An extremum-seeking control approach for constrained robotic motion tasks

Vasiliki Koropouli[a,1,*], Azwirman Gusrialdi[b,2], Sandra Hirche[c,3], Dongheui Lee[a,4]

[a] *Chair of Automatic Control Engineering, Technische Universität München, Munich, Germany.*
[b] *Dept. of Electrical Engineering and Computer Science, University of Central Florida, USA.*
[c] *Chair of Information-Oriented Control, Technische Universität München, Munich, Germany.*

Abstract

In this article, we propose two adaptive control schemes for multiple-input systems for execution of robot end-effector movements in presence of parametric system uncertainties. The design of these schemes is based on Model Reference Adaptive Control (MRAC) while the adaptation of the controller parameters is achieved by Extremum Seeking Control (ESC). The two control schemes, which are called Multiple-Input ESC-MRAC and Multiple-Input Adaptive-Dynamic-Inversion ESC-MRAC, are suitable for linear and nonlinear systems respectively. Lyapunov and averaging analysis show that the proposed schemes achieve practical asymptotic reference state tracking. The proposed methods are evaluated in simulations and in a real-world robotic experiment.

Keywords: `Model reference adaptive control, state tracking, robot control, extremum seeking`

1. Introduction

Robots are often required to perform constrained motion tasks inside unknown and deformable environments such as while engraving an object's surface or writing on a soft material. In such cases, uncertainty about the end-effector and the environment

[*] Corresponding author
[1] Email: vicky@lsr.ei.tum.de
[2] Email: Azwirman.Gusrialdi@ucf.edu
[3] Email: hirche@tum.de
[4] Email: dhlee@tum.de

makes the execution of these tasks a challenge since the robot should efficiently compensate for this uncertainty to accurately execute a desired motion path. In addition, during movement inside deformable environments, motion dynamics between different directions are physically coupled and this coupling is non-negligible and has to be taken into account for proper system modeling and control. An illustration of an example engraving task where this dynamic coupling occurs, is shown in Fig. 1. In order to deal with uncertain systems, adaptive control is a suitable approach to follow since it is specially tailored to accommodate uncertainties. In this article, we propose two task-space adaptive control schemes for the execution of multi-directional robot end-effector movements inside unknown and deformable environments where the task uncertainty and the physical coupling between different directions of movement are taken into account. Our task-space control approach could be then combined with any inverse-kinematics algorithm for motion of manipulators [1].

In recent robotic research, significant effort has been devoted to learning of control skills from demonstrated data or combining control approaches with learning-by-demonstration for execution of various motion tasks. In [2], motion primitives are learned by demonstration and combined with impedance control to perform physical contact tasks. In [3], dynamic models are learned and combined with a Linear Quadratic Regulator for performing constrained motion tasks by assuming, however, knowledge of the system dynamics. A composite adaptive control scheme is developed in [4] based on locally weighted regeression for state tracking of single-input single-output systems. In [5], following learning of a task-space force control policy from multiple task demonstrations, generalization of this policy to new motions is proposed by combining regression learning with Taylor-like polynomial approximation while, in [6], regression learning of force patterns from single demonstrations is proposed for performance of constrained motion tasks inside deformable environments. In addition, in [7] learning of motion and force regulation skills from multiple demonstrations is proposed for object grasping. However, such approaches depend on the demonstrated data and their ability to deal with uncertainty is obviously rather limited. Another family of works employ reinforcement learning to iteratively accomplish a motion task while optimizing task performance [8, 9]. However, during manipulation of deformable

environments, a motion task has to be successfully accomplished in a single execution in contrast to reinforcement learning which requires multiple trials and causes, in this way, non-desired cumulative deformation of the environment.

Apart from learning-based approaches, adaptive control approaches have also been employed in robotic tasks. In [10], adaptive Jacobian control of robot manipulators with kinematic and dynamic uncertainties is accomplished while in [11] existing adaptive robust control schemes are implemented and evaluated in tracking control of manipulators. Model Reference Adaptive Control (MRAC) is employed in [12] for motion control of an underwater robotic vehicle and in [13] for control of a six degrees-of-freedom robotic manipulator whose dynamics are simulated by a Newton-Euler model.

Among existing adaptive control approaches, MRAC stands out as one of the most well-established and widely used approaches [14, 15]. MRAC assumes that a reference model exists, which expresses the desired response of a system to a reference signal, and ensures that the system follows the reference model. The use of reference models dates back to the foundation of aircraft control and, since then, MRAC has been widely used in multiple applications [12, 13, 16, 17]. Recent work on MRAC has been realized in [17, 18, 19, 20]. In [17], a multiple-input multiple-output MRAC approach is proposed for output tracking by using state feedback. Although MRAC is an appealing approach which is widely employed for tracking, it depends strictly on the system model. To avoid this strict dependence on the system dynamics, combination of Extremum Seeking Control (ESC) with MRAC is proposed in [18] for state tracking of Single-Input Single-Output (SISO) systems. ESC is a model-free optimization scheme [21, 22] and has been employed in various engineering applications such as in formation flight control [23] and tuning of PID controllers [24]. ESC aims at tracking the minimum or maximum (extremum) of the performance function of a system. The ability to express the ideal behavior of a system in terms of a reference model, as in MRAC, is beneficial because the reference model serves in the design of the control law of the system and can regulate its transient behavior [14], [18]. However, MRAC imposes certain matching conditions between the system and its reference model and these conditions may not always be able to be established, which, in turn, makes MRAC inapplicable. In addition, as it is earlier discussed, MRAC is strictly

model-dependent and mainly refers to systems with parametric uncertainty. By combining MRAC with ESC, model-free parameter adaptation is achieved, which can offer certain robustness to non-paramteric uncertainties as well compared to single MRAC. In [19], an ESC-MRAC approach for output tracking of nonlinear MIMO systems is proposed by performing feedback linearization and the initial problem is converted into output tracking of multiple SISO systems and solved by employing the method proposed in [18]. The method in [19] assumes that the control input can be expressed in a specific linear parameterized form while each SISO system optimizes an individual cost function which solely depends on this system's output.

Our present work focuses on the development of an adaptive control approach for reference state tracking of multiple-input systems. This approach refers both to linear and nonlinear systems which involve parametric uncertainty. The goal is that the system's states track some reference states by optimizing a global state cost criterion for the task. The proposed approach is motivated by the scenario of our robotic application where an end-effector is required to make a specific motion in multiple directions of a workspace. To solve the state-tracking problem we combine MRAC with ESC similar to [18]. MRAC addresses linear systems and in order to address nonlinear systems as well, we combine MRAC with Adaptive Dynamic Inversion (ADI). According to ADI, the parametrically-uncertain nonlinearities of the system are taken into account in the control design and its unknown weighting parameters adapt in real-time to compensate for the relative uncertainty. The design of the proposed control laws is based on MRAC while adaptation of the controller parameters is achieved by ESC.

The principal goal of the work of this article is to solve the state tracking problem for an end-effector with uncertain properties, which interacts with an uncertain environment. Therefore, apart from proposing an adaptive control approach for reference state tracking, we also focus on demonstrating how this approach works in simulations and how it can be used in real-world robotic tasks to improve robot tracking performance in presence of parametric system uncertainties. In particular, the contributions of this article can be described as follows: (1) a multiple-input ESC-MRAC and multiple-input ADI ESC-MRAC scheme are proposed, which achieve practical asymptotic reference state tracking for parametrically-uncertain linear and nonlinear systems respectively,

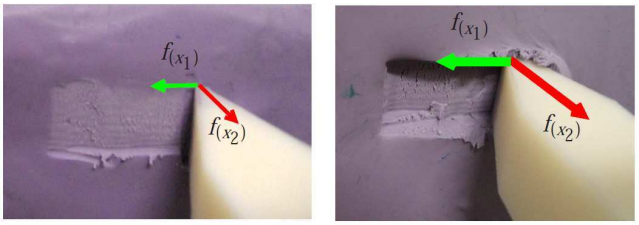


Figure 1: An example of a constrained robotic motion task in deformable environment; engraving is realized at different depths inside a plasticine object. Different environmental disturbance $\{f_{(x_1)}, f_{(x_2)}\}$ is experienced in each case due to the changing manipulating mass. Engraving in a (a) low depth, (b) high depth.

(2) evaluation of the proposed control schemes is provided in simulations and also in a real-world robotic experiment and the proposed control approach is compared with fixed-gain Proportional-Integral-Derivative (PID) control and SISO ESC-MRAC [18]. Our evaluation shows that the proposed schemes not only perform well in simulations but are also successfully employed in real-world robotic tasks for accurate reference state tracking and execution of desired motions inside unknown and deformable objects, see Section 4.2. Preliminary results of the currently presented approach are introduced in [25] and [26].

The present article is structured as follows. First, in Section 2, the motion dynamics of a robot end-effector in contact with the environment are analyzed and, then, two different state tracking problems are formulated for linear and nonlinear systems respectively. In Section 3, two adaptive control schemes are proposed and analyzed to solve the state tracking problems and, in Section 4, the proposed approaches are evaluated in simulations as well as in a real-world robot engraving task.

2. Problem formulation

In our scenario, we consider a robot end-effector which moves in multiple directions of the workspace, and has uncertain physical and geometrical properties. We also assume that the properties of the environment are also uncertain. The goal is to make the end-effector follow a reference position trajectory in multiple dimensions.

The system dynamics are first analyzed and then two state-tracking problems are

formulated, which refer to linear and nonlinear systems respectively.

2.1. Impedance-based system modelling

Let us assume an end-effector interacting with an elastically compliant environment in N directions of movement. The environment behaves as an admittance, i.e. its motion is regulated according to the applied force. The end-effector, in turn, acts as an impedance and its motion dynamics are expressed by the nonlinear impedance law [27], [28]

$$M_d\ddot{\mathbf{x}} + K_D\dot{\mathbf{x}} + K_P\mathbf{x} - \boldsymbol{\epsilon}(\mathbf{x}, \dot{\mathbf{x}}) = \mathbf{f} - \mathbf{f}_e(\mathbf{x}, \dot{\mathbf{x}}) \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^N$ is end-effector position, $\mathbf{f} \in \mathbb{R}^N$ is the end-effector force induced by the motor torques, $\mathbf{f}_e \in \mathbb{R}^N$ the end-effector force which is equal to the environmental reaction force, $M_d \in \mathbb{R}^{N\times N}$, $K_D \in \mathbb{R}^{N\times N}$ and $K_P \in \mathbb{R}^{N\times N}$ are unknown end-effector's inertia, damping and stiffness respectively and $\boldsymbol{\epsilon} \in \mathbb{R}^N$ is a parameterized nonlinear function of the system's position and velocity states with uncertain parameters. In real-world scenarios, characteristics of the interaction such as end-effector's geometry or environmental inhomogeneities may introduce state-dependent nonlinearities to the system and noticeably affect the interaction. To account for such phenomena, the nonlinear impedance term $\boldsymbol{\epsilon}(\cdot)$ is introduced in the system. The presence of $\boldsymbol{\epsilon}(\cdot)$ renders system control a challenging problem. Let us now assign a model to the force $\mathbf{f}_e$. The modeling of realistic contact requires the accurate knowledge of the end-effector and environment's properties which are, however, unknown in our scenario. Instead, we assume that the contact force can be approximated by the following essential model as $\mathbf{f}_e = K_e\mathbf{x} + D_e\dot{\mathbf{x}} + \boldsymbol{\epsilon}^*$ where $K_e \in \mathbb{R}^{N\times N}$ and $D_e \in \mathbb{R}^{N\times N}$ represent the environment's stiffness and damping respectively [29] and $\boldsymbol{\epsilon}^* \in \mathbb{R}^N$ represents the structural uncertainty of the model of $\mathbf{f}_e$. It is assumed that $K_e = diag(k_{e_j})$ and $D_e = diag(d_{e_j})$, $j = 1, ..., N$ where $diag(\cdot)$ symbolizes a diagonal matrix. It is also assumed that $M_d = diag(m_{d_j})$, $j = 1, ..., N$ [27].

