Expert Systems With Applications 245 (2024) 123070

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Surrogate indirect adaptive controller tuning based on polynomial response surface method and bioinspired optimization: Application to the brushless direct current motor controller

Alam Gabriel Rojas-López [a], Miguel Gabriel Villarreal-Cervantes [a,*], Alejandro Rodríguez-Molina [b]

[a] Mechatronics Section, Postgraduate Department, Instituto Politécnico Nacional - Centro de Innovación y Desarrollo Tecnológico en Cómputo, Mexico City 07700, Mexico

[b] Colegio de Ciencia y Tecnología, Universidad Autónoma de la Ciudad de México, Mexico City 06720, Mexico

ARTICLE INFO

Keywords:
Online controller tuning
Indirect adaptive controller tuning
Surrogate model
Response surface method
Bioinspired optimization algorithms
BLDC motor

ABSTRACT

The increment of autonomous systems has stimulated the research of new controller tuning techniques to face the unpredictable disturbances and parametric uncertainties inherent in any autonomous system that affect its performance. The indirect adaptive controller tuning approach based on the general dynamic model (IACTA-GDM) and bioinspired optimization is one of the most successful elections facing parametric uncertainties and disturbances, which are intricate to handle by other controller tuning techniques. However, this controller tuning approach is limited by the complexity of the dynamic model due to the computational burden, restricting its application to relatively small systems or systems with slow responses where the tuning is updated at large time intervals. The present work proposes a novel surrogate indirect adaptive controller tuning approach based on the response surface method (SIACTA-RSM) to address computational burden limitations. The proposal is tested on the speed regulation controller of a brushless direct current motor, with the aim of reducing the speed regulation error and the control system's power consumption. The closed-loop system performance and the required computational time obtained by the proposed SIACTA-RSM are compared to the ones of a well-established IACTA-GDM. The descriptive and inferential statistics, as well as graphical comparisons, show that the system performance obtained by the SIACTA-RSM proposal is as competitive as the IACTA-GDM approach, keeping a mean difference among the results by up to 3.18% while reducing the computational burden of IACTA-GDM by up to 90%. These outcomes show that the SIACTA-RSM proposal is a reliable alternative to overcome the computational burden limitations that affect the IACTA-GDM approach while maintaining competitive performance.

1. Introduction

1.1. Background

In the last decades, the dependency on robotic/mechatronic systems has increased (Setchi, Dehkordi, & Khan, 2020) in many fields, making the human–robot collaboration a common trend (Arents et al., 2021; Spatola, Kühnlenz, & Cheng, 2021). In this sense, there are some cases where human work requires the aid of automated systems, like in hazardous environments (Stock-Homburg, 2022). Because of this, there is an unquestionable tendency towards autonomous expert systems (Gnambs & Appel, 2019; Kuru & Yetgin, 2019; Luckcuck, Farrell, Dennis, Dixon, & Fisher, 2019; O'Sullivan et al., 2019; Rizk, Awad, & Tunstel, 2019; Shi & Zhang, 2021), even if they are applied in safe conditions, due to their accuracy and reliability. This implies that autonomous expert systems are increasingly being implemented into more complex and demanding applications, which means that the performance of a system of this kind is not only affected by its own requirements but also by unpredictable external and internal agents (Nagy, Lăzăroiu, & Valaskova, 2023).

Despite the employed system type, its performance (accuracy, reliability, robustness, etc.) relies on the control strategy (Bubnicki, 2005; Edwards & Spurgeon, 1998; Hull, 2013; Leigh, 2004; Liu & Yao, 2016). Various control strategies have been developed to guarantee the convergence of the system error to zero in a finite time (Chen, 1998; Khalil,

* Corresponding author.
E-mail addresses: arojasl2101@alumno.ipn.mx (A.G. Rojas-López), mvillarrealc@ipn.mx (M.G. Villarreal-Cervantes), alejandro.rodriguez.molina@uacm.edu.mx (A. Rodríguez-Molina).

https://doi.org/10.1016/j.eswa.2023.123070
Received 25 August 2023; Received in revised form 3 December 2023; Accepted 26 December 2023
Available online 30 December 2023

2002). Nevertheless, the behavior of the closed-loop system response depends on the controller tuning. Controller tuning consists of finding the control parameters that satisfy the established performance requirements (Åström & Hägglund, 2006; Foley, Julien, & Copeland, 2005). This process is of utmost importance because the accuracy with which the established criteria are met depends on it (Passino, Yurkovich, & Reinfrank, 1998; Smith, 2009). There is a wide variety of controller tuning methods and different classifications of them, where one of the most popular separates them into four groups (Borase, Maghade, Sondkar, & Pawar, 2021; Rodríguez-Molina, Mezura-Montes, Villarreal-Cervantes, & Aldape-Pérez, 2020; Villarreal-Cervantes & Alvarez-Gallegos, 2016).

The first one is the analytic method, where the tuning is performed offline through mathematical tools related to control theory, where the closed-loop stability of the system is analyzed. Also, this method relies on the characteristics of the system model. If the model is linear, the Root Locus Method or Frequency Response Analysis (Bode, Nyquist, and Nichols diagrams) could be employed (Alyoussef & Kaya, 2022; Beudaert, Franco, Erkorkmaz, & Zatarain, 2020; Miyani & Sant, 2022; Osman & Zhu, 2017; Zacher, 2023; Zanasi, Cuoghi, & Ntogramatzidis, 2011). On the other hand, if it is nonlinear, approaches based on Lyapunov analysis are employed (Cai, Su, Dai, Lin, & Lin, 2009; Makhamreh, Trabelsi, Kükrer, & Abu-Rub, 2021; Wang, Liu, & Chen, 2013).

The second tuning method is based on heuristics, where the tuning is conducted offline, implementing empirical rules based on the system response (open and closed loop) and characteristics like frequency, overshoot, response time, etc. The data can be obtained experimentally or in simulation. There are different techniques in this class like Ziegler–Nichols, Cohen–Coon, Armstrong–Hagglund, Tyreus–Luyben, Ciancone–Marlin, Chien–Hrones–Reswick (Aisuwarya & Hidayati, 2019; Anitha & Gopu, 2021; Azman et al., 2017; Dang & Gostomski, 2021; Hambali, Masngut, Ishak, & Janin, 2014; Meshram & Kanojiya, 2012; Wang et al., 2019). These techniques have a mathematical background but are also influenced by the user experience.

The third tuning method considers the tuning problem as an optimization problem like the one in (1), where $\mathcal{F}(\boldsymbol{\phi}) = \{\mathrm{f}_1, \ldots, \mathrm{f}_n\}$ is the set of objective functions (that quantify the control performance of given controller parameters), g_i is the ith inequality constraint, h_j is the jth equality constraint, and l.b. and u.b. are the lower and upper boundaries of the vector of design variables $\boldsymbol{\phi}$ (controller parameters). It is worth pointing out that the optimization problem is also subject to the dynamic model of the system, which makes the objective functions and the constraints time-dependent, according to the changes in the states of the system $\boldsymbol{x}(t)$ and its control inputs $\boldsymbol{u}(t)$. Then the established optimization problem is solved offline by an optimizer (Benitez-Garcia, Villarreal-Cervantes, & Mezura-Montes, 2022; Fang, Chen, & Li, 2011; Habbi, El Houda Gabour, Bounekhla, & Boudissa, 2021; Mamizadeh, Genc, & Rajabioun, 2018; Pareek, Kishnani, & Gupta, 2014; Rojas-López, Villarreal-Cervantes, Rodríguez-Molina, & García-Mendoza, 2020; Serrano-Pérez, Villarreal-Cervantes, González-Robles, & Rodríguez-Molina, 2020; Serrano-Pérez, Villarreal-Cervantes, Rodríguez-Molina, & Serrano-Pérez, 2021; Sinha, Prasad, & Patel, 2009; Wang, Juang, & Chan, 1993; Wenge, Deyuan, Siyuan, Shaoming, & Zeyu, 2010).

$$\text{s.t.} \begin{cases} \min\limits_{\boldsymbol{\phi}\in\mathbb{R}^{n_\phi}} \mathcal{F}(\boldsymbol{\phi}) \\ \dot{x} = f(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{\phi}) \\ g_i(\boldsymbol{\phi}) \le 0, \ i = 1, \ldots, r \\ h_j(\boldsymbol{\phi}) = 0, \ j = 1, \ldots, s \\ \text{l.b.} \le \boldsymbol{\phi} \le \text{u.b.} \end{cases} \tag{1}$$

These three first controller tuning methods have their perks but share a common disadvantage. They tune the controller offline before the task is executed, which makes it essential to know all the conditions under which the system will operate. In most real-world cases, this might be complicated when they face parametric uncertainties and external disturbances. To avoid this, the fourth tuning method, known as the adaptive controller tuning method, stands out. This method is divided into direct and indirect adaptive variants. In the direct adaptive variant, an explicit reference model is employed to make comparisons against the signals of the plant, where the explicit reference model also has to be known *a priori*, making this variant as limited as the three previous controller tuning methods (Alavi, Akbarzadeh, & Farughian, 2011; Aliman, Ramli, Mohamed Haris, Soleimani Amiri, & Van, 2022; Boubakir, Labiod, & Boudjema, 2021; Chang, Hwang, & Hsieh, 2002; Gholaminejad, Khaki-Sedigh, & Bagheri, 2016; Higashiyama et al., 2000; Ortega, Praly, & Tang, 1987; Zhu, Gao, & Huang, 2022). On the other hand, the indirect adaptive variant changes the explicit reference model to an adjustable predictor, based on the general dynamic model (GDM) of the system, whose output is compared to the plant signals. Then, the predictor is updated concurrently with the controller at each time interval. This approach makes the control tunning be considered entirely online and better in the handling of parametric uncertainties and external disturbances (Rodríguez-Molina, Villarreal-Cervantes, & Aldape-Pérez, 2019; Rodríguez-Molina, Villarreal-Cervantes, Álvarez-Gallegos, & Aldape-Pérez, 2019; Rodríguez-Molina, Villarreal-Cervantes, Serrano-Pérez, Solís-Romero, & Silva-Ortigoza, 2022; Villarreal-Cervantes, Rodríguez-Molina, García-Mendoza, Peñaloza-Mejía, & Sepúlveda-Cervantes, 2017 ;Villarreal-Cervantes, Rodríguez-Molina, & Serrano-Pérez, 2021). Nevertheless, it has to be noted that indirect adaptive controller tuning has a drawback. As the controller parameters are updated at fixed time intervals through the execution of the task, the indirect method has the highest computational burden among all the remaining tuning approaches (as it computes multiple simulations iteratively within brief time intervals), limiting its application to small/simpler systems. Considering this limitation, investigations like Villarreal-Cervantes et al. (2021) have been developed to face computational burden. Such work implemented an asynchronous activation of the indirect adaptive controller tuning, limiting the computational cost for the online controller tuning process to the interval when required. Even if this approach reduces by up to 60% the computational cost related to the controller tuning, it still requires computing the dynamic model of the system, which can be very complex.

In the computer systems research field, some methods based on pattern recognition techniques have been developed to obtain a simplified system model. One of the most popular is surrogate modeling (SM), which started as a statistical tool called *meta-models* or *emulators*, which were used to work with large amounts of data (Bliek, 2022; Gramacy, 2020). These methods based on pattern recognition techniques evolved, creating different machine learning approaches to get surrogate models (Cohen, 2021). There are diverse methods to obtain the surrogate model of a complex system, such as Gaussian Process Regression, Support Vector Machines, Artificial Neural Networks, and Bayesian Linear Regression (Gramacy, 2020; Iuliano & Pérez, 2016; Koziel & Leifsson, 2013), where optimizers are implemented to approximate the outputs of the regression systems to the values of an initial data set (Jones, 2001). Moreover, one of the most popular methods is the response surface methodology (RSM) (Chelladurai et al., 2021; Jiang, Zhou, & Shao, 2020; Sobester, Forrester, & Keane, 2008) because of its simplicity in implementing a multivariate-multitarget polynomial regression. Despite these methods have been employed in a wide variety of applications like optimal design (Liu, Meng, Yuan, Ren, & Chen, 2023), safety-maintenance operations (Suganya, Swaminathan, & Anoop, 2023), vulnerability analysis (Jiang, Xia, Yao, Sun, & Xia, 2023), among others, little has been done in the controller tuning. Next, a summarization of works related to the surrogate modeling techniques applied to controller tuning methods is described.

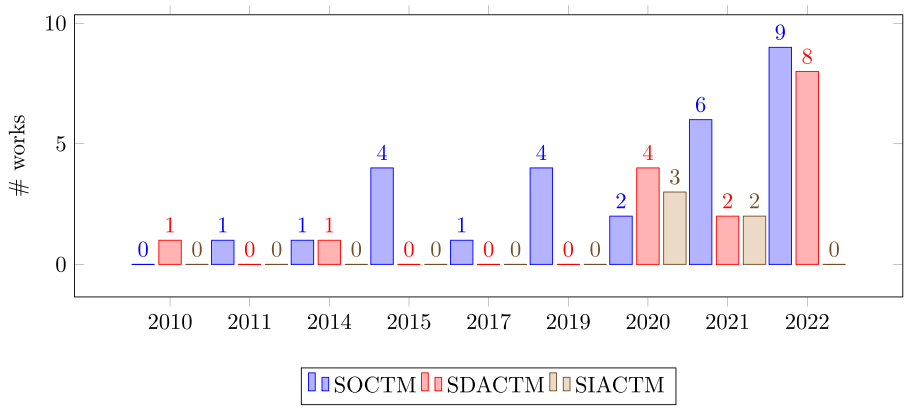


Fig. 1. Histogram of SM applied to controller tuning approaches.

1.2. State-of-the-art about surrogate models applied to controller tuning methods

Table 1 presents a compilation of works related to surrogate models applied to controller tuning. The reference work and its publication year are presented in the first column. The second column refers to the category of a proposed taxonomy related to the controller tuning method based on surrogate models: surrogate optimization-based controller tuning method (SOCTM), surrogate direct adaptive controller tuning method (SDACTM), or surrogate indirect adaptive controller tuning method (SIACTM). A more detailed description of the proposed taxonomy will be given shortly. The third column describes the kind of surrogate modeling method employed. Finally, the fourth column regards the optimization algorithm implemented, where in some of the works, this column is reported as "itself" because some surrogate model incorporates an optimization process.

The taxonomy of the second column might be complicated to categorize because many works refer to themselves as *adaptive* or *online* approaches, when they do not necessarily perform controller tuning in this way. However, the following information helps to identify their category according to all the previous information:

- Surrogate optimization-based controller tuning method (SOCTM): The optimizations are assisted/performed by surrogate models, and the optimization problem related to the controller tuning is solved offline. The controller parameters are constant during the task execution. Even if the surrogate model or optimization is obtained in a real plant, all those tests are done only to acquire fixed controller parameters.
- Surrogate direct adaptive controller tuning method (SDACTM): A surrogate model is obtained offline (even if done in a real plant) considering the whole time to be executed and sometimes different conditions applied to the experiment. The model is then used as a reference model during the adaptive controller tuning. In this approach, the controller parameters vary over the task execution in short time intervals, but the reference model remains the same during the execution.
- Surrogate indirect adaptive controller tuning method (SIACTM): The surrogate model is updated during the task execution in short time intervals, so this model dynamically changes during the task. Also, the parameters of the controller vary in these intervals. This tuning method is of interest in this work.

Analyzing the aforementioned works, a histogram is presented in Fig. 1, where the research items are divided firstly by year (since 2010) and secondly by their surrogated-based tuning approach considering the proposed taxonomy. It is observed that the strategy of controller tuning based on surrogate models clearly is a recent area of investigation, with only a few works (less than fifty) in the last decade.

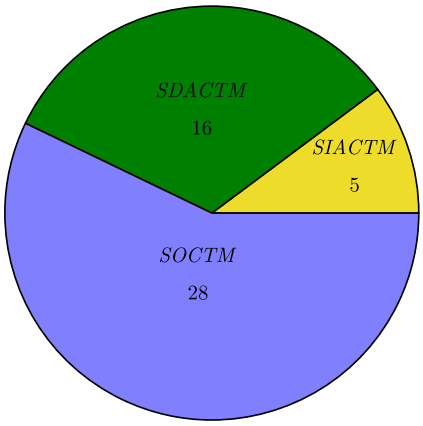


Fig. 2. Controller tuning works based on SM.

However, there is an increasing number of investigations/applications of surrogate models applied in controller tuning, particularly in the SOCTM approach. Moreover, in the last three years, there has been an interest in the SDACTM approach due to its advantages over the other approaches. Nevertheless, Fig. 2 shows that IACTM and SIACTM have been less studied than SOCTM, with less than half of the investigations (43.75%). Furthermore, there is a particular lack of research on SIACTAM (the method of interest for the present work), whose representation is only 10.41%.

For more detailed information, Fig. 3 presents the surrogate models that were applied in each tuning approach, where the surrogate models in red indicate that they have not been applied. It stands out that the kNN method has never been applied in any work about controller tuning, even though it has been applied in surrogated-assisted constrained optimization problems with acceptable results (Miranda-Varela & Mezura-Montes, 2016). Another highlight is that the Gaussian Process (GP) has been applied in 56% of the works. The GP represents more than 50% of the SOCTM and the SDACTM applications in those works. In particular, in the controller tuning method of interest for this paper, i.e., in the SIACTM, the most used surrogate model is the GP, which is observed in the 80% of the related works, while the remaining (20%) use ANN (see Fig. 3(c)). Then, to the best of the author's knowledge, the surrogate model based on RSM has not been investigated in the SIACTM.

On the other hand, it is noteworthy that ANN is the second most implemented method among surrogate model techniques with 24% of the works. This is because deep learning neural networks can achieve highly accurate surrogate models of the system, and they have shown their effectiveness in diverse applications (Brunton & Kutz, 2022; Dong,

Table 1
Surrogate model applied to controller tuning.

Reference	Tuning method	Surrogate model	Optimization algorithm
Parnianifard, Rezaie, Chaudhary, Imran, and Wuttisittikulkij (2021)	SOCTM	GP	PSO
Pan and Das (2015)	SOCTM	GP	itself
Leavy, Xu, Filizadeh, and Gole (2019)	SOCTM	GP, MARS[a]	itself
Taha, Bakr, and Emadi (2020)	SOCTM	S-M	itself
Yang, Gaida, Bäck, and Emmerich (2015)	SOCTM	GP	EGO
Faruq, Abdullah, Fauzi, and Nor (2011)	SOCTM	RBF	GA
Zhu, Piga, and Bemporad (2022)	SOCTM	RBF	GLISP
Zhao, Alimo, Beyhaghi, and Bewley (2019)	SOCTM	RSM	Δ-DOGS
Bowels, Xu, and Chen (2015)	SOCTM	GP	BO
Pirayeshshirazinezhad, Biedroń, Cruz, Güitrón, and Martínez-Ramón (2022)	SOCTM	GP, ANN, SVM	itself
Gurung, Naetiladdanon, and Sangswang (2021)	SOCTM	SVM, ANN, RF	BA
Chen and Xu (2015)	SOCTM	GP	BO
Schillinger et al. (2017)	SOCTM	GP	BO
Frasnedo et al. (2015)	SOCTM	GP	BO
Price, Radaideh, and Kochunas (2022)	SOCTM	ANN	ES, DE, PSO, GW, HH, MF
Pai, Nguyen, Prasad, and Rajendran (2022)	SOCTM	ANN	NSGA-II
Amini et al. (2022)	SOCTM	GP	GA
Kim, Lee, Kim, Park, and Lim (2022)	SOCTM	GP	Gradient Descent
McClement et al. (2022)	SOCTM	ANN	RL
Lima, Alves, and Araujo (2020)	SOCTM	GP	DACE
Hosamo, Tingstveit, Nielsen, Svennevig, and Svidt (2022)	SOCTM	ANN, SVM, GP	NSGA-II
Bhattacharya et al. (2021)	SOCTM	GP	BO
Yang, Emmerich, Deutz, and Bäck (2019)	SOCTM	GP	MOBGO
Pinto, Deltetto, and Capozzoli (2021)	SOCTM	ANN	RL
Breschi, van Meer, Oomen, and Formentin (2021)	SOCTM	GP	BO
Büchler, Calandra, and Peters (2019)	SOCTM	GP	BO
Petrusev et al. (2023)	SOCTM	ANN	RL
Antal, Péni, and Tóth (2022)	SOCTM	GP	BO
van Niekerk, le Roux, and Craig (2022)	SDACTM	GP	BO
Liu, Patton, and Shi (2022)	SDACTM	GP	BO
Sorourifar, Makrygirgos, Mesbah, and Paulson (2021)	SDACTM	GP	BO
Stenger, Ay, and Abel (2020)	SDACTM	GP	BO
Sabug, Ruiz, and Fagiano (2022)	SDACTM	SMGO-Δ[b]	itself
Fernandez-Gauna, Graña, Osa-Amilibia, and Larrucea (2022)	SDACTM	RBF	GS, RL
Zhang, Lu, et al. (2022)	SDACTM	LTI	SA
Petsagkourakis et al. (2022)	SDACTM	GP	BO, RL
Yin, Zhang, Jiang, and Pan (2020)	SDACTM	SVM	SMPSO
Kontes, Valmaseda, Giannakis, Katsigarakis, and Rovas (2014)	SDACTM	SVM	itself
Shin, Smith, and Hwang (2020)	SDACTM	ANN	LM
Yin, Zhao, Lin, and Karcanias (2020)	SDACTM	SVM	GDE-3, MOEA, MOPSO, MOGOA, NSGA-III
Martins, Rodrigues, Loureiro, Ribeiro, and Nogueira (2021)	SDACTM	ANN	PSO
Kudva, Sorouifar, and Paulson (2022)	SDACTM	GP	DRACO
Lü, Zhu, Huang, Jiang, and Jin (2010)	SDACTM	ANN	DE
Mowbray, Petsagkourakis, del Rio-Chanona, and Zhang (2022)	SDACTM	GP	RL
Roveda, Forgione, and Piga (2020a)	SIACTM	GP	BO
Chen and Cheng (2021)	SIACTM	GP	BO
Roveda, Forgione, and Piga (2020b)	SIACTM	GP	BO
Šafarič, Bencak, Fister, Šafarič, and Fister (2020)	SIACTM	ANN	ES, DE, PSO, BA
Lu, González, Kumar, and Zavala (2021)	SIACTM	GP	BO

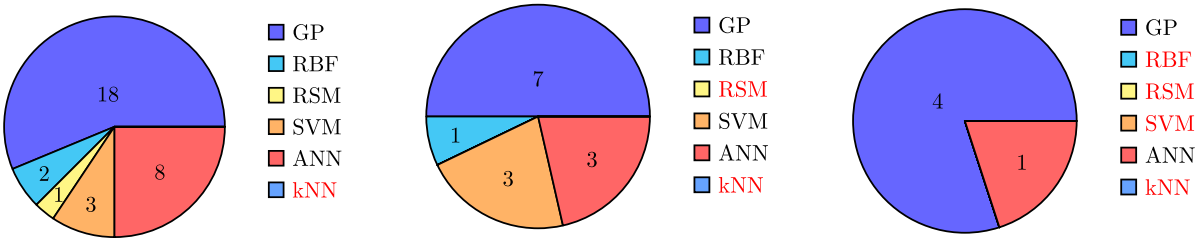
[a] RBF type.
[b] RSM type.

Wang, & Abbas, 2021; Geneva & Zabaras, 2020; Qian, Kramer, Peherstorfer, & Willcox, 2020). However, it is essential to keep in mind that the deep learning approach, specifically deep ANN, has only been applied in SOCTM and SDACTM so far. This is because the training of the ANN is performed before the system operation due to its high computational burden. In the SIACTM, attempts have been made to use deep learning techniques as surrogate models. However, only Šafarič et al. (2020) includes a single-layer ANN with a single neuron in the SIACTM, which, according to Dong et al. (2021), is not indicative of a deep learning approach because it lacks several layers and numerous neural nodes.

Some other highlights concern the optimization algorithm employed. Firstly, the BO algorithm was the most implemented with the 33.3% of applications, which may be related to the fact that BO requires fewer function evaluations (Roveda et al., 2020a). However, in Lan, Tomczak, Roijers, and Eiben (2022), it is shown that fewer function evaluations do not necessarily imply a faster solution, which is helpful in the SIACTM to provide the controller gains within the required short time intervals. Secondly, 12.5% of the works do not require an optimization algorithm, as the surrogate modeling incorporates an optimization process. Also, only three applications (6.25%) employed deterministic optimization algorithms, while the rest adopted stochastic optimization techniques.

Finally, in a more particular analysis, it is worth mentioning that the works of SIACTM analyze mostly the controller's performance, leaving aside the analysis of the computational time required to solve the online controller tuning. The computational time analysis is an essential assessment to give information about the experimental use of the SIACTM on a test-bed platform because the SIACTM must produce results within short periods (which may be a limitation in the SIACTM), resulting in a challenging task. Also, the works related to the SIACTM do not change the conditions of the experiments to analyze the limitations and characteristics of their proposals.

As can be observed from this information, the surrogate assisted controller tuning is a recent and increasing field of research. However, in a more specific way, the SIACTM is still an unaddressed topic, especially regarding using surrogate models based on the RSM method. In addition, there is a gap in formal methodologies to address the SIACTM



(a) SOCTM (b) SDACTM (c) SIACTM

Fig. 3. SM implemented by controller tuning approach.

based on RSM and bioinspired optimization, and this represents a recent research direction whose benefits and drawbacks have not been well examined.

1.3. Contributions

Considering the increase in applications of autonomous systems that require improved control techniques to face parametric uncertainties and external disturbances, this research makes significant contributions to the field of adaptive control. It addresses the current gap in state-of-the-art analysis and introduces a novel approach for the surrogate indirect adaptive controller tuning method. Next, the main contributions of the work are presented:

- **The proposal of a generalized approach for surrogate indirect adaptive controller tuning:** This work presents an innovative general approach for surrogate indirect adaptive controller tuning based on the RSM and bioinspired optimization. The proposal deals with the computational burden limitations that affect the well-established indirect adaptive controller tuning approach based on the model of the system, giving a tool to implement indirect adaptive controller tuning to expert systems. Moreover, the proposed approach can be applied to any system as long as it meets the conditions and assumptions listed in Section 2.6.
- **The application of the proposal to the BLDC motor controller and the comparative statistical study with other approaches:** Considering to demonstrate the characteristics of the proposed approach, it is applied to the Brushless Direct Current (BLDC) motor study case. This system is selected considering its increasing popularity in industrial, commercial, and research fields (Sakunthala, Kiranmayi, & Mandadi, 2017). Also, its properties, like a fast response, susceptibility to noise, and complexity of control (Mohanraj et al., 2022) make the proposed approach an interesting option for future BLDC motor applications. Moreover, the formal statistical study comparing the unaddressed proposal of SIACTM based on the RSM and the traditional IACTM based on GDM is evaluated by considering the controller performance and the computational burden related to the online controller tuning problem. This study aims to better clarify the proposal's advantages and limitations through different experiment conditions where the parametric uncertainties and disturbances are increased. This also will become an important reference for practitioners and researchers who want to adopt the proposed SIACTM in their applications by providing information about the benefits and drawbacks of the proposal, as all the methodology, characteristics, and limitations are described.

1.4. Structure of the work

The present work has the following structure. Section 2 explains the generalized form of the surrogate indirect adaptive controller tuning approach proposal, defining its form of operation, its stages, and the optimization algorithms employed. Section 3 presents the dynamic model of the BLDC motor and its controller to be optimally online tuned through the proposal, as well as the proposal's particularities for this case study. The experimentation results are portrayed in Section 4, where statistical comparisons (descriptive and inferential) regarding system performance and the computational burden are discussed, and graphical comparisons about the speed regulation and power consumption of the BLDC motor are analyzed. Finally, Section 5 discusses the conclusions and highlights of the present work.

2. Surrogate indirect adaptive controller tuning approach

The surrogate indirect adaptive control tuning implements two adaptation stages for a given instant of time at which the tuning process is performed. Fig. 4 illustrates the operation of these stages. As can be seen, different time intervals are employed in the proposed control tuning strategy. Some descriptions, as well as hints, are detailed below.

- t_p represents the present time during the execution of the system.
- ΔT_U is the update time interval. Once t_p reaches this interval, the two stages of the proposal are performed. This will occur periodically each ΔT_U.
- δt_l is the time interval related to the system's sampling. The intervals should be within a representative range. If they are too wide, they might not represent changes in the system execution. On the other hand, if they are too short, there might be an oversampling of repetitive information (approximated similar values). Also, the sensor system might not accomplish the sampling (sensing/transducing/filtering) if the intervals are too short. As a recommendation, δt_l should be shorter than ΔT_U, i.e., $\delta t_l \leq \Delta T_U$.
- T_I denotes the identification period that employs n_l samplings taken each δt_l, resulting in the period $T_I = [t_p - n_l \delta t_l, t_p]$. The samples are gathered into a vector that represents the output of the real system $\boldsymbol{y}$, which is used to update a reference model $\hat{\boldsymbol{y}} = \hat{f}(\boldsymbol{x}(t), \boldsymbol{a})$. The update of the reference model is achieved by finding the new constant parameter of the reference model $\boldsymbol{a}^*$ that makes $\hat{\boldsymbol{y}} \approx \boldsymbol{y}$. T_I period can be shorter or wider than the ΔT_U time interval, and is only limited by the quantity of information employed (based on the n_l samplings), relying more on the computational capacity of the system. This period should be wide enough to cover changes in the system but sufficiently short to compute the identification and prediction stages.
- T_P represents the prediction period that simulates the future behavior of m_l future intervals δt_l, resulting in $T_P = [t_p, t_p + m_l \delta t_l]$. The simulation of this period uses the updated reference model $\hat{\boldsymbol{y}} = \hat{f}(\boldsymbol{x}(t), \boldsymbol{a}^*, \boldsymbol{k})$ to find the new controller parameters $\boldsymbol{k}^*$ that improve a desired performance indicator of the system. Once the $\boldsymbol{k}^*$ values are calculated, they are implemented into the adjustable controller. Then, the system will operate normally until the next update tuning interval ΔT_U. Like T_I, this period should be short enough to compute the identification and prediction stages within the ΔT_U interval.

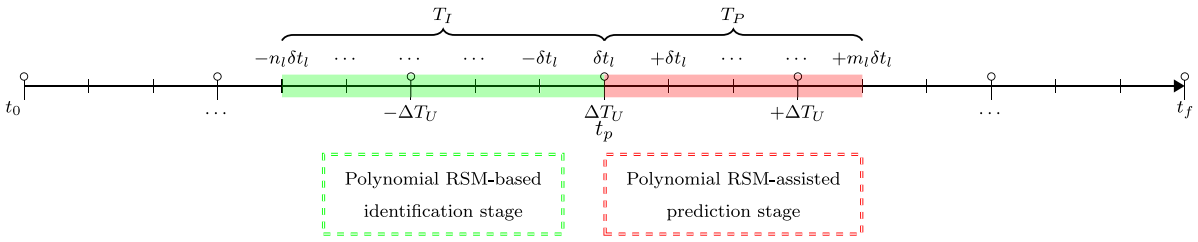


Fig. 4. Online controller tuning operation scheme.

Considering the mentioned hints, the specific description of the proposal is as follows.

Let $\boldsymbol{x} = [x_1, \ldots, x_{n_x}]^T \in \mathbb{R}^{n_x}$, $\boldsymbol{u} = [u_1, \ldots, u_{n_u}]^T \in \mathbb{R}^{n_u}$, and $\boldsymbol{y} = [y_1, \ldots, y_{n_y}]^T \in \mathbb{R}^{n_y}$ be respectively the state, input signal, and output signal vectors of the dynamic system (2), where $\mathbf{C} \in \mathbb{R}^{n_y \times n_x}$ is the relationship matrix between the states and outputs of the system.

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= f(\boldsymbol{x}(t), \boldsymbol{u}(t)) \\ \boldsymbol{y}(t) &= \mathbf{C}\boldsymbol{x}(t) \end{aligned} \tag{2}$$

Suppose that the system execution at the present time t_p has reached its update interval ΔT_U. At that moment, the adaptive mechanism of the controller will execute the following two stages.

2.1. Polynomial RSM-based identification stage

Instead of using the computationally expensive GDM approach to obtain a reference model, the Response Surface Method (RSM) (Khuri & Mukhopadhyay, 2010; Tenne & Goh, 2010; Walpole, Myers, Myers, & Ye, 1993) is implemented.

Assume that the output $\boldsymbol{y}$ of the dynamic system (2) is to be estimated by $\hat{\boldsymbol{y}} = [\hat{y}_1, \ldots, \hat{y}_k, \ldots, \hat{y}_{n_y}]^T \in \mathbb{R}^{n_y}$ through a multivariate-multitarget polynomial model. This can be achieved by RSM modeling (a surrogate model technique) (Jiang et al., 2020; Myers, Montgomery, & Anderson-Cook, 2016; Ross, 2020).

Before starting, it is important to describe some concepts. Consider that an instance (an observation) is formed by input $\tilde{\boldsymbol{x}}^l$ and output $\boldsymbol{y}^l$ attributes (input and output dynamic system's features) at a specific time, i.e., the lth instance ($l \in \{1, \ldots, n_l\}$) is the set $\{\tilde{\boldsymbol{x}}^l, \boldsymbol{y}^l\}$ at the next specific time $t_l + 1$ for the output features $\boldsymbol{y}^l$ and, at the time t_l, for the input features $\tilde{\boldsymbol{x}}^l$ (considering that $\boldsymbol{y}(t_l + 1) = \int f(\boldsymbol{x}(t_l), \boldsymbol{u}(t_l))\, dt$ is the relation between inputs and outputs features), where $\boldsymbol{y}^l = [y_1^l, \ldots, y_k^l, \ldots, y_{n_y}^l]^T \in \mathbb{R}^{n_y}$ is the vector formed by the n_y output attributes at the next specific time $t_l + 1$, and $\tilde{\boldsymbol{x}}^l = [\boldsymbol{x}^l, \boldsymbol{u}^l]^T = [x_1^l, \ldots, x_{n_x}^l, u_1^l, \ldots, u_{n_u}^l]^T = [\tilde{x}_1^l, \ldots, \tilde{x}_{n_x+n_u}^l]^T \in \mathbb{R}^{n_x+n_u}$ is the vector formed by $n_x + n_u$ input attributes at the time t_l.

