The Variable-Inertia Modified Computed-Torque Control of Robot Manipulators[1]

Jan Cvejn

Faculty of Electrical Engineering and Informatics, University of Pardubice, Studenska 95, 532 10 Pardubice, Czech Republic

e-mail: jan.cvejn@upce.cz

Abstract: This paper describes a modification of the computed-torque method of motion control of robot manipulators, which utilizes inner feedback to partially decouple inertial effects of individual links, but at the same time it minimizes the influence of the inner feedback component of the control input. In this way it is possible to increase the control efficiency for given magnitude of the control signals, which are also more easily kept within saturation limits. It is shown that uniform asymptotical stability of the control error during reference trajectory tracking can be ensured by choosing sufficiently high controller gains, and that the region of attraction can be made arbitrarily large. For the situations when some robot model parameters are not known precisely, an adaptive extension of the algorithm preserving asymptotical stability of the control error is proposed.

Keywords: *motion control, robot control, computed-torque control, inverse dynamics, adaptive control*

1. Introduction

In literature the approaches to motion control of robot manipulators are divided into decentralized and centralized ones [1, 2]. In the decentralized control architecture, the motion of each link is controlled separately from the others. The centralized robot control algorithms, where the robot dynamics is considered in full complexity, are more suitable in the cases of fast motions or direct drives, where nonlinearities in the equations of motion have strong influence. On the other hand, the centralized control poses much higher demands on the control system hardware performance.

The computed-torque control method [2, 3], also known as the inverse dynamics method [1], utilizes inner nonlinear feedback loop to transform the robot dynamics into linear and decoupled one. The outer-loop controller then can be designed in the same way as for a linear plant, and for each axis independently. Another advantage in comparison to simpler approaches, such as the PD or PID control [1-4], is easy incorporation of the information about the reference trajectory in the control law. This makes this approach effective especially for the reference trajectory tracking, which is a typical task in the case of motion control of robot manipulators.

An alternative to the computed-torque method is the PD+ motion control algorithm [3, 5], based on multivariable PD controller with nonlinear terms that compensate the gravity effects and the reference trajectory influence. Unlike the computed-torque method, the error dynamics remains nonlinear and coupled. In [6] and [7] the PD+ algorithm performance is improved by using fractional derivative operators.

Since applicability of the computed-torque method depends on precision of the robot mathematical model, which often cannot be fully guaranteed, extensions are needed in real cases. One possibility is an adaptive approach, where estimates of the model parameters are improved in real time. The classical adaptive algorithm by Craig et al. [8] ensures global asymptotical stability, but can fail due to the singularity of the inertia matrix estimate. This problem can be fixed by resetting the parameter estimates, but in [9] a more effective solution, based on additional feedback signal, is proposed. The method in [10] removes the need for acceleration measurements in the adaptation law. The mentioned difficulties are also avoided by using the inertia-related and passivity-based approaches [11, 12]. A fractional-order passivity-based adaptive controller of a robot manipulator is described in [13].

Alternatively, the control law can also be modified to preserve the closed-loop asymptotical stability even in the case of the model inaccuracies. One possible approach utilizes additional component in the controller that draws the system state to a sliding surface, where decreasing trend of a Lyapunov function is preserved [14, 1-2]. In more recent works [15-17] the computed-torque approach has been utilized for control of robots with uncertain dynamics.

A drawback of the computed-torque method may be the fact that the control inputs generated by the inner linearizing feedback are not included in the outer-loop controller design objectives. Available ranges of the control signals are not utilized efficiently, and it is possible to reach saturation limits of the actuators. This could cause the inner feedback malfunction with impacts on the control performance or closed-loop stability. Robot control algorithms with bounded inputs have been treated in literature extensively – for instance, in

[18] simple extensions of the PD control with gravity compensation are described. A modified computed-torque algorithm considering bounded control signals is proposed in [19]. In [20] the problem of finite-time tracking with saturated inputs is addressed.

In this paper the problem with limited ranges of the control signals is approached indirectly. A goal of this work is to modify the computed-torque method to utilize the available ranges of the control signals more effectively. The presented method utilizes the inner feedback to partially decouple the robot dynamics, which enhances controllability and enables efficient reference trajectory tracking. At the same time, however, the inner feedback influence is minimized by a suitable setting of the inner-loop system inertia parameter. The control signals thus predominantly depend on the control error, which increases the control efficiency for given magnitude of the control input, and helps to keep the control signals within saturation limits.

The basic idea of partial decoupling and reduction of the inner feedback influence has been presented in [21], but the corresponding analysis was simplified and considered only fixed inner-loop inertia settings and only the target position regulation problem. In this paper, the inner-loop inertia is considered time-variable, which required some modifications of the control signal generation to ensure uniform asymptotical stability of the control error. In addition, adaptive extensions have been proposed for the situations with uncertainties in the model parameters. The proposed control algorithm is described and analyzed in Section 3. Section 4 describes the adaptive extensions. Section 5 shows simulated results in comparison with the computed-torque control method and the PD+ algorithm.

2. Preliminaries

A rigid fully actuated serial robot manipulator consisting of n links in an open chain and moving freely in the operational space is considered. The position in the joint space is described by an n-dimensional vector of joint coordinates. The robot equations of motion can be written in the general form [1, 2]

$$\mathbf{B}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau}, \qquad (1)$$

where $\mathbf{q}$ is the vector of joint positions and $\boldsymbol{\tau}$ the vector of generalized force effects of the actuators. If $K(\mathbf{q},\dot{\mathbf{q}})$ and $P(\mathbf{q})$ denote the arm total kinetic and potential energy, respectively, $\mathbf{B}(\mathbf{q}) = \partial^2 K / \partial \dot{\mathbf{q}}^2$ is a symmetric positive definite position-dependent inertia matrix,

$$\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} = \dot{\mathbf{B}}(\mathbf{q})\dot{\mathbf{q}} - \partial K / \partial \mathbf{q} \tag{2}$$

is a nonlinear term corresponding to the effects of centrifugal and Coriolis forces and $\mathbf{g}(\mathbf{q})$ is the vector function corresponding to the gravity-force effects. The symbols $\partial K / \partial \mathbf{q}$ and $\partial P / \partial \mathbf{q}$ denote column vectors of partial derivatives. The matrix $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$ cannot be expressed uniquely from (2), but by substituting $K(\mathbf{q},\dot{\mathbf{q}}) = \dot{\mathbf{q}}^T \mathbf{B}(\mathbf{q})\dot{\mathbf{q}} / 2$ into (2) one realization of $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$ can be obtained in the form

$$\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) = \sum_{i=1}^{n} \frac{\partial \mathbf{B}}{\partial q_i}\dot{q}_i - \frac{1}{2}\left[\frac{\partial \mathbf{B}}{\partial q_1}\dot{\mathbf{q}} \ \ldots \ \frac{\partial \mathbf{B}}{\partial q_n}\dot{\mathbf{q}}\right]^T . \tag{3}$$

Total force effects of the actuators can be expressed in the form

$$\boldsymbol{\tau} = \mathbf{K}\mathbf{u} - \mathbf{F}_V \dot{\mathbf{q}} - \mathbf{F}_C \operatorname{sig}(\dot{\mathbf{q}}), \tag{4}$$

where $\mathbf{u}$ is the control variable,

$$\operatorname{sig}(\dot{\mathbf{q}}) = \left[\operatorname{sign}(\dot{q}_1),...,\operatorname{sign}(\dot{q}_n)\right]^T, \tag{5}$$

and $\mathbf{K} > 0$, $\mathbf{F}_V > 0$ and $\mathbf{F}_C > 0$ are diagonal matrices. The last term in (4) corresponds to the Coulomb friction effects in joints, whereas $\mathbf{F}_V\dot{\mathbf{q}}$ corresponds to the viscous and electro-magnetic friction. Note, however, that the Coulomb friction model is simplified and does not consider possible dependence on position.

Consequently, the robot model can be hereafter considered in the form

$$\mathbf{B}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{Z}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{F}_C \operatorname{sig}(\dot{\mathbf{q}}) = \mathbf{K}\mathbf{u}, \tag{6}$$

where $\mathbf{Z}(\mathbf{q},\dot{\mathbf{q}}) = \mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{F}_V$. The variables $\mathbf{q}^{(k)}$ and $\mathbf{u}$ are actually functions of time, but the time argument is omitted for brevity. Since $\mathbf{B}(\mathbf{q}) > 0$ for any $\mathbf{q}$,

$$0 < \lambda^B_{\min}(\mathbf{q})\|\mathbf{y}\|^2 \le \mathbf{y}^T\mathbf{B}(\mathbf{q})\mathbf{y} \le \lambda^B_{\max}(\mathbf{q})\|\mathbf{y}\|^2 \tag{7}$$

for any $\mathbf{y} \in R^n$, where $\lambda^B_{\min}(\mathbf{q})$ and $\lambda^B_{\max}(\mathbf{q})$ denote the minimal and the maximal eigenvalue of $\mathbf{B}(\mathbf{q})$, respectively. The symbol $\|.\|$ denotes the Euclidean norm, and the same notation will be used hereafter. The inequality

$$\|\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\| \le C_m \|\dot{\mathbf{q}}\| \tag{8}$$

holds in general, where $C_m > 0$ is a constant for given manipulator [2].

Finally, the left-hand side of equation (1) can be expressed as a linear function of a certain number of dynamic parameters, such as the masses and the 1st and 2nd-order moments of inertia of the robot links [1, 2]. Since the linear dependence on the parameters is preserved at the terms $\mathbf{F}_V\dot{\mathbf{q}}$ and $\mathbf{F}_C\,\mathrm{sig}(\dot{\mathbf{q}})$ as well, equation (6) can be written in the form

$$\mathbf{Y}(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})\boldsymbol{\theta} = \mathbf{K}\mathbf{u}, \tag{9}$$

where $\boldsymbol{\theta} \in R^p$ is a parameter vector and the function $\mathbf{Y}(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})$, called regressor, does not depend on $\boldsymbol{\theta}$.

3. The variable-inertia computed-torque control

3.1. Principle of the method

Denote $\mathbf{q}_d(t)$ the reference trajectory and $\mathbf{e} = \mathbf{q}_d - \mathbf{q}$ the control error vector. The classical computed-torque control method uses inner feedback compensation of nonlinear terms in equation (6) to make the dynamics linear and decoupled. If $\mathbf{u}$ is set as

$$\mathbf{u} = \mathbf{K}^{-1}\left[\,\mathbf{B}(\mathbf{q})\mathbf{v} + \mathbf{Z}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{F}_C\,\mathrm{sig}(\dot{\mathbf{q}})\right], \tag{10}$$

where $\mathbf{v}$ is an input signal, the resulting dynamics is described by the linear equation $\ddot{\mathbf{q}} = \mathbf{v}$. By choosing the outer-loop controller in the form

$$\mathbf{v} = \mathbf{R}_0\mathbf{e} + \mathbf{R}_1\dot{\mathbf{e}} + \ddot{\mathbf{q}}_d, \tag{11}$$

where the matrices $\mathbf{R}_0$ and $\mathbf{R}_1$ are positive definite and diagonal, the control error dynamics will obey the linear equation

$$\ddot{\mathbf{e}} + \mathbf{R}_0\mathbf{e} + \mathbf{R}_1\dot{\mathbf{e}} = \mathbf{0}. \tag{12}$$

The parameters in $\mathbf{R}_0$ and $\mathbf{R}_1$ can be chosen with respect to suitable criteria, such as the response time and damping. However, since the variable $\mathbf{u}$ magnitude is not taken into account, available ranges of the control signals cannot be utilized efficiently. An alternative to the computed-torque method is the PD+ algorithm, where

$$\mathbf{u} = \mathbf{K}^{-1}\left[\,\mathbf{R}_0\mathbf{e} + \mathbf{R}_1\dot{\mathbf{e}} + \mathbf{B}(\mathbf{q})\ddot{\mathbf{q}}_d + \mathbf{Z}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}_d + \mathbf{g}(\mathbf{q}) + \mathbf{F}_C\,\mathrm{sig}(\dot{\mathbf{q}})\right]. \tag{13}$$

Unlike the computed-torque method, the effect of the terms $\mathbf{B}(\mathbf{q})$ and $\mathbf{Z}(\mathbf{q},\dot{\mathbf{q}})$ is not compensated, so the error dynamics remains nonlinear and coupled.

