

# A closed-loop automatic tuning method for velocity control of oscillatory mechatronic systems

Marco Giacomelli

*Department of Mechanical and Industrial Engineering, University of Brescia*  
m.giacomelli009@unibs.it

Giovanna Finzi

*Department of Mechanical and Industrial Engineering, University of Brescia*  
giovanna.finzi@unibs.it

Luca Simoni

*Department of Mechanical and Industrial Engineering, University of Brescia*  
luca.simoni@unibs.it

Davide Colombo

*Gefran Drives and Motion Srl, Gerenzano (VA), Italy*  
davide.colombo@gefran.com

Vlastimil Šetka

*Department of Cybernetics, University of West Bohemia*  
setka@ntis.zcu.cz

Antonio Visioli

*Department of Mechanical and Industrial Engineering, University of Brescia*  
antonio.visioli@unibs.it

This is the pre-peer reviewed version of the following article: A closed-loop automatic tuning method for velocity control of oscillatory mechatronic systems, which has been published in final form at

DOI: 10.1109/IECON.2019.8926986.

This article may be used for non-commercial purposes in accordance with Journal terms and conditions for Self-Archiving.

## Abstract

In this paper a closed loop automatic tuning procedure for the velocity control of oscillatory mechatronic systems is proposed. The transfer function of the system is estimated relying only on the measurements on the motor side, resonances are identified and biquadratic filters and the PID controller are tuned in order to improve the pre-existing control system by reducing the oscillations on the load side. Experimental results obtained with a Hardware-In-the-Loop setup show the effectiveness of the method.

## Index Terms

Autotuning, mechatronic systems, PID controllers, oscillations.

## I. INTRODUCTION

In the field of motion control for mechatronic systems, Proportional-Integral-Derivative (PID) controllers are widely employed. In fact, despite their relatively simplicity, PID controllers are able to provide acceptable performance for most industrial applications.

Although much work has been done for the development of automatic tuning techniques for PID controllers in process control applications [1], [2], not many methodologies are available for the automatic tuning of PID controllers for mechatronic applications.

In fact, while there are many papers regarding identification techniques [3] or tuning rules [4]–[6] for mechatronic systems, the development of robust and effective autotuning procedures where the identification and tuning phases are suitably connected is still an important research topic.

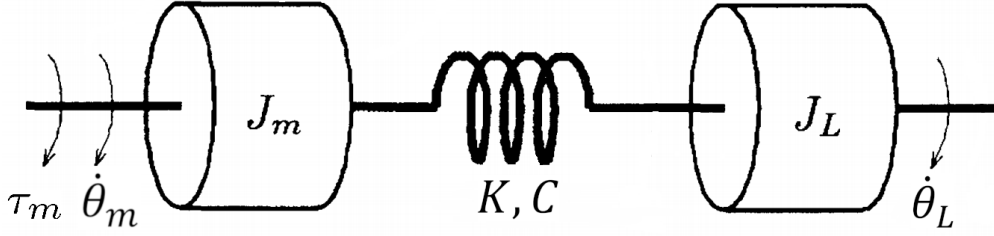


Fig. 1. Two-inertia system.

For example, recently, in [7] an automatic procedure based on the identification of the system physical parameters is used to tune current, velocity and position control loops using PID controllers. In [8] the so-called Optimal Bandwidth Search Method is exploited together with an inertia identification method in order to tune a PI controller.

However, in these cases, the system elasticity is not taken into account. Indeed, rapid motions and reduced stiffness in modern mechatronic systems often lead to unwanted mechanical vibrations and to excitations of oscillatory modes. Such oscillations are identified as the most limiting factor for the achievable bandwidth [9], so that they can not be overlooked when performing the tuning of a non-rigid mechatronic system. While several researches focus on the identification of oscillatory mechatronic systems and on the tuning of control schemes that specifically takes into account the oscillatory modes of the system [10]–[13], few works have been done in the direction of a complete autotuning procedure for the control of oscillatory mechatronic systems. For example, a two mass system is explicitly addressed in [14], where an automatic procedure for the velocity control of a servo motor is proposed, based on the identification of the inertia. However, the tuning rules are empirically obtained, and their formula structure varies depending on the natural frequency value.

In this paper, an autotuning procedure is proposed for the PI velocity controller of an oscillatory mechatronic system. The oscillatory modes are also identified and suppressed by means of properly tuned biquadratic filters.

Thus, unlike most of existing autotuning techniques, the presence of oscillations is explicitly addressed, and the oscillations on the load side are avoided by adding standard biquadratic filters to the velocity PI control loop. Remarkably, the oscillations on the load side are identified by analyzing data collected on the motor side, as most of the servo actuated systems do not have sensors on the load side. Moreover, differently from the open-loop technique proposed in [15], the whole procedure operates in closed loop, so that no assumptions on the system are required and the safety during the identification phase is assured by the feedback controller already in place (even if badly tuned).

