POLITECNICO DI TORINO
Repository ISTITUZIONALE

Learning enhanced model reference adaptive control algorithms via a cost-effective educational plant

(Article begins on next page)

13 May 2026

Learning enhanced model reference adaptive control algorithms via a cost-effective educational plant

Umberto Montanaro[1] iD,
George Whitehead[1] iD,
and Aldo Sorniotti[2] iD

Abstract

The enhanced model reference adaptive control (EMRAC) strategy is an effective extension of the model reference adaptive control (MRAC) algorithm, designed to steer the trajectories of plants towards those of a reference model despite uncertain plant parameters, unknown nonlinear dynamics, and disturbances by adapting the control gains in operation. However, MRAC algorithms are usually taught only in advanced control engineering courses, as they are more complex compared to linear control methods. To demonstrate the potential of EMRAC solutions for controlling systems with limited knowledge of plant dynamics and to facilitate students' learning, this article presents the control formulation, design, and experimental validation of EMRAC strategies for an educational plant (i.e., the speed control of a DC motor through Arduino). Despite being modelled as a first order system, this plant is characterised by system nonlinearities and parameter uncertainties which can jeopardise closed-loop tracking performance of a controller with fixed gains. An incremental approach is used to introduce the EMRAC algorithm, with only knowledge in linear control design, e.g., pole-placement methods, assumed as a prerequisite. Specifically, the need for MRAC strategies is first introduced as a possible solution to overcome limitations of pole-placement techniques by adapting the control gains. Then, the classical MRAC strategy, along with methods for limiting the growth of the gains, is reviewed, and finally, two EMRAC strategies are

[1]Centre for Aerodynamics, Aerospace and Automotive Engineering, School of Engineering, University of Surrey, Guildford, United Kingdom
[2]Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Torino, Italy

Corresponding author:
Umberto Montanaro, Stag Hill, University Campus, Guildford GU2 7XH.
Email: u.montanaro@surrey.ac.uk

presented. The improved closed-loop tracking performance given by the EMRAC strategies are also assessed quantitatively and compared to those obtained with two linear control solutions and two classical MRAC methods.

Keywords
Adaptive control, model reference adaptive control, control education

Introduction

Adaptive control emerged in the early 1950s with the aim of controlling the dynamics of systems with significant parametric uncertainty and high variability.[1] Within this framework, model reference adaptive control (MRAC) is an effective design method to control engineering plants with unknown parameters. In the MRAC formulations, control specifications (e.g., closed-loop step response parameters, such as settling time) are embedded into the design of a reference model that describes the ideal behaviour of the closed-loop control system. The dynamics of the reference model are then imposed to the plant though a control algorithm where the control gains are adjusted, or preferably adapted, in operation based on the tracking error, i.e., the mismatch between the plant and reference model dynamics, with an adaptive mechanism consisting of a set of differential-integral equations. The resulting evolution of the control gains is the key to steer the plant dynamics towards those of the reference model despite plant parameter uncertainties. The working principle schematic of any MRAC algorithm is shown in Figure 1 where the tracking error is used by the adaptive block for the computation of the time-varying control gains. Since its early formulations, researchers and practitioners have been interested in the design, application, and theoretical analysis of MRAC strategies (the reader is referred to textbooks, e.g., Tao,[1] Landau,[2] Ioannou and Fidan,[3] Landau et al.,[4] Nguyen[5] for the design and applications of classical MRAC strategies).

Nowadays, MRAC research targets improving the closed-loop tracking performance of classical algorithms, for instance, by augmenting the MRAC control action with
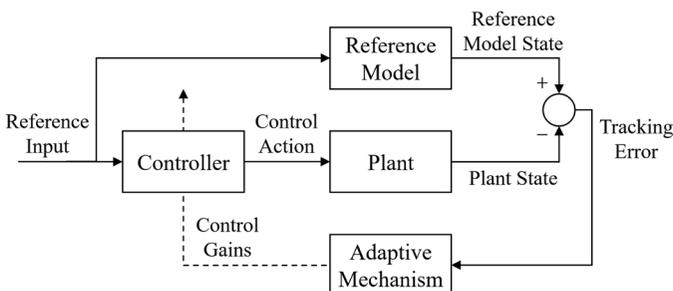
Figure 1. Block diagram of a typical MRAC algorithm.

other control strategies, such as repetitive learning control,[6] sliding mode control,[7] fuzzy strategies,[8] and artificial intelligence techniques,[9] to name a few. Additionally, MRAC formulations and theory have been expanded to address dynamic systems with fractional order,[10] switching control systems,[11,12] piecewise smooth systems,[13–15] systems with actuation constraints,[16,17] and networks of dynamic systems.[18,19] Reviews on the application of MRAC strategies to various engineering control problems are also available, such as for robotic arms,[20] sensorless control of induction motor drives,[21] aerospace systems[5] (Chapter 10) and other engineering applications like power systems and process control.[22]

To improve closed-loop tracking performance not only in the presence of plant parameter uncertainties but also in response to unmodeled plant dynamics, rapidly varying disturbances, and unknown system nonlinearities, the MRAC strategy was augmented in di Bernardo et al.[23] with an adaptive integral control action and an adaptive switching control action. Since then, this augmented MRAC algorithm — referred to in the literature as enhanced MRAC (EMRAC) strategy[24] — has proven to be an effective solution for controlling the dynamics of engineering plants affected by disturbances and model uncertainties toward the behavior of a reference model.

Engineering control applications that have been tackled through EMRAC solutions include *(i)* automotive electromechanical valves, i.e., the electronic throttle body,[23,25] *(ii)* common rail systems for gasoline direct injection engines,[26] *(iii)* thermo-hygrometric control of a single thermal zone[27] and multi-enclosed thermal zones,[28] *(iv)* path tracking control for autonomous vehicles,[29] *(v)* longitudinal vehicle control to impose linear dynamics for vehicle platooning applications,[30] *(vi)* robotic manipulators,[31] *(vii)* direct yaw moment control for improving vehicle safety during cornering,[32] and *(viii)* control of quadcopters.[33] Theoretical extensions of the EMRAC algorithm are also available. For instance, a discrete-time formulation of the EMRAC strategy was presented in Montanaro et al.[25] Its closed-loop tracking performance was experimentally evaluated against other robust adaptive solutions and classical MRAC techniques, showing that the additional EMRAC control actions play a key role in improving the tracking of the reference dynamics. Further theoretical extensions of the EMRAC algorithm include its formulation for multi-input systems,[31] and an output-based formulation based on the concept of closed-loop reference models.[32]

From an academic curriculum perspective, MRAC strategies are a fundamental tool for controlling plants with uncertain parameters. Furthermore, the analysis of the resulting closed-loop system provides students with a practical case where stability theories for nonlinear systems (such as the use of Lyapunov functions or passivity) are required to prove the convergence of the closed-loop tracking error. Consequently, MRAC solutions are typically introduced in graduate-level control theory modules and are not covered in undergraduate engineering courses, which tend to focus more on linear control methods for linear time invariant (LTI) systems in the Laplace and state-space domains. However, undergraduate students can gain exposure to the general working principles of MRAC strategies and their application to engineering control problems through modules based on individual projects or dissertations, where they might have the opportunity to explore, design, and apply control methods beyond those covered in linear control system modules.

Although a formal study on students' learning processes of adaptive control strategies is beyond the scope of this work, authors' experience has shown that when students with knowledge only in linear control theory learn the key idea and working principles behind the MRAC and EMRAC strategies along with their advantages through simple engineering examples, the grasping of more sophisticated EMRAC formulations and their application to more challenging engineering case studies is facilitated, e.g., those considered for individual students' projects.[34–38] This is in line with learning-by-doing pedagogy theories, which include benefits such as enhanced retention of knowledge, improved motivation, and strengthened problem-solving skills,[39] as well as the use of benchmark problems in control education recently outlined in Visioli and Stapleton.[40]

Hence, this paper provides an introduction to the EMRAC algorithm, its design and the experimental assessment of the closed-loop tracking performance for a first-order linear system on an educational plant. The only pre-requisite is knowledge in modelling and control of LTI systems in the state-space domain (e.g., pole-placement control methods) usually taught in linear control modules. Moreover, an incremental approach is adopted to introduce the EMRAC strategy. Specifically, a link to classical linear control methods is first set by outlining the potential limitations of the pole-placement method in the presence of plant parameter uncertainties. Within this framework, MRAC solutions are presented as a remedy to impose the required closed-loop dynamics without knowledge of the plant parameters. Second, the control objective of the MRAC formulation is outlined along with the classical MRAC algorithm for first order LTI systems. Then, the drift of the adaptive control gains that might occur when unmodeled nonlinear dynamics and/or disturbances act on the plant[5] is discussed. Moreover, the parameter-projection (PP) method and the σ-modification strategy are presented as solutions available in the adaptive control literature to prevent this unwanted gain instability. Finally, the EMRAC algorithm is introduced, and the two additional adaptive control terms characterising EMRAC strategies are discussed along with the possibility to equip the method with solutions to limit the growth of the gains. Thus the EMRAC with parameter-projection and the EMRAC with σ-modification, denoted as EMRAC-PP and EMRAC-σ, respectively, are presented. Key closed-loop properties guaranteed by EMRAC methods are also discussed through a set of remarks, but the related proofs are omitted as they are beyond the scope of this article. However, a sketch of the proof of the convergence to zero of the tracking error through Lyapunov theory for first-order systems is provided in the Appendix. Although it is presented for a simple case, the steps outlined in the proof are also those followed to show convergence of the tracking error for more advanced EMRAC formulations.

The educational system used to investigate the closed-loop tracking performance of EMRAC strategies and benchmark solutions is the speed control of a DC motor through a cost-effective Arduino system. Cost-effective programming boards are nowadays commonly used in educational environments for different purposes ranging from promoting students' creativity to exploring scientific concepts.[41] The adopted system can be programmed using Simulink, which is commonly utilized in undergraduate control engineering modules, allowing students to seamlessly deploy control strategies from the simulation environment to the Arduino board. Examples of using Arduino

boards for educational purposes in control engineering can be found in the literature, e.g., Li,[42] Takács et al.,[43] Barbosa,[44] Chuckpaiwong and Boekfah,[45] Vargová et al.,[46] but to the best of the authors' knowledge, they do not cover MRAC solutions. A linear model of the plant is also identified for the design of *(i)* two linear benchmark controllers, and *(ii)* the reference model embedded in four MRAC solutions (i.e., the EMRAC-PP and the EMRAC-σ algorithms and two baseline MRAC strategies). The residual parameter uncertainties and plant nonlinearities, which act as unmodeled dynamics, are also discussed. Furthermore, the analysis of the closed-loop experimental dynamics is used to unfold and discuss several key features of the implemented algorithms, including the effectiveness of the EMRAC solutions in tracking the reference model state and the detrimental effects of actuation limits on the baseline adaptive solutions.

Summarising the contributions of this paper are:

- a revision of the EMRAC strategies for first-order systems for educational purposes, requiring as a prerequisite only knowledge of linear control systems typically taught in undergraduate engineering modules;
- the design and implementation of two EMRAC solutions within a cost-effective educational plant for supporting teaching and learning of MRAC methods;
- the experimental assessment of the closed-loop tracking performance of the two EMRAC strategies against four benchmarking control solutions to further motivate the learning of advanced MRAC methods.

Experimental setup

The experimental setup is based on the Arduino Engineering Kit Rev2[47] and includes *(i)* an Arduino Nano 33 IoT, a 3.3 V microcontroller with a SAMD21 Cortex-M0+ 32bit ARM processor with 256 kB Flash and a maximum clock frequency of 48 MHz; *(ii)* a 12 V DC motor TRK-370CA-17260-51V-EN which includes *(iii)* a quadrature hall effect encoder which reads at 48 counts per revolution, and is attached to *(iv)* a flywheel with a diameter of 0.1 m and a mass of 0.075 kg; *(v)* a support for the motor and flywheel made with components provided with the kit; and *(vi)* a 4 V Li-ion (Samsung INR 18650-25R) battery which is increased to near 12 V for the motor by using a synchronous boost converter on the Arduino nano motor carrier.

The computer used for programming the microcontroller, data collection and post-processing has an Intel i7 8700k with 16 GB RAM. A USB cable is used to connect the computer to the Arduino board for enabling data transfer, the deployment of the control architecture, and collection of the experimental data. The experimental setup is depicted in Figure 2.