The method proposed in this paper utilizes inner feedback loop to simplify the robot dynamics in a similar way as in the computed-torque method case. Consider the modified control law in the form

$$\mathbf{Ku} = \beta^{-1}\mathbf{B}(\mathbf{q})\left(\mathbf{R}_0\mathbf{e} + \mathbf{R}_1\dot{\mathbf{e}}\right) + \mathbf{w}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) + \mathbf{F}_C\,\mathrm{sig}(\dot{\mathbf{q}}) \tag{14}$$

where $\beta > 0$ is a scalar parameter, and

$$\mathbf{w}(\mathbf{q},\dot{\mathbf{q}}) = \left[\mathbf{I} - \beta^{-1}\mathbf{B}(\mathbf{q})\right]\mathbf{Z}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}\,. \tag{15}$$

Since $\mathbf{B}(\mathbf{q}) > 0$, a substitution into (6) and a multiplication by $\beta\mathbf{B}^{-1}(\mathbf{q})$ yields

$$\beta\ddot{\mathbf{q}} + \mathbf{Z}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} = \mathbf{R}_0\mathbf{e} + \mathbf{R}_1\dot{\mathbf{e}}\,. \tag{16}$$

In equation (16) the dynamical coupling is partially removed, and the acceleration of each link is controlled separately from the others. The matrices $\mathbf{R}_0$ and $\mathbf{R}_1$ are considered diagonal in general, but (16) suggests that $\mathbf{R}_0 = r_0\mathbf{I}$, where $r_0 > 0$, could be used without significant loss of generality. On the other hand, $\mathbf{R}_1$ should be considered in a more general form to enable to set different damping effects of the controller on individual axes.

If $\mathbf{q}_d = \text{const.}$, equation (16) is equivalent to

$$\beta\ddot{\mathbf{e}} + \mathbf{Z}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{e}} + \mathbf{R}_1\dot{\mathbf{e}} + \mathbf{R}_0\mathbf{e} = \mathbf{0}\,. \tag{17}$$

However, if $\mathbf{q}_d(t)$ is a general twice continuously differentiable function, the term $\beta\ddot{\mathbf{q}}_d + \mathbf{Z}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}_d$ appears on the right-hand side of (17), so $\mathbf{e}(t)$ cannot converge to zero, unless this term is compensated. The corresponding modification of the control law is in the form

$$\mathbf{Ku}^* = \mathbf{Ku} + \mathbf{B}(\mathbf{q})\left[\ddot{\mathbf{q}}_d + \beta^{-1}\mathbf{Z}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}_d\right], \tag{18}$$

where $\mathbf{Ku}$ is given by (14). Although the resulting error dynamics (17) remains nonlinear, an advantage of the control law (14) is the possibility to reduce the influence of the feedback term $\mathbf{w}(\mathbf{q},\dot{\mathbf{q}})$ by a suitable choice of β, as discussed in Section 3.2.

As regards technical realization, it does not differ significantly from the computed-torque method. Both the approaches have similar demands on the controller hardware. In particular,

the inner feedback requires a very short scan period, typically about 1 ms or shorter. Within the scan period all the terms of the robot model have to be computed, so a high-performance controller unit has to be used. Efficient computation of the terms in the robot model is a stand-alone problem, which is studied in literature extensively - see e.g. [22-24].

3.2. The variable β computation

In principle, since $\beta\ddot{\mathbf{q}}$ replaces $\mathbf{B}(\mathbf{q})\ddot{\mathbf{q}}$ in (16), the inertia parameter β should be set so that the terms $\beta\ddot{\mathbf{q}}$ and $\mathbf{B}(\mathbf{q})\ddot{\mathbf{q}}$ have similar magnitudes. Therefore, a proper choice of β should respect the inequalities

$$\lambda_{\min}^{B}(\mathbf{q}) \le \beta \le \lambda_{\max}^{B}(\mathbf{q}) \tag{19}$$

at any moment. However, it may be impossible to find a constant value of β such that (19) always holds. It is possible to require that

$$\inf\left\{\lambda_{\min}^{B}(\mathbf{q})\right\} \le \beta \le \sup\left\{\lambda_{\max}^{B}(\mathbf{q})\right\} \tag{20}$$

for the considered extent of position, but the sides of (20) can differ strongly and do not provide a conclusive information how to obtain a suitable setting of β.

Therefore, it is more advantageous to work with variable β, depending on $\mathbf{q}$ and $\dot{\mathbf{q}}$. Since the controller terms are multiplied by β^{-1} in (14), to suppress the influence of $\mathbf{w}(\mathbf{q},\dot{\mathbf{q}})$ in the control signal, β should be selected so that

$$f(\beta) = \left\| \left[\mathbf{I}\beta - \mathbf{B}(\mathbf{q})\right]\mathbf{Z}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} \right\| \tag{21}$$

is minimized. For $\mathbf{y} = \mathbf{Z}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}$, this problem is equivalent to the minimization of $\beta^2\|\mathbf{y}\|^2 - 2\beta\mathbf{y}^T\mathbf{B}(\mathbf{q})\mathbf{y}$, which yields

$$\beta(\mathbf{q},\dot{\mathbf{q}}) = \mathbf{y}^T\mathbf{B}(\mathbf{q})\mathbf{y} / \|\mathbf{y}\|^2 . \tag{22}$$

The value of β given by (22) is always positive and respects the inequalities (19). Note that due to the multiplication of $\mathbf{v}$ by $\mathbf{B}(\mathbf{q})$ in (10), the feedback control signals ranges can depend strongly on the trajectory at the computed-torque method. In the control law (14) this effect is partially suppressed by the multiplication of $\mathbf{B}(\mathbf{q})$ with β^{-1}, because β given by (22) lies in the interval (19).

Although β defined by (22) is theoretically always bounded, there is a practical issue with the computation of β for $t = 0$, because both the numerator and denominator in (22) are zero due to $\dot{\mathbf{q}}(0) = \mathbf{0}$. A simple possibility is to set $\beta = \mathrm{tr}(\mathbf{B})/n$ at $t = 0$, which preserves validity of (19). A more sophisticated solution is to use the value corresponding to the reference trajectory for t close to 0. It is true that $\mathbf{y} = 0$ can happen even for $t > 0$, but this problem can be solved by the computation of β based on Eq. (24), discussed below.

Another problem is the fact that the closed-loop stability, analyzed in Section 3.3, requires that also $\dot{\beta}$ is bounded. Actually, if β is given by (22), $\dot{\beta}$ is unbounded for $\mathbf{y} \to 0$. In particular,

$$\frac{d\beta}{dt} = \frac{\mathbf{y}^T}{\|\mathbf{y}\|}\left[2\mathbf{B}(\mathbf{q})\frac{\dot{\mathbf{y}}\mathbf{y}^T - \mathbf{y}\dot{\mathbf{y}}^T}{\|\mathbf{y}\|^2} + \dot{\mathbf{B}}(\mathbf{q})\right]\frac{\mathbf{y}}{\|\mathbf{y}\|}, \tag{23}$$

where the term $\left(\dot{\mathbf{y}}\mathbf{y}^T - \mathbf{y}\dot{\mathbf{y}}^T\right)/\|\mathbf{y}\|^2$ goes to infinity for $\mathbf{y} \to 0$. Therefore, it is proposed to calculate β for $t > 0$ as the solution to the differential equation

$$\mu\dot{\beta} + \beta = \mathbf{y}^T\mathbf{B}(\mathbf{q})\mathbf{y}/\|\mathbf{y}\|^2, \tag{24}$$

where the parameter $\mu > 0$ is considered time-variable in general. The computation of β using (24) also enables to easily avoid the problem with $\mathbf{y} = 0$ mentioned above. If $\|\mathbf{y}\|$ is very close to zero, $\dot{\beta} = 0$ can be set instead of calculating $\dot{\beta}$ from (24), so β is held constant. The following result shows that the validity of (20) is preserved by any solution to (24).

Proposition 1. If $\mu(t) > 0$ is bounded, and β satisfies (20) for $t = 0$, then β obtained as the solution to (24) along any trajectory $\mathbf{q}(t)$ satisfies (20) for all $t \geq 0$.

Proof. For given $\mathbf{q}(t)$, denote $v(t)$ the right-hand side of (24) and let $t = t(p)$, where the function $t(p)$ is differentiable, increasing, and $t(0) = 0$. Let $\hat{\beta}(p) = \beta(\mathbf{q}, \dot{\mathbf{q}})$, and $\hat{v}(p) = v(t)$ for $p \geq 0$, whereas $\hat{v}(p) = \beta(0)$ for $p < 0$. For $\mu = dt/dp$, equation (24) can be written as

$$d\hat{\beta}/dp + \hat{\beta}(p) = \hat{v}(p). \tag{25}$$

The solution to (25) can be expressed in the form

$$\hat{\beta}(p) = \int_{-\infty}^{p} e^{-(p-\tau)} \hat{v}(\tau)\, d\tau \le \sup\{\hat{v}(p)\} \int_{-\infty}^{p} e^{-(p-\tau)} d\tau = \sup\{\hat{v}(p)\} \qquad (26)$$

and analogously also $\hat{\beta}(p) \ge \inf\{\hat{v}(p)\}$ is obtained. These bounds are equivalent to (20).

□

As a basic version, the parameter μ can be chosen fixed and in such a case it has the meaning of a time constant. However, since

$$|\dot{\beta}| \le \frac{1}{\mu}\left[\sup\{\lambda_{\max}^{B}(\mathbf{q})\} - \inf\{\lambda_{\min}^{B}(\mathbf{q})\}\right] \qquad (27)$$

follows from (20) and (24), $|\dot{\beta}|$ can be rather large if μ is chosen short. Since the magnitude of $\dot{\beta}$ influences the stabilizing region of the controller parameters, this choice will lead to unnecessarily high controller gains.

Therefore, it seems to be more advantageous to set $\mu^{-1} = \mu_1 \|\dot{\mathbf{q}}\|$, where $\mu_1 > 0$ is a chosen constant, and this variant is considered hereafter. In this case $|\dot{\beta}| \le \kappa_\beta \|\dot{\mathbf{q}}\|$ holds, where

$$\kappa_\beta = \mu_1\left[\sup\{\lambda_{\max}^{B}(\mathbf{q})\} - \inf\{\lambda_{\min}^{B}(\mathbf{q})\}\right]. \qquad (28)$$

The term $\mu_1 \|\dot{\mathbf{q}}\|$ ensures that for larger $\|\dot{\mathbf{q}}\|$ the value of β is close to (22) at any moment and that β is changing slowly when $\dot{\mathbf{q}}$ is close to zero. A proper value of μ_1 can be set empirically from the requirement that μ should not be lower than a chosen value, e.g. 0.01s, by estimating the maximum of $\|\dot{\mathbf{q}}\|$ along the trajectory.

3.3. The closed-loop stability and convergence analysis

The robot model is considered precise in this sub-section. The reference trajectory $\mathbf{q}_d$ is assumed continuously differentiable, and such that $\mathbf{q}_d^{(k)}$, $k = 0,1,2$, are bounded. The compensation of $\mathbf{q}_d$ discussed in Section 3.1 is assumed. In correspondence with Eq. (17), denote $\boldsymbol{\xi}^T = \left[\mathbf{e}^T, \dot{\mathbf{e}}^T\right]$ the error state vector. The matrices $\mathbf{R}_0$ and $\mathbf{R}_1$ are considered diagonal in the form $\mathbf{R}_0 = \mathrm{diag}(r_{01},...,r_{0n})$ and $\mathbf{R}_1 = \mathrm{diag}(r_{11},...,r_{1n})$, and the matrix $\mathbf{F}_V$ of the viscous friction coefficients is in the form $\mathbf{F}_V = \mathrm{diag}(f_{V1},...,f_{Vn})$. The variable β is computed as the solution to (24), where $\mu^{-1} = \mu_1 \|\dot{\mathbf{q}}\|$, $\mu_1 > 0$.

