Mechanism and Machine Theory 173 (2022) 104833

Contents lists available at ScienceDirect

Mechanism and Machine Theory

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

On-line motion profile optimization for reciprocating mechanisms

Foeke Vanbecelaere [a,e,*], Nick Van Oosterwyck [b,1], Stijn Derammelaere [b,e], Annie Cuyt [c,d], Michael Monte [a,e], Kurt Stockman [a,e]

[a] *Department of Electromechanical, Systems and Metal Engineering, Ghent University Campus Kortrijk, Graaf Karel De Goedelaan 5, 8500 Kortrijk, Belgium*
[b] *Department of Electromechanics, Cosys-Lab, University of Antwerp, Groenenborgerlaan 171, 2020 Antwerp, Belgium*
[c] *Department of Mathematics and Computer Science, University of Antwerp, Middelheimlaan 1, 2020 Antwerp, Belgium*
[d] *College of Mathematics and Statistics, Shenzhen University, Guangdong 518060, China*
[e] *FlandersMake@UGent - Corelab EEDT-MP, Belgium*

ARTICLE INFO

Keywords:
Multi-body dynamics
Parameter estimation
Gradient-based optimization
Inertia variation
Motion control
Sliding discrete Fourier transform (SDFT)

ABSTRACT

Reciprocating mechanisms are widely used in industry because a complex movement is achieved by a simple rotation of the driven axis. Given the tendency to evolve towards more energy-efficient machines and flexible production, motion profile optimization offers a cost-effective solution as it results in large energy savings without any hardware adaptions. However, the existing optimizers are used off-line because the position-dependent parameters such as load torque and inertia of the system model must be known in advance. When the actual machine differs from the model, or when parameters change during operation due to process flexibility, the off-line determined motion profile is no longer optimal and results in unnecessary energy consumption.

This paper therefore presents an on-line approach in which the varying inertia is estimated on the actual machine and used for updating the motion profile. The sliding discrete Fourier transform is proposed for real-time estimation and a gradient-based algorithm combined with Chebyshev polynomials is proposed for on-line optimization. Experimental validation on an industrial pick-and-place unit proves that the presented method leads to similar energy savings as off-line optimizers, but without prior knowledge of the parameters, and is moreover capable of handling mass changes during operation.

1. Introduction

In modern automated machinery a load often needs to be driven repetitively between two specified positions at high speed. Reciprocating mechanisms, which are often rod mechanisms, are suited for this purpose because the driven rotary motion is inherently converted to the desired and often complex motion at the tool linkage. Next to that, one actuator can be used for each task. This not only enables faster operation compared to mechanical transmissions such as camshafts [1] but also enables modular system design [2]. But as the number of actuators increases, optimization of each individual movement becomes more important to keep energy consumption at a minimum. Considering that electrical motors are responsible for about 45% of the global electricity

[*] Corresponding author at: Department of Electromechanical, Systems and Metal Engineering, Ghent University Campus Kortrijk, Graaf Karel De Goedelaan 5, 8500 Kortrijk, Belgium.
E-mail address: foeke.vanbecelaere@ugent.be (F. Vanbecelaere).

[1] Equal contribution.

https://doi.org/10.1016/j.mechmachtheory.2022.104833
Received 16 December 2021; Received in revised form 10 March 2022; Accepted 10 March 2022
Available online 6 April 2022

consumption [3], there is a strong interest in developing techniques that achieve the same machine performance with less energy and/or increased performance for the same hardware.

Motion profile optimization is one of these techniques. This technique takes into account that rod mechanisms are characterized by a high position-dependency of the mechanical parameters such as load torque and especially inertia [4]. By solely considering these varying dynamics in the definition of the point-to-point (PTP) profile, large energy savings can be achieved while maintaining the desired motion time and using the same motor drive. Recent off-line motion profile optimization techniques, based on the genetic algorithm (GA) prove that the root mean square (RMS) torque can be reduced by 38% [5] compared to standard motion profiles such as the 1/3-profile. As proven in [6], reducing the RMS torque directly leads to energy savings. An important remark is that these savings are perceptible every cycle and also prevents the motor from overheating, which is one of the main reasons for motor failures [7].

Among other off-line optimizers found in literature like sequential quadratic programming in [8] reporting an energy reduction of 9% and the pattern search algorithm in [6] mentioning a reduction of 25%, a drawback of these algorithms is that they are based on a mathematical model of the machine of which the mechanical parameters have to be known in advance. Obtaining a realistic model of a rod mechanism is not straightforward and requires advanced in-house knowledge. Because most machine builders already obtained 3D CAD models of their machines during the design phase and assuming that there is a will to invest in additional software licenses, an option is to use multi-body dynamics software. The computer model inherently contains accurate values for the mass and inertia of the different motion bodies but assumes rigid and undamped system behavior. Stiffness and damping can be added by the software user, but obtaining accurate values of these parameters is a challenge and requires measurements on the actual machine. When an optimized motion profile is achieved based on a mathematical model that does not match the actual machine, optimal efficiency is not guaranteed.

The same is true when the mechanical parameters change during operation of the machine. For example, the mass of the load of a pick and place application is often unknown and varies due to process flexibility. Or in the case of a weaving process, the air resistance at the fibers strongly depends on humidity and temperature and alters the load condition.

A first challenge is thus to ensure that the actual parameter values are captured during operation. A recently developed algorithm in [9], based on the sliding discrete Fourier transformation (SDFT), experimentally proves that the inertia of a reciprocating mechanism is accurately tracked at high speed. The mechanism is shown in Fig. 1 and is used as an industrial pick and place unit. The literature review included in [9] states that none of the state-of-the-art estimation techniques are capable of accurately estimating a parameter that varies that fast.

A second challenge is updating the motion profile on-line according to the estimated parameters. This requires fast evaluation of the objective function. Heuristic algorithms like GA are not suited as for each optimization iteration the objective function must be computed for a large number of possible solutions. This results in long calculation times from e.g. 8 min in [5] to 1 h in [10]. Gradient-based algorithms (GBA) are computationally more efficient because in each iteration, only the objective function and its gradient are computed for 1 candidate. This is recently proven by comparing the achieved results from a novel GBA approach in [11] with the GA approach in [5] for the forward motion profile of the same pick and place unit in Fig. 1. Not only the calculation time is reduced to 70 ms while achieving the same torque reductions of 38%, but the GBA approach also requires only 2 optimization variables while GA requires 12. Higher savings up to 45% with GBA are possible by increasing the number of optimization variables, but this leads to a significantly higher calculation time which hampers on-line applicability. In addition, GBA is proven to be suited for real-time implementation in [12] where the algorithm is implemented on a Speedgoat to optimize the velocity profile, resulting in a calculation time of only 7 ms.

By tackling the mentioned challenges, this paper contributes towards optimal energy efficiency for reciprocating mechanisms by combining the SDFT approach in [9] for inertia estimation and the GBA approach in [11] for on-line motion profile optimization. An additional feature is that process changes or the use of new products is automatically considered because parameter changes are

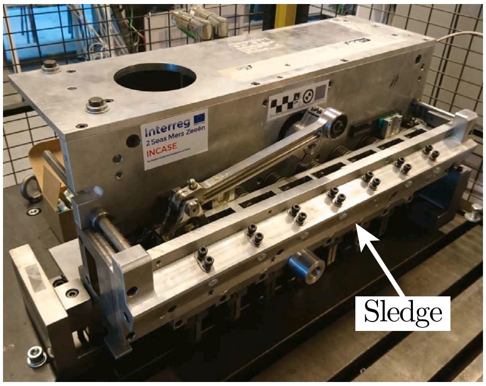

Fig. 1. The industrial pick and place unit.

captured on-line without needing a new off-line experiment. The proposed method is validated on the pick and place unit in Fig. 1. The main improvements compared to previous work are:

- Enabling inertia estimation during PTP movement instead of constant speed.
- Enabling the selection of higher tracking frequency for faster and more accurate estimation.
- Enabling optimization of the backward motion profile and not only the forward one.
- Using GBA as an on-line motion profile optimizer in a closed-loop control system.

After this introduction, Section 2 discusses the representation of the actual machine as a two-mass system model, the definition of its PTP-profile as a Chebychev polynomial, and how this enables the usage of a gradient-based optimization algorithm for optimization. In the next Section 3, the general principle of the SDFT-based estimator is explained, together with the selection of the tracking frequency and control design for accurate estimation on the one hand and efficient position control on the other hand. Next in Section 4, the proposed method is experimentally validated. The scope of the developed algorithms is further expanded in Section 5 by considering a mass change during operation and by increasing the tracking frequency. Finally, a conclusion is formulated in Section 6.

2. System and motion profile definition

2.1. Two-mass system

Representing a rod mechanism as a two-mass system is a common simplification leading to an acceptable match between the system model and the actual machine [4]. The model is shown in Fig. 2 and assumes that the driven rotor, represented with the rotor inertia J_r, is connected with the load, represented with the load inertia J_l, by a spring–damper coupling. The stiffness of the spring is represented with k and b is the damping. The load is assumed to be the equivalent of all rigidly connected linkages. The load torque is represented with T_l and is a result of gravitational and process forces acting on the linkages. This application only involves the gravitational forces. Note that due to reducing the different masses and inertias of each linkage to one load coupled to the rotor, both load inertia J_l and load torque T_l depend on the load position θ_l.