First, the dynamics (1) are analyzed in a single direction of movement, and then, generalized to the N directions of the workspace. Dynamics (1) can be written in the

j-th direction as

$$m_{d_j}\ddot{x}_j + (k_{D_{jj}} + d_{e_j})\dot{x}_j + (k_{P_{jj}} + k_{e_j})x_j + \sum_l (k_{D_{jl}}\dot{x}_l + k_{P_{jl}}x_l) - \epsilon_j = f_j \tag{2}$$

where ϵ_j^* is incorporated into ϵ_j, x_j represents the end-effector position in the j-th direction, $k_{D_{jj}}$, $k_{P_{jj}}$ are unknown end-effector's damping and stiffness parameters respectively in the j-th direction and $k_{D_{jl}}, k_{P_{jl}}$ are coupling stiffness and damping parameters of the K_D and K_P between the j-th and all other $l \neq j$ directions. The ϵ_j depends upon the states not only of the j-th but also the $l \neq j$ directions. The total driving force f_j is the control command which is to be computed by adaptive control and we substitute it by the control input $m_{d_j}u_j$ for the reasons of our analysis. By expanding (2) to N directions and writing in state-space form, the following multiple-input system is obtained

$$\dot{\mathbf{x}}_t = A\mathbf{x}_t + B\mathbf{u} + B\boldsymbol{\epsilon}, \tag{3}$$

where $\mathbf{x}_t = [x_1\ \dot{x}_1\ \cdots\ x_N\ \dot{x}_N]^T$ is the state vector, $\mathbf{u} = [u_1\ \cdots\ u_N]^T$ the control input vector, $A \in \mathbb{R}^{kN\times kN}$ a matrix of unknown constant parameters, $B \in \mathbb{R}^{kN\times N}$ a known matrix, k the number of states per direction, here $k = 2$, and $\boldsymbol{\epsilon} = [\epsilon_1 \cdots \epsilon_N]^T$ a vector of uncertain functions ϵ_i. More specifically, the A and B are defined by $A = \left[A_{ji}\right]_{N\times N}$ where

$$A_{jj} = \begin{bmatrix} 0 & 1 \\ a_{jj,1} & a_{jj,2} \end{bmatrix}, \quad A_{ji} = \begin{bmatrix} 0 & 0 \\ a_{ji,1} & a_{ji,2} \end{bmatrix} \text{for } j \neq i\ , \tag{4}$$

$a_{jj,1} = -(k_{P_{jj}} + k_{e_j})/m_{d_j}$, $a_{jj,2} = -(k_{D_{jj}} + d_{e_j})/m_{d_j}$, $a_{ji,1} = -k_{P_{ji}}/m_{d_j}$, $a_{ji,2} = -k_{D_{ji}}/m_{d_j}$, $i, j = 1, ..., N$, $i \neq j$ and

$$B = diag(\mathbf{b}_1, \cdots, \mathbf{b}_N), \quad \mathbf{b}_i = \begin{bmatrix} 0 \\ 1 \end{bmatrix}\ , \quad \text{for } \forall i \in \{1, ..., \mathbb{N}\} \tag{5}$$

If the nonlinearities are negligible, then the nonlinear impedance term $\boldsymbol{\epsilon}$ can be omitted and the end-effector motion dynamics are expressed by a linear system. However, if the end-effector configuration is such that it cannot be approximated as a uniform, point-like mass or the environment contains non-negligible or unknown dissimilarities, then nonlinearities in the system dynamics are introduced and have to be taken into

account for proper model-based system control. In the remainder of this section, we formulate two different control problems which assume that $\boldsymbol{\epsilon} = \mathbf{0}$ and $\boldsymbol{\epsilon} \neq \mathbf{0}$, respectively. When $\boldsymbol{\epsilon} \neq \mathbf{0}$, then the problem of the representation of ϵ arises. Realistic contact phenomena are challenging to be described and also require accurate knowledge of the end-effector and environmental characteristics. Given that the system properties are unknown though, $\boldsymbol{\epsilon}$ is generically represented as a parameterized mixture of known basis functions with unknown parameters. This representation is inspired by the machine learning literature [30].

2.2. Problem definition

We assume that the ideal behavior of our system is represented by a reference model. The goal is to make the states of the system track the states of the reference model. Let us assume that the reference model, which expresses the desired response of system (3) to a reference signal $\mathbf{r}$, has the form

$$\dot{\mathbf{x}}_t^* = A^* \mathbf{x}_t^* + B^* \mathbf{r} \tag{6}$$

where $\mathbf{x}_t^* = [x_1^* \ \dot{x}_1^* \ \cdots \ x_N^* \ \dot{x}_N^*]^T \in \mathbb{R}^{kN}$ is the reference state vector, x_i^* the reference position, $A^* \in \mathbb{R}^{kN \times kN}$ and $B^* \in \mathbb{R}^{kN \times N}$ are known matrices and $\mathbf{r} = [r_1 \ \cdots \ r_N]^T, \in \mathbb{R}^N$ is a reference signal. The objective is to design the control law $\mathbf{u}$ in (3) such that the state vector $\mathbf{x}_t$ tracks the reference state vector $\mathbf{x}_t^*$ of (6) by minimizing the cost

$$J = \mathbf{e}^T Q \mathbf{e} \tag{7}$$

where Q is a symmetric positive-definite matrix, $\mathbf{e} = [e_1 \ \dot{e}_1 \cdots e_N \ \dot{e}_N]^T$, $e_i = x_i - x_i^*$ and $\dot{e}_i = \dot{x}_i - \dot{x}_i^*$ where $i = 1, ..., N$.

Let us now formulate the two state tracking problems.

Problem 1. Let us assume that $\boldsymbol{\epsilon} = \mathbf{0}$ in (3). Then, the corresponding system becomes linear and is given by

$$\dot{\mathbf{x}}_t = A\mathbf{x}_t + B\mathbf{u}. \tag{8}$$

where A is unknown and B is known. The goal is to design $\mathbf{u}$ in (8) such that the state

vector $\mathbf{x}_t$ tracks the reference state vector $\mathbf{x}_t^*$ of the reference model (6) by minimizing the cost J expressed by (7).

Problem 2. In this problem, the term $\boldsymbol{\epsilon}$ of (3) is assumed to be nonlinear with its components ϵ_i parametrized by a weighted mixture of nonlinear basis functions with unknown weights. The basis functions themselves are assumed to be known. The representation of the nonlinearity $\boldsymbol{\epsilon}$ as a weighted mixture of basis functions is inspired by the machine learning literature [30]. The goal is to design the control law $\mathbf{u}$ such that the state vector $\mathbf{x}_t$ of (3) tracks the reference state vector $\mathbf{x}_t^*$ of (6) by minimizing the cost J.

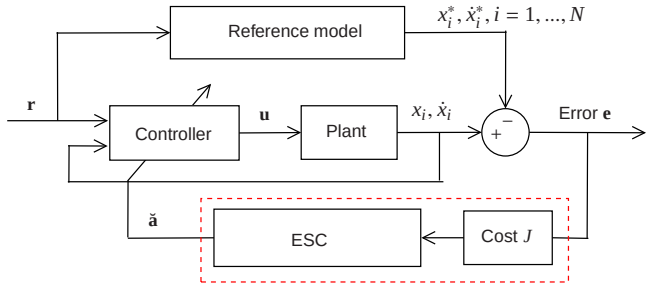
3. Multiple-Input ESC-MRAC and Multiple-Input ADI ESC-MRAC

In this section, we develop two control schemes, the Multiple-Input ESC-MRAC and Multiple-Input ADI ESC-MRAC, to solve the problems 1 and 2 introduced in Section 2. Let us first present briefly the principles of ESC and MRAC and the basic control law which is employed by ESC-MRAC and ADI ESC-MRAC and, then, analytically describe the proposed approaches.

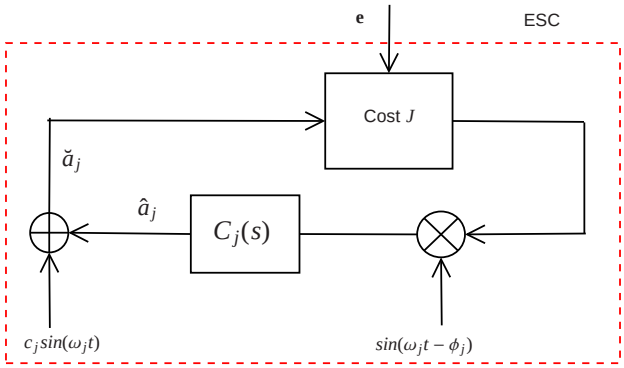
In the remainder of this article, for the sake of simplicity and without loss of generality, we assume $N = 2$. This consideration only serves in the simplicity of the analysis and does not imply any restriction for extending the approach to any number of inputs N.

3.1. MRAC and ESC principle

Fig. 2 illustrates the ESC-MRAC control scheme of a multiple-input system. MRAC expresses the system's desired response to a reference signal $\mathbf{r}$ in terms of a reference model, see Fig. 2(a). In state tracking, the objective consists of designing the control law $\mathbf{u}$ such that the error $\mathbf{e}$ converges to zero. The parameters of the control law are adapted by ESC while seeking the extremum of a cost criterion J [21, 22]. The most common ESC technique is the Perturbation-based ESC (PESC) because of its simple implementation and fast convergence compared to other ESC techniques. PESC is a gradient-based optimization technique where a sinusoidal perturbation $c_j sin(\omega_j t)$ is



(a) ESC-MRAC



(b) ESC

Figure 2: (a) General ESC-MRAC control scheme of a multiple-input system. The plant's state vector $[x_i\ \dot{x}_i \cdots x_N\ \dot{x}_N]$ is ensured to track the reference state vector $[x_i^*\ \dot{x}_i^* \cdots x_N^*\ \dot{x}_N^*]$ through appropriate design of the control law $\mathbf{u}$. The control parameter vector $\check{\mathbf{a}} = \{\check{a}_1, \cdots, \check{a}_M\}$ is adapted based on the minimization of the cost J through PESC. (b) PESC adaptation loop which applies on every single parameter a_j, $j = 1, ..., M$ where M is the total number of the adaptive parameters.

used to update the estimates $\hat{a}_j$ of the control parameters a_j at each time step, see Fig. 2(b). A more detailed explanation of ESC can be found in [21, 22].

3.2. Multiple-Input ESC-MRAC

Here, Multiple-Input ESC-MRAC is analyzed for reference state tracking of linear systems as described in Problem 1.