Finally, a relevant feature of the proposed technique is its ability to identify the system by collecting data that span over the whole range of position, velocity and torque values, given the constraints on these variables specified by the user.

The paper is structured as follows: in Section II the problem is formulated, in Section III the proposed method is explained. The experimental setup is described in Section 5, while in Section VI the obtained experimental results are shown and analyzed; Section VII concludes the paper.

## II. PROBLEM FORMULATION

The basic component of a mechatronic system can be generally schematized as a two-inertia system shown in Figure 1, where  $\tau_m$  is the motor torque,  $\dot{\theta}_m$  is the motor velocity and  $\dot{\theta}_L$  is the load velocity. The system is characterised by the following parameters:

- $J_m$  inertia of the motor,
- $J_L$  inertia of the load,
- $C$  damping coefficient of the transmission,
- $K$  elastic constant of the transmission,
- $K_f$  static friction coefficient,
- $B_m$  viscous friction coefficient.

The transfer function of the system in Figure 1, considering the motor torque  $\tau_m$  as the input and the load velocity  $\dot{\theta}_L$  as the output and ignoring the nonlinearity introduced by the static friction, is given by

$$G(s) = \frac{Cs + K}{J_L J_m s^3 + (J_L B_m + C J_L + C J_m) s^2 + (C B_m + J_L K + J_m K) s + K B_m}. \quad (1)$$

In most of industrial mechatronic systems the control scheme is based on the feedback of the motor velocity/position, as it is usually impractical and/or too expensive to directly measure the velocity/position of the load. Due to this fact, it is appropriate

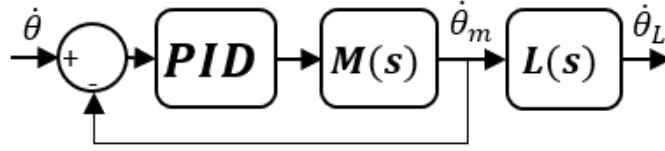


Fig. 2. Standard velocity control scheme of a mechatronic system with the feedback on the load velocity.

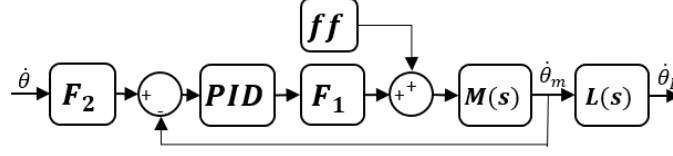


Fig. 3. Proposed velocity control scheme: bi-quadratic filters  $F_1$ ,  $F_2$  and friction feedforward compensator  $ff$ .

to divide the transfer function (1) into two transfer functions: one, denoted as  $M(s)$ , describing the relationship that links the motor torque  $\tau_m$  and the motor velocity  $\dot{\theta}_m$ , and the other one, denoted as  $L(s)$ , that describes the relationship between motor velocity  $\dot{\theta}_m$  and load velocity  $\dot{\theta}_L$ , so that

$$G(s) = M(s)L(s). \quad (2)$$

where

$$M(s) = \frac{J_L s^2 + C s + K}{J_L J_m s^3 + (J_L B_m + C J_L + C J_m) s^2 + (C B_m + J_L * K + J_m K) s + K B_m} \quad (3)$$

and

$$L(s) = \frac{C s + K}{J_L s^2 + C s + K}. \quad (4)$$

In this sense, the velocity control of the system of Figure 1 with the motor velocity feedback can effectively be divided into a closed loop control part (the control of the motor velocity) and an open loop control part (the control of the load velocity), as shown in Figure 2.

Two main problems arise in defining an autotuning procedure for the scheme in Figure 2. Firstly, both transfer function  $M(s)$  and  $L(s)$  have to be identified while acquiring informations only on the motor side. Secondly, in order to avoid undesired oscillations on the load side, a proper control system must take into account both the closed-loop control of the motor and the open-loop part which characterizes load and elastic transmission and which can be critical when considering poorly damped systems, as oscillations on the load side drastically decrease the performance of the system.

### III. PROPOSED METHODOLOGY

In order to cope with the problems formulated in Section II, the control scheme shown in Figure 3 is proposed. Two bi-quadratic filters  $F_1$  and  $F_2$  are used in this architecture, one filtering the velocity setpoint and one filtering the output of the internal velocity PID controller, while a feedforward torque signal compensates for the static friction.

Hereafter, we assume that a velocity controller is already present and needs to be retuned, for example because of changes in the system parameters (e.g. different load inertias) that deteriorate the overall control performance. For the purposes of system identification, the two bi-quadratic filters in Figure 3, if already present, are set as unitary gains. In Section IV a method for the tuning of a first conservative velocity loop controller is presented, useful in the case when no pre-existent tuned controller is available.