Considering the aforementioned information, the polynomial model of 2nd degree that reproduces the behavior of the kth output at lth instance is given by $\check{y}_k^l$ (3), where the set $\{\beta_{k,0}, \beta_{k,i}, \beta_{k,i,j}\}$ represent the coefficients of the polynomial (Khuri, 2006).

$$\check{y}_k^l = \underbrace{\beta_{k,0}}_{bias} + \underbrace{\sum_{i=1}^{n_x+n_u} \beta_{k,i}\tilde{x}_i^l}_{\text{1st degree terms}} + \underbrace{\sum_{i=1}^{n_x+n_u}\sum_{j\geq i}^{n_x+n_u} \beta_{k,i,j}\tilde{x}_i^l\tilde{x}_j^l}_{\text{2nd degree terms}} \tag{3}$$

Fig. 5 depicts how the proposed polynomial aims to replicate the behavior of the real system by considering multiple samples/instances over a defined identification period T_I. To create a form that represents the behavior of all the n_l instances, which cover a period T_I, of each kth output, use the following:

- Consider a vector $\check{\boldsymbol{\beta}}_k \subseteq \{\beta_{k,0}, \beta_{k,i}, \beta_{k,i,j}\} = [{}^0\check{\beta}_k, {}^1\check{\beta}_k, \ldots, {}^m\check{\beta}_k, \ldots, {}^{\nu-1}\check{\beta}_k]^T \in \mathbb{R}^{\nu}$ as the coefficient vector of the polynomial (a.k.a. regressor parameters), corresponding to the kth estimated output grouped in the vector $\check{\boldsymbol{y}}_k = [\check{y}_k^1, \ldots, \check{y}_k^l, \ldots, \check{y}_k^{n_l}]^T \in \mathbb{R}^{n_l}$. It is important to consider which polynomial degree will be applied; for a 1st degree polynomial, only the independent terms are considered, giving $\nu = n_x + n_u + 1$ independent terms. On the other hand, if a 2nd degree polynomial is employed, the total terms are $\nu = \frac{(n_x+n_u+1)(n_x+n_u+2)}{2}$.
- Create an input feature vector for the lth instance, which includes a constant value, resulting in $\check{\boldsymbol{x}}^l \subseteq \{1, (\tilde{\boldsymbol{x}}^l)^T\} = [1, \check{x}_1^l, \ldots, \check{x}_{\nu-1}^l]^T \in \mathbb{R}^{\nu}$, where the elements of the vector follow that $\check{x}_i^l = \tilde{x}_i^l \ \forall\ i \in \{1, \ldots, n_x+n_u\}$ (for the independent terms) and $\{\check{x}_{n_x+n_u+1}^l, \ldots, \check{x}_{\nu-1}^l\} \subseteq \left\{\tilde{x}_i^l\tilde{x}_j^l \;\middle|\; \sum_{i=1}^{n_x+n_u}\sum_{j\geq i}^{n_x+n_u} \tilde{x}_i^l\tilde{x}_j^l\right\}$ (for the interaction terms). To clarify how an input vector is formed when a second-degree polynomial is implemented, the following example is provided: consider $n_x + n_u = 2$, the resulting lth input vector is $\check{\boldsymbol{x}}^l = [1, \check{x}_1^l, \check{x}_2^l, \check{x}_3^l, \check{x}_4^l, \check{x}_5^l]^T = [1, \tilde{x}_1^l, \tilde{x}_2^l, \tilde{x}_1^l\tilde{x}_1^l, \tilde{x}_1^l\tilde{x}_2^l, \tilde{x}_2^l\tilde{x}_2^l]^T$. Then, a matrix that groups all of the feature vectors for all n_l instances is formed as $\check{\mathbf{X}} = [(\check{\boldsymbol{x}}^1)^T, \ldots, (\check{\boldsymbol{x}}^{n_l})^T]^T \in \mathbb{R}^{n_l \times \nu}$.

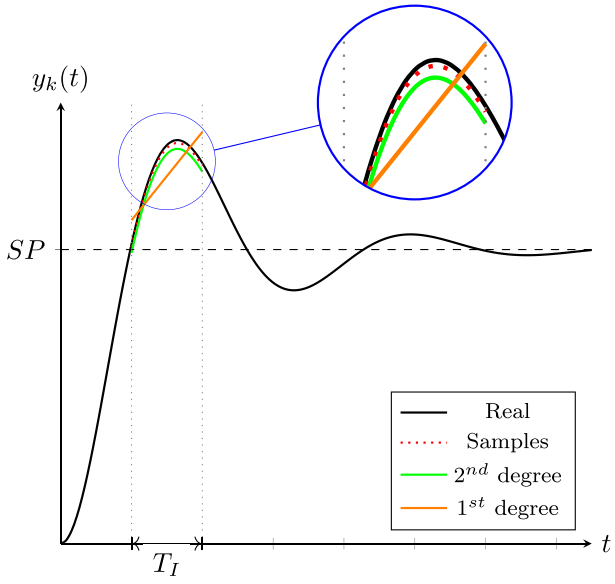


Fig. 5. Online RSM identification by polynomial regression.

Representing all n_l instances of (3) in a matrix form for the kth output regressor results in (4).

$$\check{\boldsymbol{y}}_k = \check{\mathbf{X}}\check{\boldsymbol{\beta}}_k \tag{4}$$

The general form of a regressor expressed as a multivariate-multitarget polynomial function for all of the n_y outputs simultaneously is presented in (5), where $\overline{\mathbf{X}} = \mathbf{I}_{n_y} \otimes \check{\mathbf{X}} \in \mathbb{R}^{(n_y \cdot n_l)\times(n_y \cdot \nu)}$ is the Kronecker product between an identity matrix $\mathbf{I}_{n_y} \in \mathbb{R}^{n_y \times n_y}$ and the matrix $\check{\mathbf{X}}$, resulting in a diagonal matrix formed by n_y matrices $\check{\mathbf{X}}$. Also, $\overline{\boldsymbol{\beta}} = [\check{\boldsymbol{\beta}}_1, \ldots, \check{\boldsymbol{\beta}}_k, \ldots, \check{\boldsymbol{\beta}}_{n_y}]^T \in \mathbb{R}^{n_y \cdot \nu}$ is the vector constructed by n_y coefficient vectors $\check{\boldsymbol{\beta}}_k$, and $\overline{\boldsymbol{y}} = [\check{\boldsymbol{y}}_1, \ldots, \check{\boldsymbol{y}}_k, \ldots, \check{\boldsymbol{y}}_{n_y}]^T \in \mathbb{R}^{n_y \cdot n_l}$ is the vector formed

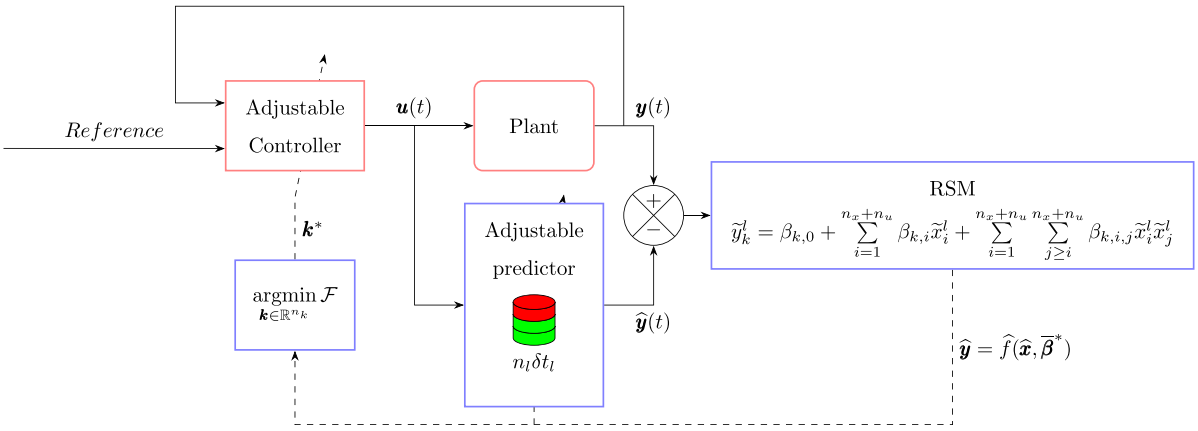


Fig. 6. Surrogate indirect adaptive controller tuning method based on RSM.

by n_y output vectors $\mathring{\boldsymbol{y}}_k$.

$$\overline{\boldsymbol{y}} = \overline{\mathbf{X}}\overline{\boldsymbol{\beta}} \tag{5}$$

To achieve a regression function that equalizes the lth instance of a kth output with the corresponding value of a real dynamic system, i.e., $\overline{y}_k^l = y_k^l$, it is necessary to find the multivariate-multitarget polynomial coefficients values ${}^m\mathring{\beta}_k \mid m \in \{0, 1, \ldots, \nu\}$ of $\overline{\boldsymbol{\beta}}$ that accomplish such equality.

The values of $\overline{\boldsymbol{\beta}}$ are calculated through the Least Squares Method (LSM) (Myers, Montgomery, Vining, & Robinson, 2012; Weisberg, 2005), which aims to minimize the squared error between the real and approximation functions, resulting in the objective function $\mathrm{J} = \frac{1}{2}(\mathring{\boldsymbol{y}} - \overline{\mathbf{X}}\overline{\boldsymbol{\beta}})^T(\mathring{\boldsymbol{y}} - \overline{\mathbf{X}}\overline{\boldsymbol{\beta}})$, where all the n_l instances sampled from all the n_y outputs of the real system are gathered in the vector $\mathring{\boldsymbol{y}} = [y_1^1, \ldots, y_1^{n_l}, \ldots, y_{n_y}^1, \ldots, y_{n_y}^{n_l}]^T \in \mathbb{R}^{n_y \cdot n_l}$.

Therefore, the optimization problem is formally stated as (6).

$$\min_{\overline{\boldsymbol{\beta}}} \mathrm{J} \tag{6}$$

To solve (6), the optimality condition of first order must satisfy i.e., $\frac{\partial \mathrm{J}}{\partial \overline{\boldsymbol{\beta}}} = 0$, which results in (7).

$$\frac{1}{2}(\mathring{\boldsymbol{y}} - \overline{\mathbf{X}}\overline{\boldsymbol{\beta}})^T(-\overline{\mathbf{X}}) + \frac{1}{2}(-\overline{\mathbf{X}})^T(\mathring{\boldsymbol{y}} - \overline{\mathbf{X}}\overline{\boldsymbol{\beta}}) = 0 \tag{7}$$

The coefficients of the regressor $\overline{\boldsymbol{\beta}}^*$ that satisfy (7) result in (8).

$$\overline{\boldsymbol{\beta}}^* = \left(\overline{\mathbf{X}}^T\overline{\mathbf{X}}\right)^{-1}\overline{\mathbf{X}}^T\overline{\boldsymbol{y}} \tag{8}$$

Once the best coefficients of the regressor $\overline{\boldsymbol{\beta}}^*$ are obtained, the final estimated output given by the regressor output $\widehat{\boldsymbol{y}} \in \mathbb{R}^{n_y}$ can be calculated with (9), where $\widehat{\boldsymbol{x}} = [1, \widehat{x}_1, \ldots, \widehat{x}_{\nu-1}]^T \in \mathbb{R}^{\nu}$ is the vector of ν terms of the polynomial regressor constructed by the input attributes.

$$\widehat{\boldsymbol{y}} = (\mathbf{I}_{n_y} \otimes \widehat{\boldsymbol{x}}^T)\overline{\boldsymbol{\beta}}^* \tag{9}$$

This final form reproduces the behavior of all the instances of $\mathring{\boldsymbol{y}}$, hence estimates the output of the dynamic system $\boldsymbol{y}(t)$ in (2), with the input attributes $\widehat{\boldsymbol{x}}$. Finally, it is worth pointing out that this method requires that the number of instances n_l must be greater than the number of terms of the polynomial ν, i.e., $n_l \geq \nu$.

2.2. Surrogate-assisted prediction stage

Once the surrogate reference model ($\widehat{\boldsymbol{y}} = \widehat{f}(\widehat{\boldsymbol{x}}, \overline{\boldsymbol{\beta}}^*)$) is obtained, it can be used as the dynamic model to simulate future behaviors over a prediction period $T_P = [t_p, t_p + m_l\delta t_l]$. This prediction is used to improve desired performances or criteria. To accomplish this, an optimization problem is stated as in (10). Where $\mathcal{F}(\boldsymbol{k}) = \{\mathrm{f}_1, \ldots, \mathrm{f}_n\}$ is the set of n objective functionals related to the controller tuning, and g_i and h_j are the ith and jth inequality and equality constraints, respectively. Also, $\boldsymbol{k} \in \mathbb{R}^{n_k}$ is the controller parameter vector of n_k elements, with its own lower and upper boundaries ($\boldsymbol{k}_{min}$ and $\boldsymbol{k}_{max}$, respectively). It is important to notice that the optimization problem is subject to the surrogate reference model rather than the general dynamic model (GDM). Finally, this optimization problem can be solved by any bioinspired algorithm, and the obtained control parameter vector $\boldsymbol{k}$ is then implemented in the adaptive controller of the plant until the next update interval $+\Delta T_U$.

$$\text{s.t.} \begin{cases} \min\limits_{\boldsymbol{k}\in\mathbb{R}^{n_k}} \mathcal{F}(\boldsymbol{k}) \\ \widehat{\boldsymbol{y}} = \widehat{f}(\widehat{\boldsymbol{x}}, \overline{\boldsymbol{\beta}}^*, \boldsymbol{k}) \\ g_i(\boldsymbol{k}) \leq 0, \ i = 0, 1, \ldots, r \\ h_j(\boldsymbol{k}) = 0, \ j = 0, 1, \ldots, s \\ \boldsymbol{k}_{min} \leq \boldsymbol{k} \leq \boldsymbol{k}_{max} \end{cases} \tag{10}$$

The form that the polynomial RSM-based identification stage and the surrogate-assisted prediction stage of the proposed approach interact is depicted in Fig. 6, where the adjustable predictor takes the information of the plant (a dataset of n_l samples gathered every δt_l) to form the final regressor used in the prediction stage to set the controller gains.

2.3. Optimizers

To solve optimization problems, it is important to consider some intuitive notions. The solution of the optimization problem might change at each update interval ΔT_U because any disturbance/uncertainty might change the problem conditions. Deterministic optimization algorithms are ineffective because the initial search point is unknown at each interval. Also, deterministic algorithms based on the gradient of the dynamic model are limited to using only continuous functions as the objective functionals and constraints of the optimization problem. Furthermore, those sorts of optimization algorithms are affected by the system's parametric uncertainties and unknown disturbances that influence the completeness of the dynamics. Hence, the optimization problem related to the indirect adaptive controller tuning is faced through bioinspired algorithms. These algorithms are unaffected by parametric uncertainties as they do not require the system's dynamic model, discrete or continuous functions do not limit them, and they do not require a specific start point but a general delimited search area. The only requirement for their application in the indirect adaptive controller tuning is to be fast enough to solve the optimization problem within the update interval ΔT_U. Some of the most used bioinspired algorithms in controller tuning problems are Differential Evolution (DE), Particle Swarm Optimization (PSO), and Genetic Algorithm (GA) (Kachitvichyanukul, 2012; Ouyang & Pano, 2015; Pano & Ouyang, 2014; Yarat, Senan, & Orman, 2021).

Consequently, the solution to the optimization problems in the surrogate-assisted prediction stage is obtained by three optimization algorithms, DE, PSO, and GA, in their online versions (Rodríguez-Molina et al., 2022) (hereinafter they will be ODE, OPSO, and OGA, respectively), where the online versions include the best solution of the update interval into the initial population of the next interval. This is done supposing that the changes between intervals do not move far away the location of the best solution. Next is a brief description of the optimization algorithms implemented.

The exact DE variant in ODE is the best/1/bin, which has been used in the online controller tuning approach (Villarreal-Cervantes, Mezura-Montes, & Guzmán-Gaspar, 2018). The description of the implemented version of ODE is presented in Algorithm 1. For this algorithm the mutation and crossover operators are shown in (11) $\forall\ j \in \{1,\dots,n_v\}$, where n_v is the number of design variables and $[F_{min}, F_{max}]$ is the mutation range from where the mutation factor F is selected randomly at each generation. Also, Cr is the crossover rate that determines the crossover probability among the population. Finally, NP and G_{max} are the population size and the maximum generations considered, respectively (these two hyperparameters are also used in the bioinspired optimizers presented below).

Algorithm 1 ODE pseudocode.

Inputs: ▷ Result from previous update interval ($\boldsymbol{\phi}_{prev}$), ▷ Maximum generations (G_{max}), ▷ Population size (NP), ▷ Boundaries range ($[\boldsymbol{\phi}_{min}, \boldsymbol{\phi}_{max}]$), ▷ Crossover rate ($Cr$), ▷ Mutation factor range ($[F_{min}, F_{max}]$), ▷ Fitness function ($fitness$)
Output: ◁ Best solution $\boldsymbol{\phi}^*$.

$G \leftarrow 1$.
Create randomly an initial population $\boldsymbol{\Phi}^G$ of $(NP-1)$ individuals, where each individual $\boldsymbol{\phi} \in [\boldsymbol{\phi}_{min}, \boldsymbol{\phi}_{max}]$.
Include $\boldsymbol{\phi}_{prev} \in \boldsymbol{\Phi}^G$.
Evaluate the $fitness$ of the individuals in $\boldsymbol{\Phi}^G$.
Get the best individual $\boldsymbol{\phi}^G_{best}$ from $\boldsymbol{\Phi}^G$.
while $G < G_{max}$ **do**
  Get randomly mutation factor $F \in [F_{min}, F_{max}]$.
  for each $\boldsymbol{\phi}^G_i \in \boldsymbol{\Phi}^G$. **do**
    Get randomly two individuals $\boldsymbol{\phi}_{r_1}$ and $\boldsymbol{\phi}_{r_2}$ from $\boldsymbol{\Phi}^G$, where $\boldsymbol{\phi}_i \neq \boldsymbol{\phi}_{r_1} \neq \boldsymbol{\phi}_{r_2}$.
    Generate an individual $\boldsymbol{\phi}_{new}$, through (11), using $\boldsymbol{\phi}^G_{best}$, $\boldsymbol{\phi}_{r_1}$ and $\boldsymbol{\phi}_{r_2}$.
    Obtain the $fitness$ of $\boldsymbol{\phi}_{new}$ by solving (10).
    Select the one to pass to $\boldsymbol{\Phi}^{G+1}$ between $\boldsymbol{\phi}_{new}$ and $\boldsymbol{\phi}^G_i$, based on their $fitness$.
  end for
  $G \leftarrow G+1$.
  Get the best individual $\boldsymbol{\phi}^G_{best}$ from $\boldsymbol{\Phi}^G$.
end while
$\boldsymbol{\phi}^* \leftarrow \boldsymbol{\phi}^G_{best}$.
return $\boldsymbol{\phi}^*$.

$$\phi_{new,j} = \begin{cases} \phi^G_{best,j} + F\left(\phi^G_{r_1,j} - \phi^G_{r_2,j}\right), & \text{if } rand(0,1) < CR \text{ or } j_{rand} = j \\ \phi^G_{i,j}, & \text{otherwise} \end{cases} \tag{11}$$

In a similar vein, the Algorithm 2 presents the implemented variant of OPSO, which uses the PSO variant with fully-connected topology and a inertial factor (Shi & Eberhart, 1998) that takes its values between a proposed range of minimum and maximum velocities, i.e., $[\boldsymbol{v}_{min,j}, \boldsymbol{v}_{max,j}]\ \forall\ j \in \{1,\dots,n_v\}$. The inertial factor is updated with (12) reducing its value at each iteration, producing a fast convergence within a small region (Yang, Yuan, Yuan, & Mao, 2007). Also, the velocity update function is in (13), and the position update function is in (14). Finally, the individual and global experience coefficients are represented by the constants C_1 and C_2, respectively.

Algorithm 2 OPSO pseudocode.

Inputs: ▷ Particle from previous update interval ($\boldsymbol{\phi}_{prev}$), ▷ Maximum generations (G_{max}), ▷ Swarm size (NP), ▷ Position boundaries range ($[\boldsymbol{\phi}_{min}, \boldsymbol{\phi}_{max}]$), ▷ Velocity boundaries range ($[\boldsymbol{v}_{min}, \boldsymbol{v}_{max}]$), ▷ Individual experience coefficient (C_1), ▷ Global experience coefficient (C_2), ▷ Fitness function ($fitness$)
Output: ◁ Best solution $\boldsymbol{\phi}^*$.

$G \leftarrow 1$.
Create randomly a swarm with positions $\boldsymbol{\Phi}^G$ and velocities $\mathbf{V}^G$ of $(NP-1)$ particles, where each particle values $\boldsymbol{\phi} \in [\boldsymbol{\phi}_{min}, \boldsymbol{\phi}_{max}]$ and $\boldsymbol{v} \in [\boldsymbol{v}_{min}, \boldsymbol{v}_{max}]$.
Include $\boldsymbol{\phi}_{prev} \in \boldsymbol{\Phi}^G$.
Evaluate the $fitness$ of the particles in $\boldsymbol{\Phi}^G$.
Create a memory of the particles' best positions $\boldsymbol{\Phi}^{own}$ and the best global position $\boldsymbol{\phi}^{global}$.
while $G < G_{max}$ **do**
  Update inertia factor $\boldsymbol{\omega}$ using (12) .
  for each $\boldsymbol{\phi}^G_i \in \boldsymbol{\Phi}^G$ **do**
    Update the velocity $\boldsymbol{v}^{G+1}_i$ using (13).
    Update the position $\boldsymbol{\phi}^{G+1}_i$ using (14).
    Obtain the $fitness$ of $\boldsymbol{\phi}^{G+1}_i$ by solving (10).
    Update $\boldsymbol{\phi}^{own}_i$ if $\boldsymbol{\phi}^{G+1}_i$ is better, based on their $fitness$.
  end for
  Update $\boldsymbol{\phi}^{global}$ if the best element of the swarm $\boldsymbol{\Phi}^{G+1}_{best}$ is better, based on their $fitness$.
  $G \leftarrow G+1$.
end while
$\boldsymbol{\phi}^* \leftarrow \boldsymbol{\phi}^{global}$.
return $\boldsymbol{\phi}^*$.

$$\omega_j = v_{max,j} - \frac{G}{G_{max}}\left(v_{max,j} - v_{min,j}\right), \forall\ j \in \{1,\dots,n_v\} \tag{12}$$

$$\boldsymbol{v}^{G+1}_i = \boldsymbol{\omega} \cdot \boldsymbol{v}^G_i + rand(0,1) \cdot C_1\left(\boldsymbol{\phi}^{own}_i - \boldsymbol{\phi}^G_i\right) + rand(0,1) \cdot C_2\left(\boldsymbol{\phi}^{global} - \boldsymbol{\phi}^G_i\right) \tag{13}$$

$$\boldsymbol{\phi}^{G+1}_i = \boldsymbol{\phi}^G_i + \boldsymbol{v}^{G+1}_i \tag{14}$$

Similarly, the Algorithm 3 presents the variant of OGA used in this work, which includes a real-coded GA with Simulated Binary Crossover (SBX) and Normally Distributed Mutation (NDM). In this implementation, the parents are selected to produce offspring through a tournament, creating a subset of T_s elements (tournament size) randomly selected from the population, whose best individual in the subset is selected as a parent for the offspring. The simulated binary crossover (SBX) function is presented in (15) and (16), with a distribution index η_c. The NDM function is in (17), where μ is the mutation factor and ${\sigma_\mu}^2$ is the variance of the distribution. This mutation variant ensures that the variation will be within the vicinity of the original individual (Kramer & Kramer, 2017; Mirjalili, 2019; Song, Wang, & Chen, 2019).

$$\begin{aligned} \boldsymbol{\psi}_{i+1} &= 0.5\left((1+\beta)\boldsymbol{\phi}_{p_1} + (1-\beta)\boldsymbol{\phi}_{p_2}\right) \\ \boldsymbol{\psi}_{i+2} &= 0.5\left((1-\beta)\boldsymbol{\phi}_{p_1} + (1+\beta)\boldsymbol{\phi}_{p_2}\right) \end{aligned} \tag{15}$$

$$\beta = \begin{cases} (2u)^{\frac{1}{\eta_c+1}}, & \text{if } u \leq 0.5 \\ \left(\frac{1}{2(1-u)}\right)^{\frac{1}{\eta_c+1}}, & \text{otherwise} \end{cases}, \text{ where } u = rand(0,1) \tag{16}$$

$$\boldsymbol{\psi}_i = \begin{cases} \boldsymbol{\psi}_i + \mathcal{N}(0, {\sigma_\mu}^2), & \text{if } rand(0,1) < \mu \\ \boldsymbol{\psi}_i, & \text{otherwise} \end{cases} \tag{17}$$

2.4. General overview of the SIACTM-RSM

The Algorithm 4 presents the general form of implementation of adaptive controller tuning proposal and how the aforementioned identification and prediction processes interact. It is observed that there

Algorithm 3 OGA pseudocode.

Inputs: ▷ Result from previous update interval ($\boldsymbol{\phi}_{prev}$), ▷ Maximum generations (G_{max}), ▷ Population size (NP), ▷ Boundaries range ($[\boldsymbol{\phi}_{min}, \boldsymbol{\phi}_{max}]$), ▷ Tournament size ($T_s$), ▷ Distribution index (η_c), ▷ Mutation rate (μ), ▷ Mutation variance ($\sigma_\mu{}^2$), ▷ Fitness function ($fitness$)
Output: ◁ Best solution $\boldsymbol{\phi}^*$.
$G \leftarrow 1$.
Create randomly an initial population $\boldsymbol{\Phi}^G$ of $(NP-1)$ individuals, where each individual $\boldsymbol{\phi} \in [\boldsymbol{\phi}_{min}, \boldsymbol{\phi}_{max}]$.
Include $\boldsymbol{\phi}_{prev} \in \boldsymbol{\Phi}^G$.
Evaluate the $fitness$ of the individuals in $\boldsymbol{\Phi}^G$.
while $G < G_{max}$ **do**
  Set an empty offspring population $\boldsymbol{\Psi}$ with a capacity of NP individuals to be filled.
  $i \leftarrow 0$.
  while $i < NP$ **do**
    Get randomly T_s individuals from $\boldsymbol{\Phi}^G$, and select the best as the 1st parent $\boldsymbol{\phi}_{p_1}$.
    Repeat the previous step to create the 2nd parent $\boldsymbol{\phi}_{p_2}$, until $\boldsymbol{\phi}_{p_1} \neq \boldsymbol{\phi}_{p_2}$.
    Perform crossover with $\boldsymbol{\phi}_{p_1}$ and $\boldsymbol{\phi}_{p_2}$, using (15) and (16), to create offspring individuals $\boldsymbol{\psi}_{i+1}$ and $\boldsymbol{\psi}_{i+2}$.
    Perform mutation of $\boldsymbol{\psi}_{i+1}$ and $\boldsymbol{\psi}_{i+2}$ using (17).
    Obtain the $fitness$ of $\boldsymbol{\psi}_{i+1}$ and $\boldsymbol{\psi}_{i+2}$ by solving (10).
    $i \leftarrow i+2$.
  end while
  Select the best NP individuals among $\boldsymbol{\Phi}^G$ and $\boldsymbol{\Psi}$ to pass to $\boldsymbol{\Phi}^{G+1}$, based on their $fitness$.
  $G \leftarrow G+1$.
end while
Get the best individual $\boldsymbol{\phi}^G_{best}$ from $\boldsymbol{\Phi}^G$.
$\boldsymbol{\phi}^* \leftarrow \boldsymbol{\phi}^G_{best}$.
return $\boldsymbol{\phi}^*$.

Algorithm 4 SIACTM based on RSM operation.

Inputs: ▷ Initial controller parameters ($\boldsymbol{k}_0$), ▷ Selected the RSM's polynomial degree *regressor* (1st or 2nd degree), ▷ Selected the *optimizer* (ODE, OGA or OPSO)
$\boldsymbol{k} \leftarrow \boldsymbol{k}_0$.
$m \leftarrow 1$.
$n \leftarrow 1$.
System operation begins.
while operation continues **do in parallel thread** (sampling)
  if $t_p = n\delta t_l$ **then**
    Get samples from system states $\boldsymbol{x}$, control signals $\boldsymbol{u}$, and system outputs $\boldsymbol{y}$.
    $n \leftarrow n+1$.
  end if
end parallel thread
while operation continues **do in parallel thread** (SIACTM)
  if $t_p = m\Delta T_U$ **then**
    Gather previous n_l samples.
    Update reference model using the *regressor*. Hence, obtain $\overline{\boldsymbol{\beta}}^*$, using (8).
    Considering $\overline{\boldsymbol{\beta}}^*$ solve (10) to obtain $\boldsymbol{k}^*$ using the *optimizer*.
    Set control parameters $\boldsymbol{k}^*$ into the adaptive controller.
    $m \leftarrow m+1$.
  end if
end parallel thread

are two parallel threads. The first one (sampling) is only dedicated to sampling the system's signals at each δt_l interval. Meanwhile, the second thread (SIACTM) implements the proposal stages. In a generic form, the implementation only requires selecting the polynomial degree of the RSM regressor and the preferred optimization algorithm. Also, initial control parameters $\boldsymbol{k}_0$ must be proposed to execute the system during enough time to gather sufficient information (n_l instances). These initial control parameters can be selected by any method or based on previous experience with the system. Remember that the proposal is focused on solving problems where parametric uncertainties and disturbances may occur during the execution of the system and fixed control parameters become unreliable. Therefore, the proposal will take these initial control parameters $\boldsymbol{k}_0$ and improve them in the next updated interval ΔT_U.

2.5. Time complexity description

As previously mentioned, the proposed surrogated indirect adaptive controller tuning approach based on the response surface method (SIACTA-RSM) consists of two stages. Considering the processes of these stages, the computational complexity can be estimated similarly to that reported in Rodríguez-Molina et al. (2023). The polynomial RSM-based identification stage consists of evaluating n_l previous instances. On the other hand, the surrogate-assisted prediction stage implements bioinspired algorithms, which perform G_{max} cycles over a population of NP individuals, and each individual employs n_v design variables to evaluate m_l future intervals. Finally, these two stages are iteratively repeated each ΔT_U update interval from initial an time t_0 to a final one t_f. Therefore, the computational complexity of the proposed tuning approach can be expressed in the *Big O* notation as in (18).

$$O\left(\left(n_l + G_{max} \cdot NP \cdot n_v \cdot m_l\right) \cdot \frac{t_f - t_0}{\Delta T_U}\right) \tag{18}$$

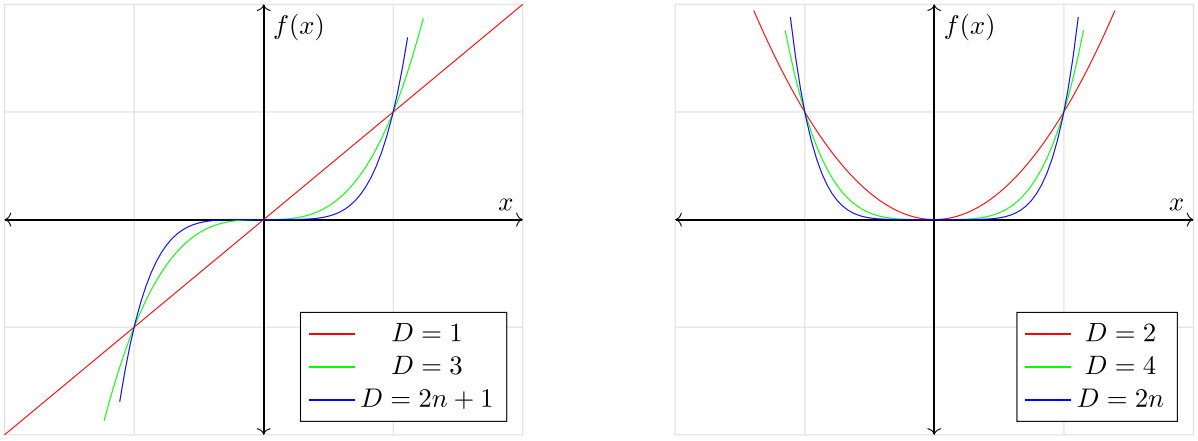
As a matter of completeness, the computational complexity of the traditional indirect adaptive controller tuning approach based on the general dynamic model (IACTA-GDM) can be expressed in the *Big O* notation as in (19). In this traditional approach, both identification and prediction stages are computed through bioinspired algorithms (Villarreal-Cervantes et al., 2017), which perform G_{max} cycles over a population of NP individuals. Each individual employs n_v^I design variables during the identification stage to evaluate n_l previous instances. Meanwhile, each individual uses n_v design variables to evaluate m_l future intervals during the prediction stage.

$$O\left(\left(G_{max} \cdot NP\left(n_v^I \cdot n_l + n_v \cdot m_l\right)\right) \cdot \frac{t_f - t_0}{\Delta T_U}\right) \tag{19}$$

2.6. Requirements, characteristics, and assumptions

Finally, the following requirements, characteristics, and assumptions about the proposed surrogated indirect adaptive controller tuning approach based on the response surface method are provided.