The control architecture, including the control strategies and signal conditioning operations, have been coded in Simulink version 10.6 available in MATLAB R2022b. The schematic of the information flow among the key components and the control software coded in Simulink and running on the microcontroller are depicted in Figure 3. The readings from the encoder are sent to the control software, converted to radians and differentiated with respect to time according to the method recommended by the
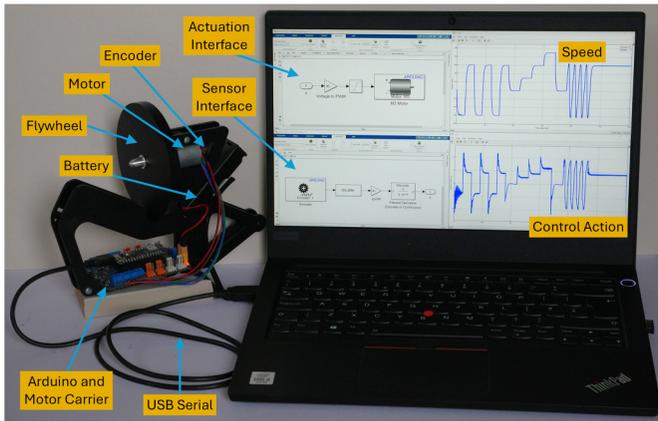
Figure 2. Experimental setup.
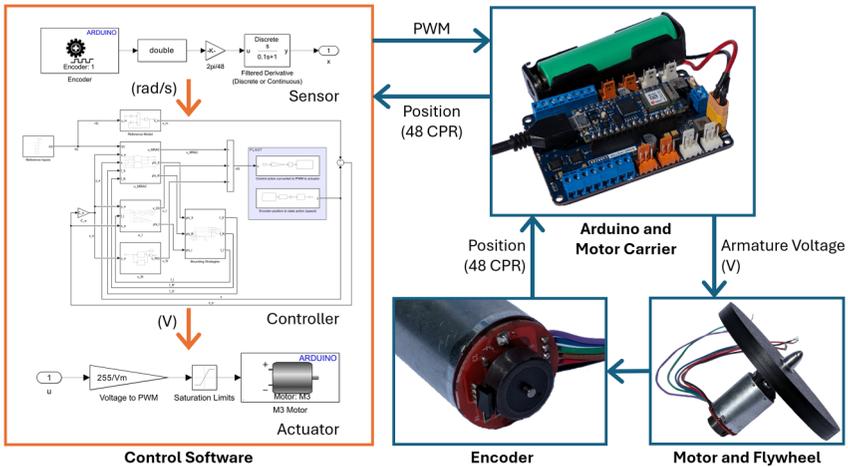
Figure 3. Schematic of the information flow among the components and the control software.

documentation of the Arduino Engineering Kit Rev 2[48] to obtain the motor/flywheel speed (see also the "Sensor" section in the control software of Figure 3). This speed information is processed together with other signals, such as, the reference speed, by the control algorithm which generates a request of armature voltage to steer the motor speed towards the reference speed. The corresponding pulse width modulation (PWM) signal sent to the power electronics of the motor is represented by a real number in the interval $[-255, 255]$ and is obtained from the control action with PWM =

sat(255u/V$_m$) (see also the "Actuator" section in Figure 3), where $V_m = 12$ V is the maximum voltage across the motor and sat is the saturation function to limit the PWM variable in the admissible range.

The "Encoder" block and "M3 Motor" block are provided by the Simulink Support Package for Arduino Hardware add on for MATLAB/Simulink. The MATLAB Support Package for Arduino Hardware is also required to allow the Arduino to connect to MATLAB/Simulink. The Simulink model is deployed to the Arduino by selecting Arduino Nano 33 IoT as the target "Hardware Board". The code is deployed in the Arduino board by using the "Monitor and Tune" mode, thus allowing monitoring of the experiment and data acquisition. The Simulink model is compiled with the discrete solver ode2 (i.e., the Heun solver) with a sampling time of $t_s = 10$ ms. Experimental data is collected over the USB serial connection and saved in a MATLAB workspace.

Modelling and experimental validation

The DC motor speed dynamics can be modelled as a LTI first-order system when the electrical dynamics are much faster than the mechanical dynamics and nonlinearities are neglected.[49] Under these assumptions, by denoting ω (rad/s) as the motor speed (plant state) and u (V) as the armature voltage (control input), the plant model is

$$\dot{\omega} = -\left(\frac{b_f}{J} + \frac{K_t K_e}{JR}\right)\omega + \frac{K_t}{JR}u, \tag{1}$$

where J (kgm^2) is the moment of inertia of the flywheel and the motor, b_f (Nms) is the coefficient of the linear friction torque acting on the system, R (Ω) is the armature resistance, K_e (Vs/rad) is the back emf coefficient, and K_t (Nm/A) is the motor torque coefficient.

For the design of control algorithms with fixed control gains (such as model-based proportional integral (PI) and pole placement strategies), estimates of plant parameters are needed. As the moment of inertia of the rotor of the motor can be neglected compared to that of the flywheel, then $J = mr^2/2$, where the flywheel mass m and radius r can be precisely measured, and thus the parameter J can be accurately estimated.

The remaining plant parameters are affected by uncertainties and can be identified through solving a least-square optimisation problem.[50] Specifically, by denoting $p = [K_e\ K_t\ b_f\ R]$ as the vector parameter, the optimal set of the parameters $p^\star$ is selected as the one that minimizes the following cost function

$$W(p) = \int_0^{T_I} (\omega_E(t) - \widehat{\omega}(t; p))^2 dt, \quad \text{with } \mathrm{p}^l \le \mathrm{p} \le \mathrm{p}^u, \tag{2}$$

where $\omega_E(t)$ is the experimental motor speed obtained when the armature voltage is a step function, changing from 11.78 V to -11.78 V with the step occurring at the time instant $t = 20$ s; $\widehat{\omega}(t; p)$ is the solution of model (1) with plant parameters p and subjected to the same armature voltage step function as the experimental plant; $T_I = 40$ s is the duration of the selected identification manoeuvre; and p^l and p^u are vectors that define the lower

and upper bounds of each plant parameter, respectively. Moreover, the integral (2) is discretized with the sampling time t_s.

The optimal estimate of the plant parameters $p^\star$ has been found numerically using the *fmincon* function in MATLAB to minimize the W-function in (2), which represents a measure of the mismatch between the experimental results and model predictions. The reader is referred to articles and books dedicated to system identification, for example[50] and the references therein, for further details on least-squares methods for plant parameter estimation.

The accuracy of model (1) equipped with the parameter $p^\star$ to capture the experimental motor speed when the armature voltage is the step function used for the plant parameter's identification is shown in Figure 4(a). As a validation test, a sinusoidal wave variation of the armature voltage with an amplitude of 11.78 V and a frequency of 0.25 Hz is considered. Figure 4(b) shows a good correlation between the experimental results and the model predictions for the selected validation scenario.

Although the linear model (1) is able to describe the speed dynamics well for the working conditions depicted in Figure 4, further experimental analysis has confirmed the presence of plant nonlinearties which cannot be captured by the DC motor LTI model. Specifically, Figure 5(a) shows that the experimental steady-state speed does not depend linearly with respect to armature voltage. Moreover, *(i)* the linear approximation provided by model (1) is less precise for inputs around ± 5 V, and *(ii)* the flywheel does not spin till the magnitude of the input exceeds a threshold of about 2 V (i.e., there is a dead-zone when $u \in [-2, 2]$). This dead-zone is due to static nonlinear friction torques not included in model (1). Stiction torques also generate nonlinear stick-slip dynamics,[51] in other words motions where the flywheel speed suddenly becomes stuck despite smooth variations of the armature voltage. Figure 5(b) shows an example of stick-slip motion experimentally obtained in steady-state conditions when the armature voltage
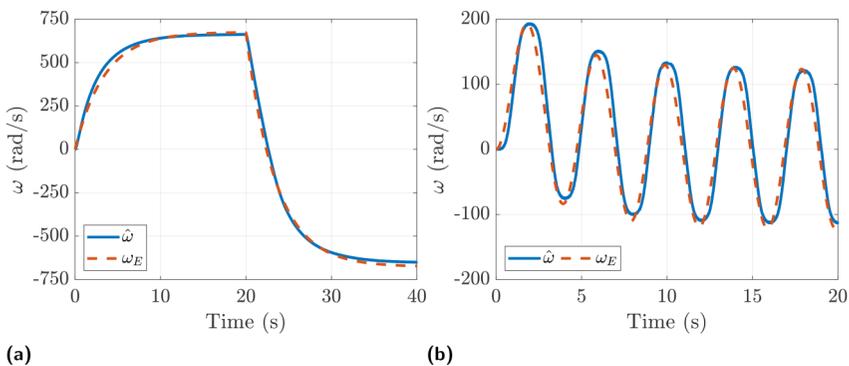
Figure 4. Experimental model validation: ω_E experimental data (red dashed line), and $\widehat{\omega}$ model predictions (blue solid line) when the armature voltage is (a) a step function from 11.78 V to -11.78 V and step time $t = 20$ s (identification manoeuvre); and (b) a sinusoidal wave with amplitude 11.78 V and frequency 0.25 Hz (validation manoeuvre).
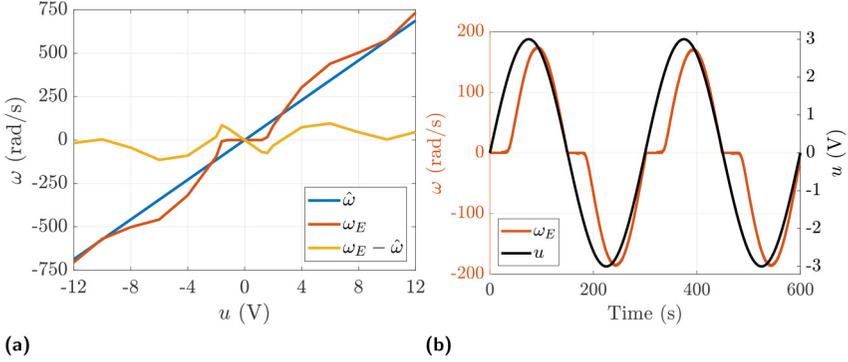
Figure 5. Experimental plant nonlinearities: (a) variation of the steady-state gain, ω_E experimental speed (red line), $\hat{\omega}$ model predictions (blue line) and corresponding mismatch (yellow line); and (b) stick-slip phenomenon, armature voltage input (gray line) and ω_E experimental speed (red line).

varies as a sinusoidal wave with amplitude 3 V and period 300 s. This nonlinear plant behaviour cannot be captured by model (1) as, in accordance with the frequency analysis of LTI systems (as presented in Franklin et al.[52] for example), the steady-state output of an LTI system to any sinusoidal input is a sinusoidal wave with the same frequency as the input.

A further system nonlinearity which is not captured by the LTI model (1) is the saturation of the armature voltage. Specifically, the control input u is saturated when its magnitude exceeds V_m (i.e., the actual armature voltage applied to the system is $u_a = \text{sgn}(u) \min\{|u|, V_m\}$). Saturations in control systems can induce unwanted oscillations of the closed-loop plant response around the required reference and integral windup phenomena when the controller generates control actions exceeding the saturation limits,[53,54] which degrade the tracking performance as also confirmed later with the experimental closed-loop results in the section "Experimental Results".

The aforementioned nonlinearities are examples of unmodeled nonlinearities and disturbances, which, together with the residual plant parameter mismatches, represent the uncertainties towards which the control system should be robust.

Motivation to MRAC and EMRAC solutions

The target for the tracking control problem is the design of a control strategy to impose to the plant a given reference signal. Consider as a simplifying example a first-order system of the form

$$\dot{x} = ax + bu + \delta, \tag{3}$$

where x, u, $\delta \in \mathbb{R}$ are the plant state, control input and disturbance, respectively, while a, $b \in \mathbb{R}$ are the dynamic matrix and input matrix of the plant, respectively. When the plant

parameters are known and in absence of disturbances (meaning $\delta = 0$), a constant reference signal $r \in \mathbb{R}$ can be imposed with preassigned transient dynamics (for example expressed in terms of the settling time of the closed-loop step response) with the following control action

$$u = \phi_X^\star x + \phi_R^\star r, \tag{4}$$

with $\phi_X^\star$ and $\phi_R^\star \in \mathbb{R}$ being constant control gains. The feedback gain $\phi_X^\star$ can be tuned with the pole-placement strategy to guarantee given transient dynamics while the feedforward gain $\phi_R^\star$ is selected to have a closed-loop static gain equal to one, thus having zero steady-state tracking error with respect to the reference signal.[52] Specifically, when the control action (4) is substituted into the plant model (3) and $\delta = 0$, then the closed-loop dynamics are

$$\dot{x} = a_m x + b_m r, \tag{5}$$

with $a_m = a + b\phi_X^\star$ and $b_m = b\phi_R^\star$. The position of the close-loop pole, which determines the closed-loop transient response, is $p = a_m = a + b\phi_X^\star$ and can be adjusted to match a desired pole with an *ad hoc* selection of the control gain $\phi_X^\star$, while the closed-loop static gain can be made equal to one by selecting the feedforward gain as $\phi_R^\star = -b^{-1}(a + b\phi_X^\star)$.[52]

However, plants of engineering interest are often affected by parameter uncertainties and disturbances. By denoting Δa and Δb the uncertainties on the plant parameters a and b, respectively, then actual closed-loop dynamics when the control strategy (4) is applied are

$$\dot{x} = \left(a_m + \Delta a + \Delta b\phi_X^\star\right)x + \left(b_m + \Delta b\phi_R^\star\right)r + \delta. \tag{6}$$

Because the closed-loop vector field (6) differs from the ideal one due to the presence of the term $(\Delta a + \Delta b\phi_X^\star)x + \Delta b\phi_R^\star r + \delta$, the actual evolution of the closed-loop system can deviate drastically from the ideal one in (5). However, MRAC solutions can impose desired dynamics of the form (5) despite parameter uncertainties and disturbances by adjusting, or better adapting, the control gains in operation based on the mismatch between the actual plant dynamics and the desired or reference dynamics.