3.2.1. System dynamics

We consider the system

$$\begin{aligned} a_2\ddot{x}_1 + a_1\dot{x}_1 + a_0x_1 + a_3x_2 + a_4\dot{x}_2 = u_1, \\ a_7\ddot{x}_2 + a_6\dot{x}_2 + a_5x_2 + a_8x_1 + a_9\dot{x}_1 = u_2, \end{aligned} \tag{9}$$

which is of the form 3 where $\boldsymbol{\epsilon} = \mathbf{0}$. The $x_i, \dot{x}_i,\ i = 1, 2$ are measurable position and velocity states respectively and a_j are unknown parameters. The reference model of the system is defined by

$$\begin{aligned} a_2^*\ddot{x}_1^* + a_1^*\dot{x}_1^* + a_0^*x_1^* + a_3^*x_2^* + a_4^*\dot{x}_2^* = r_1, \\ a_7^*\ddot{x}_2^* + a_6^*\dot{x}_2^* + a_5^*x_2^* + a_8^*x_1^* + a_9^*\dot{x}_1^* = r_2. \end{aligned} \tag{10}$$

The goal consists of designing a control law $\mathbf{u} = [u_1\ u_2]^T$ such that the state tracking error $\mathbf{e} = [e_1\ \dot{e}_1\ e_2\ \dot{e}_2]^T$ converges to zero where $e_1 = x_1 - x_1^*$ and $e_2 = x_2 - x_2^*$. We define the control signals as

$$\begin{aligned} u_1 = \breve{a}_2z_1 + \breve{a}_1\dot{x}_1 + \breve{a}_0x_1 + \breve{a}_3x_2 + \breve{a}_4\dot{x}_2, \\ u_2 = \breve{a}_7z_2 + \breve{a}_6\dot{x}_2 + \breve{a}_5x_2 + \breve{a}_8x_1 + \breve{a}_9\dot{x}_1. \end{aligned} \tag{11}$$

The parameters $\breve{a}_i$ are continuously updated by PESC according to $\breve{a}_j = \hat{a}_j + c_j sin\omega_j t$, see Fig. 2, and the parameter error is defined by $\tilde{a}_j = a_j - \hat{a}_j$ which is the difference between the unknown a_j and the estimated $\hat{a}_j$ value of the parameter a_j which is shown in Fig. 2. The signals z_i in (11) are defined as

$$\begin{aligned} z_1 = \ddot{x}_1^* - \beta_0e_1 - \beta_1\dot{e}_1 - \beta_2e_2 - \beta_3\dot{e}_2, \\ z_2 = \ddot{x}_2^* - \beta_4e_2 - \beta_5\dot{e}_2 - \beta_6e_1 - \beta_7\dot{e}_1 \end{aligned} \tag{12}$$

where β_i are design parameters.

3.2.2. Dynamics of state tracking and parameter error

Fig. 2(b) illustrates the PESC adaptation law which is employed for adaptation of the $\breve{a}_0, ..., \breve{a}_9$ parameters. The compensator $C_j(s)$ in Fig. 2 is defined as [18]

$$C_j(s) = -g_j\left(\frac{1 + d_j s}{s}\right) \tag{13}$$

where $g_j,\ d_j \in \mathbb{R}$ are design parameters, $j \in \mathbb{R}^+$ and s is the complex Laplace variable. Based on Fig. 2(b), the dynamics of the parameter error are

$$\dot{\tilde{a}}_j = -\dot{\hat{a}}_j = g_j(1 + d_j s)\ [sin(\omega_j t - \phi_j)J(t)] \tag{14}$$

which can be written as

$$\dot{\tilde{a}}_j = g_j\left(sin(\omega_j t - \phi_j) + d_j\omega_j cos(\omega_j t - \phi_j)\right) J(t) + g_j d_j sin(\omega_j t - \phi_j)\dot{J}(t) \tag{15}$$

where $j = 0, ..., 9$. The cost function is defined by (7). By combining all $\dot{\tilde{a}}_j$ from (15), we write

$$\dot{\tilde{\mathbf{a}}} = G\mathbf{l}J + G\mathbf{m}\dot{J} \tag{16}$$

where

$$\tilde{\mathbf{a}} = [\tilde{a}_0 \cdots \tilde{a}_9]^T \tag{17}$$

$$\mathbf{l} = \begin{bmatrix} l_0 & \cdots & l_9 \end{bmatrix}^T, \quad \mathbf{m} = \begin{bmatrix} m_0 & \cdots & m_9 \end{bmatrix}^T$$

$$\begin{aligned} l_j &= sin(\omega_j t - \phi_j) + d_j\omega_j cos(\omega_j t - \phi_j), \\ m_j &= d_j\ sin(\omega_j t - \phi_j), \end{aligned} \tag{18}$$

and $G = diag([g_j]) \in \mathbb{R}^{10\times 10}$.

In addition, by combining (9) with (11)-(12), we infer the tracking error dynamics

$$\ddot{e}_1 + \beta_0 e_1 + \beta_1 \dot{e}_1 + \beta_2 e_2 + \beta_3 \dot{e}_2 = \frac{1}{a_2}((p_2 - \tilde{a}_2)z_1 + (p_0 - \tilde{a}_0)x_1 + (p_1 - \tilde{a}_1)\dot{x}_1 + (p_3 - \tilde{a}_3)x_2 + (p_4 - \tilde{a}_4)\dot{x}_2), \tag{19}$$

$$\ddot{e}_2+\beta_4 e_2+\beta_5\dot{e}_2+\beta_6 e_1+\beta_7\dot{e}_1 = \frac{1}{a_7}((p_7-\tilde{a}_7)z_2+(p_5-\tilde{a}_5)x_2+(p_6-\tilde{a}_6)\dot{x}_2+(p_8-\tilde{a}_8)x_1+(p_9-\tilde{a}_9)\dot{x}_1), \tag{20}$$

where $p_i = c_i sin\omega_i t$ and c_i are design parameters which represent the amplitude of the perturbation signals, see Fig. 2. The $e_i, \dot{e}_i, \ddot{e}_i$ represent the position, velocity and acceleration error respectively. By combining (19)-(20), we write the total tracking error dynamics as

$$\dot{\mathbf{e}} = E\mathbf{e} + B_e P^* \mathbf{v} \tag{21}$$

where $\mathbf{v} = \begin{bmatrix} x_1 & \dot{x}_1 & z_1 & x_2 & \dot{x}_2 & z_2 \end{bmatrix}^T$, $P^* = P - \tilde{A}$,

$$E = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\beta_0 & -\beta_1 & -\beta_2 & -\beta_3 \\ 0 & 0 & 0 & 1 \\ -\beta_6 & -\beta_7 & -\beta_4 & -\beta_5 \end{bmatrix}, \quad B_e = \begin{bmatrix} 0 & 0 \\ \frac{1}{a_2} & 0 \\ 0 & 0 \\ 0 & \frac{1}{a_7} \end{bmatrix}, \tag{22}$$

$$\tilde{A} = \begin{bmatrix} \tilde{a}_0 & \tilde{a}_1 & \tilde{a}_2 & \tilde{a}_3 & \tilde{a}_4 & 0 \\ \tilde{a}_8 & \tilde{a}_9 & 0 & \tilde{a}_5 & \tilde{a}_6 & \tilde{a}_7 \end{bmatrix}, \tag{23}$$

$$P = \begin{bmatrix} p_0 & p_1 & p_2 & p_3 & p_4 & 0 \\ p_8 & p_9 & 0 & p_5 & p_6 & p_7 \end{bmatrix} \tag{24}$$

We are now ready to state the following result.

Theorem 1. *Consider the system* (9) *with reference model* (10) *and control law* (11)-(12), *whose parameter error dynamics are governed by* (16)-(18) *and the cost function* J *is given by* (7). *Then, there exist parameters* d_i, g_i, ω_i, β_i, ϕ_i *and* c_i *of the system such that the error* $\mathbf{e}$ *globally and asymptotically converges to an* $O(\varepsilon)$ *neighbourhood of the origin where* $\varepsilon = 1/\omega$ *is a small positive parameter,* $\omega_i = n_i\omega$, $n_i \in \mathbb{N}$ *and* $\omega = min\{\omega_i\}$ *is the minimum frequency of the sinusoidal signals of the system.*

Proof: see Appendix 6.1.

We present some steps about how to set the system parameters, which derive from the

proof of the theorem, which is later presented in the Appendix 6.1. These steps are as follows:

1. First, the β_i parameters are appropriately set such that the E matrix is Hurwitz.

2. Afterwards, the P_1 matrix is computed by solving the equation $E^T P_1 + P_1 E = -H$ where H is a positive-definite matrix. The $P_1 \in \mathbb{R}^{4\times4}$ is a symmetric positive-definite matrix.

3. We set the cost matrix Q of (7) as $Q = P_1$. This is a condition which derives from the proof in the Appendix 6.1.

4. We set the g_i, ω_i, d_i, c_i and ϕ_i parameters such that the condition (25) holds, which is represented by

$$A_s \mathbf{w}_2^T = 2AVG(Pw^*)\mathbf{w}_1^T \tag{25}$$

where P is given by (24), $\mathbf{w}_2 \in \mathbb{R}^{1\times10}$ is a vector of real-valued design parameters, $w^* = \mathbf{w}_2 P_2 G \mathbf{m} \in \mathbb{R}$, $P_2 \in \mathbb{R}^{10\times10}$ is a diagonal matrix, $\mathbf{w}_1 = [1\ 1\ 1\ 1\ 1\ 1]$, A_s is a matrix of 0s and 1s and is given by (43) and AVG denotes the time-average. The c_i and ω_i parameters are involved in the P matrix. Because only one condition is available for the parameters, the way to set these parameters is not unique. One way to tune the g_i, ω_i, d_i, c_i and ϕ_i parameters is as follows. First, assign to the $\mathbf{w}_2$ some random real values. Afterwards, tune the parameters ϕ_i, g_i, ω_i and c_i to achieve certain desired system performance. Finally, compute the d_i from the condition (25). Some useful insights which can make the tuning process easier are the following. From the theorem, it holds that ω has to be large enough such that the neighbourhood of convergence ε is sufficiently narrow, i.e. the tracking error $\mathbf{e}$ is small. In addition, the c_i should be set large enough to cause noticeable perturbations to the estimated $\hat{a}_j$ parameters, see Fig. 2. Large values of g_i can increase the rate of change of the parameters while large values of the parameters of the P_2 matrix may decrease the parameter adaptation rate. By setting these parameters in our evaluation cases, we observe that our closed-loop system is

highly sensitive to the ω_i, β_i and g_i parameters and less sensitive to the rest of the parameters. This parameter sensitivity refers to how much the system response may change as a result of small changes in certain parameters. In general, for good convergence performance, we kept d_i low where $d_i = O(1)$.

In real-world robotic tasks, there may exist significant differences in the speed, the difficulty (applied forces) or other attributes of the task. This means that the system sensitivity to the parameters may change according to the task. Thus, in practice, it is important to infer from a few trials first the system sensitivity to the various parameters and pay relevant attention to tune these parameters while less attention can be paid to tune the remainder of the parameters which may move in a wider range of values. In our scenario, we notice that even relatively small changes in the ω_i and β_i values could cause considerable differences in the system performance. On the contrary, the system was less sensitive to changes in the c_i or d_i parameters.

Our current approach is designed for real-world robotic motion tasks where the end-effector may move in up to few directions only and the considered system dynamics remain low-order. However, in the case of very high-order systems, the tuning of the parameters of our system can become challenging. Specifically, in the case of a system with N control inputs, the total number of the design parameters ω_i, d_i, c_i, ϕ_i and g_i becomes equal to $5(2N + 1)N$ while only one condition on these parameters is available for their computation. In very high-order systems other extremum seeking control techniques, such as a dither-free gradient-based extremum seeking control approach, can be employed [31]. In the case of a dither-free gradient-based technique, the dither signal and signal modulation functions, which are currently involved in the ESC loop of Fig. 2 (b), are omitted and, instead, a parameter $\breve{a}_i$ for instance is updated according to $\dot{\breve{a}}_i = -\mu \frac{\partial J}{\partial \breve{a}_i}$ where μ is a design parameter. In this way, the adaptation of each parameter $\breve{a}_i$ just involves a single instead of five design parameters. In this scheme, a gradient-estimator has to be designed, which computes the $\frac{\partial J}{\partial \breve{a}_i}$ [31]. Although such a scheme significantly reduces the number of the design parameters, it would have to be compared with other dither-based extremum seeking control schemes to examine its efficiency and convergence performance.