The motion equations are given in (1), with torque T and angular velocity $\dot{\theta}$ being the input and output of the system:

$$\begin{cases} T - b(\dot{\theta} - \dot{\theta}_\mathrm{l}) - k(\theta - \theta_\mathrm{l}) = J_\mathrm{r}\ddot{\theta} \\ T_\mathrm{l}(\theta_\mathrm{l}) + b(\dot{\theta} - \dot{\theta}_\mathrm{l}) + k(\theta - \theta_\mathrm{l}) = \dfrac{1}{2}\dfrac{\mathrm{d}J_\mathrm{l}(\theta_\mathrm{l})}{\mathrm{d}\theta_\mathrm{l}}\dot{\theta}_\mathrm{l}^{\,2} + J_\mathrm{l}(\theta_\mathrm{l})\ddot{\theta}_\mathrm{l} \end{cases} \tag{1}$$

An equivalent transfer function $H(s)$ of the two-mass system is achieved after transforming the motion Eqs. (1) to the frequency domain [4]:

$$H(s) = \frac{J_\mathrm{l}s^2 + bs + k}{J_\mathrm{r}J_\mathrm{l}s^3 + (J_\mathrm{r} + J_\mathrm{l})bs^2 + (J_\mathrm{r} + J_\mathrm{l})ks} \tag{2}$$

The position-dependent load inertia J_l in (2) is shown in Fig. 3 and will serve as a reference for validating the developed real-time estimator. The reference profile is calculated analytically [9] from the kinetic energy of the mechanism with the mass and inertia of each motion body known from the CAD model. The rotor inertia $J_\mathrm{r} = 3200\ \mathrm{kg{\cdot}mm^2}$ is found from a data-sheet of the motor. The remaining parameters $k = 4220\ \mathrm{Nm/rad}$ and $b = 0.4\ \mathrm{Nm{\cdot}s/rad}$ are estimated by fitting a measured frequency response, plotted in Fig. 4, with the equivalent transfer function (2). The gain characteristic is measured using torque noise injection [4,13]. The fitting procedure is illustrated in the figure by highlighting the influence of the stiffness k and damping b on the gain characteristic of

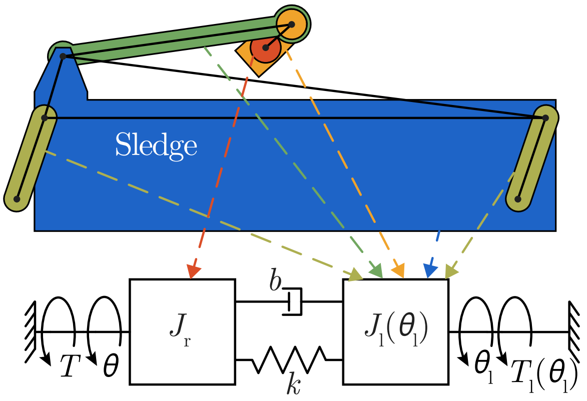
Sledge
b
J_r
$J_\mathrm{l}(\theta_\mathrm{l})$
T
θ
k
θ_l
$T_\mathrm{l}(\theta_\mathrm{l})$

Fig. 2. Model of the two-mass system with variable inertia and load torque.

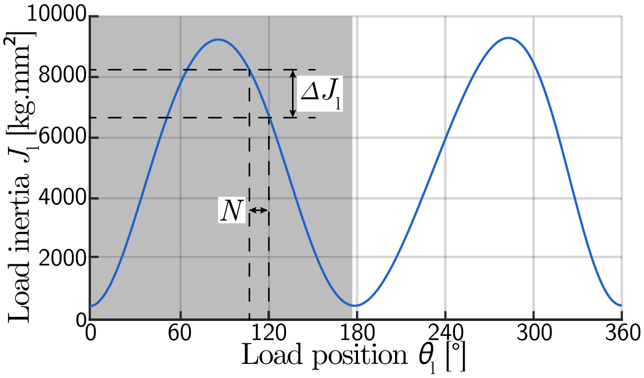

Fig. 3. Position-dependent load inertia J_l.

$H(s)$. A change of k results in a shift of both the resonance and anti-resonance peak and a change of b affects the sharpness of the peaks. In green, a frequency region is depicted where the gain of the two-mass system equals the gain of the equivalent one-mass system [9], which assumes that the load is rigidly connected with the driven rotor. This means that for frequencies $f \leq 20$ Hz of the torque input T, the parameters k and b of the coupling do not influence the output velocity $\dot{\theta}$. This property enables model reduction and is applied for construction of the objective function of the motion profile optimizer in Section 2.3 and for control design in Section 3.2.

Since a frequency-domain representation $H(s)$ of the actual machine is available, the SDFT can be used for parameter tracking at a well-selected frequency. The main approach and how to select the tracking frequency are covered in Section 3.

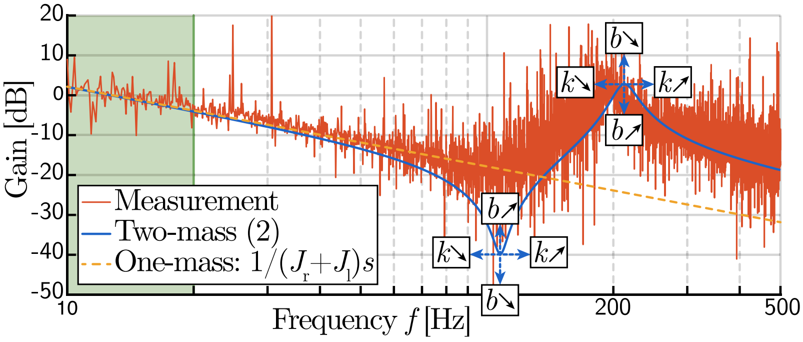

Fig. 4. Measured and theoretical gain characteristic at the position of maximum load inertia J_l.

2.2. Motion profile

Reciprocating mechanisms repetitively move from position point A to B and back. Their movement, illustrated in Fig. 5, is therefore called a PTP profile. At full operating speed, the observed pick and place unit moves from $\theta_A = 0°$ to $\theta_B = \Delta\theta = 173.8°$ in a forward time of $t_{AB} = 73.75$ ms and back to θ_A in a time of $t_{BA} = 57.5$ ms. The rest times are $t_B = 93.75$ ms and $t_A = 75$ ms, resulting in a cycle time of $\Delta t = 300$ ms. The relevant position interval between θ_A and θ_B is depicted in gray in Fig. 3.

The definition of the forward motion profile and the accompanying objective function is further handled in this section. More details for the case of the backward motion are briefly listed in Appendix B.

For mathematical efficient and robust optimization, Chebyshev polynomials of the first kind [14] are used to describe both the forward and backward profile. These orthogonal polynomials T_n of degree n are defined on the interval $x \in [-1, 1]$ and are described by the recurrence relation:

$$T_0(x) = 1; \qquad T_1(x) = x; \qquad T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x) \tag{3}$$

Fig. 5 however shows that the time t of the forward profile is defined on the interval $t \in [0, t_{AB}]$. The time t is thus scaled into the interval $x \in [-1, 1]$:

$$x = \frac{2}{t_{AB} - 0}(t - 0) - 1 = \frac{2}{t_{AB}}t - 1 \tag{4}$$

Similarly, the position θ of the interval $\theta \in [0, \Delta\theta]$ is scaled to the interval $\phi \in [-1, 1]$:

$$\phi = \frac{2}{\theta_B - 0}(\theta - 0) - 1 = \frac{2}{\Delta\theta}\theta - 1 \tag{5}$$

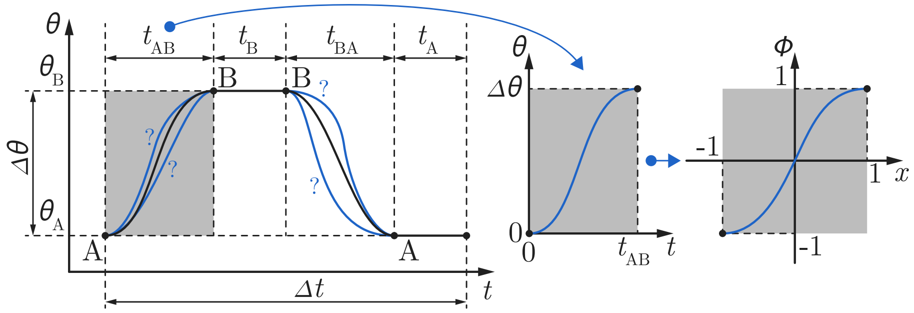

Fig. 5. PTP profile of a reciprocating mechanism with the indication of the original θ and scaled ϕ forward profile from A to B.

The scaled profile $\phi(x)$ can now be described as the sum of a product of coefficients p_i and Chebychev polynomials $T_i(x)$ in (3):

$$\phi(x) = \sum_{i=0}^{n} p_i T_i(x) \qquad x \in [\text{-}1,\ 1] \tag{6}$$

Previous research in [11] concludes that a degree of $n = 7$ in (6) is a good trade-off between calculation time and reduction of RMS-torque. Therefore, this paper continues with this 7th-degree Chebychev polynomial, hereafter called the *cheb7*-profile. The *cheb7*-profile exists of coefficients $p_0, \dots, p_7$ of which their values determine how much each of the polynomials affect the final shape of the motion profile. The values must take the motion constraints in points A and B into account. Besides the obvious fact that the machine starts at A and ends at B, the velocity $\dot{\phi}$ and acceleration $\ddot{\phi}$ are set to 0 in these points:

$$\begin{matrix} \phi(-1) = -1 & \dot{\phi}(-1) = 0 & \ddot{\phi}(-1) = 0 \\ \phi(1) = 1 & \dot{\phi}(1) = 0 & \ddot{\phi}(1) = 0 \end{matrix} \tag{7}$$

Referring to (6), and by incorporating the constraints in (7), it is clear that the lower degree coefficients $p_0, \dots, p_5$ can be written as a function of the remaining coefficients p_6, p_7:

$$\begin{cases} p_5 = -5p_7 + {}^{3}/_{128} \\ p_4 = -6p_6 \\ p_3 = 9p_7 - {}^{25}/_{128} \\ p_2 = 15p_6 \\ p_1 = -5p_7 + {}^{75}/_{64} \\ p_0 = -10p_6 \end{cases} \tag{8}$$

In other words, the coefficients p_6, p_7 are free to be optimized, while the motion constraints are met by using (8). The derivation of the relation between p_6, p_7 and $p_0, \dots, p_5$ is given in Appendix A. Note that setting $p_6, p_7 = 0$ results in a non-optimized 5th-degree Chebychev polynomial. This polynomial is further referred to as the *cheb5*-profile.

2.3. Objective function

Through the motion equations in (1), the torque T can be calculated for any given motion profile θ. Yet, accurate values of the stiffness k and damping b of the equivalent spring–damper coupling must be obtained. However, as concluded with Fig. 4, this complex and time-consuming task is avoided by keeping the frequency content of the torque T in a region where the two-mass system behaves as a one-mass system [9]. This system model assumes a rigidly connected inertial load $J = J_r + J_l$ and thus excludes the dynamics of the spring–damper coupling. Moreover, recent work [5] shows that reductions of the RMS-torque T_{RMS} up to −38% are possible by using this simplified model. Substitution of $\theta = \theta_l$ in (1), leads to the motion equation of the one-mass system model:

$$T = T_l(\theta) + \frac{1}{2}\frac{\mathrm{d}J(\theta)}{\mathrm{d}\theta}\dot{\theta}^2 + J(\theta)\ddot{\theta} \tag{9}$$

Note that (9) is a non-linear second order differential equation with varying coefficients that depend on the position θ. Solving the equation numerically for achieving a minimum RMS-torque leads to a highly complex mathematical problem, which favors the use of direct methods [15].

