

1 Introduction

In recent decades, the increasing demand for automation and robotics in various industries has evidenced the importance of optimizing energy efficiency in robotic and mechatronic systems. As these systems become more prevalent in diverse manufacturing and robotic applications, there is a growing need of advanced modelling techniques to enhance their energy performance.

Several approaches have been explored to optimize energy consumption in robotic systems. These include optimizing the parameters that define robot trajectories [1,2], or strategically positioning the robot task within the workspace to minimize unnecessary movements and reduce energy expenditure [3,4]. Moreover, integrating elements such as elastic components or regenerative drivers into the system design can recycle energy and further enhance efficiency [5].

Among the variety of robotic and mechatronic systems, those with one degree of freedom (DOF) are extensively employed across various applications in production, packaging, and logistics facilities. Examples are elevators, Cartesian axis and CNC machines [6]. In the literature, several studies that address the energy efficiency in 1-DOF systems can be found. For instance, Wang et al. [7] present a real-time algorithm capable of planning energy-optimal trajectories for a servomotor system subjected to acceleration and speed constraints in point-to-point motions. The algorithm divides the trajectory in different sections (e.g., speed- and acceleration-constrained phases) and optimizes the execution times of the various sections, fixing the total time. Donà et al. propose in [8] a method for real-time designing minimum-energy trajectories for servo-actuated systems that exploits the knowledge of the shape of the optimal trajectory, optimizing the trajectory parameters.

The authors in [9] present a simple offline framework based on basic kinematic relationships for generating optimal motion trajectories for different performance objectives, including energy efficiency. Carabin et al. develop an analytical method to minimize the energy consumption of mechatronic systems performing point-to-point trajectories, optimizing execution times and/or primitive parameters [6]. In that work, energy saving performance of several motion primitives like trapezoidal, cycloidal, and polynomial trajectories are compared. Differently, the authors in [10] present a CAD-based method for energy-optimal point-to-point trajectories adopting Chebyshev polynomials, achieving an energy consumption reduction equal to 62.9% with respect to the reference case.

However, only in few examples an online optimization is performed, e.g., in [7,8]. In these two examples, the total time is fixed and only the shape of the trajectory is optimized for energy efficiency, without considering the execution time variation. Nevertheless, modern production systems require flexibility and real-time adaptability as they are often employed in complex and dynamic environments. Therefore, systems capable of online trajectory optimization that consider both execution time and energy consumption are needed.

In this work, we propose an approach for online optimization and planning of minimum-time and minimum-time-energy trajectories for a 1-DOF linear belt driven system, based on a verified dynamic model of the system. The experimental results highlight the capabilities of the proposed approach in optimizing online the point-to-point motions, and its performance in terms of minimizing the execution time and the energy consumption, allowing the execution of sequential optimal motions without idle times.

The paper is organized as follows: in Sect. 2 the online optimization and trajectory planning approaches are illustrated. Section 3 describes the experimental setup, whereas Sect. 4 reports the experimental results. Finally, Sect. 5 concludes the work.

2 Dynamic modelling and online trajectory planning

In this work, we consider a robotic linear axis with 1-DOF, as illustrated in Fig. 1, whose dynamic modelling can be formulated as follows:

$$F = \left[m + \frac{(J_m + J_g)\, {i_r}^2 + 2J_p}{r^2} \right] \ddot{x} + a_1 \operatorname{sign}(\dot{x}) + a_2\, \dot{x} + a_3 \operatorname{atan}(a_4\, \dot{x}) \tag{1}$$

where F is the traction force on the cart, $\ddot{x}$ and $\dot{x}$ are the cart linear acceleration and velocity, respectively. Knowing the gear ratio i_r and the pulley radius r, the torque τ_m required by the motor can be computed as $\tau_m = (F\,r)/i_r$. The elasticity of the belt has been neglected. The dynamics parameters of the linear axis are considered known (Tab. 1). Anyway, in cases where the available dynamics is inaccurate or uncertain, an alternative approach based on interval arithmetic, as in [11], can be considered. The model adopted for modelling the friction force F_c is described in [12]. a_1 represents the Coulomb friction term, a_2 the viscous friction term, whereas a_3 and a_4 consider friction at low velocity. However, alternative friction formulations can be found in [13]. For the identification of friction parameters, a sinusoidal trajectory with variable amplitude and frequency is chosen as excitation motion (Fig. 2), since it covers a wide range of velocities and accelerations, allowing a reliable friction modelling. The friction parameters are optimized so that the torques calculated using the dynamic model match the experimental ones.