- The dynamic system's structure to use in the surrogate-assisted prediction stage is given by $\hat{\boldsymbol{y}}$. It will only estimate/simulate information based on the sampling considered during the identification stage. Therefore, only system states, control signal inputs, and system outputs used/sensed to create the surrogate model can be estimated.
- As the response surface method is essentially a system of equations, the number of instances n_l must be equal or greater than the number of terms of the polynomial ν in the polynomial RSM-based identification stage, i.e., $n_l \geq \nu$.
- As the polynomial RSM-based identification stage establishes a surrogate model that describes the changes between input and outputs at an established δt_l time interval (related to the sampling time), the prediction stage cannot use a different time interval. Hence, the relationship between inputs and outputs during the prediction stage is necessarily at δt_l time interval.
- Any optimization algorithm employed, besides those presented in this work, must solve the optimization problem of the surrogate-assisted prediction stage within the update time interval ΔT_U. Considering this, it is also assumed that the selected computing system has the required capacity to deal with the computational load, remembering that the proposal diminishes the



(a) Odd-degree polynomials (b) Even-degree polynomials

Fig. 7. Polynomial degree's relationship.

computational burden, giving a chance to systems with reduced capabilities.

- The proposal only considers first and second-degree polynomial regression for the response surface method, assuming that any polynomial degree has a fundamental relationship with other degrees depending on whether the degree is even or odd (Borwein & Erdélyi, 1995; Stewart, Clegg, & Watson, 2020), as can be seen in Fig. 7. The odd degrees can be considered an extension of a first-degree polynomial (see Fig. 7(a)). In contrast, the even degrees are an extension of a second-degree polynomial (see Fig. 7(b)). Therefore, the most common applications employ first or second-degree polynomial regression, evolving into greater degrees only when necessary. Furthermore, the increase in the polynomial degree does not directly imply that the regression is more accurate. An analysis with a higher degree of polynomial regression is done in the result section to confirm the previous statement.

3. Application of the SIACTM to a study case

To present an example of the proposal, the online controller tuning of a Proportional Integral (PI) controller for the speed regulation of a Brushless Direct Current (BLDC) motor will be conducted.

3.1. System and controller description

The model of the BLDC motor is obtained from Rojas-López et al. (2020), where a Proportional Integral controller is implemented. The Eq. (20) displays the mathematical model, where $\boldsymbol{x} = [x_1, x_2, x_3, x_4, x_5]^T = [i_A, i_B, i_C, \Omega, \theta]^T \in \mathbb{R}^5$ is the state vector, $\boldsymbol{a} = [a_1, a_2, a_3, a_4, a_5, a_6, a_7]^T = \left[\frac{R}{L}, \frac{k_e}{2L}, \frac{k_t}{2J}, \frac{B}{J}, \frac{1}{2L}, \frac{1}{J}, \frac{P}{2}\right]^T \in \mathbb{R}^7$ is the parameter vector of the BLDC motor, and $\mathbf{C} = \mathbf{I}_5 \in \mathbb{R}^{5\times5}$ is the identity matrix. Furthermore, the control signal $u(t)$ (the voltage input signal) is displayed in (23), where k_p and k_i are respectively the proportional and integral constants, which conform the controller parameter vector $\boldsymbol{k} = [k_p, k_i]^T \in \mathbb{R}^2$. Also, $e(t) = \Omega_d - x_4$ is the error in the speed regulation between the desired angular speed Ω_d and the real angular speed x_4. Moreover, Table 2 describes the variables employed, and Table 3 gives a more detailed description of the Back-electromagnetic force (Back-emf) and Phase voltage functions. Finally, Table 4 offers the nominal values (given by the manufacturer) of a Maxon Motor model EC 90 flat (motor 607327) used in this work.

$$\begin{aligned}\dot{\boldsymbol{x}}(t) &= \mathbf{A}\boldsymbol{x}(t) + \mathbf{B}[u(t), \tau_l]^T \\ \boldsymbol{y}(t) &= \mathbf{C}\boldsymbol{x}(t)\end{aligned} \tag{20}$$

Table 2
Model variables.

Variable	Description
$i_\gamma \forall\ \gamma \in \{A, B, C\}$	Line current
Ω, θ	Angular speed and position
R, L	Line resistance and inductance
k_e, k_t	Back-emf and torque constants
J, B	Inertia and magnetic friction coefficients
P	Pair poles
u	Voltage input
τ_l	Torque load
$f_\gamma(\theta) \forall\ \gamma \in \{A, B, C\}$	Back-emf function
$f_{V_\gamma}(\theta) \forall\ \gamma \in \{A, B, C\}$	Phase voltage function

$$\mathbf{A} = \begin{bmatrix} -a_1 & 0 & 0 & a_2 f_A(\theta) & 0 \\ 0 & -a_1 & 0 & a_2 f_B(\theta) & 0 \\ 0 & 0 & -a_1 & a_2 f_C(\theta) & 0 \\ a_3 f_A(\theta) & a_3 f_B(\theta) & a_3 f_C(\theta) & -a_4 & 0 \\ 0 & 0 & 0 & a_7 & 0 \end{bmatrix} \tag{21}$$

$$\mathbf{B} = \begin{bmatrix} a_5 f_{VA}(\theta) & 0 \\ a_5 f_{VB}(\theta) & 0 \\ a_5 f_{VC}(\theta) & 0 \\ 0 & -a_6 \\ 0 & 0 \end{bmatrix} \tag{22}$$

$$u(t) = k_p e(t) + k_i \int e(t)dt \tag{23}$$

3.2. Polynomial RSM-based identification stage characterization

The present work tackles the SIACTM by applying the proposal (9) in (20). The values of the constants required by the proposal are the following: $n_x = 5$ represents the states of the dynamic system (20), $n_u = 1$ denotes the control signal (23), and $n_y = 5$ represents the outputs of the dynamic system (20). Also, for this particular work, the polynomial RSM-based identification stage will consider $n_l = 500$ instances sampled at $\delta t_l = 10$ (μs), which results in an identification period $T_I = 5$ (ms). For this work, the identification period T_I is selected to cover all the information gathered during an update interval $\Delta T_U = 5$ (ms), which is reported in the literature (Rodríguez-Molina et al., 2022). This ensures the indirect adaptive controller tuning operation in a BLDC motor, meaning that there is enough information at this interval to perform the update of the controller parameters. The $n_l \geq \nu$ instances and the δt_l sampling time interval are selected by trial and error. The procedure to select these values was done by using a second-degree polynomial regression for the particular case of the BLDC motor, considering the

Table 3
Position dependent functions.

Angular position	Back-emf			Phase voltage		
	$f_A(\theta)$	$f_B(\theta)$	$f_C(\theta)$	$f_{V_A}(\theta)$	$f_{V_B}(\theta)$	$f_{V_C}(\theta)$
$0 \leq \theta < \frac{\pi}{3}$	1	−1	$1 - \frac{6 \cdot \theta}{\pi}$	1	−1	0
$\frac{\pi}{3} \leq \theta < \frac{2\pi}{3}$	1	$-3 + \frac{6 \cdot \theta}{\pi}$	−1	1	0	−1
$\frac{2\pi}{3} \leq \theta < \pi$	$5 - \frac{6 \cdot \theta}{\pi}$	1	−1	0	1	−1
$\pi \leq \theta < \frac{4\pi}{3}$	−1	1	$-7 + \frac{6 \cdot \theta}{\pi}$	−1	1	0
$\frac{4\pi}{3} \leq \theta < \frac{5\pi}{3}$	−1	$9 - \frac{6 \cdot \theta}{\pi}$	1	−1	0	1
$\frac{5\pi}{3} \leq \theta < 2\pi$	$-11 + \frac{6 \cdot \theta}{\pi}$	−1	1	0	−1	1

Table 4
BLDC motor values.

Parameter	Value
R	0.422 (Ω)
L	0.000535 (H)
k_e	0.207 (V s/rad)
k_t	0.231 (N m/A)
B	0.000181437 (N m s)
J	0.000506 (kg m^2)
P	11

total terms of the polynomial as $\nu = 28$ elements. Taking this into account, a δt_l of 1 (μs) was proposed initially. The mean value of the first twenty-eight samples was compared with the mean value of the next twenty-eight samples. Then, the δt_l value was increased until the mean values differed. This procedure is done to avoid an oversampling of similar and useless information.

3.3. Surrogate-assisted prediction stage characterization

In this work, $m_l = 5$ future intervals of $\delta t_l = 10$ (μs) are simulated, which results in a prediction period $T_P = 50$ (μs). The prediction period T_P is selected from Rodríguez-Molina et al. (2022), and the m_l future intervals are chosen considering one of the characteristics reported in Section 2.6, which states that the prediction stage can only predict values at the same δt_l time used for the polynomial RSM-based identification stage; hence, the future intervals are $m_l = \frac{T_P}{\delta t_l}$. Two criteria are considered for the optimization problem. The first one is the Integral Absolute Error (IAE) (24) (Mousakazemi, 2021) related to the speed regulation, where $\hat{e} = \Omega_d - \hat{y}_4$ is the error between the desired angular speed Ω_d and the angular speed simulated by the surrogate state $\hat{y}_4$. The second criterion is the power average (P_{avg}) (25) (Krishnan, 2017) consumed in the prediction simulation by the surrogate model, where $\hat{u}$ is the control signal generated to the surrogate model and $[\hat{y}_1(t), \hat{y}_2(t), \hat{y}_3(t)]^T$ is the surrogate model that represents the estimation of the phase currents i_a, i_b and i_c, respectively.

$$\mathrm{J}_1 = \int_{t \in T_P} \left|\hat{e}(t)\right| dt \tag{24}$$

$$\mathrm{J}_2 = \frac{1}{T_P} \int_{t \in T_P} \hat{u}(t) \left\| [\hat{y}_1(t), \hat{y}_2(t), \hat{y}_3(t)]^T \right\| dt \tag{25}$$

The multi-objective optimization problem in this work is transformed into a mono-objective one through the Weighted Product Method (WPM) (Marler & Arora, 2004) resulting in the functional of (26), where w_j is the weight assigned to the jth functional. This method is advantageous because it is not susceptible to different ranges in the optimization criteria (Abdullah, Siraj, & Hodgett, 2021; Aruldoss, Lakshmi, & Venkatesan, 2013; Wang, Jing, Zhang, & Zhao, 2009), whereas other methods require a normalization considering possible maximum values; in the online controller tuning approach, such information is lacking. For the proposal, the weights $\boldsymbol{w} = [0.225, 0.775]^T$ were taken from Rojas López (2020) and guarantee that the selected BLDC motor operates within its security ranges. Also, it is important to mention that these weights give less ponderation to the IAE criterion, considering that during the operation, once the steady state is reached, the error is lower than the error during the beginning of the operation.

$$\mathcal{J}_{\mathcal{T}} = \prod_{j=1}^{2} \mathrm{J}_j{}^{w_j} \tag{26}$$

Finally, the optimization problem of the prediction stage is formally presented in (27), whose upper and lower boundaries ($\boldsymbol{k}_{max} = [2, 200]^T$ and $\boldsymbol{k}_{min} = [0.5, 100]^T$, respectively) were set by previous knowledge of the system from Rojas-López et al. (2020). Another necessary clarification is that the mono-objective problem is selected instead of the multi-objective one because the latter requires a decision-making process like the used in Al-Tashi, Abdulkadir, Rais, Mirjalili, and Alhussian (2020), Chen, Ding, Yang, and Chai (2020), Lin, Liu, Tan, and Gu (2021), which means additional computational burden and tuning of the decision-making process. Therefore, the more straightforward first research step is to use a mono-objective problem.

$$\min_{\boldsymbol{k} \in \mathbb{R}^2} \mathcal{J}_{\mathcal{T}} \quad \text{s.t} \begin{cases} \hat{\boldsymbol{y}} = \hat{f}(\hat{\boldsymbol{x}}, \overline{\boldsymbol{\beta}}^*, \boldsymbol{k}) \\ \boldsymbol{k}_{min} < \boldsymbol{k} < \boldsymbol{k}_{max} \end{cases} \tag{27}$$

3.4. Online optimizers portrayal

Even if the aforementioned bioinspired algorithms have been used in similar problems, their parameters should be updated because the controller, tuning criteria, and conditions are different (Huang, Li, & Yao, 2020; Yang, 2020). Also, it is important to consider that the no-free-lunch theorem mentions that there is no single algorithm and its tuning that solves all problems satisfactorily (Joyce & Herrmann, 2018); hence, a particular tuning of the bioinspired algorithms must be performed when a new problem is faced. Considering that, during the online controller tuning, it is supposed that the user has little information about what is happening with the execution, selecting parameters for the algorithm becomes difficult. The best-proposed procedure is to use general parameters reported in similar works and the algorithm variants that use functions based on known/proposed information as the design variables' boundaries. Next is a particular description of the optimizers and their parameter selection for this case study.

- ODE: The mutation factor's range ($[F_{min}, F_{max}] = [0.29, 0.89]$) is taken from Rodríguez-Molina, Villarreal-Cervantes, and Aldape-Pérez (2019) (where the online controller tuning of a DC motor was obtained), while the crossover rate ($Cr = 0.6$) is selected by trial an error, as criteria and system to tune are different from the reference work. The population size ($NP = 20$) and the maximum generations ($G_{max} = 20$) are taken from Rojas López (2020), whose values ensure the solution within the update interval.

Table 5
Optimization algorithm parameters.

Tag	Algorithm	Parameters
ODE	Online DE best/1/bin	$NP = 20,\ G_{max} = 20,\ Cr = 0.6,\ [F_{min}, F_{max}] = [0.29, 0.89]$
OPSO	Online PSO with inertial weight	$NP = 20,\ G_{max} = 20,\ C_1 = 2,\ C_2 = 2$
OGA	Online real coded GA/SBX/NDM	$NP = 20,\ G_{max} = 20,\ T_s = 4,\ \eta_c = 2,\ \mu = 0.7$

- OPSO: The velocity boundaries are proposed as $\left[v_{min,j}, v_{max,j}\right] = \left[0, \frac{\phi_{max,j} - \phi_{min,j}}{10}\right]\ \forall\ j \in \{1, \ldots, n_v\}$. Such ranges are selected to compact the search within smaller regions. Finally, the individual experience coefficient ($C_1 = 2$) and the global experience coefficient ($C_2 = 2$) are stated following the general suggestion presented in the literature (Fan & Chiu, 2007; Guangyou, 2007; Jiao, Lian, & Gu, 2008). Finally, looking for a fair comparison, the swarm size ($NP = 20$) and maximum generations ($G_{max} = 20$) are the same as ODE.
- OGA: The distribution index ($\eta_c = 2$) is taken from Deb et al. (1995), which suggests taking values in the range $[2, 5]$ to have a more uniform distribution improving the exploration. The mutation factor $\mu = 0.7$ and the variance of the distribution $\sigma_{\mu,j}{}^2 = \frac{\phi_{max,j} - \phi_{min,j}}{10}\ \forall\ j \in \{1, \ldots, n_v\}$ assure that the variation will be within the vicinity of the original individual, which might affect the exploration, but improves the exploitation (Kramer & Kramer, 2017; Mirjalili, 2019; Song et al., 2019).

Finally, Table 5 summarizes the parameters of the optimization algorithms used in this work.

4. Experimentation

In this work, the Surrogate Indirect Adaptive Controller Tuning Approach based on the RSM regressor (SIACTA-RSM) is applied to tune a PI controller for the speed regulation of a BLDC motor. The proposal is compared with other controller tuning approaches reported in the literature and tested under four different experiments whose complexity (parametric uncertainties and disturbances) increases gradually by experiment to analyze the trade-off between the controller performance and the computational time required through experiments. The comparative study provides enough information about the advantages and limitations of the proposed SIACTA-RSM with respect to well-established controller tuning approaches (IACTA-GDM and optimization-based controller tuning method).

This section is divided into four paths. The first one set the controller tuning variants to be analyzed and the conditions of the experiments. The second one is a statistical comparison of the performance of the BLDC motor controller obtained in the experiments by each applied controller tuning version. This second path also includes a comparison of the solver time, where the solver time, in this case, is defined as the average time required by the online controller tuning approaches (SIACTA-RSM and IACTA-GDM) to finish the identification and prediction stages through the complete execution. The trade-off between the controller performance and the solver time is also analyzed to highlight the advantage of the proposal. The third path is a graphical comparison to observe/understand how much the statistical differences between SIACTA-RSM and IACTA-GDM approaches impact the behavior of the BLDC motor. The fourth path analyses the impact of including a higher degree polynomial for the regression in the proposal.

4.1. Experiment conditions

The experiment conditions for the different analyses are detailed in this section. In the proposed SIACTA-RSM, the first and second-degree regressions are considered (being named as SIACTA-RSM/1 and SIACTA-RSM/2/). Therefore, the following six versions of the proposal are implemented: SIACTA-RSM/1/ODE, SIACTA-RSM/1/OGA, SIACTA-RSM/1/OPSO, SIACTA-RSM/2/ODE, SIACTA-RSM/2/OGA, and SIACTA-RSM/2/OPSO. The first element in the used nomenclature is the controller tuning proposal, the second refers to the regressor's polynomial degree, and the third stands for the employed optimizer. The other two compared controller tunning approaches are based on the offline controller tuning and the IACTA-GDM, respectively. The first controller tuning method for the comparison is the optimum offline controller tuning (optimization-based controller tuning method) of a BLDC motor presented in Rojas-López et al. (2020), which used the same tuning criteria (IAE and P_{avg}) presented in this work. The best-reported controller gains $\boldsymbol{k} = [k_p, k_i]^T = [1.14459, 112.61539]^T$ are used, and hereinafter this controller tuning approach is called *Offline*, where these gains are fixed values and do not change over time. This offline controller tuning method (optimization-based controller tuning method) is included to observe if the SIACTA-RSM proposals keep the advantage of the indirect adaptive controller tuning approach over the optimization-based controller tuning method under parametric uncertainties and/or disturbances. The second controller tuning approach for the comparison is another indirect adaptive controller tuning approach based on the work developed in Rodríguez-Molina, Villarreal-Cervantes, and Aldape-Pérez (2017). Unlike the proposed SIACTA approach, in Rodríguez-Molina et al. (2017), a surrogate model is not implemented. Instead, well-established methodologies such as an energy-based approach (Euler–Lagrange formalism) or balancing of forces/torques (Newton–Euler formalism) are used to set the system's dynamic model. In this work, the approach in Rodríguez-Molina et al. (2017) is referred to as Indirect Adaptive Controller Tuning Approaches based on the general dynamic model (IACTA-GDM). So, the chosen IACTA-GDM versions for comparison are IACTA-GDM/ODE, IACTA-GDM/OGA, and IACTA-GDM/OPSO, where the last term in the nomenclature refers to the optimizer that is used to solve optimization problems in the tuning approach. It is important to point out that the same optimization algorithms used in the proposed SIACTA-RSM are implemented in the IACTA-GDM to make a fair comparison. The only difference is that the identification period on the IACTA-GDM approach is smaller considering only $n_l = 100$ instances sampled at intervals $\delta t_l = 10$ (μs), resulting in an identification period $T_I = 1$ (ms). The IACTA-GDM versions are included for a deeper comparison against the proposal, where the trade-off between the motor controller performance and the computational time required by the indirect adaptive approaches are evaluated. This can prove that the SIACTA-RSM proposal is a trustworthy approach when the computational burden is a limitation for the well-established indirect adaptive controller tuning approach based on the model of the system (IACTA-GDM).

To clearly understand all controller tuning versions implemented, Fig. 8 is provided, where it is observed that nine versions of the controller tuning methods are online controller tuning methods.

On the other hand, the desired angular speed is set as $\Omega_d = 40$ (rad/s), and the regulation task is performed during 0.5 (s) in four different experiments where the parametric uncertainties and disturbances gradually appear through experiments. The increment in the experiment's complexity is established due to not all systems operate in hazardous, noisy, or chaotic environments. Even though parametric uncertainties and disturbances are always present in real-world applications, their effects have different magnitudes regarding the application. Hence, different experiment complexities are tested to observe under which conditions the proposed SIACTA-RSM is as reliable as the traditional IACTA-GDM approach and to see if the SIACTA-RSM proposal keeps the advantage of the online controller tuning approaches. The conditions of the experiments are described next:

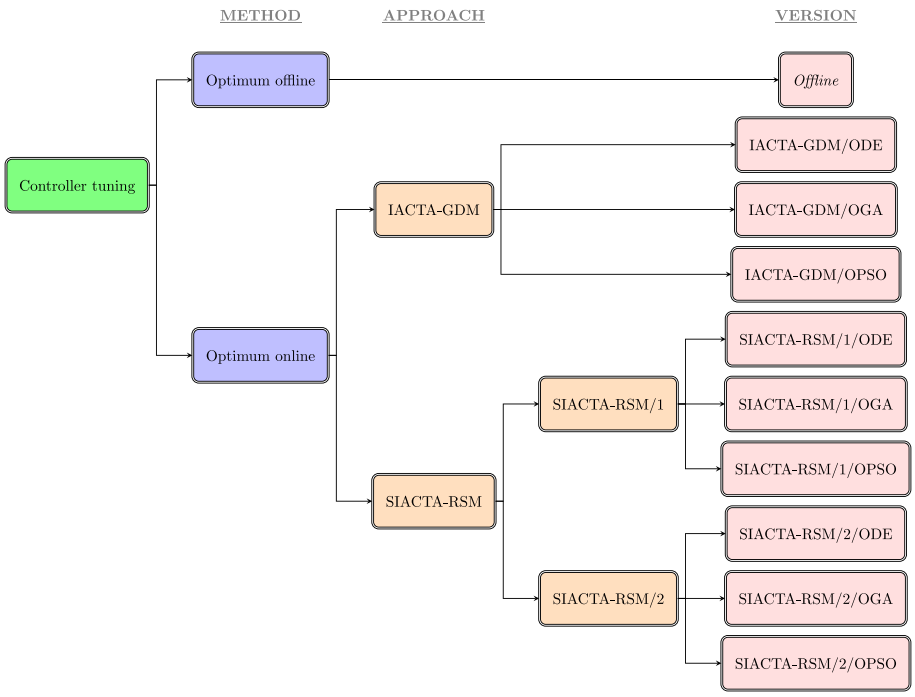


Fig. 8. Controller tuning versions implemented and compared in this work.

- Experiment 1: Considers an ideal environment without disturbances or uncertainties during the whole execution.
- Experiment 2: Considers a load τ_l applied to the motor that changes in intervals as in (28). This experiment considers disturbances in the input signal.
- Experiment 3: Considers the same load changes of (28) and also the motor's parameter vector $\boldsymbol{a}$ change continuously as in (29) $\forall\ i \in \{1,\dots,7\}$. This third experiment includes uncertainties in the system parameters and disturbances in the input signal.
- Experiment 4: Considers the same conditions of Experiment 3 and includes random noise (around $\pm 1\%$) in the sensed states of the system. The last experiment considers uncertainties in the system parameters and disturbances in the input signal and the sensing system.

$$\tau_l = \begin{cases} 1\ (\text{N m}), & \text{if } 0\ (\text{s}) \le t \le 0.1\ (\text{s}) \\ -0.6\ (\text{N m}), & \text{if } 0.1\ (\text{s}) < t \le 0.2\ (\text{s}) \\ 0.5\ (\text{N m}), & \text{if } 0.2\ (\text{s}) < t \le 0.3\ (\text{s}) \\ -0.86\ (\text{N m}), & \text{if } 0.3\ (\text{s}) < t \le 0.4\ (\text{s}) \\ 0.4\ (\text{N m}), & \text{if } 0.4\ (\text{s}) < t \le 0.5\ (\text{s}) \end{cases} \tag{28}$$

$$\mathrm{a}_i = \begin{cases} \bar{\mathrm{a}}_i(1 + 0.25\sin(20\pi t)), & \text{if } i \text{ is even} \\ \bar{\mathrm{a}}_i(1 + 0.25\cos(20\pi t)), & \text{otherwise} \end{cases} \tag{29}$$

It is worth pointing out that thirty executions are carried out for each alternative presented in Fig. 8. The simulations are executed in MATLAB version 2022b on a computer with a CPU Intel® Core™ i7-7700HQ at 2.80 (GHz) with 16 (GB) of RAM. The simulations incorporate the Euler's integration method, with an integration time of $dt = 10$ (μs).

4.2. Statistical result analysis

The statistical comparisons (descriptive and inferential) are provided in this section. Such comparisons evaluate the performance of the optimization tuning functional ($\mathcal{J}_\mathcal{T}$), the computational time required by the online controller tuning approaches, the trade-off between $\mathcal{J}_\mathcal{T}$ and the computational time, as well as the fulfillment of the tuning criteria (IAE and power P_{avg}).

Tables 6, 9, 12 and 15 present the descriptive statistics results for each experiment, where the first column corresponds to the version implemented, the next three grouped columns are the best, worst, and mean results regarding the IAE criteria of the thirty executions. After these, the next three grouped columns are the best, worst, and mean results regarding the power P_{avg} criteria of the thirty executions. Finally, the last five grouped columns are the best, worst, mean, standard deviation, and confidence intervals (95%) about the tuning functional $\mathcal{J}_\mathcal{T}$ of the thirty executions. The best results are depicted in **boldface**.

On the other hand, Tables 7, 10, 13 and 16 present inferential statistics using the Friedman multi-comparative test (Derrac, García, Molina, & Herrera, 2011) regarding the tuning functional $\mathcal{J}_\mathcal{T}$ of the online versions implemented. The implemented online controller tuning version is presented in the first column, whose obtained rank is depicted in superscript inside parentheses. The second column presents the mean Friedman rank assigned to each version. Finally, the third and fourth columns show the test statistic result and the obtained p-value, with a statistical significance $\alpha = 0.05$. In this way, if the p-value is smaller than α, the null hypothesis (which states that all the controller tuning versions are similar) is rejected.

Furthermore, a summary of the post-hoc pairwise comparisons of the Friedman tests (Derrac et al., 2011) is presented in Tables 8, 11, 14, and 17 (the comparisons include Bonferroni, Holm, and Shaffer p-value corrections (Garcia & Herrera, 2008) aiming to have a more robust analysis). In such tables, the symbols "W", "T", and "L" stand for win, tie, and lose, respectively. These symbols within the table summarize the result of comparing the row label version against the column header version. The final column sums all comparisons' wins, ties, and losses. It is worth pointing out that the summary of the post-hoc pairwise comparisons is in Appendix C.

Next, the statistical comparisons of the results is presented. An important detail is that if the full name of a specific version is mentioned, it is because that is the one with the best outcome (regarding the evaluated criterion) of its respective approach.

Table 6
Descriptive statistics results — Experiment 1.

Version	IAE			P_{avg}			$\mathcal{J}_{\mathcal{T}}$				
	best	worst	mean	best	worst	mean	best	worst	mean	std.	C.I. 95%
Offline[a]	0.1415	**0.1415**	**0.1415**	28.1400	28.1400	28.1400	–	–	–	–	–
IACTA-GDM/ODE	0.1475	0.1476	0.1475	28.0905	**28.0917**	28.0913	8.6217	8.6234	8.6229	**0.0004**	[8.623–8.623]
IACTA-GDM/OGA	0.1465	0.1475	0.1472	28.0879	28.0956	28.0905	8.6080	8.6224	8.6184	0.0029	[8.617–8.619]
IACTA-GDM/OPSO	0.1447	0.1472	0.1462	**28.0710**	28.1016	**28.0890**	8.5818	**8.6192**	**8.6050**	0.0085	**[8.602–8.608]**
SIACTA-RSM/1/ODE	0.1948	0.2930	0.2350	28.5153	29.6237	28.9493	9.2861	10.4847	9.7959	0.2595	[9.703–9.889]
SIACTA-RSM/1/OGA	0.1895	0.2899	0.2472	28.4493	29.6461	29.1082	9.2125	10.4586	9.9468	0.3328	[9.828–10.066]
SIACTA-RSM/1/OPSO	0.1745	0.2889	0.2350	28.3118	29.5535	28.9764	9.0095	10.4328	9.8005	0.3109	[9.689–9.912]
SIACTA-RSM/2/ODE	0.1422	0.2135	0.1533	28.2371	29.2369	28.3950	8.5864	9.6656	8.7616	0.2896	[8.658–8.865]
SIACTA-RSM/2/OGA	0.1421	0.2364	0.1671	28.2356	29.5483	28.4902	8.5858	9.9249	8.9464	0.4100	[8.800–9.093]
SIACTA-RSM/2/OPSO	**0.1404**	0.2188	0.1669	28.2502	29.1561	28.4872	**8.5751**	9.6975	8.9495	0.3110	[8.838–9.061]

[a] Results remain unchanged as there are no stochastic factors also $\mathcal{J}_{\mathcal{T}}$ is not considered as the tuning method is different.

4.2.1. Experiment 1

This experiment considers ideal conditions where there are no parametric uncertainties or disturbances. First, a descriptive statistical comparison is presented in Table 6 regarding the online optimization functional $\mathcal{J}_{\mathcal{T}}$ and the tuning criteria. As much information is obtained, the following highlights are provided:

- Online controller tuning approaches regarding $\mathcal{J}_{\mathcal{T}}$: The IACTA-GDM versions have better mean reported results, where IACTA-GDM/OPSO reduces the mean result of SIACTA-RSM/2/ODE by only 1.78%. It also shows that IAE is the criterion that affects the tuning performance the most. In addition, the IACTA-GDM versions have more stable results, with best means and standard deviations than the SIACTA-RSM approaches. This is reflected in the confidence intervals as IACTA-GDM versions have a lower/better operation range than SIACTA-RSM versions.
- Online vs. offline versions regarding IAE: The reported mean outcome of the thirty executions of the online controller tuning versions is worse than *Offline*'s mean result. In the particular cases, IACTA-GDM/OPSO increments it by 3.32%, and SIACTA-RSM/2/ODE by 8.33% the IAE.
- Online vs. offline versions regarding P_{avg}: Only the IACTA-GDM versions improve the mean outcome of the thirty executions. The IACTA-GDM/OPSO reduces by 0.18% the *Offline*'s mean result.
- In this experiment, where disturbances and parametric uncertainties are not considered, the *Offline* controller tuning can be considered as good as the results of the online controller. Nonetheless, real-world applications without those disturbances and uncertainties are unusual.

The above-reported information only gives a general description of the executions carried out by each version. Table 7 presents the multi-comparative Friedman test to have more representative information about the online versions' behavior for future implementations. The p-value shows the rejection of the null hypothesis, confirming that at least one of the versions differs from the rest. Before continuing with the post-hoc analysis, it is worth mentioning that the IACTA-GDM versions have the first ranks, followed by the SIACTA-RSM/2 versions, and finally, the SIACTA-RSM/1's. As the null hypothesis is rejected, a post-hoc pairwise comparison is conducted to verify the statistical differences between the online versions. For the sake of simplicity, Table 8 is created, whose values summarize the results of the post-hoc pairwise comparisons of the Friedman test. In a straightforward way to represent the results, the following highlights are given:

- The three most promising versions are SIACTA-RSM/2/ODE, IACTA-GDM/OGA, and IACTA-GDM/OPSO because there are no other tuning approaches that win over them (see the last column of Table 7) and the rest of the comparisons with those versions present non-significant statistical differences among the results (ties such that the outperformance of an algorithm is not confirmed). These results indicate that the proposed SIACTA-RSM/2/ODE performs as well as the IACTA-GDM versions when the closed-loop system does not have uncertainties and disturbances.
- In the comparisons of SIACTA-RSM/2 and IACTA-GDM versions, the 77.77% of the comparisons cannot draw a conclusion about the winner (tying 7 out of 9 comparisons). Only in two particular cases (SIACTA-RSM/2/OGA and SIACTA-RSM/2/OPSO), the IACTA-GDM outperforms the SIACTA-RSM/2 variants (see the footnote of the table). These results indicate that it is possible that the proposed SIACTA-RSM/2 can present a similar behavior to IACTA-GDM variants.
- In this experiment, the versions of SIACTA-RSM/1 were completely inefficient, losing all the comparisons against the other approaches (IACTA-GDM and SIACTA-RSM/2).

Table 7
Friedman multi-comparative test considering $\mathcal{J}_{\mathcal{T}}$ - Experiment 1.

Version	Mean Friedman rank	Test statistic	p-value
IACTA-GDM/ODE$^{(4)}$	4.1333		
IACTA-GDM/OGA$^{(2)}$	3.0667		
IACTA-GDM/OPSO$^{(1)}$	1.8333		
SIACTA-RSM/1/ODE$^{(7)}$	7.6667		
SIACTA-RSM/1/OGA$^{(9)}$	8.2333	177.1644	4.0542E−34
SIACTA-RSM/1/OPSO$^{(8)}$	7.9000		
SIACTA-RSM/2/ODE$^{(3)}$	3.2333		
SIACTA-RSM/2/OGA$^{(5)}$	4.2333		
SIACTA-RSM/2/OPSO$^{(6)}$	4.7000		

4.2.2. Experiment 2

This experiment considers an external load modified each 0.1 (s) as presented in (28). Following a similar structure, Table 9 presents the descriptive statistical result (the tuning criteria and online objective functional) of the thirty executions of each version. The following information is provided to have a quick understanding:

- Online controller tuning approaches regarding $\mathcal{J}_{\mathcal{T}}$: The IACTA-GDM versions have better mean results than the SIACTA-RSM alternatives, where IACTA-GDM/OGA reduces only 2.67% the SIACTA-RSM/2/ODE's mean result. The IACTA-GDM versions retain the best standard deviations and confidence intervals as in the previous experiment.
- Online vs. offline versions regarding IAE: All the online versions report better mean outcomes than *Offline* controller tuning mean result. The IACTA-GDM/OGA, SIACTA-RSM/2/ODE, and SIACTA-RSM/1/OPSO reduce their mean value by 26.68%, 17.55% and 15.77%, respectively, when compared to the offline versions.
- Online vs. offline versions regarding P_{avg}: None of the online versions have better results than *Offline* versions. The IACTA-GDM/OPSO, SIACTA-RSM/1/OPSO and SIACTA-RSM/2/OPSO increases the *Offline*'s mean result by 1.73%, 1.82%, and 2.32%, respectively.

