IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 12, 2021, accepted October 4, 2021, date of publication October 8, 2021, date of current version October 15, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3118658

Novel Asynchronous Activation of the Bio-Inspired Adaptive Tuning in the Speed Controller: Study Case in DC Motors

MIGUEL GABRIEL VILLARREAL-CERVANTES[1], (Member, IEEE), ALEJANDRO RODRÍGUEZ-MOLINA[2], (Member, IEEE), AND OMAR SERRANO-PÉREZ[1]

[1]Centro de Innovación y Desarrollo Tecnológico en Cómputo (CIDETEC), Instituto Politécnico Nacional, Mexico City 07700, Mexico
[2]Research and Postgraduate Division, Tecnológico Nacional de México/IT de Tlalnepantla, Estado de México 54070, Mexico

Corresponding authors: Miguel Gabriel Villarreal-Cervantes (mvillarrealc@ipn.mx) and Alejandro Rodríguez-Molina (alejandro.rm@tlalnepantla.tecnm.mx)

This work was supported in part by the Secretaría de Investigación y Posgrado (SIP) under Grant 20200150 and Grant 20210374 and the Comisión de Operación y Fomento de Actividades Académicas of the Instituto Politécnico Nacional, in part by the Dirección de Posgrado through the Investigación e Innovación del Tecnológico Nacional de México under Grant 11013.21-P, and in part by the Mexican Consejo Nacional de Ciencia y Tecnología (CONACyT).

ABSTRACT The fusion of bio-inspired algorithms into online controller tuning (adaptive controller tuning) is one of the main topics in Intelligent Control. One crucial issue is to reduce the times that the tuning process performs over time. In this work, a novel Asynchronous Adaptive Controller Tuning (AACT) approach is proposed to reduce the number of tuning process activations, and hence, this promotes resource savings in the overall computational cost for the tuning process. In this approach, an event function is designed to determine the control parameter update where the use of identification and predictive stages set the current control parameters. Furthermore, an elitist initialization in the differential evolution algorithm is also incorporated for solving the optimization problems at each stage. The speed regulation of the DC motor under disturbances is the study case in the ACCT approach. Comparative results with state-of-the-art bio-inspired algorithms in control tuning reveal in the AACT approach that the elitist initialization in differential evolution notably benefits the controller performance. Moreover, the comparative results with the Synchronous Adaptive Controller Tuning (SACT) approach show that the proposal reduces 61% the tuning process computation frequency with a similar speed regulation performance when disturbances appear.

INDEX TERMS Optimal controller tuning, DC motor, event based tuning, bio-inspired algorithm.

I. INTRODUCTION

From the beginning of the control theory in the 1940s [1], the main objective in the closed-loop system of the electromechanical actuators in a plant is the improvement of its performance. Two important issues emerge in the fulfillment of such objectives. The first one is to guarantee the closed-loop control system stabilization from the control theory point of view [2], [3]. The second one is related to achieve a suitable closed-loop control performance fulfilling a diverse set of specifications and requirements, called controller tuning. The work presented here is related to the latter issue.

The associate editor coordinating the review of this manuscript and approving it for publication was Ehab Elsayed Elattar.

At present, several works have focused on controller tuning, which is one of the most important problems in Intelligent Control (IC) [4]. The IC uses computational intelligence and soft computing to establish rule-based and knowledge-based system methodologies in the control tuning problem.

Several tuning methods have been developed since the introduction of the Proportional-Integral-Derivative (PID) controller tuning rules in 1942 [5]. Some of these methods are based on linear system theory [6]–[9], or optimization approaches for nonlinear systems [10]–[19]. Other works handle diverse control performance trade-offs to obtain different control gain alternatives to the designer [20]–[27].

The most promising controller tuning approach based on the classification in [19], [28] is the adaptive tuning method,

since it can effectively handle the uncertainties in practical applications [29], [30]. In this approach, the controller gains are continuously updated at each fixed time period (online control parameter update), typically by using predefined update rules [31] or computational intelligence techniques such as neural networks [32], fuzzy logic [33], and optimization methods [34].

Among computational intelligence techniques, evolutionary algorithms have been widely accepted in the tuning methods' continuous learning because they can obtain suitable solutions in noisy, nonlinear, or discontinuous environments present in real-world problems. For instance, the use of the Genetic Algorithm (GA) to dynamically tune the PI control gains of a linear induction motor is presented in [35]. Several bio-inspired algorithms, including Differential Evolution (DE), Particle Swarm Optimization (PSO), Bat Algorithm (BAT), Firefly Algorithm (FFA), Wolf Search Algorithm (WSA), and GA are studied in [36] for the speed controller tuning of a Direct Current (DC) motor. In [37], the (1+1)-Dynamic Evolution Strategy has been used to find the PI controller gains for a one-degree-of-freedom robotic mechanism. The use of different multi-objective evolutionary algorithms has been analyzed in the controller tuning of the four-bar mechanism [38].

The adaptive tuning methods' main feature is the periodic execution of the optimization process to set the new control gains at each predefined time period. By repeatedly optimizing the parameters of a controller, the performance of the plant increases, making it more robust to disturbances and parametric uncertainties, even more so when meta-heuristic optimizers are used, as observed in several recent works [35]–[38]. However, the computational cost required to perform such meta-heuristic optimization in practical applications is relatively high and could not be affordable for many computer systems. The above means that the implementation of those methods in a real process, a hardware-in-the-loop platform, or embedded devices may be prohibitive [39] due to the demand for high-performance computing. So, developing novel methods to support the online tuning process and save resources is one of the important research directions for the real-time implementation of adaptive tuning methods in practical applications.

A. CONTRIBUTIONS

In this paper, a novel Asynchronous Adaptive Controller Tuning (AACT) approach is proposed. In contrast to some reported works where the tuning process is set at each specific time interval, in the present work, the control parameter update times are reduced by the incorporation of an event condition. The proposed event function relates the rate of change of the Lyapunov function between two instants [40], [41]. Decreasing the control parameter updating, the computational cost for the overall tuning process can be reduced. This represents the first contribution.

The second contribution is related to the efficiency of the proposed AACT approach, allowing the control strategy to perform the optimized online tuning with an additional online parameter identification of the plant. Different from recent proposals that utilize a fixed-parameter model to perform the adaptive control [42]–[44], the proposed AACT includes an identification process to obtain the parameters of the plant and a predictive stage to know the future system behavior to find suitable control gains for the next time intervals. Both stages are based on an online optimization process solved through the bio-inspired algorithm called Online Differential Evolution (ODE), where an unusual elitist initialization to enhance the control performance is incorporated.

The proposed AACT approach is applied to the speed regulation of the Direct Current (DC) motor, which is one of the most used electromechanical actuators for industrial applications [45]–[49]. At this point, it is important to mention that bio-inspired algorithms are approximated stochastic methods, which means that every run deploys different outcomes. Therefore, although reasonable solutions are expected, their performance could be variable. The use of statistical tools allows verifying that the behavior of the solutions remains within a useful and competitive margin [50]. Moreover, it helps to perform a fair comparative analysis between the effects of different meta-heuristics in the control tuning strategy to determine an elite alternative for the addressed problem. The proposal can be in turn compared with advanced and classical control techniques to highlight its benefits. Because of the above, the obtained result is compared with the previous one in [51], other similar approaches that use different state-of-the-art bio-inspired algorithms, and also with a classical control approach through different study cases. On the other hand, no matter which meta-heuristic is adopted to optimize the controller, it has a set of hyper-parameters that compromise the quality of the solutions it can find [52]. A bad solution could affect the overall performance or even impact the system's stability. Unlike other recent researches, this work carefully selects the hyper-parameters of all the utilized bio-inspired algorithms using a large set of tests with the aid of the *i-race* package [53]. Simulation results for the speed regulation show that the proposal reduces the times that the tuning process is computed; also, the formal statistical study with the use of a careful selection of the hyper-parameters of the utilized bio-inspired algorithms generalizes the control tuning performance under disturbances. These last features represent the third contribution of this work.

This paper is organized as follows. The proposed AACT approach is detailed in Section II. The application of the proposal in the study case of the DC motor is given in Section III. Section IV includes the analysis and discussion of simulation results of the proposed AACT in the DC motor's speed control. Finally, the conclusions are drawn in Section V.

II. THE ASYNCHRONOUS ADAPTIVE CONTROLLER TUNING APPROACH

A. GENERAL DESIGN OF THE ACTIVATION FUNCTION FOR THE TUNING PROCESS

Equations in (1) describe the system dynamics in the state-space form of interest in this work, where $x(t) \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}$ is the input vector and $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^n$.

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{1}$$

In the proposed asynchronous adaptive controller tuning, the activation of the online tuning process is related to the function $\bar{e}(x(t), x(t-\Delta t)) : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$. It is called an event function, and it notifies the changes in the system stabilization behavior between the current time t and the previous one $t - \Delta t$, where Δt is related to the sampling period. Then, at each time $t > 0$, the event function $\bar{e}$ is computed. If the event condition is satisfied, the adaptive controller tuning process must be executed to find the controller's most suitable parameters. On the contrary, the tuning process is not carried out if the event condition is not satisfied.

As the Lyapunov function can describe the variation of system states, its derivative can inform about the rate at which it changes concerning such states. So, the proposed event function (2) is related to the difference of the rate of change of the Lyapunov function at the current time t and the previous one $t - \Delta t$. The stabilization problem is assumed in the control design, and the existence of a control system u that asymptotically stabilizes the closed-loop system to the origin must be guaranteed.

$$\bar{e}(x(t), x(t-\Delta t)) = \dot{V}(x(t)) - \dot{V}(x(t-\Delta t)) \tag{2}$$

If there exists a function $V(x(t))$ positive-definite where the derivative of V satisfies that $\dot{V}(x(t))$ is a negative definite function (guarantee of asymptotic stability), then, the function $\bar{e}$ (2) can be used as the event one. When the event function $\bar{e}$ satisfies the condition $\bar{e} \leq 0$, the controller performance in the current time t tends to be worse than in the previous time $t - \Delta t$. In this situation, the adaptive controller tuning process's activation is done, and new control parameters must be found. On the contrary, if the current controller performance is still better than the previous one, the condition $\bar{e} > 0$ is satisfied. This means that there is no need for new control parameters, and then the same control parameters are used in the current time (the adaptive tuning process is not required).

Proposition 1: Consider the variable change $z = x - \bar{x}$ and the control strategy u with the control gain vector $K \in \mathbb{R}^{1\times n}$ and the constant reference state vector $\bar{x}$

$$u = -Kz - \frac{B^T A\bar{x}}{\|B\|^2} \tag{3}$$

and the function $V : \mathbb{R}^n \to \mathbb{R}$ given by

$$V(z(t)) = z^T Pz \tag{4}$$

Then, a sufficient condition for using $\dot{V}(z(t))$ in the event function $\bar{e}(z(t), z(t-\Delta t)) : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ is that there exist two symmetric positive definite matrices $P \in \mathbb{R}^{n\times n}$ and $Q \in \mathbb{R}^{n\times n}$ [54], such that the matrix $(A-BK)^T P + P(A-BK) + Q$ is negative definite. This condition also implies that the origin of the transformed closed-loop system is asymptotically stable and the function $V(z(t))$ is a Lyapunov function.

Proof: Let assume that $P \in \mathbb{R}^{n\times n}$ and $Q \in \mathbb{R}^{n\times n}$ are two symmetric positive-definite matrices. Furthermore, the elements of $K \in \mathbb{R}^{1\times n}$ are positive real numbers. The system (1) expressed in the state vector z becomes

$$\dot{z}(t) = Az(t) + Bu(t) + A\bar{x} \tag{5}$$

Evaluating the time derivative of $V(z(t))$ and substituting the closed-loop dynamic system (5) with (3), results

$$\begin{aligned} \dot{V}(z(t)) &= \dot{z}^T Pz + z^T P\dot{z} \\ &= \dot{z}^T \Big((A-BK)^T P + P(A-BK) \Big) z \\ &= \underbrace{-z^T Qz}_{<0} \end{aligned} \tag{6}$$

Thus, $\dot{V}(z(t))$ is negative definite, and it can use in the event function $\bar{e}(z(t), z(t-\Delta t))$. As $\dot{V}(z(t)) < 0$, V is a Lyapunov function, and the controller u asymptotically stabilizes the closed-loop system to the origin. □

B. STAGES OF TUNING PROCESS

The asynchronous adaptive controller tuning requires the use of the event function previously explained. Once this is designed, AACT can be implemented. AACT includes two stages to set the parameters of the control system u (3). These stages consist of an identification stage to obtain the estimated parameter of the plant (1) grouped in $\tilde{\Theta}(t)$, and a predictive stage to set the controller parameter $K(t)$. The stages are described in Section II-B1 and Section II-B2. In Fig. 1, the proposed AACT is displayed for the plant dynamics expressed in (1). In this section, the explanation of the proposal is detailed.

The continuous time $t \in \mathbb{R}^+$ is divided into the discrete time sequence $\{t_l\}_{l\in\mathbb{N}}$ with $\mathbb{N} := \{1, 2, \ldots, n_{\mathbb{N}}\}$ and $t_0 = 0$, considering $t_{n_{\mathbb{N}}}$ as the final time instant. As in the emulation approach [55] to the digital controller implementation, the sampling time is referred to as $\Delta t = t_l - t_{l-1} > 0$. The time at which the event function activates, i.e., when it satisfies the condition $\bar{e} \leq 0$, is determined by $\Sigma_e = \{t_k\}_{k\in\mathbb{M}}$ with $\mathbb{M} := \{1, 2, \ldots, n_{\mathbb{M}}\}$, where $t_{n_{\mathbb{M}}} < t_{n_{\mathbb{N}}}$. The time interval between two consecutive events is referred as $t_{k+1} - t_k = n_s \Delta t \geq \Delta t$, considering the number of sampling time instants as $n_s \geq 1$. The graphical representation of the time sequence, the event time, and the evolution of the control system's parameters are visualized in Fig. 2 to get a better understanding.

