Research Article

Slime Mould Algorithm-Based Tuning of Cost-Effective Fuzzy Controllers for Servo Systems

Radu-Emil Precup[1,*], Radu-Codrut David[1], Raul-Cristian Roman[1], Emil M. Petriu[2], Alexandra-Iulia Szedlak-Stinean[1]

[1]*Department of Automation and Applied Informatics, Politehnica University of Timisoara, Bd. V. Parvan 2, 300223 Timisoara, Romania*
[2]*School of Electrical Engineering and Computer Science, University of Ottawa, 800 King Edward, Ottawa, ON K1N 6N5, Canada*

ARTICLE INFO

ABSTRACT

This paper suggests five new contributions with respect to the state-of-the-art. First, the optimal tuning of cost-effective fuzzy controllers represented by Takagi–Sugeno–Kang proportional-integral fuzzy controllers (TSK PI-FCs) is carried out using a fresh metaheuristic algorithm, namely the Slime Mould Algorithm (SMA), and a fuzzy controller tuning approach is offered. Second, a relatively easily understandable formulation of SMA is offered. Third, a real-world application of SMA is given, focusing on the optimal tuning of TSK PI-FCs for nonlinear servo systems in terms of optimization problems that target the minimization of discrete-time cost functions defined as the sum of time multiplied by squared control error. Fourth, using the concept of improving the performance of metaheuristic algorithms with information feedback models, proposed by Wang and Tan, Improving metaheuristic algorithms with information feedback models, IEEE Trans. Cybern. 49 (2019), 542–555, Gu and Wang, Improving NSGA-III algorithms with information feedback models for large-scale many-objective optimization, Fut. Gen. Comput. Syst. 107 (2020), 49–69, and Zhang *et al.*, Enhancing MOEA/D with information feedback models for large-scale many-objective optimization, Inf. Sci. 522 (2020), 1–16, new metaheuristic algorithms are introduced in terms of inserting the model F1 in SMA and other representative algorithms, namely Gravitational Search Algorithm (GSA), Charged System Search (CSS), Grey Wolf Optimizer (GWO) and Whale Optimization Algorithm (WOA). Fifth, the real-time validation of the cost-effective fuzzy controllers and their tuning approach is performed in the framework of angular position control of laboratory servo system. The comparison with other metaheuristic algorithms that solve the same optimization problem for optimal parameter tuning of cost-effective fuzzy controllers suggestively highlights the superiority of SMA. Experimental results are included.

1. INTRODUCTION

Nonlinear models characterize the majority of physical systems and industrial processes, inspiring others to develop suitable approaches for the analysis of nonlinear systems. The rapidly increasing research on fuzzy control that copes with nonlinear systems has taken place in recent decades. This is important because fuzzy control is a robust and inexpensive mathematical approach to control highly complex nonlinear or nonanalytical systems in many industrial applications, and can be considered as a relatively simple initial nonlinear and convenient control approach. As shown by Precup *et al.* [1] and Precup and David [2], respectively, the systematic design and tuning of fuzzy control systems is supported by analyses that include stability, controllability, observability, sensitivity and robustness. The optimal tuning of fuzzy controllers is combined with these analyses. These analyses are viewed in the context of one of the hot research topic in the recent years, namely the development of the improvement of fuzzy systems performance, owing

to their extensively growing applications. To improve the performance of fuzzy control systems, type-1 fuzzy control systems were also extended and generalized to type-2 and type-3 ones especially focusing on interval ones in this regard.

Metaheuristic algorithms are successfully applied because they ensure higher performance and require lower computing capacity and time *versus* deterministic algorithms in several optimization problems. Such challenging optimization problems are those specific to the optimal (parameter) tuning of fuzzy (logic) controllers, where both the process and the controller are nonlinear and deterministic algorithms are not successful. The following metaheuristic algorithms have been applied most recently to the optimal tuning of fuzzy controllers in representative examples: adaptive weight Genetic Algorithm (GA) for gear shifting control [3], GA-based multiobjective optimization for electric vehicle powertrain control [4], GA for hybrid power systems control [5], engines control [6], energy management in hybrid vehicles [7], servo system control [2], wellhead back pressure control systems [8], micro-unmanned helicopter control [9], Particle Swarm Optimization (PSO) algorithm with compensating coefficient of inertia weight factor

Corresponding author. Email: radu.precup@aut.upt.ro

for filter time constant adaptation in hybrid energy storage systems control [10], set-based PSO algorithm with adaptive weights for optimal path planning of unmanned aerial vehicles [11], PSO algorithm for zinc production [12] and inverted pendulum control [13], hybrid PSO-Artificial Bee Colony algorithm for frequency regulation in microgrids [14], Imperialist Competitive Algorithm for human immunodeficiency control [15], Grey Wolf Optimizer (GWO) algorithms for sun-tracker systems [16] and servo system control [2], PSO, Cuckoo Search and Differential Evolution (DE) for gantry crane systems position control [17], Whale Optimization Algorithm (WOA) for vibration control of steel structures [18], Grasshopper Optimization Algorithm for load frequency control [19], DE for electro-hydraulic servo system control [20], Gravitational Search Algorithm (GSA) and Charged System Search (CSS) for servo system control [2].

Slime Mould Algorithm (SMA) is a fresh metaheuristic algorithm proposed in [21]. It mimics the oscillation mode of slime mould (i.e., the Physarum polycephalum) in nature when producing positive and negative propagation feedback in the path toward the food. Compared to other metaheuristic algorithms, it is proved in [21] that SMA exhibits improved exploratory and exploitation features.

Other popular metaheuristic algorithms with fresh results are hybrid PSO-GA [22] and hybrid GSA-GA [23] for constrained solutions, water cycle [24] and bat [25] algorithms for combinatorial optimization, GWO with hierarchical fuzzy operator [26], hierarchical GA multiobjective optimization of neural networks [27], Cross-Entropy algorithm for manufacturing processes [28], Water Circle algorithm for traveling salesman problem [29], Conflict Monitoring algorithm [30], Jaya optimization algorithm for biped robots [31] and load forecast [32] and Chemical Reaction algorithm for community detection [33].

Some of the most recently proposed metaheuristic algorithms concern both the improvement of other algorithms and the development of new ones. Suggestive approaches are the improvement of Nondominated Sorting Genetic Algorithm-III (NSGA-III) algorithm with adaptive mutation operator and the application of this improved NSGA-III to Big Data tuning [34], the analysis of the behavior of crossover operators in NSGA-III algorithm and the operator improvement such that to work in large-scale optimization [35], the improvement of DE algorithm by a selection mechanism and its application to fuzzy job-shop scheduling problems [36], and population extremal optimization algorithms applied to continuous optimization and nonlinear controller optimal tuning problems [37–39].

An important and general improvement of metaheuristic algorithms in terms of modifying their structure by including information feedback models is proposed and formulated generally by Wang and Tan [40]. Six types of information feedback models are defined in [40], where individuals from previous iterations are selected in either a fixed or random manner and embedded in the update process of the algorithms. A detailed exemplification on PSO is offered in [40], but experimental applications to other representative algorithms are included as well. This concept is included successfully by Gu and Wang in NSGA-III [41] and Zhang et al. in multiobjective evolutionary algorithms based on decomposition in [42], and has a big potential to be incorporated in other algorithms.