Model reference adaptive control

The control objective of MRAC solutions is to steer the dynamics of the plant toward those of a linear reference system despite parameter mismatches, unmodeled dynamics and disturbances while guaranteeing the boundedness of all closed-loop signals. The reference model dynamics are given by an asymptotically stable LTI system, which, for first-order systems, can be expressed as

$$\dot{x}_m = a_m x_m + b_m r, \tag{7}$$

where $x_m \in \mathbb{R}$ is the state of the reference model, $r \in \mathbb{R}$ is the reference input assumed to

be bounded, and a_m and $b_m \in \mathbb{R}$ are the dynamic matrix and the input matrix of the reference system, respectively.

Mathematically, the control target is then to design a control solution such that x converges to x_m asymptomatically (i.e., $x \to x_m$ when the time $t \to +\infty$), for any reference input, initial states of the plant, and reference model (that is, for any $x(t_0)$ and $x_m(t_0)$, with t_0 being time instant when the controller is activated), despite plant parameter uncertainties Δa and Δb and disturbances δ. MRAC solutions guarantee this control goal by equipping the control strategy with adaptive laws (also known as adaptive rules or mechanics).

The simplest MRAC solution is the following[3]

$$u_{\mathrm{MRAC}}(t) = K_X(t)x + K_R(t)r, \tag{8}$$

where the feedback gain K_X and the feedforward gain K_R change over time according to the following adaptive laws

$$K_X = \phi_X, \quad \text{and} \quad \dot{\phi}_X = \alpha_X y_e x, \tag{9a}$$

$$K_R = \phi_R, \quad \text{and} \quad \dot{\phi}_R = \alpha_R y_e r, \tag{9b}$$

where α_X and $\alpha_R \in \mathbb{R}$ are constants known as adaptive weights, and

$$y_e = b_m x_e. \tag{10}$$

with x_e being the tracking error which is defined as the mismatch between the reference state and the plant state, i.e.,

$$x_e = x_m - x. \tag{11}$$

Remarks:

- For MRAC solutions, the reference model is designed based on the control specifications to be imposed onto the closed-loop control system (for example specifications on the transient response or frequency response).
- To guarantee that the plant state tracks that of the reference model, MRAC solutions usually require that some *matching conditions* are met. Specifically, the existence (but not the knowledge) of a control action of the form $u = \widehat{\phi}_X x + \widehat{\phi}_R r$, where $\widehat{\phi}_X$ and $\widehat{\phi}_R$ are constant gains, which is able to transform the plant dynamics into the reference model system (such that $a_m = a + b\widehat{\phi}_X$ and $b_m = b\widehat{\phi}_R$) is required. Matching conditions are always met for plants whose model is expressed in control canonical form (for example first-order systems and electromechanical plant models derived by using the Euler-Lagrange equation). Moreover, for control problems of engineering interest, nominal plant parameters are usually available and can be used to design the reference model such that matching conditions are satisfied. A possible approach is to select as the reference dynamics the nominal plant model controlled through feedback/feedforward control solutions (for example linear quadratic strategies or pole-placement strategies, thus $\widehat{\phi}_X = \phi_X^\star$ and $\widehat{\phi}_R = \phi_R^\star$ in (4)). For first-order

systems the ideal gains $\widehat{\phi}_X$ and $\widehat{\phi}_R$ are

$$\widehat{\phi}_R = b^{-1}b_m \quad \text{and} \quad \widehat{\phi}_X = b^{-1}(a_m - a). \tag{12}$$

- The adaptive weights α_X and α_R must have the same sign as $\widehat{\phi}_R$ and represent the rate of variation of the adaptive gains.
- The dynamics of the adaptive gains of MRAC solutions (as in (9)) are nonlinear and become states of the control system. Consequently, the analysis of the closed-loop system cannot be carried out via methods used for linear systems (for example utilising the study of the position of eigenvalues[52]) and more sophisticated methods for assessing closed-loop stability and convergence of the error, for example Lyapunov theory and passivity theory, must be used.[3]
- The classical MRAC solution in (8) can asymptotically impose the dynamics of the reference model to the plant (i.e., $x_e \to 0$ when $t \to +\infty$) in the case of uncertain plant parameters and square-integrable unknown disturbances, that is $\delta \in \mathcal{L}_2$ (in other words disturbances for which $\int_{t_0}^{+\infty} \|\delta(t)\|^2 dt$ is finite). On the contrary, EMRAC control solutions can guarantee zero tracking error for a larger class of disturbances, namely bounded disturbances (expressly $\delta \in \mathcal{L}_\infty$, or equivalent disturbances for which there exists a positive constant $\mathcal{G}$ such $\|\delta(t)\| < \mathcal{G}$ for all $t \geq t_0$). This is a key benefit of EMRAC solutions which also opens the way for their applications to plants characterized by non-vanishing disturbances.

Adaptive gains locking strategies

In the case where the plant dynamics are affected by persistent disturbances or unmodeled dynamics (e.g., $\delta = \delta(x, t)$) an unbounded drift of adaptive gains may arise when classical adaptive laws, such as those in (9), are utilized to update the adaptive gains,[5] as is also shown in the adaptive control literature through the Rohrs' example.[55,56] Unbounded adaptive gains may not only lead to a degradation of the tracking performance of the reference model but may also lead to closed-loop instability. Mathematically, boundedness of the adaptive gains when adaptive laws as in (9) are used cannot be guaranteed in the presence of persistent disturbances and unmodeled dynamics since the derivative of the Lyapunov function of the closed-loop system is not strictly negative with respect to the adaptive gains.[3,5]

To guarantee bounded adaptive gains despite disturbances and unmodeled dynamics, robust modifications to the classical MRAC adaptive mechanisms are presented in the literature.[3,5] Broadly speaking, these techniques aim to either *(i)* limit the adaptive gains within a preassigned set; or *(ii)* equip the adaptive laws with damping mechanisms to prevent the unbounded drift of the adaptive gains. Within this framework, the parameter projection strategy falls under the first category, while the σ-modification method is an example of robust modification belonging to the second category. In what follows both of these methods are revised for first order systems. Moreover, both the robust

modifications adjust the adaptive laws (9) as

$$\dot{\phi}_X = \alpha_X y_e x + f_X, \tag{13a}$$

$$\dot{\phi}_R = \alpha_R y_e r + f_R, \tag{13b}$$

where the functions f_X and f_R depend on the strategy adopted to bound the evolution of ϕ_X and ϕ_R, respectively.

For the design of the robust terms f_X and f_R, the only extra assumption that is required, compared to the classical MRAC algorithm, is that some bounds for the plant parameters are known. This hypothesis is not as restrictive as it might appear at first because the range of variation of the plant parameters are often available for engineering control problems. Specifically, the parameter-projection strategy requires

$$\phi_X^l \leq \widehat{\phi}_X \leq \phi_X^u, \quad \text{and} \quad \phi_R^l \leq \widehat{\phi}_R \leq \phi_R^u, \tag{14}$$

with ϕ_X^l, ϕ_R^l, ϕ_X^u, and ϕ_R^u being known bounds for $\widehat{\phi}_X$ and $\widehat{\phi}_R$. Furthermore, by defining $\widehat{\phi} = [\widehat{\phi}_X \ \widehat{\phi}_R]^T$, the σ-modification method requires

$$\|\widehat{\phi}\| \leq \mathcal{M}, \tag{15}$$

with $\mathcal{M}$ being a known positive constant.

Note that, the bounds ϕ_X^l, ϕ_R^l, ϕ_X^u, ϕ_R^u and $\mathcal{M}$ can be found based on bounds on the plant parameters, their nominal values, and reference model parameters, for example, as shown in Montanaro and Olm,[24] Montanaro and Olm[57] for systems in control canonical forms.

In the case of parameter-projection, the locking functions f_X and f_R in (13) are selected as

$$f_X(t) = \begin{cases} 0, & \text{if } \phi_X \in \left(\phi_X^l, \phi_X^l\right), \text{ or } \phi_X = \phi_X^l \text{ and } h_X \geq 0, \\ & \qquad\qquad \text{or } \phi_X = \phi_X^u \text{ and } h_X \leq 0, \\ -h_X(t), & \text{otherwise,} \end{cases} \tag{16a}$$

$$f_R(t) = \begin{cases} 0, & \text{if } \phi_R \in \left(\phi_R^l, \phi_R^l\right), \text{ or } \phi_R = \phi_R^l \text{ and } h_R \geq 0, \\ & \qquad\qquad \text{or } \phi_R = \phi_R^u \text{ and } h_R \leq 0, \\ -h_R(t), & \text{otherwise,} \end{cases} \tag{16b}$$

with $h_X = \alpha_X y_e x$ and $h_R = \alpha_R y_e r$.

When the adaptive strategies (13) are coupled with (16), then the dynamics of the adaptive gains ϕ_X and ϕ_R are constrained in the sets $[\phi_X^l, \ \phi_X^u]$ and $[\phi_R^l, \ \phi_R^u]$, respectively, that is

$$\phi_X(t) \in \left[\phi_X^l, \ \phi_X^u\right], \text{ and } \phi_R(t) \in \left[\phi_R^l, \ \phi_R^u\right], \text{ for all } t \geq 0. \tag{17}$$

In the case of the σ-modification strategy, the adaptive mechanism (13) is completed with

$$f_X = -\rho_X \sigma_\phi(\|\phi\|)\phi_X, \quad \text{and} \quad f_R = -\rho_R \sigma_\phi(\|\phi\|)\phi_R, \tag{18}$$

where ρ_X and ρ_R are positive constants known as leakage factors, $\phi \in \mathbb{R}^2$ is the stack of the

adaptive gains, that is $\phi = [\phi_X \ \phi_R]^T$, and the function $\sigma_\phi(\|\phi\|)$ is defined as

$$
\sigma_\phi(\|\phi\|) = \begin{cases} 0, & \text{if } \|\phi\| \leq \widehat{\mathcal{M}}_\phi, \\ \eta_\phi\left(\dfrac{\|\phi\|}{\widehat{\mathcal{M}}_\phi} - 1\right), & \text{if } \widehat{\mathcal{M}}_\phi < \|\phi\| \leq 2\widehat{\mathcal{M}}_\phi, \\ \eta_\phi, & \text{if } \|\phi\| > 2\widehat{\mathcal{M}}_\phi, \end{cases} \tag{19}
$$

where η_ϕ is a strictly positive constant and

$$
\widehat{\mathcal{M}}_\phi = \sqrt{\frac{\Lambda_{\max}}{\Lambda_{\min}}}\mathcal{M}, \tag{20a}
$$

$$
\Lambda_{\min} = \min\left\{\frac{\rho_X}{\alpha_X}, \frac{\rho_R}{\alpha_R}\right\}, \quad \text{and} \quad \Lambda_{\max} = \max\left\{\frac{\rho_X}{\alpha_X}, \frac{\rho_R}{\alpha_R}\right\}. \tag{20b}
$$

Qualitatively, when the magnitude of the adaptive gains exceeds the threshold $\widehat{\mathcal{M}}_\phi$, negative feedback of the gains are switched on in the adaptive laws (13) which regulate the growth of the adaptive gains and prevent their unbounded drifting.

Enhanced model reference adaptive control

The EMRAC algorithm improves the closed-loop tracking performance of the reference model dynamics by augmenting the MRAC action with an explicit adaptive integral action and an adaptive switching action. When the plant dynamics and reference model are a first-order system, the EMRAC action is

$$
u_{\text{EMRAC}}(t) = K_X(t)x + K_R(t)r + K_I(t)x_I + K_N(t)\text{sgn}(y_e), \tag{21}
$$

where x_I is the integral of the tracking error and its dynamics are computed as

$$
\dot{x}_I = x_e + f_e, \tag{22}
$$

with f_e being a term, modelled later, used to prevent a possible unbounded drift of x_I in the case of residual tracking error because of unmatched disturbance and unmodeled plant dynamics.