In AACT for the plant in (1), the controller parameter vector $K(t)$ and the estimated plant parameter $\tilde{\Theta}(t)$ evolve through time. This evolution depends on the event function condition presented in (7)-(8), which is evaluated at each

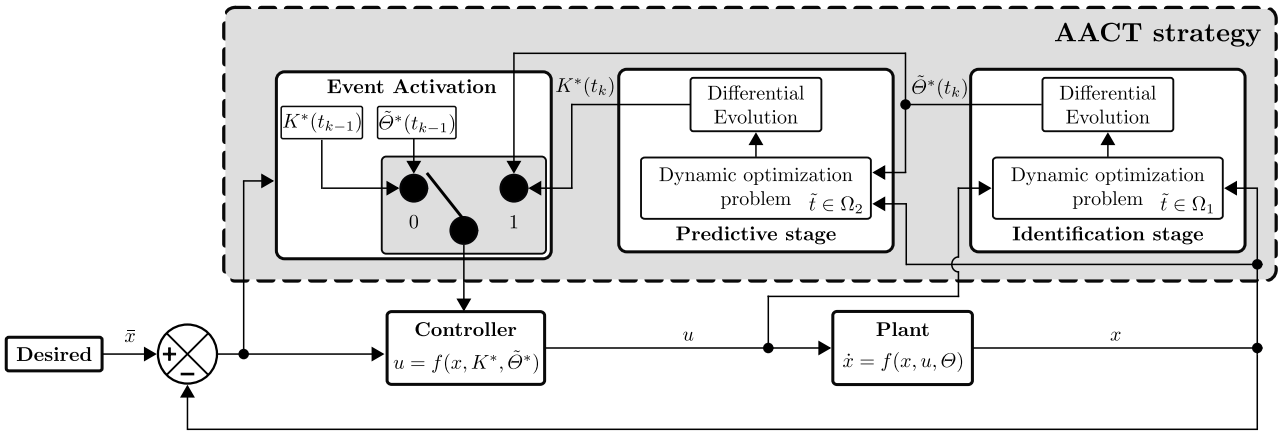

FIGURE 1. Schematic diagram of the proposed asynchronous adaptive controller tuning.

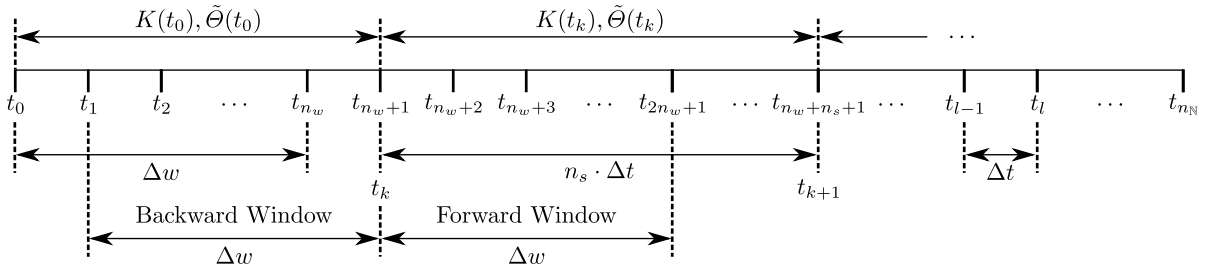

FIGURE 2. Graphical representation of the time division in the proposed asynchronous adaptive controller tuning.

sampling time Δt and explained next.

$$K(t) = \begin{cases} K(t_0) & \text{if } t_l \leq \Delta w \\ K(t_{k-1}) & \text{elseif } \bar{e}(t_l) > 0 \\ K(t_k) & \text{elseif } \bar{e}(t_l) \leq 0 \end{cases} \tag{7}$$

$$\tilde{\Theta}(t) = \begin{cases} \tilde{\Theta}(t_0) & \text{if } t_l \leq \Delta w \\ \tilde{\Theta}(t_{k-1}) & \text{elseif } \bar{e}(t_l) > 0 \\ \tilde{\Theta}(t_k) & \text{elseif } \bar{e}(t_l) \leq 0 \end{cases} \tag{8}$$

A backward/forward time window $\Delta w = n_w \Delta t \; \forall \; n_w > 1$ is required in AACT for the identification and predictive stages to estimate the plant and controller parameters, respectively. The evolution of $K(t)$ and $\tilde{\Theta}(t)$ presented in (7)-(8) requires the initial parameters of the controller $K(t_0)$ and the plant $\tilde{\Theta}(t_0)$. Those are set by the user to regulate the plant states with the controller (3) in the first $t \leq \Delta w$ seconds. When $t > \Delta w$ and the event function satisfies $\bar{e}(t_l) \leq 0$, the tuning process is activated to compute the new optimal set of parameters for both the plant and the controller $\tilde{\Theta}(t_k)$ and $K(t_k)$. The identification and predictive stages carry out the update of these parameters. Hence, the parameters $K(t)$ and $\tilde{\Theta}(t)$ are updated according to the parameters $K(t_k)$ and $\tilde{\Theta}(t_k)$ obtained by AACT. The new parameter vectors $K(t_k)$ and $\tilde{\Theta}(t_k)$ are incorporated in the control system u to reduce the regulation error in the interval $[t_k, t_{k+1}]$. On the other hand, if the event function is not activated ($\bar{e} > 0$) and $t > \Delta w$ in (7)-(8), the control action u is computed using the parameter vectors $K(t_{k-1})$ and $\tilde{\Theta}(t_{k-1})$ obtained in the last time when the controller parameter tuning was performed. Alternatively, if there is no previous event activation, the initial parameters $K(t_0)$ and $\tilde{\Theta}(t_0)$ are considered.

1) IDENTIFICATION STAGE

In the identification stage, the plant parameter $\tilde{\Theta}$ is found through the solution of an optimization problem. The optimization problem (9)-(11) consists of finding the optimum estimated plant parameter vector $\tilde{\Theta}^*$, which minimizes the error among the state vector $x(t)$ and the estimated ones $\tilde{x}(t)$ of the plant in the time spam $\Omega_1 \in [\tilde{t}_0, \tilde{t}_f] = [t_k - \Delta w, t_k]$ subject to the open-loop estimated plant dynamics (10) with its initial condition $\tilde{x}(\tilde{t}_0)$ and the bounds in the estimated plant parameter vector (11). The subscripts *max* and *min* represent the minimum and maximum allowed values in the associated variable.

$$\underset{\tilde{\Theta}^* \in R^3}{\textbf{min}} \int_{\tilde{t} \in \Omega_1} \left(x(\tilde{t}) - \tilde{x}(\tilde{t})\right)^T \left(x(\tilde{t}) - \tilde{x}(\tilde{t})\right) d\tilde{t} \tag{9}$$

$$\text{Subject to : } \dot{\tilde{x}} = f(\tilde{x}, u, \tilde{\Theta}), \;\; \tilde{x}(\tilde{t}_0) = x(t_k - \Delta w) \tag{10}$$

$$\tilde{\Theta}_{min} \leq \tilde{\Theta} \leq \tilde{\Theta}_{max} \tag{11}$$

The open-loop estimated plant dynamics (10) presents the same structure of (1). The state vector x with the control signal u is obtained from the plant dynamics and the control signal applies to it in the time window Ω_1. Therefore, this stage

provides a more realistic estimated plant behavior (model) once the optimization problem is solved. This model is used as a state predictor to the next stage.

2) PREDICTIVE STAGE

In the predictive stage, the state predictor computes the future system behavior considering the optimum motor parameter $\tilde{\Theta}^*(t_k)$ previously found in the identification stage with the purpose of setting the most suitable controller gains $K(t_k)$ in the control system. Hence, a second optimization problem (12)-(15) is established in the predictive stage to find the optimum controller parameter vector $K^*(t_k)$ in the time interval $\Omega_2 \in [\hat{t}_0, \hat{t}_f] = [t_k, t_k + \triangle w]$. The optimization problem consists of finding the optimum controller parameter $K^*(t_k)$ that minimizes the speed error (12) between the predictive state $\hat{x}$ and the desired state $\bar{x}$ subject to the closed-loop predictor dynamics (13), the bounds of the controller gains (14) and the limits of the control signal $\hat{u}$ (15). The solution of the closed-loop predictor dynamics requires the initial condition $\hat{x}(\hat{t}_0)$ and the control signal $\hat{u} = -K(\hat{x} - \bar{x}) - \frac{B^T A \bar{x}}{\|B\|^2}$. As in the previous state, the subscripts *max* and *min* represent the minimum and maximum allowed values in the associated variable.

$$\underset{K^* \in R^{1\times 2}}{\textbf{min}} \int_{\hat{t}\in\Omega_2} \left(x(\hat{t}) - \tilde{x}(\hat{t})\right)^T \left(x(\hat{t}) - \tilde{x}(\hat{t})\right) \, dt \quad (12)$$

$$\text{Subject to : } \dot{\hat{x}} = f(\hat{x}, \hat{u}, \tilde{\Theta}^*), \quad \hat{x}(\hat{t}_0) = x(t_k), \quad (13)$$

$$K_{min} \leq K \leq K_{max} \quad (14)$$

$$\hat{u}_{min} \leq \hat{u}(\hat{t}) \leq \hat{u}_{max} \quad (15)$$

The predictor's dynamics (13) presents the same structure of (1) with the optimum plant parameters $\tilde{\Theta}^*$. At the end of this optimization process, the optimum controller parameter $K^*(t_k)$ is found. Then, in the next time interval $t_k - t_{k+1}$, the vectors $K^*(t_k)$ and $\tilde{\Theta}^*(t_k)$ are included in the controller (3).

3) GENERAL OVERVIEW OF THE TUNING PROCESS STAGES

The incorporation of the ACCT approach into the numerical simulation of the plant dynamics is presented in Algorithm 1. The details of the optimization problem solution in the identification and predictive stages by the optimizer is detailed in Section II-C.

C. BIO-INSPIRED TECHNIQUES

One of the main elements of AACT is the bio-inspired optimizer, which solves the identification and predictive stages. According to the *No-free Lunch Theorem* [56], there is no universal optimizer that solves all kinds of optimization problems, i.e., a single optimizer cannot achieve the best performance for all kinds of problems.

In practical applications, the deviation of the controlled system from the desired task is an important factor in improving the quality of a process's final product. The measurement errors and actuating elements in the system, and external disturbances on it can deteriorate the regulation task. Then, the optimizer in the AACT approach must be as efficient as possible to obtain the control parameters under different scenarios. In order to improve the accuracy in the performance of the AACT approach, a variant of the DE technique, named Online Differential Evolution (ODE), is proposed to enhance the tuning process so that the task will be executed with higher precision. The pseudocode of the ODE is shown in Algorithm 2.

Algorithm 1 The Numerical Simulation of the Plant Dynamics With the AACT Approach

1: **Begin**
2: Set the initial condition $x(0) = x_0$ to the differential equation that describes the plant dynamics.
3: Time discretization $\{t_0, t_l\}_{l\in\mathbb{N}}$ with $\mathbb{N} := \{1, 2, \ldots, n_{\mathbb{N}}\}$ with the step size (integration time) Δt.
4: **for** $t \leftarrow t_0$ *to* $t_{n_{\mathbb{N}}}$ **do**
5: Set $K(t)$ and $\tilde{\Theta}(t)$ according to (7)-(8).
6: Evaluate the controller $u(t)$ (3).
7: Solve the differential equation of the plant dynamics (1) for the time $t + \Delta t$ with the Euler's method. $x(t + \Delta t) = x(t) + \Delta t \frac{dx(t)}{dt}$
8: **if** $(t > \Delta w)$ & $(\bar{e}(t) \leq 0)$ **then**
Event activation:
9: Identification stage for $\tilde{t} \in \Omega_1$. $\tilde{\Theta}(t)$=**Identify**$(x(\tilde{t})$,u$(\tilde{t})$,$\tilde{\Theta}(t))$
10: Predictive stage for $\hat{t} \in \Omega_2$. $K(t)$=**Predict**$(x(t), \tilde{\Theta}(t), K(t))$
11: **end if**
12: **end for**
13: **End**

The ODE operates over a population $\check{\mathbf{X}}^G \in \{\vec{\check{x}}_i^G\}$, $\forall\ i = 1, \ldots, NP$ that evolves during an established maximum number of generations G_{max}. Since several consecutive optimization processes are performed to update the controller parameters in the dynamic environment, a new search is started with ODE for every event time t_k in the proposed AACT approach. With the assumption that the evolution of the plant and control parameters is smooth through time in the dynamic environment, the initial population of the ODE does not start the search from scratch. Instead, ODE stores the best solution obtained in the previous optimization process from the tuning process in the time t_{k-1} and incorporates it in the random initial population of the optimization process for the current tuning task in the time t_k when the event function is activated. This does not waste time re-discovering an outstanding control parameter at the beginning of the search in the current optimization process. Hence, this solution is preserved unchanged through the evolutionary process unless other superior solutions are found. This promotes elitism through executions of the tuning process in the dynamic environment. So, it provides the probability of mixing the information of the best individual through the evolutionary process for the next tuning process. This initialization is given in steps 3-7 of the Algorithm 2.