The flowchart of the proposed procedure is shown in Figure 4. First, the noise affecting the measurement of the motor velocity is detected by acquiring the signal with the motor stopped for a significant amount of time (e.g. one second), and the maximum of the signal is determined. Then, a velocity ramp with a slight gradient is given as reference to the system in order to identify the static friction torque  $\hat{K}_f$ , determined as the value of the torque when the measured velocity overcomes the maximum estimated value of the noise for a meaningful number of acquisitions. The input slope gradient  $\Delta\tau$  can easily be defined as a percentage of the velocity limit  $\dot{\theta}_m a x$ , so that

$$\Delta\dot{\theta} = \frac{\dot{\theta}_m a x}{N_s}, \quad (5)$$

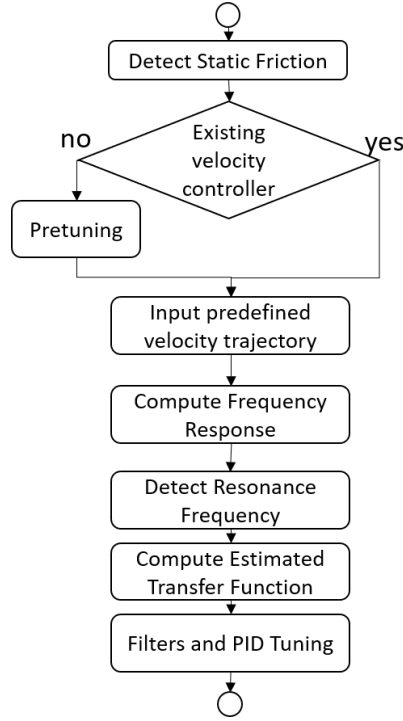


Fig. 4. Flowchart of the proposed autotuning procedure.

where  $N_s$  is the number of sampling periods necessary to reach the maximum velocity  $\dot{\theta}_{max}$ . In this paper the value of  $N_s$  is arbitrarily set to 100000, value that can be increased or reduced depending on the relevance of the static friction on the particular system.

In order to identify the frequency response of the system, a velocity reference signal is built considering the limits of the system. We assume to have position, speed and torque constraints denoted as  $\theta_{max}$ ,  $\dot{\theta}_{max}$  and  $\tau_{max}$  respectively. In particular, a value  $\bar{\theta}$  is chosen randomly in the range  $[0.2\dot{\theta}, \dot{\theta}_{max}]$ . A velocity step signal with amplitude  $\bar{\theta}$  is kept constant for a time  $T_p$ , so that the position constraint is respected, that is  $T_p = \frac{\theta_{max}}{\bar{\theta}}$ . Then, the system is brought to its initial position by means of a velocity step in the opposite direction, with the same amplitude as before. In order to decrease the undesirable effects of a step signal on the velocity, such as the saturation of the torque, the velocity signal is filtered by a slope limiter that limits the slope to a value of  $\frac{J_m}{\tau_{max}}$ , where  $J_m$  is known from the motor datasheet.

It has to be noted that the total inertia of the system is greater than the inertia of the motor, so that the control will probably saturate. Nonetheless, without any information regarding the system inertia, the introduction of this slope is an improvement with respect to a step in the velocity reference.

This procedure is executed iteratively, changing the value of  $\bar{\theta}$  in a random fashion, for a total time  $T_{tot}$  that is chosen depending on the desired spectral resolution, remembering that the spectral resolution  $\Delta f$  is given by

$$\Delta f = \frac{1}{T_{tot}}.$$

For most of mechatronic systems, a total time  $T_{tot} = 80$  s is sufficient, considering that it gives a spectral resolution (and a minimum frequency on the spectral density analysis) of 0.0125 Hz.

In order to enrich the frequency components of the input signal, a logarithmic chirp torque signal is added, spanning from the minimum frequency  $\Delta f$  to the maximum identifiable frequency for the Nyquist theorem, that is

$$f_{max} = \frac{1}{T_s},$$

where  $T_s$  is the system sampling period. The amplitude of this chirp is set as the 20% of the maximum torque  $\tau_{max}$ .

For the identification of the system frequency response, the Ljung method is employed [16]. This method, using the velocity reference signal, the torque input signal and the feedforward logarithmic chirp torque signal, along with the measured (motor) velocity, computes a high-order transfer function of the system. The high-order transfer function obtained through the Ljung method must be reduced to a transfer function in the form (3). In order to obtain this reduced transfer function, the analytical method in [12] is used, as it is specially devised in order to identify this kind of oscillatory systems, using the frequency

response of the high-order transfer function found with the method in [16] as data for the reduced-order transfer function. The method in [12] requires three meaningful frequencies at which the frequency response is considered, in order to identify the transfer function (3). The first of these points is chosen to be close to the mechanical pole of the system, that is, by choosing the frequency at which the module of the frequency response is reduced by 3 dB with respect to its lowest frequency value. The other two points must be chosen in order to be close to the resonance and antiresonance peaks. In order to find these frequencies, the resonance and anti-resonance peaks of the frequency response are identified by detecting the maximum of the product of the prominence and the width of each phase peak. Then, the resonance and antiresonance frequencies can be approximately identified assuming that at these frequencies their phase is the peak phase minus half of its prominence. Once the three points have been determined, the method in [12] gives a set of parameters which is used as the initial set for a nonlinear Levenberg–Marquardt optimization algorithm [17], similarly as what is done in [18]. Nonetheless, in [18] no indication is given regarding the initial set of parameters for the nonlinear optimization, which could easily converge to a local minimum that could be far from the global optimal solution, resulting in an inaccurate identification system. The identified transfer function is finally in the form