The accuracy of the one-mass system model is verified in Fig. 6 where the measured torque for the forward movement of an arbitrary motion profile, the *cheb5*-profile in this case, is compared with the calculated torque according to (9). A clear correlation is found, which means that the model is valid for minimizing the RMS-torque. In addition, both the calculated torque and load position θ_l according to the two-mass motion Eqs. (1) is plotted. Due to the small difference between the actual rotor position θ and estimated load position θ_l, the torque according to the two-mass system is nearly equal to the torque according to the one-mass

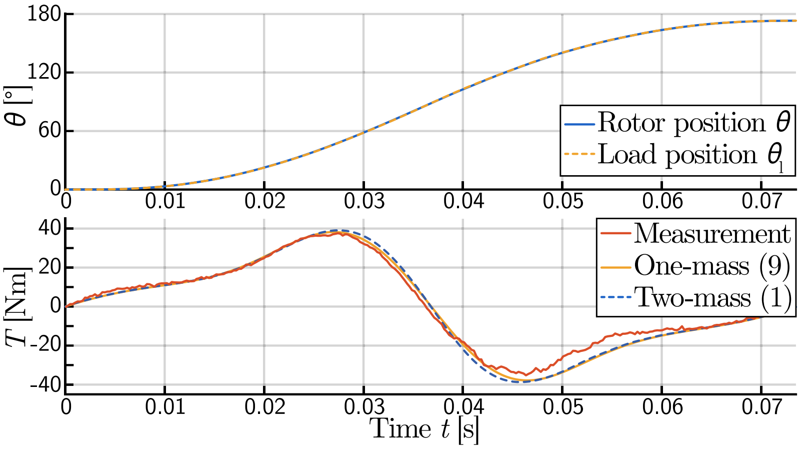

Fig. 6. Comparison of the calculated (9) and measured torque T (bottom) for an arbitrary motion profile θ and the accompanying load position θ_l (top).

system. This confirms that model reduction is suited if the controller bandwidth is sufficiently low. The control design is further discussed in Section 3.2.

After substituting the scaling laws (4) and (5) of respectively the time t to x and the position θ to ϕ in (9), the scaled torque profile with retained [11] RMS-value is obtained:

$$T = \frac{\Delta\theta}{{t_{AB}}^2}\left(\frac{\mathrm{d}J(\phi)}{\mathrm{d}\phi}\dot{\phi}^2 + 2J(\phi)\ddot{\phi}\right) \tag{10}$$

The load torque T_l in (9) is disregarded because it is negligible when the inertial forces are dominant at high speed.

By finally taking the RMS-value of (10), an analytical description of the objective function is obtained:

$$T_{\mathrm{RMS}} = \sqrt{\frac{1}{2}\int_{-1}^{1} T^2 \mathrm{d}x} \tag{11}$$

In short, the coefficients p_6, p_7 can be optimized with the objective to minimize T_{RMS}. The objective function is graphically represented in Fig. 7 for $t_{AB} = 73.75$ ms and proves the effectiveness of choosing a gradient-based algorithm as the gradients indicate a clear global optimum without any local optima.

Because there are no constraints, the Quasi-Newton method [11] is selected for on-line optimization. A condition for fast evaluation of the objective function however is that the load inertia J_l must be available as a continuous function, which is achieved in Section 4.

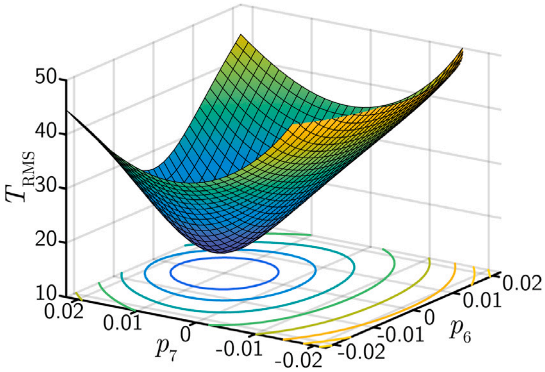

Fig. 7. Objective function of the forward *cheb7*-profile [11].

3. SDFT for inertia estimation

3.1. General principle

The general estimation principle, shown in Fig. 8, exists of two parts: calculating the modulus M, also called the gain when expressed in dB, at a specific frequency f and converting this modulus to the load inertia J_l. This frequency f at which the system's response is calculated is further called the tracking frequency f.

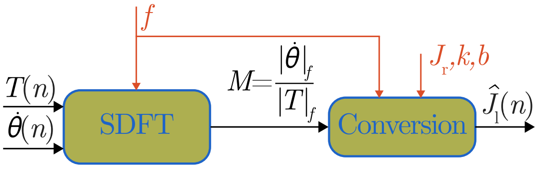

Fig. 8. General scheme for estimating the load inertia using SDFT.

The first part is tracking the modulus real-time, for which the computationally efficient SDFT algorithm is of interest. At a discrete time instance n, the algorithm uses Fourier analysis to determine the h^{th} harmonic component $X_h(n)$ based on N samples of a measured signal $x(n)$:

$$X_h(n) = (X_h(n-1) + x(n) - x(n-N))e^{jh\frac{2\pi}{N}} \tag{12}$$

The proof of (12) is found in [16,17] and its implementation is shown in Fig. 9. The observed signals are in this case the input signal, torque T, and output signal, velocity $\dot{\theta}$. For parameter estimation, only the response of the first harmonic ($h = 1$) is of interest. The window size N is thus set to one period of the tracking frequency f, with t_{s} the sample time:

$$N = \frac{1}{f t_{\mathrm{s}}} \tag{13}$$

By taking the magnitude of the Fourier components $\dot{\theta}(j\omega)$ and $T(j\omega)$, the modulus M of the system is found as:

$$M = \frac{|\dot{\theta}|_f}{|T|_f} \tag{14}$$

For the second part, the relation between the mechanical parameters of the two-mass system and the modulus M is considered. The actual modulus M is found by substitution of $s = j\omega$, with $\omega = 2\pi f$, in the transfer function (2) and calculating the magnitude:

$$M = \sqrt{\frac{b^2\omega^2 + (J_{\mathrm{l}}\omega^2 - k)^2}{(J_{\mathrm{r}} J_{\mathrm{l}}\omega^3 - k\omega J_{\mathrm{l}} - k\omega J_{\mathrm{r}})^2 + (b\omega^2 J_{\mathrm{l}} + b\omega^2 J_{\mathrm{r}})^2}} \tag{15}$$

By re-writing (15), the estimated load inertia $\hat{J}_{\mathrm{l}}$ is found as:

$$\hat{J}_{\mathrm{l}} = \begin{cases} J_{\mathrm{l}1} = \frac{A+\sqrt{B}}{C} \\ J_{\mathrm{l}2} = \frac{A-\sqrt{B}}{C} \end{cases}$$
$$A=(k-b^2)M^2 J_{\mathrm{r}}\omega^3-(M^2 J_{\mathrm{r}} k^2+1)k\omega$$
$$B=-M^4 J_{\mathrm{r}}^4 b^2\omega^8+2M^2 J_{\mathrm{r}}^2 b^2\omega^6+(M^2 b^2-1)b^2\omega^4+2M^2 b^2 k^2\omega^2+M^2 k^4 \tag{16}$$
$$C=M^2 J_{\mathrm{r}}^2\omega^5-(2M^2 J_{\mathrm{r}} k-M^2 b^2+1)\omega^3+M^2 k^2\omega$$

Note that the modulus M, and thus the estimated inertia $\hat{J}_{\mathrm{l}}$ after conversion, is inherently delayed with $N/2$ due to the lag of the SDFT-algorithm [9]. The position θ is therefore also delayed with $N/2$ so that the correct position-dependency is obtained. A second remark is that the conversion Eq. (16) consists of two solutions with a common numerator C and root function B. Section 4 demonstrates how this is taken into account.

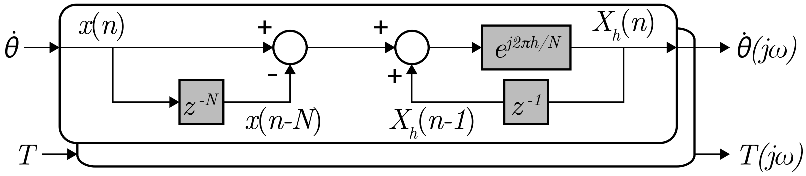

Fig. 9. Implementation of the SDFT algorithm [17].

3.2. Selection of the tracking frequency

For the selection of the tracking frequency, 3 arguments are considered, starting with the window size N of SDFT. Every N samples, an inertia estimate is obtained. As depicted in Fig. 3, the inertia is variable during this window which inherently results in an averaging error. The smaller the window size N, the smaller the difference ΔJ_{l} and the lower this error. According to (13) this means that the tracking frequency f should be selected as high as possible. On the other hand, the smaller N, the less accurate a signal at f is reconstructed using Fourier. According to the Nyquist–Shannon sampling theorem at least $N = 2$ samples are required, but the measured gain in Fig. 4 shows that at frequencies from about 100 Hz the reconstruction starts getting problematic due to the inevitable presence of noise. In short, the first guideline is to select $f \leq 100$ Hz. According to (13) with $t_s = 0.25$ ms this means a windows size of $N \geq 50$.

A second argument is to select a frequency where the actual system can be represented by the same transfer function $H(s)$ (2) of the equivalent two-mass system at all positions. Due to the varying inertia, the frequency regions where $H(s)$ is valid are dependent on the load position. Previous research in [9] covers this and further limits the selection to $80 \geq f \leq 100$ Hz. As illustrated in Fig. 4, an important remark is that the stiffness k and damping b strongly affect the gain in this frequency region. Because the values of k and b are used for converting the measured gain to the estimated load inertia $\hat{J}_l$, this means that an inaccuracy on k and b results in an estimation error on $\hat{J}_l$. This uncertainty is also handled in [9], concluding that rough guesses of k and b lead to an acceptable accuracy of the estimated load inertia profile.

The final argument for finer selection of f is to consider the control design. The block diagram of the control scheme is shown in Fig. 10, with the system $H(s)$ given in (2) for maximum load inertia, and the position controller C_p and speed controller C_s:

$$C_p = K_{pp} \qquad C_s = K_{ps}(1 + K_{is}\frac{1}{s}) \tag{17}$$

Using Mason's rule [18], three transfer functions are constructed:

$$\begin{aligned} G_1(s) &= \frac{\theta(s)}{\theta^*(s)} = \frac{C_p C_s H}{(1 + C_s H)s + C_p C_s H} \\ G_2(s) &= \frac{\dot{\theta}(s)}{\theta^*(s)} = \frac{C_p C_s H s}{(1 + C_s H)s + C_p C_s H} \\ G_3(s) &= \frac{\dot{\theta}(s)}{T_{inj}(s)} = \frac{H s}{(1 + C_s H)s + C_p C_s H} \end{aligned} \tag{18}$$

For accurate estimation, the control settings (K_{pp} = 20 1/s, K_{ps} = 1.1 Nms/rad, K_{is} = 50 1/s) are on the one hand tuned with $G_3(s)$ for maximum disturbance transmission because the injected torque $T_{inj} = A\sin(2\pi f t)$ must result in a high amplitude of both the torque T and speed $\dot{\theta}$ at the tracking frequency f. On the other hand, tuning with $G_2(s)$ is needed because the influence of the desired profile θ^* on the speed $\dot{\theta}$ must be limited to make sure that the amplitude at the tracking frequency f is strongly represented.