Once the initial population is created, the evolution of the population is similar to the DE/rand/1/bin variant [57], i.e., the offsprings $\vec{u}_i^G$ are created for each generation G by using differential mutation and binomial crossover operators [58] with a constant scale factor F and a constant crossover rate CR. Those operators are given in steps 10-11 of the Algorithm 2. Each offspring $\vec{u}_i^G$ compete against the original individual $\vec{x}_i^G$ in the steps 12-16 of the Algorithm 2. The best alternative based on the performance function and the constraints is selected to generate the next generation's population. The selection criterion to define the solution that pass to the next generation is based on Deb's feasibility rules [59] stated as follows:

- Any feasible solution is preferred to any infeasible one.
- Among two feasible solutions, the one having better objective function value is preferred.
- Among two infeasible solutions, the one having smaller constraint violation number is preferred.
- Among two infeasible solutions with the same constraint violation number, one of them is preferred randomly with same probability.

At the end of the optimization process, promising solutions are found in the last generation. The best promising solution is considered the optimum one and those are used in the time $[t_k, t_{k+1}]$ of the dynamic simulation of the plant.

It is important to point out that due to the stochastic behavior of the bio-inspired search technique, the non-parametric statistics is required to guarantee the reliability of the obtained results in the AACT approach and to make general conclusions in the performance of the AACT approach.

III. APPLICATION OF THE AACT APPROACH IN THE DC MOTOR

A. ACTIVATION FUNCTION IN THE SPEED REGULATION OF DC MOTORS

Before presenting the event function $\bar{e}$ for the particular case in the DC motor, the corresponding dynamics is introduced. In Fig. 3, a simplified representation of the brushed DC motor is given, where the parameters associated with it represent the angle q, the angular speed $\dot{q}$, and the angular acceleration $\ddot{q}$ of the shaft. The armature current, the rotor inertia moment, the torque constant, the viscous friction coefficient, the armature resistance, the armature inductance, the load torque, the back electromotive force constant, and the input voltage are given by i_a, J_0, k_m, b_0, R_a, L_a, τ_L, k_e, V_{in}, respectively.

The corresponding dynamic model of the brushed DC motor [3] is presented in (16) for the electrical circuit equation and in (17) for the mechanical equation.

$$L_a\frac{di_a}{dt} + R_a i_a + k_e\dot{q} = V_{in} \tag{16}$$

$$J_0\ddot{q} + b_0\dot{q} = k_m\left(i_a - \frac{\tau_L}{k_m}\right) \tag{17}$$

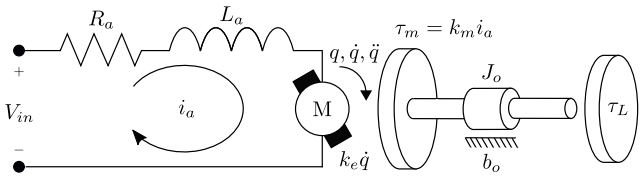

FIGURE 3. Schematic diagram of the DC motor.

The transformed dynamics of the DC motor in the state vector $x = [\dot{q}, \ddot{q}]^T$, and assuming $\tau_L = 0$, results

$$\dot{x}_1 = x_2 \tag{18}$$

$$\dot{x}_2 = \frac{1}{\theta_2}u - \frac{\theta_0}{\theta_2}x_1 - \frac{\theta_1}{\theta_2}x_2 \tag{19}$$

where $\theta_0 = k_e + \frac{R_a b_0}{k_m}$, $\theta_1 = \frac{J_0 R_a}{k_m} + \frac{L_a b_0}{k_m}$, $\theta_2 = \frac{J_0 L_a}{k_m}$ are the DC motor parameters.

Considering the desired state vector in the speed regulation of the DC motor as $\bar{x} = [\bar{x}_1, 0]^T$, the DC motor dynamics in the error state space $z = x - \bar{x}$ is given by

$$\dot{z}_1 = z_2 \tag{20}$$

$$\dot{z}_2 = \frac{1}{\theta_2}u - \frac{\theta_0}{\theta_2}z_1 - \frac{\theta_1}{\theta_2}z_2 + \frac{\theta_0}{\theta_2}\bar{z}_1 \tag{21}$$

With the DC motor dynamics (20)-(21) expressed in the form of (1), the controller u in (3) can be used to regulate the speed of the DC motor. For the particular case the controller u is given in (22), where $K = [k_1, k_2] \in \mathbb{R}^{1\times 2}$ contains the controller parameters and $\tilde{\theta}_0$ is the estimated parameter of the DC motor parameter θ_0. In the case that $\tilde{\theta}_0 = \theta_0$, the asymptotic convergence is guaranteed.

$$u = -Kz + \tilde{\theta}_0\bar{x}_1 \tag{22}$$

Then, the resulting event function $\bar{e}$ for the activation of the tuning process is given by

$$\bar{e}(t) = \underbrace{\sum\nolimits_{i=1}^{n} z_i^2(t - \Delta t))}_{-\dot{V}(z(t-\Delta t))} - \underbrace{\sum\nolimits_{i=1}^{n} z_i^2(t))}_{-\dot{V}(z(t))} \tag{23}$$

B. OPTIMIZATION PROBLEMS FOR IDENTIFICATION AND PREDICTIVE STAGES

For the particular study case in the DC motor, the estimated DC motor parameters are grouped in the vector $\tilde{\Theta} = [\tilde{\theta}_0, \tilde{\theta}_1, \tilde{\theta}_2]^T \in R^3$. The particular optimization problem for the identification stage is presented in (24)-(26). In this case, the objective function is based on the transformed dynamics of the DC motor in the state vector x, i.e., the error is among the current states $x(t)$ and the estimated ones $\tilde{x}(t)$ of the DC motor in the time spam $\Omega_1 \in [t_k - \triangle w, t_k]$. The dynamic constraint of the estimated DC motor dynamics (25) is considered in the problem with the bounds of the estimated DC motor parameters (26).

$$\underset{\tilde{\theta}^* \in R^3}{\textbf{min}} \int_{\tilde{t}\in\Omega_1} \left(x(\tilde{t}) - \tilde{x}(\tilde{t})\right)^T \left(x(\tilde{t}) - \tilde{x}(\tilde{t})\right) d\tilde{t} \tag{24}$$

$$\text{Subject to : } \dot{\tilde{x}} = \frac{1}{\tilde{\theta}_2}u(\tilde{t}) - \frac{\tilde{\theta}_0}{\tilde{\theta}_2}\tilde{x}_1 - \frac{\tilde{\theta}_1}{\tilde{\theta}_2}\tilde{x}_2$$

Algorithm 2 Online Differential Evolution in the Identification Stage or Predictive Stage

$\tilde{\Theta}(t) =$ **Identify**$(x(\tilde{t}),u(\tilde{t}),\tilde{\Theta}(t))$ or $K(t) =$ **Predict**$(x(t),\ \tilde{\Theta}(t),\ K(t))$

Input: The design parameter vector (estimated plant parameter vector) $\tilde{\Theta}(t)$, the state vector $x(\tilde{t})$ and the control signal vector $u(\tilde{t})$ in the backward time window for the identification stage.
Or
The design parameter vector (control gain vector) $K(t)$ and the current state vector $x(t)$ for the predictive stage.

Output: The current estimated plant parameter vector $\tilde{\Theta}^*(t)$ for the identification stage.
Or
The current gain vector $K^*(t)$ for the predictive stage.

Begin
  $G \leftarrow 1$
  Set the first individual of the initial population as the last time that were tuning.
    $\vec{\check{x}}_1^{G=1} = \tilde{\Theta}(t)$ for the identification stage.
    $\vec{\check{x}}_1^{G=1} = K(t)$ for the predictive stage.
  Randomly initialize the rest of the initial population $\check{\mathbf{X}}^G \in \{\vec{\check{x}}_i^G\}$, $\forall\, i = 2, \ldots, NP$ in the search space.
  for $i \leftarrow 1$ *toNP* **do**
    Evaluate the identification performance function $\tilde{J}(\vec{\check{x}}_i^G)$ (9) or the predictive performance function $\hat{J}(\vec{\check{x}}_i^G)$ (12) with the individual $\vec{\check{x}}_i^G$ of the initial population:
      In the identification stage: Solve the differential equation of the plant dynamics with Euler's method in the time $\tilde{t}$ (backward time (10) window), considering the state vector $x(\tilde{t})$ and the control signal $u(\tilde{t})$.
      Or
      In the predictive stage: Solve the differential equation of the plant dynamics (13) with Euler's method in the time $\hat{t}$ (forward time window), considering the estimated parameter vector $\tilde{\Theta}(t)$ with the current state $x(t)$ as the initial condition.
  end for
  while $G \leq G_{max}$ **do**
    for $i \leftarrow 1$ to NP **do**
      Generate the offspring $\vec{\check{u}}_i^G$ with differential mutation and binomial crossover operators.
      for $j \leftarrow 1$ **to** D **do**

$$\check{u}_{i,j}^G = \begin{cases} \check{x}_{r_1,j}^G + F(\check{x}_{r_2,j}^G - \check{x}_{r_3,j}^G) & \textbf{if } rand(0,1) < CR_i^G \textbf{ or } j = j_{rand} \\ \check{x}_{i,j}^G & \textbf{otherwise} \end{cases}$$

        Bound constraint handling for $\check{u}_{i,j}^G$ by random technique [60].
      end for
      Evaluate the identification performance function $\tilde{J}(\vec{\check{u}}_i^G)$ (9) or the predictive performance function $\hat{J}(\vec{\check{u}}_i^G)$ (12) with the offspring $\vec{\check{u}}_i^G$:
        In the identification stage: Solve the differential equation of the plant dynamics (10) with Euler's method in the time $\tilde{t}$ (backward time window), considering the state vector $x(\tilde{t})$ and the control signal $u(\tilde{t})$.
        Or
        In the predictive stage: Solve the differential equation of the plant dynamics (13) with Euler's method in the time $\hat{t}$ (forward time window), considering the estimated parameter vector $\tilde{\Theta}(t)$ with the current state $x(t)$ as the initial condition.
      if $\tilde{J}(\vec{\check{u}}_i^G)$ is better than $\tilde{J}(\vec{\check{x}}_i^G)$ **then**
        $\vec{\check{x}}_i^{G+1} \leftarrow \vec{\check{u}}_i^G$
      else
        $\vec{\check{x}}_i^{G+1} \leftarrow \vec{\check{x}}_i^G$
      end if
    end for
    $G \leftarrow G + 1$
  end while
  Select the best individual among individuals of the last generation.
  $\tilde{\Theta}^*(t) = best(\check{\mathbf{X}}^{G_{max}})$ for the identification stage.
  $K^*(t) = best(\check{\mathbf{X}}^{G_{max}})$ for the predictive stage.
  Return $\tilde{\Theta}^*(t)$ or $K^*(t)$
End

$$\tilde{x}(\tilde{t}_0) = x(\tilde{t}_0) \tag{25}$$

$$\tilde{\Theta}_{min} \leq \tilde{\Theta} \leq \tilde{\Theta}_{max} \tag{26}$$

On the other hand, in the predictive stage, the optimum controller parameter vector $K = [k_1, k_2] \in \mathbb{R}^{1\times 2}$ is found by solving the particular optimization problem (27)-(30). The objective function is related to the speed error of the predictive state vector $\hat{x}$ and the reference state one $\bar{x}$ considering the state predictor dynamics (28) with the corresponding initial condition as an equality dynamic constraint, and the bounds in the controller parameter vector (29) and in the control signal (30).

$$\underset{K^* \in R^{1\times 2}}{\textbf{min}} \int_{\hat{t}\in\Omega_2} \left(\hat{x}(\hat{t}) - \bar{x}(\hat{t})\right)^T \left(\hat{x}(\hat{t}) - \bar{x}(\hat{t})\right) dt \tag{27}$$

$$\text{Subject to}: \dot{\hat{x}}_2 = \frac{1}{\tilde{\theta}_2}u(\hat{x}, K) - \frac{\tilde{\theta}_0}{\tilde{\theta}_2}\hat{x}_1 - \frac{\tilde{\theta}_1}{\tilde{\theta}_2}\hat{x}_2 + \frac{\tilde{\theta}_0}{\tilde{\theta}_2}\bar{x}_1$$
$$\hat{x}(\hat{t}_0) = x(\hat{t}_0) \quad (28)$$
$$K_{min} \leq K \leq K_{max} \quad (29)$$
$$\hat{u}_{min} \leq \hat{u}(\hat{t}) \leq \hat{u}_{max} \quad (30)$$

The bounds of the motor model parameters $\tilde{\Theta}$ are established according to the suggestions given in [29]. On the other hand, the upper K_{max} and lower K_{min} bound vectors for the gains in the vector K are respectively chosen as $K_{max} = 2 \cdot K^{TEP}$ and $K_{min} = 2 \cdot K^{TEP}/10$, where K^{TEP} is the control gain vector tuned by a trial and error procedure considering $\tilde{\Theta}$ as a constant vector. On the other hand, the control signal bounds are set as $\hat{u}_{min} = -50V$ and $\hat{u}_{max} = 50V$. The controller parameter's lower and upper bounds are displayed in Table 1.

TABLE 1. Bounds of the controller parameters used for the optimization.