The EMRAC adaptive mechanisms are

$$
K_X = \phi_X + \beta_X y_e x, \quad \text{and} \quad \dot{\phi}_X = \alpha_X y_e x + f_X, \tag{23a}
$$

$$
K_R = \phi_R + \beta_R y_e r, \quad \text{and} \quad \dot{\phi}_R = \alpha_R y_e r + f_R, \tag{23b}
$$

$$
K_I = \phi_I + \beta_I y_e x_I, \quad \text{and} \quad \dot{\phi}_I = \alpha_I y_e x_I + f_I, \tag{23c}
$$

$$
K_N = \phi_N, \quad \text{and} \quad \dot{\phi}_N = \alpha_N h_N(|y_e|) - \sigma_N(|\phi_N|)\rho_N\phi_N, \tag{23d}
$$

where $\alpha_X, \alpha_R, \alpha_I, \alpha_N, \beta_X, \beta_R, \beta_I$ are constants with the same sign of $\widehat{\phi}_R$, ρ_N is a strictly

positive constant and

$$h_N(|y_e|) = \frac{|y_e|^{c_3}}{c_1 + c_2|y_e|^{c_3}}, \tag{24}$$

with c_1, c_2 and c_3 being positive constants. The σ-modification function $\sigma_N(|\phi_N|)$ in (23d) is defined as

$$\sigma_N(|\phi_N|) = \begin{cases} 0, & \text{if } |\phi_N| \leq \widehat{\mathcal{M}}_N, \\ \eta_N\left(\frac{|\phi_N|}{\widehat{\mathcal{M}}_N} - 1\right), & \text{if } \widehat{\mathcal{M}}_N < |\phi_N| \leq 2\widehat{\mathcal{M}}_N, \\ \eta_N, & \text{if } |\phi_N| > 2\widehat{\mathcal{M}}_N, \end{cases} \tag{25}$$

where η_N and $\widehat{\mathcal{M}}_N$ are strictly positive constants.

For EMRAC solutions, the σ-modification strategy in (23d) is used to prevent the adaptive gain ϕ_N to diverge in the case of unmatched unmodeled dynamics and disturbances. Furthermore, the positiveness of h_N-function (24) limits the use of the parameter-projection method as a gain locking strategy for ϕ_N, as it prevents the gain ϕ_N from being unlocked once it reaches its admissible upper bound.

For the adaptive control gains ϕ_X, ϕ_R and ϕ_I in (23), both parameter-projection strategies and σ-modification methods can be used. When the EMRAC algorithm is equipped with parameter-projection (PP) strategy it is denoted as EMRAC-PP algorithm. The functions f_X and f_R in (23a) and (23b) are computed as in (16a) and (16b), respectively, while the f_I in (23c) and f_e in (22) are defined as

$$f_I(t) = \begin{cases} 0, & \text{if } \phi_I \in \left(\phi_I^l, \phi_I^u\right), \text{ or } \phi_I = \phi_I^l \text{ and } h_I \geq 0, \\ & \quad\quad \text{or } \phi_I = \phi_I^u \text{ and } h_I \leq 0, \\ -h_I(t), & \text{otherwise}, \end{cases} \tag{26}$$

$$f_e(t) = \begin{cases} 0, & \text{if } x_I \in \left(x_I^l, x_I^u\right), \text{ or } x_I = x_I^l \text{ and } x_e \geq 0, \\ & \quad\quad \text{or } x_I = x_I^u \text{ and } x_e \leq 0, \\ -x_e(t), & \text{otherwise}, \end{cases} \tag{27}$$

where $h_I = \alpha_I y_e x_I$, ϕ_I^l and ϕ_I^u are the selected lower and upper bound for the gain ϕ_I (i.e., $\phi_I(t) \in [\phi_I^l\ \phi_I^u]$, for all $t \geq t_0$), and x_I^l and x_I^u are the lower and upper bound for the integral of the tracking error (i.e., $x_I(t) \in [x_I^l\ x_I^u]$, for all $t \geq t_0$).

In the case the adaptive mechanism (23) is equipped with σ-modification, the resulting strategy is denoted as EMRAC-σ and the function f_X, f_R and f_I are defined as

$$f_X = -\rho_X \sigma_\phi\big(\|\phi\|\big)\phi_X, \ f_R = -\rho_R \sigma_\phi\big(\|\phi\|\big)\phi_R, \text{ and} \\ f_I = -\rho_I \sigma_\phi\big(\|\phi\|\big)\phi_I, \tag{28}$$

where ρ_X, ρ_R and ρ_I are strictly positive leakage factors, while the σ-modification function $\sigma_\phi(\|\phi\|)$ is that in (19) but with the vector ϕ defined as

$$\phi = \begin{bmatrix} \phi_x & \phi_R & \phi_I \end{bmatrix}^T, \tag{29}$$

and the terms $\Lambda_{\min}$ and $\Lambda_{\max}$ used for 20a defined as

$$\Lambda_{\min} = \min\left\{\frac{\rho_X}{\alpha_X}, \frac{\rho_R}{\alpha_R}, \frac{\rho_I}{\alpha_I}\right\}, \quad \text{and} \quad \Lambda_{\max} = \max\left\{\frac{\rho_X}{\alpha_X}, \frac{\rho_R}{\alpha_R}, \frac{\rho_I}{\alpha_I}\right\}. \tag{30}$$

For the EMRAC-σ strategy, the function f_e for limiting the evolution of the integral of the tracking error is designed accordingly as

$$f_e = -\rho_I \sigma_I(|x_I|)x_I \tag{31}$$

with ρ_I being the corresponding strictly positive leakage factor and

$$\sigma_I(|x_I|) = \begin{cases} 0, & \text{if } |x_I| \leq \widehat{\mathcal{M}}_I, \\ \eta_I\left(\frac{|\phi_I|}{\widehat{\mathcal{M}}_I} - 1\right), & \text{if } \widehat{\mathcal{M}}_I < |x_I| \leq 2\widehat{\mathcal{M}}_I, \\ \eta_I, & \text{if } |x_I| > 2\widehat{\mathcal{M}}_I, \end{cases} \tag{32}$$

where η_I and $\widehat{\mathcal{M}}_I$ are strictly positive constants.

Figure 6 shows the block diagram of the EMRAC algorithm which can be used for representing both the EMRAC-PP and the EMRAC-σ strategies. In Figure 6, the adaptive mechanism block updates the feedforward and feedback gains based on the adaptive laws (23).

Remarks:

- Compared to the MRAC strategy, the EMRAC algorithm is augmented with the adaptive integral control action ($u_I(t) = K_I(t)x_I$) and the adaptive switching control term ($u_N(t) = K_N(t)\text{sgn}(y_e)$). The adaptive integral control action improves the tracking accuracy of the reference model by compensating for unmodeled biases in the plant dynamics, such as constant disturbances. Moreover, the adaptive switching control
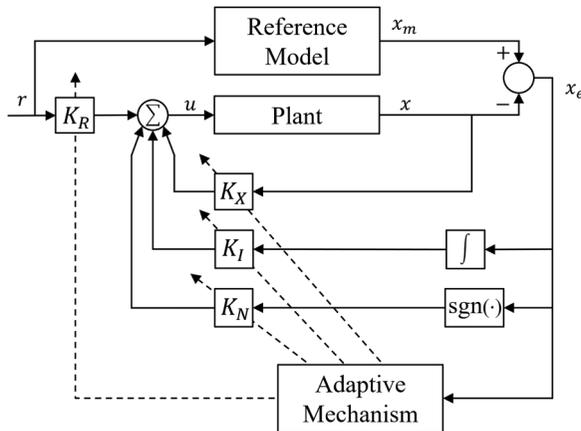
Figure 6. Schematic of the EMRAC strategy.

action boosts the robustness of the closed-loop tracking performance against rapidly varying bounded disturbances.

- The proportional-integral adaptive law used for the gains of the continuous control actions, i.e., K_X, K_R and K_I in (23), improves convergence of the adaptive gains.[4] Indeed, the inclusion of the proportional term in the adaptive mechanism allows for quick adjustments of the control gains when there are rapid changes in the tracking error.

- The EMRAC approach is more computationally demanding than traditional MRAC algorithms due to the additional adaptive integral and switching control actions. However, implementations of EMRAC solutions on different control units has been investigated (for example dSpace,[23,25] ROS,[31] and pixhawk[33]) which suggest that the two extra control terms, u_I and u_N, do not compromise the real-time implementability of EMRAC strategies. Moreover, in this paper it is shown that EMRAC solutions can be also implemented in cost-effective microcontrollers, such as Arduino boards.

- It has been shown that EMRAC solutions guarantee convergence of the tracking error to zero to square measurable ($\mathcal{L}_2$) as well as bounded ($\mathcal{L}_\infty$) matched disturbance and unmodeled dynamics. In the case of first order-systems, the asymptotic convergence to zero of the closed-loop tracking error x_e in the presence of bounded disturbances is sketched in the Appendix for the sake of completeness.

- The sign function included in EMRAC solutions can induce high-frequency oscillations in the control action which can increase wear of the actuation system. These unwanted oscillations are known in the literature of discontinuous systems as the chattering phenomenon[58] and it is a common side effect of control strategies including discontinuous control terms, for example relay control systems with small hysteresis and sliding mode control algorithms.[59] In order to avoid the chattering phenomenon, the discontinuous term is commonly smoothed during its implementation, as in Montanaro et al.,[31,32] by adopting

$$\mathrm{sgn}(y_e(t)) = \frac{y_e}{|y_e| + \epsilon} \tag{33}$$

with ϵ being a strictly positive constant. Small values of ϵ implies a better approximation of the sign function. Hence, the ϵ-parameter is chosen to be sufficiently large to smooth the control action but without compromising closed-loop tracking performance.

- The EMRAC algorithms guarantee that the closed-loop system is globally uniformly ultimately bounded for any bounded external disturbance and unmodeled dynamics (see also[58] for the definition of ultimate bounded systems). Practically, this means that the magnitude of the closed-loop state is bounded by a converging function and after a time interval it remains upper-bounded by a strictly positive constant also known as the ultimate bound. Specifically, by defining the closed-loop state as

$$\tilde{x}_e = \begin{bmatrix} x_e & \phi_e \end{bmatrix}^T, \tag{34}$$

with $\phi_e = \widehat{\phi} - \phi$ being the mismatch between the adaptive gain vector in (29) and the ideal gain vector $\widehat{\phi}$ defined as

$$\widehat{\phi} = \begin{bmatrix} \widehat{\phi}_X & \widehat{\phi}_R & 0 \end{bmatrix}^T. \tag{35}$$

it is possible to prove that

$$\left\| \widetilde{x}_e(t) \right\| \leq \gamma_b(t), \quad \text{for all } t \geq t_0, \tag{36}$$

with $\gamma_b(t)$ defined as

$$\gamma_b(t) = \begin{cases} \Psi\big(\left\| \widetilde{x}(t_0) \right\|, t - t_0\big), & \text{if } t_0 \leq t \leq t_0 + \mathcal{T}, \\ \kappa_{ub}, & \text{if } t > t_0 + \mathcal{T}, \end{cases} \tag{37}$$

where the time interval $\mathcal{T}$, after which the tracking error is bounded by the ultimate bound κ_{ub}, depends on the initial condition $\widetilde{x}(t_0)$; and $\Psi : \mathbb{R}^+ \times \mathbb{R}^+ \to \mathbb{R}^+$ is a $\mathcal{KL}$-class function (in other words a function such that for any value $\bar{z}$, the mapping $\Psi(r, \bar{z})$ is an increasing function of the variable r and $\Psi(0, \bar{z}) = 0$, while for any $\bar{r}$, the mapping $\Psi(\bar{r}, z)$ is a decreasing function of z converging to zero when $z \to +\infty$[58]).

Controller design

In this paper four adaptive controllers are designed and experimentally compared: *(i)* EMRAC-PP algorithm, *(ii)* EMRAC-σ algorithm, *(iii)* the classical MRAC strategy[3] and *(iv)* the classical MRAC method equipped with the σ-modification strategy which is denoted hereafter as MRAC-σ. In addition, two control solutions with fixed control gains are designed and used as additional benchmark solutions: *(i)* a PI controller and *(ii)* a pole-placement control strategy.[52]

The reference model selected for the adaptive solutions is a linear model of the form (7) with time constant $\tau = 2.5$ s and unitary static gain (i.e., $a_m = -1/\tau$ and $b_m = -a_m$). Considering that the plant time constant is about 3.45 s, this selection of the reference model reduces the settling time and the rise time of the closed-loop system by approximately 27.5% compared to the open-loop plant.