$$M(s) = \frac{a_1 s^2 + a_3 s + a_4}{s + a_2 s^2 + a_5 s + a_6}. \quad (6)$$

Once the model of the system has been obtained, the oscillatory components can be compensated using bi-quadratic filters as shown by the proposed control scheme in Figure 3. The bi-quadratic filter acting inside the velocity control loop deletes the system resonance and antiresonance, obtaining the normalized filter

$$F_1(s) = \frac{\frac{1}{a_6} s^2 + \frac{a_5 s}{a_6} + 1}{\frac{1}{a_4} s^2 + \frac{a_3 s}{a_4} + 1} \quad (7)$$

so that the filtered system transfer function becomes a first-order transfer function in the form

$$G(s)F_2(s) = \frac{a_1 a_4}{a_6(s + a_2)} \quad (8)$$

Subsequently, a velocity PI controller in the form

$$C(s) = K_p \left( 1 + \frac{1}{T_i s} \right) \quad (9)$$

is tuned based on (8) imposing a user-chosen phase margin  $\phi_m$  and cutoff frequency  $\omega_c$ . The resulting PI parameters are

$$T_i = \frac{\tan\left(\phi_m - \frac{\pi}{2} + \arctan\left(\frac{\omega_c}{a_2}\right)\right)}{\omega_c} \quad (10)$$

$$K_p = \frac{a_1 a_4}{a_6} \frac{T_{i2}}{\sqrt{1 + T_{i2}^2 \omega_c^2 (\omega_c^4 + \omega_c^2 a_2^2)}}.$$

While the closed-loop control of the motor can guarantee satisfactory performance on the motor side, the performance on the load side could even decrease if only the motor is taken into account. In fact, a resonance is present in the transfer function (4), which describes the relation between the velocity of the motor and the velocity of the load and this transfer function is outside of the control loop. Even if this transfer function can not be directly identified as there is no sensor on the load side, its resonance can be obtained knowing that it has the form of (4), and its complex poles are the same as the complex zeroes in (3), which have been identified as (6). Knowing that, in order to delete the oscillations that affect the open-loop part of the scheme in Figure 3, the setpoint is filtered by a second bi-quadratic filter  $F_1$ . The parameters of  $F_1$  are obtained by cancelling the oscillatory poles in (4), which are in the form  $s^2 + a_3 s + a_4$ , by imposing a unitary gain and by adding two real poles  $p_1 = p_2$  at the same frequency of the zeroes, that is,

$$p_1 = p_2 = -\frac{1}{2} \sqrt{a_3^2 + |a_3^2 - 4a_4|}. \quad (11)$$

The external filter is therefore expressed as

$$F_2(s) = \frac{\frac{1}{a_4} s^2 + \frac{a_3 s}{a_4} + 1}{\frac{1}{p_1^2} s^2 + \frac{2}{p_1} s + 1}. \quad (12)$$

Moreover, as shown in Figure 3, a feedforward signal compensates for the static friction: being  $\bar{\theta}$  the velocity reference, the feedforward signal  $ff$  is calculated as

$$ff = \hat{K}_f \text{sign}(\bar{\theta}). \quad (13)$$

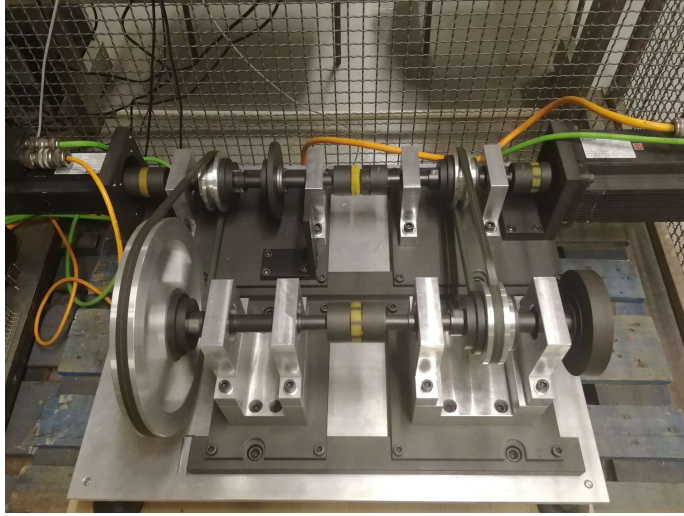


Fig. 5. Experimental HIL setup.