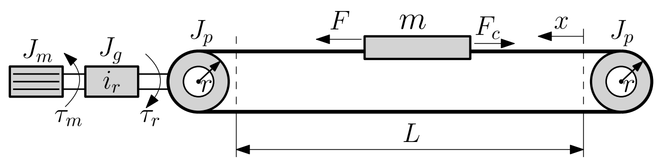


Fig. 1: Model of the robotic linear axis.

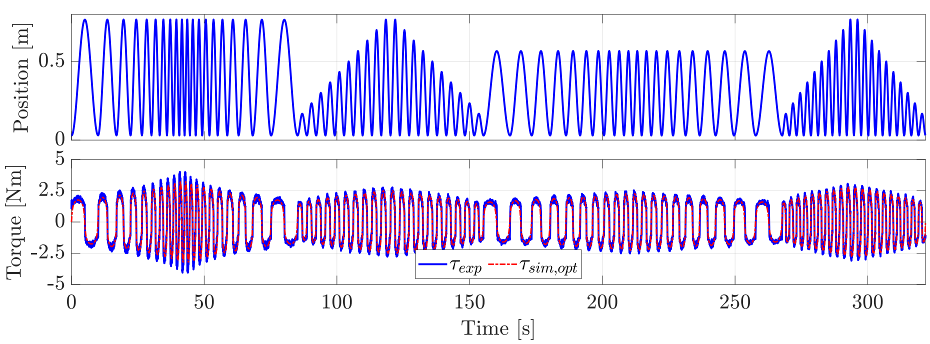


Fig. 2: Excitation trajectory used to tune the dynamic model of the linear axis.

The online optimization of point-to-point trajectories for the linear axis is performed by considering polynomials motion laws of degree d equal to 3, 5, and 7 (poly3, poly5, poly7), i.e., $x(t) = \sum_{i=0}^{d} p_i\, t^i$, where p_i represents the i-th coefficient. The execution time for the generic trajectory from an initial point to a final point is defined as $\Delta t = t_f - t_i$. The constraints for the poly3 motion law are $x(t_i) = x_i$, $x(t_f) = x_f$, $\dot{x}(t_i) = 0$, and $\dot{x}(t_f) = 0$. In the poly5 trajectory, the constraints $\ddot{x}(t_i) = 0$, and $\ddot{x}(t_f) = 0$ are also considered, whereas for the poly7 motion law $\dddot{x}(t_i) = 0$, and $\dddot{x}(t_f) = 0$ are also imposed. The optimization problem aims at finding the optimal execution time of the point-to-point motion for two different cases that can be formulated as follows:

- Case (1): minimum-time optimization:

$$\min_{t} \quad w_0\, t \tag{2}$$

- Case (2): minimum-time-energy optimization:

$$\min_{t} \quad \omega\, t + (1-\omega)\, E \quad \text{where} \quad E = \int_0^t F\, \dot{x}\, dt \tag{3}$$

In both cases, the optimization problem is subjected to the following constraints:

$$\begin{cases} x_{min} < x < x_{max} & x_{min}, x_{max} \text{ lower and upper joint position bounds} \\ \mid \dot{x} \mid < \dot{x}_{max} & \dot{x}_{max} \text{ joint velocity bound} \\ \mid \ddot{x} \mid < \ddot{x}_{max} & \ddot{x}_{max} \text{ joint acceleration bound} \\ \mid \tau_m \mid < \tau_{m,max} & \tau_{m,max} \text{ motor torque bound} \end{cases} \tag{4}$$

Case (1) aims at finding the minimum execution time t that does not exceed the constraints of the linear axis. ω_0 is a positive weight used to improve the convergence of the optimization. Differently, Case (2) optimizes the time that minimizes the weighted sum of time t and mechanical energy E. The trade-off between these two contributions is set by means of the parameter $\omega \in [0,\, 1]$.

3 Experimental setup

The proposed approach is validated on a robotic linear axis by AutomationWare available at the Robotics and Mechatronics Laboratory at the University of Udine (Italy), shown in Fig. 3. The linear axis employs a toothed belt made of polyurethane with steel strands. The system is actuated by a brushless motor HDT SR08L by HDT Srl, connected to the driven pulley of the linear axis through a planetary gearbox with reduction ratio i_r equal to 10. The system does not provide the position of the cart with respect to a fixed reference frame using external sensors, but only through the angular position of the driven pulley.