The adaptive weights for the EMRAC-PP and EMRAC-σ solutions are the same and have been selected as a trade-off between convergence time and reactivity of the control action. Moreover, the adaptive weights of the proportional part are set ten times smaller than the integral part (that is $\beta_i = 10^{-1}\alpha_i$ with $i \in \{X, R, I\}$). The bounds of the parameter-projection strategy for the EMRAC-PP algorithm, and the thresholds for the EMRAC-σ solution have been chosen such that conditions (14) and (15) are satisfied, respectively. Furthermore, the switching action of the EMRAC solutions are computed by using (33) to avoid chattering phenomena of the control action.

The adaptive weights and parameters of the MRAC and MRAC-σ algorithms are a subset of those of the EMRAC solutions, and their values have been selected identical to those used for the EMRAC algorithms such that MRAC strategies have the same adaptation rates as the EMRAC solutions.

The plant model used for tuning the pole-placement control gains and PI gains is system (1) where the parameters are those obtained through the procedure detailed in the section "Modelling and Experimental Validation". Specifically, the fixed control gains of the pole-placement strategy, i.e., $\phi_X^\star$ and $\phi_R^\star$ in (4), have been tuned as described in the "Motivation to MRAC and EMRAC Solution" section to have the position of the closed-loop pole equal to that of the reference model (i.e., $-1/\tau$) and unitary closed-loop static gains. Hence, the closed-loop system controlled via the pole-placement strategy should ideally behave as the reference model selected for the adaptive solutions, thus making the comparison of the closed-loop tracking performance among these control solutions possible.

Moreover, for a fair comparison, the reference signal provided to the PI algorithm is the state of the reference model (i.e., the solution of the LTI system (7)), i.e., the control action computed by the PI strategy is

$$u_{PI}(t) = K_p x_e(t) + K_I \int_{t_0}^{t} x_e(\hat{t}) d\hat{t} \qquad (38)$$

where x_e is computed as in (11), while K_p and K_I are the PI control gains. In so doing, the speed behavior that the PI controller will try to impose is that of the reference model. The PI control gains have been tuned based on the plant parameters estimated in the section "Modelling and Experimental Validation" by using the tool "pidtune" in MATLAB.

Experimental results

The closed-loop performance of the controllers designed in the previous section "Controller Design" are experimentally tested on the educational plant presented in the section "Experimental Setup" in different operating conditions. First a long manoeuvre composed of different reference speed profiles with a total duration of 10 minutes is considered to evaluate closed-loop performance and the corresponding evolution of the closed-loop signals. Then an experimental analysis is presented for different square waves, and the closed-loop tracking performance is quantitatively evaluated via a set of key performance indicators (KPIs) which are usually used in control engineering to compare control solutions.[52] Square wave reference inputs have been selected for the second analysis because they are challenging as they generate reference state profiles with steep speed variations which are more difficult to impose.

Long manoeuvre test

The reference input for the long manoeuvre is composed of 4 sub-manoeuvres denoted as M_i, $i = 1, \ldots, 4$, where *(i)* M1 is a square wave with an amplitude of 300 rad/s, a period of 60 s, and a duration of 180 s; *(ii)* M2 is a sequence of steps at 150, 300, 450 rad/s each lasting 40 s for a total of 120 s; *(iii)* M3 is a sinusoidal wave with an amplitude of 400 rad/s and a period of 20 s with a duration of 85 s; and *(iv)* M4 is a constant at 300 rad/s with a duration of 135 s.

Figure 7 shows the closed-loop performance in terms of speed, error and control action provided by each controller, the evolution of the adaptive gains are shown in Figure 8 for the MRAC solutions, while Figures 9 and 10 depict the dynamics of the adaptive control gains for the EMRAC-PP and EMRAC-σ strategies, respectively. Based on the experimental results, the following remarks can be drawn:

- Figures 7(a) and 7(d) show that the control strategies with fixed gains do not provide precise tracking of the reference model dynamics mainly because of residual plant parameter uncertainties and nonlinear phenomena discussed in the earlier section "Modelling and Experimental Validation". Specifically, the pole-placement algorithm does not provide good tracking during the transient nor during the constant steady-state operating conditions. For instance, during the sub-manoeuvre M4, the
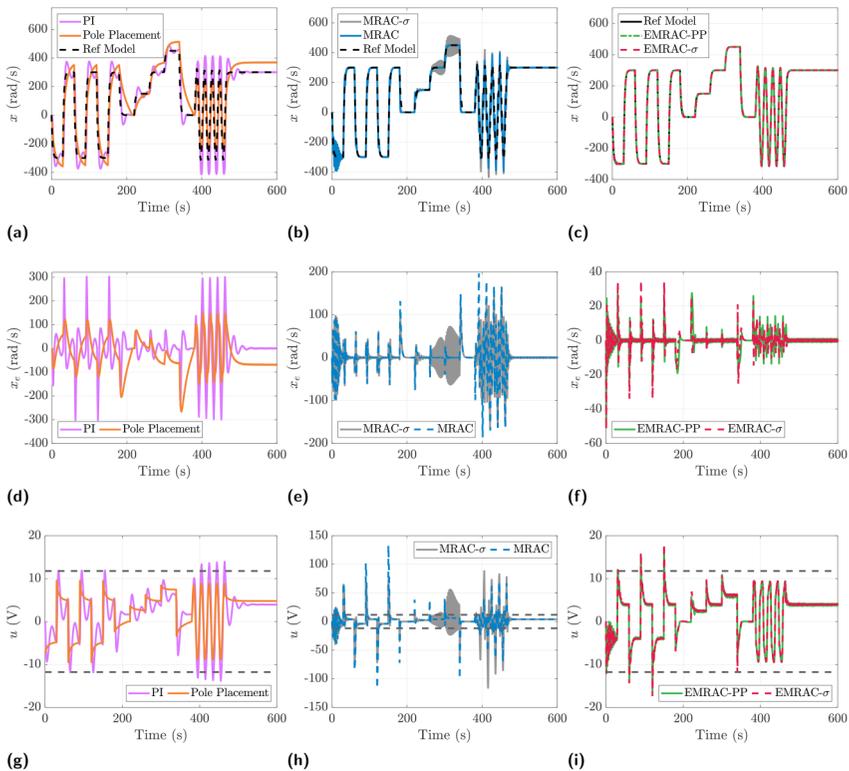
**Figure 7.** Experimental speed and reference speed (black line), tracking error and control action generated by the controller along with the saturation limits (dark gray dashed lines) for (a, d, g) pole-placement strategy (magenta line) and PI control (orange line); (b, e, h) MRAC (blue line) and MRAC-σ (gray line) algorithms; and (c, f, i) EMRAC-PP (green line) and EMRAC-σ (red line) strategies.
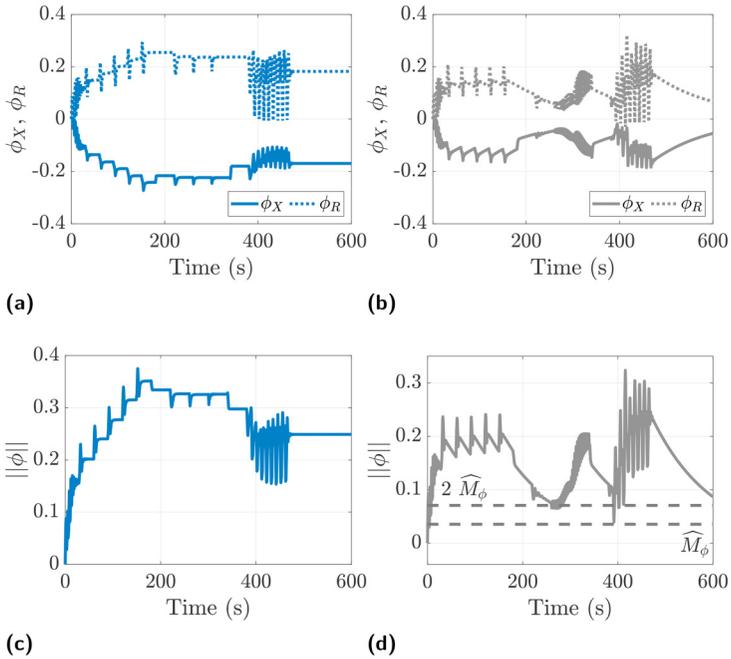
Figure 8. Experimental adaptive gains: ϕ_X (solid line), ϕ_R (dotted line) and magnitude of the adaptive gains $\|\phi\|$ for: (a, c) the MRAC algorithm; and (b, d) the MRAC-σ strategy.
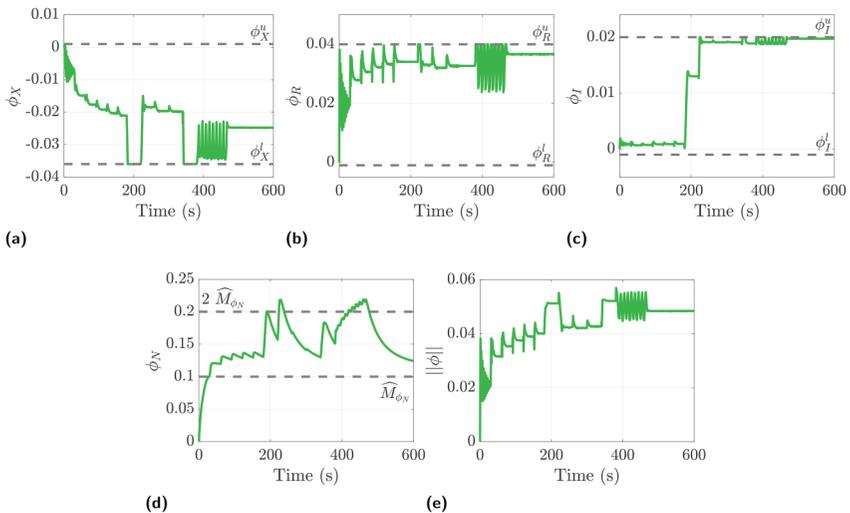
Figure 9. Experimental evolution of the EMRAC-PP adaptive gains: (a) ϕ_X, (b) ϕ_R, (c) ϕ_I, (d) ϕ_N and (e) $\|\phi\|$.
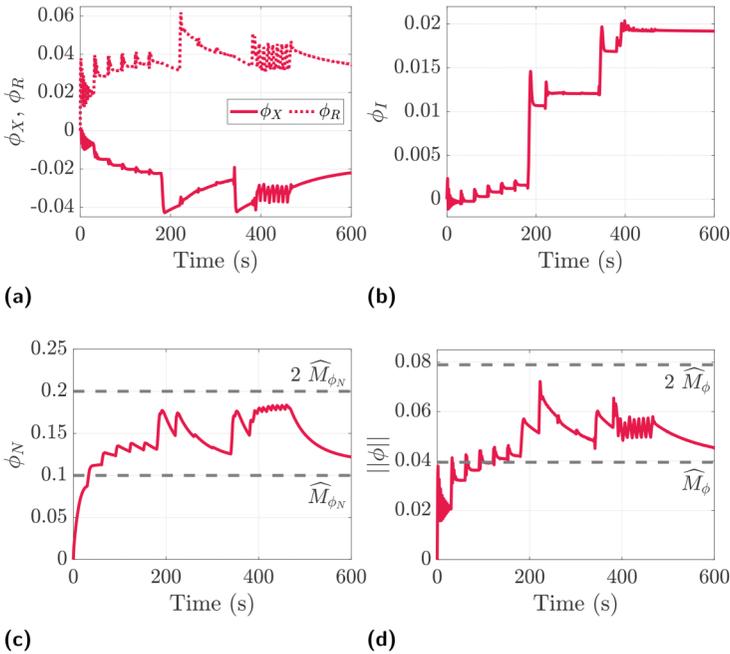
Figure 10. Experimental evolution of the EMRAC-σ gains: (a) ϕ_X (solid line) and ϕ_R (dotted line), (b) ϕ_I, (c) ϕ_N and (d) $\|\phi\|$.

constant reference is tracked with a steady-state error with a magnitude of about 68 rad/s (see also Figure 7(d)). This residual error is caused by the residual parameter mismatch between model (1) and the actual plant dynamics. Indeed, Figure 5(a) shows that for steady-state speeds around 300 rad/s there is significant mismatch between the model predictions and experimental data, and the steady-state armature voltage to keep the speed at 300 rad/s is experimentally around 3.9 V. However, with the tuning of the control gains $\phi_X^\star$ and $\phi_R^\star$ as presented in the section "Motivation to MRAC and EMRAC Solution" and based on the plant parameters estimated in the section "Modelling and Experimental Validation", the steady-state control action for M4 is about 5 V which, in agreement with Figure 5(a), gives a steady-state speed of about 368 rad/s as also observed in Figure 7(a).

- As expected the PI strategy provides zero steady-state tracking error for constant reference signals (for example M4). However, plant parameter mismatch and unmodeled dynamics induce overshoots in the step response observed in Figures 7(a) and 7(d) during the tracking of the M2 and M4 signals. Moreover, the closed-loop systems have a bandwidth that does not allow the tracking of the sinusoidal wave M3. As the amplitude of the steady-state sinusoidal speed is larger then the required one, it is deduced that the natural frequency of the experimental closed-loop system is around that of M3.