The optimal tuning of fuzzy controller can also be seen, discussed and investigated in the more general and generous context of

fuzzy modeling. The estimation capability of fuzzy systems depends on their rule parameters and structure of fuzzy sets. Up to now, many learning methods have been developed for optimizing rule parameters, i.e., which belongs to topic of optimal tuning of parameters of fuzzy model structures. Moreover, evolving fuzzy systems are also a popular topic as they also learn and optimally build the structure of fuzzy systems along with their parameters. Also, research is also focused on the tuning rules, which can be extracted such that to adapt the structure of the fuzzy sets, but these approaches usually also require that the output derivative of fuzzy systems to be computed with respect to the parameters which are actually learned and specifically tuned.

This paper is built upon authors' recent papers on the optimal tuning of Takagi–Sugeno–Kang proportional-integral fuzzy controllers (TSK PI-FCs) [1,2,43–45] applied to servo systems control such that to obtain a reduced parametric sensitivity (with respect to process gain and time constants), and suggests a novel SMA-based tuning approach. The approach is focused on the position control of nonlinear servo systems viewed as controlled processes. SMA is involved in solving a minimization-type optimization problem, which involves a cost function that is equal to the sum of time multiplied by squared control error, representing the discrete-time version of the Integral of Time Multiplied by Squared Error (ITSE) used in continuous-time control.

The paper suggests the following new contributions:

- a fuzzy controller tuning approach,

- a relatively easily understandable formulation of SMA,

- a real-world application of SMA,

- using the concept of improving the performance of metaheuristic algorithms with information feedback models [40–42], new metaheuristic algorithms are proposed by inserting the model F1 in SMA and other representative algorithms, namely GSA, CSS, GWO and WOA,

- the real-time validation of cost-effective fuzzy controllers and their tuning approach.

These contributions are significant and also advantageous in the context of the state-of-the-art briefly discussed in this section as the superiority of SMA *versus* other optimization algorithms, namely PSO, GSA, CSS, GWO and WOA, is proved by means a comparison included in the paper. The comparison is supported by experimental results on a real-world application, namely angular position control of nonlinear servo system; real data from authors' lab is included.

The next sections in the sequel are structured as follows: the optimization problem is defined in Section 2, and the associated models of the process and fuzzy controller are specified. SMA and, based on this algorithm, the new tuning approach that produces optimal TSK PI-FCs are presented in Section 3. Starting with the introduction of the information feedback model F1 in PSO in [40], details on the new metaheuristic algorithms obtained by inserting the information feedback model F1 in GSA, CSS, GWO and WOA are given in Section 4. The tuning approach is validated in Section 5 in terms of offering experimental results and a comparison as well. The conclusions are drawn in Section 6.

2. PROCESS MODELS, CONTROLLER MODELS AND OPTIMIZATION PROBLEM

The control system structure is illustrated in Figure 1. Figure 1 represents an extended version of the block diagram and description of sub-systems given in [46], highlighting that a continuous-time process is controlled ($t \geq 0$ indicates the continuous time) by a digital control ($t_d \in \mathbf{Z}$, $t_d \geq 0$ is the discrete-time index).

The sub-systems and variables involved in Figure 1 are [46]: F – the setpoint (or the reference input) filter, FC – the fuzzy controller, which is represented by the TSK PI-FC, PWM-DAC – the Pulse Width Modulated (PWM) Digital to Analog Converter that produces the analog voltage applied to the process P (in fact, the actuator), ADC – the Analog to Digital Converter needed for digital process control, r – the reference input, r_1 – the reference input filtered through F, y – the controlled output, u – the control signal, and $e = r - y$ – the control error. The sub-system F is included in Figure 1 because this control system is a setpoint filter type two-degree-of-freedom (2-DOF) structure, which usually operates with Proportional-Integral (PI) or Proportional-Integral-Derivative (PID) linear controllers, but nonlinear TSK PI-FCs will be used in this paper. Disturbance inputs are not included in Figure 1 because the integral component of the controller copes with load-type disturbance inputs. The presentation in this section is based on authors' past papers as, e.g., [2,43–46], therefore figures and equations are reused in this regard.

The servo system that plays the role of P in Figure 1 is characterized by the continuous-time state space model

$$m(t) = \begin{cases} -1, & \text{if } u(t) \leq -u_b, \\ \left[u(t) + u_c \right] / \left(u_b - u_c \right), & \text{if } -u_b < u(t) < -u_c, \\ 0, & \text{if } -u_c \leq |u(t)| \leq u_a, \\ \left[u(t) - u_a \right] / \left(u_b - u_a \right), & \text{if } u_a < u(t) < u_b, \\ 1, & \text{if } u(t) \geq u_b, \end{cases}$$

(1)

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -1/T_\Sigma \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ k_P/T_\Sigma \end{bmatrix} m(t),$$

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) & x_2(t) \end{bmatrix}^T,$$

where the dynamics of both measurement instrumentation and actuator are included in P, the parameters and variables represent [46]: $k_P > 0$ – the servo system (or process) gain, $T_\Sigma > 0$ – the small time constant of the process, $u(t)$ – the control signal as a PWM signal within the interval (due to the PWM duty cycle) $-1 \leq u(t) \leq 1$, $x_1(t) = \alpha(t)$ (rad) – the angular position, $x_2(t) = \omega(t)$ (rad/s) – the angular speed, $m(t)$ – the output of the saturation and dead zone
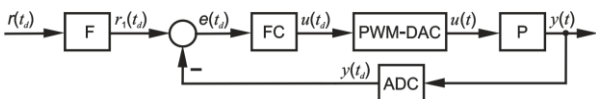

Figure 1 | Setpoint filter type two-degree-of-freedom (2-DOF) fuzzy control system structure.

static nonlinearity, T – matrix transposition, and the parameters of the nonlinearity fulfill the conditions $0 < u_a < u_b$ and $0 < u_c < u_b$.

As shown in [2,44–46], the nonlinearity in (1) is neglected in order to enable the cost-effective linear and nonlinear (including fuzzy) controller design. The transfer function of this simplified model of P, which is next used in the controller design and tuning, is

$$P(s) = k_{EP} / \left[s \left(1 + T_\Sigma s \right) \right],$$

(2)

where k_{EP} is the equivalent process gain:

$$k_{EP} = \begin{cases} k_P / (u_b - u_c), & \text{if } -u_b < u(t) < -u_c, \\ k_P / (u_b - u_a), & \text{if } u_a < u(t) < u_b. \end{cases}$$

(3)

The servo system process parameters are: $k_{EP} = 170, T_\Sigma = 0.76 \text{ s}, u_a = 0.15, u_b = 1$ and $u_c = 0.15$.

PI controllers are recommended in [47,48] for the process modeled in (2). The transfer function of the PI controller (to be replaced with FC in Figure 1) is

$$C(s) = k_c \left(1 + s T_i \right) / s = k_C \left[1 + 1 / \left(s T_i \right) \right], k_C = k_c T_i,$$

(4)

where $k_C > 0$ or $k_c > 0$ is the controller gain and $T_i > 0$ is the controller integral time constant.