By assuming that both forward and backward motion profiles θ^* can be represented as a sum of sines of which the fundamental one is half a sine wave, the frequency f_{PTP} and amplitude A_{PTP} can be related to their PTP time t_{PTP} of either t_{AB} or t_{BA} and their angular difference $\Delta\theta$:

$$f_{PTP} \hat{=} \frac{1}{2t_{PTP}} \qquad A_{PTP} \hat{=} \frac{\Delta\theta}{2} \tag{19}$$

Using (19), a guideline to maximize the ratio between the response of T_{inj} on $\dot{\theta}$ through $G_3(s)$ and the response of θ^* on $\dot{\theta}$ through $G_2(s)$ can be constructed. This ratio is here-after called the disturbance-to-position ratio (DPR). Through simulations with the equivalent two-mass system model (1), it has been found that the DPR should at least be 0.05:

$$\frac{A|G_3(s)|_f}{\frac{\Delta\theta}{2}|G_2(s)|_{f\,PTP}} \geq 0.05 \tag{20}$$

The responses of (18) are plotted in Fig. 11 of which the ones related to the motion profile θ^* are plotted as a function of the PTP-time t_{PTP} by converting the frequency f_{PTP} according to (19). The gain concerning the added torque is plotted as a function of the frequency but spaced according to (13) because only a frequency f can be selected that results in an integer value of the window size N. The bandwidth (−3 dB ≈ 0.7) of the closed-loop transfer function $G_1(s)$ is depicted in red and is found to be f_{BW} = 3.6 Hz. According to (19), this corresponds to a PTP-time of $t_{BW} \hat{=} 140$ ms.

A first observation, illustrated with the red box, is that the PTP-times during operation t_{AB} = 73.75 ms ($\hat{=}$ 6.8 Hz) and t_{BA} = 57.5 ms ($\hat{=}$ 8.7 Hz) are above the bandwidth of f_{BW} = 3.6 Hz. Both the gain and phase of $G_1(s)$ show that this means that the actual position θ is not able to follow the desired position θ^*. The entire position interval is thus not completed on time and as a result, the inertia profile is not fully estimated. During estimation, the PTP-time is therefore slowed down with a factor of 10 where the gain and phase are acceptable. Moreover, the speed $\dot{\theta}$ to position θ^* gain is very high at e.g. t_{BA}. This means that for obtaining the desired DTP in (20) a very high amplitude A of the added torque should be selected, which is not efficient. By slowing down with

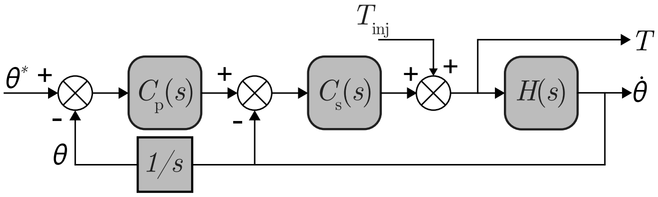

Fig. 10. Block diagram of the control scheme.

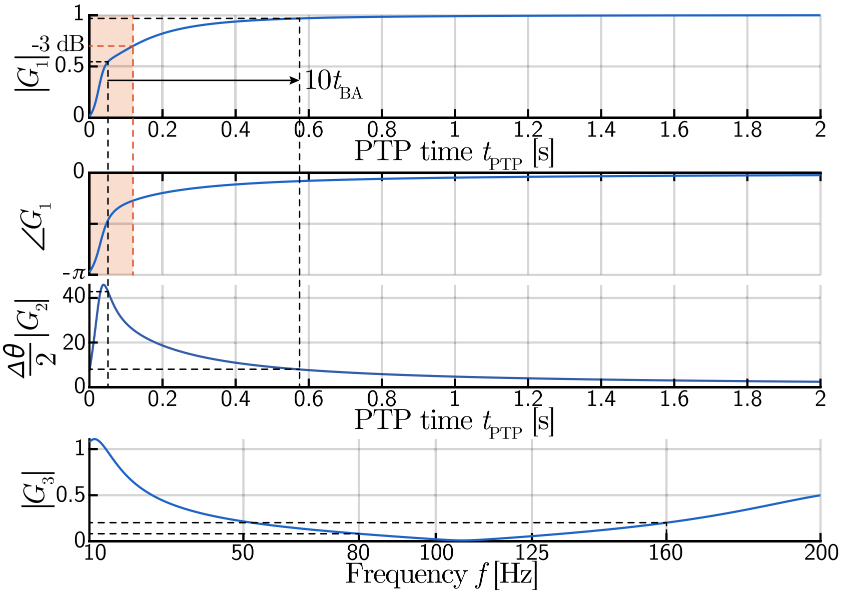

Fig. 11. Controller responses. Top to bottom: gain and phase of the desired θ to actual position θ^*; gain of desired position θ^* to actual speed $\dot{\theta}$; gain of added torque T_{inj} to actual speed $\dot{\theta}$.

the factor of 10, this gain is strongly reduced from about 40 to 8. As clarified in Section 4, slowing down is not applied during operation. Despite the low control bandwidth, accurate PTP-movement at operating speed is achieved by implementing speed - and torque feedforward [5].

Next, the torque T_{inj} to speed $\dot{\theta}$ gain confirms that the controller transmits disturbances at most of the frequencies. At the lower frequencies, the added torque is better transmitted, but these are not selected due to the mentioned averaging error. Selecting $f = 160$ Hz or $f = 200$ Hz is more feasible due to a higher gain of $G_3(s)$, but these frequencies deliver inaccurate system responses due the mentioned reconstruction problem. what remains is $f = 80$ Hz with a gain of 0.08. By considering the guideline in (20), the amplitude is set to $A = 5$ Nm resulting in a DPR-ratio of 0.052. For comparison, a less feasible tracking frequency $f = 160$ Hz is selected with an amplitude of $A = 2.5$ Nm which also satisfies the ratio (20). In short, the selected frequencies and amplitudes for the next chapters are:

$$\begin{cases} A = 5\,\text{Nm}, \; f = 80\,\text{Hz} \rightarrow N = 50 \\ A = 2.5\,\text{Nm}, \; f = 160\,\text{Hz} \rightarrow N = 25 \end{cases} \tag{21}$$

4. Implementation and validation

4.1. Principle and control sequence

By combining the GBA approach in Section 2 for optimization and the SDFT approach in Section 3 for estimation, on-line motion profile optimization for reciprocating mechanisms is established. The general principle is shown in Fig. 12, of which the function blocks are developed in Simulink and implemented on the real-time platform using Simulink code generation. Except for the profile optimization block which requires to be compiled as a Matlab function. The real-time target is a Beckhoff-PLC, configured in torque mode, with a sample time of $t_s = 0.25$ ms. The figure shows that both the estimation and optimization blocks are either enabled and/or triggered by the boolean x_{est}, which is true during estimation. Further details on the working principle are given through experimental validation in this section where the performance is evaluated with the selected tracking frequency of $f = 80$ Hz.

For validation of the developed principle, the control sequence is set as depicted in Fig. 13. During the first PTP cycle $i_{PTP} = 1$, the mechanism operates at full speed with the non-optimized *Cheb5*-profile. The second cycle $i_{PTP} = 2$ is used for estimation of the varying inertia profile and optimization of the motion profile accordingly. For accurate estimation and achieving a sufficiently large DPR, the mechanism is slowed down during the backward BA-movement. Note that the backward movement is selected for estimation, but the forward AB-movement could also be selected. The rest time in A is used for executing the optimization algorithm. Next in the third cycle $i_{PTP} = 3$, the mechanism operates back at full speed. Only now the motion profile is set to the optimized *Cheb7*-profile. The forward AB-movement and backward BA-movement of $i_{PTP} = 3$ compared to $i_{PTP} = 1$ is used for validation of the optimized motion profile.

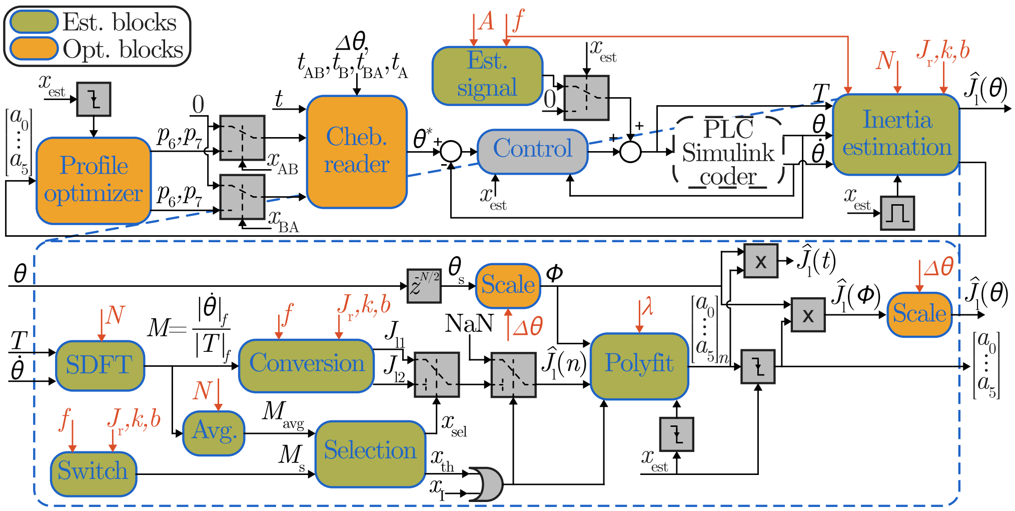

Fig. 12. General principle of on-line motion profile optimization for reciprocating mechanisms.

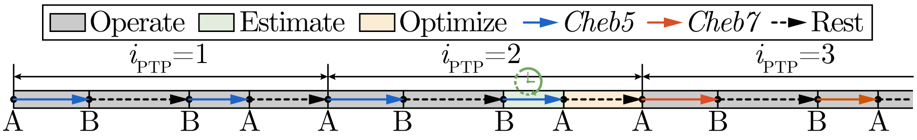

Fig. 13. Control sequence for validating on-line motion profile optimization.