#### IV. PRE-TUNING

In some cases, for example during the commissioning of a mechatronic system, no pre-existent velocity controller is available, and guessing the correct parameters for a PID controller from scratch is not trivial. Nonetheless, in order to respect the limits of the system, a closed-loop autotuning procedure would be preferable also in this case. Here we present a rapid pre-tuning method for the tuning of a proportional controller, that can be effectively exploited by the proposed autotuning procedure. Firstly, the whole system is considered to be composed only by the motor itself. Based on the motor inertia  $J_m$ , which is known from the motor datasheet and an unknown viscous friction  $B_m$ , a proportional controller is tuned considering the system in the form

$$G(s) = \frac{1}{J_m s + B_m}. \quad (14)$$

A proportional velocity controller  $C(s) = K_p$  is tuned in order to keep the system under control while performing the autotuning procedure. Having already determined the value of the static friction torque  $\hat{K}_f$ , a first guess is to choose the value of  $K_p$  so that, when the velocity reference is set to the maximum value  $\dot{\theta}_{max}$ , the torque output of the controller exceeds the value of the static friction torque  $\hat{K}_f$  by a factor of two, in order to be very conservative with the tuning, but being nonetheless sure that the system will move.

Then, a step in the velocity reference with an amplitude  $\dot{\theta}_{ref} \leq \dot{\theta}_{max}$  is given to the closed-loop system, followed by a step of opposite amplitude that brings the system back to the initial position. The duration of those steps is computed so that the limits in the position are not exceeded. The torque signal  $\tau_m$  and the velocity of the motor  $\dot{\theta}_m$  are collected and analyzed the instant before the velocity reference is switched. It can be assumed that, by then, the system has reached its steady state, so that, taking into account the transfer function (14), the viscous friction  $B_m$  can be identified as

$$B_m = \frac{\tau_m - \hat{K}_f}{\dot{\theta}_m} \quad (15)$$

Then, knowing that the steady state error of the closed-loop control, if the static friction is compensated, is given by

$$e_{ss} = 1 - \frac{K_p}{K_p + B_m}, \quad (16)$$

the proportional gain of the controller can be set depending on the desired closed-loop steady state error. As for the autotuning procedure, a non-aggressive control is sufficient, so that a value of 0.1 for the steady state error is acceptable. This results in

$$K_p = 9B_m = 9 \frac{\tau_m - \hat{K}_f}{\dot{\theta}_m}. \quad (17)$$

#### V. EXPERIMENTAL SETUP

The closed-loop autotuning method proposed in Section III has been implemented on a Hardware-In-the-Loop (HIL) setup in Figure 5. The HIL setup consists in two brushless servomotors, controlled by two Gefran ADV200S servo drives, connected together by means of a rigid transmission.

Parameter	Value	Parameter	Value
$C$	0.003 Nms/rad	$K$	1.0 Nm/rad
$J_m$	0.0079 kgm	$J_L$	0.0079 kgm
$B_m$	0.005 Nms/rad	$K_f$	0.3 Nm
$\theta_{max}$	400 rad	$\dot{\theta}_{max}$	150 rad/s
$\tau_{max}$	5 Nm		

TABLE I  
PARAMETERS OF THE SYSTEM SIMULATED ON THE HIL SETUP.

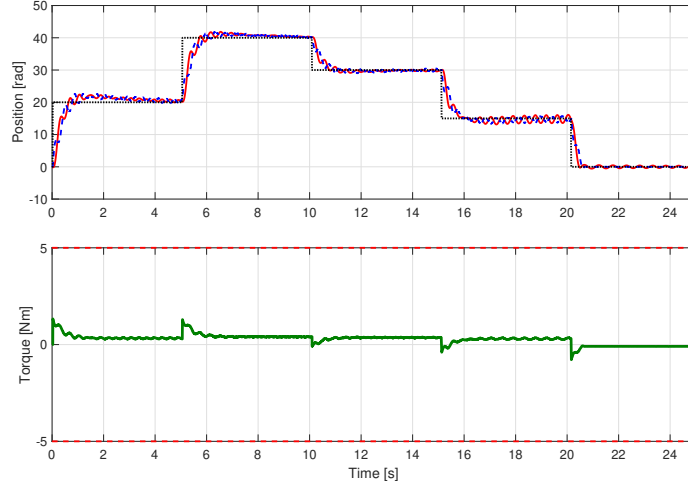


Fig. 6. Performance of the system with the badly tuned velocity controller. Load velocity (red solid line), motor velocity (blue dashed line), velocity reference (black dotted line) and torque (green solid line). The torque limits are shown in red dashed line.

While the identification trajectory and the feedforward signal are generated by the drives using standard function blocks, the identification and tuning phases have been implemented on a standard PC running Matlab<sup>TM</sup> 2017b as they require functions and libraries that are not easily implementable on standard industrial drives due to their limited memory and limited computational power. However, the implementation of the procedure using a standard PC does not entail a loss of generality for industrial applications, as the code running on Matlab has been tested for auto code generation compatibility and could run on other standard industrial hardwares (PLCs, industrial PCs...). The data are collected and stored using a Real-time Controller with REXYGEN software platform [19] which communicates with the drives by means of the EtherCAT protocol, while the communication with the Matlab script running on the PC uses HTTP REST API provided by REXYGEN.