Parameter	Lower bound	Upper bound
$\tilde{\theta}_0$	0.1	5.0
$\tilde{\theta}_1$	$1.0E{-}3$	$5.0E{-}2$
$\tilde{\theta}_2$	$1.0E{-}5$	$5.0E{-}4$
k_1	$5.0E{-}2$	0.5
k_2	$1.0E{-}3$	$1.0E{-}2$

TABLE 2. Nominal motor parameters.

Parameter	Nominal Value
R_a	9.665 (Ω)
L_a	$102.44E{-}3$ (H)
k_m	0.3946 (N m/A)
k_e	0.4133 (V s/rad)
b_0	$5.85E{-}4$ (N m s)
J_0	$3.45E{-}4$ (N m s^2)
τ_L	0 (N m)

IV. RESULTS

This section shows the proposed Asynchronous Adaptive Controller Tuning (AACT) effectiveness through three comparative analyses. In this way, different AACT alternatives, based on various bio-inspired (BI) techniques, are compared in Section IV-A to evaluate their performance. The AACT performance is compared to the synchronous variant of this proposal in Section IV-B. Section IV-C contrasts the AACT's performance with that of the Proportional Integral (PI) controller, a widely used controller in the industry for speed regulation tasks.

Two test cases are considered for each comparative analysis:

- **Test Case 1 (TC1)**: The motor speed must be regulated to $\bar{x}_1 = 52.35$ (rad/s) during $t_f = 15$ (s), considering a sampling interval $\triangle t = 0.005$ (s), and the initial condition $x(0) = [0, 0]^T$. The nominal DC motor parameters remain static and are displayed in Table 2. This case is considered ideal and does not include noise, disturbances or changes in the speed profile.
- **Test Case 2 (TC2)**: Different from TC1, this case considers a more realistic scenario where random Gaussian noise with mean $\mu = 0$ and standard deviation $\sigma = 0.1$ is added to speed signal x_1, the torque load τ_l varies according to (31), and the speed profile $\bar{x}_1$ alternates as described in (32). The same nominal DC motor parameters, sampling interval and initial condition in TC1 are utilized in this case.

$$\tau_L = \begin{cases} 0.00\ (\text{N m}), & 0\ (\text{s}) \leq t < 2.5\ (\text{s}) \\ 0.20\ (\text{N m}), & 2.5\ (\text{s}) \leq t < 7.5\ (\text{s}) \\ 0.10\ (\text{N m}), & 7.5\ (\text{s}) \leq t < 12.5\ (\text{s}) \\ 0.00\ (\text{N m}), & 12.5\ (\text{s}) \leq t \leq 15\ (\text{s}) \end{cases} \quad (31)$$

$$\bar{x}_1 = \begin{cases} 52.35\ (\text{rad/s}), & 0\ (\text{s}) \leq t < 5\ (\text{s}) \\ 65.43\ (\text{rad/s}), & 5\ (\text{s}) \leq t < 10\ (\text{s}) \\ 39.26\ (\text{rad/s}), & 10\ (\text{s}) \leq t \leq 15\ (\text{s}) \end{cases} \quad (32)$$

All experiments required in the comparative analyses are carried out with a PC with a 3.60 GHz i7-4790 processor, and the results are detailed next.

A. COMPARATIVE ANALYSIS AMONG BIO-INSPIRED TECHNIQUES IN AACT

It is important to remark that the AACT proposal performs an identification process to obtain an optimized set of model parameters for a short backward time window. The identified model is then used to optimize the controller gains in a predictive stage for a forward time horizon. These two processes are established as optimization problems solved by a Bio-inspired (BI) technique ODE.

On the other hand, there are many optimizers in the specialized literature in control system tuning tasks, and three of them have been shown to have superior performance. These alternatives are the Differential Evolution (DE), the Genetic Algorithm (GA), and the Particle Swarm Optimization (PSO) [20], [27], [61]. Then, these three additional optimizers (BI techniques) are selected to work along with the AACT proposal for comparative purposes. The well-known DE/rand/1/bin variant of DE is adopted in this work [57], as well as the GA variant proposed in [29], and a PSO alternative with fully-connected topology and linear-decreasing inertia factor [62].

The bio-inspired optimizers have a set of hyper-parameters that determines their performance (and consequently, the controller performance) to a large extent. A subset of these hyper-parameters is shared by the aforementioned optimizers and determines the allowable number of problem evaluations. These hyper-parameters are the maximum number of generations/iterations G_{max} and the population/swarm size NP. Both parameters and the backward/forward time window $\triangle\omega$ clearly influence the utilized computational resources to perform the optimization. The larger the values of these parameters, the more computational time required for each optimization process. In the worst-case scenario, the

TABLE 3. The BI optimizers' parameters adjusted with *i-race* using the conditions of TC1 for AACT.

BI optimizer	Hyper-parameters
DE	$CR = 0.765$, $F = 0.692$
GA	$P_c = 1.000$*, $P_m = 0.755$, $\eta_c = 36$, $\eta_m = 80$
PSO	$C_1 = 1.717$, $C_2 = 2.321$, $\omega_1 = 0.433$, $\omega_2 = 0.060$
ODE	$CR = 0.352$, $F = 0.764$

* Manually set to ensure the same problem evaluations.

computational time used on any optimization process for the identification and predictive stages must be lesser than the sampling time interval Δt. So, considering a time window of $\Delta\omega = 50$ (ms), the parameters are set as $G_{max} = 200$ and $NP = 25$ for all optimizers to fulfill the above requirement with the available hardware. Regarding the rest of the hyper-parameters, they are adjusted iteratively with *i-race* [53] to perform fair comparisons in AACT. For this purpose, each optimizer's performance in AACT is determined in terms of the Integral Square Error (*ISE*) of the speed, within the interval $t \in [0.5, 15]$ (s), and considering the conditions of the TC1.

The next comparative analysis of the AACT, based on the selected BI techniques, shows the effectiveness of the proposal and allows determining a successful alternative to be used in practice. For simplicity, the AACT alternatives based on DE, GA, PSO, and ODE are referred to as AACT-DE, AACT-GA, AACT-PSO, and AACT-ODE, respectively. Due to the stochastic behavior of these optimizers, thirty independent runs are carried out to obtain relevant information about each AACT alternative.

Table 4 shows the descriptive statistical results of all AACT alternatives over the independent runs for TC1 and TC2. The first column indicates the test case, and the second one shows the evaluated AACT alternative. The mean and the standard deviation of the *ISE* values, calculated in the interval $t \in [0.5, 15]$ (s), are displayed in the third and fourth columns. The last two columns show the mean and the standard deviation of the number of activations of the event function (*AC*), i.e., when the optimizer is used to search for a new set of control parameters. Values in boldface indicate the best result of each column. As observed in the $\overline{ISE}$ column, AACT-ODE is the one that best regulates the motor speed when there are no disturbances, and is closely followed by AACT-DE. For the same TC1, AACT-PSO requires the least computational time to perform the speed regulation task according to the event activation (see column $\overline{AC}$ column). On the other hand, the reliability of AACT-GA and AACT-PSO is limited due to the notably larger standard deviation values (see $SD(\overline{ISE})$ and $SD(\overline{AC})$ columns), compared with those of AACT-DE and AACT-ODE. For the TC2, when a more realistic scenario is considered, AACT-DE and AACT-ODE develop similar performances in terms of the $\overline{ISE}$ column, while AACT-GA and AACT-PSO perform poorly. Moreover, AACT-ODE uses less computational burden regarding the event activation (see $\overline{AC}$). The reliability level of all the alternatives in TC2, given by the columns $SD(\overline{ISE})$ and $SD(\overline{AC})$, is similar to that

TABLE 4. Descriptive statistical results obtained with AACT alternatives over the ISE and the AC.

Test case	Alternative	$\overline{ISE}$	$SD(ISE)$	$\overline{AC}$	$SD(AC)$
TC1	AACT-DE	1.8698E-5	1.5567E-5	186.1000	**21.4546**
	AACT-GA	7.1528	12.0337	679.5333	432.3397
	AACT-PSO	0.4493	2.0593	**153.6000**	73.5112
	AACT-ODE	**1.0864E-5**	**1.4807E-5**	180.0666	31.0293
TC2	AACT-DE	27.6284	0.8198	1190.6000	**11.1559**
	AACT-GA	42.6191	12.5633	1191.9333	11.6705
	AACT-PSO	83.9568	55.9218	1198.9333	13.2377
	AACT-ODE	**27.6033**	**0.7063**	**1190.4666**	11.4462

TABLE 5. Results of the Wilcoxon test over the *ISE* distributions of the AACT alternatives.

Test Case	Comparative test	R_+	R_-	p-$value$
TC1	**AACT-DE** vs AACT-GA	462	3	9.3132E-9
	AACT-DE vs AACT-PSO	198	267	0.4898
	AACT-DE vs **AACT-ODE**	96	369	0.0040
	AACT-GA vs **AACT-PSO**	29	436	3.2391E-6
	AACT-GA vs **AACT-ODE**	0	465	1.8626E-9
	AACT-PSO vs AACT-ODE	241	224	0.8712
TC2	**AACT-DE** vs AACT-GA	461	4	1.3038E-08
	AACT-DE vs AACT-PSO	464	1	3.7252E-09
	AACT-DE vs AACT-ODE	237	228	0.9353
	AACT-GA vs AACT-PSO	426	39	1.3966E-05
	AACT-GA vs **AACT-ODE**	6	459	2.6077E-08
	AACT-PSO vs **AACT-ODE**	1	464	3.7252E-09

of the TC1, where AACT-ODE and AACT-DE are the most reliable alternatives followed by AACT-GA and, at a distance, by AACT-PSO.

Although the results in Table 4 give a good idea of the AACT performance, the stochastic nature of optimizers leads to different than normal distributions, such that it requires the analysis of nonparametric statistical tests. Because of the above, the Wilcoxon signed-rank test is adopted to draw strong conclusions about the results obtained with the AACT alternatives. The Wilcoxon test is applied to all possible pairs of the *ISE* distributions. The *two-sided* alternative hypothesis (this establishes that two distributions are different) is selected for this test, and the statistical significance is set as $\alpha = 5\%$. Table 5 shows the Wilcoxon test results over the *ISE* distributions of the AACT alternatives for TC1 and TC2. The test case is indicated in the first column, and the applied Wilcoxon test is observed in the second one. The third and fourth columns display the rank-sums, where R_+ indicates the times that the first distribution elements overcome the elements of the second one, and R_- indicates the opposite. The last column indicates the probability of rejecting the alternative hypothesis. Results in boldface highlight the winner alternative of each test. According to Table 5, AACT-ODE is the best alternative for the TC1, and its performance is comparable to that of AACT-DE for the TC2. Table 6 summarizes the number of wins in Table 5. Then, AACT-ODE is the overall best alternative and is shown in boldface.

It is also interesting to observe the behavior of the control signal in the AACT alternatives. For this, Table 7 shows the mean and the standard deviation of the Root Mean Square (*RMS*), indicated respectively as $\overline{RMS}$ and $SD(RMS)$, and

TABLE 6. Wins obtained with each AACT alternative after the Wilcoxon test.

Alternative	Wins in TC1	Wins in TC2	Total wins
AACT-DE	1	2	3
AACT-GA	0	1	1
AACT-PSO	1	0	1
AACT-ODE	2	2	4

calculated over the control signal history in the interval $t \in [0, 15]$ (s) for all AACT alternatives, runs, and test cases. In this table, when the $\overline{RMS}$ values of both test cases are analyzed separately, it is noticed that the magnitude of all control signals is very similar. Also, the $SD(RMS)$ indicates that the control signal magnitude has little variation between run and run.

Additionally, Figs. 4 and 5 illustrate the operation of the AACT alternatives for TC1 and TC2, respectively. These figures show the behavior of the best AACT execution for all optimizers based on the lowest number of event activations. The left column plots show the speed profile followed by the AACT using the adaptive controller parameters, and the speed regulation error is displayed in the inner plot. The corresponding behavior of the tuning process activation (given by the event function) is depicted in the right column plots. A high value denotes the computation of new control gains ($\bar{e} \leq 0$) in these plots, while a low value indicates the opposite. In the TC1, the first event activation in the AACT starts from $\triangle\omega$, and additional activations are performed at given instants until the control parameters suitably stabilize the motor speed. For the TC2, activations are observed at different instants during the task execution, since more tuning processes are required in the AACT to stabilize the output when disturbances, noise, or abrupt changes in the followed speed profile are detected by the event function. Regarding the plots at the left, all alternatives can stabilize the motor output for TC1 and TC2. In the AACT-GA and AACT-PSO plots, a steady-state error is noticeable in some sampled execution instances. The above is related to the high values of standard deviation in the ISE (see $SD(\overline{ISE})$ in Table 4) for both alternatives. Particularly for the TC2, it is important to highlight that all the AACT strategies can successfully compensate for the load disturbances, the presence of noise, and the changes in the desired speed through the online optimization of the controller parameters based on the identification and predictive stages. Moreover, each AACT requires some initial control tuning processes (activated by the event function $\bar{e}$) to stabilize the motor speed, as observed in the right plots. In this way, AACT-PSO and AACT-GA re-optimize the controller parameters a few times at the beginning of TC1, while AACT-ODE and AACT-DE do it for a longer period. In the TC2, all alternatives require a similar number of control tuning processes to handle the adverse scenario conditions.

Concerning the behavior of the control signals, Figs. 6 and 7 show the best responses (considering the run with the least number of event activations) calculated by the

TABLE 7. Descriptive statistical results obtained with AACT alternatives over the RMS of the control signal.