Theorem 1. Assume that $\mathbf{R}_0 > 0$ and $\|\dot{\mathbf{q}}_d\| \le r_d$. Then, to any $\rho > 0$ it is possible to find γ_1 such that if $r_{1j} \ge \gamma_1$ for all $j = 1,...,n$, the error dynamics given by (17) is uniformly asymptotically stable at the origin during the reference trajectory tracking, and any error state ξ such that $\|\xi\| < \rho$ lies in the region of attraction.

Proof. Let

$$V(\xi,t) = \frac{\beta}{2}\|\dot{\mathbf{e}} + \alpha\mathbf{e}\|^2 + \frac{1}{2}\mathbf{e}^T\mathbf{R}_0\mathbf{e}. \tag{29}$$

For constant $\beta > 0$, $V(\xi,t)$ is positive definite, since $V(\xi,t) \ge 0$, and $V(\xi,t) = 0$ only if $\xi = 0$. Considering β variable and such that (20) are satisfied, it follows that

$$V_1(\xi) \le V(\xi,t) \le V_2(\xi) \tag{30}$$

where $V_1(\xi)$ and $V_2(\xi)$ are some time-invariant positive definite functions. In addition, $V(\xi,t)$ is radially unbounded, because $V(\xi,t) \to \infty$ whereas $\xi \to \infty$.

The time derivative of $V(\xi,t)$ along the system (17) trajectory is in the form

$$\begin{aligned} \dot{V} &= \beta(\dot{\mathbf{e}} + \alpha\mathbf{e})^T(\ddot{\mathbf{e}} + \alpha\dot{\mathbf{e}}) + \frac{\dot{\beta}}{2}\|\dot{\mathbf{e}} + \alpha\mathbf{e}\|^2 + \dot{\mathbf{e}}^T\mathbf{R}_0\mathbf{e} = \\ &= -(\dot{\mathbf{e}} + \alpha\mathbf{e})^T[\mathbf{Z}\dot{\mathbf{e}} + \mathbf{R}_1\dot{\mathbf{e}} - \alpha\beta\dot{\mathbf{e}}] - \alpha\mathbf{e}^T\mathbf{R}_0\mathbf{e} + \frac{\dot{\beta}}{2}\|\dot{\mathbf{e}} + \alpha\mathbf{e}\|^2. \end{aligned} \tag{31}$$

Let $\mathbf{A} = \mathbf{Z} + \mathbf{R}_1 - (\alpha\beta + \dot{\beta})\mathbf{I}$. Eq. (31) then can be written as

$$\begin{aligned} \dot{V} &= -\begin{bmatrix} \dot{\mathbf{e}} \\ \alpha\mathbf{e} \end{bmatrix}^T \begin{bmatrix} \mathbf{A} + \dot{\beta}\mathbf{I}/2 & -\dot{\beta}\mathbf{I}/2 \\ \mathbf{A} + \dot{\beta}\mathbf{I}/2 & \alpha^{-1}\mathbf{R}_0 - \dot{\beta}\mathbf{I}/2 \end{bmatrix} \begin{bmatrix} \dot{\mathbf{e}} \\ \alpha\mathbf{e} \end{bmatrix} = \\ &= -\frac{1}{2}\begin{bmatrix} \dot{\mathbf{e}} \\ \alpha\mathbf{e} \end{bmatrix}^T \begin{bmatrix} \mathbf{A} + \mathbf{A}^T + \dot{\beta}\mathbf{I} & \mathbf{A}^T \\ \mathbf{A} & 2\alpha^{-1}\mathbf{R}_0 - \dot{\beta}\mathbf{I} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{e}} \\ \alpha\mathbf{e} \end{bmatrix}. \end{aligned} \tag{32}$$

By using the Schur complement rule [25], $\dot{V}$ is negative definite if and only if $\mathbf{A} + \mathbf{A}^T + \dot{\beta}\mathbf{I} > 0$ and

$$2\alpha^{-1}\mathbf{R}_0 - \dot{\beta}\mathbf{I} - \mathbf{A}(\mathbf{A} + \mathbf{A}^T + \dot{\beta}\mathbf{I})^{-1}\mathbf{A}^T > 0. \tag{33}$$

The first requirement is satisfied if

$$\mathbf{Z} + \mathbf{R}_1 - (\dot{\beta}/2 + \alpha\beta)\mathbf{I} - \delta\mathbf{I}/2 \ge 0, \tag{34}$$

where $\delta > 0$ is arbitrary. Assume $\|\dot{\mathbf{e}}\| \le r_e$ for some $r_e > 0$. Considering (8), (20) and $|\dot{\beta}| \le \kappa_\beta \|\dot{\mathbf{q}}_d - \dot{\mathbf{e}}\|$ for κ_β given by (28), it follows that (34) holds if $\mathbf{R}_1$ is chosen so that

$$f_{Vi} + r_{1i} \ge \left(C_m + \kappa_\beta / 2\right)\left(r_d + r_e\right) + \alpha\beta_m + \delta/2 \tag{35}$$

for all $i = 1,...,n$, where $\beta_m = \sup\left\{\lambda_{\max}^B\right\}$, which is clearly possible to any given $r_e > 0$ and $\delta > 0$. Further, let

$$A_m = \|\mathbf{R}_1 + \mathbf{F}_V\| + \left(C_m + \kappa_\beta\right)\left(r_d + r_e\right) + \alpha\beta_m \ge \|\mathbf{A}\|. \tag{36}$$

Since $\mathbf{A} + \mathbf{A}^T + \dot{\beta}\mathbf{I} - \delta\mathbf{I} \ge 0$, condition (33) is satisfied if $\mathbf{R}_0$ is set so that

$$2\alpha^{-1} r_{0i} > \kappa_\beta\left(r_d + r_e\right) + A_m^2/\delta \tag{37}$$

for all $i = 1,...,n$, which is always possible for any given $\alpha > 0$. Moreover, since α can be chosen arbitrarily close to zero, it is sufficient that $r_{0i} > 0$.

Since the matrix in (32) is made positive definite by the settings of $\mathbf{R}_1$ and $\mathbf{R}_0$, it is $\dot{V} \le -\nu\|\dot{\mathbf{e}} + \alpha\mathbf{e}\|$ for some $\nu > 0$, so there exists a time-invariant positive definite function $V_3(\boldsymbol{\xi})$ such that $\dot{V}(\boldsymbol{\xi}, t) \le -V_3(\boldsymbol{\xi})$. The facts that (30) holds and $\dot{V} \le -V_3$ mean that the system (17) is uniformly asymptotically stable by the Lyapunov theorem for non-autonomous systems [26, 27]. Since $V(\boldsymbol{\xi}, t)$ is radially unbounded, the stability region depends only on r_e. Considering (20), it is possible to write

$$k_1 \|\boldsymbol{\xi}\|^2 \le V(\boldsymbol{\xi}, t) \le k_2 \|\boldsymbol{\xi}\|^2 \tag{38}$$

for some $k_1 > 0$ and $k_2 > 0$. Since $\dot{V} < 0$, for any $\tau \ge 0$ it is

$$\|\dot{\mathbf{e}}(t+\tau)\|^2 \le \|\boldsymbol{\xi}(t+\tau)\|^2 \le \frac{V(\boldsymbol{\xi})}{k_1} \le \frac{k_2}{k_1}\|\boldsymbol{\xi}\|^2. \tag{39}$$

Consequently, it is possible to set $r_e = \rho\sqrt{k_2/k_1}$. □

The fact that the region of attraction can be made arbitrarily large means that the asymptotical stability is semi-global. Note that $\mathbf{R}_1 > \mathbf{0}$ need not be necessary in some cases, but the controller with $\mathbf{R}_1 = \mathbf{0}$ is not sufficient in general. A possible initial setting of $\mathbf{R}_1$ could be based on Eq. (35) for $r_e = 0$.

4. The adaptive version of the method

4.1. The adaptation of uncertain parameters

The proposed method is based on a precise model of the robot, which is often not available in real situations. It was verified by simulations that the quality of control is sensitive to errors of the model, although the effect of small errors need not be important. In special, the errors in the term $\mathbf{g}(\mathbf{q})$ cause a loss of precision of reaching the target point, which can be only partially compensated by increasing the $\mathbf{R}_0$ gain. One way how to overcome this problem, followed below, is an adaptive extension based on continual improvements of the estimates of unknown model parameters. The proposed extension utilizes the linear property of the robot model (9), like most other adaptive robot control algorithms.

Consider the parameter vector $\boldsymbol{\theta}$ in (9) divided as $\boldsymbol{\theta}^T = \left[\boldsymbol{\theta}_1^T, \boldsymbol{\theta}_2^T\right]$, where $\boldsymbol{\theta}_1$ contains the parameters known precisely, whereas the part $\boldsymbol{\theta}_2$ is to be adapted. The regressor function output $\mathbf{Y}$ is divided accordingly as $\mathbf{Y} = \left[\mathbf{Y}_1, \mathbf{Y}_2\right]$. The actual values of the terms $\mathbf{B}(\mathbf{q})$, $\mathbf{Z}(\mathbf{q},\dot{\mathbf{q}})$, $\mathbf{g}(\mathbf{q})$ and $\mathbf{F}_C \operatorname{sig}(\dot{\mathbf{q}})$ in equation (6) are considered in the form $\mathbf{B}+\boldsymbol{\Delta}_B$, $\mathbf{Z}+\boldsymbol{\Delta}_Z$, $\mathbf{g}+\boldsymbol{\Delta}_g$, and $\mathbf{F}_C \operatorname{sig}(\dot{\mathbf{q}})+\boldsymbol{\Delta}_F$, where the arguments $\mathbf{q}$ and $\dot{\mathbf{q}}$ are omitted for brevity – for instance, $\boldsymbol{\Delta}_B$ is a function of $\mathbf{q}$. Substituting the actual values, Eq. (6) is rewritten as follows

$$\left(\mathbf{B}+\boldsymbol{\Delta}_B\right)\ddot{\mathbf{q}} + \mathbf{Z}\dot{\mathbf{q}} + \boldsymbol{\Delta}_Z\dot{\mathbf{q}} + \mathbf{g} + \boldsymbol{\Delta}_g + \mathbf{F}_C \operatorname{sig}(\dot{\mathbf{q}}) + \boldsymbol{\Delta}_F = \mathbf{K}\mathbf{u}\,. \tag{40}$$

Note that $\mathbf{B}$, $\mathbf{Z}$, $\mathbf{g}$ and $\mathbf{F}_C$ in the controller (14) are now estimates of the terms in the robot model, so these symbols have different meaning than in Sections 2 and 3. Substitution of (14) into (40) yields

$$\left(\mathbf{B}+\boldsymbol{\Delta}_B\right)\ddot{\mathbf{q}} + \beta^{-1}\mathbf{B}\mathbf{Z}\dot{\mathbf{q}} + \boldsymbol{\Delta}_Z\dot{\mathbf{q}} + \boldsymbol{\Delta}_g + \boldsymbol{\Delta}_F = \beta^{-1}\mathbf{B}\left(\mathbf{R}_0\mathbf{e} + \mathbf{R}_1\dot{\mathbf{e}}\right). \tag{41}$$

If the actual value of $\boldsymbol{\theta}_2$ is $\boldsymbol{\theta}_2 + \boldsymbol{\Delta}_\theta$, then it is possible to write

$$\boldsymbol{\Delta}_B\ddot{\mathbf{q}} + \boldsymbol{\Delta}_Z\dot{\mathbf{q}} + \boldsymbol{\Delta}_g + \boldsymbol{\Delta}_F = \mathbf{Y}_2\left(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}}\right)\boldsymbol{\Delta}_\theta\,. \tag{42}$$

Using (41), (42) and (18), equation (17) is replaced with

$$\beta\ddot{\mathbf{e}} + \mathbf{Z}\dot{\mathbf{e}} + \mathbf{R}_1\dot{\mathbf{e}} + \mathbf{R}_0\mathbf{e} = \beta\mathbf{B}^{-1}\mathbf{Y}_2\left(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}}\right)\boldsymbol{\Delta}_\theta\,. \tag{43}$$

Note that $\mathbf{Z}$, $\mathbf{B}$ and β now also depend on $\boldsymbol{\theta}_2$. Let $\sigma : R^+ \to R$ be a non-increasing function such that $\sigma(t) \geq \sigma_0$, where $\sigma_0 > 0$. In addition, $\dot{\sigma}(t)$ is assumed continuous and bounded. The proposed adaptation of $\boldsymbol{\theta}_2$ is based on the integration of the equation

$$\sigma(t)\boldsymbol{\Gamma}\dot{\boldsymbol{\theta}}_2 = \beta \mathbf{Y}_2^T(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\mathbf{B}^{-1}(\mathbf{q})[\dot{\mathbf{e}} + \alpha \mathbf{e}], \tag{44}$$

where $\boldsymbol{\Gamma} > 0$ is a constant diagonal matrix and $\alpha > 0$ is a chosen constant. The adaptation law (44) was inspired by other robot adaptive control methods, especially [8]. The influence of $\mathbf{e}$ in (44) is necessary to ensure $\mathbf{e} \to \mathbf{0}$ for $t \to \infty$. Although $\alpha > 0$ can be theoretically chosen arbitrarily low, choosing too low α leads to a sluggish convergence of $\boldsymbol{\theta}_2$ and $\boldsymbol{\xi}$. On the other hand, the analysis in Section 4.2 reveals that increasing α reduces the stability region.