Table 8
Pairwise result summary considering $\mathcal{J}_T$ - Experiment 1.

vs	IACTA-GDM/			SIACTA-RSM/1/			SIACTA-RSM/2/			Total
	ODE	OGA	OPSO	ODE	OGA	OPSO	ODE	OGA	OPSO	W/L/T*
IACTA-GDM/ODE	–	T	L	W	W	W	T	T	T	3/1/4
IACTA-GDM/OGA	T	–	T	W	W	W	T	T	T	3/0/5
IACTA-GDM/OPSO	W	T	–	W	W	W	T	W	W	6/0/1
SIACTA-RSM/1/ODE**	L	L	L	–	T	T	L	L	L	0/6/2
SIACTA-RSM/1/OGA**	L	L	L	T	–	T	L	L	L	0/6/2
SIACTA-RSM/1/OPSO**	L	L	L	T	T	–	L	L	L	0/6/2
SIACTA-RSM/2/ODE***	T	T	T	W	W	W	–	T	T	3/0/5
SIACTA-RSM/2/OGA***	T	T	L	W	W	W	T	–	T	3/1/4
SIACTA-RSM/2/OPSO***	T	T	L	W	W	W	T	T	–	3/1/4

* Total of Win/Lose/Tie between comparisons among adaptive controller tuning approaches.
** SIACTA-RSM/1 versions vs. IACTA-GDM versions - W/L/T: 0/9/0.
*** SIACTA-RSM/2 versions vs. IACTA-GDM versions - W/L/T: 0/2/7.

Table 9
Descriptive statistics results — Experiment 2.

Version	IAE			P_{avg}			$\mathcal{J}_T$				
	best	worst	mean	best	worst	mean	best	worst	mean	std.	C.I. 95%
Offline[a]	0.3800	0.3800	0.3800	**58.3479**	**58.3479**	**58.3479**	–	–	–	–	–
IACTA-GDM/ODE	0.2792	**0.2795**	0.2793	59.6587	59.6719	59.6670	17.8443	17.8512	17.8470	0.0015	[17.846–17.848]
IACTA-GDM/OGA	0.2772	0.2796	**0.2786**	59.6166	59.6883	59.6635	17.8117	**17.8498**	**17.8355**	0.0086	**[17.832–17.839]**
IACTA-GDM/OPSO	0.3098	0.3098	0.3098	59.3607	59.3607	59.3607	18.1957	18.1957	18.1957	**0.00008**	[18.196–18.196]
SIACTA-RSM/1/ODE	0.2940	0.3862	0.3350	59.1667	60.2816	59.5000	17.9364	19.3502	18.5481	0.2730	[18.450–18.646]
SIACTA-RSM/1/OGA	0.2876	0.3782	0.3221	59.1644	59.9808	59.4818	17.8660	19.1845	18.3780	0.3034	[18.269–18.487]
SIACTA-RSM/1/OPSO	0.2803	0.3658	0.3200	59.2124	59.8885	59.4099	17.7560	19.0182	18.3347	0.2677	[18.239–18.430]
SIACTA-RSM/2/ODE	**0.2499**	0.4673	0.3133	59.2692	62.1421	59.9220	**17.3150**	20.3715	18.3262	0.8770	[18.012–18.640]
SIACTA-RSM/2/OGA	0.2504	0.5218	0.3353	59.2803	61.5070	60.0151	17.3257	20.8653	18.6251	0.9265	[18.294–18.957]
SIACTA-RSM/2/OPSO	0.2680	0.4093	0.3360	59.2386	60.5689	59.7043	17.6110	19.6173	18.5985	0.4810	[18.426–18.771]

[a] Results remain equals as there are no stochastic factors also $\mathcal{J}_T$ is not considered as the tuning method is different.

Table 10 shows the Friedman test concerning the online optimization functional $\mathcal{J}_T$ for the nine online approaches implemented. As can be seen, the null hypothesis is rejected, confirming that at least one of the versions is statistically different from the rest. Before the post-hoc analysis, it is important to notice that the IACTA-GDM versions have the first ranks, followed by the SIACTA-RSM/2 variants and, finally, the SIACTA-RSM/1 alternatives.

The pairwise comparison summary is in Table 11, aiming for a clearer comparison. Some relevant observations of the results are:

- The IACTA-GDM/ODE and IACTA-GDM/OGA versions are indisputably the winners, being the only ones with several triumphed comparisons , and there are no other tuning approaches that win over them (see the last column of Table 11). The following outstanding performance is related to SIACTA-RSM/2/ODE and IACTA-GDM/OPSO because only one version outperforms them, and those outstanding versions tie with the other tuning alternatives. It is important to point out that the controller performance difference between the most outstanding version in IACTA-GDM and the second one given by the proposed SIACTA-RSM/2/ODE is at most of 2.67% (according to the mean result given in the descriptive statistics).
- In this Experiment 2, where the complexity increases by adding load disturbances, the SIACTA-RSM/2 versions reduce their competitiveness against the IACTA-GDM versions. In this case, the overall performance among those controller tuning versions indicates that SIACTA-RSM/2 alternatives lose 55.55% of the comparisons, and the rest are tied (see the table's footnote).
- The worse performance is related to SIACTA-RSM/1 versions, which lose 77.77% of the comparisons with respect to IACTA-GDM ones (see the table's footnote).

It is worth pointing out that the SIACTA-RSM and IACTA-GDM versions have better performances than the offline tuning method with the inclusion of the external load.

Table 10
Friedman multi-comparative test considering $\mathcal{J}_T$ - Experiment 2.

Version	Mean Friedman rank	Test statistic	p-value
IACTA-GDM/ODE$^{(2)}$	2.8000		
IACTA-GDM/OGA$^{(1)}$	1.8667		
IACTA-GDM/OPSO$^{(3)}$	4.5000		
SIACTA-RSM/1/ODE$^{(9)}$	7.0667		
SIACTA-RSM/1/OGA$^{(7)}$	5.9667	95.1466	4.1760E−17
SIACTA-RSM/1/OPSO$^{(5)}$	5.5667		
SIACTA-RSM/2/ODE$^{(4)}$	4.8667		
SIACTA-RSM/2/OGA$^{(6)}$	5.6667		
SIACTA-RSM/2/OPSO$^{(8)}$	6.7000		

4.2.3. Experiment 3

In this experiment, in addition to the load changes (28) of the previous experiments, the parameter vector $\boldsymbol{a}$ of the BLDC motor changes as in (29). Table 12 presents the descriptive statistical comparison of the thirty executions of each version. Now that the complexity of the experiment is more significant since additional parametric uncertainties are applied to the parameter vector, the advantage of the online controller tuning versions becomes evident, reducing the tuning criteria of the offline approach. Also, some trends are observed. The following points summarize noteworthy information:

- Online controller tuning approaches regarding $\mathcal{J}_T$: The IACTA-GDM versions have the best mean results than SIACTA-RSM versions, where IACTA-GDM/OGA diminishes only 2.97% the mean result of SIACTA-RSM/2/OPSO. Yet, the IACTA-GDM versions are more reliable as they reported the best standard deviations and confidence intervals. The difference regarding the online objective function $\mathcal{J}_T$ is also relatively small, as in Experiment 2.
- Online vs. offline versions regarding IAE: As in the previous experiment, the online versions improve the *Offline*'s mean result. The IACTA-GDM/OGA (and OPSO version), SIACTA-RSM/2/ODE

Table 11
Pairwise result summary considering $\mathcal{J}_T$ - Experiment 2.

vs	IACTA-GDM/			SIACTA-RSM/1/			SIACTA-RSM/2/			TOTAL
	ODE	OGA	OPSO	ODE	OGA	OPSO	ODE	OGA	OPSO	W/L/T*
IACTA-GDM/ODE	–	T	T	W	W	W	T	W	W	5/0/3
IACTA-GDM/OGA	T	–	W	W	W	W	W	W	W	7/0/1
IACTA-GDM/OPSO	T	L	–	W	T	T	T	T	T	1/1/6
SIACTA-RSM/1/ODE**	L	L	L	–	T	T	T	T	T	0/3/5
SIACTA-RSM/1/OGA**	L	L	T	T	–	T	T	T	T	0/2/6
SIACTA-RSM/1/OPSO**	L	L	T	T	T	–	T	T	T	0/2/6
SIACTA-RSM/2/ODE***	T	L	T	T	T	T	–	T	T	0/1/7
SIACTA-RSM/2/OGA***	L	L	T	T	T	T	T	–	T	0/2/6
SIACTA-RSM/2/OPSO***	L	L	T	T	T	T	T	T	–	0/2/6

* Total of Win/Lose/Tie between comparisons among adaptive controller tuning approaches.
** SIACTA-RSM/1 versions vs. IACTA-GDM versions - W/L/T: 0/7/2.
*** SIACTA-RSM/2 versions vs. IACTA-GDM versions - W/L/T: 0/5/4.

Table 12
Descriptive statistics results — Experiment 3.

Version	IAE			P_{avg}			$\mathcal{J}_T$				
	best	worst	mean	best	worst	mean	best	worst	mean	std.	C.I. 95%
Offline[a]	0.5873	0.5873	0.5873	64.3599	64.3599	64.3599	–	–	–	–	–
IACTA-GDM/ODE	0.4717	0.4718	0.4718	64.2821	**64.2891**	64.2852	21.2740	21.2767	21.2752	**0.0007**	[21.275–21.275]
IACTA-GDM/OGA	0.4707	**0.4717**	**0.4713**	64.2656	64.2914	**64.2805**	21.2644	**21.2742**	**21.2691**	0.0025	**[21.268–21.270]**
IACTA-GDM/OPSO	0.4705	0.4718	**0.4713**	64.2649	64.2918	64.2821	21.2604	21.2745	21.2699	0.0036	[21.269–21.271]
SIACTA-RSM/1/ODE	0.5324	0.5874	0.5715	65.6753	66.4628	66.2396	22.2278	22.8311	22.7344	0.1218	[22.691–22.778]
SIACTA-RSM/1/OGA	0.5099	0.6012	0.5660	65.4044	66.8470	66.1957	21.9417	23.1531	22.6715	0.2733	[22.574–22.769]
SIACTA-RSM/1/OPSO	0.5136	0.5790	0.5504	65.4793	66.3480	65.9324	22.0061	22.8308	22.4600	0.2550	[22.369–22.551]
SIACTA-RSM/2/ODE	0.4381	0.6528	0.5163	64.3518	67.3029	65.3620	20.9647	23.7168	21.9737	0.7167	[21.717–22.230]
SIACTA-RSM/2/OGA	**0.4359**	0.6936	0.5522	**64.2169**	67.5715	65.6505	**20.9079**	23.9621	22.3781	0.8407	[22.077–22.679]
SIACTA-RSM/2/OPSO	0.4642	0.5939	0.5289	64.2301	65.2951	64.6653	21.3350	22.6213	21.9220	0.3717	[21.789–22.055]

[a] Results remain equals as there are no stochastic factors also $\mathcal{J}_T$ is not considered as the tuning method is different.

and SIACTA-RSM/1/OPSO reduce the *Offline*'s mean result by 18.92%, 11.18% and 5.31%, respectively.

- Online vs. offline versions regarding P_{avg}: For this criterion, only the IACTA-GDM versions can reduce the mean results of the thirty executions. The IACTA-GDM/OGA reduces *Offline*'s mean result by 0.12%. In this experiment, SIACTA-RSM/2/OPSO worsens by 0.47% the *Offline*'s mean result. It is also observed that the percentage difference between versions is relatively small regarding the power P_{avg} criterion, which is near the minimum energy required by the motor to operate.

Table 13 shows another prominent trend regarding the online controller tuning functional $\mathcal{J}_T$ when the Friedman test is executed. The IACTA-GDM versions are first ranked, followed by the SIACTA-RSM/2 and SIACTA-RSM/1 versions. As can be observed, the null hypothesis is rejected, meaning that at least one of the online versions is statistically different from the rest. Therefore, a post-hoc pairwise analysis is carried out, whose results are summarized in Table 14. The following highlights are given:

- As the complexity of the experiment increases with the inclusion of disturbances and parametric uncertainties, the competitivity of the SIACTA-RSM/2 versions diminishes when compared to IACTA-GDM versions. In this case, the SIACTA-RSM/2 versions lose 77.77% of the comparisons. Even if the inferential comparison considers the IACTA-GDM alternatives as the winners, it is necessary to remember that the percentage difference among the executions regarding the tuning functional $\mathcal{J}_T$ related to the controller performance is only 2.97%.
- The SIACTA-RSM/1 versions provide the worst result. They entirely lose in all comparisons regarding IACTA-GDM variants. Considering all the outcomes mentioned above, it is evident that the second-degree regressor is a better alternative for the online controller tuning problem for a BLDC motor.

Table 13
Friedman multi-comparative test considering $\mathcal{J}_T$ - Experiment 3.

Version	Mean Friedman rank	Test statistic	p-value
IACTA-GDM/ODE$^{(3)}$	3.3333		
IACTA-GDM/OGA$^{(1)}$	1.6667		
IACTA-GDM/OPSO$^{(2)}$	2.0000		
SIACTA-RSM/1/ODE$^{(8)}$	7.9667		
SIACTA-RSM/1/OGA$^{(8)}$	7.4000	168.5600	2.5836E−32
SIACTA-RSM/1/OPSO$^{(7)}$	6.7333		
SIACTA-RSM/2/ODE$^{(4)}$	4.6000		
SIACTA-RSM/2/OGA$^{(6)}$	6.2333		
SIACTA-RSM/2/OPSO$^{(5)}$	5.0667		

4.2.4. Experiment 4

This last experiment considers a disturbance applied to the sensing/sampling of the system with a random noise signal of ±1% in addition to the previous load changes (28) and parametric uncertainties (29) for the vector $\boldsymbol{a}$ of the BLDC motor described in the previous experiment. Table 15 presents the descriptive statistical comparison of the thirty executions carried out by the implemented controller tuning versions, where the following points give notable highlights:

- Online controller tuning approaches regarding $\mathcal{J}_T$: As a common trend, the IACTA-GDM versions have better mean results, where IACTA-GDM/OPSO reduces by 3.18% the mean result of SIACTA-RSM/2/OPSO. Furthermore, it is evident that the IACTA-GDM versions have the best standard deviations and confidence intervals.
- Online vs. offline versions regarding IAE: As in the previous experiments, the online versions have better mean results than *Offline*'s mean result. In this experiment, IACTA-GDM/OPSO and SIACTA-RSM/2/ODE reduce the *Offline*'s mean value by 17.85% and 9.25%, respectively.
- Online vs. offline versions regarding P_{avg}: In this experiment, the online versions are no match against the offline one, where IACTA-GDM/ODE increases by 0.009% *Offline*'s mean result, and

Table 14
Pairwise result summary considering $\mathcal{J}_T$ - Experiment 3.

vs	IACTA-GDM/			SIACTA-RSM/1/			SIACTA-RSM/2/			TOTAL
	ODE	OGA	OPSO	ODE	OGA	OPSO	ODE	OGA	OPSO	W/L/T*
IACTA-GDM/ODE	–	T	T	W	W	W	T	W	T	4/0/4
IACTA-GDM/OGA	T	–	T	W	W	W	W	W	W	6/0/2
IACTA-GDM/OPSO	T	T	–	W	W	W	W	W	W	6/0/2
SIACTA-RSM/1/ODE**	L	L	L	–	T	T	L	T	L	0/5/3
SIACTA-RSM/1/OGA**	L	L	L	T	–	T	L	T	L	0/5/3
SIACTA-RSM/1/OPSO**	L	L	L	T	T	–	T	T	T	0/3/5
SIACTA-RSM/2/ODE***	T	L	L	W	W	T	–	T	T	2/2/4
SIACTA-RSM/2/OGA***	L	L	L	T	T	T	T	–	T	0/3/5
SIACTA-RSM/2/OPSO***	T	L	L	W	W	T	T	T	–	2/2/4

* Total of Win/Lose/Tie between comparisons among adaptive controller tuning approaches.
** SIACTA-RSM/1 versions vs. IACTA-GDM versions - W/L/T: 0/9/0.
*** SIACTA-RSM/2 versions vs. IACTA-GDM versions - W/L/T: 0/7/2.

Table 15
Descriptive statistics results — Experiment 4.

Version	IAE			P_{avg}			$\mathcal{J}_T$				
	best	worst	mean	best	worst	mean	best	worst	mean	std.	C.I. 95%
Offline[a]	0.5831	0.6099	0.5988	63.4928	65.4324	**64.3554**	–	–	–	–	–
IACTA-GDM/ODE	0.4857	**0.5026**	0.4950	63.2947	**65.2386**	64.3616	21.2222	**21.7711**	21.5260	**0.1450**	[21.474–21.578]
IACTA-GDM/OGA	0.4841	0.5122	0.4943	**63.1524**	65.5125	64.3672	21.1721	21.9105	21.5211	0.1929	**[21.452–21.590]**
IACTA-GDM/OPSO	0.4838	0.5029	**0.4919**	63.4416	65.6301	64.4506	21.2204	21.9322	**21.5193**	0.1770	[21.456–21.583]
SIACTA-RSM/1/ODE	0.5127	0.6326	0.5527	63.6605	66.6362	65.2343	21.7961	23.0436	22.2963	0.2834	[22.195–22.398]
SIACTA-RSM/1/OGA	0.5101	0.6182	0.5567	64.3892	66.5622	65.4209	21.7715	23.0908	22.3812	0.3009	[22.274–22.489]
SIACTA-RSM/1/OPSO	0.5300	0.6111	0.5562	64.3515	66.5935	65.3828	21.8569	22.9703	22.3678	0.2513	[22.278–22.458]
SIACTA-RSM/2/ODE	**0.4434**	0.7922	0.5434	64.2700	68.7417	65.8520	**21.1015**	25.1391	22.3478	0.9627	[22.003–22.692]
SIACTA-RSM/2/OGA	0.4522	0.8462	0.5588	64.2892	68.4810	65.7111	21.1058	25.4832	22.4462	0.9546	[22.105–22.788]
SIACTA-RSM/2/OPSO	0.4910	0.6178	0.5536	63.6688	66.8484	64.9630	21.5677	22.9450	22.2279	0.4266	[22.075–22.381]

[a] $\mathcal{J}_T$ is not considered as the tuning method is different.

SIACTA-RSM/2/OPSO increases it by 0.99%. The relatively small difference regarding the results of this criterion is attributed to the fact that the optimization problem reaches the point of the minimum energy required by the motor to operate.

In this experiment, where the conditions raise its complexity, the inferential statistical comparison of the Friedman test regarding $\mathcal{J}_T$ shows another interesting highlight. Even if the order of the ranks remains (first IACTA-GDM versions, second SIACTA-RSM/2 variants, and third SIACTA-RSM/1 alternatives), the difference between IACTA-GDM and SIACTA-RSM ones is more significant. This is confirmed as the null hypothesis is rejected. The post-hoc pairwise analysis is carried out and summarized in Table 17 to understand the meaning of the increment of the ranks. The following noteworthy points are presented:

- The SIACTA-RSM versions lose all the comparisons against the IACTA-GDM versions. Nevertheless, according to the descriptive statistics, the difference between the mean performance (regarding the tuning functional $\mathcal{J}_T$) of the controller tuning versions is not as significant (around 3.18%). This confirms that, despite showing IACTA-GDM a superior performance in the inferential statistical comparison, the difference in the mean value is relatively small.

4.2.5. General discussion of the controller performance through the four experiments

Considering the results obtained through the previous four experiments, it is observed that the controller performance given by the SIACTA-RSM proposal (specifically SIACTA-RSM/2/ODE) can be as competitive as the well-established IACTA-GDM when there are no parametric uncertainties and disturbances (based on Experiment 1). The closed-loop controller performance of the SIACTA-RSM proposal decreases as parametric uncertainties and disturbances increase. Specifically, the increment of the loss percentages in the comparisons against

Table 16
Friedman multi-comparative test considering $\mathcal{J}_T$ - Experiment 4.

Version	Mean Friedman rank	Test statistic	p-value
IACTA-GDM/ODE(1)	2.3333		
IACTA-GDM/OGA(2)	2.4000		
IACTA-GDM/OPSO(3)	2.4667		
SIACTA-RSM/1/ODE(7)	6.3667		
SIACTA-RSM/1/OGA(9)	6.9000	126.2577	1.6857E−23
SIACTA-RSM/1/OPSO(8)	6.8000		
SIACTA-RSM/2/ODE(4)	5.7000		
SIACTA-RSM/2/OGA(6)	6.0333		
SIACTA-RSM/2/OPSO(5)	6.0000		

SIACTA-RSM are 55.55%, 77.77%, and 100% from the second to the fourth experiments with respect to IACTA-GDM. This discrepancy is because the IACTA-GDM uses a more accurate yet highly complex plant model to emulate the closed-loop system's behavior in the predictive stage. Nevertheless, the controller performance difference between the proposed SIACTA-RSM and the IACTA-GDM is relatively small, for instance, 1.78%, 2.67%, 2.97% and 3.18% through the four experiments. The higher difference of the mean results during the experiments is only 3.18%. However, the proposal has the benefit of requiring less computational load, which is an essential factor in implementing autonomous systems. The computational load analysis is described in the next section to confirm this advantage.

4.2.6. Computational time analysis related to the indirect adaptive control tuning approaches

Another crucial criterion in the online controller tuning approaches is the computational burden related to the time consumed by the indirect adaptive tuning approaches for the update intervals. Considering that the time for each approach is not related to the conditions of the experiment but to the approach itself, the results of only the first experiment are presented (the times are similar in all experiments). In this work, the average time to solve the identification and prediction

Table 17
Pairwise result summary considering $\mathcal{J}_T$ - Experiment 4.

vs	IACTA-GDM/			SIACTA-RSM/1/			SIACTA-RSM/2/			TOTAL
	ODE	OGA	OPSO	ODE	OGA	OPSO	ODE	OGA	OPSO	W/L/T*
IACTA-GDM/ODE	–	T	T	W	W	W	W	W	W	6/0/2
IACTA-GDM/OGA	T	–	T	W	W	W	W	W	W	6/0/2
IACTA-GDM/OPSO	T	T	–	W	W	W	W	W	W	6/0/2
SIACTA-RSM/1/ODE	L	L	L	–	T	T	T	T	T	0/3/5
SIACTA-RSM/1/OGA	L	L	L	T	–	T	T	T	T	0/3/5
SIACTA-RSM/1/OPSO	L	L	L	T	T	–	T	T	T	0/3/5
SIACTA-RSM/2/ODE	L	L	L	T	T	T	–	T	T	0/3/5
SIACTA-RSM/2/OGA	L	L	L	T	T	T	T	–	T	0/3/5
SIACTA-RSM/2/OPSO	L	L	L	T	T	T	T	T	–	0/3/5

* Total of Win/Lose/Tie between comparisons among adaptive controller tuning approaches.
** SIACTA-RSM/1 versions vs. IACTA-GDM versions - W/L/T: 0/9/0.
*** SIACTA-RSM/2 versions vs. IACTA-GDM versions - W/L/T: 0/9/0.

Table 18
Descriptive statistics — Solver time.

Version	Solver time (s)				
	best	worst	mean	std.	C.I.
IACTA-GDM/ODE	0.1082	0.1487	0.1407	0.0088	[0.138–0.144]
IACTA-GDM/OGA	0.1255	0.1862	0.1666	0.0127	[0.162–0.171]
IACTA-GDM/OPSO	0.0978	0.1328	0.1253	0.0075	[0.123–0.128]
SIACTA-RSM/1/ODE	0.0140	**0.0198**	0.0177	**0.0015**	[0.0172–0.0182]
SIACTA-RSM/1/OGA	0.0187	0.0359	0.0238	0.0045	[0.0222–0.0254]
SIACTA-RSM/1/OPSO	**0.0094**	0.0215	**0.0124**	0.0033	**[0.0112–0.0136]**
SIACTA-RSM/2/ODE	0.0220	0.0323	0.0298	0.0022	[0.0290–0.0306]
SIACTA-RSM/2/OGA	0.0267	0.0458	0.0339	0.0048	[0.0322–0.0356]
SIACTA-RSM/2/OPSO	0.0193	0.0289	0.0247	0.0023	[0.0239–0.0255]

Table 19
Friedman multi-comparative test considering solver time.

Version	Mean Friedman rank	Test statistic	p-value
IACTA-GDM/ODE$^{(8)}$	7.8000		
IACTA-GDM/OGA$^{(9)}$	9.0000		
IACTA-GDM/OPSO$^{(7)}$	7.2000		
SIACTA-RSM/1/ODE$^{(1)}$	1.5000		
SIACTA-RSM/1/OGA$^{(5)}$	4.7000	232.8355	7.4395E−46
SIACTA-RSM/1/OPSO$^{(1)}$	1.5000		
SIACTA-RSM/2/ODE$^{(4)}$	4.0333		
SIACTA-RSM/2/OGA$^{(6)}$	6.0000		
SIACTA-RSM/2/OPSO$^{(3)}$	3.2667		

stages during a whole execution is used to evaluate the computational time of the indirect adaptive control tuning approaches. From now on, the computational time will be called "Solver time". Table 18 displays the descriptive statistical results of the thirty executions, where the first column describes the versions tested, and their best and worst results are in the second and third columns, respectively. Also, the mean, standard deviation, and confidence intervals (at 95%) are in the fourth, fifth, and sixth columns, respectively. The best results of each column are in **boldface**. The following highlights of Table 18 are provided:

- The SIACTA-RSM versions are the fastest approaches. The SIACTA-RSM/1/OPSO reduces the mean time of the best-reported version of IACTA-GDM (IACTA-GDM/OPSO) in 90.10% and the SIACTA-RSM/2/ODE diminishes it in 80.28%.
- It is observed that there are significant differences in the mean time when the optimizers change in the same controller tuning approach. In the particular case of SIACTA-RSM/1 versions, the SIACTA-RSM/1/OPSO and SIACTA-RSM/1/ODE diminish the mean result with reference to SIACTA-RSM/1/OGA by 29.94% and 47.89%, respectively. Likewise, in SIACTA-RSM/2 alternatives, the SIACTA-RSM/2/OPSO and SIACTA-RSM/2/ODE decreases the mean result regarding SIACTA-RSM/2/OGA by 17.11% and 27.13%, respectively. In the case of the IACTA-GDM approach, the IACTA-GDM/OPSO and IACTA-GDM/ODE reduce the mean result with respect to IACTA-GDM/OGA by 10.94% and 24.78%, respectively.

A Friedman test regarding the solver time is carried out to make inferences about the online controller tuning versions implemented. The results are in Table 19, where it can be seen that the null hypothesis is rejected, which means that at least one of the versions compared is statistically different from the rest. Before continuing with a post-hoc analysis, it is noteworthy that the SIACTA-RSM/1 versions have the first ranks (except for SIACTA-RSM/1/OGA), followed by the SIACTA-RSM/2 variants, and the IACTA-GDM alternatives. For the sake of simplicity, Table 20 presents the summary of the pairwise post-hoc analysis, whose complete results are in Table C.30. The following outstanding points of Table 20 are given:

- The SIACTA-RSM/1 versions win all the comparisons against the IACTA-GDM ones, i.e., SIACTA-RSM/1 versions consume less computational load. The next competitive tuning approach is related to SIACTA-RSM/2 versions, with seven wins and only two ties with reference to the IACTA-GDM alternatives (the ties only happen when SIACTA-RSM/2/OGA is implemented). IACTA-GDM versions provide the worst result regarding the solver time analysis.
- Despite the SIACTA-RSM versions of the proposal consider a wider identification period ($T_I = 5$ (ms)) than the given in IACTA-GDM variants ($T_I = 1$ (ms)), the computational time of the proposed SIACTA-RSM is sufficiently reduced with respect to the traditional IACTA-GDM approach.
- The OGA optimizer is considered the slower one, as it is the only one with losses when it is compared against similar online controller tuning approaches. This is attributed to the tournament selection step in Algorithm 3 to choose individuals for offspring creation. This process requires more time than only selecting them randomly.

4.2.7. Online control tuning performance vs. solver time analysis

Previously, the closed-loop performance and average time required to solve the adaptive stages of the online controller tuning approaches have been evaluated separately. However, their implementation is a unique element with its benefits and weaknesses. Hence, the following weighted sum method evaluation is developed. In this case, the mean Friedman's ranks of both the optimization problem performance $\mathcal{J}_T$ and the solver time through experiments are equally weighted to form one criterion and to compare them fairly. Therefore, the weights assigned to the performances obtained in the four experiments are proportional and equally distributed, where $\omega_i = 0.125$ is the weight assigned to the ith experiment $\forall\, i \in 1, \ldots, 4$. This distribution contributes to a total weight of 0.5 for the closed-loop experiment performances. Additionally, a weight of $\omega_5 = 0.5$ is assigned to the solver time.

Table 20
Pairwise result summary considering solver time.

vs	IACTA-GDM/			SIACTA-RSM/1/			SIACTA-RSM/2/			TOTAL
	ODE	OGA	OPSO	ODE	OGA	OPSO	ODE	OGA	OPSO	W/L/T*
IACTA-GDM/ODE	–	T	T	L	L	L	L	T	L	0/5/3
IACTA-GDM/OGA	T	–	T	L	L	L	L	L	L	0/6/2
IACTA-GDM/OPSO	T	T	–	L	L	L	L	T	L	0/5/3
SIACTA-RSM/1/ODE**	W	W	W	–	W	T	W	W	T	6/0/2
SIACTA-RSM/1/OGA**	W	W	W	L	–	L	T	T	T	3/2/3
SIACTA-RSM/1/OPSO**	W	W	W	T	W	–	W	W	T	6/0/2
SIACTA-RSM/2/ODE***	W	W	W	L	T	L	–	T	T	3/2/3
SIACTA-RSM/2/OGA***	T	W	T	L	T	L	T	–	L	1/3/4
SIACTA-RSM/2/OPSO***	W	W	W	T	T	T	T	W	–	4/0/4

* Total of Win/Lose/Tie between comparisons among adaptive controller tuning approaches.
** SIACTA-RSM/1 versions vs. IACTA-GDM versions - W/L/T: 9/0/0.
*** SIACTA-RSM/2 versions vs. IACTA-GDM versions - W/L/T: 7/0/2.

Table 21
Weighted sum method analysis of Friedman's ranks about experiment performance and solver times.

Version	Mean Friedman's ranks					Version total	Approach total
	Experiment 1 $\omega_1 = 0.125$	Experiment 2 $\omega_2 = 0.125$	Experiment 3 $\omega_3 = 0.125$	Experiment 4 $\omega_4 = 0.125$	Solver time $\omega_5 = 0.5$		
IACTA-GDM/ODE	4.1333	2.8000	3.3333	2.3333	7.8000	5.4750	
IACTA-GDM/OGA	3.0667	1.8667	1.6667	2.4000	9.0000	5.6250	16.0500
IACTA-GDM/OPSO	1.8333	4.5000	2.0000	2.4667	7.2000	4.9500	
SIACTA-RSM/1/ODE	7.6667	7.0667	7.9667	6.3667	1.5000	4.3834	
SIACTA-RSM/1/OGA	8.2333	5.9667	7.4000	6.9000	4.7000	5.9125	14.4209
SIACTA-RSM/1/OPSO	7.9000	5.5667	6.7333	6.8000	1.5000	4.1250	
SIACTA-RSM/2/ODE	3.2333	4.8667	4.6000	5.7000	4.0333	4.3167	
SIACTA-RSM/2/OGA	4.2333	5.6667	6.2333	6.0333	6.0000	5.7708	14.5292
SIACTA-RSM/2/OPSO	4.7000	6.7000	5.0667	6.0000	3.2667	4.4417	

The data of this evaluation is shown in Table 21. This table shows in its first column the online controller tuning versions implemented. The following four columns include the mean Friedman's ranks obtained by each experiment regarding the performance of the optimization problem $\mathcal{J}_{\mathcal{T}}$. The sixth column shows the mean Friedman's ranks regarding the solver time. The seventh column presents the total of the weighted sum. Finally, the eighth column indicates the total results of each approach (GDM, SIACTA-RSM/1, and SIACTA-RSM/2). The following noteworthy points are given:

- If the controller performance and the solver time are simultaneously evaluated with the balanced trade-off assigned previously, the SIACTA-RSM/1/OPSO have the best results, followed by SIACTA-RSM/2/ODE, SIACTA-RSM/1/ODE, and SIACTA-RSM/2/OPSO. The worst result is provided by IACTA-GDM versions, where the best result in it, given by IACT-GDM/OPSO, loses its effectiveness by 16.66% and 12.76% with reference to the best results in the other versions (SIACTA-RSM/1/OPSO and SIACTA-RSM/2/ODE, respectively).
- Considering the total results per approach (given in eighth column of Table 21), it is observed that IACTA-GDM is outperformed in 10.15% and 9.97% by SIACTA-RSM/1 and SIACTA-RSM/2, respectively.
- Both SIACTA-RSM approaches (SIACTA-RSM/1 and SIACTA-RSM/2) present competitive performance due to the difference between them is only 0.74% according to the total results per approach in Table 21. This difference can be considered a negligible one. So, in both approaches, slightly different trade-offs are presented, and each can benefit a particular application.
- Considering all statistical comparisons, the proposed SIACTA-RSM is a suitable alternative for online controller tuning when the computational burden has surpassed the capabilities of the traditional IACTA-GDM. In addition, the proposal presents a competitive controller performance with respect to the IACTA-GDM.