Test case	Alternative	$\overline{RMS}$	$SD(RMS)$
TC1	AACT-DE	22.3967	2.4986E-4
	AACT-GA	22.4355	6.6067E-2
	AACT-PSO	**22.3947**	7.5938E-2
	AACT-ODE	22.3967	**2.1116E-4**
TC2	AACT-DE	26.1171	7.5243E-3
	AACT-GA	26.1648	4.7884E-2
	AACT-PSO	**26.0636**	6.8777E-2
	AACT-ODE	26.1165	**7.3431E-3**

AACT alternatives for the TC1 and TC2, respectively. For the TC1, Fig. 6 indicates that all asynchronous strategies have an akin response, requiring a short initial interval to reach the speed reference. In the case TC2, Fig. 7 reveals some differences among the control signals. In this figure, AACT-DE and AACT-ODE have smoother responses than those of AACT-GA and AACT-PSO when disturbances or reference speed changes appear. The above is related to the capacity of AACT-DE and AACT-ODE to recover from those variations.

B. COMPARATIVE ANALYSIS BETWEEN THE AACT AND SACT APPROACHES

To gain insights about the AACT advantages, this approach is compared with a tuning approach where the controller parameters update is synchronous, i.e., the controller parameters are updated at every predefined time interval Δt. This tuning approach is called Synchronous Adaptive Controller Tuning (SACT). For a fair comparison, the only difference concerning AACT is the lack of asynchronous activation in SACT.

In [29], the comparative statistical study of SACT with other controllers, such as the Proportional Integral (PI) controller, the Model Reference Adaptive Controller (MRAC), and the Generalized Proportional Integral Observer-based Robust Controller (GPIO-RC), indicates that SACT is the best option under parametric uncertainties. The main discussion in this section is related to the performance of asynchronous (AACT) and synchronous (SACT) adaptive controller tuning based on bio-inspired algorithms.

Hence, the best AACT alternative, AACT-ODE, is compared with SACT based on ODE, referred to as SACT-ODE, to illustrate the asynchronous tuning benefits.

The hyper-parameters of ODE for SACT were also adjusted with *i-race* using the conditions of TC1 and the *ISE* indicator in the interval $t \in [0.5, 15]$ (s). Then, these parameters are set as $F = 0.870$ and $CR = 0.519$, while the rest of them remain as $G_{max} = 200$ and $NP = 25$ to perform fair comparisons.

Thirty independent runs of SACT-ODE are compared to those of AACT-ODE. Table 8 includes the results obtained by each approach for the TC1 and TC2. The meaning of each column is identical to that of Table 4. In the TC1, the performances of SACT-ODE and AACT-ODE are very

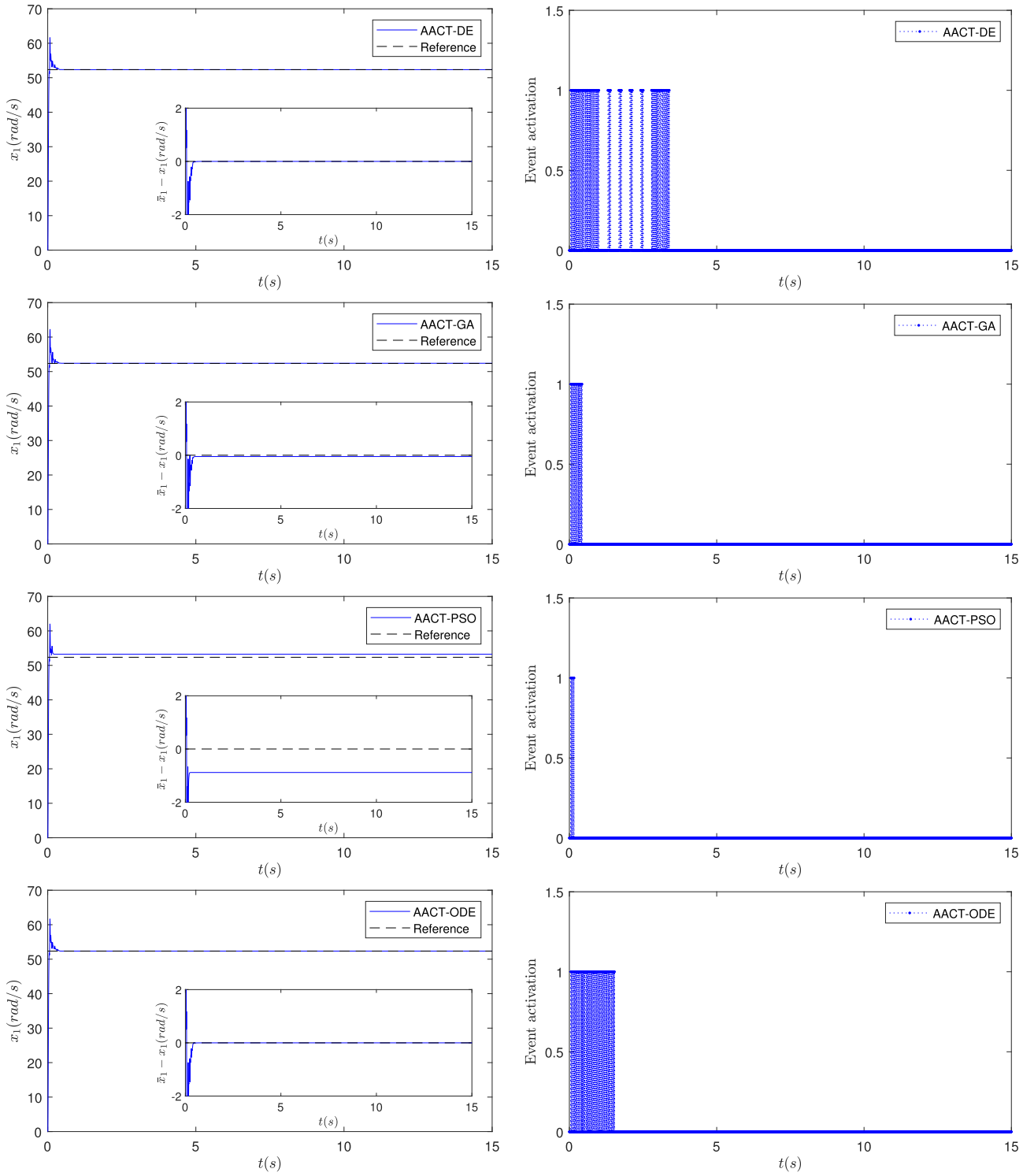

FIGURE 4. The best results obtained by AACT-DE, AACT-GA, AACT-PSO, and AACT-ODE concerning the least number of event activations for the TC1.

similar concerning the column $\overline{ISE}$ and are also reliable in terms of the standard deviation column $SD(\overline{ISE})$. Nevertheless, there is a noticeable difference in the computational burden required for both strategies. On average, AACT-ODE uses about 6% of the resources required by SACT-ODE (94% of saved computational time) to develop analogous performances according to $\overline{AC}$ and $SD(\overline{AC})$ columns. When considering the TC2, the response of SACT-ODE excels concerning that of AACT-ODE. However, the latter still completes the regulation task with an acceptable

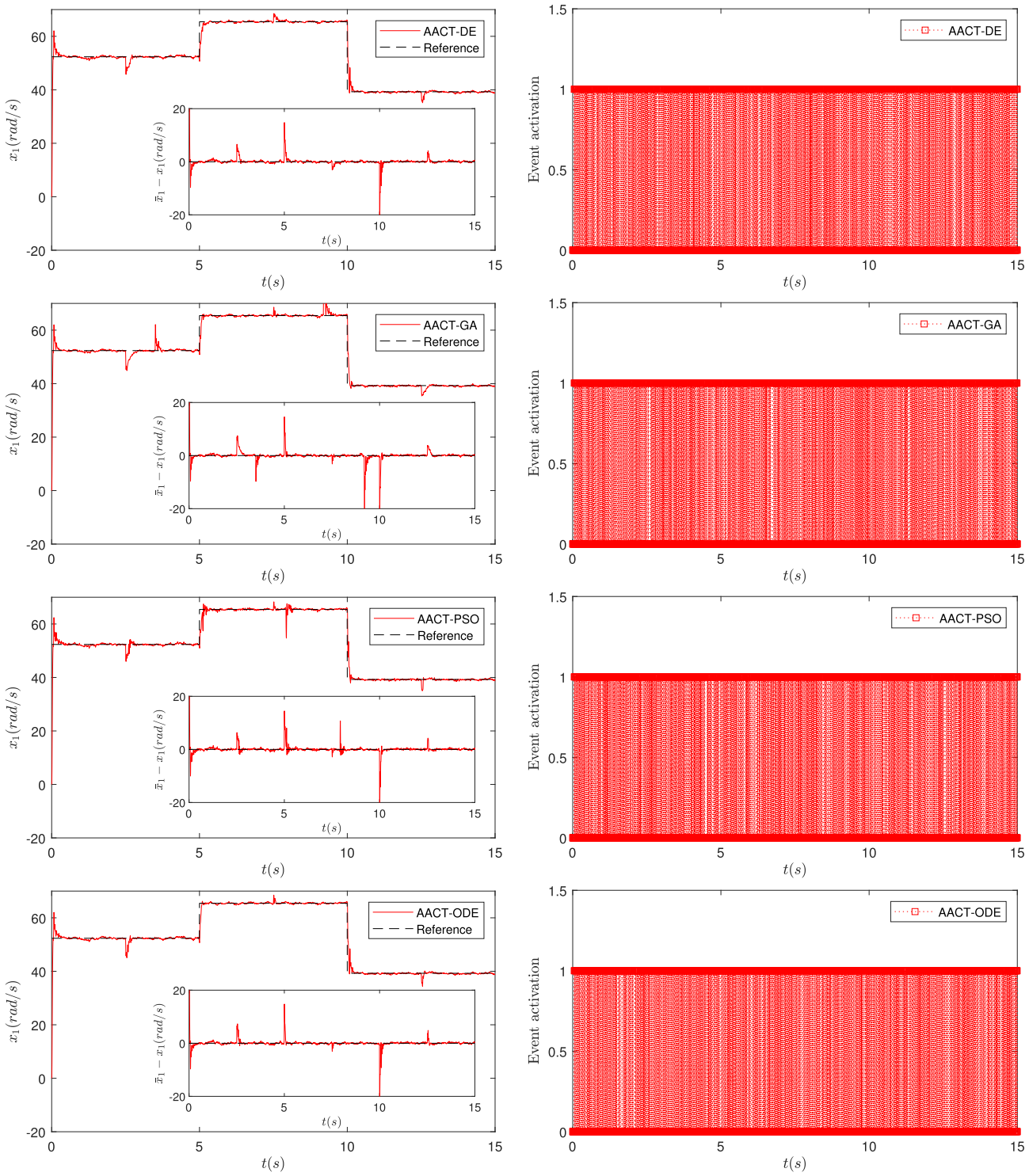

FIGURE 5. The best results obtained by AACT-DE, AACT-GA, AACT-PSO, and AACT-ODE concerning the least number of event activations for the TC2.

performance regarding the $\overline{ISE}$ column. Concerning to the reliability, the standard deviation (see $SD(\overline{ISE})$ column) of both alternatives remains proportional as in the results of TC1. Again, the major difference between SACT-ODE and AACT-ODE falls in the amount of used computational resources. Although AACT-ODE performs additional optimization processes to compensate for the load disturbances, the noise, and the reference changes, the invested resources are far from those used by SACT-ODE. Quantitatively, AACT-ODE requires about 39% of the computational time

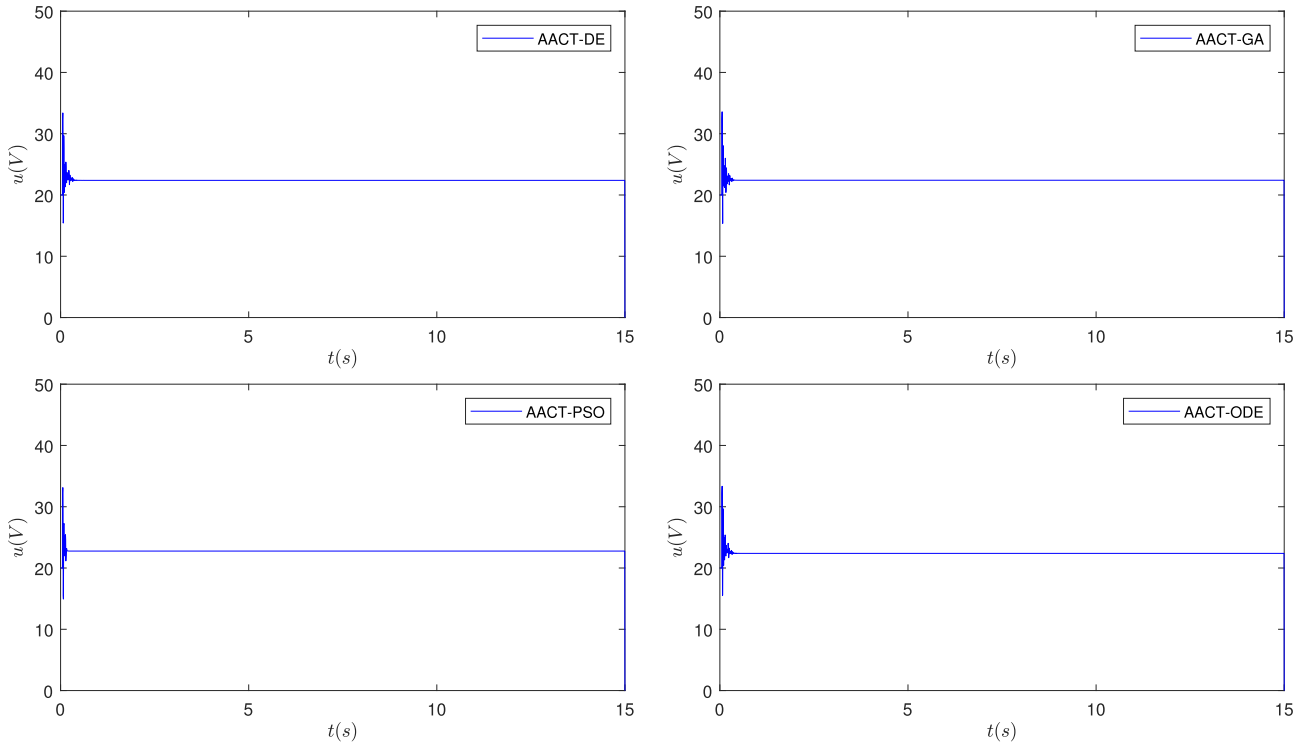

FIGURE 6. Behavior of the control signals calculated by AACT-DE, AACT-GA, AACT-PSO, and AACT-ODE concerning the least number of event activations for the TC1.