The Extended Symmetrical Optimum (ESO) method suggested in [47,48] is recommended to be applied in order to carried out the PI controller tuning as it guarantees a trade-off to a set of empirical control system quality indices: percent overshoot, settling time, rise time and phase margin. That trade-off is ensured by a single design parameter with the notation β and the recommended domain $1 < \beta \leq 20$. The PI tuning conditions specific to the ESO method are [47,48]

$$k_c = 1 / \left(\beta \sqrt{\beta} \, k_{EP} \, T_\Sigma^2 \right), T_i = \beta \, T_\Sigma, k_C = 1 / \left(\sqrt{\beta} k_{EP} \, T_\Sigma \right),$$

(5)

and the transfer function of the setpoint filter F, which additionally helps the control system performance enhancement by canceling a zero in the closed-loop transfer function (in the linear case) with respect to the setpoint, is

$$F(s) = 1 / \left(1 + \beta T_\Sigma s \right).$$

(6)

The quality of the control system is increased by inserting the cost-effective nonlinear controller represented by TSK PI-FC in terms of the structure given in Figure 1. Incidentally, the design and tuning start with employing the knowledge and experience gained in terms of experimenting PI controllers in various applications. The structure and input membership functions of TSK PI-FC are presented in Figure 2, where [46]: q^{-1} – the backward shift operator, TISO-FC – the Two Inputs-Single Output Fuzzy Controller sub-system (it is actually a nonlinear sub-system without dynamics as it is introduced outside), $e(t_d)$ – control error, $u(t_d)$ – control signal, $\Delta e(t_d)$ – $e(t_d)$ variation, and $\Delta u(t_d)$ – $u(t_d)$ variation.

The expressions of the two increments illustrated in Figure 2 are obtained by Tustin's discretization method that leads to the recurrent equation of the incremental discrete-time PI controller

$$\Delta u(t_d) = K_P \left[\Delta e(t_d) + \mu e(t_d) \right],$$

(7)

where K_P and μ are

$$K_P = k_c \left(T_i - T_s/2 \right), \mu = 2\,T_s / \left(2\,T_i - T_s \right), \qquad (8)$$

where $T_s > 0$ is the sampling period, which has to be set such that to have a quasi-continuous digital control implementation.

As considered in [47], TISO-FC in Figure 2 uses for defuzzification a weighted average based approach, while the inference engine employs SUM and PROD operators. The rule base of this subsystem consists of only two rules for the sake of cost-effective fuzzy control design and implementation [2,43–46], and it is expressed in Table 1, where the parameter η, $0.25 < \eta < 0.75$, is inserted for overshoot diminution.

The parameter η makes the difference in the rule consequents, which consist of two different linear discrete-time PI controllers. This structure of the fuzzy controller makes it behave as a bumpless interpolator between the two linear PI controllers in the rule consequents.

The modal equivalence principle [49] is applied to express one tuning equation for TSK PI-FC

$$B_{\Delta e} = \mu\,B_e, \qquad (9)$$

so the parameters of TSK PI-FC are grouped in the parameter vector ρ:

$$\rho = \left[\beta \ B_e \ \eta \right]^T \qquad (10)$$

of only three elements (the first one from the linear part and the last two ones from the TISO-FC part), illustrating that cost-effective fuzzy control is targeted.
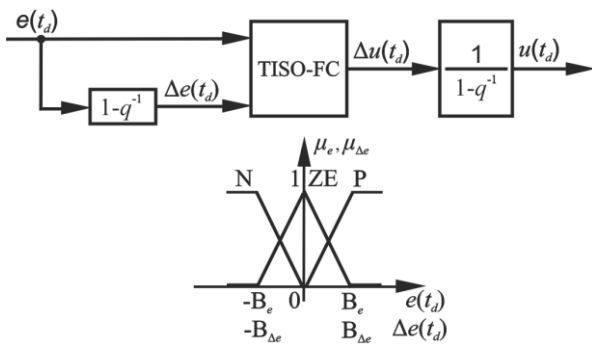

Figure 2 | Takagi--Sugeno--Kang proportional-integral fuzzy controllers (TSK PI-FC) configuration.

As specified in Section 1, the optimization problem is defined as

$$\rho^* = \arg \min_{\rho \in D_\rho} J(\rho), J(\rho) = \sum_{t_d=0}^{\infty} \left[t_d e^2(t_d, \rho) \right], \qquad (11)$$

where J is the cost function, which is expressed as the sum of discrete times multiplied by squared control errors, ρ^* is the optimal parameter vector

$$\rho^* = \left[\beta^* \ B_e^* \ \eta^* \right]^T, \qquad (12)$$

and D_ρ is the feasible domain of ρ. Several recommendations are given in [2,43–46] to set to set the domain D_ρ, but the most important ones concern technical and economical issues including stability constraints [50–55]. Although the stability analysis appears to be the source of many other synthesis issues, it is always important so less conservative stability conditions need to be established. Therefore, one of the first objectives of the stability investigation is to evaluate the maximum permissible parameter variations in order to sustain the delay model's stability. In addition, that should be analyzed and discussed in relation to the accuracy of the controlled process model in (1) and also the accuracy of the cost function model in (11). Different approaches are used in case of model-free data-driven control. Following, SMA is inserted in solving (11).

3. SMA AND FUZZY CONTROLLER TUNING APPROACH

The operating mechanism of SMA described in accordance with the standard formulation introduced in [21] starts with the random initialization of the population of agents, i.e., agent positions in the slime mould, such that to belong to the search domain D_s. A total number of N agents (i.e., elements of the slime mould) is used, and each agent is assigned to a position vector $X_i(k)$

$$X_i(k) = \left[x_i^1(k) \ldots x_i^f(k) \ldots x_i^q(k) \right]^T \in D_s, i = 1 \ldots \text{N}, \qquad (13)$$

where [46] $x_i^f(k)$ is the position of ith agent in f^{th} dimension, $f = 1 \ldots q$, k is the index of the current iteration, $k = 1 \ldots k_{\max}$, and $k_{\max}$ is the maximum number of iterations. The expression of the search domain is

$$D_s = \left[l^1, u^1 \right] \times \ldots \times \left[l^f, u^f \right] \times \ldots \times \left[l^q, u^q \right] \subset \mathfrak{R}^q, \qquad (14)$$

where l^f are the lower bounds, u^f are the upper bounds, $f = 1 \ldots q$, and

$$x_i^f(k) \in \left[l^f, u^f \right], i = 1 \ldots \text{N}, f = 1 \ldots q. \qquad (15)$$

Table 1 | Rule base of TISO-FC.