4.2. Motion profile, estimation signal and controller

For the definition of the PTP profile, the following booleans are defined: x_{AB} is true during the forward movement from A to B, x_B is true during the rest time in B, x_{BA} is true during the backward movement from B to A, and x_A is true during the rest time in A. The PTP profile θ^*, plotted in Fig. 14 for the second PTP cycle $i_{PTP} = 2$, is defined according to the motion features in Section 2:

$$\theta^* = \begin{cases} Cheb, & \text{if } x_{AB} = 1 \text{ or } x_{BA} = 1 \\ \theta_A = 0, & \text{if } x_A = 1 \\ \theta_B = \Delta\theta, & \text{if } x_B = 1 \end{cases} \tag{22}$$

During the rest times, the position θ^* is set to $\theta_A = 0$ in A and to $\theta_B = \Delta\theta$ in B. During both forward and backward movement, the position θ^* is defined according to the Chebychev polynomial in (6). For the non-optimized case, or when the inertia profile is not yet estimated, the polynomial coefficients $p_0, \ldots, p_5$ are found by setting $p_6, p_7 = 0$ in (8). What leaves is the non-optimized *Cheb5*-profile. After estimation, the optimized coefficients p_6, p_7 are used, resulting in the optimized *cheb7*-profile.

As mentioned in Section 2, the Chebychev polynomials are defined in fixed intervals according to (6), with $x \in [-1, 1]$ the scaled time and $\phi \in [-1, 1]$ the scaled position. This is taken into account by implementing the scaling laws for the time t in (4) and for the position θ^* in (5).

Fig. 14 shows both the input (torque T) and output (speed $\dot\theta$) signal at $i_{PTP} = 2$. Due to the added torque which is activated when $x_{est} = 1$, both signals contain oscillations of $1/f$. This is necessary for tracking the modulus M at this selected frequency. Note that the amplitude of these oscillations is strongly reduced in the actual position θ and therefore does not cause additional estimation errors.

During operation ($x_{est} = 0$), both speed - and torque feedforward [5] are activated in the controller for achieving a low tracking error at full operating speed. The forward part ($x_{AB} = 1$) in Fig. 14 confirms the effectiveness of this smart feedforward structure as no residual vibrations occur.

During estimation ($x_{est} = 1$), the feedforward is deactivated so that the disturbance transmitting behavior of the controller is maintained. As a result, it takes some settling time t_{set} to reach the end position B. The end of the estimation interval ($x_{est} = 1$) is defined accordingly. Also, note that x_{est} starts one SDFT period $1/f$ before x_{BA}. This is to make sure that the SDFT-window is filled with relevant data before the backward motion starts.

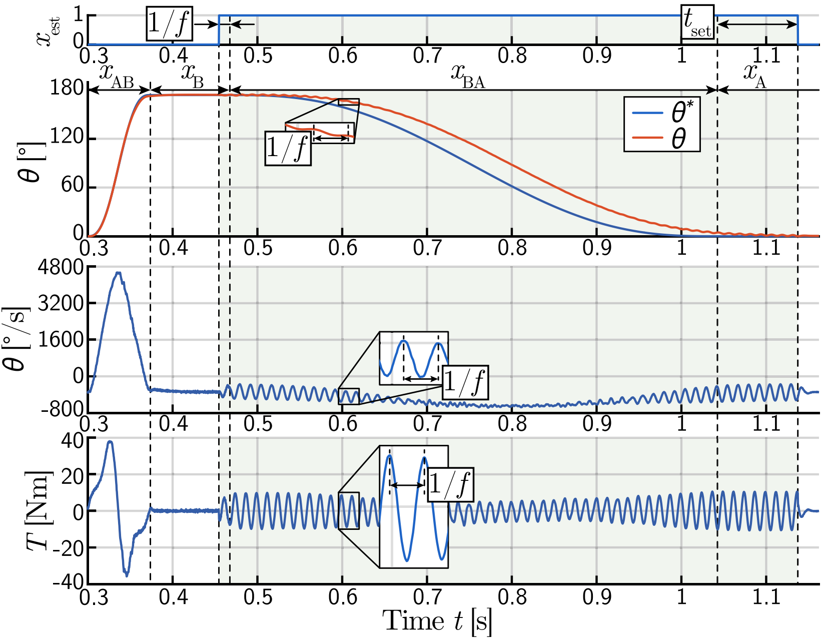

Fig. 14. Desired θ^* and actual position θ, actual speed $\dot{\theta}$ and motor torque T at $i_{\text{PTP}} = 2$ with an injected sine wave of $f = 80$ Hz and $A = 5$ Nm. The estimation interval ($x_{\text{est}} = 1$) is depicted in green.

4.3. Solution switching

As clarified in Section 3, the SDFT-algorithm tracks the modulus M at the tracking frequency f. The conversion (16) from M to an inertia estimate $\hat{J}_l$ however consists of two solutions J_{l1}, J_{l2}, which are plotted in Fig. 15 for $f = 80$ Hz and $f = 160$ Hz. In the case of $f = 80$ Hz, J_{l2} does not exist in the relevant domain of the load inertia of the system, which means that the first solution J_{l1} is always the correct one. In the other case of $f = 160$ Hz, both solutions exist as depicted for $M = 0.2$ for example. Only one is correct, requiring the implementation of a selection algorithm. The plot of $f = 160$ Hz shows a clear minimum where the solution switches. Or in other words, the solution switches when the slope crosses zero. Solving $\partial M / \partial J_l = 0$ for J_l leads to the switching function J_{ls}:

$$J_{ls} = \frac{A + D\sqrt{B}}{C} \qquad \begin{aligned} &A{=}{-}2\omega^2{+}b^2\omega^2{+}k^2 \\ &D{=}b^2\omega^2{+}k^2 \\ &B{=}4J_r^2b^2\omega^6{+}b^4\omega^4{+}2b^2k^2\omega^2{+}k^4 \\ &C{=}2\omega^2(J_rb^2\omega^4{-}J_rk^2\omega^2{+}b^2k\omega^2{+}k^3) \end{aligned} \tag{23}$$

Substitution of (23) in the modulus function (15), leads to the switching function M_s. The switching point $(J_{ls}; M_s)$ is depicted in Fig. 15. How the switching algorithm is implemented in the case of $f = 160$ Hz is explained in Section 5.

In the case of $f = 80$ Hz, Fig. 16 confirms that no switching occurs as the measured modulus M never crosses the switching value M_s. As a result, the first solution J_{l1} is selected ($x_{\text{sel}} = 0$ in the general scheme 12) as the discrete estimate $\hat{J}_l(n)$ of the inertia. The figure further shows that the selection algorithm takes into account that during the first SDFT-window, the modulus M is inaccurate. During the first two windows, a boolean x_I is set to true and cancels the inertia estimates by returning NaN (Not a number). What leaves is a clear agreement between the discrete estimates $\hat{J}_l(n)$ and the true inertia $\bar{J}_l$.

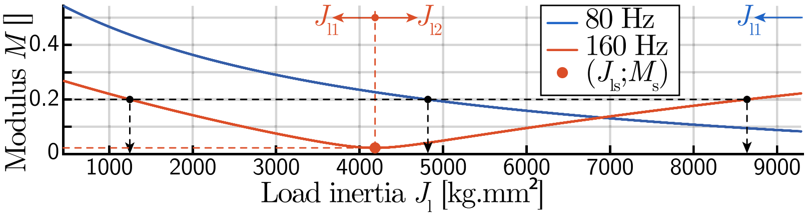

Fig. 15. Modulus M as a function of load inertia J_l according to (15) for $f = 80$ Hz and $f = 160$ Hz.

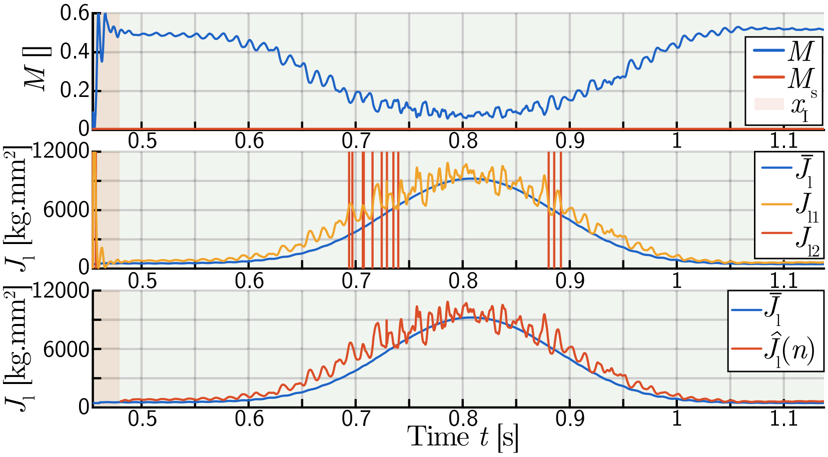

Fig. 16. Obtaining a discrete inertia estimate during the estimation interval ($i_{\mathrm{PTP}} = 2$ & $x_{\mathrm{est}} = 1$). Top to bottom: modulus M and switching value M_s; double solutions J_{l1}, J_{l2}; selected inertia estimate $\hat{J}_l(n)$.

4.4. Recursive polynomial fitting

Using the discrete estimates $\hat{J}_l(n)$ of the inertia for solving the objective function (11) in Section 2 can lead to unfeasible results due to the noise present in the signal. More accurate and faster evaluation is possible with a continuous function of the inertia profile. This function is obtained by fitting the discrete estimates $\hat{J}_l(n)$ with a 5th-degree polynomial $\hat{J}_l(\phi)$ with respect to the shifted and scaled (see (5)) position ϕ:

$$\hat{J}_l(\phi) = a_0 + a_1\phi + a_2\phi^2 + a_3\phi^3 + a_4\phi^4 + a_5\phi^5 \tag{24}$$

The degree of 5 is chosen from a trade-off between computational efficiency and accuracy. By re-writing (24), it is clear that the recursive least-squares (RLS) algorithm is suited for obtaining $\hat{J}_l(\phi)$:

$$\hat{J}_l(\phi) = \varphi^T\Theta \qquad \varphi = \begin{bmatrix}\phi^5\\ \phi^4\\ \phi^3\\ \phi^2\\ \phi\\ 1\end{bmatrix} \qquad \Theta = \begin{bmatrix}a_5\\ a_4\\ a_3\\ a_2\\ a_1\\ a_0\end{bmatrix} \tag{25}$$

In (25), φ is the vector of regressors, Θ is the parameter vector, and $\hat{J}_l(\phi)$ is the estimated output of the general linear regression model [19]. The parameters Θ are estimated recursively with the objective to minimize the error between the discrete output $\hat{J}_l(n)$ and the estimated output $\hat{J}_l(\phi)$. The most important setting of the RLS algorithm is the forgetting factor. Setting $\lambda = 1$ means that all data of the true output is equally important and setting $\lambda < 1$ implies that past data is less significant. Because the fitting is only relevant inside the estimation interval ($x_{\mathrm{est}} = 1$) which mainly consists of the backward motion with N_{BA} samples, the forgetting factor λ is set to this interval [20]:

$$\lambda = 1 - \frac{1}{N_{\mathrm{BA}}} \qquad N_{\mathrm{BA}} = \frac{10t_{\mathrm{BA}}}{t_s} \tag{26}$$

As shown in Fig. 17, the coefficients $[a_0, \dots, a_5]_n$ converge towards their final values $[a_0, \dots, a_5]$ at the end of the backward motion. The initial values, at the first cycle $i_{\mathrm{PTP}} = 1$ are set to $[a_0, \dots, a_5] = 0$, which means that no initial guess is necessary. From the second cycle $i_{\mathrm{PTP}} \geq 2$, the final values $[a_0, \dots, a_5]$ of the previous cycle are set to the new initial values of the next cycle.