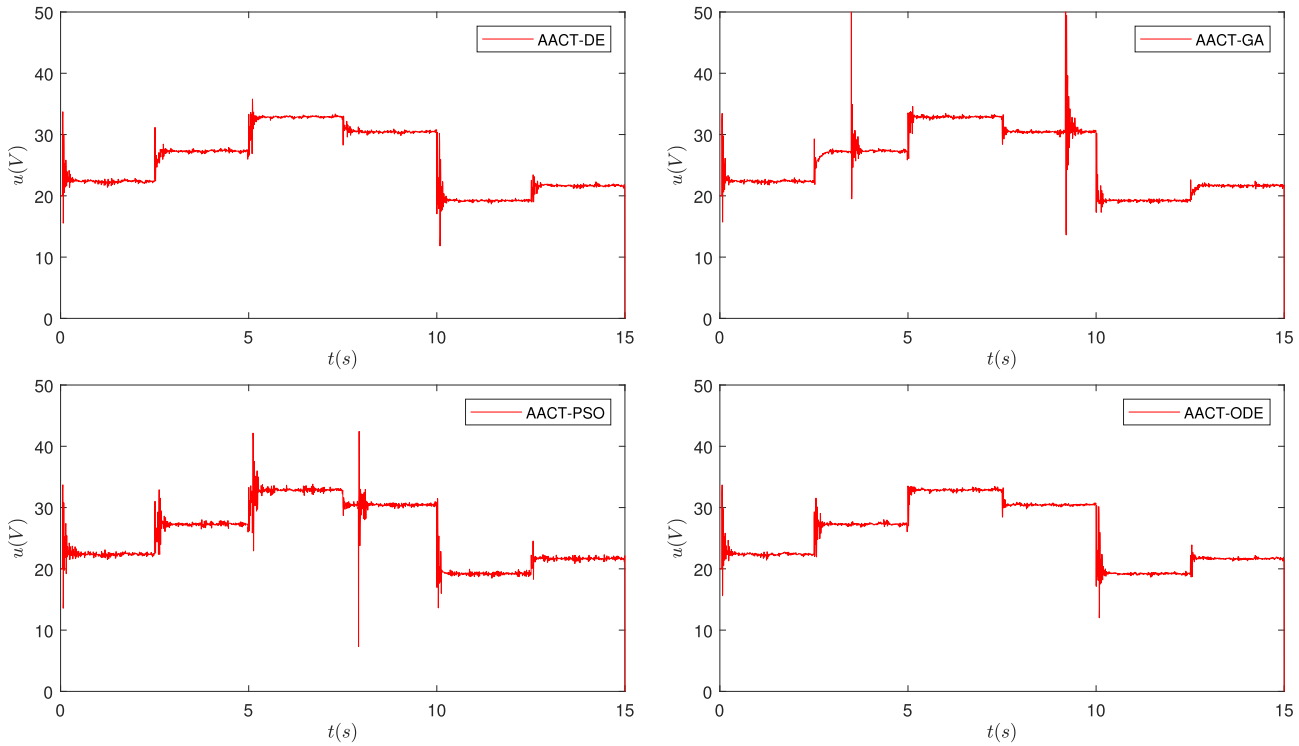

FIGURE 7. Behavior of the control signals calculated by AACT-DE, AACT-GA, AACT-PSO, and AACT-ODE concerning the least number of event activations for the TC2.

spent by SACT-ODE (61% of saved computational time) regarding the activation of the tuning process (see $\overline{AC}$ and $SD(\overline{AC})$).

Due to the stochastic behavior of AACT-ODE and SACT-ODE, the Wilcoxon test is performed again to draw strong conclusions about their differences. A *two-sided*

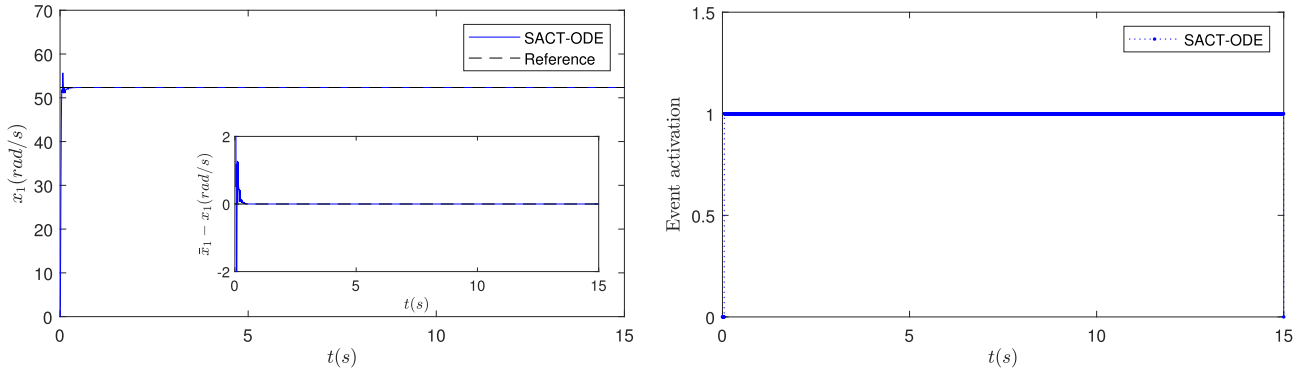

FIGURE 8. The best result obtained by SACT-ODE regarding the minimum value of *ISE* for the TC1.

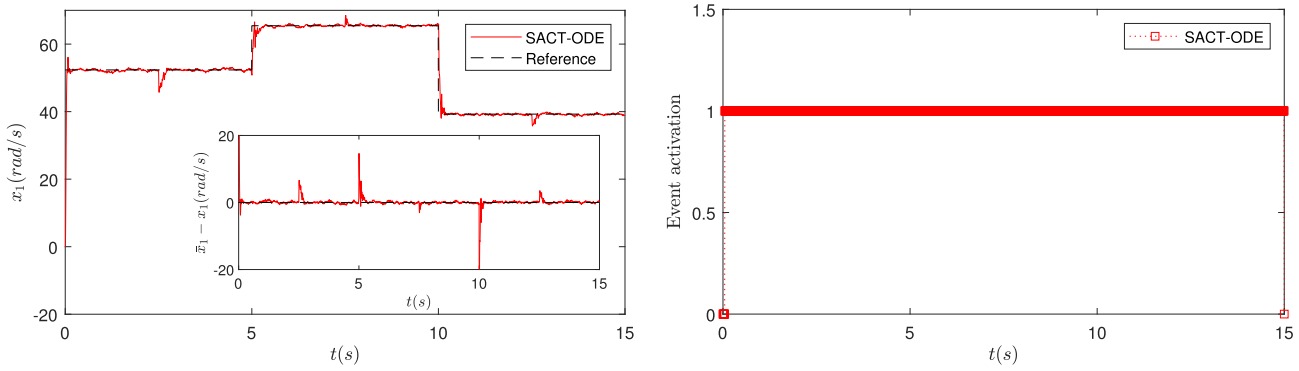

FIGURE 9. The best result obtained by SACT-ODE regarding the minimum value of *ISE* for the TC2.

TABLE 8. Descriptive statistical results obtained with AACT-ODE and SACT-ODE over the ISE and the AC.

Test case	Alternative	$\overline{ISE}$	$SD(\overline{ISE})$	$\overline{AC}$	$SD(\overline{AC})$
TC1	AACT-ODE	1.0864E-5	**1.4807E-5**	**180.0666**	31.0293
	SACT-ODE	**1.0298E-5**	2.7519E-5	2990.0000	**0.0000**
TC2	AACT-ODE	27.6033	0.7063	**1190.4666**	11.4462
	SACT-ODE	**25.9013**	**0.2292**	2990.0000	**0.0000**

TABLE 9. Results of the Wilcoxon test over the *ISE* distributions of the AACT alternatives.

Test Case	Comparative test	R_+	R_-	p-$value$
TC1	SACT-ODE vs AACT-ODE	294	171	0.2129
TC2	**SACT-ODE** vs AACT-ODE	464	1	3.7252E-09

alternative hypothesis is chosen as well as a test significance of $\alpha = 5\%$. Table 9 reveals the results of the Wilcoxon test between AACT-ODE and SACT-ODE for the TC1 and TC2. This table includes the test case and the comparative test in the first two columns. The next columns are the rank-sums R_+ and R_-, and the probability p-$value$. The winner of each test is highlighted in boldface. According to Table 9, there are no significant differences in the SACT-ODE and AACT-ODE behavior in the TC1. On the other hand, SACT-ODE clearly overcomes AACT-ODE for the TC2.

Concerning the behavior of the control signal in the synchronous and asynchronous cases, Table 10 shows the mean

TABLE 10. Descriptive statistical results obtained with AACT-ODE and SACT-ODE over the RMS of the control signal.

Test case	Alternative	$\overline{RMS}$	$SD(\overline{RMS})$
TC1	AACT-ODE	22.3967	**2.1116E-4**
	SACT-ODE	**22.3790**	2.6369E-4
TC2	AACT-ODE	26.1165	7.3431E-3
	SACT-ODE	**26.0905**	**3.4518E-3**

and the standard deviation of the response *RMS*, i.e., $\overline{RMS}$ and $SD(RMS)$ obtained by AACT-ODE and SACT-ODE in the interval $t \in [0, 15]$ (s) for all runs and test cases. In this table, it is observed in the $\overline{RMS}$ values that the magnitude of the SACT-ODE responses are slightly less than those of the AACT-ODE in TC1 and TC2. In addition, these magnitudes are mildly less variable in the AACT-ODE, according to the $SD(RMS)$ results.

The best execution of SACT-ODE regarding the minimum value of *ISE* is observed in Figs. 8 and 9 for the TC1 and TC2, respectively. The plot at the left shows the motor speed, and the subplot exhibits the error in the speed regulation. Also, the plot at the right displays the activation of the controller tuning processes. On the left column of these figures, it can be observed that the speed profiles of SACT-ODE are not visibly distinct from the responses of AACT-ODE in the left columns of Figs. 4 and 5. Concerning the tuning process

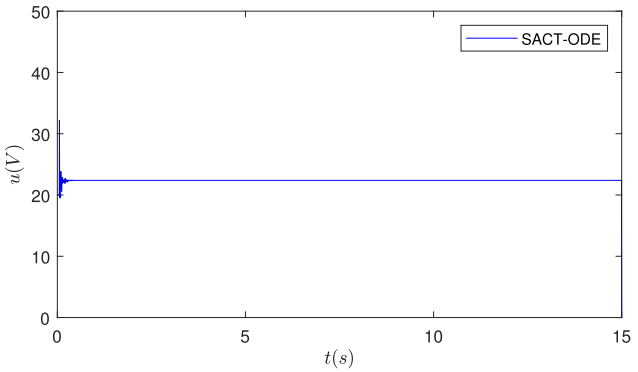

FIGURE 10. Behavior of the control signal calculated by SACT-ODE concerning the minimum value of *ISE* for the TC1.

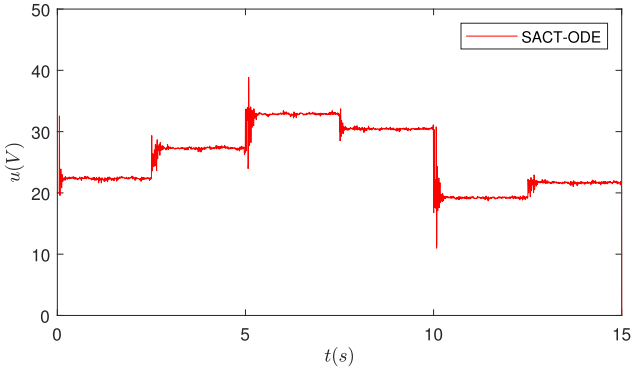

FIGURE 11. Behavior of the control signal calculated by SACT-ODE concerning the minimum value of *ISE* for the TC2.

activations, as it was expected, the plot at the right highlights that these processes are performed synchronously from the instant $\triangle\omega = 50$ (ms) at each sampling time.

On the other hand, the behavior of the control signal computed by SACT-ODE in the best run (concerning the minimum value of *ISE*) is observed in Figs. 10 and 11 for the TC1 and TC2, respectively. The plotted control signals do not present noticeable differences from the responses shown in Figs. 6 and 7 for the AACT-ODE. In the same way, abrupt control actions are observed that compensate for the changes in the speed profile and the variations in the torque load.