The purpose of the function $\sigma(t)$ in (44) is to speed up the convergence rate in the final phase of the trajectory, where $\mathbf{e}$ and $\dot{\mathbf{e}}$ are close to zero. One possibility is to choose $\sigma(t)$ in the form

$$\sigma(t) = \sigma_0 + \left(1 + \sigma_1 t^{\nu}\right)^{-1}, \tag{45}$$

where $\sigma_1 \geq 0$ and $\nu > 0$. The constant $\sigma_0 > 0$ can be usually chosen very low. The choice $\nu = 2$ or 3 seems to be suitable in most cases. It is advantageous to set σ_1 so that $\sigma(t)$ has a prescribed value by the end of the trajectory – for instance, one half of $\sigma(0)$. It is possible to choose $\sigma_1 = 0$ at first to find proper settings of $\boldsymbol{\Gamma}$ and α.

4.2. The adaptive controller - stability analysis

Theorem 2. Consider the same situation as in the case of Theorem 1. Then, if the adaptation of $\boldsymbol{\theta}_2$ is based on (44), to any $\rho > 0$ it is possible to find $\gamma_0 > 0$ and γ_1 so that if $r_{0i} \geq \gamma_0$ and $r_{1i} \geq \gamma_1$ for all $i = 1, ..., n$, the system (43) is uniformly stable in the region such that $\left\|(\boldsymbol{\xi}, \boldsymbol{\Delta}_\theta)\right\| < \rho$, and any trajectory of $\boldsymbol{\xi}$ starting at any point in this region converges to zero.

Proof. The function

$$W(\boldsymbol{\xi}, \boldsymbol{\Delta}_\theta, t) = V(\boldsymbol{\xi}, t) + \frac{\sigma}{2}\boldsymbol{\Delta}_\theta^T \boldsymbol{\Gamma} \boldsymbol{\Delta}_\theta \tag{46}$$

where $V(\boldsymbol{\xi},t)$ is given by (29), satisfies $W_1(\boldsymbol{\xi},\boldsymbol{\Delta}_\theta) \le W(\boldsymbol{\xi},\boldsymbol{\Delta}_\theta,t)$ for some time-invariant positive definite function $W_1(\boldsymbol{\xi},\boldsymbol{\Delta}_\theta)$, because $\sigma \ge \sigma_0 > 0$ and (30) holds. The time derivative of $W(\boldsymbol{\xi},\boldsymbol{\Delta}_\theta,t)$ along the system (43) trajectory is in the form

$$\begin{aligned}\dot{W} &= \beta\left(\dot{\mathbf{e}}+\alpha\mathbf{e}\right)^T\left(\ddot{\mathbf{e}}+\alpha\dot{\mathbf{e}}\right)+\frac{\dot{\beta}}{2}\left\|\dot{\mathbf{e}}+\alpha\mathbf{e}\right\|^2+\dot{\mathbf{e}}^T\mathbf{R}_0\mathbf{e}+\sigma\dot{\boldsymbol{\Delta}}_\theta^T\boldsymbol{\Gamma}\boldsymbol{\Delta}_\theta+\frac{\dot{\sigma}}{2}\boldsymbol{\Delta}_\theta^T\boldsymbol{\Gamma}\boldsymbol{\Delta}_\theta = \\ &= -\left(\dot{\mathbf{e}}+\alpha\mathbf{e}\right)^T\left[\mathbf{Z}\dot{\mathbf{e}}+\mathbf{R}_1\dot{\mathbf{e}}-\beta\mathbf{B}^{-1}\mathbf{Y}_2\boldsymbol{\Delta}_\theta-\alpha\beta\dot{\mathbf{e}}\right]+ \\ &+\frac{\dot{\beta}}{2}\left\|\dot{\mathbf{e}}+\alpha\mathbf{e}\right\|^2-\alpha\mathbf{e}^T\mathbf{R}_0\mathbf{e}+\sigma\dot{\boldsymbol{\Delta}}_\theta^T\boldsymbol{\Gamma}\boldsymbol{\Delta}_\theta+\frac{\dot{\sigma}}{2}\boldsymbol{\Delta}_\theta^T\boldsymbol{\Gamma}\boldsymbol{\Delta}_\theta .\end{aligned} \tag{47}$$

Since $\dot{\boldsymbol{\theta}}_2+\dot{\boldsymbol{\Delta}}_\theta=\mathbf{0}$, (44) ensures that

$$\beta\left(\dot{\mathbf{e}}+\alpha\mathbf{e}\right)^T\mathbf{B}^{-1}\mathbf{Y}_2\boldsymbol{\Delta}_\theta+\sigma\dot{\boldsymbol{\Delta}}_\theta^T\boldsymbol{\Gamma}\boldsymbol{\Delta}_\theta=0 . \tag{48}$$

Consequently, $\dot{W}=\dot{V}+\dot{\sigma}\boldsymbol{\Delta}_\theta^T\boldsymbol{\Gamma}\boldsymbol{\Delta}_\theta/2$, where $\dot{V}$ is given by (32).

Proceeding analogously to the proof of Theorem 1 and considering $\|\dot{\mathbf{e}}\| \le r_e$ for some $r_e > 0$, it is possible to ensure that $\dot{V}$ given by (32) is negative definite. Note that increasing α in (34) and (37) requires setting higher values of r_{0i} and r_{1i}, so α has ill effect on the stability margin. Since $\dot{\sigma} \le 0$, this means that $\dot{W} \le 0$. Consequently, considering $W_1 \le W$, the system (43) is uniformly stable by the Lyapunov stability theorem [26, 27].

Further, since $\dot{W} \le 0$ and W is bounded from below, W converges to a limit. Since $\boldsymbol{\xi}$ and $\boldsymbol{\Delta}_\theta$ are bounded, $\dot{W}$ is bounded. In addition, $\boldsymbol{\xi}$ and $\boldsymbol{\Delta}_\theta=-\boldsymbol{\theta}_2$ are generated as the solutions to the differential equations (43) and (44), where $\ddot{\mathbf{e}}$ and $\dot{\boldsymbol{\theta}}_2$ are bounded, so $\boldsymbol{\xi}$, $\boldsymbol{\Delta}_\theta$, and consequently also $\dot{W}$, are continuous in time. This means that $\dot{W} \to 0$ for $t \to \infty$ (see e.g. [3], Lemma A.6). Analogously, $\dot{\sigma}$ must converge to zero. Since $\|\boldsymbol{\Delta}_\theta\|$ is bounded, it follows that $\dot{W} \to \dot{V}$ for $t \to \infty$, so $\boldsymbol{\xi}$ converges to zero, because $\dot{V}$ is negative definite.

Finally, in analogy with the proof of Theorem 1, since

$$k_1\|\boldsymbol{\xi}\|^2 \le W(\boldsymbol{\xi},\boldsymbol{\Delta}_\theta,t) \le k_2\|\boldsymbol{\xi}\|^2+k_\Delta\|\boldsymbol{\Delta}_\theta\|^2 \tag{49}$$

for some constants $k_1,k_2,k_\Delta>0$, and $\dot{W}<0$, for any $\tau \ge 0$ it is

$$\|\dot{\mathbf{e}}(t+\tau)\|^2 \le \frac{W(\boldsymbol{\xi},\boldsymbol{\Delta}_\theta,t)}{k_1} \le \frac{k_m}{k_1}\left(\|\boldsymbol{\xi}\|^2+\|\boldsymbol{\Delta}_\theta\|^2\right), \tag{50}$$

where $k_m = \max\{k_2, k_\Delta\}$. Consequently, it is possible to choose $r_e = \rho\sqrt{k_m / k_1}$. □

Note that $\boldsymbol{\Delta}_\theta \to \mathbf{0}$ is not strictly ensured by Theorem 2, although $\boldsymbol{\Delta}_\theta = \mathbf{0}$ satisfies (43) for $t \to \infty$, but from (44) it follows that $\boldsymbol{\theta}_2$ must converge to a limit value. Unlike the non-adaptive case, the choice of α in the proof of Theorem 2 is not arbitrary, since this parameter influences the adaptation rate in (44) and has to be properly chosen.

4.3. Avoiding singularity of $\mathbf{B}(\mathbf{q})$ in the adaptive algorithm

A practical implementation of the adaptive controller may encounter two difficulties. First, $\mathbf{B}(\mathbf{q}) > 0$ is not guaranteed for all $t \geq 0$ due to unknown value of $\boldsymbol{\theta}_2$. Since the singularity of $\mathbf{B}(\mathbf{q})$ can occur only if some dynamical parameters in $\boldsymbol{\theta}_2$ are negative or zero, this problem can be easily solved by adding constraints and using saturations in the integration of (44) as described in the following paragraph.

Consider the constraints written in the vector form $\boldsymbol{\theta}_2 \geq \boldsymbol{\theta}_{2min}$. If some of these constrains is active (i.e. is satisfied as equality), the corresponding component of $\dot{\boldsymbol{\theta}}_2$ is set zero if it is originally negative, so that $\boldsymbol{\theta}_2$ cannot further decrease in this direction. Analogously, in the case $\boldsymbol{\theta}_2 \leq \boldsymbol{\theta}_{2\max}$, the active component of $\dot{\boldsymbol{\theta}}_2$ is set zero if it is positive. This modification of the adaptation law preserves validity of Theorem 2, as specified by the following result.

Proposition 2. Assume that that the actual vector of the parameters $\boldsymbol{\theta}_2 + \boldsymbol{\Delta}_\theta$ satisfies

$$\boldsymbol{\theta}_{2min} \leq \boldsymbol{\theta}_2 + \boldsymbol{\Delta}_\theta \leq \boldsymbol{\theta}_{2\max}. \tag{51}$$

If the integration of (44) uses the saturation bounds $\boldsymbol{\theta}_{2\min}$ and $\boldsymbol{\theta}_{2\max}$, then Theorem 2 keeps valid.