$\dfrac{\Delta e(t_d)}{e(t_d)}$	N	ZE	P
P	$K_P[\Delta e(t_d) + \mu\,e(t_d)]$	$K_P[\Delta e(t_d) + \mu\,e(t_d)]$	$\eta\,K_P[\Delta e(t_d) + \mu\,e(t_d)]$
ZE	$K_P[\Delta e(t_d) + \mu\,e(t_d)]$	$K_P[\Delta e(t_d) + \mu\,e(t_d)]$	$K_P[\Delta e(t_d) + \mu\,e(t_d)]$
N	$\eta\,K_P[\Delta e(t_d) + \mu\,e(t_d)]$	$K_P[\Delta e(t_d) + \mu\,e(t_d)]$	$K_P[\Delta e(t_d) + \mu\,e(t_d)]$

The search process specific to SMA continues with approaching, wrapping and grabbling the food. This is modeled by several equations. Using the notation $S_i(k)$ for the fitness (i.e., cost function in the context of (11)) of ith agent with the position vector $X_i(k)$, the population is ranked in the ascending order of the fitness function values (i.e., cost function values in the context of (11)) for optimization problems that target the minimization of the cost functions. The following sets of position vectors are obtained:

$$\begin{aligned} &\{X_1(k), X_2(k), \dots, X_H(k), X_{H+1}(k), \dots, X_N(k)\} \\ &= \text{Set}_{FH} \cup \text{Set}_{SH}, \\ &\text{Set}_{FH} = \{X_1(k), X_2(k), \dots, X_H(k)\}, \\ &\text{Set}_{SH} = \{X_{H+1}(k), \dots, X_N(k)\}, \end{aligned} \quad (16)$$

where

$$S_1(k) \le S_2(k) \le \dots \le S_H(k) \le S_{H+1}(k) \le \dots \le S_N(k), \quad (17)$$

$H = \lfloor N/2 \rfloor$ is the integer part of N/2, Set_{FH} is the ranked set of agents in the first half of the population, and Set_{SH} is the ranked set of agents in the second half of the population.

The vector $\mathbf{W}_i(k)$ of weights of slime mould consists of the elements (weights) $w_i^f(k), f = 1\dots q$

$$\mathbf{W}_i(k) = \left[w_i^1(k) \dots w_i^f(k) \dots w_i^q(k) \right]^T, i = 1\dots N, \quad (18)$$

which are computed in terms of

$$\begin{aligned} w_i^f(k) &= \begin{cases} 1 + r_i^f \log\left(g_i(k)\right), & \text{if } X_i(k) \in \text{Set}_{FH}, \\ 1 + r_i^f \log\left(g_i(k)\right), & \text{otherwise}, \end{cases} \\ g_i(k) &= \left(S_b(k) - S_i(k)\right) / \left(S_b(k) - S_w(k) + \varepsilon\right) + 1, \\ i &= 1\dots N, f = 1\dots q, \end{aligned} \quad (19)$$

where log is the notation for the decimal logarithm, r_i^f are random numbers, $0 \le r_i^f \le 1$, $S_b(k)$ is the best fitness obtained at the current iteration, $S_w(k)$ is the worst fitness obtained at the current iteration, with the values (using (17))

$$\begin{aligned} S_b(k) &= \min_{i=1\dots N} S_i(k) = S_1(k), \\ S_w(k) &= \max_{i=1\dots N} S_i(k) = S_N(k), \end{aligned} \quad (20)$$

and ε, $\varepsilon = \text{const} > 0$, is relatively small to avoid zero denominator in (19).

The parameters involved in the position update equation are computed in terms of

$$\begin{aligned} a &= \text{arctanh}\left(1 - k/k_{max}\right), \\ b &= 1 - k/k_{max}, \end{aligned} \quad (21)$$

which allow the computation of the uniformly distributed random numbers v_a^f and v_b^f

$$v_a^f \in [-a, a], v_b^f \in [-b, b], f = 1\dots q. \quad (22)$$

Using the notation $X_b(k)$ for the best agent, i.e., the solution obtained in all iterations so far, with the elements $x_b^f(k), f = 1\dots q$:

$$X_b(k) = \left[x_b^1(k) \dots x_b^f(k) \dots x_b^q(k) \right]^T, \quad (23)$$

and introducing two agents $X_A(k)$ and $X_B(k)$ randomly selected in the population:

$$\begin{aligned} X_A(k) &= \left[x_A^1(k) \dots x_A^f(k) \dots x_A^q(k) \right]^T, \\ X_B(k) &= \left[x_B^1(k) \dots x_B^f(k) \dots x_B^q(k) \right]^T, \end{aligned} \quad (24)$$

with randomly selected A and B, A,B = 1...N, the position update equation is

$$x_i^f(k+1) = \begin{cases} r^f\left(u^f - l^f\right) + l^f, & \text{if } r^f < z, \\ x_b^f(k) + v_a^f(w_i^f(k) \\ \quad \cdot x_A^f(k) - x_B^f(k)), & \text{if } r^f \ge z \text{ and } r_i^f < p_i, \\ v_b^f x_i^f(k), & \text{otherwise}, \end{cases} \quad (25)$$

$i = 1\dots N,$

where r^f are random numbers, $0 \le r^f \le 1$, z is a parameter with recommended values in [21] in the interval $0 \le z \le 0.1$, and the parameters p_i are computed in terms of [21]

$$p_i = \tanh |S_i(k) - S_{min}(k)|, i = 1\dots N, \quad (26)$$

and $S_{min}(k)$ is the best fitness obtained in all iterations so far:

$$S_{min}(k) = \min_{i=1\dots N, k_b=0\dots k} S_i(k_b). \quad (27)$$

SMA consists of steps SMA1 to SMA7. These steps are given as follows after the revision of the algorithm presented in [21]:

- *SMA1.* The initial random population, which consists of the position vectors $X_i(k) \in D_s$ of N agents, i.e., $i = 1\dots N$, is generated such that to fulfill (13)–(15). The maximum number of iterations is set to k_{max}. The iteration index is initialized to $k = 0$. The best fitness S_{best} is initialized to $S_{best} = \infty$ because the optimization problem defined in (11) is a minimization-type one.

- *SMA2.* The performance of each element of the population of agents (i.e., the slime mould) is evaluated; this evaluation involves simulations and/or experiments conducted on the fuzzy control systems if the optimal tuning of fuzzy controllers is carried out. The evaluation leads to the fitness function values of all agents.

- *SMA3.* The best fitness $S_{min}(k)$ obtained in all iterations is computed using (27). If $S_{min}(k) < S_{best}$, then the best fitness is S_{best} will be updated to $S_{min}(k)$ and the corresponding agent will represent the updated best agent (vector) $X_b(k)$.

- *SMA4.* The vector $\mathbf{W}_i(k)$ of weights of slime mould is calculated using (18)–(20).

- *SMA5.* Agents' positions are updated using (25), which is supported by (21)–(24), (26) and (27).

- *SMA6.* The iteration index k is incremented with one. The algorithm goes on with step 2 until k_{max} is reached.

- *SMA7.* The solution to the optimization problem is the last best agent obtained so far, i.e., the vector $X_b(k_{max})$, and the minimum value of the cost function is S_{best}.

SMA is used in the optimization problem defined in (11) in terms of the following relationships:

$$
\begin{aligned}
&X_i(k) = \rho, i = 1...N, \\
&S_i(k) = J, i = 1...N, \\
&D_s = D_\rho, \\
&X_b(k_{max}) = \rho^*, \\
&q = 3.
\end{aligned} \tag{28}
$$

Using the aspects presented in this section and the previous one, the SMA-based tuning approach dedicated to TSK PI-FCs consists of proceeding the steps 1–4. These steps are briefly described as follows:

Step 1. As pointed out in [46], the sampling period T_s is set in accordance with the requirements of quasi-continuous digital control.

Step 2. The feasible domain D_ρ is set in order to include all constraints imposed to the elements of ρ.

Step 3. SMA is integrated using steps SMA1 to SMA7 and (28) in solving the optimization problem defined in (11) leading to the optimal parameter vector ρ^* in (12) and three of the optimal parameters of TSK PI-FC, namely β^*, B_e^* and η^*.