Based on the above results, AACT-ODE showed an outstanding performance in the speed regulation of the DC motor for the two proposed test cases. This performance is comparable to that of the synchronous strategy SACT-ODE, but with the advantage of using a much smaller amount of computational resources. The saved computational time in the AACT-ODE can be used in sensory information processing, communication, or general calculations.

C. COMPARATIVE ANALYSIS BETWEEN THE AACT APPROACH AND THE PI CONTROLLER

The Proportional Integral (PI) controller is one of the schemes more often used in the industry for DC motors' speed regulation [63]–[65]. Hence, it can be used as a reference technique to determine some facts about the AACT operation.

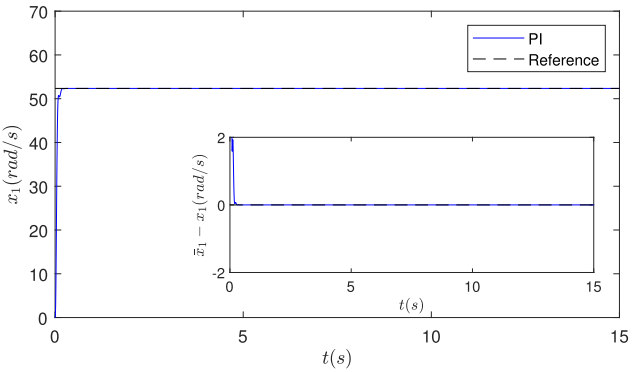

FIGURE 12. The result obtained by PI controller for the TC1.

The proportional and integral gains compromise the performance of the PI controller, respectively named as k_p and k_i. These gains are adjusted automatically through *i-race* considering the *ISE* metric in the interval $t \in [0.5, 15]$ (s) for the scenario TC1. The obtained gains are $k_p = 0.021$ and $k_i = 9.134$.

Since the PI controller has a deterministic behavior, a single run is necessary to evaluate its performance. The information about the single runs for TC1 and TC2 is shown in Table 11. In this table, the first column indicates the test case, the second one shows the *ISE* values computed within the interval $t \in [0.5, 15]$ (s), and the last one exhibits the *RMS* of the control signal calculated for $t \in [0, 15]$ (s).

TABLE 11. Results obtained with the PI controller concerning the ISE and the RMS of the control signal.

Test case	ISE	RMS
TC1	1.0175E-12	22.3559
TC2	40.3452	26.0946

When ideal and controlled operating conditions are considered (scenario TC1), the behavior of the PI controller is almost perfect, as shown in the *ISE* column in Table 11. On the other side, when the motor is subject to disturbances and uncertainties (scenario TC2), the performance in the response of the PI controller is far from the obtained in the ideal case. If the above effect is translated to the practice, where all dynamic systems find uncertainties and disturbances, the performance of the PI controller can decay even more. Moreover, if it is taken into account that the PI controller tuning was performed using an estimated model of the DC motor, the performance is also compromised by the model accuracy.

Unlike the PI controller, the AACT approach can address the above difficulties through the optimized identification and prediction processes, thereby reducing the dependence on an exact model and, in consequence, being less affected by the effects of uncertainties and disturbances as observed in the results for the TC2 in Table 4.

The above can be observed in Figs. 12 and 13 for the TC1 and TC2, respectively. In these figures, it can be noticed

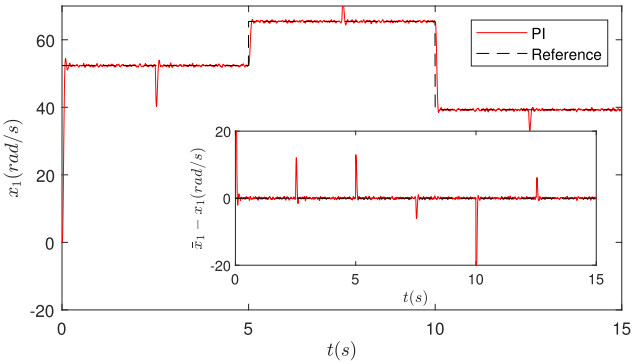

FIGURE 13. The result obtained by PI controller for the TC2.

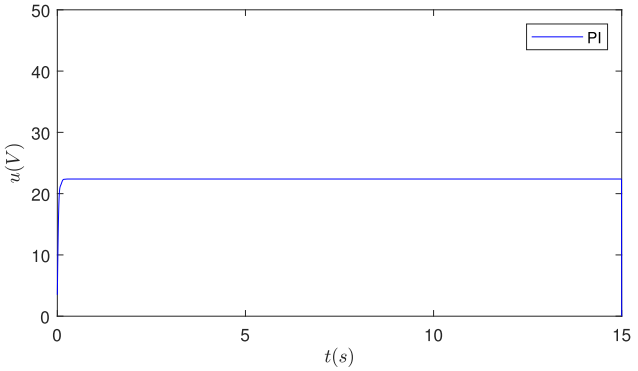

FIGURE 14. Behavior of the control signal calculated by the PI controller for the TC1.

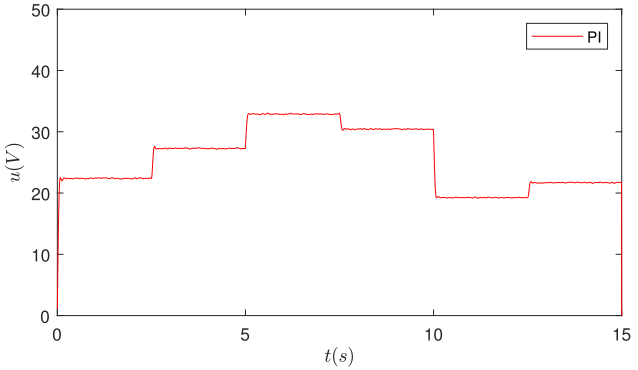

FIGURE 15. Behavior of the control signal calculated by the PI controller for the TC2.

that the PI controller has an outstanding performance from the beginning of TC1. However, for TC2, the motor load disturbances, the noise signal, and the changes in the speed reference affect the response of the PI controller response more than it does with AACT-ODE in Fig. 4.

Additionally, Figs. 14 and 15 describe the behavior of the control signals calculated by the PI controller for the TC1 and TC2, respectively. Due to the linear nature of this control scheme, the plotted control actions are notably smoother than those of the AACT alternatives for both test cases (see Figs. 6 and 7). Nevertheless, the stochastic behavior of the AACT responses is useful to compensate for the negative effects of noise, disturbances, and changes in the reference.

Regarding the computation time, the cost of the PI controller is negligible, which contrasts with the computational load required in AACT. However, in applications that require high-performance rates under disturbances and uncertainties, the cost of an AACT alternative can be affordable.

V. CONCLUSION

In this work, a novel Asynchronous Adaptive Controller Tuning (AACT) approach is proposed to reduce the overall computational cost of the controller parameter adjustment for a plant. This saves computational resources which can be used for other tasks such as in the sensory information processing, communications, or general calculations. Also, the AACT approach can provide a suitable control performance under disturbances, noise, and changes in the reference.

The main features of the AACT approach are: *i*) The tuning process is only activated when an event occurs. The above is based on the Lyapunov function changes, which successfully determine the time instant when the system states deviate from the regulation point. *ii*) The AACT approach incorporates an identification stage and a predictive one, as sequential optimization problems when the event activation occurs. *iii*) The elitist initialization in the proposed online differential evolution algorithm exhaustively finds solutions in the search space of the two stages, promoting control parameter solutions that can suitably tackle the regulation task of the plant. *iv*) The approach can be applied to dynamic systems represented by the form $\dot{x} = Ax(t) + Bu(t)$.

This approach is applied to the speed regulation task of the DC motor subject to torque load variations, random Gaussian noise in the speed signal, and abrupt changes in the speed profile. After analyzing the behavior of three well-known optimizers and the proposed improved version of the differential evolution algorithm, it is observed that the proposed Online Differential Evolution (ODE) achieved the best results in the AACT approach.

When comparing the AACT approach with the Synchronous Adaptive Controller Tuning (SACT) approach, some former advantages are revealed. An important benefit is that the performance of the AACT is comparable with that of the SACT approach, but the computational burden required by the AACT is significantly lower in around 61% and 94% than the SACT with and without adverse conditions, respectively.

On the other hand, when AACT is compared to the PI controller, an outstanding performance of AACT is observed in the presence of load disturbances, random noise, an changes in the speed reference. The above is attributed to the adaptive tuning based on identification and prediction, and the incorporation of ODE, which reduces the dependence on an exact motor model.

The comparative results obtained through numerical simulations confirm that the proposed AACT approach can significantly reduce the continuous computation of controller

parameters while paying with a slight increment of the speed error when disturbances appear.

The complexity of the plant dynamics in the proposed AACT approach may significantly increase the convergence time of the optimizer and, in the worst case, this time could be higher than the sampling interval. So, future work involves incorporating micro evolutionary algorithms with the use of neuronal networks in the estimated dynamics for using them in the adaptive control tuning for more complex systems.

REFERENCES

[1] K. J. Åström and R. M. Murray, *Feedback Systems: An Introduction for Scientists Engineers*. Princeton, NJ, USA: Princeton Univ. Press, 2008.

[2] R. Krishnan, *Electric Motor Drives: Modeling, Analysis, and Control*, vol. 626. Upper Saddle River, NJ, USA: Prentice-Hall, 2001.

[3] J. Chiasson, *Modeling and High-Performance Control of Electric Machines*, vol. 26. Hoboken, NJ, USA: Wiley, 2005.

[4] A. E. Ruano, "Intelligent control–the road ahead," in *Proc. Eur. Control Conf. (ECC)*, Jul. 2007, pp. 4442–4443.

[5] J. G. Ziegler, N. B. Nichols, and N. Y. Rochester, "Optimum settings for automatic controllers," *Trans. ASME*, vol. 64, pp. 759–768, Nov. 1942.

[6] K. Astrom and T. Hagglund, *PID Controllers: Theory, Design and Tuning*. Gurugram, India: ISA, 1995.

[7] M. Sadeghpour, V. D. Oliveira, and A. Karimi, "A toolbox for robust PID controller tuning using convex optimization," *IFAC Proc. Volumes*, vol. 45, no. 3, pp. 158–163, 2012.

[8] M. Hast, K. J. Åström, B. Bernhardsson, and S. Boyd, "PID design by convex-concave optimization," in *Proc. Eur. Control Conf. (ECC)*, 2013, pp. 4460–4465.

[9] P. Mercader, K. J. Astrom, A. Banos, and T. Hagglund, "Robust PID design based on QFT and convex–concave optimization," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 2, pp. 441–452, Mar. 2017.

[10] D. Izci, "Design and application of an optimally tuned PID controller for DC motor speed regulation via a novel hybrid Lévy flight distribution and Nelder–Mead algorithm," *Trans. Inst. Meas. Control*, vol. 2021, Jun. 2021, Art. no. 01423312211019633.

[11] T. Marques and G. Reynoso-Meza, "Single-objective optimization with gain scheduling control," in *Proc. IEEE Int. Conf. Automat./XXIV Congr. Chilean Assoc. Automat. Control (ICA-ACCA)*, Mar. 2021, pp. 1–6.

[12] O. Serrano-Pérez, M. G. Villarreal-Cervantes, A. Rodríguez-Molina, and J. Serrano-Pérez, "Offline robust tuning of the motion control for omnidirectional mobile robots," *Appl. Soft Comput.*, vol. 110, Oct. 2021, Art. no. 107648.

[13] T. Marques and G. Reynoso-Meza, "Applications of multi-objective optimisation for PID-like controller tuning: A 2015-2019 review and analysis," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 7933–7940, 2020.

[14] M. J. Mahmoodabadi and J. Rafee, "A new optimal robust adaptive fuzzy controller for a class of non-linear under-actuated systems," *Syst. Sci. Control Eng.*, vol. 8, no. 1, pp. 359–368, Jan. 2020.

[15] E. Yazid, M. Garratt, and F. Santoso, "Position control of a quadcopter drone using evolutionary algorithms-based self-tuning for first-order Takagi–Sugeno–Kang fuzzy logic autopilots," *Appl. Soft Comput.*, vol. 78, pp. 373–392, May 2019.

[16] O. Serrano-Pérez, M. G. Villarreal-Cervantes, J. C. González-Robles, and A. Rodríguez-Molina, "Meta-heuristic algorithms for the control tuning of omnidirectional mobile robots," *Eng. Optim.*, vol. 52, no. 2, pp. 325–342, Feb. 2020.

[17] H. V. H. Ayala and L. dos Santos Coelho, "Tuning of PID controller based on a multiobjective genetic algorithm applied to a robotic manipulator," *Expert Syst. Appl.*, vol. 39, no. 10, pp. 8968–8974, Aug. 2012.

[18] L. Amador-Angulo and O. Castillo, "A new fuzzy bee colony optimization with dynamic adaptation of parameters using interval type-2 fuzzy logic for tuning fuzzy controllers," *Soft Comput.*, vol. 22, no. 2, pp. 571–594, Jan. 2018.

[19] M. G. Villarreal-Cervantes and J. Alvarez-Gallegos, "Off-line PID control tuning for a planar parallel robot using DE variants," *Expert Syst. Appl.*, vol. 64, pp. 444–454, Dec. 2016.

[20] G. Reynoso-Meza, J. Sanchis, X. Blasco, and M. Martínez, "Algoritmos evolutivos y su empleo en el ajuste de controladores del tipo PID: Estado actual y perspectivas," *Revista Iberoamericana Automática Informática Ind.*, vol. 10, no. 3, pp. 251–268, Jul. 2013.