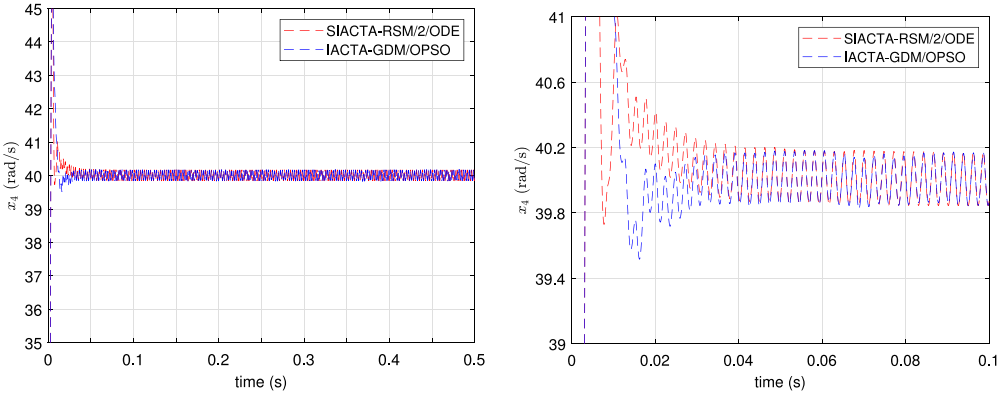
4.3. Graphical comparisons

After the statistical comparison, it might be helpful to comprehend the meaning of performance differences graphically. To achieve this, two comparisons are carried out. The first one regards the best-case execution of the best versions (according to their Friedman's rank) of SIACTA-RSM/2 and IACTA-GDM approaches. Meanwhile, the second comparison examines the worst-case execution of the wort versions (according to their Friedman's rank) of SIACTA-RSM/2 and IACTA-GDM approaches. The BLDC motor's angular speed and power consumption are evaluated in both comparisons.

4.3.1. Best-case graphical analysis

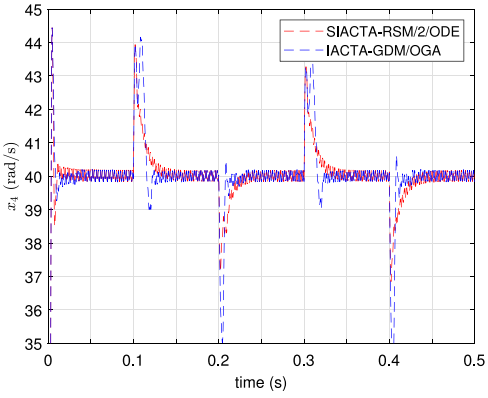
Fig. 9 collects the angular speed comparisons between the best versions of SIACTA-RSM/2 and IACTA-GDM approaches of each experiment, and Fig. 10 presents their respective power consumption (P) during their executions. The particular outcomes of these executions are presented in Tables 6, 9, 12, and 15. A necessary clarification is that all images from now on depicted are zoomed in to clarify the differences between the compared versions. The following notes describe the remarks by experiment.

- Experiment 1: In this experiment, SIACTA-RSM/2/ODE and IACTA-GDM/OPSO have the best outcomes. Figs. 9(a) and 10(a) show that SIACTA-RSM/2/ODE has almost the same performance as IACTA-GDM/OPSO, regarding the angular speed regulation and the power consumption, respectively. The only difference is perceived during the starting of the BLDC motor in the first 0.05 (s) as shown in Figs. 9(b) and 10(b), where SIACTA-RSM/2/ODE has a smoother/slower response to reach the steady state. Finally, according to Table 6 in this particular case, the total closed-loop performance ($\mathcal{J}_{\mathcal{T}}$) difference between the compared versions shows that SIACTA-RSM/2/ODE has slight improvements over IACTA-GDM/OPSO by reducing its outcome by 0.05%.

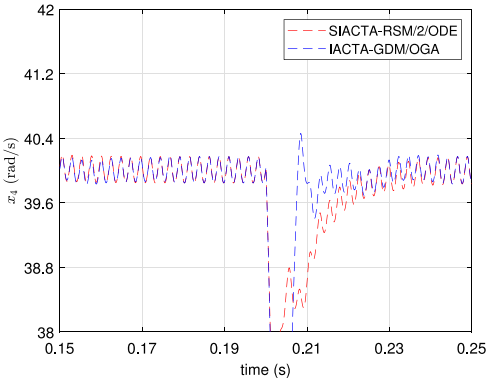


(a) Experiment 1

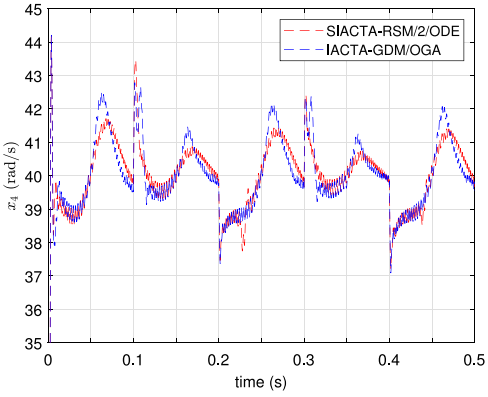
(b) Experiment 1 - close-up view



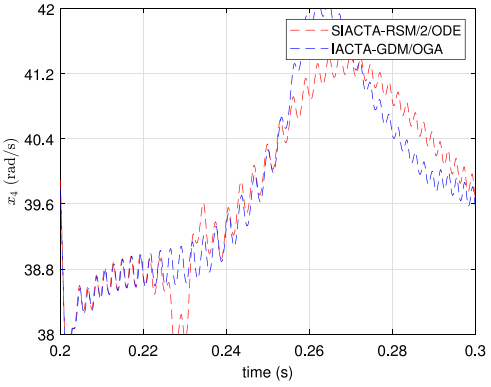
(c) Experiment 2



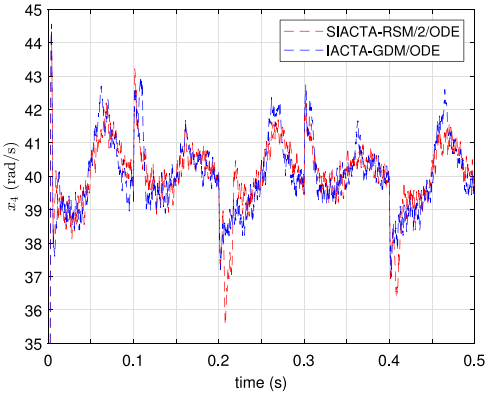
(d) Experiment 2 - close-up view



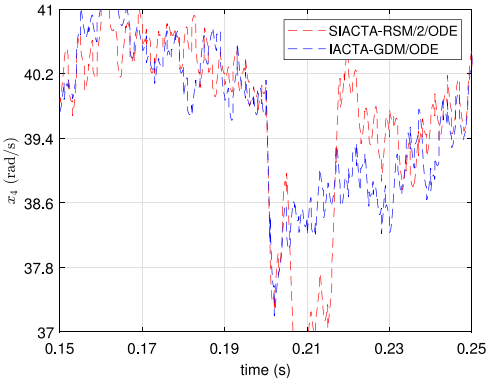
(e) Experiment 3



(f) Experiment 3 - close-up view

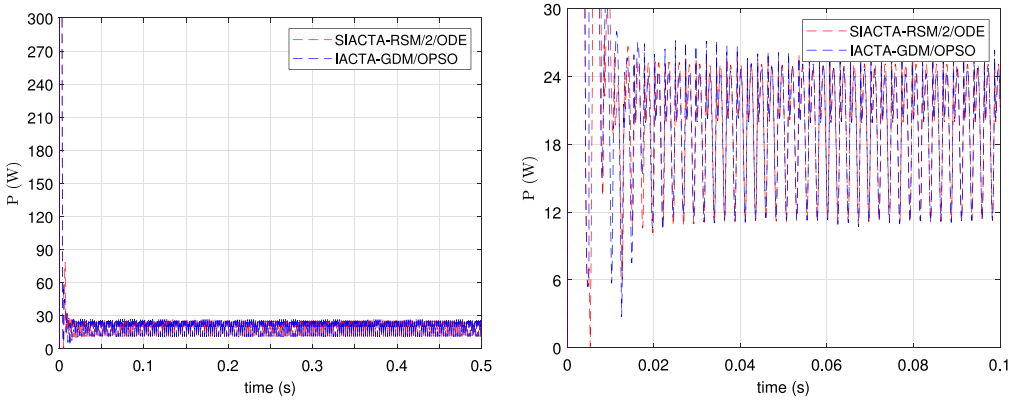


(g) Experiment 4



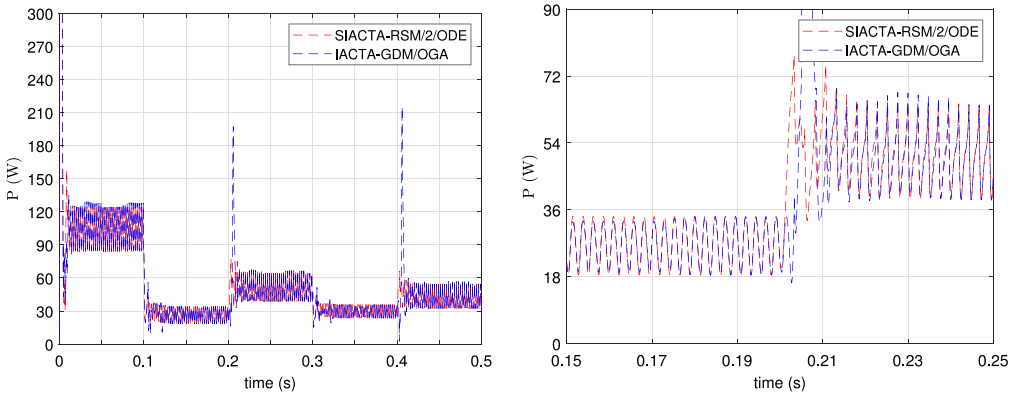
(h) Experiment 4 - close-up view

Fig. 9. Angular speed comparison of the best versions by approach.



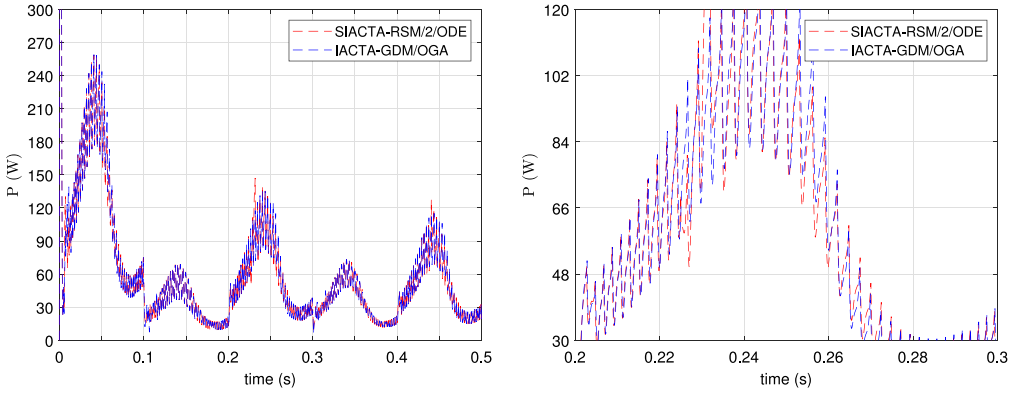
(a) Experiment 1

(b) Experiment 1 - close-up view



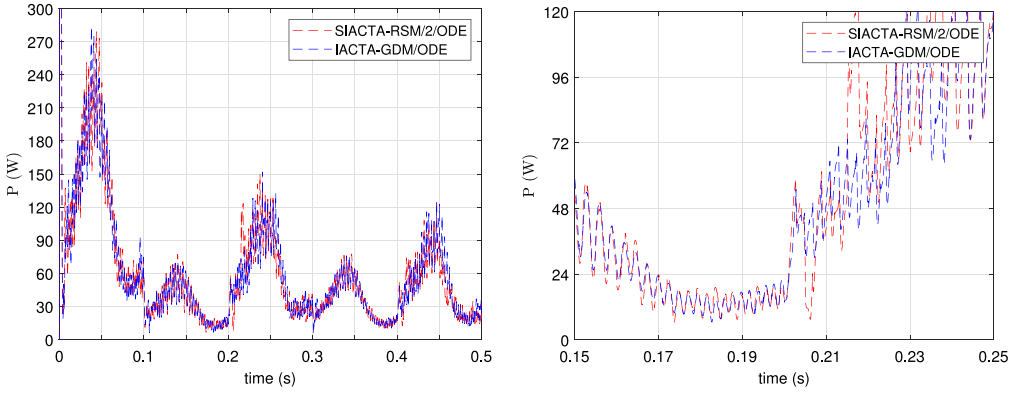
(c) Experiment 2

(d) Experiment 2 - close-up view



(e) Experiment 3

(f) Experiment 3 - close-up view



(g) Experiment 4

(h) Experiment 4 - close-up view

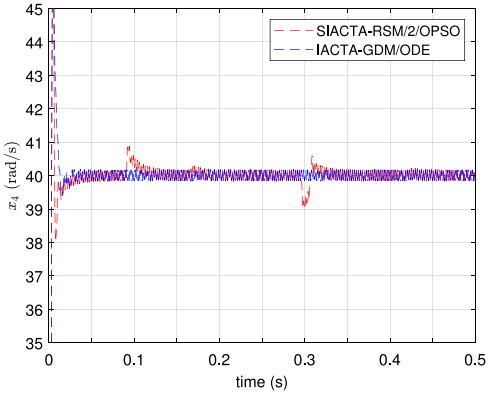
Fig. 10. Power consumption comparison of the best versions by approach.

- Experiment 2: SIACTA-RSM/2/ODE and IACTA-GDM/OGA report the best results in this experiment. Fig. 9(c) presents the angular speed comparison between SIACTA-RSM/2/ODE and IACTA-GDM/OGA, where SIACTA-RSM/2/ODE shows slower error compensations than IACTA-GDM/OGA. However, SIACTA-RSM/2/ODE achieves shorter oscillation, which, with the slower compensations, produces a smoother response, as in instants 0.2 (s) and 0.4 (s). For example, Fig. 9(d) shows a close-up view of the instant 0.2 (s), where it is observed that SIACTA-RSM/2/ODE has slower responses than IACTA-GDM/OGA, requiring more time to return to a steady state. Nonetheless, SIACTA-RSM/2/ODE reduces by 40% the error's magnitude at these intervals. On the other hand, Fig. 10(c) shows that SIACTA-RSM/2/ODE requires less power impulse, while IACTA-GDM/OGA employs additional power impulses to achieve its error compensations. This is more visible in Fig. 10(f) at the time 0.21 (s), where the oscillations of SIACTA-RSM/2/ODE barely surpasses the 72 (W) while the oscillation of IACTA-GDM/OGA comes out of the figure. Finally, according to the best result about closed-loop performance ($\mathcal{J}_{\mathcal{T}}$) of Table 9, it is observed that SIACTA-RSM/2/ODE decreases by 2.78% the best reported result of IACTA-GDM/OGA.
- Experiment 3: The best results between the approaches for this experiment are reported by SIACTA-RSM/2/ODE and IACTA-GDM/OGA. Fig. 9(e) displays a common trend where SIACTA-RSM/2/ODE responds faster, while IACTA-GDM/OGA has a smoother response. This experiment presents a drawback of the online controller tuning approaches. In the interval from 0.2 (s) to 0.25 (s), SIACTA-RSM/2/ODE has an undesired oscillation observed in Fig. 9(f). This kind of oscillation is presented when the optimizer of the prediction stage does not find a suitable solution, producing an overcompensation of the error. Nevertheless, Fig. 10(e) depicts that this error does not significantly affect the BLDC motor's power consumption. Particularly, Fig. 10(f) shows that SIACTA-RSM/2/ODE has minor oscillations during the interval from 0.22 (s) to 0.24 (s), while the rest of the operation is similar to IACTA-GDM/OGA. Finally, regarding the results of Table 12, the difference in the evaluation of the closed-loop performance ($\mathcal{J}_{\mathcal{T}}$) shows that SIACTA-RSM/2/ODE diminishes the outcome of IACTA-GDM/OGA by 1.40%.
- Experiment 4: The best results are obtained by SIACTA-RSM/2/ODE and IACTA-GDM/ODE for this experiment. Fig. 9(g) portrays that in this experiment, SIACTA-RSM/2/ODE has more considerable ripples, especially at the time 0.2 (s). This increase in the oscillations explains the reason for the competitiveness drop of the SIACTA-RSM approaches as the complexity of the experiments increases. Nonetheless, these oscillations are corrected by the own control system at future intervals as observed in Fig. 9(h) where after the instant 0.23 (s), the speed regulation performance of SIACTA-RSM/2/ODE equalizes the one of IACTA-GDM/ODE. On the other hand, Fig. 10(g) shows that these oscillations affect the power consumption. This is visible in the performance difference between SIACTA-RSM/2/ODE and IACTA-GDM/ODE. Particularly after the instant 0.2 (s) interval in Fig. 10(h), it is observed that SIACTA-RSM/2/ODE requires more power consumption to equalize the speed regulation performance of IACTA-GDM/ODE. Ultimately, a common trend is observed in the best-case executions. It is that SIACTA-RSM/2 versions have slightly better results than IACTA-GDM alternatives. In this experiment, SIACTA-RSM/2/ODE reduces by 0.58% the closed-loop performance ($\mathcal{J}_{\mathcal{T}}$) of IACTA-GDM/ODE (according to the best result of Table 15).

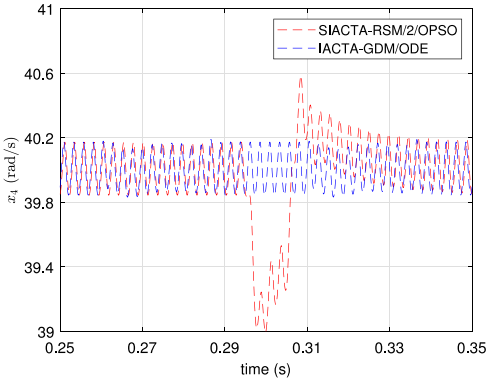
4.3.2. Worst-case graphical analysis

As a matter of completeness and looking for a complete representation of the results, a graphical comparison of the worst execution of the worst version (according to Friedman's ranks) of the IACTA-GDM and SIACTA-RSM/2 approaches is carried out. Fig. 11 summarizes the speed regulation result of the four experiments, and Fig. 12 collects the power consumption results. Also, the statistical outcomes of this execution are portrayed in Tables 6, 9, 12, and 15. Next, some noteworthy points by experiment are offered:

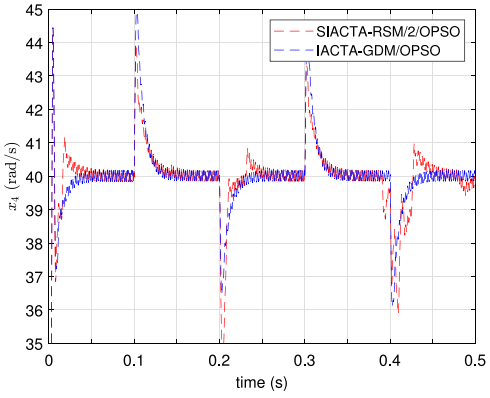
- Experiment 1: SIACTA-RSM/2/OPSO and IACTA-GDM/ODE have the worst outcomes for this experiment. Fig. 11(a) graphically shows the performance of the SIACTA-RSM/2/OPSO and IACTA-GDM/ODE. In this case, SIACTA-RSM/2/OPSO presents undesired oscillations in the speed regulation at 0.1 (s) and 0.3 (s) intervals. Nonetheless, it is worth pointing out that the errors are compensated at future update intervals, as observed in Fig. 11(b), where after 0.31 (s), the speed regulation of SIACTA-RSM/2/OPSO returns to its steady state. Despite the oscillation in the speed regulation, Fig. 12(a) displays that there are no significant differences regarding the power consumption between SIACTA-RSM/2/OPSO and IACTA-GDM/ODE besides the minor overshoot of the interval between 0.29 (s) and 0.31 (s) presented in Fig. 12(b). Finally, the overall closed-loop performance ($\mathcal{J}_{\mathcal{T}}$) of SIACTA-RSM/2/OPSO reduces its effectiveness by 11.07% with respect to IACTA-GDM/ODE (according to the worst result of Table 6).
- Experiment 2: For this experiment, SIACTA-RSM/2/OPSO and IACTA-GDM/OPSO report the worst results at each approach. Fig. 11(c) depicts once again ripples in the speed regulation outcome of SIACTA-RSM/2/OPSO. However, it is interesting that SIACTA-RSM/2/OPSO achieves the same quickness compensation for speed regulation of IACTA-GDM/OPSO, as observed in Fig. 11(d). Fig. 12(c) presents a more visible difference in the power consumption between SIACTA-RSM/2/OPSO and IACTA-GDM/OPSO, resulting from the undesired oscillation. Particularly, Fig. 12(d) shows that SIACTA-RSM/2/OPSO requires more energy to compensate for the speed variations. Finally, according to the worst result of Table 9, SIACTA-RSM/2/OPSO drops its competitiveness by decreasing its closed-loop performance ($\mathcal{J}_{\mathcal{T}}$) in 7.74% when compared to IACTA-GDM/OPSO.
- Experiment 3: The worst results are reported by SIACTA-RSM/2/OGA and IACTA-GDM/ODE. Fig. 11(e) shows again that SIACTA-RSM/2/OGA has undesired oscillations in the speed regulation, which also affect their power consumption in Fig. 12(e). However, it is important to notice that the system immediately corrects these errors in future update intervals. It is worth pointing out that IACTA-GDM/ODE has undesired oscillations, as presented in Fig. 11(f), where during the interval between 0.44 (s) and 0.46 (s) SIACTA-RSM/2/OGA achieves less oscillations in the speed regulation than IACTA-GDM/ODE. Nonetheless, this behavior results in worse power consumption, as shown in Fig. 12(f), where SIACTA-RSM/2/OGA requires more energy. Ultimately, SIACTA-RSM/2/OGA loses against IACTA-GDM/ODE by increasing its closed-loop performance ($\mathcal{J}_{\mathcal{T}}$) in 11.20% (according to the worst results reported in Table 12).
- Experiment 4: In this experiment the worst results are those of SIACTA-RSM/2/OGA and IACTA-GDM/OPSO. Fig. 11(g) depicts the undesired oscillations in the speed regulation. Even if the oscillation of SIACTA-RSM/2/OGA is bigger at 0.4 (s), IACTA-GDM/OPSO also has an undesired oscillation at 0.05 (s). Fig. 11(h) shows that the error of SIACTA-RSM/2/OGA at 0.44 (s) is compensated and equalizes the behavior of IACTA-GDM/OPSO. Nonetheless, Fig. 12(g) demonstrates that only SIACTA-RSM/2/OGA has an oscillation that affects power consumption. Even if the SIACTA-RSM/2/OGA version presents errors around 0.4 (s) interval and does not equalize the behavior of the IACTA-GDM/OPSO, the SIACTA-RSM/2/OGA version still has a pretty similar response (as observed in Fig. 12(h)). Finally, as observed in the results of Table 15, SIACTA-RSM/2/OGA increases by 13.93% the closed-loop performance ($\mathcal{J}_{\mathcal{T}}$) of IACTA-GDM/OPSO.



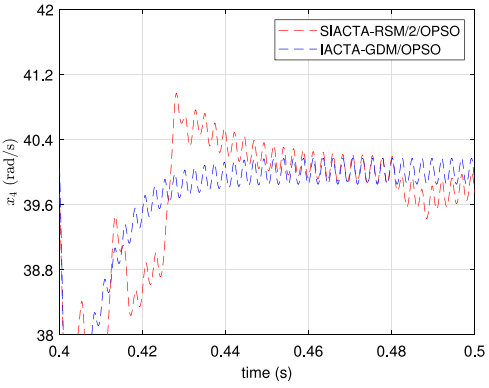
(a) Experiment 1



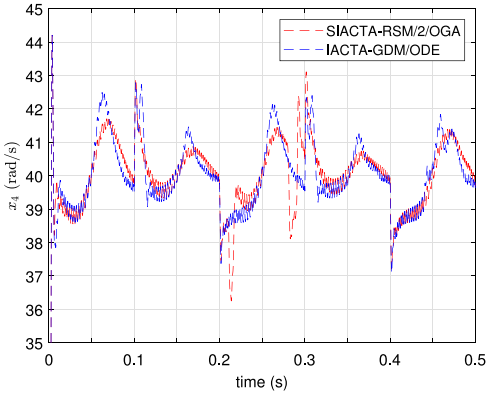
(b) Experiment 1 - close-up view



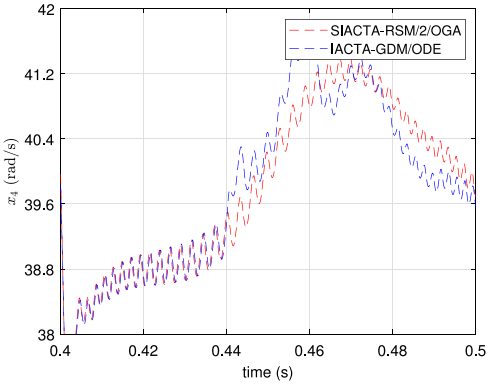
(c) Experiment 2



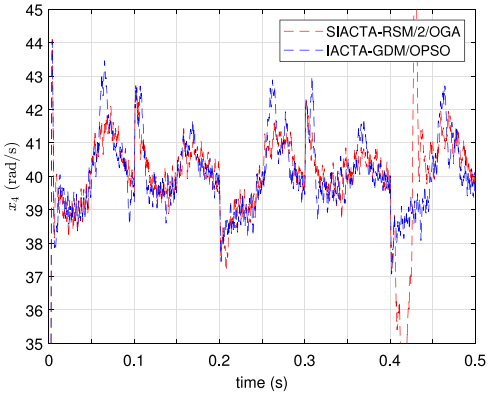
(d) Experiment 2 - close-up view



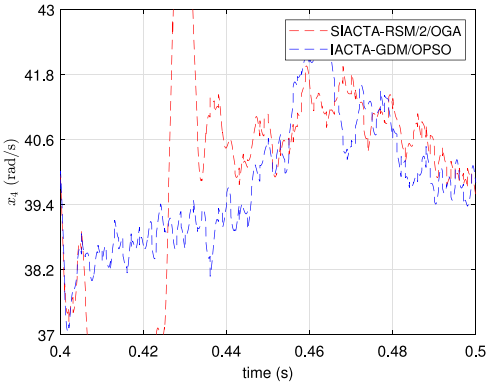
(e) Experiment 3



(f) Experiment 3 - close-up view

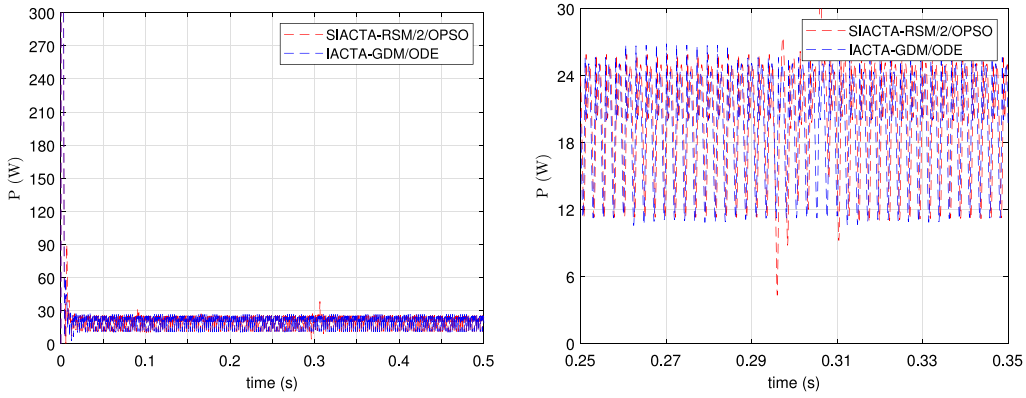


(g) Experiment 4



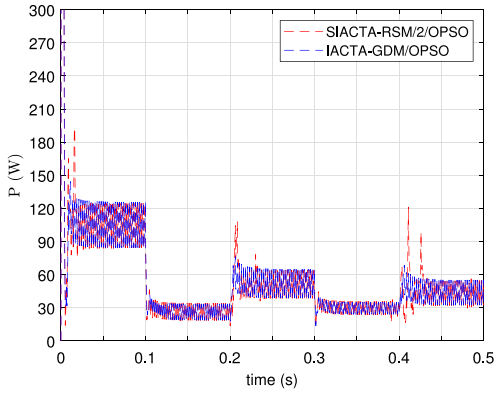
(h) Experiment 4 - close-up view

Fig. 11. Angular speed comparison of worst versions by approach.

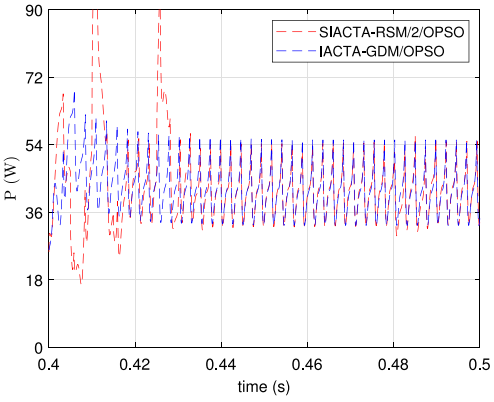


(a) Experiment 1

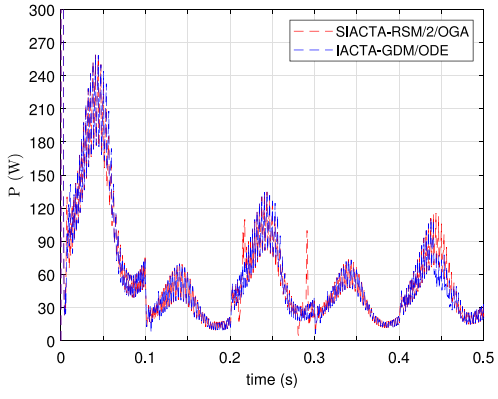
(b) Experiment 1 - close-up view



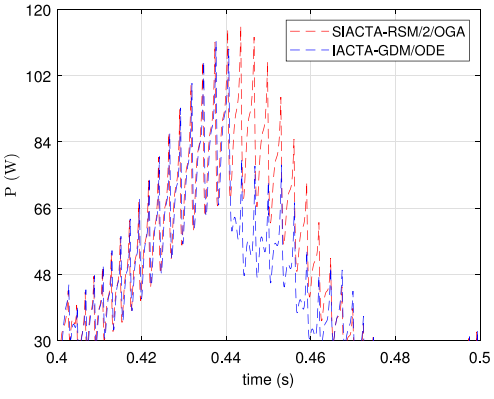
(c) Experiment 2



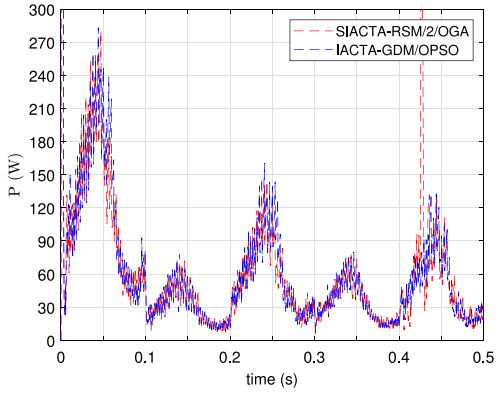
(d) Experiment 2 - close-up view



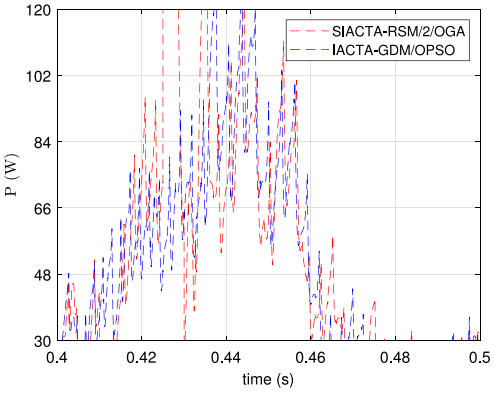
(e) Experiment 3



(f) Experiment 3 - close-up view



(g) Experiment 4



(h) Experiment 4 - close-up view

Fig. 12. Power consumption comparison of worst versions by approach.

4.3.3. Discussion of the graphical comparison

The above-provided graphical comparisons offer a better understanding of the statistical differences between SIACTA-RSM/2 and IACTA-GDM approaches. Figs. 9 and 10 show that both approaches have nearly the same outcomes, where the slight differences result from undesired oscillations. It is worth pointing out that these oscillations are characteristic of all the online controller tuning approaches based on bioinspired algorithms (SIACTA-RSM and IACTA-GDM).

Nonetheless, these oscillations affect slightly more the SIACTA-RSM/2 approach than the IACTA-GDM one in the closed-loop system performance. This is because SIACTA-RSM/2 uses an approximated surrogate model as the reference model, accumulating an inherent error, while IACTA-GDM employs the actual dynamic model of the system closest to the actual physical system. Therefore, the SIACTA-RSM/2 presents more ripples, as shown in Figs. 11 and 12.

However, it is important to remember that the SIACTA-RSM/2 approach can equalize the outcomes of the IACTA-GDM one, reducing the computational burden by up to 80%. A better understanding of the proposal and its characteristics is achieved when all the present information is considered. Thus, the SIACTA-RSM/2 approach is a trustworthy option when the computational burden is a limitation of the traditional IACTA-GDM approach, especially for those applications where small oscillations are accepted, like in heavy management systems or unmanned vehicles.