Proof. Denote $(.)_i$ the *i*-th component of the vector argument. It is easily seen from (51) that

$$\left(\beta \mathbf{Y}_2^T(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\mathbf{B}^{-1}(\mathbf{q})[\dot{\mathbf{e}} + \alpha\mathbf{e}]\right)_i \times (\boldsymbol{\Delta}_\theta)_i \leq 0 \tag{52}$$

for i such that $(\dot{\boldsymbol{\theta}}_2)_i$ is set zero due to the saturation. Note that for such indices it is $(\sigma\dot{\boldsymbol{\Delta}}_\theta^T)_i = \mathbf{0}$, because $\dot{\boldsymbol{\Delta}}_\theta = -\dot{\boldsymbol{\theta}}_2$. For the other indices it is

$$\left(\beta \mathbf{Y}_2^T \mathbf{B}^{-1}[\dot{\mathbf{e}} + \alpha\mathbf{e}] + \sigma\boldsymbol{\Gamma}\dot{\boldsymbol{\Delta}}_\theta\right)_i = 0 \tag{53}$$

due to (44), so

$$\left(\beta\left(\dot{\mathbf{e}}+\mathbf{e}\right)^T \mathbf{B}^{-1}\mathbf{Y}_2 + \sigma\dot{\boldsymbol{\Delta}}_\theta^T\boldsymbol{\Gamma}\right)\boldsymbol{\Delta}_\theta \le 0 \qquad (54)$$

holds instead of (48) in the proof of Theorem 2. Consequently, $\dot{W} \le 0$ is preserved. The rest of the proof of Theorem 2 remains unchanged. □

4.4. Computation of the regressor $\mathbf{Y}_2\left(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}}\right)$

The second problem is the fact that the measurements of $\ddot{\mathbf{q}}$ needed in the regressor are usually not directly available. This problem has been discussed in literature in connection with the adaptive extensions of the computed-torque method. For instance, in [10] the idea of filtering the equation (9) is used. In [11], the measurements of $\ddot{\mathbf{q}}$ are avoided by means of a transformation of the robot dynamics using a suitable auxiliary variable. The adaptive algorithm described in this section is intended mainly as a straightforward extension of the method proposed in Section 3. Therefore, further modifications seem to be beyond the scope of this paper, although they may by a topic for future work.

Nevertheless, although it is usually not possible to obtain $\ddot{\mathbf{q}}$ by a direct differentiation of the $\dot{\mathbf{q}}$ signal, digital implementation enables to compute $\ddot{\mathbf{q}}$ along with $\dot{\mathbf{q}}$ indirectly on the basis of the position measurements. Considering the robot trajectory sufficiently smooth, one possibility is to use a polynomial interpolation through several last measured values of $\mathbf{q}$. Let $t_k = k\Delta$, where $\Delta > 0$ is a constant scan period, and denote $\mathbf{v}_k$ and $\mathbf{a}_k$ the estimated values of $\dot{\mathbf{q}}(t_k)$ and $\ddot{\mathbf{q}}(t_k)$, respectively. For instance, cubic interpolation yields

$$\mathbf{v}_k = \frac{22\mathbf{q}_k - 36\mathbf{q}_{k-1} + 18\mathbf{q}_{k-2} - 4\mathbf{q}_{k-3}}{12\Delta},\ \mathbf{a}_k = \frac{2\mathbf{q}_k - 5\mathbf{q}_{k-1} + 4\mathbf{q}_{k-2} - \mathbf{q}_{k-3}}{\Delta^2}, \qquad (55)$$

where $\mathbf{q}_k = \mathbf{q}(t_k)$. Usability of formulas (55) for implementation of the adaptive controller described in this section was verified by simulations, see Section 5.3. However, working with a larger number of recent points requires a shorter scan period, which poses higher demands on the resolution of the position sensors. For the best precision, the information from the sensors such as optical encoders should be processed at variable scan instants generated by the encoder events, as discussed in [28].

5. A demonstration of the method

5.1. The robot model

The presented methods are demonstrated on the 5-degree-of-freedom (DOF) robot model in Figure 1. For simplicity the inertial effects are considered concentrated in the mass points m_k, one for each robot link. The vector of generalized coordinates is defined as $\mathbf{q} = \left[\varphi, \psi, \vartheta, \eta, \varepsilon\right]^T$. The values of the robot mechanical parameters are listed in Tab. 1. The friction parameters were chosen as $\mathbf{F}_V = 2 \times \mathrm{diag}\left(2,1,1,1,1\right)$ and $\mathbf{F}_C = \mathbf{0}$.

Table 1. The Robot Mechanical Parameters

Parameter	Value	Unit
l_1	0.5	m
l_2	0.5	m
l_3	0.4	m
l_4	0.15	m
l_5	0.3	m
e	0.2	m
m_1	2.0	kg
m_2	1.0	kg
m_3	1.0	kg
m_4	0.3	kg
m_5	0.7	kg

The model has been constructed to roughly represent the dynamics of in practice the most common anthropomorphic-type robot manipulators. Precise models of these robots seem to be too complex for the purposes of this work. Although the robot control algorithms are often demonstrated only on 2-link or 3-link models in literature, the differences between the control methods are more apparent for a higher DOF model. The fact that the inertial effects are concentrated in the mass points makes computation of the terms in the equation of motion (1) much simpler, though the robot dynamics is not significantly simplified from the control viewpoint.

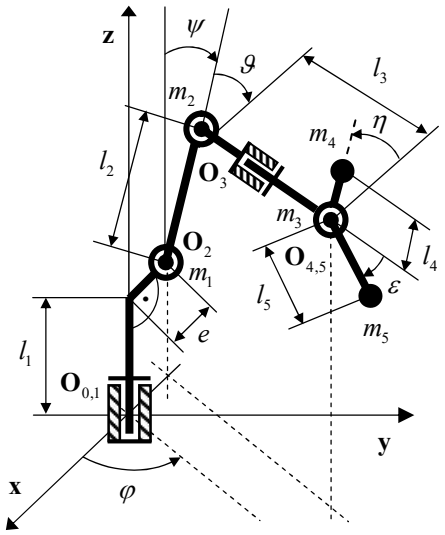

Fig. 1. The 5-DOF robot mass-point model

The symbols $\mathbf{O}_k$ in Fig. 1 denote chosen origins of the coordinate systems of individual links. The identical origins $\mathbf{O}_0$ and $\mathbf{O}_1$, as well as $\mathbf{O}_4$ and $\mathbf{O}_5$, are denoted by the symbols $\mathbf{O}_{0,1}$ and $\mathbf{O}_{4,5}$. The mass point positions $\mathbf{p}_k$ in the link coordinate systems are

$$\mathbf{p}_1 = \begin{bmatrix} 0 \\ e \\ l_1 \end{bmatrix}, \ \mathbf{p}_2 = \begin{bmatrix} 0 \\ 0 \\ l_2 \end{bmatrix}, \ \mathbf{p}_3 = \begin{bmatrix} 0 \\ 0 \\ l_3 \end{bmatrix}, \ \mathbf{p}_4 = \begin{bmatrix} 0 \\ l_4 \\ 0 \end{bmatrix}, \ \mathbf{p}_5 = \begin{bmatrix} 0 \\ 0 \\ l_5 \end{bmatrix}. \tag{56}$$

Denote $\mathbf{p}_k^0$ the mass point positions with respect to the base frame. By expressing $K(\mathbf{q}, \dot{\mathbf{q}})$ and $P(\mathbf{q})$ it is easily possible to obtain

$$\mathbf{B}(\mathbf{q}) = \sum_{k=1}^{n} m_k \left(\partial \mathbf{p}_k^0 / \partial \mathbf{q} \right)^T \left(\partial \mathbf{p}_k^0 / \partial \mathbf{q} \right) \tag{57}$$

and

$$\mathbf{g}^T(\mathbf{q}) = [0, 0, g_c] \sum_{k=1}^{n} m_k \left(\partial \mathbf{p}_k^0 / \partial \mathbf{q} \right), \tag{58}$$

where g_c denotes the gravity constant. To compute the Jacobians $\partial \mathbf{p}_k^0 / \partial \mathbf{q}$, $k = 1, ..., n$, in an efficient manner, the positions $\mathbf{p}_k^0$ are expressed as

$$\mathbf{p}_k^0 = \mathbf{O}_k^0 + \mathbf{T}_k^0 \mathbf{p}_k, \tag{59}$$

where $\mathbf{O}_k^0$ and $\mathbf{T}_k^0$ denote the link origin and the 3×3 orthogonal link rotation matrix with respect to the base frame, respectively. First, the terms $\mathbf{O}_k^0$ and $\mathbf{T}_k^0$ in (59) can be computed recursively from the equations

$$\mathbf{O}_k^0 = \mathbf{O}_{k-1}^0 + \mathbf{T}_{k-1}^0 \mathbf{r}_k \,,\ \mathbf{O}_0^0 = \mathbf{0} \tag{60}$$

and

$$\mathbf{T}_k^0 = \mathbf{T}_{k-1}^0 \mathbf{T}_k \,,\ \mathbf{T}_0^0 = \mathbf{I} \,, \tag{61}$$

where $\mathbf{r}_k$ and $\mathbf{T}_k$ are the offset and the rotation matrix of the k-th link frame with respect to the previous link frame, respectively. For the situation in Fig. 1 it is $\mathbf{r}_1 = \mathbf{r}_5 = \mathbf{0}$, and $\mathbf{r}_k = \mathbf{p}_{k-1}$ for $k = 2,...,4$. The rotation matrices $\mathbf{T}_k$ are in the form

$$\mathbf{T}_1 = \mathbf{R}_z(\varphi),\ \mathbf{T}_2 = \mathbf{R}_y(\psi),\ \mathbf{T}_3 = \mathbf{R}_y(\vartheta),\ \mathbf{T}_4 = \mathbf{R}_z(\eta),\ \mathbf{T}_5 = \mathbf{R}_y(\varepsilon), \tag{62}$$

where

$$\mathbf{R}_z(x) \overset{def}{=} \begin{bmatrix} \cos x & -\sin x & 0 \\ \sin x & \cos x & 0 \\ 0 & 0 & 1 \end{bmatrix},\ \mathbf{R}_y(x) \overset{def}{=} \begin{bmatrix} \cos x & 0 & \sin x \\ 1 & 0 & 1 \\ -\sin x & 0 & \cos x \end{bmatrix}. \tag{63}$$

Further, based on (59)-(61), the Jacobians $\partial \mathbf{p}_k^0 / \partial \mathbf{q}$ in (57) and (58) can be obtained by differentiating (59) with respect to q_j, $k = 1,...,n$, where $\partial \mathbf{O}_k^0 / \partial q_j$ and $\partial \mathbf{T}_k^0 / \partial q_j$ for $k \geq 1$ are computed recursively using

$$\frac{\partial \mathbf{O}_k^0}{\partial q_j} = \frac{\partial \mathbf{O}_{k-1}^0}{\partial q_j} + \frac{\partial \mathbf{T}_{k-1}^0}{\partial q_j} \mathbf{r}_k \,,\ \frac{\partial \mathbf{T}_k^0}{\partial q_j} = \frac{\partial \mathbf{T}_{k-1}^0}{\partial q_j} \mathbf{T}_k + \mathbf{T}_{k-1}^0 \frac{\partial \mathbf{T}_k}{\partial q_j} \,. \tag{64}$$

In (64), it is $\partial \mathbf{O}_0^0 / \partial q_j = \mathbf{0}$ and $\partial \mathbf{T}_0^0 / \partial q_j = \mathbf{0}$. The terms $\partial \mathbf{T}_k / \partial q_j$ are zero for $k \neq j$, and $\partial \mathbf{T}_k / \partial q_k$ can be directly computed from (62)-(63).

The matrix $\mathbf{C}(\dot{\mathbf{q}}, \mathbf{q})$ was computed using (3). Although a recursive computation is possible as well, for a simplification the terms $\partial \mathbf{B} / \partial q_i$ were obtained from (57) by numerical differentiation based on the 5-point formula [29]

$$f'(x) = \frac{-f(x+2h) + 8f(x+h) - 8f(x-h) + f(x-2h)}{12h} + O(h^4). \tag{65}$$

5.2. Simulated results – control with a precise model

The chosen joint initial position $\mathbf{q}_0$ and target position $\mathbf{q}_f$ for the robot model described in Section 5.1 are listed in Tab.2.