Motivated by our two-dimensional real-world experimental scenario, we analyze here the Multiple-Input ESC-MRAC in two directions of movement only, i.e. $N = 2$, to also present in a more clear way the details of the derivation of the state tracking and parameter error dynamics. However, the proposed approach can be extended in theory in a straightforward manner to any $N > 2$. By expanding the system dynamics to $N > 2$, the system (9) would take the canonical form of the system (3) where $\boldsymbol{\epsilon} = \mathbf{0}$. In addition, the reference model (10) would take the canonical form of system (6). Therefore, the control inputs would expand from 2 to N and each control input would now depend on all the system states $\{x_1, \dot{x}_1, ..., x_N, \dot{x}_N\}$, see Eq. (11). In turn, the z_i signals would expand from 2 to N and each z_i signal would now depend on all the state tracking errors $\{e_1, \dot{e}_1, ..., e_N, \dot{e}_N\}$, see Eq. (12). This would also induce an increase in the adaptable parameters $\breve{a}_i$ and also the β_i parameters, see Eq. (12).

The result of the Theorem 1 can also be extended to systems with nonlinear uncertainties given that these uncertainties belong to a particular class of periodic functions as it is described in the following corollary.

Corollary 1. *Consider the system*

$$\begin{aligned} a_2\ddot{x}_1 + a_1\dot{x}_1 + a_0x_1 + a_3x_2 + a_4\dot{x}_2 - \epsilon_1 &= u_1, \\ a_7\ddot{x}_2 + a_6\dot{x}_2 + a_5x_2 + a_8x_1 + a_9\dot{x}_1 - \epsilon_2 &= u_2. \end{aligned} \tag{26}$$

The reference model of the system is given by (10)*, the control law by* (11)-(12)*, the parameter error dynamics by* (16)-(18) *and the cost J by* (7)*. Let* ϵ_1 *and* ϵ_2 *be* T_1*- and* T_2*-periodic functions with zero average values within* T_1 *and* T_2 *respectively with* $T_1 = \frac{2\pi}{\omega_1^*}$, $T_2 = \frac{2\pi}{\omega_2^*}$, $\omega_1^* = n_1^*\omega$, $\omega_2^* = n_2^*\omega$, $n_1^*, n_2^* \in \mathbb{N}$ *where* $\omega = min\{\omega_1^*,\ \omega_2^*\}$*. Then, there exist parameters* $d_i,\ g_i,\ \omega_i,\ \beta_i,\ \phi_i$ *and* c_i *such that the error* $\mathbf{e}$ *globally and asymptotically converges to an* $O(\varepsilon)$ *neighbourhood of the origin where* $\varepsilon = 1/\omega$ *is a small positive parameter and* $\omega_i = n_i\omega$, $n_i \in \mathbb{N}$.

Proof: see Appendix 6.2.

The assumption of zero-mean periodicity of ϵ_1 and ϵ_2 assumes certain knowledge about the system nonlinearities. For instance, if ϵ_1 and ϵ_2 are known to be sinusoidal func-

tions, then the assumption of zero mean-periodicity holds and the corollary above applies. If ϵ_1 and ϵ_2 are not zero-mean and periodic but linearly parametrizable via a set of basis functions, then the Multiple-Input ADI ESC-MRAC applies, which is presented in Section 3.3. If ϵ_1 and ϵ_2 are totally unknown, then robust control would be an alternative to the current adaptive control schemes [14].

3.3. Multiple-Input ADI ESC-MRAC

To solve problem 2 defined in Section 2.2, we propose a multiple-input ESC-MRAC method based on adaptive dynamic inversion. We assume that the components ϵ_i of $\boldsymbol{\epsilon}$ in (3) are represented by weighted mixtures of known basis functions of the system states with unknown weights. As it is earlier explained, given that the exact end-effector and environmental properties are unknown in our scenario, the nonlinearity $\boldsymbol{\epsilon}$ of (3) cannot be described in exact physical terms and is, thus, generically represented by mixtures of basis functions. According to the number and shape of the basis functions, a mixture can represent a function of an abstract-complexity. Based on this, system (3) is written in two directions as

$$\begin{aligned} a_2\ddot{x}_1 + a_1\dot{x}_1 + a_0x_1 + a_3x_2 + a_4\dot{x}_2 + \mathbf{a}_\theta\boldsymbol{\theta}(x_1, \dot{x}_1, x_2, \dot{x}_2) = u_1, \\ a_7\ddot{x}_2 + a_6\dot{x}_2 + a_5x_2 + a_8x_1 + a_9\dot{x}_1 + \mathbf{a}_\psi\boldsymbol{\psi}(x_1, \dot{x}_1, x_2, \dot{x}_2) = u_2, \end{aligned} \tag{27}$$

where

$$\mathbf{a}_\theta = [a_{10} \cdots a_{10+\Theta-1}], \quad \mathbf{a}_\psi = [a_{10+\Theta} \cdots a_{M-1}]$$

and $a_i,\ i = 0, ..., M-1$ are unknown parameters. The $M = 10 + \Theta + \Psi$ represents the total number of unknown parameters, $\Theta, \Psi \in \mathbb{N}$ and $\boldsymbol{\theta} \in \mathbb{R}^\Theta$ and $\boldsymbol{\psi} \in \mathbb{R}^\Psi$ are vectors of known and bounded nonlinear functions of the system's position and velocity states. Let us assume that the reference model of system (27) is given by (10).

The goal consists of designing $\mathbf{u} = [u_1\ u_2]^T$ such that the error $\mathbf{e} = [x_1\ \dot{x}_1\ x_2\ \dot{x}_2] - [x_1^*\ \dot{x}_1^*\ x_2^*\ \dot{x}_2^*]$ converges to zero. We design the control signals as

$$\begin{aligned} u_1 = \breve{a}_2z_1 + \breve{a}_1\dot{x}_1 + \breve{a}_0x_1 + \breve{a}_3x_2 + \breve{a}_4\dot{x}_2 + \breve{\mathbf{a}}_\theta\boldsymbol{\theta}, \\ u_2 = \breve{a}_7z_2 + \breve{a}_6\dot{x}_2 + \breve{a}_5x_2 + \breve{a}_8x_1 + \breve{a}_9\dot{x}_1 + \breve{\mathbf{a}}_\psi\boldsymbol{\psi} \end{aligned} \tag{28}$$

where $\boldsymbol{\theta}$ and $\boldsymbol{\psi}$ are explicitly taken into account. The parameters $\breve{a}_i$, $i = 0, \cdots, M-1$ are updated according to PESC as described in Section 3.2 while the signals z_1, z_2 are defined by (12). By combining (27)-(28), we derive the tracking error dynamics

$$\dot{\mathbf{e}} = E\mathbf{e} + B_e P_n^* \mathbf{v}_n \tag{29}$$

where E and B_e are defined by (22), $P_n^* = P_n - \tilde{A}_n$,
$\mathbf{v}_n = \begin{bmatrix} x_1 & \dot{x}_1 & z_1 & \boldsymbol{\theta}^T & x_2 & \dot{x}_2 & z_2 & \boldsymbol{\psi}^T \end{bmatrix}^T$,

$$\begin{aligned} \tilde{A}_n &= \begin{bmatrix} \tilde{a}_0 & \tilde{a}_1 & \tilde{a}_2 & \tilde{\mathbf{a}}_\theta & \tilde{a}_3 & \tilde{a}_4 & 0 & \mathbf{0}_\Psi \\ \tilde{a}_8 & \tilde{a}_9 & 0 & \mathbf{0}_\Theta & \tilde{a}_5 & \tilde{a}_6 & \tilde{a}_7 & \tilde{\mathbf{a}}_\psi \end{bmatrix}, \\ P_n &= \begin{bmatrix} p_0 & p_1 & p_2 & \mathbf{p}_\theta & p_3 & p_4 & 0 & \mathbf{0}_\Psi \\ p_8 & p_9 & 0 & \mathbf{0}_\Theta & p_5 & p_6 & p_7 & \mathbf{p}_\psi \end{bmatrix} \end{aligned} \tag{30}$$

where $\mathbf{p}_\theta = [p_j]_\Theta \in \mathbb{R}^{1\times\Theta}$, $\mathbf{p}_\psi = [p_j]_\Psi \in \mathbb{R}^{1\times\Psi}$ and $\mathbf{0}_n \in \mathbb{R}^{1\times n}$ denotes a row vector of all zeros.

We also define the parameter error vector

$$\tilde{\mathbf{a}}_n = [\tilde{a}_0\ \tilde{a}_1\ \tilde{a}_2\ \tilde{\mathbf{a}}_\theta\ \tilde{a}_3\ \tilde{a}_4\ \tilde{a}_5\ \tilde{a}_6\ \tilde{a}_7\ \tilde{a}_8\ \tilde{a}_9\ \tilde{\mathbf{a}}_\psi]^T. \tag{31}$$

By combining the parameter adaptation law (15) for all the parameters, the total parameter error dynamics are expressed by

$$\dot{\tilde{\mathbf{a}}}_n = G_n \mathbf{l}_n J + G_n \mathbf{m}_n \dot{J} \tag{32}$$

where

$$\begin{aligned} \mathbf{l}_n &= \begin{bmatrix} l_0\ l_1\ l_2\ \mathbf{l}_\theta & l_3 \cdots l_9 & \mathbf{l}_\psi \end{bmatrix}^T, \\ \mathbf{m}_n &= \begin{bmatrix} m_0\ m_1\ m_2\ \mathbf{m}_\theta & m_3 \cdots m_9 & \mathbf{m}_\psi \end{bmatrix}^T, \\ \boldsymbol{l}_\theta &= [l_{10} \cdots l_{10+\Theta-1}],\ \boldsymbol{l}_\psi = [l_{10+\Theta} \cdots l_{M-1}], \\ \boldsymbol{m}_\theta &= [m_{10} \cdots m_{10+\Theta-1}],\ \boldsymbol{m}_\psi = [m_{10+\Theta} \cdots m_{M-1}], \end{aligned} \tag{33}$$

the l_i, m_i are defined by (18), the cost J is defined by (7) and $G_n = diag([g_j]) \in \mathbb{R}^{M\times M}$.

We are now ready to state the following result.

Theorem 2. *Consider the system* (27) *with reference model* (10)*, control law* (12) *and* (28)*, the parameter error dynamics given by* (32)-(33) *and the cost function J by* (7)*. Then, there exist parameters* $d_i,\ g_i,\ \omega_i,\ \beta_i,\ \phi_i$ *and* c_i *such that the error* $\mathbf{e}$ *globally and asymptotically converges to an* $O(\varepsilon)$ *neighbourhood of the origin, where* $\varepsilon = 1/\omega$ *is a small positive parameter,* $\omega_i = n_i\omega$, $n_i \in \mathbb{N}$ *and* $\omega = min\{\omega_i\}$ *is the minimum frequency of the sinusoidal signals of the system.*

Proof: see Appendix 6.3.

To tune the system parameters, steps similar to the steps presented in Section 3.2 are followed.