4.4. Higher polynomial degree regression analysis

Aiming to better understand the proposal's behavior as the degree of the polynomial regression increases, an additional approach based on 3rd degree polynomial is developed. The approach is called SIACTA-RSM/3/ODE, where the ODE is selected as the optimizer because it is one of the most outstanding with different uncertainties and disturbances. This new version is executed thirty times and compared against SIACTA-RSM/2/ODE (the most promising SIACTA-RSM variant) with the conditions of Experiment 4 (with more parametric uncertainties and disturbances). Tables 22 and 24 present the descriptive statistical results of the online controller tuning performance criteria and the computational time required by the online controller tuning versions, respectively. These tables report the best results by criterion/columns in **boldface**. Additionally, the inferential analyses are performed using Wilcoxon signed-rank tests. Tables 23 and 25 present the respective results of the Wilcoxon signed-rank tests about the online controller tuning performance and computational time of the versions compared. In these tables, the p-values are reported in the last columns, where if the resulting p-value is lower than the statistical significance $\alpha = 0.05$, the null hypothesis is rejected, meaning that one of the versions compared is statistically better. The winner is selected through the signed ranked sums, where if the positive ranked sum R^- (second column) is lower than the negative ranked sum R^+ (third), the winner is the left side of the comparison expressed in the first column; otherwise, the right side is the winner. The winner is reported in **boldface** in the first column of these tables.

Next, a summary of the results is offered:

- Analysis about online controller tuning performance: Table 22 shows that SIACTA-RSM/2/ODE has the best mean result of the thirty executions regarding the online controller tuning performance. For this comparison, SIACTA-RSM/2/ODE reduces the online controller tuning functional $\mathcal{J}_\mathcal{T}$ by 8.16%, the IAE by 25.88%, and the power P_{avg} by 2.29%. In a deeper inferential statistic analysis, Table 23 shows that the null hypothesis is rejected, meaning that there is a winner among the versions, which in this case is SIACTA-RSM/2/ODE. These results show that increasing the polynomial degree does not necessarily lead to better performance of the online controller tuning.
- Analysis about computational time: Table 24 exhibits that SIACTA-RSM/2/ODE is able to reduce the computational time by 71.29% with respect to SIACTA-RSM/3/ODE to solve the online controller tuning. This outstanding difference is reflected in the inferential analysis, where Table 25 shows that the null hypothesis is rejected and that SIACTA-RSM/2/ODE is the absolute winner (as the positive ranked sum value is 0). This shows that increasing the polynomial degree of the proposal beyond the second degree significantly affects the computational time required to solve the online controller tuning.
- Considering the obtained result by a third-degree polynomial regression, it is observed that increasing the degree in the particular case of the BLDC motor does not improve the online controller tuning performance or the computational time.

5. Conclusions

This work proposes the surrogate indirect adaptive controller tuning approach based on the response surface method (SIACTA-RSM) and bioinspired algorithms, using multivariate-multitarget polynomial regression to obtain the reference model dynamically and, hence, to update the controller gains that improve the controller's performance during the execution of a task while reducing the computational time required to solve such update processes.

To evaluate the proposal, the SIACTA-RSM approach is applied for tuning a Proportional Integral (PI) controller of a brushless direct current (BLDC) motor. This application aims to optimize the speed regulation and power consumption of a BLDC motor by simultaneously minimizing its integral absolute error (IAE) and average power (P_{avg}). These criteria are addressed through the weighted product method (WPM), resulting in a single optimization functional $\mathcal{J}_\mathcal{T}$ related to the online controller tuning performance. The proposal also incorporates different bioinspired optimization algorithms (ODE, OGA, and OPSO) for the optimization problem solution.

The proposal is tested through four experiments where disturbances and parametric uncertainties increase in each experiment. The proposed SIACTA-RSM versions (based on first and second-degree polynomial regressions) are compared to the well-established Indirect Adaptive Controller Tuning Approach based on the General Dynamic Model (IACTA-GDM) and an optimization-based controller tuning method (optimum offline controller tuning).

Analyzing and evaluating equitably the trade-off between the closed-performance ($\mathcal{J}_\mathcal{T}$) of the BLDC motor and the average computational time required to solve the tuning (solver time), it is observed that the SICTA-RSM proposal outperforms the weighted criterion with respect to the traditional IACTA-GDM approach. The results show that the best versions of SIACTA-RSM, given by SIACTA-RSM/1/OPSO and SIACTA-RSM/2/ODE, outperform by 16.66% and 12.76% the weighted criterion of the best version of IACTA-GDM (IACTA-GDM/OPSO). Furthermore, in a general comparison per approach (evaluating its versions simultaneously), it is observed that SIACTA-RSM/1 has the best trade-off results, followed by SIACTA-RSM/2. The SIACTA-RSM/1 and SIACTA-RSM/2 reduce by 10.15% and 9.97% the trade-off obtained by the IACTA-GDM approach.

For more particularities of the analyses, it is observed that, when only the online controller tuning performance ($\mathcal{J}_\mathcal{T}$) is evaluated, the SIACTA-RSM proposal drops its competitiveness as the parametric uncertainties and external disturbances increase. The inferential statistical comparison results show that the loss percentage in the comparison increases from 55.55% to 100% through the increment of the uncertainties and disturbances. Without uncertainties and disturbances, the proposal is as competitive as the IACTA-GDM. Nonetheless, it is essential to point out that the disparity between the mean results of each approach (SIACTA-RSM/2 and IACTA-GDM) variates between only 1.78% to 3.18% through the experiments. Final users should

Table 22
Descriptive statistics results — 3rd degree polynomial regression.

Version	IAE			P_{avg}			$\mathcal{J}_T$				
	best	worst	mean	best	worst	mean	best	worst	mean	std.	C.I. 95%
SIACTA-RSM/2/ODE	**0.4434**	**0.7922**	**0.5434**	**64.2700**	**68.7417**	**65.8520**	**21.1015**	**25.1391**	**22.3478**	**0.9627**	**[22.003–22.692]**
SIACTA-RSM/3/ODE	0.4903	0.9108	0.7332	64.6275	69.7558	67.3983	21.6073	26.1961	24.3346	1.1566	[23.921–24.748]

Table 23
Wilcoxon signed rank test regarding $\mathcal{J}_T$.

Comparison		R^-	R^+	p-value
SIACTA-RSM/2/ODE	SIACTA-RSM/3/ODE	23	442	1.6394E−05

Table 24
Descriptive statistics - Solver time - 3rd degree polynomial regression.

Version	Solver time (s)				
	best	worst	mean	std.	C.I.
SIACTA-RSM/2/ODE	**0.0220**	**0.0323**	**0.0298**	**0.0022**	**[0.0290–0.0306]**
SIACTA-RSM/3/ODE	0.0723	0.1274	0.1038	0.0163	[0.0980–0.110]

Table 25
Wilcoxon signed rank test regarding solver time.

Comparison		R^-	R^+	p-value
SIACTA-RSM/2/ODE	SIACTA-RSM/3/ODE	0	465	1.7344E−06

consider such differences to determine whether they are significant or acceptable in other applications.

The other particularities of the analysis is when the computational time (solver time) is only evaluated, where the SIACTA-RSM proposal is the outstanding winner with respect to IACTA-GDM versions. According to the results of the inferential statistics, only two ties in SIACTA-RSM/2/OGA are confirmed through the experiments, and the rest of the comparisons of the proposal present unquestionable triumphs. The descriptive statistics indicate that the fastest versions of the proposed SIACTA-RSM (SIACTA-RSM/ 1/OPSO and SIACTA-RSM/2/ODE) provide a reduction of the computational time of around 90.10% and 80.28% regarding the best-reported result in IACTAGDM (IACTA-GDM/OPSO).

As a consequence, the proposed SIACTA-RSM is an outstanding option when the well-established IACTA-GDM approach has reached its computational burden limit, maintaining a competitive outcome about the control system performance while significantly reducing the computational time required to solve the indirect adaptive tuning.

In future work, the proposal's implementation on experimental platforms is considered. Also, the analysis of other surrogate model methods based on machine learning and deep learning techniques is another future research direction according to the state-of-the-art described in Section 1.2. The way of handling a diverse set of control performance functions based on the multiobjective optimization framework is another future work. The implementation of parallel/granular computing is also another research direction to enhance the computational time capability of the SIACTA-RSM proposal, considering that parallel/granular computing technology has been successfully implemented in applications where large amounts of data and processes are required like in Huang, Tang, Zhao, Zhang, and Pedrycz (2022), Kuo, Su, Zulvia, and Lin (2018), Singh and Dhiman (2018), Yen, Chen, Chen, Hsu, and Wu (2014), Zhang, Pedrycz, Fayek, and Dong (2022). This technology can be helpful to deal with the computational burden limitations of deep learning techniques, making the combination of such techniques with the SIACTA-RSM proposal to make it feasible.

Funding

This research was funded in part by the Secretaría de Investigación y Posgrado (SIP) of the Instituto Politécnico Nacional, Mexico under Grant 20220255, 20230320, and Consejo Nacional de Humanidades, Ciencia y Tecnología (CONAHCYT) under Grant CBF2023-2024-4525.

CRediT authorship contribution statement

Alam Gabriel Rojas-López: Conceptualization, Validation, Formal analysis, Investigation, Data curation, Software, Visualization, Methodology, Writing – original draft, Writing – review & editing. **Miguel Gabriel Villarreal-Cervantes:** Methodology, Conceptualization, Formal analysis, Investigation, Writing – original draft, Writing – review & editing, Resources, Supervision. **Alejandro Rodríguez-Molina:** Conceptualization, Investigation, Methodology, Writing – review & editing, Supervision.

Declaration of competing interest

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

Data availability

Data will be made available on request.

Acknowledgments

Authors acknowledge the support from the Comisión de Operación y Fomento de Actividades Académicas of the Instituto Politécnico Nacional (COFAA-IPN), the Secretaría de Investigación y Posgrado of the IPN (SIP-IPN), the Colegio de Ciencia y Tecnología (CCyT) of the Universidad Autónoma de la Ciudad de México (UACM), and the Consejo Nacional de Humanidades, Ciencia y Tecnología (CONAHCYT). The first author acknowledges the support from the CONAHCYT in Mexico through a scholarship to pursue graduate studies at CIDETEC-IPN.

Appendix A. List of acronyms

BLDC	Brushless direct current.
PID	Proportional–Integral–Derivative.
AI	Artificial intelligence.
GDM	General dynamic model.
SM	Surrogate modeling.
GP	Gaussian process.
SVM	Support Vector Machines.
ANN	Artificial Neural Network.
RSM	Response Surface Method.
kNN	k-Nearest Neighbors.
SOCTM	Surrogate optimization-base controller tuning method.
SDACTM	Surrogate direct adaptive controller tuning method.

SIACTM	Surrogate indirect adaptive controller tuning method.
MARS	Multivariate Adaptive Regression Spline.
S-M	Space Mapping.
RBF	Radial Basis Function.
RF	Random Forest.
SMGO-Δ	Set Membership Global Optimization with black-box constraints.
LTI	Linear Time invariant.
PSO	Particle Swarm Optimization.
EGO	Efficient Global Optimization.
GA	Genetic Algorithm.
GLISP	GLobal minimum, Inverse distance, Surrogate RBF, Preferred-based optimization.
Δ-DOGS	Delaunay-based Derivative-free Optimization via Global Surrogate.
BO	Bayesian Optimization.
BA	Bat Algorithm.
ES	Evolutionary Strategies.
DE	Differential Evolution.
GW	Gray Wolf.
HH	Harris Hawks.
MF	Moth Flame.
NSGA-II	Non-dominated Sorting Genetic Algorithm-II.
RL	Reinforcement Learning.
DACE	Design and Analysis of Computer Experiments.
MOBGO	Multi-Objective Bayesian Global Optimization.
GS	Grid Search.
SA	Simulated Annealing.
SMPSO	Speed-constrained Multi-objective Particle Swarm Optimization.
GDE-3	Generalized Differential Evolution.
MOEA	Multi-Objective Evolutionary Algorithm.
MOPSO	Multi-Objective Particle Swarm Optimization.
MOGOA	Multi-Objective Grasshopper Optimization Algorithm.
NSGA-III	Non-dominated Sorting Genetic Algorithm-III.
DRACO	Decomposed Robust Alternating Confidence-bound Optimization.
BA	Bat Algorithm.
LSM	Least Squares Method.
ODE	Online Differential Evolution.
OGA	Online Particle Swarm Optimization.
OPSO	Online Genetic Algorithm.
SBX	Simulated Binary Crossover
NDM	Normally Distributed Mutation.
Back-emf	Back-electromagnetic force.
WPM	Weighted Product Method.
IAE	Integral Absolute Error.
P_{avg}	Power average.
SIACTA-RSM	Surrogate Indirect Adaptive Controller Tuning Approach based on the RSM regressor.
IACTA-GDM	Indirect Adaptive Controller Tuning Approach based on the General Dynamic Model.

Appendix B. List of symbols

$\mathbb{R}^n$	n-dimensional Euclidean space.
$\mathbb{R}^{n\times m}$	$n \times m$ Matrix representation.
$\boldsymbol{p}$	Vector notation.
$\mathbf{P}$	Matrix notation.
t_p	Present time.
ΔT_U	Update interval.
δt_l	System's sampling interval.
T_I, T_P	Identification and prediction periods.
n_l	Number of samplings by identification period.
m_l	Future intervals that form the prediction period.
$\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{y}$	Dynamic system's state, input and output vectors.
n_x, n_u, n_y	Number of states, input, and outputs of a dynamic system.
$\widehat{\boldsymbol{y}}$	Surrogate model by multivariate-multitarget polynomial regressor.
$\overrightarrow{\boldsymbol{\beta}}^*$	Optimum regressor's coefficients vector.
ν	Number of polynomial regressor's terms.
n_v	Number of design variables.
NP	Population size.
G_{max}	Maximum generations.
Cr	Crossover rate.
$[F_{min}, F_{max}]$	Mutation scale factor range.
$[\boldsymbol{v}_{min}, \boldsymbol{v}_{max}]$	Velocity range.
C_1, C_2	Individual and global experience coefficients.
$\boldsymbol{\omega}$	Inertia factor.
T_s	Tournament size.
η_c	Distribution index.
$\mu, \boldsymbol{\sigma}_\mu{}^2$	Mutation factor and variance distribution.
$\boldsymbol{a}$	BLDC motor's parameter vector.
R, L	BLDC motor's Line resistance and inductance.
k_e, k_t	BLDC motor's Back-emf and torque.
$\boldsymbol{J}, \boldsymbol{B}$	BLDC motor's inertia and magnetic friction coefficients.
P	BLDC motor's pair poles.
τ_l	BLDC motor's torque load.
$i_\gamma \ \forall \ \gamma \in \{A, B, C\}$	BLDC motor's A,B, and C line current.
Ω, θ	BLDC motor's angular speed and position.
k_p, k_i	Controller's proportional and integral gains.
$\boldsymbol{k}$	Controller's gains vector.
J_1	Optimization functional related to the IAE.
J_2	Optimization functional related to the power P_{avg}.
$\mathcal{J}_\mathcal{T}$	Controller tuning functional (WPM's result of J_1 and J_2).
$\boldsymbol{w}$	WPM's weights vector for $\mathcal{J}_\mathcal{T}$.
$\boldsymbol{k}_{max}, \boldsymbol{k}_{min}$	Controller's gains vector limits.
α	Statistical significance value.
dt	Euler's numerical integration step.

Appendix C. Post-hoc pairwise comparison tables

Tables C.26, C.27, C.28, and C.29 present the post-hoc pairwise comparison about the tuning functional $\mathcal{J}_\mathcal{T}$, and Table C.30 displays the post-hoc pairwise comparison result regarding the solver time required for the online controller tuning versions. These tables follow the same structure, where the two first columns indicate the pairs of versions compared, the third column indicates the unadjusted p-values, while the fourth to sixth columns contain the adjusted values (by Bonferroni, Holm, and Shaffer's corrections, respectively), and the last column presents the Friedman's rank difference between the versions compared. In these Tables, if any of the p-values (unadjusted or adjusted columns) is lower than the statistical significance $\alpha = 0.05$, the null hypothesis (the compared versions are equal) is rejected, and the p-value is reported in **boldface**. Looking for robust analysis, the comparisons are considered tie until all the p-values reject the null hypothesis. If this condition is accomplished, a winner of the comparison is selected through the rank difference. If the rank difference is positive, the right side of the comparison is the winner, and if it is negative, the left side is better. If there are winners in the comparisons, these are in **boldface** (see Table C.29).

Table C.26
Post-hoc pairwise comparison regarding J_T - Experiment 1.

vs		Unadjusted	Bonferroni	Holm	Shaffer	Rank diff.
IACTA-GDM/ODE	IACTA-GDM/OGA	1.3143E−01	1.0000E+00	1.0000E+00	1.0000E+00	1.07
IACTA-GDM/ODE	**IACTA-GDM/OPSO**	**1.1432E−03**	**4.1154E−02**	**1.8291E−02**	**1.8291E−02**	2.30
IACTA-GDM/ODE	SIACTA-RSM/1/ODE	**5.8263E−07**	**2.0975E−05**	**1.3400E−05**	**1.2818E−05**	−3.53
IACTA-GDM/ODE	SIACTA-RSM/1/OGA	**6.7000E−09**	**2.4120E−07**	**1.8090E−07**	**1.4740E−07**	−4.10
IACTA-GDM/ODE	SIACTA-RSM/1/OPSO	**9.9919E−08**	**3.5971E−06**	**2.4980E−06**	**2.1982E−06**	−3.77
IACTA-GDM/ODE	SIACTA-RSM/2/ODE	2.0309E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.90
IACTA-GDM/ODE	SIACTA-RSM/2/OGA	8.8754E−01	1.0000E+00	1.0000E+00	1.0000E+00	−0.10
IACTA-GDM/ODE	SIACTA-RSM/2/OPSO	4.2291E−01	1.0000E+00	1.0000E+00	1.0000E+00	−0.57
IACTA-GDM/OGA	IACTA-GDM/OPSO	8.1125E−02	1.0000E+00	9.7350E−01	9.7350E−01	1.23
IACTA-GDM/OGA	SIACTA-RSM/1/ODE	**7.7496E−11**	**2.7899E−09**	**2.2474E−09**	**2.1699E−09**	−4.60
IACTA-GDM/OGA	SIACTA-RSM/1/OGA	**2.7364E−13**	**9.8510E−12**	**9.0301E−12**	**7.6619E−12**	−5.17
IACTA-GDM/OGA	SIACTA-RSM/1/OPSO	**8.1796E−12**	**2.9447E−10**	**2.5357E−10**	**2.2903E−10**	−4.83
IACTA-GDM/OGA	SIACTA-RSM/2/ODE	8.1366E−01	1.0000E+00	1.0000E+00	1.0000E+00	−0.17
IACTA-GDM/OGA	SIACTA-RSM/2/OGA	9.8960E−02	1.0000E+00	1.0000E+00	1.0000E+00	−1.17
IACTA-GDM/OGA	SIACTA-RSM/2/OPSO	**2.0895E−02**	7.5221E−01	3.1342E−01	3.1342E−01	−1.63
IACTA-GDM/OPSO	SIACTA-RSM/1/ODE	**1.5895E−16**	**5.7223E−15**	**5.4044E−15**	**4.4507E−15**	−5.83
IACTA-GDM/OPSO	SIACTA-RSM/1/OGA	**1.4171E−19**	**5.1015E−18**	**5.1015E−18**	**5.1015E−18**	−6.40
IACTA-GDM/OPSO	SIACTA-RSM/1/OPSO	**9.5235E−18**	**3.4285E−16**	**3.3332E−16**	**2.6666E−16**	−6.07
IACTA-GDM/OPSO	SIACTA-RSM/2/ODE	**4.7715E−02**	1.0000E+00	6.2029E−01	6.2029E−01	−1.40
IACTA-GDM/OPSO	SIACTA-RSM/2/OGA	**6.8851E−04**	**2.4787E−02**	**1.1705E−02**	**1.1016E−02**	−2.40
IACTA-GDM/OPSO	SIACTA-RSM/2/OPSO	**5.0332E−05**	**1.8120E−03**	**9.0598E−04**	**9.0598E−04**	−2.87
SIACTA-RSM/1/ODE	SIACTA-RSM/1/OGA	4.2291E−01	1.0000E+00	1.0000E+00	1.0000E+00	−0.57
SIACTA-RSM/1/ODE	SIACTA-RSM/1/OPSO	7.4141E−01	1.0000E+00	1.0000E+00	1.0000E+00	−0.23
SIACTA-RSM/1/ODE	**SIACTA-RSM/2/ODE**	**3.6179E−10**	**1.3024E−08**	**1.0130E−08**	**1.0130E−08**	4.43
SIACTA-RSM/1/ODE	**SIACTA-RSM/2/OGA**	**1.2010E−06**	**4.3237E−05**	**2.5222E−05**	**2.5222E−05**	3.43
SIACTA-RSM/1/ODE	**SIACTA-RSM/2/OPSO**	**2.7227E−05**	**9.8017E−04**	**5.1731E−04**	**4.9009E−04**	2.97
SIACTA-RSM/1/OGA	SIACTA-RSM/1/OPSO	6.3735E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.33
SIACTA-RSM/1/OGA	**SIACTA-RSM/2/ODE**	**1.5375E−12**	**5.5349E−11**	**4.9199E−11**	**4.3049E−11**	5.00
SIACTA-RSM/1/OGA	**SIACTA-RSM/2/OGA**	**1.5417E−08**	**5.5502E−07**	**4.0085E−07**	**3.3918E−07**	4.00
SIACTA-RSM/1/OGA	**SIACTA-RSM/2/OPSO**	**5.8263E−07**	**2.0975E−05**	**1.3400E−05**	**1.2818E−05**	3.53
SIACTA-RSM/1/OPSO	**SIACTA-RSM/2/ODE**	**4.1209E−11**	**1.4835E−09**	**1.2363E−09**	**1.1539E−09**	4.67
SIACTA-RSM/1/OPSO	**SIACTA-RSM/2/OGA**	**2.1549E−07**	**7.7578E−06**	**5.1719E−06**	**4.7409E−06**	3.67
SIACTA-RSM/1/OPSO	**SIACTA-RSM/2/OPSO**	**6.0258E−06**	**2.1693E−04**	**1.2052E−04**	**1.0846E−04**	3.20
SIACTA-RSM/2/ODE	SIACTA-RSM/2/OGA	1.5730E−01	1.0000E+00	1.0000E+00	1.0000E+00	−1.00
SIACTA-RSM/2/ODE	SIACTA-RSM/2/OPSO	**3.8063E−02**	1.0000E+00	5.3288E−01	4.9481E−01	−1.47
SIACTA-RSM/2/OGA	SIACTA-RSM/2/OPSO	5.0928E−01	1.0000E+00	1.0000E+00	1.0000E+00	−0.47

Table C.27
Post-hoc pairwise comparison regarding J_T - Experiment 2.

vs		Unadjusted	Bonferroni	Holm	Shaffer	Rank diff.
IACTA-GDM/ODE	IACTA-GDM/OGA	1.8686E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.93
IACTA-GDM/ODE	IACTA-GDM/OPSO	**1.6210E−02**	5.8354E−01	3.0798E−01	2.9177E−01	−1.70
IACTA-GDM/ODE	SIACTA-RSM/1/ODE	**1.5997E−09**	**5.7590E−08**	**5.4390E−08**	**4.4792E−08**	−4.27
IACTA-GDM/ODE	SIACTA-RSM/1/OGA	**7.5225E−06**	**2.7081E−04**	**2.1815E−04**	**2.1063E−04**	−3.17
IACTA-GDM/ODE	SIACTA-RSM/1/OPSO	**9.1286E−05**	**3.2863E−03**	**2.3734E−03**	**2.0083E−03**	−2.77
IACTA-GDM/ODE	SIACTA-RSM/2/ODE	**3.4700E−03**	1.2492E−01	7.2870E−02	7.2870E−02	−2.07
IACTA-GDM/ODE	SIACTA-RSM/2/OGA	**5.0332E−05**	**1.8120E−03**	**1.3590E−03**	**1.1073E−03**	−2.87
IACTA-GDM/ODE	SIACTA-RSM/2/OPSO	**3.4792E−08**	**1.2525E−06**	**1.1134E−06**	**9.7418E−07**	−3.90
IACTA-GDM/OGA	IACTA-GDM/OPSO	**1.9602E−04**	**7.0566E−03**	**4.9004E−03**	**4.3124E−03**	−2.63
IACTA-GDM/OGA	SIACTA-RSM/1/ODE	**1.9249E−13**	**6.9297E−12**	**6.9297E−12**	**6.9297E−12**	−5.20
IACTA-GDM/OGA	SIACTA-RSM/1/OGA	**6.7000E−09**	**2.4120E−07**	**2.2110E−07**	**1.8760E−07**	−4.10
IACTA-GDM/OGA	SIACTA-RSM/1/OPSO	**1.6715E−07**	**6.0174E−06**	**5.0145E−06**	**4.6802E−06**	−3.70
IACTA-GDM/OGA	SIACTA-RSM/2/ODE	**2.2090E−05**	**7.9526E−04**	**6.1853E−04**	**6.1853E−04**	−3.00
IACTA-GDM/OGA	SIACTA-RSM/2/OGA	**7.7004E−08**	**2.7721E−06**	**2.3871E−06**	**2.1561E−06**	−3.80
IACTA-GDM/OGA	SIACTA-RSM/2/OPSO	**8.1796E−12**	**2.9447E−10**	**2.8629E−10**	**2.2903E−10**	−4.83
IACTA-GDM/OPSO	SIACTA-RSM/1/ODE	**2.8362E−04**	**1.0210E−02**	**6.8070E−03**	**6.2397E−03**	−2.57
IACTA-GDM/OPSO	SIACTA-RSM/1/OGA	**3.8063E−02**	1.0000E+00	6.4706E−01	6.1011E−01	−1.47
IACTA-GDM/OPSO	SIACTA-RSM/1/OPSO	1.3143E−01	1.0000E+00	1.0000E+00	1.0000E+00	−1.07
IACTA-GDM/OPSO	SIACTA-RSM/2/ODE	6.0408E−01	1.0000E+00	1.0000E+00	1.0000E+00	−0.37
IACTA-GDM/OPSO	SIACTA-RSM/2/OGA	9.8960E−02	1.0000E+00	1.0000E+00	1.0000E+00	−1.17
IACTA-GDM/OPSO	SIACTA-RSM/2/OPSO	**1.8628E−03**	6.7062E−02	**4.2845E−02**	**4.0983E−02**	−2.20
SIACTA-RSM/1/ODE	SIACTA-RSM/1/OGA	1.1979E−01	1.0000E+00	1.0000E+00	1.0000E+00	1.10
SIACTA-RSM/1/ODE	SIACTA-RSM/1/OPSO	**3.3895E−02**	1.0000E+00	6.1011E−01	6.1011E−01	1.50
SIACTA-RSM/1/ODE	SIACTA-RSM/2/ODE	**1.8628E−03**	6.7062E−02	**4.2845E−02**	**4.0983E−02**	2.20
SIACTA-RSM/1/ODE	SIACTA-RSM/2/OGA	**4.7715E−02**	1.0000E+00	7.6344E−01	7.6344E−01	1.40
SIACTA-RSM/1/ODE	SIACTA-RSM/2/OPSO	6.0408E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.37
SIACTA-RSM/1/OGA	SIACTA-RSM/1/OPSO	5.7161E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.40
SIACTA-RSM/1/OGA	SIACTA-RSM/2/ODE	1.1979E−01	1.0000E+00	1.0000E+00	1.0000E+00	1.10
SIACTA-RSM/1/OGA	SIACTA-RSM/2/OGA	6.7137E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.30
SIACTA-RSM/1/OGA	SIACTA-RSM/2/OPSO	2.9969E−01	1.0000E+00	1.0000E+00	1.0000E+00	−0.73
SIACTA-RSM/1/OPSO	SIACTA-RSM/2/ODE	3.2220E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.70
SIACTA-RSM/1/OPSO	SIACTA-RSM/2/OGA	8.8754E−01	1.0000E+00	1.0000E+00	1.0000E+00	−0.10
SIACTA-RSM/1/OPSO	SIACTA-RSM/2/OPSO	1.0898E−01	1.0000E+00	1.0000E+00	1.0000E+00	−1.13
SIACTA-RSM/2/ODE	SIACTA-RSM/2/OGA	2.5790E−01	1.0000E+00	1.0000E+00	1.0000E+00	−0.80
SIACTA-RSM/2/ODE	SIACTA-RSM/2/OPSO	**9.5219E−03**	3.4279E−01	1.9044E−01	1.7139E−01	−1.83
SIACTA-RSM/2/OGA	SIACTA-RSM/2/OPSO	1.4392E−01	1.0000E+00	1.0000E+00	1.0000E+00	−1.03

Table C.28
Post-hoc pairwise comparison regarding J_T - Experiment 3.

vs		Unadjusted	Bonferroni	Holm	Shaffer	Rank diff.
IACTA-GDM/ODE	IACTA-GDM/OGA	**1.8422E−02**	6.4477E−01	2.3949E−01	2.3949E−01	1.67
IACTA-GDM/ODE	IACTA-GDM/OPSO	5.9346E−02	1.0000E+00	5.9346E−01	5.9346E−01	1.33
IACTA-GDM/ODE	SIACTA-RSM/1/ODE	**5.6573E−11**	**1.9801E−09**	**1.6972E−09**	**1.5840E−09**	−4.63
IACTA-GDM/ODE	SIACTA-RSM/1/OGA	**8.8646E−09**	**3.1026E−07**	**2.3934E−07**	**1.9502E−07**	−4.07
IACTA-GDM/ODE	SIACTA-RSM/1/OPSO	**1.5220E−06**	**5.3270E−05**	**3.9572E−05**	**3.3484E−05**	−3.40
IACTA-GDM/ODE	SIACTA-RSM/2/ODE	7.3239E−02	1.0000E+00	6.5915E−01	6.5915E−01	−1.27
IACTA-GDM/ODE	SIACTA-RSM/2/OGA	**4.1098E−05**	**1.4384E−03**	**8.6306E−04**	**8.6306E−04**	−2.90
IACTA-GDM/ODE	SIACTA-RSM/2/OPSO	**1.4234E−02**	4.9819E−01	2.1351E−01	2.1351E−01	−1.73
IACTA-GDM/OGA	IACTA-GDM/OPSO	6.3735E−01	1.0000E+00	1.0000E+00	1.0000E+00	−0.33
IACTA-GDM/OGA	SIACTA-RSM/1/ODE	**5.1242E−19**	**1.7935E−17**	**1.8447E−17**	**1.8447E−17**	−6.30
IACTA-GDM/OGA	SIACTA-RSM/1/OGA	**5.1393E−16**	**1.7988E−14**	**1.7474E−14**	**1.4390E−14**	−5.73
IACTA-GDM/OGA	SIACTA-RSM/1/OPSO	**7.7589E−13**	**2.7156E−11**	**2.4828E−11**	**2.1725E−11**	−5.07
IACTA-GDM/OGA	SIACTA-RSM/2/ODE	**3.3487E−05**	**1.1720E−03**	**7.3670E−04**	**7.3670E−04**	−2.93
IACTA-GDM/OGA	SIACTA-RSM/2/OGA	**1.0593E−10**	**3.7074E−09**	**3.0719E−09**	**2.9659E-09**	−4.57
IACTA-GDM/OGA	SIACTA-RSM/2/OPSO	**1.5220E−06**	**5.3270E−05**	**3.9572E−05**	**3.3484E−05**	−3.40
IACTA-GDM/OPSO	SIACTA-RSM/1/ODE	**3.2242E−17**	**1.1285E−15**	1.1285E−15	**9.0279E−16**	−5.97
IACTA-GDM/OPSO	SIACTA-RSM/1/OGA	**2.2277E−14**	**7.7969E−13**	**7.3513E−13**	**6.2375E−13**	−5.40
IACTA-GDM/OPSO	SIACTA-RSM/1/OPSO	**2.1723E−11**	**7.6032E−10**	**6.7343E−10**	**6.0826E−10**	−4.73
IACTA-GDM/OPSO	SIACTA-RSM/2/ODE	**2.3603E−04**	**8.2612E−03**	**4.2486E−03**	**4.2486E−03**	−2.60
IACTA-GDM/OPSO	SIACTA-RSM/2/OGA	**2.1396E−09**	**7.4886E−08**	**5.9909E−08**	**5.9909E−08**	−4.23
IACTA-GDM/OPSO	SIACTA-RSM/2/OPSO	**1.4449E−05**	**5.0572E−04**	**3.3233E−04**	**3.1788E−04**	−3.07
SIACTA-RSM/1/ODE	SIACTA-RSM/1/OGA	4.2291E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.57
SIACTA-RSM/1/ODE	SIACTA-RSM/1/OPSO	8.1125E−02	1.0000E+00	6.5915E−01	6.5915E−01	1.23
SIACTA-RSM/1/ODE	**SIACTA-RSM/2/ODE**	**1.9246E−06**	**6.7361E−05**	**4.6190E−05**	**4.2341E−05**	3.37
SIACTA-RSM/1/ODE	SIACTA-RSM/2/OGA	**1.4234E−02**	4.9819E−01	2.1351E−01	2.1351E−01	1.73
SIACTA-RSM/1/ODE	**SIACTA-RSM/2/OPSO**	**4.1098E−05**	**1.4384E−03**	**8.6306E−04**	**8.6306E−04**	2.90
SIACTA-RSM/1/OGA	SIACTA-RSM/1/OPSO	3.4578E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.67
SIACTA-RSM/1/OGA	**SIACTA-RSM/2/ODE**	**7.5013E−05**	**2.6255E−03**	**1.4253E−03**	**1.3502E−03**	2.80
SIACTA-RSM/1/OGA	SIACTA-RSM/2/OGA	9.8960E−02	1.0000E+00	6.9272E−01	6.9272E−01	1.17
SIACTA-RSM/1/OGA	**SIACTA-RSM/2/OPSO**	**9.6743E−04**	**3.3860E−02**	**1.6446E−02**	**1.5479E−02**	2.33
SIACTA-RSM/1/OPSO	SIACTA-RSM/2/ODE	**2.5530E−03**	8.9355E−02	**4.0848E−02**	**4.0848E−02**	2.13
SIACTA-RSM/1/OPSO	SIACTA-RSM/2/OGA	4.7950E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.50
SIACTA-RSM/1/OPSO	SIACTA-RSM/2/OPSO	**1.8422E−02**	6.4477E−01	2.3949E−01	2.3949E−01	1.67
SIACTA-RSM/2/ODE	SIACTA-RSM/2/OGA	**2.0895E−02**	7.3131E−01	2.3949E−01	2.3949E−01	−1.63
SIACTA-RSM/2/ODE	SIACTA-RSM/2/OPSO	5.0928E−01	1.0000E+00	1.0000E+00	1.0000E+00	−0.47
SIACTA-RSM/2/OGA	SIACTA-RSM/2/OPSO	9.8960E−02	1.0000E+00	6.9272E−01	6.9272E−01	1.17