The maximum position L of the cart is equal to 0.8 m. The parameters used in the model of the system are reported in Tab. 1. The friction parameters are optimized adopting the constrained nonlinear multivariable Matlab function *fmincon* and the sequential quadratic programming (SQP) method. The system is controlled by means of a computer equipped with 31.1 GB RAM and an Intel Core i9 processor running Ubuntu 20.04, using ROS 2 (Robot Operating System) Galactic Geochelone with Python 3.8. A sampling rate of 1 kHz is used for both the input trajectories and the measured cart position and velocity, and motor torque data acquired in ROS2.

The optimization problems in (2) and (3) with (4) are implemented using the IPOPT algorithm of the open-source tool for nonlinear optimization CasADi [14]. IPOPT implements a primal-dual interior point method, using line searches based on Filter methods. The maximum number of iterations of the optimization is set equal to 10. For Case (1), the lower and upper bounds for the time

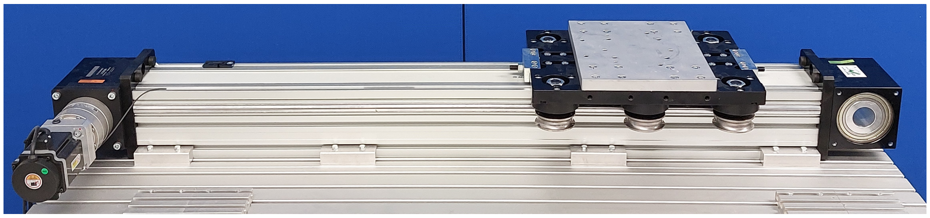


Fig. 3: Robotic linear axis.

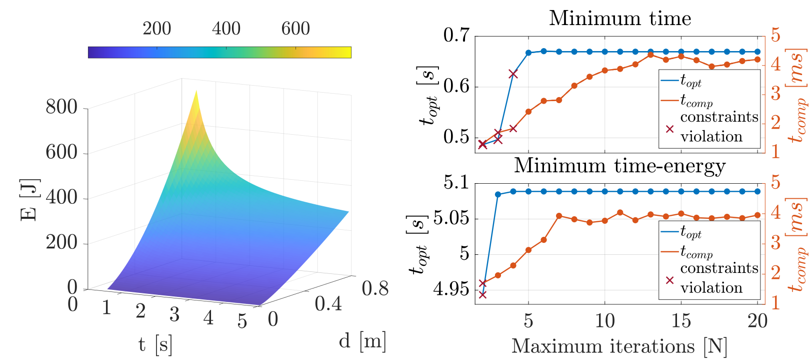
Parameter	**Symbol**	**Value**	**Parameter**	**Symbol**	**Value**
Cart mass	m	$18.69\ kg$	Gear ratio	i_r	10
Motor inertia	J_m	$9.3 \cdot 10^{-5}\ kg\,m^2$	Friction par.	a_1	$1 \cdot 10^{-4}\ N$
Gearbox inertia	J_g	$7.0 \cdot 10^{-5}\ kg\,m^2$	Friction par.	a_2	$316.05\ Ns/m$
Pulley inertia	J_p	$2.1 \cdot 10^{-3}\ kg\,m^2$	Friction par.	a_3	$232.40\ N$
Pulley radius	r	$0.045\ m$	Friction par.	a_4	22.58

Table 1: Parameters used in the dynamic model of the linear axis (friction parameters have been optimized to fit the experimental data shown in Fig. 2).

duration of each robot trajectory are 0.1 and 3 s, respectively, whereas for Case (2) they are set equal to 0.3 and 20 s, respectively. Furthermore, the value of the parameter ω for Case (2) is chosen equal to 0.9, to give more weight to the time reduction with respect to energy. The optimization algorithm is tested by planning point-to-point optimal motions through 100 random way points within the joint position bounds. These way points are the same for both Cases and for each of the considered motion laws. The optimization process is performed in parallel with the trajectory execution, i.e., while the linear axis is executing a point-to-point motion, the optimization algorithm computes the optimal time and plans the trajectory for the following motion. This parallel procedure allows the execution of one trajectory right after the previous one, without idle times.