4. Evaluation

First, Multiple-Input ESC-MRAC and Multiple-Input ADI ESC-MRAC are evaluated and compared in simulations. This evaluation aims at showing the performance of ADI ESC-MRAC and ESC-MRAC in presence of parametrically-uncertain nonlinearities. Afterwards, ADI ESC-MRAC is compared with fixed-gain Proportional-Integral-Derivative (PID) control in simulation in order to show how ESC-MRAC may outperform common fixed-gain control in presence of parametric uncertainty. Apart from simulations, Multiple-Input ADI ESC-MRAC and Multiple-Input ESC-MRAC are evaluated and compared with SISO ESC-MRAC [18] in a real-world robot engraving task to show that the currently proposed approaches apply successfully in real-world tasks.

Our evaluations are motivated by our experimental scenario where the end-effector of a 2 Degrees-of-Freedom (DoF) linear-actuated haptic device (*ThrustTube*) is required to move in two directions, so called normal and parallel directions, and manipulate an object of the environment such as a plasticine material, see Fig. 3. A sculpting tool is firmly attached on the end-effector for engraving the object and the system end-effector - tool behaves as a rigid body. The device's encoders measure the end-effector position and velocity during the task.

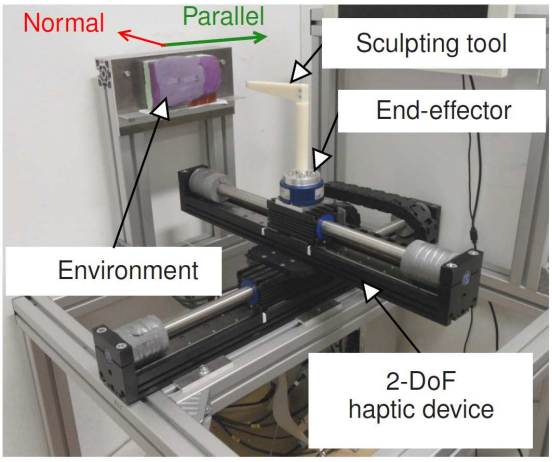


Figure 3: Experimental setup. The two directions of movement, normal and parallel, are visualized by a red- and a green-color axis respectively.

4.1. Simulations

4.1.1. ADI ESC-MRAC over ESC-MRAC in presence of system nonlinearities

Multiple-Input ADI ESC-MRAC is evaluated and compared with Multiple-Input ESC-MRAC in two test cases where the system nonlinearities differ in each case. Let us consider the system

$$\begin{aligned} 4\ddot{x}_1 + 10\dot{x}_1 + 15x_1 + x_2 + 9\dot{x}_2 + \theta_1 + \theta_2 &= u_1, \\ 5\ddot{x}_2 + 12\dot{x}_2 + 10x_2 + x_1 + 8\dot{x}_1 + \psi_1 + \psi_2 &= u_2 \end{aligned} \tag{34}$$

which represents the motion dynamics of an end-effector interacting with the environment in two directions of movement, one normal to the environmental plane and one parallel to this plane. The $\theta_1, \theta_2, \psi_1$ and ψ_2 capture the nonlinear behavior of the system end-effector - environment during their interaction. The reference model of the system is expressed by

$$\begin{aligned} \ddot{x}_1^* + 16\dot{x}_1^* + 16x_1^* + 2x_2^* + 4\dot{x}_2^* &= 16r_1, \\ \ddot{x}_2^* + 16\dot{x}_2^* + 16x_2^* + 2x_1^* + 4\dot{x}_1^* &= 16r_2. \end{aligned} \tag{35}$$

Let us consider the following test cases.

Test case 1. The system nonlinearities are expressed by

$$\begin{aligned} \theta_1 &= 150e^{-10(x_1+0.5)^2}, \quad \theta_2 = 150e^{-10(x_1-0.5)^2} \\ \psi_1 &= 500e^{-100(x_2+0.4)^2}, \quad \psi_2 = 500e^{-100(x_2-0.4)^2}. \end{aligned} \tag{36}$$

ADI ESC-MRAC takes θ_1, θ_2, ψ_1 and ψ_2 into account in contrast to ESC-MRAC which does not take them into account. Fig. 4 shows practical asymptotic tracking of the reference position x_1^*, x_2^* and velocity $\dot{x}_1^*$, $\dot{x}_2^*$ states by the position x_1, x_2 and velocity $\dot{x}_1$, $\dot{x}_2$ states of the system by ADI ESC-MRAC and ESC-MRAC. We observe that ADI ESC-MRAC achieves successful practical state tracking in contrast to ESC-MRAC which does not achieve to compensate for the uncertainties θ_1, θ_2, ψ_1 and ψ_2 of the system. In theory, the nonlinear functions of the system can be any state (position/velocity)-dependent functions. For the current evaluation, position-dependent functions are considered. Although motion is only few mm in the normal and few cm in the parallel direction, it still yields considerable forces of the range of 0-2 N on the end-effector. To set the system parameters we follow the steps described in Section 3.2. The parameters are set for the simulation as follows. It is $\beta_0 = \beta_4 = 9$, $\beta_1 = \beta_5 = 3$, $\beta_2 = \beta_6 = 2$, $\beta_3 = \beta_7 = 1$. In addition, $g_0 = 120$, $g_1 = 80$, $g_2 = 100$, $g_3 = g_4 = 120$, $g_5 = g_6 = 20$, $g_7 = g_8 = g_9 = 5$. The $g_{10} = g_{11} = g_{12} = g_{13} = 150$ and all the d_i parameters are set equal to 0.1. The frequency values are $\omega_0 = 100$, $\omega_1 = 118$, $\omega_2 = 16$, $\omega_3 = 120$, $\omega_4 = 150$, $\omega_5 = 148$, $\omega_6 = 160$, $\omega_7 = 60$, $\omega_8 = 9$, $\omega_9 = 1$, $\omega_{10} = 200$, $\omega_{11} = 55$, $\omega_{12} = 100$ and $\omega_{13} = 130$.

Test case 2. The system nonlinearities are expressed by

$$\theta_1 = 20x_1^2, \quad \theta_2 = 10x_1^4, \quad \psi_1 = 30x_2^2, \quad \psi_2 = 0.$$

Fig. 5 shows tracking of the reference position x_1^*, x_2^* and velocity $\dot{x}_1^*, \dot{x}_2^*$ states by the position x_1, x_2 and velocity $\dot{x}_1, \dot{x}_2$ states of the system respectively by Multiple-Input ADI ESC-MRAC and Multiple-Input ESC-MRAC. The parameters of the system are set for the simulation as follows. It is $\beta_0 = \beta_4 = 9$, $\beta_1 = \beta_5 = 3$, $\beta_2 = \beta_6 = 2$, $\beta_3 = \beta_7 = 1$. In addition, $g_0 = g_1 = g_2 = g_5 = g_6 = g_7 = 50$, $g_3 = g_4 = 10$, $g_8 = g_9 = 5$ and $g_{10} = g_{11} = g_{12} = 13 = 100$. The d_i parameters are set $d_0 = \cdots = d_9 = 0.1$ and

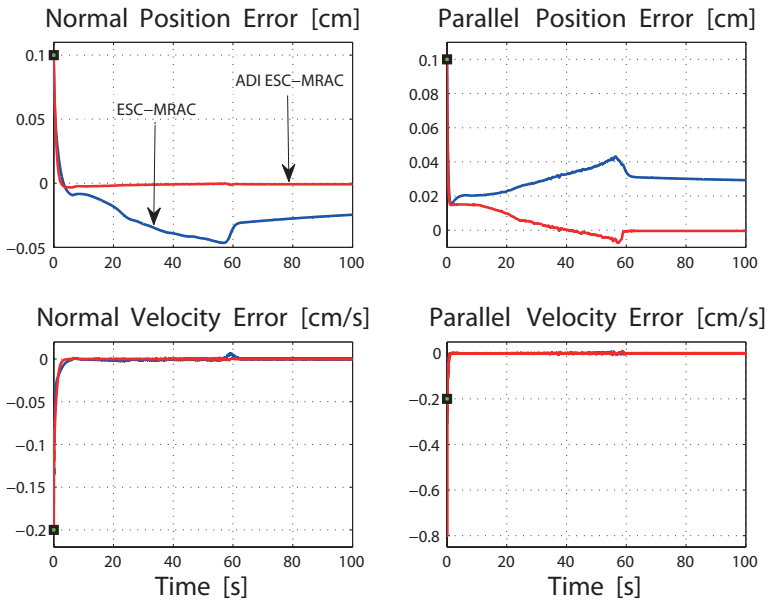


Figure 4: Test case 1: comparison of Multiple-Input ADI ESC-MRAC with Multiple-Input ESC-MRAC. State tracking error of (red) ADI ESC-MRAC and (blue) ESC-MRAC. Green markers are used to highlight the initial error value of ADI ESC-MRAC. All signals are expressed with respect to time in sec.

$d_{10} = \cdots = d_{13} = 0.2$. The frequency values are $\omega_0 = 150$, $\omega_1 = 118$, $\omega_2 = 26$, $\omega_3 = 50$, $\omega_4 = 1$, $\omega_5 = 125$, $\omega_6 = 137$, $\omega_7 = 60$, $\omega_8 = 25$, $\omega_9 = 20$, $\omega_{10} = 10$, $\omega_{11} = 80$, $\omega_{12} = 75$ and $\omega_{13} = 25$. In this test case, ESC-MRAC is observed to compensate in practice for the nonparametric uncertainties θ_1, θ_2 and ψ_1 and exhibits almost the same performance with that of Multiple-Input ADI ESC-MRAC while both approaches achieve successful practical asymptotic state tracking.

By comparing the test cases 1 and 2, we observe that ESC-MRAC can still compensate up to certain degree for the unknown nonlinearities of the system, although its control design is not made to take these uncertainties into account. This demonstrates certain robustness of ESC-MRAC to structural uncertainties, which is due to the model-free ESC adaptation law.

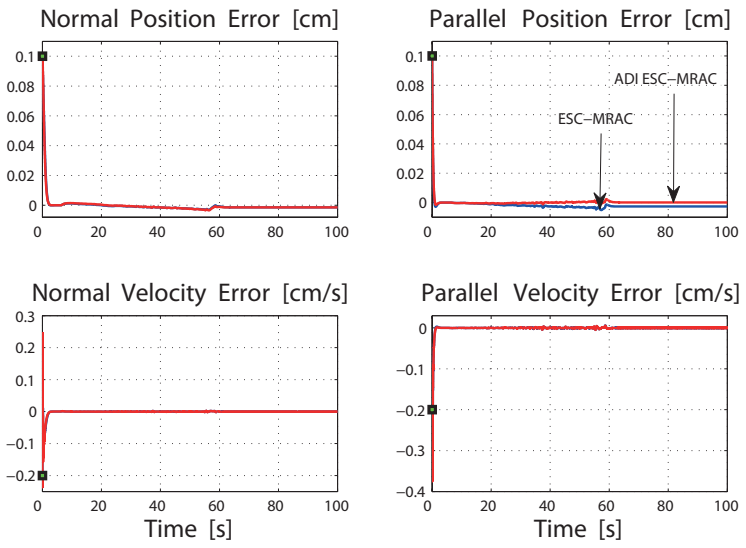


Figure 5: Test case 2: comparison of Multiple-Input ADI ESC-MRAC with Multiple-Input ESC-MRAC. State tracking error of (red) Multiple-Input ADI ESC-MRAC and (blue) Multiple-Input ESC-MRAC. Green markers are used to highlight the initial error value. All signals are expressed with respect to time in sec.