Table C.29
Post-hoc pairwise comparison regarding J_T - Experiment 4.

vs		Unadjusted	Bonferroni	Holm	Shaffer	Rank diff.
IACTA-GDM/ODE	IACTA-GDM/OGA	9.2489E−01	1.0000E+00	1.0000E+00	1.0000E+00	−0.07
IACTA-GDM/ODE	IACTA-GDM/OPSO	8.5044E−01	1.0000E+00	1.0000E+00	1.0000E+00	−0.13
IACTA-GDM/ODE	SIACTA-RSM/1/ODE	**1.1703E−08**	**4.0961E−07**	**3.5110E−07**	**3.2769E−07**	−4.03
IACTA-GDM/ODE	SIACTA-RSM/1/OGA	**1.0593E−10**	**3.7074E−09**	**3.8133E−09**	**3.8133E−09**	−4.57
IACTA-GDM/ODE	SIACTA-RSM/1/OPSO	**2.6700E−10**	**9.3450E−09**	**9.0780E−09**	**7.4760E−09**	−4.47
IACTA-GDM/ODE	SIACTA-RSM/2/ODE	**1.9246E−06**	**6.7361E−05**	**4.0416E−05**	**4.0416E−05**	−3.37
IACTA-GDM/ODE	SIACTA-RSM/2/OGA	**1.6715E−07**	**5.8503E−06**	**4.5131E−06**	**3.6773E−06**	−3.70
IACTA-GDM/ODE	SIACTA-RSM/2/OPSO	**2.1549E−07**	**7.5423E−06**	**5.6028E−06**	**4.7409E−06**	−3.67
IACTA-GDM/OGA	IACTA-GDM/OPSO	9.2489E−01	1.0000E+00	1.0000E+00	1.0000E+00	−0.07
IACTA-GDM/OGA	SIACTA-RSM/1/ODE	**2.0266E−08**	**7.0932E−07**	**5.8772E−07**	**5.6745E−07**	−3.97
IACTA-GDM/OGA	SIACTA-RSM/1/OGA	**1.9662E−10**	**6.8816E−09**	**6.8816E−09**	**5.5052E−09**	−4.50
IACTA-GDM/OGA	SIACTA-RSM/1/OPSO	**4.8917E−10**	**1.7121E−08**	**1.5653E−08**	**1.3697E−08**	−4.40
IACTA-GDM/OGA	SIACTA-RSM/2/ODE	**3.0577E−06**	**1.0702E−04**	**6.1154E−05**	**5.5039E−05**	−3.30
IACTA-GDM/OGA	SIACTA-RSM/2/OGA	**2.7722E−07**	**9.7027E−06**	**6.9305E−06**	**6.0989E−06**	−3.63
IACTA-GDM/OGA	SIACTA-RSM/2/OPSO	**3.5586E−07**	**1.2455E−05**	**8.5407E−06**	**7.8290E−06**	−3.60
IACTA-GDM/OPSO	SIACTA-RSM/1/ODE	**3.4792E−08**	**1.2177E−06**	**9.7418E−07**	**9.7418E−07**	−3.90
IACTA-GDM/OPSO	SIACTA-RSM/1/OGA	**3.6179E−10**	**1.2663E−08**	**1.1939E−08**	**1.0130E−08**	−4.43
IACTA-GDM/OPSO	SIACTA-RSM/1/OPSO	**8.8846E−10**	**3.1096E−08**	**2.7542E−08**	**2.4877E−08**	−4.33
IACTA-GDM/OPSO	SIACTA-RSM/2/ODE	**4.8165E−06**	**1.6858E−04**	**9.1514E−05**	**8.6698E−05**	−3.23
IACTA-GDM/OPSO	SIACTA-RSM/2/OGA	**4.5583E−07**	**1.5954E−05**	**1.0484E−05**	**1.0028E−05**	−3.57
IACTA-GDM/OPSO	SIACTA-RSM/2/OPSO	**5.8263E−07**	**2.0392E−05**	**1.2818E−05**	**1.2818E−05**	−3.53
SIACTA-RSM/1/ODE	SIACTA-RSM/1/OGA	4.5070E−01	1.0000E+00	1.0000E+00	1.0000E+00	−0.53
SIACTA-RSM/1/ODE	SIACTA-RSM/1/OPSO	5.3999E−01	1.0000E+00	1.0000E+00	1.0000E+00	−0.43
SIACTA-RSM/1/ODE	SIACTA-RSM/2/ODE	3.4578E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.67
SIACTA-RSM/1/ODE	SIACTA-RSM/2/OGA	6.3735E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.33
SIACTA-RSM/1/ODE	SIACTA-RSM/2/OPSO	6.0408E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.37
SIACTA-RSM/1/OGA	SIACTA-RSM/1/OPSO	8.8754E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.10
SIACTA-RSM/1/OGA	SIACTA-RSM/2/ODE	8.9686E−02	1.0000E+00	1.0000E+00	1.0000E+00	1.20
SIACTA-RSM/1/OGA	SIACTA-RSM/2/OGA	2.2033E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.87
SIACTA-RSM/1/OGA	SIACTA-RSM/2/OPSO	2.0309E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.90
SIACTA-RSM/1/OPSO	SIACTA-RSM/2/ODE	1.1979E−01	1.0000E+00	1.0000E+00	1.0000E+00	1.10
SIACTA-RSM/1/OPSO	SIACTA-RSM/2/OGA	2.7826E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.77
SIACTA-RSM/1/OPSO	SIACTA-RSM/2/OPSO	2.5790E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.80
SIACTA-RSM/2/ODE	SIACTA-RSM/2/OGA	6.3735E−01	1.0000E+00	1.0000E+00	1.0000E+00	−0.33
SIACTA-RSM/2/ODE	SIACTA-RSM/2/OPSO	6.7137E−01	1.0000E+00	1.0000E+00	1.0000E+00	−0.30
SIACTA-RSM/2/OGA	SIACTA-RSM/2/OPSO	9.6240E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.03

Table C.30
Post-hoc pairwise comparison — Solvers time.

vs		Unadjusted	Bonferroni	Holm	Shaffer	Rank diff.
IACTA-GDM/ODE	IACTA-GDM/OGA	8.9686E−02	1.0000E+00	1.0000E+00	1.0000E+00	−1.20
IACTA-GDM/ODE	IACTA-GDM/OPSO	3.9614E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.60
IACTA-GDM/ODE	**SIACTA-RSM/1/ODE**	**5.1242E−19**	**1.8447E−17**	**1.7422E−17**	**1.4348E−17**	6.30
IACTA-GDM/ODE	**SIACTA-RSM/1/OGA**	**1.1649E−05**	**4.1935E−04**	**2.2132E−04**	**2.0968E−04**	3.10
IACTA-GDM/ODE	**SIACTA-RSM/1/OPSO**	**5.1242E−19**	**1.8447E−17**	**1.6910E−17**	**1.4348E−17**	6.30
IACTA-GDM/ODE	**SIACTA-RSM/2/ODE**	**9.9919E−08**	**3.5971E−06**	**2.2981E−06**	**2.1982E−06**	3.77
IACTA-GDM/ODE	SIACTA-RSM/2/OGA	**1.0909E−02**	3.9274E−01	1.4182E−01	3.9274E−01	1.80
IACTA-GDM/ODE	**SIACTA-RSM/2/OPSO**	**1.4447E−10**	**5.2010E−09**	**4.0452E−09**	**4.0452E−09**	4.53
IACTA-GDM/OGA	IACTA-GDM/OPSO	**1.0909E−02**	3.9274E−01	1.4182E−01	3.9274E−01	1.80
IACTA-GDM/OGA	**SIACTA-RSM/1/ODE**	**2.7766E−26**	**9.9959E−25**	**9.9959E−25**	**9.9959E−25**	7.50
IACTA-GDM/OGA	**SIACTA-RSM/1/OGA**	**1.1935E−09**	**4.2965E−08**	**2.9837E−08**	**2.6256E−08**	4.30
IACTA-GDM/OGA	**SIACTA-RSM/1/OPSO**	**2.7766E−26**	**9.9959E−25**	**9.7183E−25**	**7.7746E−25**	7.50
IACTA-GDM/OGA	**SIACTA-RSM/2/ODE**	**2.1572E−12**	**7.7658E−11**	**6.2558E−11**	**6.0401E−11**	4.97
IACTA-GDM/OGA	**SIACTA-RSM/2/OGA**	**2.2090E−05**	**7.9526E−04**	**3.9763E−04**	**3.9763E−04**	3.00
IACTA-GDM/OGA	**SIACTA-RSM/2/OPSO**	**5.1393E−16**	**1.8501E−14**	**1.6446E−14**	**1.4390E−14**	5.73
IACTA-GDM/OPSO	**SIACTA-RSM/1/ODE**	**7.5662E−16**	**2.7238E−14**	**2.3455E−14**	**2.1185E−14**	5.70
IACTA-GDM/OPSO	**SIACTA-RSM/1/OGA**	**4.0695E−04**	**1.4650E−02**	**5.6973E−03**	**5.2904E−03**	2.50
IACTA-GDM/OPSO	**SIACTA-RSM/1/OPSO**	**7.5662E−16**	**2.7238E−14**	**2.2699E−14**	**2.1185E−14**	5.70
IACTA-GDM/OPSO	**SIACTA-RSM/2/ODE**	**7.5225E−06**	**2.7081E−04**	**1.5045E−04**	**1.3540E−04**	3.17
IACTA-GDM/OPSO	SIACTA-RSM/2/OGA	8.9686E−02	1.0000E+00	1.0000E+00	1.0000E+00	1.20
IACTA-GDM/OPSO	**SIACTA-RSM/2/OPSO**	**2.6583E−08**	**9.5697E−07**	**6.3798E−07**	**5.8482E−07**	3.93
SIACTA-RSM/1/ODE	SIACTA-RSM/1/OGA	**6.0258E−06**	**2.1693E−04**	**1.3257E−04**	**1.3257E−04**	−3.20
SIACTA-RSM/1/ODE	SIACTA-RSM/1/OPSO	1.0000E+00	1.0000E+00	1.0000E+00	1.0000E+00	0.00
SIACTA-RSM/1/ODE	SIACTA-RSM/2/ODE	**3.4009E−04**	**1.2243E−02**	**5.4415E−03**	**5.4415E−03**	−2.53
SIACTA-RSM/1/ODE	SIACTA-RSM/2/OGA	**1.9662E−10**	**7.0782E−09**	**5.3086E−09**	**4.3256E−09**	−4.50
SIACTA-RSM/1/ODE	SIACTA-RSM/2/OPSO	**1.2474E−02**	4.4906E−01	1.6216E−01	4.4906E−01	−1.77
SIACTA-RSM/1/OGA	**SIACTA-RSM/1/OPSO**	**6.0258E−06**	**2.1693E−04**	**1.2654E−04**	**1.2654E−04**	3.20
SIACTA-RSM/1/OGA	SIACTA-RSM/2/ODE	3.4578E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.67
SIACTA-RSM/1/OGA	SIACTA-RSM/2/OGA	6.5992E−02	1.0000E+00	8.5790E−01	1.0000E+00	−1.30
SIACTA-RSM/1/OGA	SIACTA-RSM/2/OPSO	**4.2658E−02**	1.0000E+00	5.5456E−01	1.0000E+00	1.43
SIACTA-RSM/1/OPSO	SIACTA-RSM/2/ODE	**3.4009E−04**	**1.2243E−02**	**5.1014E−03**	**5.1014E−03**	−2.53
SIACTA-RSM/1/OPSO	SIACTA-RSM/2/OGA	**1.9662E−10**	**7.0782E−09**	**5.1120E−09**	**4.3256E−09**	−4.50
SIACTA-RSM/1/OPSO	SIACTA-RSM/2/OPSO	**1.2474E−02**	4.4906E−01	1.6216E−01	4.4906E−01	−1.77
SIACTA-RSM/2/ODE	SIACTA-RSM/2/OGA	**5.4144E−03**	1.9492E−01	7.0387E−02	7.0387E−02	−1.97
SIACTA-RSM/2/ODE	SIACTA-RSM/2/OPSO	2.7826E−01	1.0000E+00	1.0000E+00	1.0000E+00	0.77
SIACTA-RSM/2/OGA	**SIACTA-RSM/2/OPSO**	**1.1085E−04**	**3.9907E−03**	**1.8845E−03**	**1.7737E−03**	2.73

References

Abdullah, M. F., Siraj, S., & Hodgett, R. E. (2021). An overview of multi-criteria decision analysis (MCDA) application in managing water-related disaster events: Analyzing 20 years of literature for flood and drought events. *Water*, *13*(10), http://dx.doi.org/10.3390/w13101358, URL: https://www.mdpi.com/2073-4441/13/10/1358.

Aisuwarya, R., & Hidayati, Y. (2019). Implementation of Ziegler-Nichols PID tuning method on stabilizing temperature of hot-water dispenser. In *2019 16th International conference on quality in research (QIR): International symposium on electrical and computer engineering* (pp. 1–5). IEEE, http://dx.doi.org/10.1109/QIR.2019.8898259.

Al-Tashi, Q., Abdulkadir, S. J., Rais, H. M., Mirjalili, S., & Alhussian, H. (2020). Approaches to multi-objective feature selection: A systematic literature review. *IEEE Access*, *8*, 125076–125096. http://dx.doi.org/10.1109/ACCESS.2020.3007291.

Alavi, S. M. S., Akbarzadeh, A., & Farughian, A. (2011). Auto-tuning smith-predictive control of three-tanks system based on model reference adaptive system. In *The 2nd international conference on control, instrumentation and automation* (pp. 711–714). IEEE, http://dx.doi.org/10.1109/ICCIAutom.2011.6356746.

Aliman, N., Ramli, R., Mohamed Haris, S., Soleimani Amiri, M., & Van, M. (2022). A robust adaptive-fuzzy-proportional-derivative controller for a rehabilitation lower limb exoskeleton. *Engineering Science and Technology, An International Journal*, *35*, Article 101097. http://dx.doi.org/10.1016/j.jestch.2022.101097, URL: https://www.sciencedirect.com/science/article/pii/S2215098622000052.

Alyoussef, F., & Kaya, I. (2022). Simple PI-PD tuning rules based on the centroid of the stability region for controlling unstable and integrating processes. *ISA Transactions*, http://dx.doi.org/10.1016/j.isatra.2022.08.007.

Amini, E., Mehdipour, H., Faraggiana, E., Golbaz, D., Mozaffari, S., Bracco, G., et al. (2022). Optimization of hydraulic power take-off system settings for point absorber wave energy converter. *Renewable Energy*, *194*, 938–954. http://dx.doi.org/10.1016/j.renene.2022.05.164, URL: https://www.sciencedirect.com/science/article/pii/S0960148122008242.

Anitha, T., & Gopu, G. (2021). Controlled mechanical ventilation for enhanced measurement in pressure and flow sensors. *Measurement: Sensors*, *16*, Article 100054. http://dx.doi.org/10.1016/j.measen.2021.100054, URL: https://www.sciencedirect.com/science/article/pii/S2665917421000167.

Antal, P., Péni, T., & Tóth, R. (2022). Backflipping with miniature quadcopters by Gaussian process based control and planning. http://dx.doi.org/10.48550/arXiv.2209.14652, arXiv:2209.14652.

Arents, J., Abolins, V., Judvaitis, J., Vismanis, O., Oraby, A., & Ozols, K. (2021). Human–robot collaboration trends and safety aspects: A systematic review. *Journal of Sensor and Actuator Networks*, *10*(3), http://dx.doi.org/10.3390/jsan10030048, URL: https://www.mdpi.com/2224-2708/10/3/48.

Aruldoss, M., Lakshmi, T. M., & Venkatesan, V. P. (2013). A survey on multi criteria decision making methods and its applications. *American Journal of Information Systems*, *1*(1), 31–43. http://dx.doi.org/10.12691/ajis-1-1-5, URL: http://pubs.sciepub.com/ajis/1/1/5.

Åström, K. J., & Hägglund, T. (2006). *Advanced PID control*. ISA-The Instrumentation, Systems and Automation Society.

Azman, A. A., Rahiman, M. H. F., Mohammad, N. N., Marzaki, M. H., Taib, M. N., & Ali, M. F. (2017). Modeling and comparative study of PID ziegler nichols (ZN) and cohen-coon (CC) tuning method for multi-tube aluminum sulphate water filter (MTAS). In *2017 IEEE 2nd international conference on automatic control and intelligent systems* (pp. 25–30). IEEE, http://dx.doi.org/10.1109/I2CACIS.2017.8239027.

Benitez-Garcia, S. E., Villarreal-Cervantes, M. G., & Mezura-Montes, E. (2022). Event-triggered control optimal tuning through bio-inspired optimization in robotic manipulators. *ISA Transactions*, *128*, 81–105. http://dx.doi.org/10.1016/j.isatra.2021.10.029, URL: https://www.sciencedirect.com/science/article/pii/S0019057821005504.

Beudaert, X., Franco, O., Erkorkmaz, K., & Zatarain, M. (2020). Feed drive control tuning considering machine dynamics and chatter stability. *CIRP Annals*, *69*(1), 345–348. http://dx.doi.org/10.1016/j.cirp.2020.04.054.

Bhattacharya, A., Vasisht, S., Adetola, V., Huang, S., Sharma, H., & Vrabie, D. L. (2021). Control co-design of commercial building chiller plant using Bayesian optimization. *Energy and Buildings*, *246*, Article 111077. http://dx.doi.org/10.1016/j.enbuild.2021.111077, URL: https://www.sciencedirect.com/science/article/pii/S0378778821003613.

Bliek, L. (2022). A survey on sustainable surrogate-based optimisation. *Sustainability*, *14*(7), http://dx.doi.org/10.3390/su14073867, URL: https://www.mdpi.com/2071-1050/14/7/3867.

Borase, R. P., Maghade, D., Sondkar, S., & Pawar, S. (2021). A review of PID control, tuning methods and applications. *International Journal of Dynamics and Control*, *9*, 818–827. http://dx.doi.org/10.1007/s40435-020-00665-4.

Borwein, P., & Erdélyi, T. (1995). *Polynomials and polynomial inequalities, vol. 161*. Springer Science & Business Media, http://dx.doi.org/10.1007/978-1-4612-0793-1.

Boubakir, A., Labiod, S., & Boudjema, F. (2021). Direct adaptive fuzzy position controller for an electropneumatic actuator: Design and experimental evaluation. *Mechanical Systems and Signal Processing*, *147*, Article 107066. http://dx.doi.org/10.

1016/j.ymssp.2020.107066, URL: https://www.sciencedirect.com/science/article/pii/S0888327020304520.

Bowels, S., Xu, J., & Chen, H. (2015). Efficient controller parameter tuning for a system with disturbance. In *2015 IEEE international conference on robotics and biomimetics* (pp. 674–679). IEEE, http://dx.doi.org/10.1109/ROBIO.2015.7418846.

Breschi, V., van Meer, M., Oomen, T., & Formentin, S. (2021). On data-driven design of LPV controllers with flexible reference models. *IFAC-PapersOnLine*, *54*(8), 95–100. http://dx.doi.org/10.1016/j.ifacol.2021.08.587, URL: https://www.sciencedirect.com/science/article/pii/S240589632101363X, 4th IFAC Workshop on Linear Parameter Varying Systems LPVS 2021.

Brunton, S. L., & Kutz, J. N. (2022). *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press.

Bubnicki, Z. (2005). *Modern control theory*. Springer, http://dx.doi.org/10.1007/3-540-28087-1.

Büchler, D., Calandra, R., & Peters, J. (2019). Learning to control highly accelerated ballistic movements on muscular robots. http://dx.doi.org/10.48550/arXiv.1904.03665, CoRR abs/1904.03665, URL: http://arxiv.org/abs/1904.03665.

Cai, X., Su, S., Dai, P., Lin, L., & Lin, R. (2009). Constructive Lyapunov functions and stabilizing feedback for nonlinear systems. In *2009 Chinese control and decision conference* (pp. 341–343). IEEE, http://dx.doi.org/10.1109/CCDC.2009.5195076.

Chang, W.-D., Hwang, R.-C., & Hsieh, J.-G. (2002). Stable direct adaptive neural controller of nonlinear systems based on single auto-tuning neuron. *Neurocomputing*, *48*(1), 541–554. http://dx.doi.org/10.1016/S0925-2312(01)00627-0, URL: https://www.sciencedirect.com/science/article/pii/S0925231201006270.

Chelladurai, S. J. S., K., M., Ray, A. P., Upadhyaya, M., Narasimharaj, V., & S., G. (2021). Optimization of process parameters using response surface methodology: A review. *Materials Today: Proceedings*, *37*, 1301–1304. http://dx.doi.org/10.1016/j.matpr.2020.06.466, URL: https://www.sciencedirect.com/science/article/pii/S2214785320349440, International Conference on Newer Trends and Innovation in Mechanical Engineering: Materials Science.

Chen, C.-T. (1998). *Linear system theory and design* (3rd ed.). USA: Oxford University Press, Inc.

Chen, H., & Cheng, H. (2021). Online performance optimization for complex robotic assembly processes. *Journal of Manufacturing Processes*, *72*, 544–552. http://dx.doi.org/10.1016/j.jmapro.2021.10.047, URL: https://www.sciencedirect.com/science/article/pii/S1526612521007799.

Chen, Q., Ding, J., Yang, S., & Chai, T. (2020). A novel evolutionary algorithm for dynamic constrained multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, *24*(4), 792–806. http://dx.doi.org/10.1109/TEVC.2019.2958075.

Chen, H., & Xu, J. (2015). Exploring optimal controller parameters for complex industrial systems. In *2015 IEEE International conference on cyber technology in automation, control, and intelligent systems* (pp. 383–388). IEEE, http://dx.doi.org/10.1109/CYBER.2015.7287967.

Cohen, S. (2021). Chapter 1 - The evolution of machine learning: past, present, and future. In S. Cohen (Ed.), *Artificial intelligence and deep learning in pathology* (pp. 1–12). Elsevier, http://dx.doi.org/10.1016/B978-0-323-67538-3.00001-4, URL: https://www.sciencedirect.com/science/article/pii/B9780323675383000014.

Dang, Q. A., & Gostomski, P. A. (2021). Development of a feedback control system for a differential biofilter degrading toluene contaminated air. *Chemosphere*, *275*, Article 129822. http://dx.doi.org/10.1016/j.chemosphere.2021.129822, URL: https://www.sciencedirect.com/science/article/pii/S0045653521002915.

Deb, K., Agrawal, R. B., et al. (1995). Simulated binary crossover for continuous search space. *Complex Systems*, *9*(2), 115–148.

Derrac, J., García, S., Molina, D., & Herrera, F. (2011). A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm and Evolutionary Computation*, *1*(1), 3–18. http://dx.doi.org/10.1016/j.swevo.2011.02.002, URL: https://www.sciencedirect.com/science/article/pii/S2210650211000034.

Dong, S., Wang, P., & Abbas, K. (2021). A survey on deep learning and its applications. *Computer Science Review*, *40*, Article 100379. http://dx.doi.org/10.1016/j.cosrev.2021.100379, URL: https://www.sciencedirect.com/science/article/pii/S1574013721000198.

Edwards, C., & Spurgeon, S. (1998). *Sliding mode control: Theory and applications*. CRC Press.

Fan, S.-K. S., & Chiu, Y.-Y. (2007). A decreasing inertia weight particle swarm optimizer. *Engineering Optimization*, *39*(2), 203–228. http://dx.doi.org/10.1080/03052150601047362, arXiv:10.1080/03052150601047362.

Fang, H., Chen, L., & Li, X. (2011). Intelligent optimal tuning of hydraulic turbine governor PID gains based on nonlinear model. In *2011 International conference on computer science and service system* (pp. 1342–1345). IEEE, http://dx.doi.org/10.1109/CSSS.2011.5973969.

Faruq, A., Abdullah, S. S., Fauzi, M., & Nor, S. (2011). Optimization of depth control for unmanned underwater vehicle using surrogate modeling technique. In *2011 Fourth international conference on modeling, simulation and applied optimization* (pp. 1–7). IEEE, http://dx.doi.org/10.1109/ICMSAO.2011.5775543.

Fernandez-Gauna, B., Graña, M., Osa-Amilibia, J.-L., & Larrucea, X. (2022). Actor-critic continuous state reinforcement learning for wind-turbine control robust optimization. *Information Sciences*, *591*, 365–380. http://dx.doi.org/10.1016/j.ins.2022.01.047, URL: https://www.sciencedirect.com/science/article/pii/S0020025522000767.

Foley, M. W., Julien, R. H., & Copeland, B. R. (2005). A comparison of PID controller tuning methods. *The Canadian Journal of Chemical Engineering*, *83*(4), 712–722. http://dx.doi.org/10.1002/cjce.5450830412.

Frasnedo, S., Bect, J., Chapuis, C., Duc, G., Feyel, P., & Sandou, G. (2015). Line of sight controller tuning using Bayesian optimization of a high-level optronic criterion. *IFAC-PapersOnLine*, *48*(25), 56–61. http://dx.doi.org/10.1016/j.ifacol.2015.11.059, URL: https://www.sciencedirect.com/science/article/pii/S2405896315023174, 16th IFAC Workshop on Control Applications of Optimization CAO'2015.

Garcia, S., & Herrera, F. (2008). An extension on "Statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons. *Journal of Machine Learning Research*, *9*(12).

Geneva, N., & Zabaras, N. (2020). Modeling the dynamics of PDE systems with physics-constrained deep auto-regressive networks. *Journal of Computational Physics*, *403*, Article 109056. http://dx.doi.org/10.1016/j.jcp.2019.109056, URL: https://www.sciencedirect.com/science/article/pii/S0021999119307612.

Gholaminejad, T., Khaki-Sedigh, A., & Bagheri, P. (2016). Adaptive tuning of model predictive control based on analytical results. In *2016 4th International conference on control, instrumentation, and automation* (pp. 226–232). IEEE, http://dx.doi.org/10.1109/ICCIAutom.2016.7483165.

Gnambs, T., & Appel, M. (2019). Are robots becoming unpopular? Changes in attitudes towards autonomous robotic systems in Europe. *Computers in Human Behavior*, *93*, 53–61. http://dx.doi.org/10.1016/j.chb.2018.11.045, URL: https://www.sciencedirect.com/science/article/pii/S0747563218305806.

Gramacy, R. B. (2020). *Surrogates: Gaussian process modeling, design, and optimization for the applied sciences*. Chapman and Hall/CRC, http://dx.doi.org/10.1201/9780367815493.

Guangyou, Y. (2007). A modified particle swarm optimizer algorithm. In *2007 8th International conference on electronic measurement and instruments* (pp. 2–675). IEEE.

Gurung, S., Naetiladdanon, S., & Sangswang, A. (2021). A surrogate based computationally efficient method to coordinate damping controllers for enhancement of probabilistic small-signal stability. *IEEE Access*, *9*, 32882–32896. http://dx.doi.org/10.1109/ACCESS.2021.3060502.

Habbi, F., El Houda Gabour, N., Bounekhla, M., & Boudissa, E. G. (2021). Output voltage control of synchronous generator using Nelder –Mead algorithm based PI controller. In *2021 18th International multi-conference on systems, signals & devices* (pp. 365–374). IEEE, http://dx.doi.org/10.1109/SSD52085.2021.9429387.

Hambali, N., Masngut, A., Ishak, A. A., & Janin, Z. (2014). Process controllability for flow control system using Ziegler-Nichols (ZN), Cohen-Coon (CC) and Chien-Hrones-Reswick (CHR) tuning methods. In *2014 IEEE international conference on smart instrumentation, measurement and applications* (pp. 1–6). IEEE, http://dx.doi.org/10.1109/ICSIMA.2014.7047432.

Higashiyama, T., Mine, M., Ohmori, H., Sano, A., Nishida, H., & Todaka, Y. (2000). Auto-tuning of motor drive system by simple adaptive control approach. In *Proceedings of the 2000. IEEE international conference on control applications. Conference proceedings (Cat. no.00CH37162)* (pp. 868–873). IEEE, http://dx.doi.org/10.1109/CCA.2000.897551.

Hosamo, H. H., Tingstveit, M. S., Nielsen, H. K., Svennevig, P. R., & Svidt, K. (2022). Multiobjective optimization of building energy consumption and thermal comfort based on integrated BIM framework with machine learning-NSGA II. *Energy and Buildings*, *277*, Article 112479. http://dx.doi.org/10.1016/j.enbuild.2022.112479, URL: https://www.sciencedirect.com/science/article/pii/S0378778822006508.

Huang, C., Li, Y., & Yao, X. (2020). A survey of automatic parameter tuning methods for metaheuristics. *IEEE Transactions on Evolutionary Computation*, *24*(2), 201–216. http://dx.doi.org/10.1109/TEVC.2019.2921598.

Huang, T., Tang, X., Zhao, S., Zhang, Q., & Pedrycz, W. (2022). Linguistic information-based granular computing based on a tournament selection operator-guided PSO for supporting multi-attribute group decision-making with distributed linguistic preference relations. *Information Sciences*, *610*, 488–507. http://dx.doi.org/10.1016/j.ins.2022.07.050, URL: https://www.sciencedirect.com/science/article/pii/S0020025522007393.

Hull, D. G. (2013). *Optimal control theory for applications*. Springer Science & Business Media, http://dx.doi.org/10.1007/978-1-4757-4180-3.

Iuliano, E., & Pérez, E. A. (2016). *Application of surrogate-based global optimization to aerodynamic design*. Springer, http://dx.doi.org/10.1007/978-3-319-21506-8.

Jiang, J., Xia, C., Yao, J., Sun, Q., & Xia, H. (2023). Vulnerability analysis of HSR bridge under near-field blast based on response surface method. *Structures*, *55*, 983–994. http://dx.doi.org/10.1016/j.istruc.2023.06.053, URL: https://www.sciencedirect.com/science/article/pii/S235201242300810X.

Jiang, P., Zhou, Q., & Shao, X. (2020). *Surrogate model-based engineering design and optimization*. Springer, http://dx.doi.org/10.1007/978-981-15-0731-1.

Jiao, B., Lian, Z., & Gu, X. (2008). A dynamic inertia weight particle swarm optimization algorithm. *Chaos, Solitons & Fractals*, *37*(3), 698–705. http://dx.doi.org/10.1016/j.chaos.2006.09.063, URL: https://www.sciencedirect.com/science/article/pii/S0960077906009131.

Jones, D. R. (2001). A taxonomy of global optimization methods based on response surfaces. *Journal of Global Optimization*, *21*(4), 345–383. http://dx.doi.org/10.1023/A:1012771025575.

Joyce, T., & Herrmann, J. M. (2018). A review of no free lunch theorems, and their implications for metaheuristic optimisation. In X.-S. Yang (Ed.), *Nature-inspired algorithms and applied optimization* (pp. 27–51). Cham: Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-67669-2_2.

Kachitvichyanukul, V. (2012). Comparison of three evolutionary algorithms: GA, PSO, and DE. *Industrial Engineering and Management Systems*, *12*, 215–223. http://dx.doi.org/10.7232/iems.2012.11.3.215.

Khalil, H. K. (2002). *Nonlinear systems* (3rd ed.). Upper Saddle River, NJ, USA: Prentice-Hall.

Khuri, A. I. (2006). *Response surface methodology and related topics*. World scientific, http://dx.doi.org/10.1142/5915.

Khuri, A. I., & Mukhopadhyay, S. (2010). Response surface methodology. *Wiley Interdisciplinary Reviews: Computational Statistics*, *2*(2), 128–149. http://dx.doi.org/10.1002/wics.73.

Kim, D.-M., Lee, S.-G., Kim, D.-K., Park, M.-R., & Lim, M.-S. (2022). Sizing and optimization process of hybrid electric propulsion system for heavy-duty vehicle based on Gaussian process modeling considering traction motor characteristics. *Renewable and Sustainable Energy Reviews*, *161*, Article 112286. http://dx.doi.org/10.1016/j.rser.2022.112286, URL: https://www.sciencedirect.com/science/article/pii/S1364032122002052.

Kontes, G. D., Valmaseda, C., Giannakis, G. I., Katsigarakis, K. I., & Rovas, D. V. (2014). Intelligent BEMS design using detailed thermal simulation models and surrogate-based stochastic optimization. *Journal of Process Control*, *24*(6), 846–855. http://dx.doi.org/10.1016/j.jprocont.2014.04.003, URL: https://www.sciencedirect.com/science/article/pii/S0959152414000961, Energy Efficient Buildings Special Issue.

Koziel, S., & Leifsson, L. (2013). *Surrogate-based modeling and optimization*. Springer, http://dx.doi.org/10.1007/978-1-4614-7551-4.

Kramer, O., & Kramer, O. (2017). *Genetic algorithm essentials*. Springer, http://dx.doi.org/10.1007/978-3-319-52156-5.