Step 4. The optimal parameter $B_{\Delta e}$ of TSK PI-FC, with the notation $B_{\Delta e}^*$, results from (9) calculated for the optimal parameters in (12) substituted in (8). The setpoint filter with the transfer function defined in (6) is used in the fuzzy control system structure using β^* instead of β.

4. INSERTING INFORMATION FEEDBACK MODEL F1 IN METAHEURISTIC ALGORITHMS

Using the notation in (13) for the position vector $X_i(k)$ of ith agent at iteration k of the metaheuristic algorithm without information feedback model (i.e., the initial algorithm), the notation $S_i(k)$ for its fitness, which is also equal to the cost function value J in terms of (28), and inserting the notation $Z_i(k)$ for the position vector of ith agent at iteration k of the metaheuristic algorithm with information feedback model F1

$$
Z_i(k) = \left[z_i^1(k) ... z_i^f(k) ... z_i^q(k)\right]^T \in D_s, i = 1...N, \tag{29}
$$

where $z_i^f(k)$ is the position of ith agent in f^{th} dimension, $f = 1...q$, the information feedback model F1 is characterized by the recurrent equation [40–42]

$$
Z_i(k + 1) = \alpha_{F1} X_i(k) + \beta_{F1} Z_i(k), \tag{30}
$$

where the expressions of the weighting factors α_{F1} and β_{F1}, which satisfy $\alpha_{F1} + \beta_{F1} = 1$, are [40–42]

$$
\begin{aligned}
&\alpha_{F1} = S_i(k)/(S_i(k) + S_i(k + 1)), \\
&\beta_{F1} = S_i(k + 1)/(S_i(k) + S_i(k + 1)).
\end{aligned} \tag{31}
$$

Equations (30) and (31) are successfully integrated in PSO algorithms [40], in NSGA-III algorithms [41], and in multiobjective evolutionary algorithms based on decomposition [42]. The agent (or particle) velocity and position update equations employed in PSO algorithms are [2]

$$
\begin{aligned}
V_i(k + 1) &= w(k)V_i(k) + c_1 r_1(P_{g,best} - X_i(k)) \\
&\quad + c_2 r_2(P_{i,best} - X_i(k)), \\
X_i(k + 1) &= X_i(k) + V_i(k + 1), i = 1...N,
\end{aligned} \tag{32}
$$

where $V_i(k)$ is i^{th} agent particle (agent) velocity vector at iteration k,

$$
V_i(k) = \left[v_i^1(k) ... v_i^f(k) ... v_i^q(k)\right]^T, i = 1...N, \tag{33}
$$

$v_i^f(k)$ is the velocity of ith agent in f^{th} dimension, $f = 1...q$, r_1, r_2 are random variables with uniform distribution between 0 and 1, $c_1, c_2 > 0$ are weighting factors of the stochastic acceleration terms that pull each particle toward their end position, $w(k) > 0$ is the inertia weight, $P_{i,best}$ is the best position (vector) achieved by i^{th} agent (particle), and $P_{g,best}$ is the best position (vector) explored by the entire swarm so far. The PSO algorithm with information feedback model F1 will be referred to as PSO-F1 algorithm.

The position and velocity updates in GSA are governed by the recurrent equations [2]

$$
\begin{aligned}
v_i^f(k + 1) &= \rho_i v_i^f(k) + a_i^f(k), \\
x_i^f(k + 1) &= x_i^f(k) + v_i^f(k + 1), i = 1...N, f = 1...q,
\end{aligned} \tag{34}
$$

where ρ_i is a uniform random variable, $0 \leq \rho_i \leq 1$, and $a_i^f(k)$ is the acceleration of ith agent at iteration k. GSA with information feedback model F1 will be referred to as GSA-F1 algorithm.

The new position and velocity vectors of ith agent (or charged particle) in CSS algorithms are calculated in terms of [2]

$$
\begin{aligned}
X_i(k + 1) &= r_{i1} k_a(k)(F_i/m_i)(\Delta k)^2 \\
&\quad + r_{i2} k_v(k)V_i(k)\Delta k + X_i(k), \\
V_i(k + 1) &= (X_i(k + 1) - X_i(k))/\Delta k, i = 1...N,
\end{aligned} \tag{35}
$$

where $k_a(k)$ is the acceleration parameter, $k_v(k)$ is the velocity parameter, r_{i1}, r_{i2}, are uniformly distributed random numbers within [0, 1], m_i is the mass of ith agent, and Δk is the time step set to 1. The CSS algorithm with information feedback model F1 will be referred to as CSS-F1 algorithm.

The updated agent (i.e., grey wolf) position vector in GWO algorithms is obtained in terms of [2]

$$
\begin{aligned}
X_i(k + 1) &= (X^\alpha(k + 1) + X^\beta(k + 1) \\
&\quad + X^\delta(k + 1))/3, i = 1...N,
\end{aligned} \tag{36}
$$

where $X^\alpha(k)$, $X^\beta(k)$ and $X^\delta(k)$ are the top three agents positions vectors computed in terms of [2]

$$J(X^\alpha(k)) = \min_{i=1...N} \left\{ J(X_i(k)) | X_i(k) \in D_s \right\},$$

$$J(X^\beta(k)) = \min_{i=1...N} \left\{ J(X_i(k)) | \atop X_i(k) \in D_s \backslash \{X^\alpha(k)\}, \right. \quad (37)$$

$$J(X^\delta(k)) = \min_{i=1...N} \left\{ J(X_i(k)) | \atop X_i(k) \in D_s \backslash \{X^\alpha(k), X^\beta(k)\}. \right.$$

The GWO algorithm with information feedback model F1 will be referred to as GWO-F1 algorithm.

Using the notation $X_{\text{best}}(k)$ for the best agents' position (i.e., the position of the prey tracked by the whales, which are the agents) vector in the population at iteration k, the position update equation in WOA is [45]

$$X_i(k+1) = \begin{cases} X_{\text{best}}(k) - f_i(k), & \text{if C1,} \\ X_{\text{rand}}(t) - g_i(k), & \text{if C2,} \\ h_i(k)e^{b1}\cos(2\pi l) + X_{\text{best}}(k), & \text{otherwise,} \end{cases} \quad (38)$$

$$\text{C1}: \text{p} < 0.5 \text{ and } |a_i^f(k)| \le 1 \forall f = 1...q,$$
$$\text{C2}: \text{p} < 0.5 \text{ and } |a_i^f(k)| > 1 \forall f = 1...q, i = 1...N,$$

where p is random number within $0 \le \text{p} \le 1$, the parameter b = const defines the shape of the logarithmic spiral of whales' path, set to b = 1 for simplifying the shape, l is a uniformly distributed random number, $-1 \le \text{l} \le 1$, rand is an arbitrary agent index, and $f_i(k), g_i(k)$ are distance vectors. WOA with information feedback model F1 will be referred to as WOA-F1 algorithm.

5. VALIDATION AND COMPARISON

SMA with information feedback model F1 (SMA-F1) and the tuning approach presented in Section 3 are validated in this section by the optimal tuning of TSK PI-FCs for the angular position control of the experimental setup described in [56], which belongs to the Intelligent Control Systems Laboratory of the Politehnica University of Timisoara, Romania. A photo of the experimental setup is given in Figure 3. The parameter values of the servo system models given in (1) and (2) are $u_a = 0.15$, $u_b = 1$, $u_c = 0.15$, $k_p = k_{EP} = 140$ and $T_\Sigma = 0.92$ s. The application of the three steps of the tuning approach is described as follows.