4.1.2. ADI ESC-MRAC over fixed-gain PID control

Multiple-Input ADI ESC-MRAC is currently compared with PID control for the system of the test case 1. In robotic tasks, it is common to employ fixed-gain control to achieve a task by properly setting the controller parameters. However, fixed-, high-gain control cannot always lead to success especially in presence of task uncertainty. The reason for this is because, when the system is uncertain and its properties vary during the task, the parameter values - initially assigned to the PID controller - may be inappropriate and lead to non-desired overshooting, excessive build-up of forces or no accomplishment of the task goal. On the other hand, adaptive control is dedicated to dealing with system uncertainty and achieving system convergence to a desired operating point or reference state. Currently, we show how Multiple-Input ADI ESC-MRAC outperforms fixed-gain PID control in presence of uncertainty. The PID control law is defined as

$$\begin{aligned} u_1 &= k_{P_1} e_1 + k_{D_1} \dot{e}_1 + k_{I_1} \int e_1 + k_{P_{12}} e_2 + k_{D_{12}} \dot{e}_2, \\ u_2 &= k_{P_2} e_2 + k_{D_2} \dot{e}_2 + k_{I_1} \int e_2 + k_{P_{21}} e_1 + k_{D_{21}} \dot{e}_1 \end{aligned} \tag{37}$$

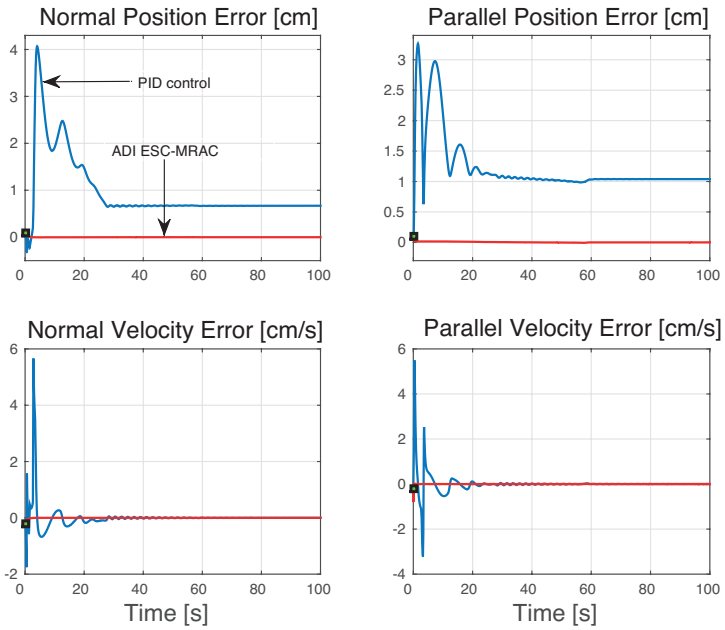


Figure 6: Comparison of Multiple-Input ADI ESC-MRAC with PID control for the test case 1: state tracking error of (red) ADI ESC-MRAC and (blue) PID control. Green markers are used to highlight the initial error value of ADI ESC-MRAC. All signals are expressed with respect to time in sec.

where $k_{P_1} = 12.8, k_{D_1} = 3, k_{I_1} = 0, k_{P_{12}} = 2, k_{D_{12}} = 2,, k_{P_2} = 10, k_{D_2} = 3, k_{I_2} = 0.003, k_{P_{21}} = 0, k_{D_{21}} = 0$. Note that coupling proportional and derivative error terms have been introduced in the design of the PID control law (37). By adding these coupling terms, the transient performance of the system is improved and the system is observed to be less oscillatory during this phase. The results of this comparison are shown in Fig. 6. We observe that the PID control achieves successful velocity tracking while its initial and transition error are rather higher than those of ADI ESC-MRAC. However, PID control does not achieve successful position tracking and exhibits a non-zero steady-state error. This shows that fixed-gain PID control is not efficient to deal with uncertainty and ensure tracking of the desired states. The parameters of the PID controller are set such that the controller exhibits the best performance.

4.2. Experimental evaluation

Here, a real-world robot engraving task is executed onto a plasticine object by using the setup of Fig. 3. The device's end-effector can move in two directions, one normal and one parallel to the object's initially planar surface. The sculpting tool of the end-effector serves in engraving the object. Prior to the task, the end-effector is placed such that the tool tip just touches the object's surface and is commanded to follow a reference motion in the two directions while the tool engraves a pattern into the plasticine material.

As it is shown in Fig. 3, the end-effector (with the sculpting tool) has a rather complex geometry which makes the interaction dynamics of the tool with its environment nonlinear. Such nonlinearities come from the complex geometry and variability in the physical properties of the system end-effector-tool which also make the physics of the interaction non-trivial to model. In our present work, the goal is not to accurately derive the dynamics of the system based on the physics of the task but instead to approximate these complex dynamics by a generic and uncertain system and make this system achieve the task goal by adjusting real-time its controller parameters. Based on this, as it is shown later, the nonlinearities of the system in our task are approximated by parameterized functions of a generic form, whose parameters are adapted in real-time by ESC to allow for achievement of the task goal, i.e. tracking of some desired states.

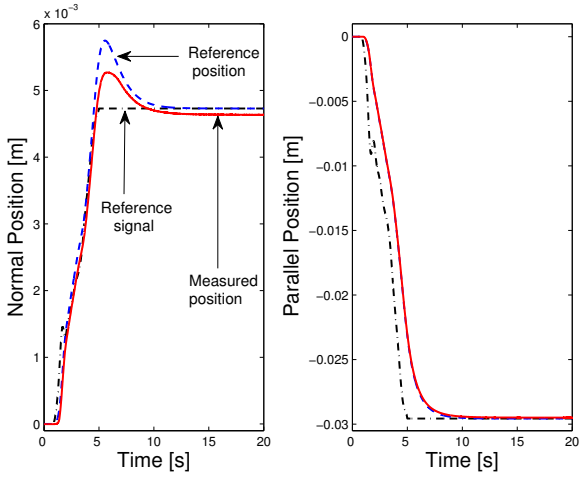
The control law employed during the task consists of the superposition of an adaptive controller and a fixed-gain position controller. The reason why the adaptive control law is superimposed with a positional control law is two-fold. First, the haptic device involves some friction and the position control compensates for this friction while the adaptive controller compensates for the system uncertainty due to the unknown end-effector and environmental properties. Second, the position control law can regulate the control output during the first couple of seconds of the task where the adaptive controller may exhibit overshoot. During this transient phase, it is important that the system exhibits an acceptable behavior in practice and does not exaggerate. Multiple-Input ESC-MRAC, Multiple-Input ADI ESC-MRAC and SISO ESC-MRAC [18] are separately employed for this engraving task and their performance is compared.

4.2.1. Multiple-Input ESC-MRAC

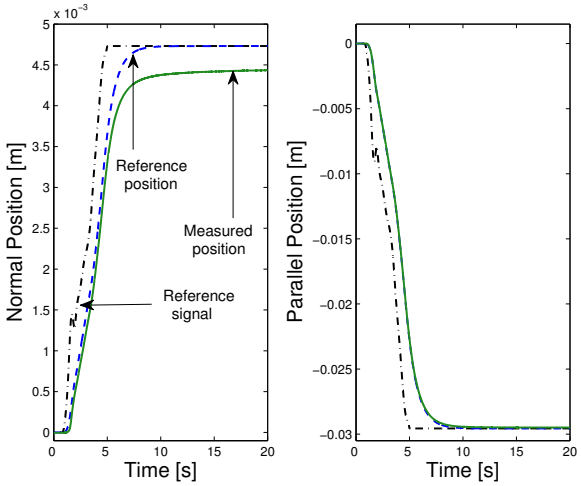
Fig. 7 shows tracking of the desired movement by applying Multiple-Input ESC-MRAC and SISO ESC-MRAC by using the setup described previously in section 4.2. In the case of SISO ESC-MRAC, movement in each one of the two directions is treated as a SISO system. SISO ESC-MRAC is observed to exhibit significantly lower performance than Multiple-Input ESC-MRAC in the normal direction of movement while tracking of the parallel motion is rather accurate by both methods. Multiple-Input ESC-MRAC exhibits some low steady-state tracking error in the normal direction, which is due to system nonlinearities, which are not taken into account in the control law, see Fig. 7(a). In Multiple-Input ESC-MRAC, the c_i signals are set $c_0 = 3$, $c_1 = c_2 = c_7 = 2$, $c_3 = c_8 = 30$, $c_4 = c_9 = 30$, $c_5 = 21$, $c_6 = 5$ and the frequencies are set $\omega_0 = 100$, $\omega_1 = 8$, $\omega_2 = 14$, $\omega_3 = 10$, $\omega_4 = 5$, $\omega_5 = 230$, $\omega_6 = 150$, $\omega_7 = 120$, $\omega_8 = 40$, $\omega_9 = 50$. In case of SISO ESC-MRAC, the system's parameters are tuned according to [18] for best performance.

4.2.2. Multiple-Input ADI ESC-MRAC

Multiple-Input ADI ESC-MRAC is employed to execute the same desired movement with the goal of compensating for the inaccuracy in the normal position tracking observed in Multiple-Input ESC-MRAC. By observing Fig. 7(a), we notice that parallel position tracking by Multiple-Input ESC-MRAC is accurate and, thus, we set $\boldsymbol{\theta} = \mathbf{0}$ in (28) since no compensation is needed. To decrease the steady-state position tracking error in the normal direction, we define a set of nonlinear functions $\boldsymbol{\psi} = [\psi_i]$, $i = 1, ..., 4$ in (28). However, as it is previously explained, the true system dynamics are not known and thus defining the nonlinearities of the system based on the physics of the task is not feasible. For this reason, the nonlinear functions of the system, and specifically the ψ_i, are defined based on the goal of the task consisting of decreasing the normal position tracking error. More specifically, they are defined as Gaussian-shaped functions centered at some desired normal position points and their variances are manually set



(a) Multiple-Input ESC-MRAC



(b) SISO ESC-MRAC

Figure 7: Position tracking by (a) Multiple-Input ESC-MRAC: (dash-dot black line) reference signal, (dashed blue line) reference position, (solid red line) measured position, (b) SISO ESC-MRAC: (dash-dot black line) reference signal, (dashed blue line) reference position, (solid green line) measured position.

by the user as

$$\psi_1 = e^{-0.048(x_2-0.0048)^2}, \quad \psi_2 = e^{-0.01(x_2-0.0049)^2},$$

$$\psi_3 = e^{-2(x_2-0.005)^2}, \quad \psi_4 = e^{-0.00725(x_2-0.005788)^2}.$$

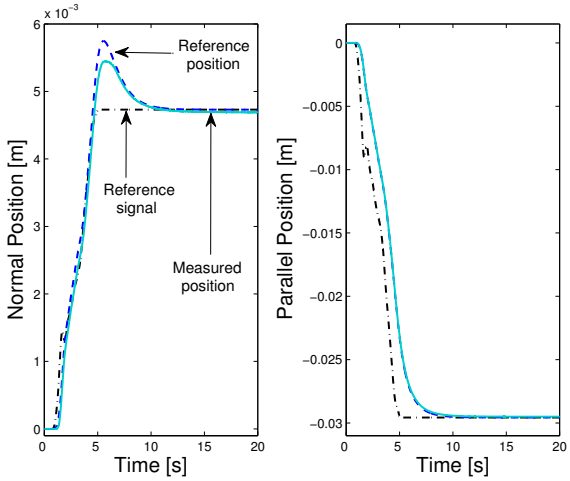


Figure 8: Position tracking by Multiple-Input ADI ESC-MRAC: (dash-dot black line) reference signal, (dashed blue line) reference position, (solid turquoise line) measured position.

such that the normal steady-state tracking error approximates zero. Fig. 8 shows execution of the desired motion by Multiple-Input ADI ESC-MRAC while Fig. 9 shows the tracking errors of Multiple-Input ESC-MRAC, Multiple-Input ADI ESC-MRAC and SISO ESC-MRAC all together. The tracking error is defined as the difference between the reference and measured position. The position signals are of order $10^{-3}m$ in the normal direction and $10^{-2}m$ in the parallel direction while the tracking errors are of order $10^{-4}m$. The reference model of the system is defined by (10) and its parameters a_i^* are set in all cases such that the reference position tracks the reference signal, see Fig. 7 and Fig. 8. We observe that, in the normal direction, Multiple-Input ADI ESC-MRAC noticeably improves the steady-state tracking performance compared to Multiple-Input ESC-MRAC. In the parallel direction, all approaches exhibit negligible error and allow for rather accurate motion tracking.