The autotuning procedure starts when the start command is received by the Matlab script, which effectively simulates the presence of a centralized supervisor that is responsible for the autotuning procedure and data analysis. All the signals are generated and read with a sampling period of 1 ms. Once the identification trajectory is completed, the signals are loaded to the PC and elaborated using Matlab environment following the steps described in Section III, in order to find the transfer function of the system. The tuning of filters and of the controller is then performed, with the phase margins and the cutoff frequencies defined by the operator. The whole procedure is completely automatic and is activated by launching a Matlab script.

One motor is used to simulate the real motor of the system in Figure 1, while the other motor is used to simulate the torque response of the system consisting of the elastic transmission and the load. For the purpose of testing the proposed autotuning procedure, a generic two-inertia system is simulated. Limits and parameters of the simulated system are shown in Table I. The transfer function (6) is given by

$$M(s) = \frac{126.58(s^2 + 0.3797s + 126.6)}{(s + 0.3166)(s^2 + 1.076s + 253.1)}. \quad (18)$$

## VI. EXPERIMENTAL RESULTS

We initially have a PI velocity controller in the form (9) with  $K_p = 0.05$  and  $T_i = 2.0$ . The performance of the system with such controller, shown in Figure 6, is considered to be unsatisfactory, and the autotuning procedure is used in order to retune the system. The value of the static friction torque, identified using a ramp in the velocity reference as explained in Section III, is

$$\hat{K}_f = 0.28 \text{ Nm}.$$

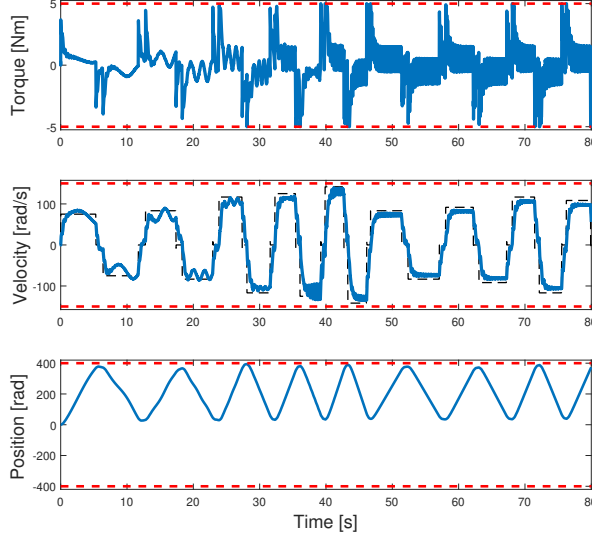


Fig. 7. Trajectory of the system during the identification. The velocity reference is the black dashed line, while the limits are shown as red dashed lines.

The identification experiment is performed for a total time  $T_{tot} = 80$  s, so that the frequency resolution of the signals is

$$\Delta f = \frac{1}{T_{tot}} = 0.0125 \text{ Hz.} \quad (19)$$

The sampling period of the setup is  $T_s = 0.001$  s, so that, for the Nyquist sampling criterion, the maximum identifiable frequency is 500 Hz. Nonetheless, the maximum frequency for the analysis of the system has been limited to 100 Hz as no resonances are expected after this value for the system.

The identification trajectory is then executed and data are collected. In Figure 7 the torque, velocity and position of the system are shown. It must be noted that the limits, plotted in red-dashed lines, are always respected during the identification phase.

The data are then elaborated using the Ljung method in order to obtain a very accurate (even if very high-order) transfer function. The pole, resonance and antiresonance are then detected based on the Bode plot of this high-order transfer function. Using the analytical method firstly and the nonlinear optimization secondly, the low-order transfer function is finally obtained in the form (6), resulting in

$$\hat{M}(s) = \frac{129.7(s^2 + 0.9193s + 124.2)}{(s + 0.3719)(s^2 + 1.104s + 257.5)}. \quad (20)$$

In Figure 8 the identified low-order transfer function  $\hat{M}(s)$  is compared with the high-order transfer function obtained with the Ljung method and the transfer function (18), showing that the system identification method used is effective.

For the tuning of the PI velocity controller, the cutoff frequency  $\omega_{c_v}$  and phase margin  $\phi_m$  are set to  $\omega_{c_v} = 20$  rad/s and  $\phi_m = 80^\circ$ , so that, from (10), the velocity PI controller results in

$$C(s) = 0.3196 \left( 1 + \frac{1}{2.6892s} \right). \quad (21)$$

The performance of the motor controlled by means of the PI controller (21), with the static friction compensation, is shown in Figure 9. While the dynamics on the motor side is improved, important oscillations affect the load velocity. In order to delete these oscillations, the biquadratic  $F_1$  and  $F_2$  are tuned by applying (7) and (12), obtaining

$$F_1(s) = \frac{0.003884s^2 + 0.004288s + 1}{0.008049s^2 + 0.0074s + 1} \quad (22)$$

and

$$F_2(s) = \frac{0.008049s^2 + 0.0074s + 1}{0.008049s^2 + 0.1794s + 1}. \quad (23)$$