- The MRAC solutions (i.e., the classical MRAC and the MRAC-σ strategies) provide better tracking of the reference model dynamics compared to the strategies with fixed gains for most of the long manoeuvre as shown in Figures 7(b) and 7(e). However, for some time intervals (during the first step of the square wave for example) both the MRAC strategy and the MRAC-σ algorithm show high frequency oscillations around the reference state with large tracking errors. This unwanted closed-loop behavior is the result of the integral windup phenomenon caused by the saturation of the control action.[52,60,61] Qualitatively speaking, for some operating conditions (for example for rapid variation of the reference input, such as step variations) the MRAC and the MRAC-σ strategies are not able to rapidly compensate the mismatch between the reference model and plant speed. The presence of such large errors is then accumulated by the integrals defining the adaptive laws for computing ϕ_X and ϕ_R, and consequently the adaptive gains increase. Large adaptive gains and errors lead to the generation of control actions above the saturation limits, and thus saturations are activated. During saturation, the adaptive gains are still continuously updated, but these changes do not have an effect on the control action and the plant speed as the actual armature voltage is saturated either at the maximum or minimum value (meaning that there is a loss of control authority). The evolution of the adaptive gains, as seen in Figure 8, during saturation implies that when the system re-enters the linear operating region of the actuator, the actual control gains might be very different from those which would lead to the reduction of the tracking error, thus erroneous control actions can be generated which may increase the mismatch between the reference speed and the actual speed and the process described above repeats, thus leading to a "vicious" cycle of saturations, large errors and demand of large control actions.

  Figure 7(h) shows the control actions demanded by the MRAC and MRAC-σ strategies along with the maximum and minimum applicable armature voltage (dark gray dashed lines). Figure 7(h) confirms the onset of the windup phenomenon, as during the time intervals where the tracking error is large and rapidly oscillates, the control action required by these adaptive solutions is saturated.

  It is noted that the windup phenomenon is not limited to adaptive control solutions but can occur also in control strategies with fixed gains including terms based on the integral of the tracking error.[52,60] However, designing anti-windup strategies to include in adaptive laws to avoid unwanted closed-loop behaviours is more complex than for fixed-gain control algorithms, thus, they are beyond the scope of this paper, and the reader is referred to dedicated book chapters[62,63] and recent research articles, e.g., Turner et al.,[16] Turner and Richards,[17] for studies of MRAC algorithms in the presence of input saturations.

- The evolution of the adaptive gains for the MRAC and MRAC-σ algorithms along with their magnitude is depicted in Figure 8. Specifically, Figures 8(c) and 8(d) show that the magnitude of the adaptive gains of the MRAC strategy during the square wave M1 is greater than that provided by the MRAC-σ solution. The growth reduction of the adaptive gains magnitude given by the MRAC-σ strategy is due to the activation of the leakage factor of the σ-modification method in (19). Moreover, the leakage factors scheduled by the MRAC-σ are constants the

majority of the time and equal to $\rho_X \eta_\phi$ for ϕ_X and $\rho_R \eta_\phi$ for ϕ_R, respectively, as the magnitude of the gains, i.e., $\|\phi\|$, is above $2\widehat{\mathcal{M}}_\phi$ most of the time during the long manoeuvre.

- The EMRAC solutions, both the EMRAC-PP and EMRAC-σ strategies, provide precise tracking of the reference model dynamics as shown in Figure 7(c). Moreover, Figure 7(f) confirms that the magnitude of the residual speed tracking errors obtained by using the EMRAC strategies are much smaller than the control algorithms with fixed gains and the MRAC methods shown in Figures 7(d) and 7(e), respectively. The EMRAC solutions outperform the fixed gains control methods as they are robust to unmodeled plant dynamics and residual plant parameter mismatches which still remain after the identification method presented in the section "Modelling and Experimental Validation". Furthermore, the improvement in tracking performance with respect to the MRAC methods is achieved as the EMRAC strategies generate control actions that do not saturate during the long manoeuvre, except for a few spikes that exceed the magnitude V_m but do not jeopardize the tracking of the reference model dynamics. This is confirmed by Figure 7(i), which shows that the control actions demanded by the EMRAC solutions are saturated only four times during M1 and for very short time intervals, when compared to the length of the manoeuvre.
- The bounded evolution of the integral part of the adaptive gains for the EMRAC-PP and EMRAC-σ strategies are shown in Figures 9 and 10, respectively. The bounded dynamics of these gains along with the boundedness of the tracking error shown in Figures 7(f), imply *(i)* the boundedness of the magnitude of the adaptive vector ϕ in (29) (see also Figures 9(e) and 10(d)), *(ii)* the boundedness of the adaptive gains K_X, K_R, K_I and K_N in (23), and thus *(iii)* the boundedness of the control actions as confirmed in Figure 7(i). For the EMRAC-PP algorithm, Figures 9(a) to 9(c) show the effectiveness of the parameter-projection strategy (16) and (26) to limit the growth of the adaptive gains within their admissible range as well as their locking and unlocking during the long manoeuvre either at the maximum or minimum admissible values, e.g., the gain ϕ_X in Figure 9(a) is locked twice at its admissible lower bound ϕ_X^l. For the EMRAC-σ solution, the magnitude of the gain of discontinuous control action, i.e., ϕ_N, and the magnitude of the integral part of adaptive gains of the continuous control actions, i.e., $\|\phi\|$, remain ultimately confined within the σ-modification limits, namely $\widehat{\mathcal{M}}_{\phi_N} \leq \phi_N(t) \leq 2\widehat{\mathcal{M}}_{\phi_N}$ and $\widehat{\mathcal{M}}_\phi \leq \|\phi(t)\| \leq 2\widehat{\mathcal{M}}_\phi$, as shown in Figures 10(c) and 10(d), respectively, because the closed-loop tracking errors provided by this control strategy (see also Figure 7(f)) are not large enough to make these adaptive gains exceed the upper limit of the corresponding σ-modification strategies.

Square-wave analysis

The analysis in the subsection "Long manoeuvre test" has shown that MRAC strategies can induce a high oscillatory behavior of the speed with regards to step variations of the reference signal because of control action saturations, while EMRAC solutions ensure a more precise tracking of the reference model and avoid speed oscillations and control saturations for these more challenging reference inputs. In this context, this section

aims to further analyse the closed-loop behavior of the adaptive control strategies under a sudden variation of the reference signal. Therefore, the adaptive control systems have been further tested for ten square waves, S_i, $i = 1, \ldots, 10$, for which amplitude, bias and period are detailed in Table 1.

To quantitatively assess the closed-loop behavior a set of KPIs is used. Specifically, the closed-loop tracking performance is evaluated through the following indices

$$\text{RMSE}(|x_e|) = \sqrt{\frac{1}{T_f - T_i} \int_{T_i}^{T_f} (x_e(t))^2 dt}, \tag{39a}$$

$$\mu(|x_e|) = \frac{1}{T_f - T_i} \int_{T_i}^{T_f} |x_e(t)| dt, \tag{39b}$$

$$\bar{\sigma}(|x_e|) = \sqrt{\mu\big([|x_e(t)| - \mu(|x_e|)]^2\big)}, \tag{39c}$$

$$\kappa(|x_e|) = \max_{T_i \leq t \leq T_f} |x_e(t)|, \tag{39d}$$

where $\text{RMSE}(|x_e|)$ is the root mean square error, $\mu(|x_e|)$ and $\bar{\sigma}(|x_e|)$ are the average error and the corresponding standard deviation, respectively, and $\kappa(|x_e|)$ is the maximum error, while T_i and T_f are the initial and final time of the time interval used for computing the KPIs, respectively.

The duration for each square wave in Table 1 is 5 cycles and the performance indexes have been computed after the transient has been extinguished (therefore over the last four periods).

The control action performance is evaluated by computing the integral of the absolute value of the control action normalized with time (IACA) of the actuated control action (i.e., the actual armature voltage which is limited between $\pm V_m$). The IACA is a

Table 1. Square waves used for the experimental analysis.

| Signal	Amplitude (rad/s)	Bias (rad/s)	Time Period (s)
S1	150	−250	40
S2			60
S3			80
S4	200	250	20
S5		−220	40
S6			60
S7			80
S8	250	300	40
S9			60
S10			80

measure of the control effort and is defined as

$$\text{IACA} = \frac{1}{T_f - T_i} \int_{T_i}^{T_f} |u_a(t)| dt, \tag{40}$$

where $u_a = \text{sgn}(u) \min \{|u|, V_m\}$ is the actual control action limited within the range $[-V_m, V_m]$.

Figures 11 and 12 show the KPIs of the adaptive control solutions for the closed-loop tracking performance and control action, respectively.

Based on Figures 11 and 12, the following remarks are made.

- The EMRAC solutions provide an RMSE smaller than the MRAC strategies. Figure 11(a) shows that the EMRAC-PP solutions reduces the RMSE index *(i)* by 2.75 to 7.4 times compared to the MRAC algorithm, and *(ii)* by 3 to 7.8 times compared to the MRAC-σ algorithm. The reduction factor improves when the MRAC approaches are compared to the EMRAC-σ solution. Indeed, the reduction factors
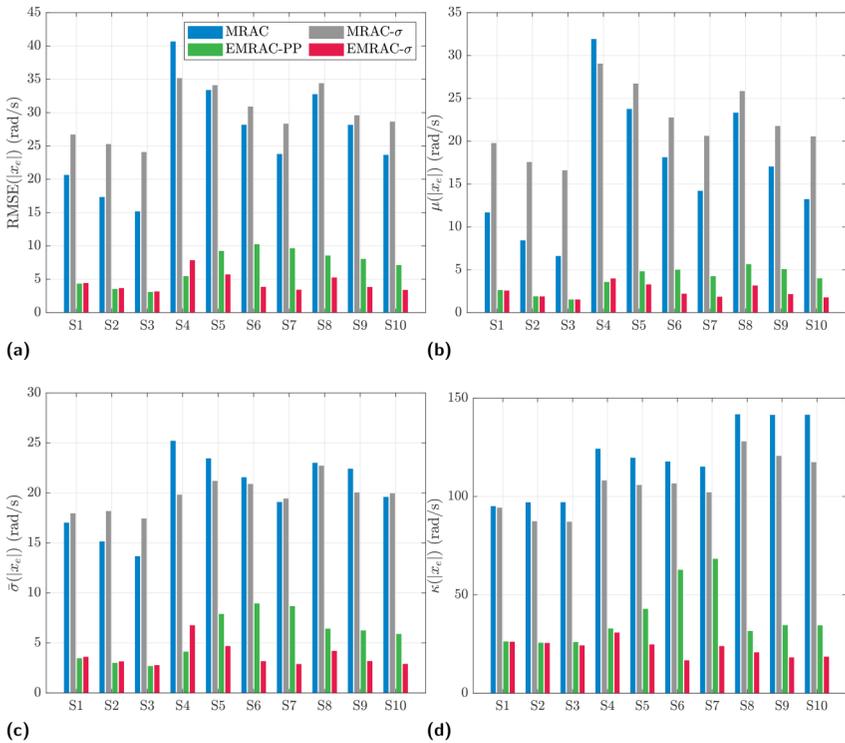
Figure 11. Experimental closed-loop tracking KPIs for the MRAC (blue), MRAC-σ (gray), EMRAC-PP (green) and EMRAC-σ (red) strategies: (a) RMSE($|x_e|$), (b) average error $\mu(|x_e|)$, (c) standard deviation $\bar{\sigma}(|x_e|)$ and (d) maximum error $\kappa(|x_e|)$.

for the RMSE provided by the EMRAC-σ strategy are in the range [4.7, 7.3] and [4.5, 8.4] times when compared to the MRAC and MRAC-σ solutions, respectively. Moreover, the RMSE for the EMRAC-PP is always below 10 rad/s, while, except for S4, the EMRAC-σ provides an RMSE always either at or below about 5 rad/s.