Table 2. The joint initial and target positions in radians

Point	φ	ψ	ϑ	η	ε
$\mathbf{q}_0$	-π/2	2π/3	5π/6	0	0.5
$\mathbf{q}_f$	π/2	0	π/4	π	-π/2

As the reference trajectory $\mathbf{q}_d$ the piecewise-linear function

$$\mathbf{q}_d(t) = \mathbf{q}_0 + (\mathbf{q}_f - \mathbf{q}_0)\min\{t/t_r, 1\} \tag{66}$$

is used in this sub-section, where $t_r = 0.5s$ represents the desired time of motion.

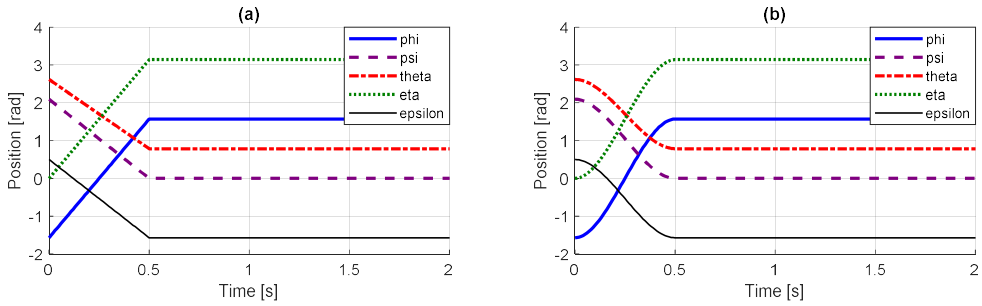

Fig. 2. The reference trajectory $\mathbf{q}_d$: (a) non-smooth, (b) polynomial

The reference trajectory, shown in Fig. 2a), has been chosen non-smooth to make the differences between the methods more apparent. From the control viewpoint, a step change of $\dot{\mathbf{q}}_d$ has the same effect as a step change of velocity during the motion. Such a kind of disturbance is not typical at normal operation, but it could represent, for instance, a momentary contact with the environment. Due to the discontinuity of $\dot{\mathbf{q}}_d$ the compensation (18) cannot be applied at $t = 0$ and $t = t_r$, where $\ddot{\mathbf{q}}_d$ does not exist. Since $\ddot{\mathbf{q}}_d = 0$ everywhere else, $\ddot{\mathbf{q}}_d$ can be omitted in (18), and also in (11). The corresponding initial and target position

of the robot in the operational space, together with two intermediate positions on the reference trajectory, are shown in Fig. 3.

The controller settings in (11) and (14) are chosen in the form

$$\mathbf{R}_0 = k_R \mathbf{I}\,,\ \mathbf{R}_1 = k_R T_R \mathbf{I}\,, \tag{67}$$

where $k_R > 0$ and $T_R \geq 0$. Note that $\mathbf{R}_0\mathbf{e} + \mathbf{R}_1\dot{\mathbf{e}}$ is the vector of outputs of n PD controllers with gain k_R and derivative time constant T_R. The $\dot{\mathbf{e}}$ signal is computed from $\mathbf{e}$ numerically processed by the filter $(0.002s+1)^{-1}$.

The inertia parameter β is computed using (24) for $\mu^{-1} = \mu_1 \|\dot{\mathbf{q}}\|$, where $\mu_1 = 10$. Considering roughly $\|\dot{\mathbf{q}}\| \leq 10$, this corresponds to $\mu \geq 0.01s$. Higher values of μ_1 produce somewhat faster responses, but the settings $\mu_1 > 100$ are not suitable since they produce bold spikes in the control signals.

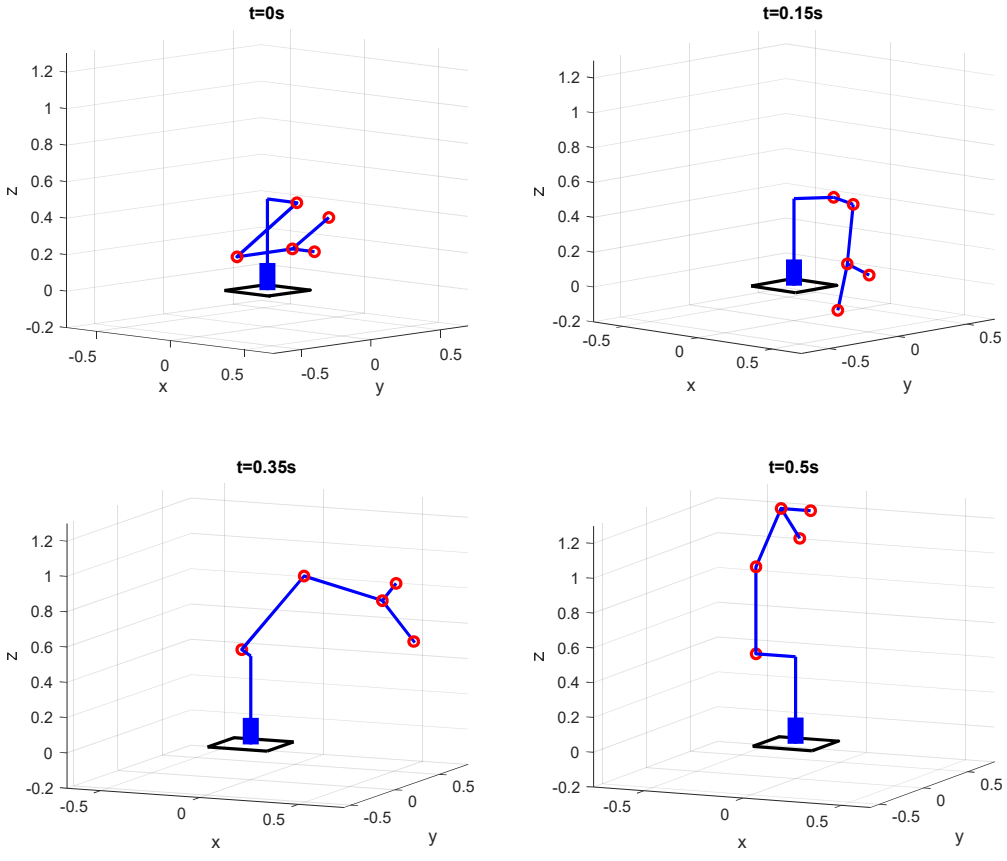

Fig. 3. Four positions of the robot on the reference trajectory in the operational space

The 4th-order Runge-Kutta method [29] was used for simulation with the step size 1×10^{-4}, using MATLAB Simulink software. Visually the same results were also obtained by the variable-step Dormand-Prince method [30], which is default in MATLAB Simulink, and other methods. However, since in the adaptive version of the algorithm the measurements of $\ddot{\mathbf{q}}$ in the regressor are generated using formula (55) by time discretization as piecewise constant, a fixed-step method, such as the 4th-order Runge-Kutta one, seems to be preferable.

The quality of control is evaluated by means of the overall integral absolute error index

$$IAE = \int_0^{t_s} \sum_{i=1}^{n} \left| e_i(t) \right| dt \tag{68}$$

where t_s is the simulation horizon. A lower value corresponds to more precise reference trajectory tracking. A goal is to compare the proposed method especially with the classical computed-torque method. However, it is not objective to simply show the responses for the same settings of k_R and T_R, because both the methods generate different ranges of the control signals. In both the cases, the response can be made faster by increasing k_R, which at the same time increases the control signals magnitudes. The parameter T_R has a damping effect, but slows down the response.

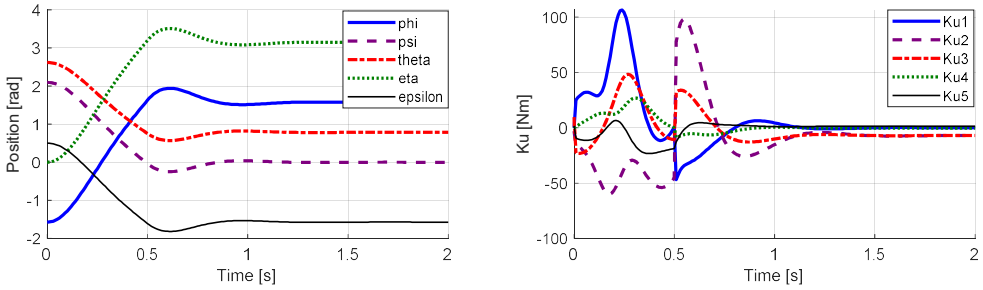

Fig. 4. Simulated histories of joint position and control: the computed-torque controller, $k_R = 100$ and $T_R = 0.1$

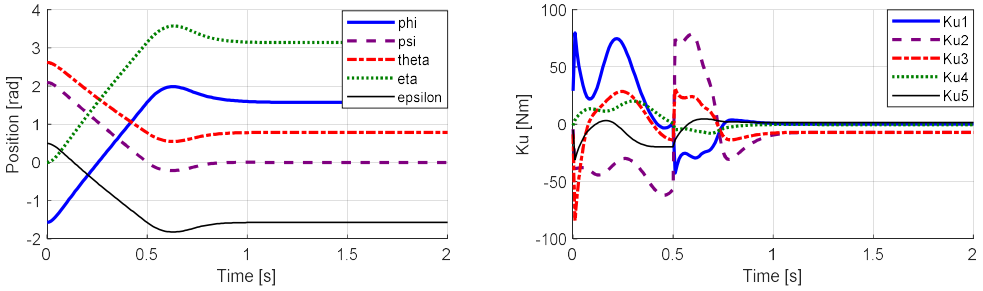

Fig. 5. Simulated histories of joint position and control: the variable-inertia controller, $k_R = 100$ and $T_R = 0.1$

Figure 4 shows the histories of the robot joint positions and corresponding histories of the signal **Ku** in the case of the classical computed-torque controller for $k_R = 100$ and $T_R = 0.1$. The time constant T_R is set so that the responses are nearly non-oscillating. Fig. 5 shows the responses corresponding to the variable-inertia method (14), for the same controller settings. The corresponding values of the IAE index are listed in Tab. 3. Although the settling times in Figs. 4 and 5 are similar, the responses in Fig. 5 are more damped, which indicates that a lower value of T_R can be used, whereas in the computed-torque case using a lower T_R already leads to oscillatory behavior.

Using a lower value of T_R in Fig. 5 attenuates the peaks of the control signals at $t = 0$ caused by too strong derivative action of the controller and enables to set a higher gain k_R to achieve a faster response, keeping similar ranges of the control signals as in Fig. 4. These modifications of the settings lead to the responses in Fig. 6. The corresponding parameters of the controller are $k_R = 140$ and $T_R = 0.05$. It can be seen that the tracking capabilities have been improved significantly in comparison to the situation in Fig. 4. This is also indicated by a much lower value of the IAE index in Tab. 3.