Krishnan, R. (2017). *Permanent magnet synchronous and brushless DC motor drives*. CRC Press, http://dx.doi.org/10.1201/9781420014235.

Kudva, A., Sorouifar, F., & Paulson, J. A. (2022). Efficient robust global optimization for simulation-based problems using decomposed Gaussian processes: Application to MPC calibration. In *2022 American control conference* (pp. 2091–2097). IEEE, http://dx.doi.org/10.23919/ACC53348.2022.9867777.

Kuo, R., Su, P., Zulvia, F. E., & Lin, C. (2018). Integrating cluster analysis with granular computing for imbalanced data classification problem – A case study on prostate cancer prognosis. *Computers & Industrial Engineering*, *125*, 319–332. http://dx.doi.org/10.1016/j.cie.2018.08.031, URL: https://www.sciencedirect.com/science/article/pii/S036083521830411X.

Kuru, K., & Yetgin, H. (2019). Transformation to advanced mechatronics systems within new industrial revolution: A novel framework in automation of everything (AoE). *IEEE Access*, *7*, 41395–41415. http://dx.doi.org/10.1109/ACCESS.2019.2907809.

Lan, G., Tomczak, J. M., Roijers, D. M., & Eiben, A. (2022). Time efficiency in optimization with a bayesian-evolutionary algorithm. *Swarm and Evolutionary Computation*, *69*, Article 100970. http://dx.doi.org/10.1016/j.swevo.2021.100970.

Leavy, A. S. C., Xu, L., Filizadeh, S., & Gole, A. M. (2019). Simulation-based optimisation of LCC-HVDC controller parameters using surrogate model solvers. In *2019 20th Workshop on control and modeling for power electronics* (pp. 1–8). IEEE, http://dx.doi.org/10.1109/COMPEL.2019.8769718.

Leigh, J. R. (2004). *Control theory, vol. 64*. Iet, http://dx.doi.org/10.1049/PBCE064E.

Lima, F. S., Alves, V. M. C., & Araujo, A. C. B. (2020). Metacontrol: A Python based application for self-optimizing control using metamodels. *Computers & Chemical Engineering*, *140*, Article 106979. http://dx.doi.org/10.1016/j.compchemeng.2020.106979, URL: https://www.sciencedirect.com/science/article/pii/S0098135420303355.

Lin, J., Liu, H.-L., Tan, K. C., & Gu, F. (2021). An effective knowledge transfer approach for multiobjective multitasking optimization. *IEEE Transactions on Cybernetics*, *51*(6), 3238–3248. http://dx.doi.org/10.1109/TCYB.2020.2969025.

Liu, S., Meng, X., Yuan, Z., Ren, L., & Chen, L. (2023). Optimization design of space radiation cooler based on response surface method and genetic algorithm. *Case Studies in Thermal Engineering*, *50*, Article 103437. http://dx.doi.org/10.1016/j.csite.2023.103437, URL: https://www.sciencedirect.com/science/article/pii/S2214157X23007438.

Liu, Y., Patton, R. J., & Shi, S. (2022). Monte Carlo analysis of Bayesian optimization-based pitch controller with pitch fault compensation for offshore wind turbine. *IFAC-PapersOnLine*, *55*(6), 384–389. http://dx.doi.org/10.1016/j.ifacol.2022.07.159, URL: https://www.sciencedirect.com/science/article/pii/S2405896322005444, 11th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes SAFEPROCESS 2022.

Liu, K.-Z., & Yao, Y. (2016). *Robust control: Theory and applications*. John Wiley & Sons.

Lu, Q., González, L. D., Kumar, R., & Zavala, V. M. (2021). Bayesian optimization with reference models: A case study in MPC for HVAC central plants. *Computers & Chemical Engineering*, *154*, Article 107491. http://dx.doi.org/10.1016/j.compchemeng.2021.107491, URL: https://www.sciencedirect.com/science/article/pii/S0098135421002696.

Lü, W., Zhu, Y., Huang, D., Jiang, Y., & Jin, Y. (2010). A new strategy of integrated control and on-line optimization on high-purity distillation process. *Chinese Journal of Chemical Engineering*, *18*(1), 66–79. http://dx.doi.org/10.1016/S1004-9541(08)60325-0, URL: https://www.sciencedirect.com/science/article/pii/S1004954108603250.

Luckcuck, M., Farrell, M., Dennis, L. A., Dixon, C., & Fisher, M. (2019). Formal specification and verification of autonomous robotic systems: A survey. *ACM Computing Surveys*, *52*(5), 1–41. http://dx.doi.org/10.1145/3342355.

Makhamreh, H., Trabelsi, M., Kükrer, O., & Abu-Rub, H. (2021). A Lyapunov-based model predictive control design with reduced sensors for a PUC7 rectifier. *IEEE Transactions on Industrial Electronics*, *68*(2), 1139–1147. http://dx.doi.org/10.1109/TIE.2020.2969122.

Mamizadeh, A., Genc, N., & Rajabioun, R. (2018). Optimal tuning of PI controller for boost DC-DC converters based on cuckoo optimization algorithm. In *2018 7th International conference on renewable energy research and applications* (pp. 677–680). IEEE, http://dx.doi.org/10.1109/ICRERA.2018.8566883.

Marler, R. T., & Arora, J. S. (2004). Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization*, *26*, 369–395. http://dx.doi.org/10.1007/s00158-003-0368-6.

Martins, M. A., Rodrigues, A. E., Loureiro, J. M., Ribeiro, A. M., & Nogueira, I. B. (2021). Artificial intelligence-oriented economic non-linear model predictive control applied to a pressure swing adsorption unit: Syngas purification as a case study. *Separation and Purification Technology*, *276*, Article 119333. http://dx.doi.org/10.1016/j.seppur.2021.119333, URL: https://www.sciencedirect.com/science/article/pii/S138358662101042X.

McClement, D. G., Lawrence, N. P., Backström, J. U., Loewen, P. D., Forbes, M. G., & Gopaluni, R. B. (2022). Meta-reinforcement learning for the tuning of PI controllers: An offline approach. *Journal of Process Control*, *118*, 139–152. http://dx.doi.org/10.1016/j.jprocont.2022.08.002, URL: https://www.sciencedirect.com/science/article/pii/S0959152422001445.

Meshram, P., & Kanojiya, R. G. (2012). Tuning of PID controller using Ziegler–Nichols method for speed control of DC motor. In *IEEE-international conference on advances in engineering, science and management* (pp. 117–122). IEEE.

Miranda-Varela, M.-E., & Mezura-Montes, E. (2016). Surrogate-assisted differential evolution with an adaptive evolution control based on feasibility to solve constrained optimization problems. In *Proceedings of fifth international conference on soft computing for problem solving* (pp. 809–822). Springer, http://dx.doi.org/10.1007/978-981-10-0448-3_67.

Mirjalili, S. (2019). Genetic algorithm. In *Evolutionary algorithms and neural networks: Theory and applications* (pp. 43–55). Cham: Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-93025-1_4.

Miyani, P. B., & Sant, A. V. (2022). Bode diagram based control system design of three phase grid tied photovoltaic systems with Quasi-Z source inverter. *Materials Today: Proceedings*, http://dx.doi.org/10.1016/j.matpr.2022.03.697.

Mohanraj, D., Aruldavid, R., Verma, R., Sathiyasekar, K., Barnawi, A. B., Chokkalingam, B., et al. (2022). A review of BLDC motor: State of art, advanced control techniques, and applications. *IEEE Access*, *10*, 54833–54869. http://dx.doi.org/10.1109/ACCESS.2022.3175011.

Mousakazemi, S. M. H. (2021). Comparison of the error-integral performance indexes in a GA-tuned PID controlling system of a PWR-type nuclear reactor point-kinetics model. *Progress in Nuclear Energy*, *132*, Article 103604. http://dx.doi.org/10.1016/j.pnucene.2020.103604, URL: https://www.sciencedirect.com/science/article/pii/S0149197020303504.

Mowbray, M., Petsagkourakis, P., del Rio-Chanona, E., & Zhang, D. (2022). Safe chance constrained reinforcement learning for batch process control. *Computers & Chemical Engineering*, *157*, Article 107630. http://dx.doi.org/10.1016/j.compchemeng.2021.107630, URL: https://www.sciencedirect.com/science/article/pii/S0098135421004087.

Myers, R. H., Montgomery, D. C., & Anderson-Cook, C. M. (2016). *Response surface methodology: process and product optimization using designed experiments*. John Wiley & Sons.

Myers, R. H., Montgomery, D. C., Vining, G. G., & Robinson, T. J. (2012). *Generalized linear models: with applications in engineering and the sciences*. John Wiley & Sons, http://dx.doi.org/10.1002/9780470556986.

Nagy, M., Lăzăroiu, G., & Valaskova, K. (2023). Machine intelligence and autonomous robotic technologies in the corporate context of SMEs: Deep learning and virtual simulation algorithms, cyber-physical production networks, and industry 4.0-based manufacturing systems. *Applied Sciences*, *13*(3), http://dx.doi.org/10.3390/app13031681, URL: https://www.mdpi.com/2076-3417/13/3/1681.

Ortega, R., Praly, L., & Tang, Y. (1987). Direct adaptive tuning of robust controllers with guaranteed stability properties. *Systems & Control Letters*, *8*(4), 321–326. http://dx.doi.org/10.1016/0167-6911(87)90098-3, URL: https://www.sciencedirect.com/science/article/pii/0167691187900983.

Osman, B., & Zhu, H. (2017). Design of milling machine control system based on root locus method. In *2017 3rd IEEE international conference on control science and systems engineering* (pp. 141–144). http://dx.doi.org/10.1109/CCSSE.2017.8087911.

O'Sullivan, S., Nevejans, N., Allen, C., Blyth, A., Leonard, S., Pagallo, U., et al. (2019). Legal, regulatory, and ethical frameworks for development of standards in artificial intelligence (AI) and autonomous robotic surgery. *The International Journal of Medical Robotics and Computer Assisted Surgery*, *15*(1), Article e1968. http://dx.doi.org/10.1002/rcs.1968.

Ouyang, P., & Pano, V. (2015). Comparative study of DE, PSO and GA for position domain PID controller tuning. *Algorithms*, *8*(3), 697–711. http://dx.doi.org/10.3390/a8030697, URL: https://www.mdpi.com/1999-4893/8/3/697.

Pai, K. N., Nguyen, T. T., Prasad, V., & Rajendran, A. (2022). Experimental validation of an adsorbent-agnostic artificial neural network (ANN) framework for the design and optimization of cyclic adsorption processes. *Separation and Purification Technology*, *290*, Article 120783. http://dx.doi.org/10.1016/j.seppur.2022.120783, URL: https://www.sciencedirect.com/science/article/pii/S1383586622003422.

Pan, I., & Das, S. (2015). Kriging based surrogate modeling for fractional order control of microgrids. *IEEE Transactions on Smart Grid*, *6*(1), 36–44. http://dx.doi.org/10.1109/TSG.2014.2336771.

Pano, V., & Ouyang, P. (2014). Comparative study of GA, PSO, and DE for tuning position domain pid controller. In *2014 IEEE international conference on robotics and biomimetics* (pp. 1254–1259). IEEE, http://dx.doi.org/10.1109/ROBIO.2014.7090505.

Pareek, S., Kishnani, M., & Gupta, R. (2014). Application of artificial bee colony optimization for optimal pid tuning. In *2014 International conference on advances in engineering & technology research* (pp. 1–5). IEEE, http://dx.doi.org/10.1109/ICAETR.2014.7012817.

Parnianifard, A., Rezaie, V., Chaudhary, S., Imran, M. A., & Wuttisittikulkij, L. (2021). New adaptive surrogate-based approach combined swarm optimizer assisted less tuning cost of dynamic production-inventory control system. *IEEE Access*, *9*, 144054–144066. http://dx.doi.org/10.1109/ACCESS.2021.3122166.

Passino, K. M., Yurkovich, S., & Reinfrank, M. (1998). *Fuzzy control, vol. 42*. Citeseer.

Petrusev, A., Putratama, M. A., Rigo-Mariani, R., Debusschere, V., Reignier, P., & Hadjsaid, N. (2023). Reinforcement learning for robust voltage control in distribution grids under uncertainties. *Sustainable Energy, Grids and Networks*, *33*, Article 100959. http://dx.doi.org/10.1016/j.segan.2022.100959, URL: https://www.sciencedirect.com/science/article/pii/S2352467722002041.

Petsagkourakis, P., Sandoval, I. O., Bradford, E., Galvanin, F., Zhang, D., & del Rio-Chanona, E. A. (2022). Chance constrained policy optimization for process control and optimization. *Journal of Process Control*, *111*, 35–45. http://dx.doi.org/10.1016/j.jprocont.2022.01.003, URL: https://www.sciencedirect.com/science/article/pii/S0959152422000038.

Pinto, G., Deltetto, D., & Capozzoli, A. (2021). Data-driven district energy management with surrogate models and deep reinforcement learning. *Applied Energy*, *304*, Article 117642. http://dx.doi.org/10.1016/j.apenergy.2021.117642, URL: https://www.sciencedirect.com/science/article/pii/S0306261921010096.

Pirayeshshirazinezhad, R., Biedroń, S. G., Cruz, J. A. D., Güitrón, S. S., & Martínez-Ramón, M. (2022). Designing Monte Carlo simulation and an optimal machine learning to optimize and model space missions. *IEEE Access*, *10*, 45643–45662. http://dx.doi.org/10.1109/ACCESS.2022.3170438.

Price, D., Radaideh, M. I., & Kochunas, B. (2022). Multiobjective optimization of nuclear microreactor reactivity control system operation with swarm and evolutionary algorithms. *Nuclear Engineering and Design*, *393*, Article 111776. http://dx.doi.org/10.1016/j.nucengdes.2022.111776, URL: https://www.sciencedirect.com/science/article/pii/S0029549322001303.

Qian, E., Kramer, B., Peherstorfer, B., & Willcox, K. (2020). Lift & Learn: Physics-informed machine learning for large-scale nonlinear dynamical systems. *Physica D: Nonlinear Phenomena*, *406*, Article 132401. http://dx.doi.org/10.1016/j.physd.2020.132401, URL: https://www.sciencedirect.com/science/article/pii/S0167278919307651.

Rizk, Y., Awad, M., & Tunstel, E. W. (2019). Cooperative heterogeneous multi-robot systems: A survey. *ACM Computing Surveys*, *52*(2), 1–31. http://dx.doi.org/10.1145/3303848.

Rodríguez-Molina, A., Mezura-Montes, E., Villarreal-Cervantes, M. G., & Aldape-Pérez, M. (2020). Multi-objective meta-heuristic optimization in intelligent control: A survey on the controller tuning problem. *Applied Soft Computing*, *93*, Article 106342. http://dx.doi.org/10.1016/j.asoc.2020.106342.

Rodríguez-Molina, A., Villarreal-Cervantes, M. G., & Aldape-Pérez, M. (2017). An adaptive control study for a DC motor using meta-heuristic algorithms. *IFAC-PapersOnLine*, *50*(1), 13114–13120. http://dx.doi.org/10.1016/j.ifacol.2017.08.2164, URL: https://www.sciencedirect.com/science/article/pii/S2405896317328331, 20th IFAC World Congress.

Rodríguez-Molina, A., Villarreal-Cervantes, M. G., & Aldape-Pérez, M. (2019). An adaptive control study for the DC motor using meta-heuristic algorithms. *Soft Computing*, *23*, 889–906. http://dx.doi.org/10.1007/s00500-017-2797-y.

Rodríguez-Molina, A., Villarreal-Cervantes, M. G., Álvarez-Gallegos, J., & Aldape-Pérez, M. (2019). Bio-inspired adaptive control strategy for the highly efficient speed regulation of the DC motor under parametric uncertainty. *Applied Soft Computing*, *75*, 29–45. http://dx.doi.org/10.1016/j.asoc.2018.11.002, URL: https://www.sciencedirect.com/science/article/pii/S1568494618306288.

Rodríguez-Molina, A., Villarreal-Cervantes, M. G., Rueda-Gutiérrez, A.-B., Aldape-Pérez, M., Álvarez-Piedras, J. D., & Parra-Ocampo, M. F. (2023). Study of differential evolution variants in the dimensional synthesis of Four-Bar Grashof-Type mechanisms. *Applied Sciences*, *13*(12), http://dx.doi.org/10.3390/app13126966, URL: https://www.mdpi.com/2076-3417/13/12/6966.

Rodríguez-Molina, A., Villarreal-Cervantes, M. G., Serrano-Pérez, O., Solís-Romero, J., & Silva-Ortigoza, R. (2022). Optimal tuning of the speed control for brushless DC motor based on chaotic online differential evolution. *Mathematics*, *10*(12), http://dx.doi.org/10.3390/math10121977, URL: https://www.mdpi.com/2227-7390/10/12/1977.

Rojas López, A. G. (2020). *Sintonización óptima del controlador de un motor sin escobillas* (Ms thesis), Mexico City, Mexico: Centro de Innovación y Desarrollo Tecnológico en Cómputo del Instituto Politécnico Nacional (CIDETEC-IPN), Maestría en Tecnología de Cómputo.

Rojas-López, A. G., Villarreal-Cervantes, M. G., Rodríguez-Molina, A., & García-Mendoza, C. V. (2020). Offline optimum tuning of the proportional integral controller for speed regulation of a BLDC motor through bio-inspired algorithms. In *International congress of telematics and computing* (pp. 169–184). Springer, http://dx.doi.org/10.1007/978-3-030-62554-2_13.

Ross, S. M. (2020). *Introduction to probability and statistics for engineers and scientists*. Academic Press, http://dx.doi.org/10.1016/C2013-0-19397-X.

Roveda, L., Forgione, M., & Piga, D. (2020a). Robot control parameters auto-tuning in trajectory tracking applications. *Control Engineering Practice*, *101*, Article 104488. http://dx.doi.org/10.1016/j.conengprac.2020.104488, URL: https://www.sciencedirect.com/science/article/pii/S0967066120301143.

Roveda, L., Forgione, M., & Piga, D. (2020b). Two-stage robot controller auto-tuning methodology for trajectory tracking applications. *IFAC-PapersOnLine*, *53*(2), 8724–8731. http://dx.doi.org/10.1016/j.ifacol.2020.12.276, URL: https://www.sciencedirect.com/science/article/pii/S2405896320305541, 21st IFAC World Congress.

Sabug, L., Ruiz, F., & Fagiano, L. (2022). SMGO-δ: Balancing caution and reward in global optimization with black-box constraints. *Information Sciences*, *605*, 15–42. http://dx.doi.org/10.1016/j.ins.2022.05.017, URL: https://www.sciencedirect.com/science/article/pii/S0020025522004376.

Šafarič, J., Bencak, P., Fister, D., Šafarič, R., & Fister, I. (2020). Use of stochastic nature-inspired population-based algorithms within an online adaptive controller for mechatronic devices. *Applied Soft Computing*, *95*, Article 106559. http://dx.doi.org/10.1016/j.asoc.2020.106559, URL: https://www.sciencedirect.com/science/article/pii/S1568494620304981.

Sakunthala, S., Kiranmayi, R., & Mandadi, P. N. (2017). A study on industrial motor drives: Comparison and applications of pmsm and BLDC motor drives. In *2017 International conference on energy, communication, data analytics and soft computing* (pp. 537–540). http://dx.doi.org/10.1109/ICECDS.2017.8390224.

Schillinger, M., Hartmann, B., Skalecki, P., Meister, M., Nguyen-Tuong, D., & Nelles, O. (2017). Safe active learning and safe Bayesian optimization for tuning a PI-controller. *IFAC-PapersOnLine*, *50*(1), 5967–5972. http://dx.doi.org/10.1016/j.ifacol.2017.08.1258, URL: https://www.sciencedirect.com/science/article/pii/S240589631731772X, 20th IFAC World Congress.

Serrano-Pérez, O., Villarreal-Cervantes, M. G., González-Robles, J. C., & Rodríguez-Molina, A. (2020). Meta-heuristic algorithms for the control tuning of omnidirectional mobile robots. *Engineering Optimization*, *52*(2), 325–342. http://dx.doi.org/10.1080/0305215X.2019.1585834.

Serrano-Pérez, O., Villarreal-Cervantes, M. G., Rodríguez-Molina, A., & Serrano-Pérez, J. (2021). Offline robust tuning of the motion control for omnidirectional mobile robots. *Applied Soft Computing*, *110*, Article 107648. http://dx.doi.org/10.1016/j.asoc.2021.107648, URL: https://www.sciencedirect.com/science/article/pii/S156849462100569X.

Setchi, R., Dehkordi, M. B., & Khan, J. S. (2020). Explainable robotics in human-robot interactions. *Procedia Computer Science*, *176*, 3057–3066. http://dx.doi.org/10.1016/j.procs.2020.09.198, URL: https://www.sciencedirect.com/science/article/pii/S1877050920321001, Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 24th International Conference KES2020.

Shi, Y., & Eberhart, R. (1998). A modified particle swarm optimizer. In *1998 IEEE international conference on evolutionary computation proceedings. IEEE world congress on computational intelligence (Cat. no.98TH8360)* (pp. 69–73). IEEE, http://dx.doi.org/10.1109/ICEC.1998.699146.

Shi, Y., & Zhang, K. (2021). Advanced model predictive control framework for autonomous intelligent mechatronic systems: A tutorial overview and perspectives. *Annual Reviews in Control*, *52*, 170–196. http://dx.doi.org/10.1016/j.arcontrol.2021.10.008.

Shin, Y., Smith, R., & Hwang, S. (2020). Development of model predictive control system using an artificial neural network: A case study with a distillation column. *Journal of Cleaner Production*, *277*, Article 124124. http://dx.doi.org/10.1016/j.jclepro.2020.124124, URL: https://www.sciencedirect.com/science/article/pii/S095965262034169X.

Singh, P., & Dhiman, G. (2018). A hybrid fuzzy time series forecasting model based on granular computing and bio-inspired optimization approaches. *Journal of Computer Science*, *27*, 370–385. http://dx.doi.org/10.1016/j.jocs.2018.05.008, URL: https://www.sciencedirect.com/science/article/pii/S1877750317300923.

Sinha, S. K., Prasad, R., & Patel, R. N. (2009). PSO tuned combined optimal fuzzy controller for AGC of two area interconnected power system. In *2009 World congress on nature & biologically inspired computing* (pp. 537–542). IEEE, http://dx.doi.org/10.1109/NABIC.2009.5393392.

Smith, C. L. (2009). *Practical process control: Tuning and troubleshooting*. John Wiley & Sons.

Sobester, A., Forrester, A., & Keane, A. (2008). *Engineering design via surrogate modelling: A practical guide*. John Wiley & Sons, http://dx.doi.org/10.1002/9780470770801.

Song, Y., Wang, F., & Chen, X. (2019). An improved genetic algorithm for numerical function optimization. *Applied Intelligence*, *49*, 1880–1902. http://dx.doi.org/10.1007/s10489-018-1370-4.

Sorourifar, F., Makrygirgos, G., Mesbah, A., & Paulson, J. A. (2021). A data-driven automatic tuning method for MPC under uncertainty using constrained Bayesian optimization. *IFAC-PapersOnLine*, *54*(3), 243–250. http://dx.doi.org/10.1016/j.ifacol.2021.08.249, URL: https://www.sciencedirect.com/science/article/pii/S2405896321010223, 16th IFAC Symposium on Advanced Control of Chemical Processes ADCHEM 2021.

Spatola, N., Kühnlenz, B., & Cheng, G. (2021). Perception and evaluation in human–robot interaction: The Human–Robot Interaction Evaluation Scale (HRIES)—A multicomponent approach of anthropomorphism. *International Journal of Social Robotics*, *13*(7), 1517–1539. http://dx.doi.org/10.1007/s12369-020-00667-4.

Stenger, D., Ay, M., & Abel, D. (2020). Robust parametrization of a model predictive controller for a CNC machining center using Bayesian optimization. *IFAC-PapersOnLine*, *53*(2), 10388–10394. http://dx.doi.org/10.1016/j.ifacol.2020.12.2778, URL: https://www.sciencedirect.com/science/article/pii/S2405896320335412, 21st IFAC World Congress.

Stewart, J., Clegg, D. K., & Watson, S. (2020). *Calculus: Early transcendentals*. Cengage Learning.

Stock-Homburg, R. (2022). Survey of emotions in human–robot interactions: Perspectives from robotic psychology on 20 years of research. *International Journal of Social Robotics*, *14*(2), 389–411. http://dx.doi.org/10.1007/s12369-021-00778-6.

Suganya, P., Swaminathan, G., & Anoop, B. (2023). Pump availability prediction using response surface method in nuclear plant. *Nuclear Engineering and Technology*, http://dx.doi.org/10.1016/j.net.2023.09.002, URL: https://www.sciencedirect.com/science/article/pii/S1738573323004096.

Taha, W., Bakr, M., & Emadi, A. (2020). PI controller tuning optimization for grid-connected VSC using space mapping. In *2020 IEEE energy conversion congress and exposition* (pp. 5689–5695). IEEE, http://dx.doi.org/10.1109/ECCE44975.2020.9236416.

Tenne, Y., & Goh, C.-K. (2010). *Computational intelligence in expensive optimization problems, vol. 2*. Springer Science & Business Media, http://dx.doi.org/10.1007/978-3-642-10701-6.

van Niekerk, J., le Roux, J., & Craig, I. (2022). On-line automatic controller tuning using Bayesian optimisation - a bulk tailings treatment plant case study. *IFAC-PapersOnLine*, *55*(21), 126–131. http://dx.doi.org/10.1016/j.ifacol.2022.09.255, URL: https://www.sciencedirect.com/science/article/pii/S2405896322014896, 19th IFAC Symposium on Control, Optimization and Automation in Mining, Mineral and Metal Processing MMM 2022.

Villarreal-Cervantes, M. G., & Alvarez-Gallegos, J. (2016). Off-line PID control tuning for a planar parallel robot using DE variants. *Expert Systems with Applications*, *64*, 444–454. http://dx.doi.org/10.1016/j.eswa.2016.08.013.

Villarreal-Cervantes, M. G., Mezura-Montes, E., & Guzmán-Gaspar, J. Y. (2018). Differential evolution based adaptation for the direct current motor velocity control parameters. *Mathematics and Computers in Simulation*, *150*, 122–141. http://dx.doi.org/10.1016/j.matcom.2018.03.007, URL: https://www.sciencedirect.com/science/article/pii/S0378475418300685.

Villarreal-Cervantes, M. G., Rodríguez-Molina, A., García-Mendoza, C.-V., Peñaloza-Mejía, O., & Sepúlveda-Cervantes, G. (2017). Multi-objective on-line optimization approach for the DC motor controller tuning using differential evolution. *IEEE Access*, *5*, 20393–20407. http://dx.doi.org/10.1109/ACCESS.2017.2757959.

Villarreal-Cervantes, M. G., Rodríguez-Molina, A., & Serrano-Pérez, O. (2021). Novel asynchronous activation of the bio-inspired adaptive tuning in the speed controller: Study case in DC motors. *IEEE Access*, *9*, 138976–138993. http://dx.doi.org/10.1109/ACCESS.2021.3118658.

Walpole, R. E., Myers, R. H., Myers, S. L., & Ye, K. (1993). *Probability and statistics for engineers and scientists, vol. 5*. Macmillan New York.

Wang, J.-J., Jing, Y.-Y., Zhang, C.-F., & Zhao, J.-H. (2009). Review on multi-criteria decision analysis aid in sustainable energy decision-making. *Renewable and Sustainable Energy Reviews*, *13*(9), 2263–2278. http://dx.doi.org/10.1016/j.rser.2009.06.021, URL: https://www.sciencedirect.com/science/article/pii/S1364032109001166.

Wang, F.-S., Juang, W.-S., & Chan, C.-T. (1993). Optimal tuning of cascade PID control systems. In *Proceedings of IEEE international conference on control and applications, vol. 2* (pp. 825–828). IEEE, http://dx.doi.org/10.1109/CCA.1993.348228.

Wang, N., Liu, H., & Chen, W. (2013). Lyapunov-based excitation control for the synchronous generator unit. In *Proceedings of the 32nd Chinese control conference* (pp. 899–903). IEEE.

Wang, Z., Qiu, S., Song, R., Wang, X., Zhu, B., & Li, B. (2019). Research on PID parameter tuning of coordinated control for ultra-supercritical units based on ziegler nichols method. In *2019 IEEE 3rd advanced information management, communicates, electronic and automation control conference* (pp. 1155–1158). http://dx.doi.org/10.1109/IMCEC46724.2019.8984069.

Weisberg, S. (2005). *Applied linear regression, vol. 528*. John Wiley & Sons, http://dx.doi.org/10.1002/0471704091.

Wenge, L., Deyuan, L., Siyuan, C., Shaoming, L., & Zeyu, C. (2010). Optimal tuning PID controller for first order lag plus time delay system by election campaign optimization algorithm. In *2010 International conference on electrical and control engineering* (pp. 1430–1433). IEEE, http://dx.doi.org/10.1109/iCECE.2010.354.

Yang, X.-S. (2020). Nature-inspired optimization algorithms: Challenges and open problems. *Journal of Computer Science*, *46*, Article 101104. http://dx.doi.org/10.1016/j.jocs.2020.101104, URL: https://www.sciencedirect.com/science/article/pii/S1877750320300144, 20 years of computational science.

Yang, K., Emmerich, M., Deutz, A., & Bäck, T. (2019). Multi-Objective Bayesian Global Optimization using expected hypervolume improvement gradient. *Swarm and Evolutionary Computation*, *44*, 945–956. http://dx.doi.org/10.1016/j.swevo.2018.10.007, URL: https://www.sciencedirect.com/science/article/pii/S2210650217307861.

Yang, K., Gaida, D., Bäck, T., & Emmerich, M. (2015). Expected hypervolume improvement algorithm for PID controller tuning and the multiobjective dynamical control of a biogas plant. In *2015 IEEE congress on evolutionary computation* (pp. 1934–1942). IEEE, http://dx.doi.org/10.1109/CEC.2015.7257122.

Yang, X., Yuan, J., Yuan, J., & Mao, H. (2007). A modified particle swarm optimizer with dynamic adaptation. *Applied Mathematics and Computation*, *189*(2), 1205–1213. http://dx.doi.org/10.1016/j.amc.2006.12.045, URL: https://www.sciencedirect.com/science/article/pii/S0096300306017048.

Yarat, S., Senan, S., & Orman, Z. (2021). A comparative study on PSO with other metaheuristic methods. In *Applying particle swarm optimization: New solutions and cases for optimized portfolios* (pp. 49–72). Cham: Springer International Publishing, http://dx.doi.org/10.1007/978-3-030-70281-6_4.

Yen, S.-J., Chen, W.-L., Chen, C., Hsu, S.-C., & Wu, I.-C. (2014). An improved differential evolution algorithm with priority calculation for unit commitment problem. In *2014 IEEE international conference on granular computing* (pp. 352–357). http://dx.doi.org/10.1109/GRC.2014.6982863.

Yin, X., Zhang, W., Jiang, Z., & Pan, L. (2020). Data-driven multi-objective predictive control of offshore wind farm based on evolutionary optimization. *Renewable Energy*, *160*, 974–986. http://dx.doi.org/10.1016/j.renene.2020.05.015, URL: https://www.sciencedirect.com/science/article/pii/S0960148120307138.

Yin, X., Zhao, X., Lin, J., & Karcanias, A. (2020). Reliability aware multi-objective predictive control for wind farm based on machine learning and heuristic optimizations. *Energy*, *202*, Article 117739. http://dx.doi.org/10.1016/j.energy.2020.117739, URL: https://www.sciencedirect.com/science/article/pii/S036054422030846X.

Zacher, S. (2023). Controller tuning. In *Closed loop control and management: Introduction to feedback control theory with data stream managers* (pp. 81–123). Springer, http://dx.doi.org/10.1007/978-3-031-13483-8.

Zanasi, R., Cuoghi, S., & Ntogramatzidis, L. (2011). Analytical design of lead-lag compensators on Nyquist and Nichols planes. *IFAC Proceedings Volumes*, *44*(1), 7666–7671. http://dx.doi.org/10.3182/20110828-6-IT-1002.02758, URL: https://www.sciencedirect.com/science/article/pii/S1474667016448391, 18th IFAC World Congress.

Zhang, H., Lu, P., Ding, Z., Li, Y., Li, H., Hua, C., et al. (2022). Design optimization and control of dividing wall column for purification of trichlorosilane. *Chemical Engineering Science*, *257*, Article 117716. http://dx.doi.org/10.1016/j.ces.2022.117716, URL: https://www.sciencedirect.com/science/article/pii/S0009250922003001.

Zhang, B., Pedrycz, W., Fayek, A. R., & Dong, Y. (2022). A differential evolution-based consistency improvement method in AHP with an optimal allocation of information granularity. *IEEE Transactions on Cybernetics*, *52*(7), 6733–6744. http://dx.doi.org/10.1109/TCYB.2020.3035909.

Zhao, M., Alimo, S. R., Beyhaghi, P., & Bewley, T. R. (2019). Delaunay-based derivative-free optimization via global surrogates with safe and exact function evaluations. In *2019 IEEE 58th conference on decision and control* (pp. 4636–4641). IEEE, http://dx.doi.org/10.1109/CDC40024.2019.9029996.

Zhu, N., Gao, X.-T., & Huang, C.-Q. (2022). A data-driven approach for on-line auto-tuning of minimum variance PID controller. *ISA Transactions*, *130*, 325–342. http://dx.doi.org/10.1016/j.isatra.2022.04.001, URL: https://www.sciencedirect.com/science/article/pii/S0019057822001549.

Zhu, M., Piga, D., & Bemporad, A. (2022). C-GLISp: Preference-based global optimization under unknown constraints with applications to controller calibration. *IEEE Transactions on Control Systems Technology*, *30*(5), 2176–2187. http://dx.doi.org/10.1109/TCST.2021.3136711.