- Figures 11(b) and 11(c) show that for all the square waves the EMRAC strategies not only substantially reduce the average error $\mu(|x_e|)$ compared to the MRAC algorithms, but also the spread of the speed error around the corresponding average decreases as the standard deviation is reduced. For instance, the EMRAC-PP solution reduces the $\mu(|x_e|)$-index by at least 3.3 and 4.3 times with respect to the MRAC and MRAC-σ methods, respectively. In the case of the EMRAC-PP, the spread of the error around the average is reduced by a factor that ranges from 3 to 6.1 when compared to the MRAC algorithm, and from 2.24 to 6.5 when compared to the MRAC-σ strategy. The standard deviation is further reduced when the EMRAC-σ algorithm is used. Specifically, for the EMRAC-σ, the reduction factor varies in the range [3.72, 7] compared to the MRAC, and [3, 7] compared to the MRAC-σ algorithms, respectively.
- Maximum errors are also reduced when comparing the EMRAC solutions to the MRAC strategies as confirmed in Figure 11(d). Specifically, the EMRAC-PP reduces the maximum error with a factor that are in the range [1.7, 4.5] times when compared to the MRAC algorithm, and [1.5, 4] when compared to the MRAC-σ strategy, respectively. The attenuation factor provided by the EMRAC-σ for the $\kappa(|x_e|)$-index is larger and varies in the range [3.6, 7.8] times when compared to those provided by the MRAC algorithm, and [3.4, 6.6] times when compared to those obtained by the MRAC-σ method, respectively.
- The analysis of the IACA-index in Figure 12 shows that the control effort required by the EMRAC solutions is comparable to that required by the MRAC strategy and either similar or smaller when compared to that demanded by the MRAC-σ algorithm. Hence, the EMRAC strategies can improve closed-loop tracking performance
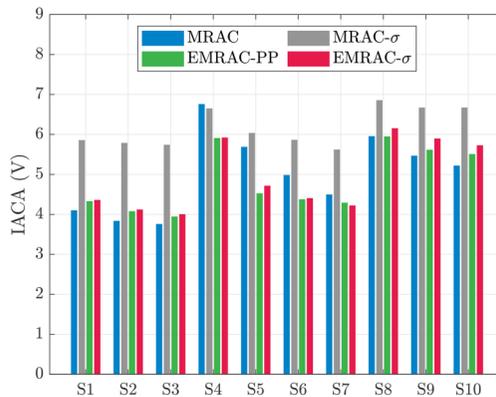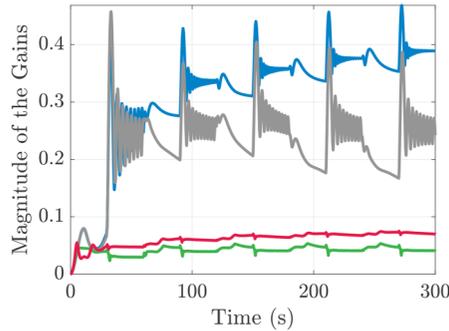

Figure 12. Experimental IACA for the MRAC (blue), MRAC-σ (gray), EMRAC-PP (green) and EMRAC-σ (red) strategies.

without an increase in the control effort. As shown in subsection "Long manoeuvre test", the tracking performance improvements are achieved as the EMRAC strategies better adjust the control action to quickly track the reference model state despite unmodeled dynamics and disturbances. On the other hand, the larger tracking errors obtained via the MRAC methods generate larger adaptive gains and control actions, which ultimately are saturated. These control action saturations then result in larger tracking error oscillations, preventing the actuator from operating in the linear region. The loss of closed-loop tracking performance due to activation of saturations is also confirmed by a further analysis of the control actions which shows that, for the reference signals in Table 1, the control action generated by the MRAC and MRAC-σ algorithms is saturated on average about 20.5% and 36% of the duration of each square wave manoeuvre, respectively. The average percentage duration of the control action saturation for the square waves in Table 1 reduces to 2.3% and 1.44% for the EMRAC-PP and EMRAC-σ solutions, respectively. Figure 13 shows the tracking performance and the control action provided by each adaptive controller for the S7 reference signal. In this case, the percentage time the control signal in Figure 13(c) is saturated is *(i)* 15.84% for the MRAC, *(ii)* 30.78% for the MRAC-σ, *(iii)* 2.26% for the EMRAC-PP, and *(iv)* 0.39% for the EMRAC-σ. Consequently, it is expected that the EMRAC-σ strategy provides the best closed-loop tracking performance while the worst is given by the MRAC-σ algorithm. This is confirmed by Figures 13(a) and 13(b) and Figure 11. Figure 13 also shows that the tracking performance given by the MRAC and MRAC-σ significantly improves when their corresponding demanded control actions are within the admissible bounds for the motor armature voltage.

- An example of the evolution of the magnitude of the adaptive gains of continuous control actions is depicted in Figure 14 for the reference signal S9 in Table 1. This figure shows a drift in the gains of the MRAC algorithm due to unmodeled dynamics, which can also ultimately induce closed-loop instability. On the contrary, the parameter-projection and σ-modification strategies, included in the MRAC-σ,
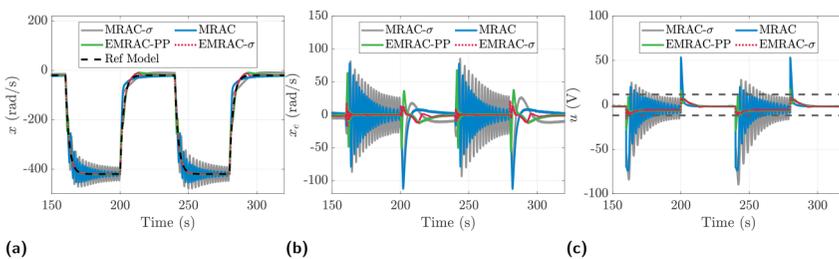
Figure 13. Experimental results for the S7 reference signal for the MRAC (blue line), MRAC-σ (gray line), EMRAC-PP (green line) and EMRAC-σ (red line) strategies: (a) speed and reference speed (dashed black line), (b) tracking error, and (c) control action generated by the controller along with the saturation limits (dark gray dashed lines).

Figure 14. Experimental magnitude of the control gain vector ϕ for the S9 signal in table 1 for the MRAC (blue), MRAC-σ (gray), EMRAC-PP (green) and EMRAC-σ (red) strategies.

EMRAC-PP and EMRAC-σ, are effective to keep the evolution of the adaptive gain vector ϕ bounded.

Conclusion

This paper has revised the EMRAC strategy, its design, and aspects of the closed-loop dynamics through the use of a first-order experimental educational plant, thus facilitating the learning of the EMRAC strategy for students who may not be familiar with adaptive control and MRAC solutions. The EMRAC tracking performance has been experimentally compared with that of linear controllers commonly taught in undergraduate control engineering modules and classical MRAC solutions, and it has been shown that EMRAC algorithms outperform these baseline control strategies. Specifically, experimental data has shown that EMRAC methods can be used as an effective approach to impose the ideal closed-loop dynamics provided by a pole-placement strategy, despite plant parameter mismatches which reduce the tracking capability of the classical linear solution with fixed gains. Moreover, compared to the baseline MRAC techniques, the EMRAC solutions are better able to keep the plant within the linear operating region, thus avoiding control input saturation, which, together with the additional adaptive EMRAC terms, improves the tracking of the reference model dynamics. Future research work on the EMRAC algorithm will include extensions to further increase its robustness to unmodelled nonlinear dynamics, e.g. by means of neural network / artificial intelligence methods.

Ethical Considerations

Ethical Approval was not required.

Consent to Participate

Not applicable.

Consent for Publication

Not applicable.

Funding

Declaration of Conflicting Interest

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

ORCID iDs

Umberto Montanaro https://orcid.org/0000-0003-0620-1906
George Whitehead https://orcid.org/0009-0007-5816-5763
Aldo Sorniotti https://orcid.org/0000-0002-4848-058X

Data Availability Statement

The datasets generated during and/or analyzed during the current study are available in the Dataset for "Learning enhanced model reference adaptive control algorithms via a cost-effective educational plant" repository, https://doi.org/10.5281/zenodo.15324454

References

1. Tao G. *Adaptive control design and analysis*. Hoboken, NJ: John Wiley & Sons, Inc., 2003.
2. Landau YD. *Adaptive control: The model reference approach*. New York: CRC Press, 1979.
3. Ioannou P and Fidan B. *Adaptive control tutorial*. Philadelphia, PA: Siam, 2007.
4. Landau ID, Lozano R, M'Saad M, et al. *Adaptive control: Algorithms, analysis and applications*. London: Springer, 2011.
5. Nguyen NT. *Model-Reference adaptive control: A primer*. Cham: Springer, 2018.
6. Jingzhuo S and Huang W. Model reference adaptive iterative learning speed control for ultrasonic motor. *IEEE Access* 2020; 8: 181815.
7. Torki W, Grouz F and Sbita L. A sliding mode model reference adaptive control of pmsg wind turbine. In: *International conference on green energy conversion systems*. pp.1–6.
8. Baidya D and Roy R. Speed control of dc motor using fuzzy-based intelligent model reference adaptive control scheme. In: *Lecture notes in electrical engineering*, Vol. 462, 2018, pp.729–735.
9. Abouheaf MW, Spinello D, et al. A data-driven model-reference adaptive control approach based on reinforcement learning. In: *2021 IEEE International symposium on robotic and sensors environments (ROSE)*, 2021.
10. Wei Y, Sun Z, Hu Y, et al. On fractional order composite model reference adaptive control. *Int J Syst Sci* 2016; 47: 2521–2531.
11. Xie J, Li S, Yan H, et al. Model reference adaptive control for switched linear systems using switched multiple models control strategy. *J Franklin Inst* 2019; 356: 2645–2667.
12. di Bernardo M, Montanaro U and Santini S. Novel hybrid mrac-lq control schemes: synthesis, analysis and applications. *Int J Control* 2008; 81: 940–961.

13. Xie J and Zhao J. H∞ model reference adaptive control for switched systems based on the switched closed-loop reference model. *Nonl Anal: Hybrid Syst* 2018; 27: 92–106.
14. di Bernardo M, Montanaro U, Ortega R, et al. Extended hybrid model reference adaptive control of piecewise affine systems. *Nonl Anal: Hybrid Syst* 2016; 21: 11–21.
15. di Bernardo M, Hoyos Velasco CI, Montanaro U, et al. Experimental implementation and validation of a novel minimal control synthesis adaptive controller for continuous bimodal piecewise affine systems. *Control Eng Pract* 2012; 20: 269–281.
16. Turner MC, Sofrony J and Prempain E. Anti-windup for model-reference adaptive control schemes with rate-limits. *Syst Control Lett* 2020; 137: 104630.
17. Turner JSMC and Richards CM. Anti-windup compensation for model reference adaptive control schemes. *Syst Control Lett* 2024; 34: 10176–10193.
18. Baldi S and Frasca P. Adaptive synchronization of unknown heterogeneous agents: an adaptive virtual model reference approach. *J Franklin Inst* 2019; 356: 935–955.
19. Yuan C, Zeng W and Dai SL. Distributed model reference adaptive containment control of heterogeneous uncertain multi-agent systems. *ISA Trans* 2019; 86: 73–86.
20. Zhang D and Wei B. A review on model reference adaptive control of robotic manipulators. *Annu Rev Control* 2017; 43: 188–198.
21. Kumar R, Das S, Syam P, et al. Review on model reference adaptive system for sensorless vector control of induction motor drive. *IET Elect Power Appl* 2015; 9: 496–511.
22. Narendra KS. *Applications of adaptive control*. New York: Academic Press, 2012.
23. di Bernardo M, di Gaeta A, Montanaro U, et al. Synthesis and experimental validation of the novel LQ-NEMCSI adaptive strategy on an electronic throttle valve. *IEEE Trans Control Syst Technol* 2010; 18: 1325–1337.
24. Montanaro U and Olm JM. Integral mrac with minimal controller synthesis and bounded adaptive gains: The continuous-time case. *J Franklin Inst* 2016; 353: 5040–5067.
25. Montanaro U, di Gaeta A and Giglio V. Robust discrete-time mrac with minimal controller synthesis of an electronic throttle body. *IEEE ASME Trans Mechatron* 2014; 19: 524–537.
26. Montanaro U, di Gaeta A and Giglio V. An MRAC approach for tracking and ripple attenuation of the common rail pressure for GDI engines. *IFAC Proc Vol* 2011; 44: 4173–4180.
27. Buonomano A, Montanaro U, Palombo A, et al. Dynamic building energy performance analysis: a new adaptive control strategy for stringent thermohygrometric indoor air requirements. *Appl Energy* 2016; 163: 361–386.
28. Buonomano A, Montanaro U, Palombo A, et al. Temperature and humidity adaptive control in multi-enclosed thermal zones under unexpected external disturbances. *Energy Build* 2017; 135: 263–285.
29. Dixit S, Montanaro U, Dianati M, et al. Integral mrac with bounded switching gain for vehicle lateral tracking. *IEEE Trans Control Syst Technol* 2020; 29: 1936–1951.
30. Montanaro U, Wroblewski M, Dixit S, et al. Linearising longitudinal vehicle dynamics through adaptive control techniques for platooning applications. *Int J Powert* 2022; 10: 312–335.
31. Montanaro U, Martini S, Hao Z, et al. Multi-input enhanced model reference adaptive control strategies and their application to space robotic manipulators. *Int J Rob Nonl Control* 2023; 33: 5246–5272.
32. Montanaro U, Chen C, Rizzo A, et al. Output-based enhanced closed-loop model reference adaptive control and its application to direct yawmoment control. *Int J Rob Nonl Control* 2024; 34: 9277–10052.
33. Martini S, Mennea SM, Mihalkov M, et al. Design and hil testing of enhanced mrac algorithms to improve tracking performance of lq-strategies for quadrotor uavs. In: *IEEE 20th International conference on automation science and engineering*.