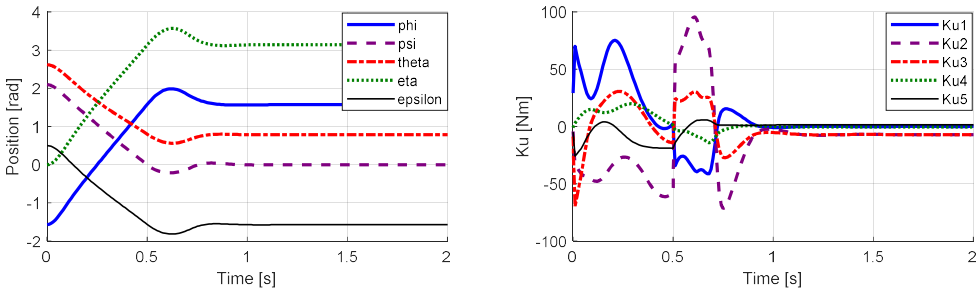

Fig. 6. Simulated histories of joint position and control: the variable-inertia controller, $k_R = 140$ and $T_R = 0.05$

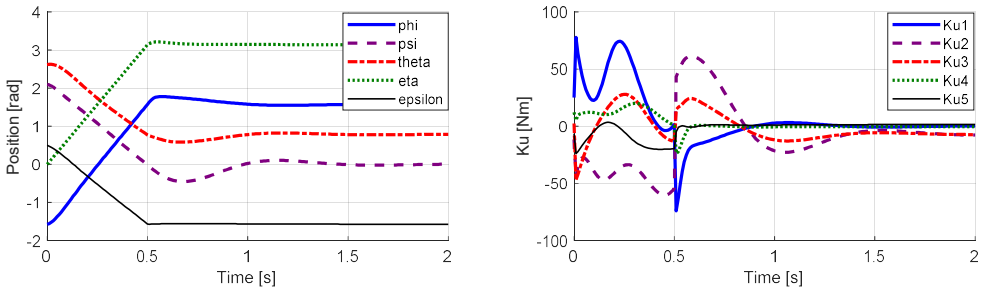

Fig. 7. Simulated histories of joint position and control: the PD+ controller, $k_R = 100$ and $T_R = 0.1$

Figure 7 shows the behavior in the case of the PD+ controller (13), for the settings (67), where $k_R = 100$ and $T_R = 0.1$. It can be seen that some components of $\mathbf{q}$ settle rather slowly, whereas the others respond very fast. The oscillations can be damped by increasing T_R to about $T_R = 15$, but this slows down the overall response and creates bold extremes of the control signals at $t = 0$ and $t = t_r$. It is possible to set different controller gains for each component of $\mathbf{q}$, but suitable settings will depend on the reference trajectory.

Table 3. The values of the IAE index corresponding to plots in Section 5.2

Fig. no.	**Method**	**Motion**	**k_R**	**T_R**	**IAE**
4	computed-torque	full-range	100	0.1	0.669
5	variable-inertia	full-range	100	0.1	0.449
6	variable-inertia	full-range	140	0.05	0.372
7	PD+	full-range	100	0.1	0.401
8	computed-torque	half-range	100	0.1	0.335
9	variable-inertia	half-range	140	0.05	0.279

Figures 8 and 9 show the responses corresponding to the computed-torque and variable-inertia methods in the case of half-range motion, where $\left(\mathbf{q}_0 + \mathbf{q}_f\right)/2$ is used instead of the original $\mathbf{q}_0$. More precise reference trajectory tracking and a lower value of the IAE index again corresponds to Fig. 9. To obtain similar ranges of the control signals in Figures 8 and 9, the gain k_R in Fig. 9 has been multiplied by 1.4, which is the same value as in Fig. 6. This is explained by the histories of β in the case of full-range and half-range motion, which are shown in Fig. 10. Since the term $\mathbf{R}_0\mathbf{e} + \mathbf{R}_1\dot{\mathbf{e}}$ is multiplied by $\beta^{-1}\mathbf{B}$ in (14), k_R has to be adjusted with respect to the values of β along the trajectory to achieve similar ranges of the control signals at both the methods. It seems that the controller gain k_R should be set roughly proportional to the maximum of β along the reference trajectory, which well corresponds to the values of k_R chosen in the simulations in Figs. 6 and 9.

On the other hand, as discussed in Section 3.2 in more detail, at the computed-torque method the ranges of the control signals for different reference trajectories are influenced by

the effect of $\mathbf{B}(\mathbf{q})$ in (10). At the variable-inertia method this effect is partially suppressed by the multiplication of $\mathbf{B}$ with β^{-1}, so for fixed parameters $\mathbf{R}_0$ and $\mathbf{R}_1$ the feedback control signals magnitudes are somewhat less dependent on the choice of the reference trajectory.

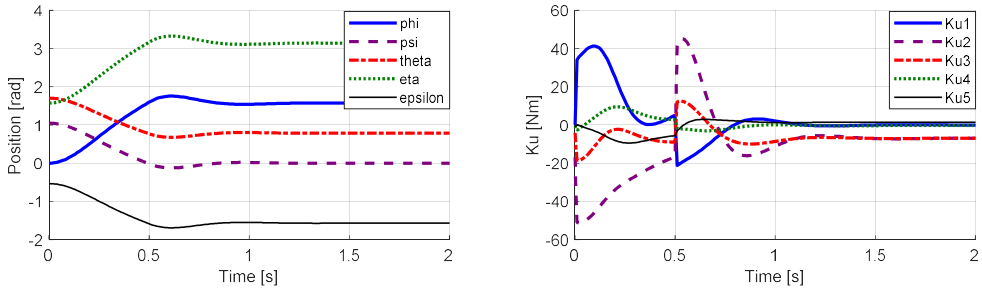

Fig. 8. Simulated histories of joint position and control, half-range motion: the computed-torque controller, $k_R = 100$ and $T_R = 0.1$

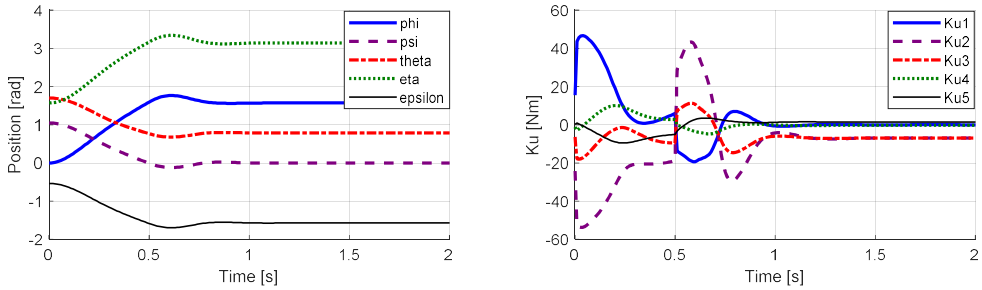

Fig. 9. Simulated histories of joint position and control, half-range motion: the variable-inertia controller, $k_R = 140$ and $T_R = 0.05$

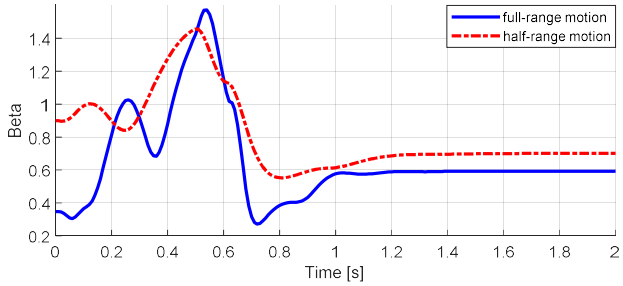

Fig. 10. The plots of β corresponding to the responses in Figs. 6 and 9

5.3. Simulated results – errors in the model and adaptive extensions

To demonstrate the effect of the adaptive extensions, the mass m_5 in the robot model is considered unknown, with initial value of the estimate $\theta_2 = 0.5$ (note that θ_2 is scalar in this case). The points $\mathbf{q}_0$ and $\mathbf{q}_f$, μ_1, and the simulation parameters are chosen the same as in Section 5.2. The reference trajectory for each joint has been chosen in the form of the 3rd-order polynomial in $[0, t_r]$ that fits the prescribed initial and target position and has zero velocity at the boundary points:

$$\mathbf{q}_d(t) = \mathbf{q}_0 + (\mathbf{q}_f - \mathbf{q}_0)(t/t_r)^2 [3 - 2(t/t_r)]. \tag{69}$$

The time of motion is set as $t_r = 0.75s$. For $t > t_r$ it is $\mathbf{q}_d(t) = \mathbf{q}_f$. The reference trajectory is shown in Fig. 2b).

First, Figure 11 shows the responses for the controller (14) without adaptation, for $k_R = 100$ and $T_R = 0.1$. The same controller settings are used in the other simulations in this section as well. The corresponding value of the IAE index equals 0.244.

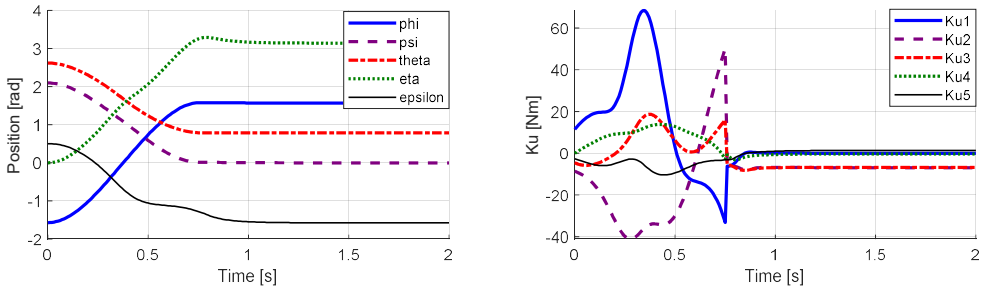

Fig. 11. Simulated histories of joint position and control: errors in the model, $k_R = 100$ and $T_R = 0.1$

Further, the function $\sigma(t)$ in (44) has been chosen in the form (45), where $\nu = 3$ and $\sigma_0 = 10^{-3}$. The parameter σ_1 was set so that $\sigma_1 t_r^{\nu} = 1$, which yields $\sigma_1 = 2.37$. This choice ensures $\sigma(t_r) \approx \sigma(0)/2$. The value of $\ddot{\mathbf{q}}$ in the regressor has been computed as piecewise constant using the cubic interpolation-based formula (55) for $\Delta = 0.002\,s$. The function $\mathbf{Y}_2(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ was not expressed directly from the model. Instead, assume that $\mathbf{Ku}_1$ and $\mathbf{Ku}_2$ are obtained from the model for given $\mathbf{q}, \dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$, and two arbitrary different values θ_{21} and θ_{22} of the parameter θ_2. Since

$$\left[\mathbf{Y}_1,\mathbf{Y}_2\right]\begin{bmatrix}\boldsymbol{\theta}_1\\ \theta_{21}\end{bmatrix}-\left[\mathbf{Y}_1,\mathbf{Y}_2\right]\begin{bmatrix}\boldsymbol{\theta}_1\\ \theta_{21}\end{bmatrix}=\mathbf{Y}_2\times\left(\theta_{21}-\theta_{22}\right)=\mathbf{K}\left(\mathbf{u}_1-\mathbf{u}_2\right),\qquad(70)$$

it was possible to compute $\mathbf{Y}_2$ as $\mathbf{K}\left(\mathbf{u}_1-\mathbf{u}_2\right)/\left(\theta_{21}-\theta_{22}\right)$. The bounds of θ_2 were chosen as $\theta_{2\min}=0.2$, $\theta_{2\max}=0.8$. The matrix $\boldsymbol{\Gamma}$ in (44) is considered in the form $\boldsymbol{\Gamma}=\gamma^{-1}\mathbf{I}$, where $\gamma>0$ is a parameter. Figure 12 shows the histories of positions and the control for $\alpha=5$ in (44) and $\gamma=0.02$. The corresponding IAE value is 0.0342.

The parameters γ and α influence the rate of adaptation and their proper setting is crucial for quality of control. Fig. 13 shows the histories of θ_2 for $\alpha=5$ and several values of γ, whereas Fig. 14 shows the histories of θ_2 for $\gamma=0.02$ and different values of α. The estimate of θ_2 should converge to 0.7, which is the actual value of this parameter.