[21] A. Susperregui, J. M. Herrero, M. I. Martínez, G. Tapia-Otaegui, and X. Blasco, "Multi-objective optimisation-based tuning of two second-order sliding-mode controller variants for DFIGs connected to non-ideal grid voltage," *Energies*, vol. 12, no. 19, p. 3782, 2019.

[22] K. Quresh, S. Rahnamayan, Y. He, and R. Liscano, "Enhancing LQR controller using optimized real-time system by GDE3 and NSGA-II algorithms and comparing with conventional method," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jun. 2019, pp. 2074–2081.

[23] X. Zhou and X. Zhang, "Multi-objective-optimization-based control parameters auto-tuning for aerial manipulators," *Int. J. Adv. Robotic Syst.*, vol. 16, no. 1, Feb. 2019, Art. no. 1729881419828071.

[24] C. Zhang, T. Peng, C. Li, W. Fu, X. Xia, and X. Xue, "Multiobjective optimization of a fractional-order PID controller for pumped turbine governing system using an improved NSGA-III algorithm under multiworking conditions," *Complexity*, vol. 2019, pp. 1–18, Feb. 2019.

[25] O. T. Altinoz, *Optimal Controller Parameter Tuning from Multi/Many-objective Optimization Algorithms*. Cham, Switzerland: Springer, 2019, pp. 51–74.

[26] X. Zhou, J. Zhou, C. Yang, and W. Gui, "Set-point tracking and multi-objective optimization-based PID control for the goethite process," *IEEE Access*, vol. 6, pp. 36683–36698, 2018.

[27] A. Rodríguez-Molina, E. Mezura-Montes, M. G. Villarreal-Cervantes, and M. Aldape-Pérez, "Multi-objective meta-heuristic optimization in intelligent control: A survey on the controller tuning problem," *Appl. Soft Comput.*, vol. 93, Aug. 2020, Art. no. 106342.

[28] M. G. Villarreal-Cervantes, A. Rodríguez-Molina, C. García-Mendoza, O. Peñaloza-Mejía, and G. Sepúlveda-Cervantes, "Multi-objective on-line optimization approach for the DC motor controller tuning using differential evolution," *IEEE Access*, vol. 5, pp. 20393–20407, 2017.

[29] A. Rodríguez-Molina, M. G. Villarreal-Cervantes, J. Álvarez-Gallegos, and M. Aldape-Pérez, "Bio-inspired adaptive control strategy for the highly efficient speed regulation of the DC motor under parametric uncertainty," *Appl. Soft Comput.*, vol. 75, pp. 29–45, Feb. 2019.

[30] A. Rodríguez-Molina, M. G. Villarreal-Cervantes, and M. Aldape-Pérez, "Indirect adaptive control using the novel online hypervolume-based differential evolution for the four-bar mechanism," *Mechatronics*, vol. 69, Aug. 2020, Art. no. 102384.

[31] J. Rothe, J. Zevering, M. Strohmeier, and S. Montenegro, "A modified model reference adaptive controller (M-MRAC) using an updated MIT-rule for the altitude of a UAV," *Electronics*, vol. 9, no. 7, p. 1104, Jul. 2020.

[32] M. K. Debnath, R. Agrawal, S. R. Tripathy, and S. Choudhury, "Artificial neural network tuned PID controller for LFC investigation including distributed generation," *Int. J. Numer. Model., Electron. Netw., Devices Fields*, vol. 33, no. 5, p. e2740, Sep. 2020, doi: 10.1002/jnm.2740.

[33] C. Conker and M. K. Baltacioglu, "Fuzzy self-adaptive PID control technique for driving HHO dry cell systems," *Int. J. Hydrogen Energy*, vol. 45, no. 49, pp. 26059–26069, Oct. 2020.

[34] K. Vanchinathan and N. Selvaganesan, "Adaptive fractional order PID controller tuning for brushless DC motor using artificial bee colony algorithm," *Results Control Optim.*, vol. 4, Sep. 2021, Art. no. 100032.

[35] F.-J. Lin, H.-J. Shieh, K.-K. Shyu, and P.-K. Huang, "On-line gain-tuning IP controller using real-coded genetic algorithm," *Electric Power Syst. Res.*, vol. 72, no. 2, pp. 157–169, Dec. 2004.

[36] A. Rodríguez-Molina, M. G. Villareal-Cervantes, and M. Aldape-Pérez, "An adaptive control study for the DC motor using meta-heuristic algorithms," *Soft Comput.*, vol. 23, no. 3, pp. 889–906, 2016.

[37] D. Fister, J. Šafarič, I. Fister, R. Šafarič, and I. Fister, "Online adaptive controller based on dynamic evolution strategies," *Appl. Sci.*, vol. 8, no. 11, p. 2076, Oct. 2018.

[38] A. Rodríguez-Molina, M. G. Villarreal-Cervantes, E. Mezura-Montes, and M. Aldape-Pérez, "Adaptive controller tuning method based on online multiobjective optimization: A case study of the four-bar mechanism," *IEEE Trans. Cybern.*, vol. 51, no. 3, pp. 1–14, Mar. 2019.

[39] J. V. Carrau, G. Reynoso-Meza, S. García-Nieto, and X. Blasco, "Enhancing controller's tuning reliability with multi-objective optimisation: From model in the loop to hardware in the loop," *Eng. Appl. Artif. Intell.*, vol. 64, pp. 52–66, Sep. 2017.

[40] N. Marchand, S. Durand, and Castellanos, "A general formula for event-based stabilization of nonlinear systems," *IEEE Trans. Autom. Control*, vol. 58, no. 5, pp. 1332–1337, May 2013.

[41] M. G. Villarreal-Cervantes, J. P. Sánchez-Santana, and J. F. Guerrero-Castellanos, "Periodic event-triggered control strategy for a (3,0) mobile robot network," *ISA Trans.*, vol. 96, pp. 490–500, Jan. 2020.
[42] D. Zhang and B. Wei, "A review on model reference adaptive control of robotic manipulators," *Annu. Rev. Control*, vol. 43, pp. 188–198, Jan. 2017.
[43] L. Li, N. Wang, and H. Qin, "Adaptive model reference sliding mode control of structural nonlinear vibration," *Shock Vibrat.*, vol. 2019, Apr. 2019, Art. no. 3612516.
[44] G.-T. Tian and G.-R. Duan, "Robust model reference control for uncertain second-order system subject to parameter uncertainties," *Trans. Inst. Meas. Control*, vol. 2020, Feb. 2020 Art. no. 0142331220904544.
[45] I. Okoro, "Feedback–feedforward compensation of a DC motor," in *Proc. IEEE PES/IAS PowerAfrica*, Aug. 2019, pp. 385–389.
[46] I. Okoro and C. Enwerem, "Model-based speed control of a DC motor using a combined control scheme," in *Proc. IEEE PES/IAS PowerAfrica*, Aug. 2019, pp. 1–6.
[47] V. Lechappe, S. Rouquet, A. Gonzalez, F. Plestan, J. D. Leon, E. Moulay, and A. Glumineau, "Delay estimation and predictive control of uncertain systems with input delay: Application to a DC motor," *IEEE Trans. Ind. Electron.*, vol. 63, no. 9, pp. 5849–5857, Sep. 2016.
[48] Y. Zhi, W. Weiqing, C. Jing, and N. Razmjooy, "Interval linear quadratic regulator and its application for speed control of DC motor in the presence of uncertainties," *ISA Trans.*, to be published.
[49] I. S. Okoro and C. O. Enwerem, "Robust control of a DC motor," *Heliyon*, vol. 6, no. 12, Dec. 2020, Art. no. e05777.
[50] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 3–18, Mar. 2011.
[51] M. G. Villarreal-Cervantes and A. Rodriguez-Molina, "Speed regulation of DC motors based on on-line optimum asynchronous controller tuning and differential evolution," in *Proc. 7th Int. Conf. Control, Decis. Inf. Technol. (CoDIT)*, Jun. 2020, pp. 373–378.
[52] A. Kazikova, M. Pluhacek, and R. Senkerik, "Why tuning the control parameters of Metaheuristic algorithms is so important for fair comparison?" *Mendel*, vol. 26, no. 2, pp. 9–16, Dec. 2020.
[53] M. López-Ibáñez, J. Dubois-Lacoste, L. P. Caceres, T. Stützle, and M. Birattari, "The irace package: Iterated racing for automatic algorithm configuration," *Oper. Res. Perspect.*, vol. 3, pp. 43–58, Jan. 2016.
[54] J. Slotine and W. Li, *Applied Nonlinear Control*. Upper Saddle River, NJ, USA: Prentice-Hall, 1991.
[55] D. A. Pierre and J. W. Pierre, "Digital controller design—Alternative emulation approaches," *ISA Trans.*, vol. 34, no. 3, pp. 219–228, Oct. 1995.
[56] D. H. Wolper and W. G. Macready, "No free lunch theorems for optimization," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 67–82, Apr. 1997.
[57] E. Mezura-Montes, J. Velázquez-Reyes, and C. A. C. Coello, "A comparative study of differential evolution variants for global optimization," in *Proc. 8th Annu. Conf. Genetic Evol. Comput.*, New York, NY, USA, 2006, pp. 485–492.
[58] K. Opara and J. Arabas, "Comparison of mutation strategies in differential evolution—A probabilistic perspective," *Swarm Evol. Comput.*, vol. 39, pp. 53–69, Apr. 2018.
[59] K. Deb, "An efficient constraint handling method for genetic algorithms," *Comput. Methods Appl. Mech. Eng.*, vol. 186, nos. 2–4, pp. 311–338, 2000.
[60] E. Juarez-Castillo, H.-G. Acosta-Mesa, and E. Mezura-Montes, "Empirical study of bound constraint-handling methods in particle swarm optimization for constrained search spaces," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jun. 2017, pp. 604–611.
[61] G. Reynoso-Meza, X. Blasco, J. Sanchis, and M. Martínez, "Controller tuning using evolutionary multi-objective optimisation: Current trends and applications," *Control Eng. Pract.*, vol. 28, pp. 58–73, Jul. 2014.
[62] M. A. Arasomwan and A. O. Adewumi, "On the performance of linear decreasing inertia weight particle swarm optimization for global optimization," *Sci. World J.*, vol. 2013, pp. 1–12, Oct. 2013.
[63] M. S. Zaky, "A self-tuning PI controller for the speed control of electrical motor drives," *Electr. Power Syst. Res.*, vol. 119, pp. 293–303, Feb. 2015.
[64] Y. Xie, X. Tang, B. Song, X. Zhou, and Y. Guo, "Model-free tuning strategy of fractional-order PI controller for speed regulation of permanent magnet synchronous motor," *Trans. Inst. Meas. Control*, vol. 41, no. 1, pp. 23–35, Jan. 2019, doi: 10.1177/0142331217751040.
[65] F. V. A. Raj and V. K. Kannan, "Adaptive ELM neural computing framework with fuzzy PI controller for speed regulation in permanent magnet synchronous motors," *Soft Comput.*, vol. 24, no. 14, pp. 10963–10980, Jul. 2020.

MIGUEL GABRIEL VILLARREAL-CERVANTES (Member, IEEE) received the B.S. degree in electronics engineering from Veracruz Technologic Institute, Veracruz, Mexico, in 2003, and the M.Sc. and Ph.D. degrees in electrical engineering from the Center for Research and Advanced Studies, CINVESTAV, Mexico City, Mexico, in 2005 and 2010, respectively. He was the Head of the Mechatronic Section, CIDETEC-IPN, from 2013 to 2016, and ENRM-IPN, from 2012 to 2017. He is currently a Full Professor with the Postgraduate Department, Centro de Innovación y Desarrollo Tecnológico en Cómputo, Instituto Politécnico Nacional (CIDETEC-IPN), Mexico City. He is a member of the National System of Researchers and the Expert Network on Robotics and Mechatronics, IPN (ENRM-IPN). His current research interests include mechatronic design based on optimization (mono and multiobjective optimization), bio-inspired meta-heuristics in the optimal mechatronic design, optimal tuning for the mechatronic system control based on meta-heuristic algorithms (offline and online strategies), and robotics.

ALEJANDRO RODRÍGUEZ-MOLINA (Member, IEEE) received the B.S. degree in computer systems engineering from the Escuela Superior de Cómputo (ESCOM), Instituto Politécnico Nacional (IPN), in 2013, the M.Sc. degree in computer science from the Centro de Investigación y de Estudios Avanzados (CINVESTAV), IPN, in 2015, and the Ph.D. degree in robotics and mechatronic systems engineering from the Centro de Innovación y Desarrollo Tecnológico en Cómputo (CIDETEC), IPN, in 2019. He is currently a full-time Professor at the Research and Postgraduate Division, Instituto Tecnológico de Tlalnepantla (ITTLA), Tecnológico Nacional de México (TecNM). His research interests include the design and implementation of bio-inspired meta-heuristics for optimization and their application to engineering problems.

OMAR SERRANO-PÉREZ received the B.S. degree in electronics engineering from the Tecnológico de Estudios Superiores de Ecatepec (TESE), Estado de México, México, in 2014, and the M.Sc. degree in computer technology and the Ph.D. degree in robotics and mechatronic systems engineering from the Centro de Innovación y Desarrollo Tecnológico en Cómputo (CIDETEC), Instituto Politécnico Nacional (IPN), México City, México, in 2017 and 2021, respectively. His research interests include optimal and robust tuning for the mechatronic system control using bioinspired metaheuristics and mobile robotics.