The sampling period was set to $T_s = 0.01$ s in step 1 of the tuning approach. The vector variables ρ of the cost function belong to the feasible (and also search) domain D_ρ set in step 2 as

$$D_\rho = \{\beta | 3 \le \beta \le 17\} \times \{B_e | 20 \le B_e \le 38\} \times \{\eta | 0.25 \le \eta \le 0.75\}. \quad (39)$$

The dynamic regimes considered in the optimization problem are characterized by the $r = 40$ rad step type modification of the set-point and zero initial conditions. The upper limit of the sum in (11) was set to 2000 instead of ∞. Therefore, the time horizon used in the evaluation of the cost function by experiments conducted on the fuzzy control system is 20 s.
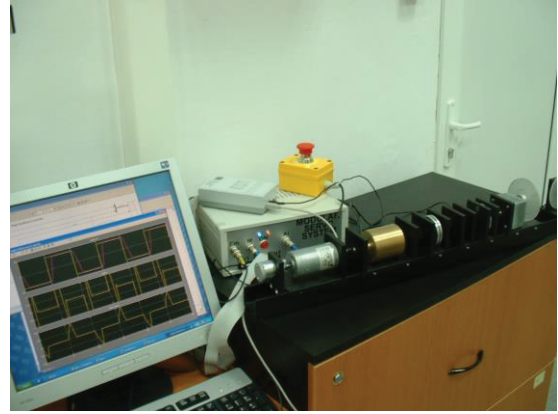

Figure 3 | Experimental setup.

The parameters of SMA were set in step 3 using the recommendations given in [21], namely z = 0.03 and $\varepsilon = 0.001$. In addition, authors' experience in other metaheuristic algorithms applied to optimal tuning of fuzzy controllers and fuzzy models as well is reflected in setting the parameters N = 20 and $k_{\max} = 20$ to achieve an acceptable trade-off to convergence and computational resources. Fast convergence is aimed in this paper *versus* our recent papers [2,43–46], where $k_{\max} = 100$ was imposed.

The simulations and experiments were run in Matlab R2007b environment on a personal computer with Windows 10 operating system and the following hardware configuration: CPU Intel Core i5-7500, 3.4 GHz quad-core, 16 GB of DDR3 RAM memory at 1600 MHz, 256 SSD. These details are convenient for other researchers to redo the experiments, thus making the work in this paper relatively easily acceptable.

The optimal controller parameters and the corresponding minimum cost function $J_{\min}$ in relation with (11) are presented in Table 2. Table 2 includes suggestively the comparison with other metaheuristic algorithms, namely PSO, GSA, CSS, GWO, WOA, PSO with information feedback model F1 (PSO-F1), GSA with information feedback model F1 (GSA-F1), CSS with information feedback model F1 (CSS-F1), GWO with information feedback model F1 (GWO-F1) and WOA with information feedback model F1 (WOA-F1). The same numbers of agents N = 20 and maximum number of iterations $k_{\max} = 20$ as in case of SMA and SMA-F1 were used for all algorithms in the comparison. In addition, also to ensure a fair comparison of the algorithms, in order to reduce the effects of random parameters in the algorithms (always present in metaheuristic optimization algorithms), the results are presented as averaged values after the first $N_t = 10$ runs (trials) of the algorithms; the comparisons included in the past papers were based on the best five runs of algorithms. The values of the other nonrandom parameters of the algorithms used in the comparison are presented in [2,44–46]. The readers are invited to contact the authors if they are interested in the implementations of all these algorithms.

The performance of SMA and other algorithms considered in Table 2 was assessed using two quality indices. The first index reflects the convergence speed c_s by measuring the number of evaluations required for the cost function $J(\rho)$ until obtaining the optimal solution ρ^*. The second index reflects the accuracy rate a_r defined as the percent value of standard deviation of $J(\rho)$ obtained

by running a certain metaheuristic algorithm divided to the average value of the solution, $Avg\left(J_{\min}\right)$:

$$a_r = \text{StDev}^{\%}\left(J_{\min}\right) = 100\frac{\text{StDev}(J_{\min})}{\text{Avg}(J_{\min})},$$

$$\text{Avg}\left(J_{\min}\right) = \frac{1}{N_t}\sum_{h=1}^{N_t} J_{\min}^{(h)}, \qquad (40)$$

$$\text{StDev}\left(J_{\min}\right) = \sqrt{\frac{1}{N_t-1}\sum_{h=1}^{N_t}\left(J_{\min}^{(h)} - \text{Avg}\left(J_{\min}\right)\right)^2},$$

where $J_{\min}$ is the value of $J(\rho)$ measured after running the algorithm, and the superscript h indicates the observed trial, $h = 1...N_t$. Since Table 2 gives contains values, the values in the last column correspond actually to $\text{Avg}(J_{\min})$ instead of $J_{\min}$.

The search performance of all metaheuristic algorithms analyzed in this section is synthesized in Table 3 in terms of the values of the quality indices c_s and a_r.

The results presented in Table 2 indicate that the best performance of the control system as far as the average value of the cost function is concerned is obtained by SMA and SMA-F1 followed by WOA, and the worse performance is obtained by CSS and WOA-F1. The comparison of accuracy and resource usage on the basis of the results given in Table 3 reveals that the best accuracy rate is obtained by SMA and SMA-F1 followed by WOA, and the worse is obtained by GSA and GWO-F1. The best convergence speed is obtained by SMA-F1 followed by SMA-F1 and GWO, and the worse by PSO-F1 and PSO. Concluding, the best overall performance is exhibited by SMA and SMA-F1, which is followed by WOA, justifying the application of SMA in this paper.

The results presented in Table 2 show that adding the information feedback model F1 is beneficial for SMA, GSA and CSS as far as the reduction of cost function values is concerned. The results presented in Table 3 highlight that adding the information feedback model F1 offers the reduction of the convergence speed of SMA, PSO and WOA. But the main advantage of adding the information feedback model, as outlined in Table 3, is the reduction of the accuracy rate of all algorithms except GSO and WOA.

These conclusions are drawn for the process, fuzzy controller, optimization problem and dynamic regime considered in this paper. Other processes, controllers, optimization problems and dynamic regimes are expected to lead to different conclusions; challenging processes in this regard are those considered in [57–64].

Two typical fuzzy control system responses are presented in Figure 4 in terms of controlled output and control signal *versus* time considering the average controller after the first iteration of SMA (with the parameters $B_e = 33.2784$, $\eta = 0.5268$ and $B_{\Delta e} = 0.0676$), and the average controller after the application of the 20 iterations of SMA (with the parameters given in the first row in Table 2). Figure 4 clearly shows the improvement of empirical quality indices after the application of SMA. Both plots in Figure 4 also highlight the effects of the process nonlinearity in the initial and final part of fuzzy control system responses.

It is certainly obvious from Figure 4 that the output trajectories blend to reference input value as time approaches the end of the time horizon. As a consequence, with the suggested controller, the control system addressed can be stabilized. However, as pointed out in [44], the stability cannot be proved.