A test of the system with the biquadratic filters activated is shown in Figure 10. The performance of the system is greatly improved with respect to Figure 6, as the motor velocity reaches the setpoint value faster, and the load velocity oscillations are reduced. With respect to the tuned PI controller without the biquadratic filters in Figure 9, the motor is slower in reaching



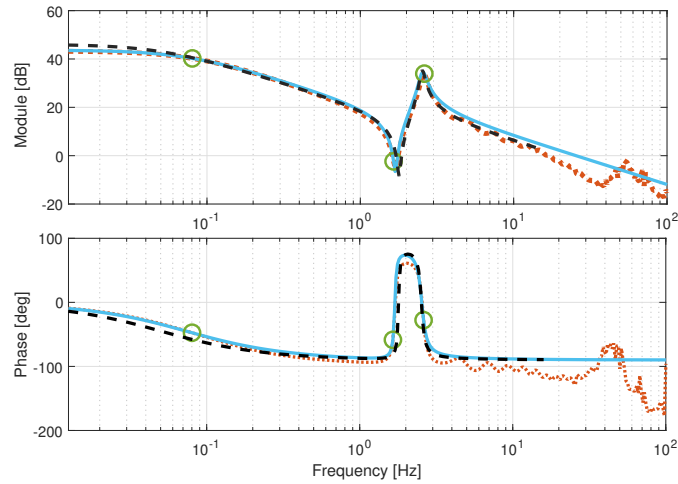


Fig. 8. Bode plot of the transfer function (18) (dashed black line), of the high-order transfer function obtained using the Ljung method (orange dotted line) and of the reduced-order transfer function (blue solid line). The three points necessary for the analytic order reduction are shown with green circles.

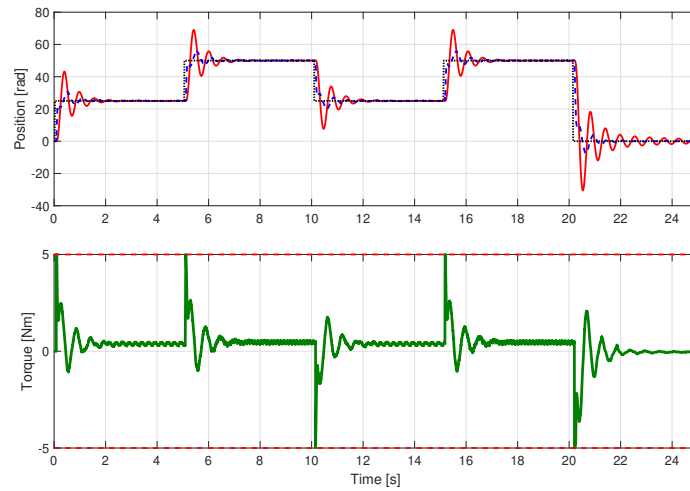


Fig. 9. Performance of the system with the obtained PI velocity controller, without the biquadratic filters. Load velocity (red solid line), motor velocity (blue dashed line), velocity reference (black dotted line) and torque (green solid line). The torque limits are shown in red dashed line.

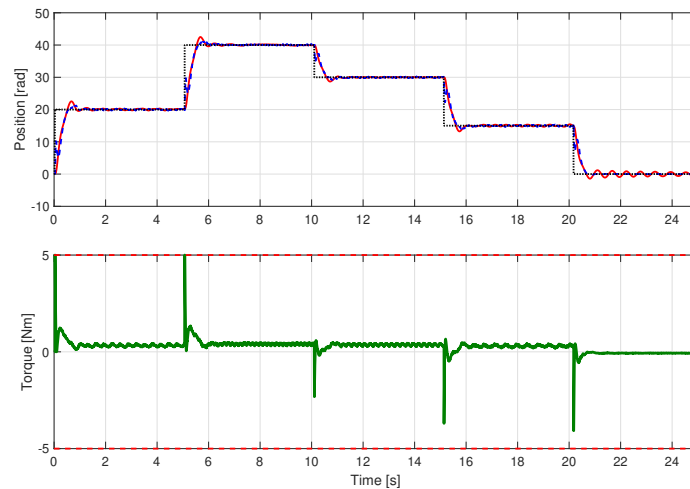


Fig. 10. Performance of the system with the obtained PI velocity controller and the biquadratic filters. Load velocity (red solid line), motor velocity (blue dashed line), velocity reference (black dotted line) and torque (green solid line). The torque limits are shown in red dashed line. The performance is increased on both motor and load side.

the setpoint velocity, but the underdamped oscillations on the load velocity are avoided by the biquadratic filters, resulting in a satisfactory performance for both load and motor velocity control. The residual oscillations that are present when the reference velocity is finally set to zero are due to the abrupt interruption of the feedforward static friction compensation. Indeed, the feedforward goes instantly from a value of  $\hat{K}_f$  to zero and, being the static friction dominant in this system, it results in residual oscillations on the load side. Nonetheless, the velocity reference step is usually filtered, so that this problem can be easily avoided in a practical case.