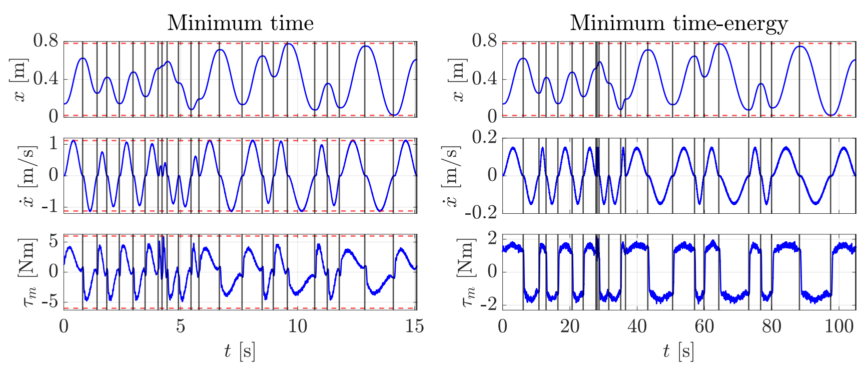
4 Experimental results

Figure 2 compares the theoretical and the experimental motor torque obtained with the excitation trajectory, highlighting a good accuracy of the dynamic model, with maximum errors of 0.5 Nm. The 3D representation of the mechanical energy E spent for one fifth-degree polynomial trajectory as a function of traveled distance d and execution time t is shown in Fig. 4a. For fixed total times higher than 3 s, the mechanical energy increases approximately linearly with the traveled distance. Differently, for smaller fixed total times, the spent energy increases faster with the traveled distance. A similar trend of the mechanical energy can be found fixing the traveled distance, with smaller values of the consumed energy for longer execution times and higher for shorter times. These results highlight the importance of properly setting the execution time of the point-to-point motion in order to reduce the energy consumption, allowing to choose the best trade-off between cycle time and consumed energy. Moreover,



(a) (b)

Fig. 4: Mechanical energy as a function of traveled distance and total time for the case of poly5 trajectories (a); optimal times and computational times as a function of the maximum number of iterations for a traveled distance equal to 0.4 m for the case of poly5 trajectories (b).



(a) (b)

Fig. 5: Joint position, joint velocity, and motor torque during the motions between 25 random way points for the case of poly5 trajectories.

Fig. 4b highlights that the computational times are lower than 5 ms, testifying that the proposed method is suitable for online trajectory optimization, and a feasible solution can be obtained with only 5 iterations. Constraint violations indicate that the solutions do not meet the ones in (4).

Figure 5a shows the joint position, velocity, and motor torque over time of the minimum-time trajectories through 25 exemplary way points for the case of poly5 trajectories, whereas Fig. 5b depicts the minimum-time-energy trajectories for the same way points and trajectory type. The red horizontal dashed lines indicate the corresponding limits, whereas the vertical solid lines represent the target way points. From Fig. 5a it can be seen that the joint velocity and the motor torque peaks are close to the limits, as can be expected for a minimum-

Approach	**Traj.**	T $[s]$	E_{th} $[J]$	E_{exp} $[J]$	$E_{\%}$	t_{opt} $[ms]$
minimum-time	poly3	47.80	15475	14001	9.52	3.97 ± 0.39
	poly5	50.53	14858	13804	7.09	4.93 ± 0.57
	poly7	58.21	13961	13237	5.19	7.46 ± 1.05
minimum-time-energy	poly3	335.73	7515	8390	-11.64	3.79 ± 0.43
	poly5	322.64	7625	8719	-14.35	5.53 ± 0.98
	poly7	319.62	7658	8527	-11.35	7.47 ± 1.05

Table 2: Experimental results: total execution time ($T = \sum_{i=1}^{99} \Delta t_i$) of the trajectories between the 100 way points, mechanical energy spent during all trajectories computed with the dynamic model (E_{th}) and retrieved from the experimental data (E_{exp}), energy computation percentage error ($E_{\%} = [(E_{th} - E_{exp})/E_{th}] \cdot 100$), and optimization times (t_{opt}, mean ± standard deviation).

time trajectory. Differently, Fig. 5b shows that joint velocity and motor torque for the minimum-time-energy trajectory are consistently smaller and the execution time is larger than the ones of the minimum-time trajectory, even though the weight of the energy with respect to the time is small. Finally, Tab. 2 reports the main experimental results in terms of total execution time, mechanical energy spent (theoretical and experimental) and optimization times. For Case (1) a smaller total time is obtained with the poly3 motion law with respect to the poly5 and poly7 ones, but the spent mechanical energy is higher. Contrarily, for Case (2) the smallest total time is obtained with the poly7 motion law, whereas the greater energy saving is achieved with the poly3 trajectory. For both Cases and trajectory type, the mechanical energy computed with the theoretical model is quite close to the one calculated from the experimental data. The optimization times are between 3 and 8 ms, and increase with the polynomial degree.