What cannot be seen in the figure but is indicated in the general scheme in Fig. 12, is that the samples used for fitting are deleted at every negative flank of x_{est}. This prevents samples of a previous cycle from affecting the estimated inertia polynomial of the current cycle. Through this reset, the estimation algorithm can capture a change of inertia from the first cycle after the change occurs. This benefit is demonstrated in Section 5.

After substitution of the coefficients $[a_0, \dots, a_5]$ in (24) and re-scaling the position from ϕ to θ (see (5)), the estimated continuous inertia profile $\hat{J}_l(\theta)$ is finally obtained. By comparing with the true inertia $\bar{J}_l$ in Fig. 18, the performance of the fitting algorithm is evaluated. A clear agreement with an RMS-error ϵ of about 0.03 is found, but a first observation is that the rising flank from B to A is less accurate than the falling flank. This is an unavoidable consequence of the real-time nature of the estimator. Due to the forgetting factor λ, the oldest samples which are in this case the ones at the rising flank from B to A, are less important than the newest samples.

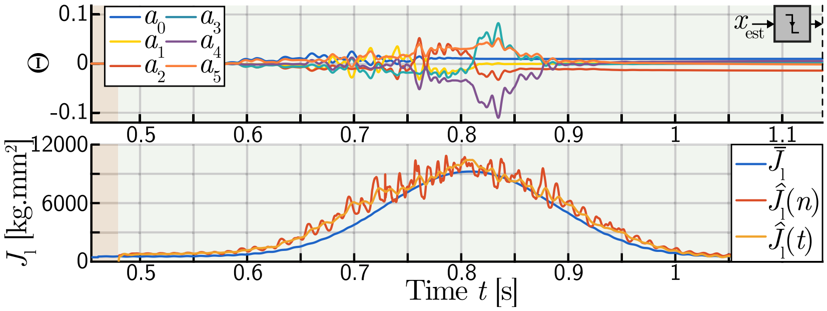

Fig. 17. Obtaining a continuous inertia estimate during the estimation interval (i_{PTP} = 2 & x_{est} = 1). Polynomial coefficients $[a_0, \dots, a_5]$ (top); discrete $\hat{J}_1(n)$ and continuous inertia $\hat{J}_1(t)$ estimate (bottom).

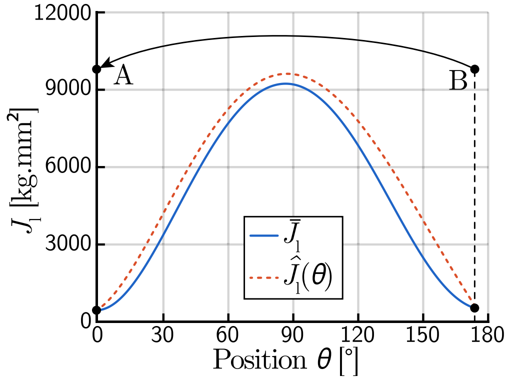

Fig. 18. Estimated inertia profile $\hat{J}_1(\theta)$ compared with the true inertia $\bar{J}_1$.

4.5. Updating the motion profile

Since a continuous function $\hat{J}_1(\theta)$ of the inertia profile is estimated in $i_{PTP} = 2$, the on-line motion profile optimizer can be executed. At the negative flank of x_{est}, the objective function for the forward movement (11) and backward movement (38) is solved according to the coefficients $[a_0, \dots, a_5]$ of the estimated inertia polynomial. As depicted in the control sequence in Fig. 13, the rest time $t_A = 75$ ms after estimation is used as calculation time for the optimizer.

In the next cycle $i_{PTP} = 3$, both forward and backward motion profiles are defined as the optimized *Cheb7*-profile with optimized values p_6, p_7. A comparison is made in Fig. 19, where both the torque and motion profiles during the optimized cycle $i_{PTP} = 3$ and the non-optimized cycle $i_{PTP} = 1$ ($p_6, p_7 = 0$) are plotted together. A clear reduction of the motor torque T is achieved. For instance, during the forward movement the RMS-value T_{RMS} is reduced from 21.3 Nm to 13.8 Nm and the maximum value T_{max} from 38.1 Nm to 21 Nm. And during backward movement, motor saturation is avoided. Table 1 summarizes and compares the reductions with the off-line optimized forward profile in [11]. The backward profile is not considered in [11].

Based on the table, the proposed on-line motion profile optimization technique is proven to be effective and a valid alternative for off-line techniques where the inertia profile is required to be known in advance. The main added value however is that the on-line approach can cope with changing inertia due to process flexibility, which is demonstrated in Section 5.

Fig. 19 furthermore confirms that the controller is well designed because at the full operating speed of 0.3 s per cycle, the tracking error $\theta^* - \theta$ is small. This is mainly thanks to the smart feedforward structures in [5] and avoids residual vibrations.

Table 1
Measured torque reductions.

	on-line		off-line [11]	
	A to B	B to A	A to B	B to A
T_{RMS}	−35.2%	−28.2%	−38.6%	/
T_{max}	−44.9%	−16.1%	−43.6%	/

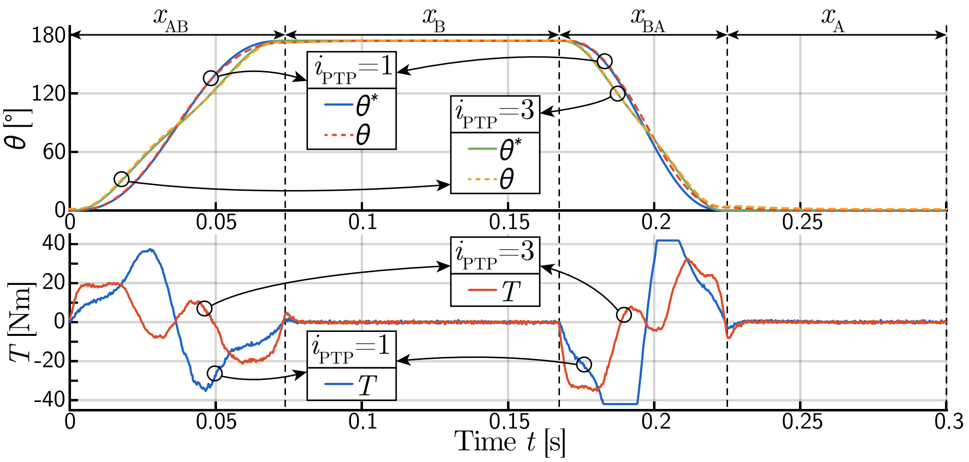

Fig. 19. Top: position setpoint θ^* at i_{PTP} = 1 and 2; bottom: motor torque T at i_{PTP} = 1 and 2.

5. Additional features

5.1. Doubling the tracking frequency

The performance of the developed algorithms is further validated by doubling the tracking frequency to f = 160 Hz. According to the guidelines in Section 3, selecting f = 160 Hz has two advantages: a smaller window size N results in a lower averaging error; a higher disturbance gain requires a lower amplitude A = 2.5 Nm of the injected torque for respecting the minimum DPR-ratio of 0.05. The main disadvantage however becomes clear with the experiment in Fig. 20. The lower N, the less accurate the fundamental signal is reproduced using Fourier, resulting in an inaccurate modulus M. This is also clearly visible in the off-line determined gain characteristic in Fig. 4. As a consequence, the solutions J_{l1} and J_{l2} are not reliable. Valid inertia estimates are however obtained at the positions of low inertia where the influence of noise is small [4], resulting in an accurate modulus M. Yet, the full inertia profile must be obtained. In other words, the real-time inertia estimator fails when the guideline of selecting a window size $N \geq 50$ is not respected. The solution is to lower the sample time t_s and thus achieve more samples in the SDFT-window N, but the real-time platform is already configured at its limit of t_s = 0.25 ms.

Despite the limit of the real-time target, the performance of the real-time estimator at doubled frequency is still demonstrated by using the equivalent two-mass system model instead of the actual machine. The PLC-block in Fig. 12 is now replaced with the motion equations in (1) and the tracking frequency is set to f = 160 Hz with an amplitude of A = 2.5 Nm. The principle is shown in Fig. 21 and is very similar to f = 80 Hz, except solution switching between the discrete estimates J_{l1} and J_{l2} is now required.

The basic concept is to switch from solution when the measured modulus M equals the switching value M_s. Fig. 21 illustrates this, but the zoom-in plot reveals that M equals M_s multiple times. Jitter is avoided by defining a threshold boolean x_{th}, which is true whenever $M_{avg} \leq M_{th}$. Wherein M_{avg} is the moving average [21] with window size N and $M_{th} = pM$ is the threshold value. In this case, $p = 2.2$ is found to be feasible through try and error. The selection boolean x_{sel} becomes true at a falling edge of x_{th} and

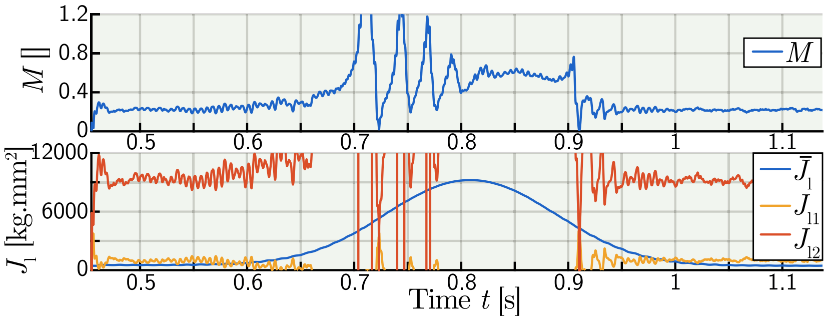

Fig. 20. Obtaining a discrete inertia estimate in the case of f = 160 Hz (i_{PTP} = 2 & x_{est} = 1). Modulus M (top) and double solutions J_{l1}, J_{l2} (bottom).