## VII. CONCLUSIONS

In this paper a new closed-loop procedure for the autotuning of the velocity control loops of an oscillatory mechatronic system. In this procedure, the static friction is firstly detected; then, the system is identified employing a trajectory that explicitly takes into account the position, velocity and torque constraints of the system. The proposed control scheme, which incorporates biquadratic filters and the static friction compensations, is then tuned based on the identified system. This scheme effectively reduces the oscillations on the load side while acquiring data from the motor side only and with a motor velocity feedback control.

The procedure has been implemented on a HIL setup and tested simulating an oscillatory mechatronic system. The results show increased overall performance and a reduction of the load oscillations with respect to a standard PI velocity control scheme.

## REFERENCES

- [1] K. J. Åström and T. Hägglund, "Automatic tuning of simple regulators," *IFAC Proceedings Volumes*, vol. 17, no. 2, pp. 1867–1872, 1984.
- [2] ———, *Advanced PID control*. ISA Press- Research Triangle Park (NJ), 2006, vol. 461.
- [3] Y.-Y. Chen, P.-Y. Huang, and J.-Y. Yen, "Frequency-domain identification algorithms for servo systems with friction," *IEEE Transactions on Control Systems Technology*, vol. 10, no. 5, pp. 654–665, 2002.
- [4] A. Y. Jaen-Cuellar, R. de J. Romero-Troncoso, L. Morales-Velazquez, and R. A. Osornio-Rios, "PID-controller tuning optimization with genetic algorithms in servo systems," *International Journal of Advanced Robotic Systems*, vol. 10, no. 9, p. 324, 2013.
- [5] P. Meshram and R. G. Kanojija, "Tuning of PID controller using Ziegler-Nichols method for speed control of DC motor," in *IEEE-International Conference on Advances in Engineering, Science and Management*, 2012, pp. 117–122.
- [6] M. Kushwah and A. Patra, "Tuning PID controller for speed control of DC motor using soft computing techniques - A review," *Advance in Electronic and Electric Engineering*, vol. 4, no. 2, pp. 141–8, 2014.
- [7] S.-M. Yang and K.-W. Lin, "Automatic control loop tuning for permanent-magnet ac servo motor drives," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 3, pp. 1499–1506, 2016.
- [8] C.-J. Hsu and Y.-S. Lai, "Novel online optimal bandwidth search and autotuning techniques for servo motor drives," *IEEE Transactions on Industry Applications*, vol. 53, no. 4, pp. 3635–3642, 2017.
- [9] R. Isermann, "Mechatronic systems - a challenge for control engineering," in *Proceedings of the American Control Conference, Albuquerque, New Mexico*, 1997, pp. 2617–2632.
- [10] G. Ferretti, G. Magnani, and P. Rocco, "Load behavior concerned PID control for two-mass servo systems," in *Proceedings IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, vol. 2, 2003, pp. 821–826.
- [11] E. J. Fuentes, C. A. Silva, and J. I. Yuz, "Predictive speed control of a two-mass system driven by a permanent magnet synchronous motor," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 7, pp. 2840–2848, 2012.
- [12] M. Goubelj, A. Krejci, and M. Schlegel, "Robust frequency identification of oscillatory electromechanical systems," in *Proceedings 19th International Conference on Process Control*, Strbske Pleso, SK, 2013.
- [13] G. Zhang, "Speed control of two-inertia system by PIPID control," *IEEE Transactions on industrial electronics*, vol. 47, no. 3, pp. 603–609, 2000.
- [14] Y. Guo, L. Huang, and M. Muramatsu, "Research on inertia identification and auto-tuning of speed controller for ac servo system," in *Proceedings of the Power Conversion Conference-Osaka*, vol. 2, 2002, pp. 896–901.
- [15] M. Giacomelli, D. Colombo, L. Simoni, G. Finzi, and A. Visioli, "A fast autotuning method for velocity control of mechatronic systems," *IFAC-PapersOnLine*, vol. 51, no. 4, pp. 208–213, 2018.
- [16] U. Forsell and L. Ljung, "Closed-loop identification revisited," *Automatica*, vol. 35, no. 7, pp. 1215–1241, 1999.
- [17] J. J. Moré, "The Levenberg-Marquardt algorithm: implementation and theory," in *Numerical analysis*. Springer, 1978, pp. 105–116.
- [18] S. Villwock and M. Pacas, "Application of the welch-method for the identification of two-and three-mass-systems," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 1, pp. 457–466, 2008.
- [19] O. Severa and M. Čech, "Rex—rapid development tool for automation and robotics," in *Proceedings 8th International Conference on Mechatronic and Embedded Systems and Applications*, 2012, pp. 184–189.