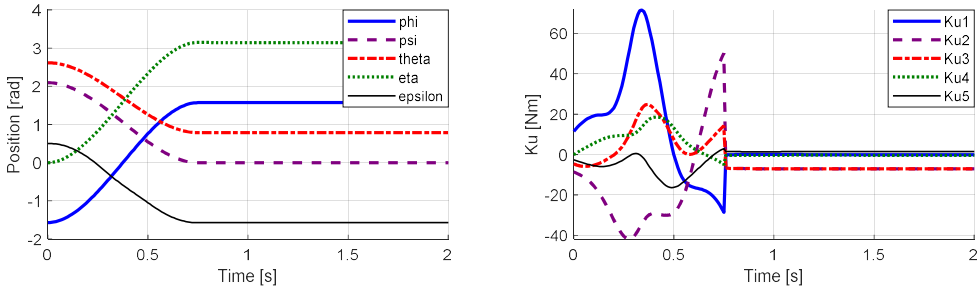

Fig. 12. Simulated histories of joint position and control: the adaptive controller, $\gamma=0.02$, $\alpha=5$, $k_R=100$ and $T_R=0.1$

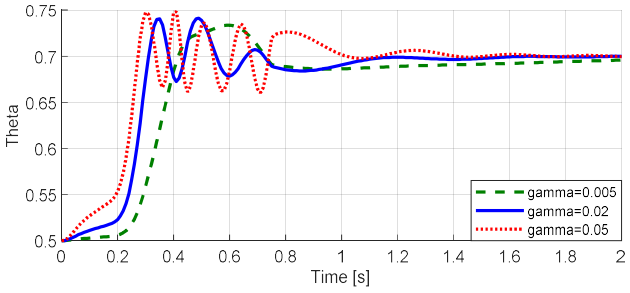

Fig. 13. Simulated histories of θ_2 for $\alpha=5$ and different values of γ

Note that the history of θ_2 for $\alpha=15$ is already influenced by the bound $\theta_{2\max}=0.8$. Apparently, $\alpha>0$ is necessary to achieve zero steady-state control error. Increasing α speeds up the convergence, but it also makes the estimate θ_2 more oscillatory. Too high α will result in instability, which corresponds to theoretical conclusions. The effect of γ is

similar, but it seems to more likely influence the frequency of oscillations of θ_2, whereas α increases the amplitude.

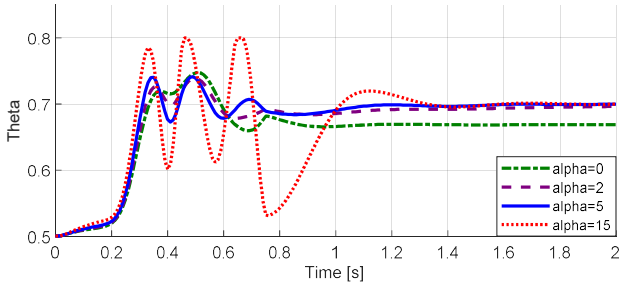

Fig. 14. Simulated histories of θ_2 for $\gamma = 0.02$ and different values of α

6. Conclusions

The classical computed-torque method of motion control of robot manipulators utilizes inner feedback loop to transform the robot dynamics into linear and decoupled one, which enables to design the feedback controller in a simple way. However, since the inner feedback signal is not taken into account in the outer-loop controller design objectives, the control signals, which are limited in real conditions, are not generated in an efficient manner. A goal of this work was to modify the computed-torque method to utilize the available ranges of the control signals more effectively.

The proposed method of motion control, defined by the feedback control law (14) and the reference trajectory compensation (18), utilizes the inner feedback to partially decouple the inertial effects of individual links, but at the same time it minimizes the influence of the inner feedback component of the control input. In comparison to the classical computed-torque control method, the control input more reflects the actual value of the control error, which increases the control efficiency for given magnitude of the control signals and helps to prevent from reaching saturation limits of the actuators.

The closed-loop behavior depends on the choice of a scalar parameter β, which represents the inner-loop system inertia. It is advantageous to consider β variable, dependent on $\mathbf{q}$ and $\dot{\mathbf{q}}$, which enables to minimize the influence of the inner feedback signal. However, the optimal setting of β given by (22) had to be modified to ensure finite bounds of $\dot{\beta}$, which is necessary to ensure the closed-loop stability. The proposed solution is based on solving the differential equation (24) in real time.

Unlike the conventional computed-torque method, the outer control loop is no longer linear, but it has been shown that semi-global uniform asymptotical stability of the control error during the reference trajectory tracking is achieved by setting $r_{0i} > 0$ and r_{1i} sufficiently high, where r_{0i} and r_{1i} are the diagonal elements of the diagonal controller matrices $\mathbf{R}_0$ and $\mathbf{R}_1$.

Simulations confirm that for comparable control effort the described 'variable-inertia' method is usually capable to provide significantly better performance in comparison to the classical computed-torque control algorithm. In comparison to the PD+ motion control method, it could be preferable especially in the situations with external disturbances, where the PD+ algorithm can generate slow responses in some joints due to dynamical coupling.

For the situations when all robot model parameters are not known precisely, an adaptive extension of the method has been proposed. The adaptive version of the algorithm poses more stringent requirements on the choice of the controller parameters, and depends on additional parameters that have to be properly set, but it provides a theoretical guarantee of uniformly stable tracking and convergence of the control error to zero if the gains in $\mathbf{R}_0$ and $\mathbf{R}_1$ are set sufficiently high. The effect of the adaptation parameters α and $\boldsymbol{\Gamma}$ has been demonstrated by simulations. The non-singularity of $\mathbf{B}(\mathbf{q})$ of the model is ensured by defining bounds of the adapted parameters and using saturations in the integration of equation (44).

Future work could be focused on enhancements of the adaptive version of the algorithm, especially towards avoiding the measurements of $\ddot{\mathbf{q}}$ in the regressor, and on extensions considering saturations of the control signals.

Statements and declarations

The author declares that there are no relevant financial or non-financial competing interests.

References

[1] Sicilliano, B., Sciavicco, L., Oriollo, G.: Robotics: Modelling, Planning and Control. Springer –Verlag (2009)

[2] Chung, W., Fu, L.C., Hsu, S. H.: Motion Control. In: Siciliano, B., Khatib, O. (eds.) Springer Handbook of Robotics, Part A Robotics Foudations. Springer-Verlag (2008)

[3] Kelly, R., Santibánez, V., Loría, A.: Control of Robot Manipulators in Joint Space. Springer-Verlag, London (2005)

[4] Cervantes, I., Alvarez-Ramirez, J.: On the PID tracking control of robot manipulators. Systems & Control Letters. 42, 37–46 (2001). https://doi.org/10.1016/S0167-6911(00)00077-3

[5] Paden B., Panja R.: Globally asymptotically stable PD+ controller
for robot manipulators. International Journal of Control, 47(6), 1697–1712 (1988).
https://doi.org/10.1080/00207178808906130
[6] Chávez-Vázquez, S., Gómez-Aguilar, J.F., Lavín-Delgado, J.E., Escobar-Jiménez, R.F., Olivares-Peregrino,V.H.: Applications of Fractional Operators in Robotics: A Review. Journal of Intelligent & Robotic Systems. 104 (63) (2022). https://doi.org/10.1007/s10846-022-01597-1
[7] Lavín-Delgado, J.E., Solís-Pérez, J.E., Gómez-Aguilar, J.F., Escobar-Jiménez, R.F.: Trajectory tracking control based on non-singular fractional derivatives for the PUMA 560 robot arm. Multibody System Dynamics. 50, 259–303 (2020). https://doi.org/10.1007/s11044-020-09752-y
[8] Craig, J.J., Ping, H., Sastry, S.S.: Adaptive control of mechanical manipulators. International Journal of Robotics Research. 6 (2), 16–28 (1987). https://doi.org/10.1177/027836498700600202
[9] Spong, M.W., Ortega, R.: On adaptive inverse dynamic control of rigid robots. IEEE Transactions of Automatic Control. 35(1), 92-95 (1990). https://doi.org/10.1109/9.45152
[10] Middleton, R.H., Goodwin, G.C.: Adaptive computed torque control for rigid link manipulations. Systems & Control Letters. 10(1), 9–16 (1988). https://doi.org/10.1016/0167-6911(88)90033-3
[11] Slotine, J.J., Li, W.: On the adaptive control of robot manipulators. International Journal of Robotics Research. 6(3), 49-59 (1987). https://doi.org/10.1177/027836498700600303
[12] Ortega, R., Spong, M.W.: Adaptive motion control of rigid robots: A tutorial. Automatica. 25(6), 877-888 (1989). https://doi.org/10.1109/CDC.1988.194594
[13] Lavín-Delgado, J.E., Chávez-Vázquez, S., Gómez-Aquilar, J.F, Delgado-Reyes, G. and Ruíz-Jaimes, M.A.: Fractional-order passivity-based adaptive controller for a robot manipulator type SCARA. Fractals. 28 (8), (2020). https://doi.org/10.1142/S0218348X20400083
[14] Spong, M.W.: On the robust control of robot manipulators. IEEE Transactions of Automatic Control. 37(11), 1782-1786 (1992). https://doi.org/10.1109/9.173151
[15] Wang, H., Xie, Y.: Adaptive inverse dynamics control of robots with uncertain kinematics and dynamics. Automatica. 45 (9), 2114-2119 (2009). https://doi.org/10.1016/j.automatica.2009.05.011
[16] Chen, Y., Ma, G., Lin, S., Ning, S., Gao, J.: Computed-torque plus robust adaptive compensation control for robot manipulator with structured and unstructured uncertainties. IMA Journal of Mathematical Control and Information. 33 (1), 37–52 (2016). https://doi.org/10.1093/imamci/dnu024
[17] Sabet, S., Poursina, M.: Computed torque control of fully-actuated
nondeterministic multibody systems. Multibody system dynamics. 41, 347–365 (2017).
https://doi.org/10.1007/s11044-017-9577-4
[18] Zavala-Rio, A., Santibanez, V.: Simple extensions of the PD-with-gravity-compensation control law for robot manipulators with bounded inputs. IEEE Transactions on Control Systems Technology. 14 (5), 958-965 (2006). https://doi.org/10.1109/TCST.2006.876932
[19] Peng, W., Li, Z., Su, J.: Computed torque control-based composite nonlinear feedback controller for robot manipulators with bounded torques. IET Control Theory and Applications. 3(6), 701-711 (2009).
https://doi.org/10.1049/iet-cta.2008.0259
[20] Su, J., Swevers, J.: Finite-time tracking control for robot manipulators with actuator saturation. Robotics and Computer-Integrated Manufacturing. 30, 91-98 (2014). https://doi.org/10.1016/j.rcim.2013.09.005

[21] Cvejn, J., Zapletal, M.: Enhancing the Robot PD-Type Feedback Control Performance by Means of Inertial Effects Compensation. In: Proc. 2020 4th International Conference on Automation, Control and Robots, Rome, pp. 1-6 (2020). https://doi.org/10.1109/ICACR51161.2020.9265498
[22] Hollerbach, J.M.: A recursive Lagrangian formulation of manipulator dynamics and a comparative study of dynamics formulation complexity. IEEE Transactions on Systems, Man, and Cybernetics. 10, 730–736 (1980). https://doi.org/10.1109/TSMC.1980.4308393
[23] Li, C.J.: A New Lagrangian Formulation of Dynamics for Robot Manipulators. Journal of Dynamic Systems, Measurement, and Control. 111, 559-567 (1989). https://doi.org/10.1115/1.3153092
[24] Fang, Y.J., Basu, A., Fang, X.D.: An Efficient Recursive Approach For Computer Generation Of Manipulator Dynamic Model. Mathematical and Computer Modelling. 20(9), 89-96 (1994). https://doi.org/10.1016/0895-7177(94)00165-0
[25] Horn, R.A., Zhang, F.: Basic Properties of the Schur Complement. In: Zhang, F. (eds) The Schur Complement and Its Applications - Numerical Methods and Algorithms, vol. 4. Springer, Boston (2005)
[26] Marquez, H.J.: Nonlinear control systems. Analysis and design. J. Wiley & sons (2003)
[27] Khalil, H.K.: Nonlinear systems. Third Edition. Prentince Hall (2002)
[28] Merry, R.J.E., van de Molengraft, M.J.G., Steinbuch, M.: Velocity and Acceleration Estimation for Optical Incremental Encoders. In: Proc. of the 17th World Congress, The International Federation of Automatic Control, Seoul, pp. 7570-7575 (2008). https://doi.org/10.3182/20080706-5-KR-1001.01280
[29] Davis, P.J., Polonsky, I.: Numerical Interpolation, Differentiation, and Integration. In: Abramowitz, M., Stegun, I.A. (eds.) Handbook of mathematical functions with formulas, graphs, and mathematical tables, 10th printing. National Bureau of Standards (1972)
[30] Dormand, J.R.: Numerical Methods for Differential Equations: A Computational Approach. CRC Press (1996)