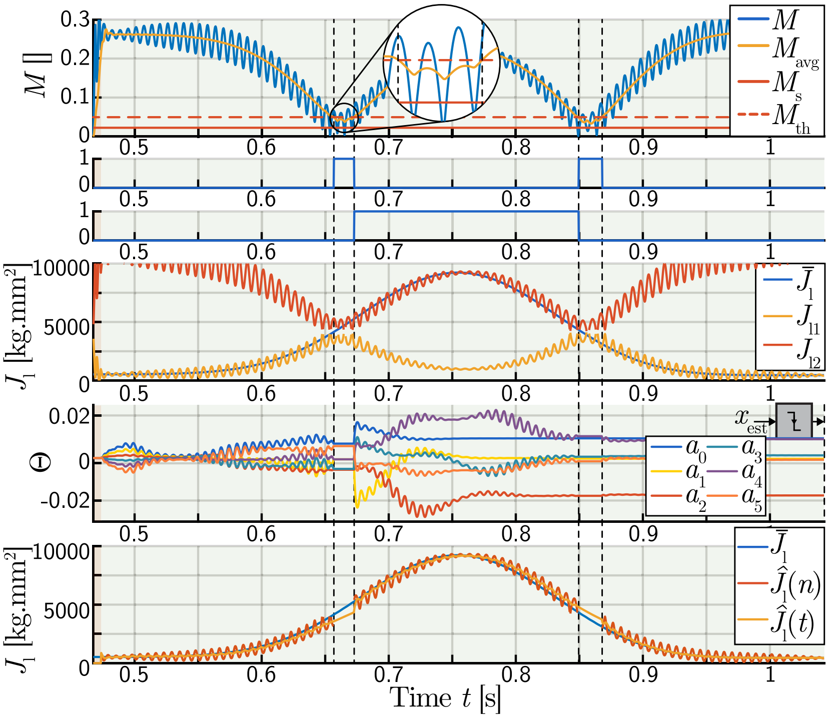

Fig. 21. Inertia estimation during i_{PTP} = 2 with f = 160 Hz.

back to false at a rising edge of x_{th}. In short, the selection algorithm is defined as:

$$J_l = \begin{cases} J_{l1}, \text{ if } x_{sel} = 0 \\ J_{l2}, \text{ if } x_{sel} = 1 \\ \nexists, \text{ if } x_{th} = 1 \end{cases} \tag{27}$$

Note that when x_{th} = 1, the estimate is defined as nonexistent by assigning NaN (not a number). Fig. 21 confirms that the selection algorithm properly selects the correct estimates of J_{l1} and J_{l1}. Furthermore, the fitting algorithm successfully cancels the false estimates by only using the samples of $\hat{J}_l(n)$ when x_{th} = 0. The polynomial coefficients $[a_0, \dots, a_5]_n$ are held constant when x_{th} = 1. In short, the simulations prove that the real-time estimation algorithm is also applicable for higher frequencies where solution switching is necessary. By comparing with the measurements in Fig. 20, it can be concluded that in noise-free conditions the SDFT is a valid approach for real-time inertia estimation. When noise is present, the SDFT-window N must be sufficiently large for canceling the noise.

The estimation accuracy is further investigated with Fig. 22 by comparing the discrete inertia estimates $\hat{J}_l(n)$ at both f = 80 Hz (N = 50) and f = 160 Hz (N = 25). The zoom-in plot, located at a high slope of the inertia $\bar{J}_l$, clarifies the influence of the window size N on the averaging error. The larger N, the larger the difference ΔJ_l and the higher the error. The consequence is that larger oscillations occur around the true inertia in the case of f = 80 Hz. The advantage of choosing the doubled frequency f = 160 Hz is however limited to the regions with a high inertia slope because, at nearly constant inertia slopes, the oscillations are rather the same as the window size N does not affect the difference ΔJ_l.

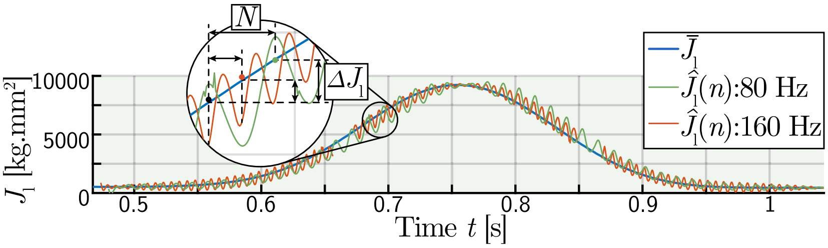

Fig. 22. Influence of the window size N on the averaging error.

Moreover, Fig. 23 shows that after polynomial fitting, the estimated inertia profile $\hat{J}_1(\theta)$ is equally accurate in both cases. In other words, the oscillations due to the averaging error are successfully canceled. Based on these simulations, the argument for selecting the tracking frequency as high as possible for limiting the averaging error thus expires if the objective is to use the inertia estimate for on-line motion profile optimization. The argument of the lower amplitude $A = 2.5$ Nm at 160 Hz instead of $A = 5$ Nm at 80 Hz of the injected torque for respecting the minimum DPR of 0.05 however remains.

In summary, the developed real-time inertia estimation algorithm is proven to be suited for higher frequencies where solution switching is required. Doubling the tracking frequency results in more accurate discrete inertia estimates. But after polynomial fitting, the obtained inertia profile is equally accurate as at lower frequencies.

Additionally, the estimated inertia profile $\hat{J}_1(\theta)$ at 80 Hz in Fig. 23 obtained through simulations is compared with the estimated profile obtained through experimental validation on the actual machine in Fig. 18. As expected, the simulations provide a higher accuracy. The main reason is the inevitable presence of noise in the actual system. Next to that, the mismatch between the observed one-mass system model and the actual system is larger than in the case of the two-mass system used for simulation.

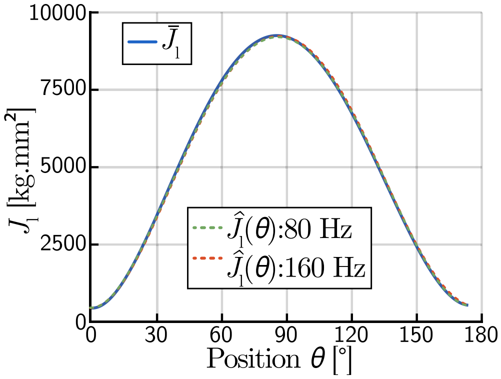

Fig. 23. Estimated inertia profile $\hat{J}_1(\theta)$ in the case of both $f = 80$ Hz and $f = 160$ Hz.

5.2. Mass change during operation

The use of the two-mass system model also enables to mimic a mass change during operation because the actual machine not yet allows objects to be picked. The control sequence is slightly changed to the one depicted in Fig. 24. A simulation is performed consisting of three PTP cycles. In the first cycle $i_{\mathrm{PTP}} = 1$, the forward profile moves according to the non-optimized *cheb5*-profile because the inertia is not yet known. As plotted in Fig. 25, the coefficients p_6, p_7 are set to 0 at $i_{\mathrm{PTP}} = 1$. In the same cycle, the inertia $\hat{J}_1(\theta)$ is estimated during the backward movement and the objective function is solved accordingly for achieving the optimized coefficients p_6, p_7. The forward profile in the second cycle $i_{\mathrm{PTP}} = 2$ is then set to the optimized *cheb7*-profile with values of p_6, p_7 plotted in Fig. 25. Just after the forward movement of $i_{\mathrm{PTP}} = 2$, a mass change occurs as depicted with a green arrow in Fig. 24. At this moment, the mass of the sledge, depicted in Figs. 1 and 2, of the pick and place unit is increased by 20%. In order to capture this mass change, which leads to an inertia change, a new estimation $\hat{J}_1(\theta)$ occurs during the backward movement. The objective function is again solved, resulting in a new set of optimized coefficients p_6, p_7. The motion profiles in the third cycle $i_{\mathrm{PTP}} = 3$ are then updated accordingly.

The achieved torque reductions during both the forward movement of the second cycle $i_{\mathrm{PTP}} = 2$ (before mass change) and during the forward movement of the third cycle $i_{\mathrm{PTP}} = 3$ (after mass change) are compared in Fig. 25. The plot of the percentage reductions δ confirms that because the inertia estimator can accurately handle mass changes, the optimization algorithm guarantees optimal performance. Similar reductions of $\delta_{\mathrm{RMS}} = -39\%$ and $\delta_{\max} = -46\%$ as depicted in Fig. 19 are achieved from $i_{\mathrm{PTP}} = 1$ to 2. And from $i_{\mathrm{PTP}} = 2$ to 3, or after the change of mass, the torque reductions δ are similar. The reductions even increase slightly which is expected as the higher the inertia variation, the higher the optimization potential.

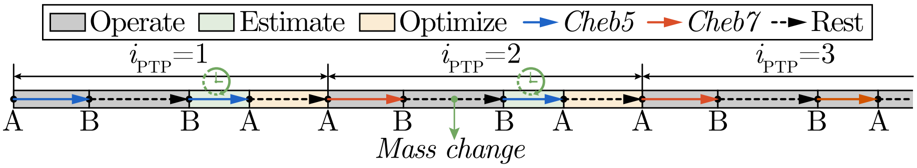

Fig. 24. Control sequence for validating on-line motion profile optimization including a mass change.

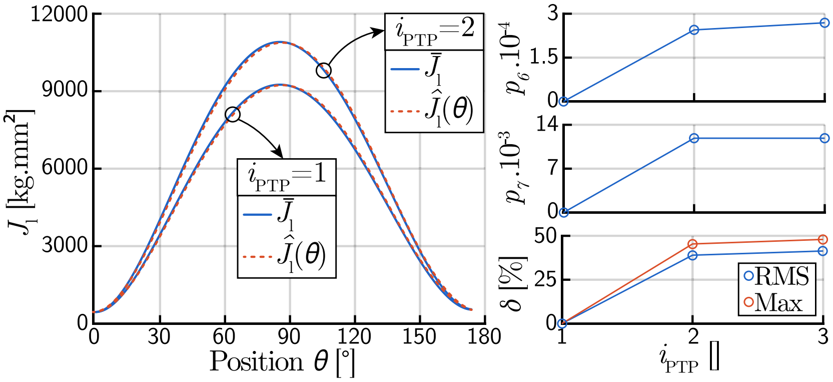

Fig. 25. Left: estimated profiles $\hat{J}_1(\theta)$ for a mass change between i_{PTP} = 1 & 2. Right: optimized coefficients p_6, p_7 and reduction δ of RMS - and maximum torque during i_{PTP} = 1–3.

6. Conclusion

This paper presented an on-line approach for motion profile optimization applied to multi-body mechanisms, driven as a PTP application, which are inherently characterized by rapidly varying load inertia. Through experimental validation on an industrial pick and place unit, the combination of using the sliding discrete Fourier transform for real-time estimation and a Gradient-based algorithm for on-line optimization is proven to be effective and a valid alternative for off-line approaches.

The developed real-time estimation algorithm can accurately estimate a continuous function of the varying load inertia during either the forward or backward movement. Additionally, the algorithm can successfully capture mass and/or inertia changes. This is demonstrated by injecting an added mass that corresponds to an object that is picked by the pick and place unit.