6. CONCLUSIONS

This paper proposed an approach to the SMA-based tuning of cost-effective fuzzy controllers for servo systems. This approach is advantageous because of two reasons, (1) and (2), which represent the pros of the approach proposed in this paper in the framework of metaheuristic optimization algorithms: (1) it is computational efficient as proved by the values of two quality indices related to accuracy and resource usage, (2) SMA has a small number of parameters. The validation and comparison on real-world servo system position control shows the superior performance of SMA *versus* other similar metaheuristic algorithms reflected by smaller value of the cost function, defined, as specified in Section 1, as the sum of time multiplied by squared control error, which is the discrete-time version of ITSE used in continuous-time control.

The limitations of SMA, which are also the cons of the approach proposed in this paper, are also twofold, (i) and (ii): its current

Table 2 | Average values of controller parameters and minimum cost function J.

Algorithm	B_e^*	η^*	$B_{\Delta e}^*$	β^*	k_C^*	T_i^* (s)	J_{min}
SMA	38	0.75	0.00857633	4.821519	0.0035384	4.435799	2797210
PSO	36.5966	0.5935440	0.0819451	4.913505	0.00352454	4.520424	3523353
GSA	33.0287	0.7414017	0.0786853	4.770794	0.00360137	4.398443	3693850
CSS	23.1039	0.6662344	0.0715622	3.776415	0.00407048	3.4743	4935620
GWO	37.9457	0.6994276	0.0857001	4.910214	0.00352702	4.517395	3315304
WOA	35.0330	0.6513980	0.0748931	5.09787	0.0034414	4.682604	3297086
SMA-F1	38	0.75	0.0857638	4.821494	0.00353585	4.435774	2797210
PSO-F1	27.5769	0.6315154	0.0601471	5.161258	0.00341579	4.748359	3905199
GSA-F1	32.6942	0.7219313	0.0685116	5.194281	0.00340791	4.778737	3444169
CSS-F1	20.8791	0.7095039	0.0464305	4.962937	0.00351122	4.565903	4213287
GWO-F1	34.9617	0.7367405	0.0770037	4.030243	0.00350845	4.550747	3355471
WOA-F1	34.2792	0.6915126	0.1020510	3.799553	0.00407587	3.495585	5236392

CSS, Charged System Search; CSS-F1, Charged System Search with information feedback model F1; GSA, Gravitational Search Algorithm; GSA-F1, Gravitational Search Algorithm with information feedback model F1; GWO, Grey Wolf Optimizer; GWO-F1, Grey Wolf Optimizer with information feedback model F1; PSO, Particle Swarm Optimization; PSO-F1, Particle Swarm Optimization with information feedback model F1; SMA, Slime Mould Algorithm; SMA-F1, Slime Mould Algorithm with information feedback model F1; WOA, Whale Optimization Algorithm; WOA-F1, Whale Optimization Algorithm with information feedback model F1.

Table 3 | Values of quality indices c_s and a_r.

Algorithm	c_s	a_r
SMA	19.7	0
PSO	311.4	18.67
GSA	167.2	18.86
CSS	97.9	12.07
GWO	20	18.52
WOA	18.7	6.24
SMA-F1	14.3	0
PSO-F1	269.9	11.77
GSA-F1	204.4	12.82
CSS-F1	162.9	11.33
GWO-F1	180.4	20.49
WOA-F1	11.9	14.9

CSS, Charged System Search; CSS-F1, Charged System Search with information feedback model F1; GSA, Gravitational Search Algorithm; GSA-F1, Gravitational Search Algorithm with information feedback model F1; GWO, Grey Wolf Optimizer; GWO-F1, Grey Wolf Optimizer with information feedback model F1; PSO, Particle Swarm Optimization; PSO-F1, Particle Swarm Optimization with information feedback model F1; SMA, Slime Mould Algorithm; SMA-F1, Slime Mould Algorithm with information feedback model F1; WOA, Whale Optimization Algorithm; WOA-F1, Whale Optimization Algorithm with information feedback model F1.
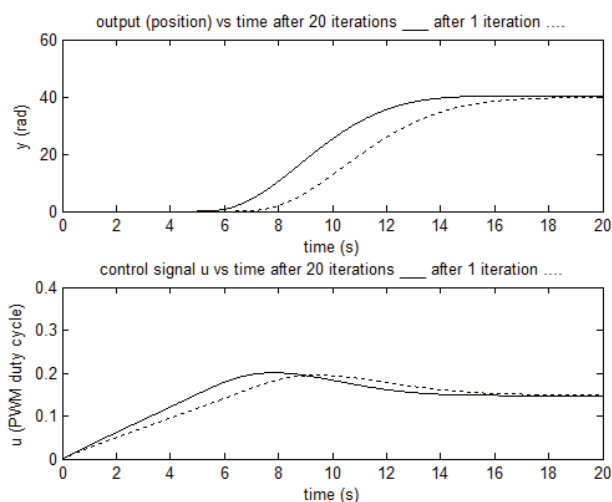

Figure 4 | Real-time experimental results expressed as fuzzy control system responses y and u with Takagi–Sugeno–Kang proportional-integral fuzzy controllers (TSK PI-FC) after one iteration of Slime Mould Algorithm (SMA) (dotted line) and after 20 iterations of SMA (continuous line).

implementation is offline, (ii) it requires a relatively large number of evaluations of the cost function (or the fitness function). But these two cons are also met at other metaheuristic algorithms.

Future research will be focused on the mitigation of these limitations by modifying SMA in order to work online in adaptive fuzzy control system structure, and solving other optimization problems including those specific to optimal tuning of controller parameters in optimization problems expressed as reference tracking control problems.

Moreover, except the algorithms used in this paper, some of the most representative computational intelligence algorithms can be used to solve the problems, like Monarch Butterfly Optimization

(MBO) [65], Earthworm Optimization Algorithm (EWA) [66], Elephant Herding Optimization (EHO) [67] and Moth Search (MS) algorithm [68]. Another group of representative algorithms is discussed in [69].

Nevertheless, a viable way to apply the results given in this paper is represented by networked control systems. With the increasing progress in advanced data communication technology, sampled-based event-triggered control protocols have been introduced in such systems and especially in complex networked control systems to effectively alleviate the communication burden by reducing computation loads and data transmission rates. Therefore this motivates the attempt to further enhance the performance of such control systems in terms of using fuzzy control, and the optimal tuning of tuning parameters is a convenient way to cope with such systems.

CONFLICT OF INTEREST

The authors declare that they do not have any conflict of interest.

AUTHOR'S CONTRIBUTION

The contribution of the authors to this paper is equal.

ACKNOWLEDGMENTS

REFERENCES

[1] R.-E. Precup, P. Angelov, B.S.J. Costa, M. Sayed-Mouchaweh, An overview on fault diagnosis and nature-inspired optimal control of industrial process applications, Comput. Ind. 74 (2015), 75–94.

[2] R.-E. Precup, R.-C. David, Nature-Inspired Optimization Algorithms for Fuzzy Controlled Servo Systems, Butterworth-Heinemann, Elsevier, Oxford, UK, 2019.

[3] J.J. Eckert, F.M. Santiciolli, R.Y. Yamashita, F.C. Corrêa, L.C.A. Silva, F.G. Dedini, Fuzzy gear shifting control optimisation to improve vehicle performance, fuel consumption and engine emissions, IET Control Theory Appl. 13 (2019), 2658–2669.