In Multiple-Input ADI ESC-MRAC, the parameters are set as $\omega_0 = 100$, $\omega_1 = 118$, $\omega_2 = 16$, $\omega_3 = 70$, $\omega_4 = 85$, $\omega_5 = 148$, $\omega_6 = 145$, $\omega_7 = 60$, $\omega_8 = 40$, $\omega_9 = 30$, $\omega_{10} = 40$, $\omega_{11} = 35$, $\omega_{12} = 30$ and $\omega_{13} = 17$. The $\omega_{10}, \cdots, \omega_{13}$ frequencies correspond to update of the a_i parameters of the $\mathbf{a}_\psi$ vector. The c_i signals are set $c_0 = 3$, $c_1 = c_2 = 2$, $c_3 = \cdots = c_9 = 30$, $c_{10} = \cdots = c_{13} = 30$. The d_i signals are set $d_0 = d_1 = d_2 = d_5 =$

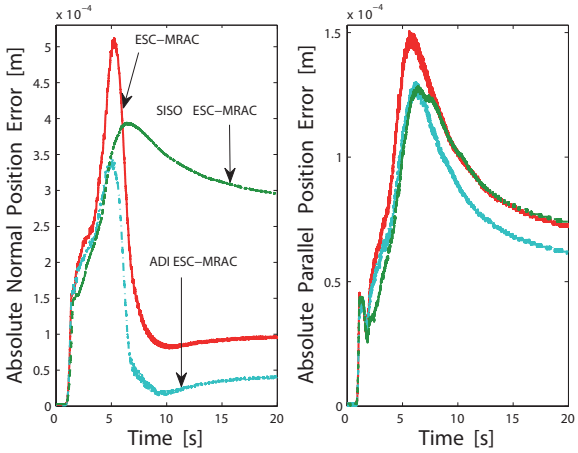


Figure 9: Error between reference and measured position of: (solid red line) Multiple-Input ESC-MRAC, (dashed green line) SISO ESC-MRAC and (dash-dot turquoise line) Multiple-Input ADI ESC-MRAC, in the two directions of movement.

$d_6 = d_7 = 0.1$, $d_3 = d_4 = d_8 = d_9 = 0.01$ and $d_{10} = \cdots = d_{13} = 10$. The g_i signals are set $g_0 = 1200$, $g_1 = 800$, $g_2 = g_5 = g_6 = g_7 = 1000$, $g_3 = g_4 = 1$, $g_8 = g_9 = 1$ and $g_{10} = g_{11} = g_{12} = g_{13} = 0.01$.

5. Conclusion and future work

In this article, we propose an adaptive control framework for execution of robot end-effector motions in presence of parametric system uncertainty. To this aim, two adaptive control schemes, namely Multiple-Input ESC-MRAC and Multiple-Input ADI ESC-MRAC, are introduced for reference state tracking of multiple-input systems with parametric uncertainty. It is shown that the proposed schemes guarantee asymptotic convergence of the system states to an $O(\varepsilon)$ neighbourhood of the system's reference states. The performance of the proposed schemes is evaluated in simulations and in a real-world robot engraving task.

The present work sets up a number of interesting questions and gives rise to certain important future steps. Given that real-world systems often involve apart from parametric also structural uncertainty, it is of interest to compare our current adaptive with robust control schemes as well. Results of such a comparison will be presented in fu-

ture work. Furthermore, the ADI ESC-MRAC is currently compared with fixed-gain PID control so as to show that common PID control employed often in various robotic tasks is not efficient to deal with uncertainty. However, by designing a PID control law whose gains adapt real-time by ESC, a more powerful control scheme may arise, which can also deal with uncertainty. Tuning of PID controllers by ESC is already proposed in [32] for single-input single-output systems. However, in our scenario, we are interested in an adaptive PID control scheme for multiple-input multiple-output systems. The design of such an ESC-PID scheme is currently under investigation and will be involved in future work.

6. Appendix

6.1. Proof of Theorem 1

We follow the approach of averaging [33] to treat the non-autonomous system

$$\dot{\mathbf{y}} = \dot{\mathbf{e}} = f(t, \mathbf{y}) = E\mathbf{e} + B_e P^* \mathbf{v}. \tag{38}$$

Notation: we denote the average by $AVG(\cdot)$ or, for single-term arguments, by $(\cdot)_{av} \triangleq AVG(\cdot)$.

The average tracking error dynamics are expressed by

$$\dot{\mathbf{e}}_{av} = f_{av} = E\mathbf{e}_{av} - B_e \tilde{A}_{av} \mathbf{v}_{av} \tag{39}$$

and the average parameter error dynamics by

$$\dot{\tilde{\mathbf{a}}}_{av} = 2\, AVG(G\mathbf{m}\mathbf{e}^T Q B_e P)\mathbf{v}_{av}. \tag{40}$$

By choosing the β_i such that E is Hurwitz, the $(\mathbf{e}_{av}, \tilde{\mathbf{a}}_{av}) = (\mathbf{0}, \mathbf{0})$ is an exponentially stable equilibrium point of the system (39). Based on the averaging theorem, the system (38) has a unique exponentially stable T-periodic solution in an $O(\varepsilon)$ neighbourhood of the origin. Let us perform here a Lyapunov analysis. We consider the Lyapunov candidate function $V = \mathbf{e}_{av}^T P_1 \mathbf{e}_{av} + \tilde{\mathbf{a}}_{av}^T P_2 \tilde{\mathbf{a}}_{av}$ where $P_1 \in \mathbb{R}^{4\times4}$ and $P_2 \in \mathbb{R}^{10\times10}$ are

symmetric positive-definite matrices. Then, $\dot{V}$ is given by

$$\dot{V} = \mathbf{e}_{av}^T(-H)\mathbf{e}_{av} - 2\mathbf{v}_{av}^T\tilde{A}_{av}^T B_e^T P_1 \mathbf{e}_{av} + 2\tilde{\mathbf{a}}_{av}^T P_2 \dot{\tilde{\mathbf{a}}}_{av} \tag{41}$$

where $-H = E^T P_1 + P_1 E$ and H is a positive-definite matrix. It is $\dot{V} \leq 0$ if $\mathbf{v}_{av}^T \tilde{A}_{av}^T B_e^T P_1 \mathbf{e}_{av} = \tilde{\mathbf{a}}_{av}^T P_2 \dot{\tilde{\mathbf{a}}}_{av}$ which can be equivalently written as

$$\mathbf{v}_{av}^T \underbrace{\tilde{A}_{av}^T B_e^T P_1 \mathbf{e}_{av}}_{\mathbf{v}_1} = \mathbf{v}_{av}^T \underbrace{2(AVG(G\mathbf{m}\mathbf{e}^T Q B_e P))^T P_2^T \tilde{\mathbf{a}}_{av}}_{\mathbf{v}_2}.$$

Notation: we denote by $(\cdot)_i$ the $i-$th element of the vector $(\cdot)$.

If $\mathbf{v}_{av} = \mathbf{0}$ or $(\mathbf{v}_{av})_i(\mathbf{v}_1 - \mathbf{v}_2)_i = -\sum_{j,j\neq i} (\mathbf{v}_{av})_j (\mathbf{v}_1 - \mathbf{v}_2)_j$ for every i, then the latter condition is always true and $\dot{V}$ is negative-definite. Otherwise, the latter condition becomes

$$\tilde{A}_{av}^T B_e^T P_1 \mathbf{e}_{av} = 2(AVG(\mathbf{m}_g \mathbf{e}^T Q^*))^T P_2 \tilde{\mathbf{a}}_{av} \tag{42}$$

where $\mathbf{m}_g = G\mathbf{m} \in \mathbb{R}^{10}$ and $Q^* = QB_eP$.

Notation: we denote by $\mathbf{1}_n$ and $\mathbf{0}_n$ the row vectors of all ones and all zeros respectively.

We express $\tilde{A}_{av} = A_s diag(\tilde{\mathbf{a}}_{av}) B_s$, $\tilde{\mathbf{a}}_{av} = diag(\tilde{\mathbf{a}}_{av})(\mathbf{1}_{10})^T$ where

$$A_s = \begin{bmatrix} \mathbf{1}_5 & \mathbf{0}_5 \\ \mathbf{0}_5 & \mathbf{1}_5 \end{bmatrix}, \quad B_s = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}^T. \tag{43}$$

By introducing the previous expressions of A_s and B_s into (42), multiplying by the vector $\mathbf{w}_1 = \mathbf{1}_6$ and following a further series of calculations, (42) finally becomes

$$A_s \mathbf{w}_2^T = 2AVG(Pw^*)\mathbf{w}_1^T \tag{44}$$

where $\mathbf{w}_2 \in \mathbb{R}^{1\times 10}$ is a known-constant vector, $\mathbf{w}_2 P_2 \mathbf{m}_g = w^* \in \mathbb{R}$ and $P_1 = Q$. Given

the condition (44), it is

$$\dot{V} = \mathbf{e}_{av}^T(-H)\mathbf{e}_{av} \leq 0 \tag{45}$$

and, thus, $\mathbf{e}_{av}$, $\tilde{\mathbf{a}}_{av}$ and, in turn, $\mathbf{v}_{av}$ are bounded. Therefore, $\ddot{V} = -\mathbf{e}_{av}^T(E^TH + HE)\mathbf{e}_{av} + \mathbf{e}_{av}^T(H^T + H)B_e\tilde{A}_{av}\mathbf{v}_{av}$ is also bounded. Based on Barbalat's lemma, it is $\lim_{t\to\infty}\dot{V} = 0$, and from (45), $\lim_{t\to\infty}\mathbf{e}_{av} = 0$. In addition, it is $V(\mathbf{e}_{av} = \mathbf{0}, \tilde{\mathbf{a}}_{av} = \mathbf{0}) = 0$ and $V > 0$, $\forall\ \mathbf{e}_{av} \neq \mathbf{0}$, $\forall\ \tilde{\mathbf{a}}_{av} \neq \mathbf{0}$. Given that $\tilde{\mathbf{a}}_{av}$ is bounded, it is $V \to \infty$ for $\mathbf{e}_{av} \to \infty$ and $\dot{V} < 0$, $\forall\ \mathbf{e}_{av} \neq \mathbf{0}$, $\forall\ \tilde{\mathbf{a}}_{av} \neq \mathbf{0}$. Thus, based on theorem 4.2 of [33], the $\mathbf{e}_{av}$ globally and asymptotically converges to zero. Based on the averaging theorem, the $\mathbf{e}$ globally and asymptotically converges to an $O(\varepsilon)$ neighbourhood of zero.