Tuning rules for the tracking frequency and amplitude of the estimation signal are discussed and are mainly based on the disturbance transmission of the controller. A drawback however is that the PTP motion must be slowed down, which hampers estimation at full operating speed. This is however not an issue in practice as the optimization routine can either be executed at fixed time stamps or when an increase of e.g. the RMS-value of the motor torque is detected.

Regarding the optimization part, the gradient-based approach with the motion profile described as a Chebychev polynomial is found to be a significant progress in terms of calculation time, enabling on-line optimization. The achieved torque reductions are comparable with off-line algorithms and due to the ability to capture changing mass real-time, the on-line approach is certainly an added value for maintaining optimal performance.

Additionally, the developed estimation and optimization algorithms are based on input/output-data which is already available in the controller, requiring no additional hardware.

Declaration of competing interest

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

Appendix A. Relation of coefficients of the forward motion profile

The *Cheb7*-profile is described according to (6) with $n = 7$:

$$\phi = p_0T_0 + p_1T_1 + p_2T_2 + p_3T_3 + p_4T_4 + p_5T_5 + p_6T_6 + p_7T_7 \tag{28}$$

The first 8 Chebychev polynomials $T_0, \ldots, T_7$ of the first kind are defined according to (3) [14]:

$$\begin{aligned}
T_0 &= 1\\
T_1 &= x\\
T_2 &= 2x^2 - 1\\
T_3 &= 4x^3 - 3x\\
T_4 &= 8x^4 - 8x^2 + 1\\
T_5 &= 16x^5 - 20x^3 + 5x\\
T_6 &= 32x^6 - 48x^4 + 18x^2 - 1\\
T_7 &= 64x^7 - 112x^5 + 56x^3 - 7x
\end{aligned} \tag{29}$$

Substituting (29) in (28) and rearranging leads to:

$$\phi = \begin{bmatrix} 64p_7 \\ 32p_6 \\ -112p_7 + 16p_5 \\ -48p_6 + 8p_4 \\ 56p_7 - 20p_5 + 4p_3 \\ 18p_6 - 8p_4 + 2p_2 \\ -7p_7 + 5p_5 - 3p_3 + p_1 \\ -p_6 + p_4 - p_2 + p_0 \end{bmatrix} \begin{bmatrix} x^7 \\ x^6 \\ x^5 \\ x^4 \\ x^3 \\ x^2 \\ x \\ 1 \end{bmatrix}^T \tag{30}$$

Differentiation of the motion profile ϕ in (30) with respect to x leads to the velocity $\dot{\phi}$ and acceleration $\ddot{\phi}$:

$$\dot{\phi} = \begin{bmatrix} 448p_7 \\ 192p_6 \\ -560p_7 + 80p_5 \\ -192p_6 + 32p_4 \\ 168p_7 - 60p_5 + 12p_3 \\ 36p_6 - 16p_4 + 4p_2 \\ -7p_7 + 5p_5 - 3p_3 + p_1 \end{bmatrix} \begin{bmatrix} x^6 \\ x^5 \\ x^4 \\ x^3 \\ x^2 \\ x \\ 1 \end{bmatrix}^T \tag{31}$$

$$\ddot{\phi} = \begin{bmatrix} 2688p_7 \\ 960p_6 \\ -2240p_7 + 320p_5 \\ -576p_6 + 96p_4 \\ 336p_7 - 120p_5 + 24p_3 \\ 36p_6 - 16p_4 + 4p_2 \end{bmatrix} \begin{bmatrix} x^5 \\ x^4 \\ x^3 \\ x^2 \\ x \\ 1 \end{bmatrix}^T \tag{32}$$

From this step, either the forward or backward movement is considered for determination of the coefficients of the *Cheb7*-profile. In the case of the forward profile, the motion constraints (7) are substituted in (30), (31) and (32), leading to a set of 6 equations:

$$\begin{cases} -1 = -p_7 + p_6 - p_5 + p_4 - p_3 + p_2 - p_1 + p_0 \\ 1 = p_7 + p_6 + p_5 + p_4 + p_3 + p_2 + p_1 + p_0 \\ 0 = 49p_7 - 36p_6 + 25p_5 - 16p_4 + 9p_3 - 4p_2 + p_1 \\ 0 = 49p_7 + 36p_6 + 25p_5 + 16p_4 + 9p_3 + 4p_2 + p_1 \\ 0 = -784p_7 + 420p_6 - 200p_5 + 80p_4 - 24p_3 + 4p_2 \\ 0 = 784p_7 + 420p_6 + 200p_5 + 80p_4 + 24p_3 + 4p_2 \end{cases} \tag{33}$$

The lower degree coefficients $p_0, \dots, p_5$ in (33) are considered to be the unknowns variables and the optimizable coefficients p_6, p_7 are considered to be the known variables. After rewriting of (33) a set of 6 equations with 6 unknowns remains:

$$\begin{bmatrix} 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 25 & -16 & 9 & -4 & 1 & 0 \\ 25 & 16 & 9 & 4 & 1 & 0 \\ -50 & 20 & -6 & 1 & 0 & 0 \\ 50 & 20 & 6 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_5 \\ p_4 \\ p_3 \\ p_2 \\ p_1 \\ p_0 \end{bmatrix} = \begin{bmatrix} 1 - p_7 + p_6 \\ 1 - p_7 - p_6 \\ -49p_7 + 36p_6 \\ -49p_7 - 36p_6 \\ 196p_7 - 105p_6 \\ -196p_7 - 105p_6 \end{bmatrix} \tag{34}$$

Solving the set of linear equations in (34) results in the relation between p_6, p_7 and $p_0, \dots, p_5$, given in (8).

Appendix B. Motion profile and objective function during backward movement

Brief summary:

- Scaling $x \to t$ and $\phi \to \theta$:

$$x = \frac{2}{t_{\mathrm{BA}}} t - 1 \qquad \phi = \frac{2}{\Delta\theta}\theta - 1 \tag{35}$$

- Motion constraints in A and B:

$$\begin{matrix} \phi(-1) = 1 & \dot{\phi}(-1) = 0 & \ddot{\phi}(-1) = 0 \\ \phi(1) = -1 & \dot{\phi}(1) = 0 & \ddot{\phi}(1) = 0 \end{matrix} \tag{36}$$

- Relation of coefficients:

$$\begin{cases} p_5 = -5p_7 - {}^{3}/_{128} \\ p_4 = -6p_6 \\ p_3 = 9p_7 + {}^{25}/_{128} \\ p_2 = 15p_6 \\ p_1 = -5p_7 - {}^{75}/_{64} \\ p_0 = -10p_6 \end{cases} \tag{37}$$

- Objective function:

$$T = \frac{\Delta\theta}{{t_{\mathrm{BA}}}^2}\left(\frac{\mathrm{d}J}{\mathrm{d}\phi}\dot{\phi}^2 + 2J\ddot{\phi}\right) \qquad T_{\mathrm{RMS}} = \sqrt{\frac{1}{2}\int_{-1}^{1} T^2 \mathrm{d}x} \tag{38}$$

References

[1] J. Van Gerwen, Electronic camming and gearing, Assem. Autom. 19 (1999).

[2] G. Schuh, C. Dölle, S. Barg, M. Kuhn, S. Breunig, Efficient modular product platform design of mechatronic systems, in: IEEE International Conference on Industrial Engineering and Engineering Management, 2018, pp. 1391–1395.

[3] P. Waide, C.U. Brunner, Energy-efficiency policy opportunities for electric motor-driven systems, IEA Energy Pap. (2011).

[4] F. Vanbecelaere, S. Derammelaere, J. Knockaert, K. Stockman, M. Monte, Identification of dynamic systems with position dependent load parameters, in: IEEE 7th International Conference on Control, Mechatronics and Automation, 2019.

[5] N. Van Oosterwyck, F. Vanbecelaere, M. Haemers, D. Ceulemans, K. Stockman, S. Derammelaere, CAD Enabled trajectory optimization and accurate motion control for repetitive tasks, 15th IEEE International Conference on Control and Automation, 2019.

[6] G. Berselli, F. Balugani, M. Pellicciari, M. Gadaleta, Energy-optimal motions for servo-systems: A comparison of spline interpolants and performance indexes using a CAD-based approach, Robot. Comput.-Integr. Manuf. 40 (2016) 55–65.

[7] R. Saidur, A review on electrical motors energy use and energy savings, Elsevier Renew. Sustain. Energy Rev. 14 (2010) 877–898.

[8] M. Pellicciari, G. Berselli, F. Balugani, M. Gadaleta, Increasing position accuracy and energy efficiency of servo-actuated mechanisms, in: 2015 IEEE International Conference on Automation Science and Engineering, CASE, 2015, pp. 1339–1344.

[9] F. Vanbecelaere, S. Derammelaere, N. Nevaranta, J.D. Viaene, F. Verbelen, K. Stockman, M. Monte, Online tracking of varying inertia using a SDFT approach, Mechatronics 68 (2020).

[10] Y.L. Hsu, M.S. Huang, R.F. Fung, Energy-saving trajectory planning for a toggle mechanism driven by a PMSM, Mechatronics 24 (2014) 23–31.

[11] N. Van Oosterwyck, A.B. yahya, A. Cuyt, S. Derammelaere, CAD Based trajectory optimization of PTP motions using Chebyshev polynomials, in: IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM, 2020.

[12] T. Herrmann, A. Wischnewski, L. Hermansdorfer, J. Betz, M. Lienkamp, Real-time adaptive velocity optimization for autonomous electric cars at the limits of handling, IEEE Trans. Intell. Veh. 6 (2021).

[13] S.E. Saarakkala, M. Hinkkanen, Identification of two-mass mechanical systems using torque excitation: Design and experimental evaluation, IEEE Trans. Ind. Appl. 51 (2015) 4180–4189.

[14] M. Abramowitz, I.A. Stegun, Orthogonal polynomials, in: Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, New York: Dover, 1972.

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

[16] E. Jacobsen, R. Lyons, The sliding DFT, IEEE Signal Process. Mag. 20 (2) (2003) 74–80.

[17] E. Jacobsen, R. Lyons, An update to the sliding DFT, IEEE Signal Process. Mag. 21 (1) (2004) 110–111.

[18] N. Norman S., Design via frequency response, in: Control Systems Engineering, sixth ed., Wiley, 2011, pp. 645–682.

[19] L.L.U.S. Ljung, System Identification: Theory for the User, second ed., Prentice Hall PTR, 1999, p. 609.

[20] Q. Zhang, Some implementation aspects of sliding window least squares algorithms, IFAC Proc. Vol. 33 (15) (2000).

[21] R. Lyons, A. Bell, The swiss army knife of digital networks, IEEE Signal Process. Mag. 21 (3) (2004) 90–100.