5 Conclusions

In this paper, the dynamic modelling of a robotic linear axis and a method for online optimization of point-to-point motions of the system have been presented. The proposed approach allows the online optimization of the trajectories and their sequential execution without idle times. The trajectories have been optimized for two different cases, i.e., minimum-time and minimum-time-energy trajectories. The experimental results have shown the feasibility and the energy saving capabilities of the proposed online optimization approach. Future work will include the analysis of different motion laws and optimization algorithms. Furthermore, we plan to adopt the robotic linear axis as a redundant prismatic joint for an industrial manipulator for energy efficiency applications.

Acknowledgments

This research was developed within the Laboratory for Big Data, IoT, Cyber Security (LABIC) funded by Friuli Venezia Giulia region, the Laboratory for

Artificial Intelligence for Human-Robot Collaboration (AI4HRC) funded by Fondazione Friuli, the Interconnected Nord-Est Innovation Ecosystem (iNEST), and partially funded by the European Union Next-Generation EU (PNRR – Mission 4.2, Investment 1.5 – D.D. 1058 23/06/2022, ECS 00000043).

References

1. F. Vanbecelaere, N. Van Oosterwyck, S. Derammelaere, A. Cuyt, M. Monte, and K. Stockman. On-line motion profile optimization for reciprocating mechanisms. *Mechanism and Machine Theory*, 173:104833, 2022.
2. G. Carabin and L. Scalera. On the trajectory planning for energy efficiency in industrial robotic systems. *Robotics*, 9(4):89, 2020.
3. L. Scalera, P. Boscariol, G. Carabin, R. Vidoni, and A. Gasparetto. Enhancing energy efficiency of a 4-DOF parallel robot through task-related analysis. *Machines*, 8(1):10, 2020.
4. F. Vidussi, P. Boscariol, L. Scalera, and A. Gasparetto. Local and trajectory-based indexes for task-related energetic performance optimization of robotic manipulators. *Journal of Mechanisms and Robotics*, 13(2):021018, 2021.
5. L. Scalera, G. Carabin, R. Vidoni, and T. Wongratanaphisan. Energy efficiency in a 4-DOF parallel robot featuring compliant elements. *International Journal of Mechanics and Control*, 20(02):49–57, 2019.
6. G. Carabin and R. Vidoni. Energy-saving optimization method for point-to-point trajectories planned via standard primitives in 1-DoF mechatronic systems. *The Int. Journal of Advanced Manufacturing Technology*, 116(1-2):331–344, 2021.
7. Y. Wang, Y. Zhao, S. A. Bortoff, and K. Ueda. A real-time energy-optimal trajectory generation method for a servomotor system. *IEEE Transactions on Industrial Electronics*, 62(2):1175–1188, 2014.
8. D. Donà, B. Lenzo, P. Boscariol, and G. Rosati. A real-time capable method for planning minimum energy trajectories for one degree-of-freedom mechatronic systems. *Control Engineering Practice*, 142:105766, 2024.
9. L. Clemen and C. J. Rupp. Single-and Multi-Degree-of-Freedom Servo Trajectory Generation: An Optimization Framework, Implementation, and Examples. In *IEEE/ASME Int. Conf. on Adv. Intell. Mech.*, pages 1040–1047. IEEE, 2023.
10. N. Van Oosterwyck, F. Vanbecelaere, F. Knaepkens, M. Monte, K. Stockman, A. Cuyt, and S. Derammelaere. Energy optimal point-to-point motion profile optimization. *Mech. Based Sesign of Structures and Machines*, 52(1):239–256, 2024.
11. L. Scalera, C. Nainer, A. Giusti, and A. Gasparetto. Robust safety zones for manipulators with uncertain dynamics in collaborative robotics. *International Journal of Computer Integrated Manufacturing*, pages 1–13, 2023.
12. M. Grotjahn, M. Daemi, and B. Heimann. Friction and rigid body identification of robot dynamics. *Int. Journal of Solids and Structures*, 38(10-13):1889–1902, 2001.
13. G. Fabris, L. Scalera, P. Boscariol, and A. Gasparetto. Experimental analysis and comparison of friction models applied to the UR5e robot. In *Proc. of the 6th IFToMM Symposium on Mechanism Design for Robotics*. Springer, 2024.
14. J. AE Andersson, J. Gillis, G. Horn, J. B Rawlings, and M. Diehl. CasADi: a software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, 11:1–36, 2019.