6.2. Proof of Corollary 1

The average tracking error dynamics are expressed by

$$\dot{\mathbf{e}}_{av} = E\mathbf{e}_{av} - B_e\tilde{A}_{av}\mathbf{v}_{av} + B_e\boldsymbol{\epsilon}_{av} \tag{46}$$

and the average parameter error dynamics $\dot{\tilde{\mathbf{a}}}_{av}$ by (40). If $\boldsymbol{\epsilon}_{av} = \mathbf{0}$, (46) becomes the same with (39) and the analysis of the Appendix 6.1 applies.

6.3. Proof of Theorem 2

Let us consider the system $\dot{\mathbf{e}} = f(t, \mathbf{y}) = E\mathbf{e} + B_eP_n^*\mathbf{v}_n$ whose average is given by $f_{av}(\mathbf{y}) = \dot{\mathbf{e}}_{av} = E\mathbf{e}_{av} - B_e\tilde{A}_{n_{av}}\mathbf{v}_{n_{av}}$. The average parameter error dynamics are expressed by $\dot{\tilde{\mathbf{a}}}_{n_{av}} = 2\,AVG(G_n\mathbf{m}_n\mathbf{e}^TQB_eP_n)\mathbf{v}_{n_{av}}$. Let us consider the Lyapunov function $V = \mathbf{e}_{av}^TP_1\mathbf{e}_{av} + \tilde{\mathbf{a}}_{n_{av}}^TP_2\tilde{\mathbf{a}}_{n_{av}}$ where $P_1 \in \mathbb{R}^{4\times4}$ and $P_2 \in \mathbb{R}^{M\times M}$ are symmetric positive-definite matrices. The $\dot{V}$ is given by

$$\dot{V} = \mathbf{e}_{av}^T(-H)\mathbf{e}_{av} - 2\mathbf{v}_{n_{av}}^T\tilde{A}_{n_{av}}^TB_e^TP_1\mathbf{e}_{av} + 2\tilde{\mathbf{a}}_{n_{av}}^TP_2\dot{\tilde{\mathbf{a}}}_{n_{av}}$$

where $-H = E^TP_1 + P_1E$ and H is a positive-definite matrix. It is $\dot{V} \leq 0$ if

$$\tilde{A}_{n_{av}}^TB_e^TP_1\mathbf{e}_{av} = 2(AVG(\mathbf{m}_g\mathbf{e}^TQ_n^*))^TP_2\tilde{\mathbf{a}}_{n_{av}}$$

where $\mathbf{m}_g = G_n\mathbf{m}_n \in \mathbb{R}^M$ and $Q_n^* = QB_eP_n$. We express $\tilde{A}_{n_{av}} = A_s diag(\tilde{\mathbf{a}}_{n_{av}})B_s$ where $B_s \in \mathbb{R}^{M\times L}$, $L = 6 + \Theta + \Psi$, $\tilde{\mathbf{a}}_{n_{av}} = diag(\tilde{\mathbf{a}}_{n_{av}})(\mathbf{1}_M)^T$,

$$A_s = \begin{bmatrix} \mathbf{1}_{M_1} & \mathbf{0}_{M_2} \\ \mathbf{0}_{M_1} & \mathbf{1}_{M_2} \end{bmatrix},$$

$M_1 = 5+\Theta$ and $M_2 = 5+\Psi$. The B_s is an appropriate matrix of 0s and 1s. For instance, if $\Theta = 1$ and $\Psi = 1$, the B_s takes the form

$$B_s = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T.$$

By following an analysis similar to the Appendix 6.1, the following condition is derived

$$A_s\mathbf{w}_2^T = 2AVG(P_n w^*)\mathbf{w}_1^T \tag{47}$$

where $P_1 = Q$, $w^* = \mathbf{w}_2P_2\mathbf{m}_g$, $\mathbf{w}_2 \in \mathbb{R}^{1\times M}$ is a known-constant vector and $\mathbf{w}_1 = \mathbf{1}_L$. Given (47), it can be shown that $\mathbf{e}$ globally and asymptotically converges to an $O(\varepsilon)$ neighbourhood of the origin, see Appendix 6.1.

6.4. Proof of the average parameter error dynamics

It holds that $\dot{\tilde{\mathbf{a}}}_{av} = G\mathbf{l}_{av}J_{av} + G\mathbf{m}_{av}\dot{J}_{av}$ and, given that $\mathbf{l}_{av} = 0$, the latter equation becomes $\dot{\tilde{\mathbf{a}}}_{av} = AVG(2G\mathbf{m}\mathbf{e}^TQ\dot{\mathbf{e}})$. By substituting $\dot{\mathbf{e}}$ into $\dot{\tilde{\mathbf{a}}}_{av}$ and given that $\mathbf{m}_{av} = 0$, we receive

$$\dot{\tilde{\mathbf{a}}}_{av} = 2\ AVG(G\mathbf{m}\mathbf{e}^TQB_eP\mathbf{v}).$$

ACKNOWLEDGMENT

This research is partly supported by the the ERC Starting Grant Control based on Human Models (con-humo) under grant agreement 337654, the TUM-Institute for Advanced Study and the scholarship 'TUM-Equal Opportunity for Women in Research and Teaching'.

References

[1] P. Falco, C. Natale, On the stability of closed-loop inverse kinematics algorithms for redundant robots, IEEE Transactions on Robotics 27 (4) (2011) 780–784. doi:10.1109/TRO.2011.2135210.

[2] D. Lee, C. Ott, Y. Nakamura, Mimetic communication with impedance control for physical human-robot interaction, in: IEEE International Conference on Robotics and Automation (ICRA), 2009, pp. 1535–1542. doi:10.1109/ROBOT.2009.5152857.

[3] D. Mitrovic, S. Klanke, S. Vijayakumar, Adaptive optimal feedback control with learned internal dynamics models, in: From Motor Learning to Interaction Learning in Robots, Vol. 264 of Studies in Computational Intelligence, Springer Berlin Heidelberg, 2010, pp. 65–84.

[4] J. Nakanishi, J. A. Farrell, S. Schaal, A locally weighted learning composite adaptive controller with structure adaptation, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2002, pp. 882–889. doi:10.1109/IRDS.2002.1041502.

[5] V. Koropouli, S. Hirche, D. Lee, Learning and generalizing force control policies for sculpting, 2012, pp. 1493–1498. doi:10.1109/IROS.2012.6385957.

[6] V. Koropouli, D. Lee, S. Hirche, Learning interaction control policies by demonstration, 2011, pp. 344–349. doi:10.1109/IROS.2011.6094657.

[7] A. Schmidts, D. Lee, A. Peer, Imitation learning of human grasping skills from motion and force data, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2011) 1002–1007doi:10.1109/IROS.2011.6094951.

[8] J. Peters, S. Schaal, Learning to control in operational space, International Journal of Robotics Research 27 (2) (2008) 197–212. doi:10.1177/0278364907087548.

[9] J. Buchli, F. Stulp, E. Theodorou, S. Schaal, Learning variable impedance control, International Journal of Robotics Research (2011) 820–833doi:10.1177/0278364911402527.

[10] J. Cheah, C. Liu, J. E. Slotine, Adaptive tracking control for robots with unknown kinematic and dynamic properties, International Journal of Robotics Research 25 (3) (2006) 283–296. doi:10.1177/0278364906063830.

[11] B. Yao, Adaptive robust control of robot manipulators: Theory and comparative experiments, in: Proc. of the 2nd CWC on ICIA, 1997, pp. 442–447.

[12] J. Garus, Model reference adaptive control of underwater robotic vehicle in plane motion, in: Proceedings of the 11th WSEAS International Conference on Systems, 2007, pp. 38–43.

[13] S. Tzafestas, G. Stavrakakis, A. Zagorianos, Robot model reference adaptive control through lower/upper part dynamic decoupling, Journal of Intelligent and Robotic Systems 1 (2) (1988) 163–184. doi:10.1007/BF00348721.

[14] K. S. Narendra, A. M. Annaswamy, Stable adaptive systems, New York: Dover Publications, 2005.

[15] K. J. Astrom, B. Wittenmark, Adaptive control, New York: Dover Publications, 2008.

[16] B. S. Reddy, H. N. Kota, J. Chandrasekhar, Design of model reference adaptive control for unstable aircraft, in: International Conference on Control, Vol. 1, 1991, pp. 521–526.

[17] J. X. Guo, Y. Liu, G. Tao, Multivariable MRAC with state feedback for output tracking, in: Proceedings of the American Control Conference, 2009, pp. 592–597. doi:10.1109/ACC.2009.5160489.

[18] H. Poorya, ES-MRAC: A new paradigm for adaptive control, theory and applications, Ph.D. thesis, Purdue University, West Lafayette, IN (2013).

[19] P. Haghi, K. Ariyur, Adaptive feedback linearization of nonlinear MIMO systems using ES-MRAC, in: American Control Conference, 2013, pp. 1823–1833.

[20] P. Haghi, K.B.Ariyur, On the extremum seeking of model reference adaptive control in higher-dimensional systems, in: Proceedings of the American Control Conference, 2011, pp. 1176–1181. doi:10.1109/ACC.2011.5991563.

[21] C. Zhang, R. Ordonez, Extremum-Seeking Control and Applications, New York: Springer, 2012.

[22] B. K. Ariyur, M. Krstić, Real-Time Optimization by Extremum-Seeking Control,, Wiley, 2003.

[23] P. Binetti, K. Ariyur, M. Krstić, F. Bernelli, Control of formation flight via extremum seeking, in: Proceedings of the American Control Conference, Vol. 4, 2002, pp. 2848–2853. doi:10.1109/ACC.2002.1025221.

[24] N. J. Killingsworth, M. Krstić, PID tuning using extremum seeking: online, model-free performance optimization, IEEE Control Systems 26 (1) (2006) 70–79. doi:10.1109/MCS.2006.1580155.

[25] V. Koropouli, A. Gusrialdi, D. Lee, ESC-MRAC of MIMO systems for constrained robotic motion tasks in deformable environments, in: European Control Conference (ECC), 2014, pp. 2109–2114. doi:10.1109/ECC.2014.6862249.

[26] V. Koropouli, A. Gusrialdi, D. Lee, An adaptive dynamic inversion-extremum seeking control approach for constrained robotic motion tasks, in: European Control Conference (ECC), 2015, pp. 2791–2796.

[27] N. Hogan, Impedance control: An approach to manipulation: Part I, Part II, Part III, ASME Transactions, Journal of Dynamic Systems, Measurement and Control 107 (1985) 1–24.

[28] C. Ott, Cartesian Impedance Control of Redundant and Flexible–Joint Robots, 1st Edition, Springer Publishing Company, Incorporated, 2008.

[29] L. Sciavicco, B. Siciliano, Modeling and Control of Robot Manipulators, Springer-Verlag New York, 2000.

[30] S. Schaal, C. G. Atkeson, Constructive incremental learning from only local information, Neural Computation 10 (1997) 2047–2084. doi:10.1162/089976698300016963.

[31] B. Hunnekens, M. Haring, N. van de Wouw, H. Nijmeijer, A dither-free extremum-seeking control approach using 1st-order least-squares fits for gradient estimation, in: Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on, 2014, pp. 2679–2684. doi:10.1109/CDC.2014.7039799.

[32] N. Killingsworth, M. Krstic, Pid tuning using extremum seeking: online, model-free performance optimization, Control Systems, IEEE 26 (1) (2006) 70–79.

[33] H.K.Khalil, Nonlinear systems, New Jersey: Prentice Hall, 2002.