34. University of Surrey, School of Mechanical Engineering Sciences. An application of the enhanced model reference adaptive control strategy for lateral vehicle stability via direct yaw moment control, 2019.
35. University of Surrey, School of Mechanical Engineering Sciences. Design and implementation of an adaptive hierarchical platoon control system, 2020.
36. University of Surrey, School of Mechanical Engineering Sciences. Design, implementation and performance analysis of enhanced model reference adaptive control (emrac) for a 3-dof anthropomorphic robotic manipulator, 2021.
37. University of Surrey, School of Mechanical Engineering Sciences. Advanced control systems for autonomous vehicle control, 2024.
38. University of Surrey, School of Mechanical Engineering Sciences. Enhanced model reference adaptive control algorithms applied to discontinuous systems, 2024.
39. Suarez A, García-Costa D, Perez J, et al. Hands-on learning: Assessing the impact of a mobile robot platform in engineering learning environments. *Sustainability* 2023; 15: 13717.
40. Visioli A and Stapleton L. Control and automation education and un sdgs: alignment of tc 9.4 - current status and future directions. *IFAC-PapersOnLine* 2024; 58: 171–175.
41. Heradio R, Chacon J, Vargas H, et al. Open-source hardware in education: A systematic mapping study. *IEEE Access* 2018; 6: 72094–72103.
42. Li JH. Control system laboratory with arduino. In: *2018 International Symposium on Computer, Consumer and Control (IS3C)*, 2018, pp.181–184.
43. Takács G, Konkoly T and Gulan M. Optoshield: a low-cost tool for control and mechatronics education. In: *2019 12th Asian control conference (ASCC)*, 2019, pp.1001–1006.
44. Barbosa RS. Educational platform for modeling and control. In: *2020 XIV Technologies applied to electronics teaching conference (TAEE)*, 2020.
45. Chuckpaiwong I and Boekfah A. Low-cost educational feedback control system: Helicopter tail rotor for yaw control. In: *2020 IEEE 7th International conference on industrial engineering and applications (ICIEA)*, 2020, pp. 266–270.
46. Vargová A, Boldock'u J, Gulan M, et al. Pressureshield: an open-source air pressure pocket lab for control engineering education. In: *2023 24th International conference on process control (PC)*, 2023, pp.96–101.
47. Arduino. Arduino engineering kit rev 2. https://store.arduino.cc/products/arduino-engineering-kit-rev2.
48. Arduino. Self-Balancing Motorcycle, 6.2 Programming the Motorcycle Components - Rotary Encoder. https://engineeringkit.arduino.cc/aekr2/module/engineering/project/06-self-balancing-motorcycle.
49. Onwubolu G. *Mechatronics principles and applications*. Oxford: Elsevier, 2005.
50. Isermann R and M'unchhof M. *Identification of dynamic systems: An introduction with applications*. Berlin, Heidelberg: Springer, 2010.
51. Fidlin A. *Nonlinear oscillations in mechanical engineering*. Berlin, Heidelberg: Springer, 2005.
52. Franklin GF, Powell JD and Emami-Naeini A. *Feedback control of dynamic systems*. 7th ed. Boston: Pearson, 2015.
53. Astrom KJ and Hagglund T. *PID controllers: Theory, design and tuning*. Research Triangle Park, NC: ISA, 1995.
54. Tarbouriech S, Garcia G, da Silva JMG, et al. *Stability and stabilization of linear systems with saturating actuators*. New York: Springer, 2011.
55. Rohrs C, Valavani L, Athans M, et al. Robustness of continuous-time adaptive control algorithms in the presence of unmodeled dynamics. *IEEE Trans Automat Contr* 1985; 30: 881–889.

56. Dey R, Jain SK and Padhy PK. Rohrs benchmark problem revisited by adaptive control with closed loop reference model. In: *IEEE international conference on information, communication, instrumentation and control*, 2017.

57. Montanaro U and Olm JM. Discrete-time integral mrac with minimal controller synthesis and parameter projection. *J Franklin Inst* 2015; 352: 5415–5436.

58. Khalil HK. *Nonlinear systems*. 2 ed NJ: Prentice hall Upper Saddle River, 2002.

59. Oliveira TR, Fridman L and Hsu L. *Sliding-mode control and variable-structure systems: The state of the art*. Cham: Springer, 2023.

60. Hippe P. *Windup in control: Its effects and their prevention*. London: Springer, 2006.

61. Tao G and Lewis FF. *Adaptive control of nonsmooth dynamic systems*. London: Springer, 2001.

62. Zhang C. Adaptive control with input saturation constraints. In: *Adaptive control of nonsmooth dynamic systems*, 2008, pp.361–381. Springer.

63. Zhou J and Wen C. Adaptive control of systems with input saturation. In: *Adaptive backstepping control of uncertain systems*, 2008, pp.189–197. Springer.

64. Mu XW, Ding ZS and Cheng GF. Uniformly ultimate boundedness for discontinuous systems with time-delay. *Appl Math Mech (English Edition)* 2011; 32: 1187–1196.

65. Shevitz D and Paden B. Lyapunov stability theory of nonsmooth systems. *IEEE Trans Automat Contr* 1994; 39: 1910–1914.

66. Nakakuki T, Shen T and Tamura K. Adaptive control design for a class of nonsmooth nonlinear systems with matched and linearly parameterized uncertainty. *Int J Rob Nonl Contr* 2009; 19: 243–255.

Appendix: EMRAC: Proof of the tracking error convergence

This section is dedicated to sketch the proof of the convergence to zero of the closed-loop tracking error x_e when EMRAC algorithms are used despite bounded disturbances and unmodeled dynamics $\delta(x, t)$ acting on the plant dynamics (3). The proof is sketched for first order systems for the sake of simplicity and consists of the following steps: *(i)* derivation of the closed-loop tracking error system; *(ii)* use of Lyapunov-based theories to show the boundedness of the closed dynamics and all closed-loop signals; and *(iii)* upper bound the derivative of the Lyapunov function along the closed-loop dynamics and the use of the Barbalat's lemma[3] to guarantee convergence of the tracking error. For simplicity, this section focuses on points *(i)* and *(iii)* while point *(ii)* is not presented in detail as it requires advanced knowledge of system dynamics including Lyapunov theory for non-smooth or discontinuous systems. The reader is referred to Montanaro et al.[31,32] for detailed proofs and closed-loop analysis for general and more complex systems.

Closed-loop dynamics

By using the EMRAC action (21) in the plant dynamics (3), the dynamics of the closed-loop system state $\widetilde{x}_e = [x_e \ \phi_e]^T$ are

$$\dot{x}_e = a_m x_e + \frac{b_m}{\widehat{\phi}_R}\left(\phi_e^T w - \phi_N \mathrm{sgn}(y_e) - y_e w^T \Gamma_\beta w\right) - \delta, \qquad (41a)$$

$$\dot{\phi}_e = -y_e \Gamma_\alpha w - f_\phi, \tag{41b}$$

where Γ_α and Γ_β are diagonal matrices belonging to $\mathbb{R}^{3\times 3}$ defined as

$$\Gamma_\alpha = \text{diag}(\alpha_X, \alpha_R, \alpha_I) : \text{and} \ \ \Gamma_\beta = \text{diag}(\beta_X, \beta_R, \beta_I), \tag{42}$$

and

$$w = \begin{bmatrix} x r x_I \end{bmatrix}^T \in \mathbb{R}^3, \tag{43a}$$

$$f_\phi = \begin{bmatrix} f_X f_R f_I \end{bmatrix}^T \in \mathbb{R}^3, \tag{43b}$$

where the function f_X, f_R and f_I are defined either as in (16) and (26) or as in (28) dependent on whether the parameter-projection strategy or the σ-modification method is used to complete the EMRAC action to avoid drifting of the adaptive control gains.

Boundedness of the closed-loop signals

The boundedness of closed-loop signals can be shown by using Lyapunov theory for proving the ultimate boundedness of smooth and discontinuous systems. Specifically, as show in Montanaro et al.,[31] Lyapunov theory for smooth systems presented in Khalil[58] can be used to prove the boundedness and the ultimate boundedness of the the gain ϕ_N by selection as Lyapunov function $V_N(\phi_N) = \phi_N^2/2$. However, the theory presented in Khalil[58] cannot be used to prove the ultimate boundedness of the dynamics (41) as the corresponding vector field is not Lipschitz. Hence, extensions of the Lyapunov theory to Filippov systems (i.e, to non-smooth systems) presented in Montanaro et al.,[31] Mu et al.[64] are used for proving the boundedness and the ultimate boundedness of $\widetilde{x}_e$, where the corresponding candidate Lyapunov function for first-order systems is

$$V(\widetilde{x}_e) = V(x_e, \phi_e) = x_e^2 + \frac{1}{\widehat{\phi}_R} \phi_e^T \Gamma_\alpha^{-1} \phi_e. \tag{44}$$

The reader is referred to Montanaro et al.[31] for the complete proof of the boundedness of ϕ_N and $\widetilde{x}_e$ for general multi-states, multi-inputs case.

Based on the boundedness of $\widetilde{x}_e$, the boundedness of the renaming closed-loop signals can be assessed as follows. The boundedness of $\widetilde{x}_e$ implies the boundedness of x_e, y_e and ϕ_e. As x_m is bounded by assumption, the boundedness of x_e also implies the boundedness of x and consequently the boundedness of x_I can be proved as in Montanaro et al.[31] The boundedness of ϕ_e implies the boundedness of ϕ. The boundedness of ϕ, x_e, x_I and r leads to the boundedness of K_X, K_R, and K_I which then implies that also u and $\dot{x}$ are bounded. Consequently, the boundedness of all closed-loop signals remains proven.

Tracking error convergence

It is possible to prove, by using the concept of generalized derivative,[65] that the derivative of the Lyapunov function (44) along with the solutions of (41) can be upper-bounded

almost everywhere (a. e.) as

$$\frac{dV}{dt} \leq -2|a_m|x_e^2 - \frac{2}{\widehat{\phi}_R} y_e^2 w^T \Gamma_\beta w - \frac{2}{\widehat{\phi}_R} \phi_N |y_e| - 2y_e \widehat{\delta}$$

$$\leq -2|a_m|x_e^2 - \frac{2}{\widehat{\phi}_R} \phi_N |y_e| - 2y_e \widehat{\delta}, \tag{45}$$

$$\leq -2|a_m|x_e^2 - 2|y_e| \left[\frac{\phi_N}{\widehat{\phi}_R} - \widehat{\mathcal{G}} \right] \quad \text{(a.e.)},$$

where $\widehat{\delta} = \delta/b_m$ and $\widehat{\mathcal{G}} = \mathcal{G}/|b_m|$.

As α_N has the same sign of $\widehat{\phi}_R$, then $\phi_N/\widehat{\phi}_R \geq 0$ for all time instants. Moreover, by assuming $\widehat{\mathcal{M}}_N > \mathcal{G}/|\widehat{\phi}_R|$, the adaptive law (23d) implies that there exists a time instant $t_0^\star > t_0$ for which $\phi_N(t)/\widehat{\phi}_R - \widehat{\mathcal{G}} > 0$ for all $t \geq t_0^\star$. Hence, the derivative (45) can be further upper-bounded as

$$\frac{dV}{dt} \leq -2|a_m|x_e^2, \quad \text{a.e.,} \quad \text{forall } t \geq t_0^\star. \tag{46}$$

From inequality (46), the following integral inequality holds for any $t \geq t_0^\star$,

$$\int_{t_0^\star}^t x_e^2(\tau)d\tau \leq \frac{1}{2|a_m|}\left[V(\widetilde{x}_e(t_0^\star)) - V(\widetilde{x}_e(t))\right] \leq \frac{V(\widetilde{x}_e(t_0^\star))}{2|a_m|}, \tag{47}$$

for any trajectory of the closed-loop system.

As (47) holds for $t \to +\infty$, then $x_e \in \mathcal{L}_2$. Moreover, the boundedness of all the closed-loop discussed in Appendix signals implies $x_e \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and $\dot{x}_e \in \mathcal{L}_\infty$, thus

$$\frac{d}{dt}(x_e^2) = 2x_e\dot{x}_e \in \mathcal{L}_\infty. \tag{48}$$

Hence, Barbalat's Lemma[1,14,66] can be applied, and thus $x_e^2(t) \to 0$, which implies $x_e(t) \to 0$ when $t \to +\infty$.