[4] J.J. Eckert, L. Corrêa de Alkmin Silva, F.G. Dedini, F.C. Corrêa, Electric vehicle powertrain and fuzzy control multi-objective optimization, considering dual hybrid energy storage systems, IEEE Trans. Veh. Technol. 69 (2020), 3773–3782.

[5] I. Abadlia, L. Hassaine, A. Beddar, F. Abdoune, M.R. Bengourina, Adaptive fuzzy control with an optimization by using genetic algorithms for grid connected a hybrid photovoltaic-hydrogen generation system, Int. J. Hydr. Energy. 45 (2020), 22589–22599.

[6] A.P. Masoumi, A.R. Tavakolpour-Saleh, A. Rahideh, Applying a genetic-fuzzy control scheme to an active free piston Stirling engine: design and experiment, Appl. Energy. 268 (2020), 115045.

[7] Z.-M. Fu, L.-L. Zhu, F.-Z. Tao, P.-J. Si, L.-F. Sun, Optimization based energy management strategy for fuel cell/battery/ultraca-pacitor hybrid vehicle considering fuel economy and fuel cell lifes-pan, Int. J. Hydr. Energy. 45 (2020), 8875–8886.

[8] H.-B. Liang, J.-L. Zou, K. Zuo, M.J. Khan, An improved genetic algorithm optimization fuzzy controller applied to the wellhead back pressure control system, Mech. Syst. Signal Process. 142 (2020), 106708.

[9] Y.-P. Hu, Y.-P. Yang, S. Li, Y.-M. Zhou, Fuzzy controller design of micro-unmanned helicopter relying on improved genetic opti-mization algorithm, Aerosp. Sci. Technol. 98 (2020), 105685.

[10] T.-Z. Wu, W.-S. Yu, L.-X. Guo, A study on use of hybrid energy storage system along with variable filter time constant to smooth DC power fluctuation in microgrid, IEEE Access. 7 (2019), 175377–175385.

[11] R.-J. Wai, A.S. Prasetia, Adaptive neural network control and opti-mal path planning of UAV surveillance system with energy con-sumption prediction, IEEE Access. 7 (2019), 126137–126153.

[12] S.-W. Xie, Y.-F. Xie, F.-B. Li, Z.-H. Jiang, W.-H. Gui, Hybrid fuzzy control for the goethite process in zinc production plant com-bining type-1 and type-2 fuzzy logics, Neurocomput. 366 (2019), 170–177.

[13] M.E. Girgis, R.I. Badr, Optimal fractional-order adaptive fuzzy control on inverted pendulum model, Int. J. Dynam. Control. 9 (2021), 288–298.

[14] A. Mohammadzadeh, E. Kayacan, A novel fractional-order type-2 fuzzy control method for online frequency regulation in AC microgrid, Eng. Appl. Artif. Intell. 90 (2020), 103483.

[15] N.A. Reisi, S. Hadipour Lakmesari, M.J. Mahmoodabadi, S. Hadipour, Optimum fuzzy control of human immunodefi-ciency virus type1 using an imperialist competitive algorithm, Informat. Med. Unlocked. 16 (2019), 100241.

[16] S. Tripathi, A. Shrivastava, K.C. Jana, Self-tuning fuzzy controller for sun-tracker system using Gray Wolf Optimization (GWO) technique, ISA Trans. 101 (2020), 50–59.

[17] M.I. Solihin, C.Y. Chuan, W. Astuti, Optimization of fuzzy logic controller parameters using modern meta-heuristic algorithm for Gantry Crane System (GCS), Mat. Today Proc. 29 (2020), 168–172.

[18] M. Azizi, R.G. Ejlali, S.A.M. Ghasemi, S. Talatahari, Upgraded whale optimization algorithm for fuzzy logic based vibration control of nonlinear steel structure, Eng. Struct. 192 (2019), 53–70.

[19] S.M. Nosratabadi, M. Bornapour, M.A. Gharaei, Grasshopper optimization algorithm for optimal load frequency control con-sidering predictive functional modified PID controller in restruc-tured multi-resource multi-area power system with redox flow battery units, Control Eng. Pract. 89 (2019), 204–227.

[20] M. Dong, X. Luan, B. Wu, J. Liang, The fuzzy control of electro-hydraulic servo system based on DE algorithm, in: Y. Jia, J. Du, W. Zhang (Eds.), Proceedings of 2018 Chinese Intelligent Sys-tems Conference, Lecture Notes in Electrical Engineering, vol. 529, Springer, Singapore, 2019, pp. 747–757.

[21] S.-M. Li, H.-L. Chen, M.-J. Wang, A.A. Heidari, S. Mirjalili, Slime mould algorithm: a new method for stochastic optimization, Fut. Gen. Comput. Syst. 111 (2020), 300–323.

[22] H. Garg, A hybrid PSO-GA algorithm for constrained optimiza-tion problems, Appl. Math. Comput. 274 (2016), 292–305.

[23] H. Garg, A hybrid GSA-GA algorithm for constrained optimiza-tion problems, Inf. Sci. 478 (2019), 499–523.

[24] E. Osaba, J. Del Ser, A. Sadollah, M.N. Bilbao, D. Camacho, A dis-crete water cycle algorithm for solving the symmetric and asym-metric traveling salesman problem, Appl. Soft Comput. 71 (2018), 277–290.

[25] E. Osaba, X.S. Yang, I. Fister Jr, J. Del Ser, P. Lopez-Garcia, A.J. Vazquez-Pardavila, A discrete and improved bat algorithm for solving a medical goods distribution problem with phar-macological waste collection, Swarm Evol. Comput. 44 (2019), 273–286.

[26] L. Rodríguez, O. Castillo, J. Soria, P. Melin, F. Valdez, C.I. González, G.E. Martinez, J. Soto, A fuzzy hierarchical opera-tor in the grey wolf optimizer algorithm, Appl. Soft Comput. 57 (2017), 315–328.

[27] P. Melin, D. Sánchez, Multi-objective optimization for modular granular neural networks applied to pattern recognition, Inf. Sci. 460–461 (2018), 594–610.

[28] G. Beruvides, R. Quiza, R.E. Haber, Multi-objective optimiza-tion based on an improved cross-entropy method. A case study of a micro-scale manufacturing process, Inf. Sci. 334–335 (2016), 161–173.

[29] J. Pimentel, C.J. Ardila Hernandez, E. Niño, D.J. Molinares, J. Ruiz-Rangel, Water cycle algorithm: implementation and anal-ysis of solutions to the bi-objective travelling salesman problem, Int. J. Artif. Intell. 17 (2019), 105–136.

[30] M. Moattari, M.H. Moradi, Conflict monitoring optimization heuristic inspired by brain fear and conflict systems, Int. J. Artif. Intell. 18 (2020), 45–62.

[31] H.P.H. Anh, T.T. Huan, Optimal walking gait generator for biped robot using modified Jaya optimization technique, Int. J. Comput. Intell. Syst. 13 (2020), 382–399.

[32] H.P.H. Anh, Fuzzy load forecast with optimized parametric adjustment using Jaya optimization algorithm, Int. J. Comput. Intell. Syst. 13 (2020), 875–892.

[33] H.-Y. Li, W. Gan, A decomposition-based multiobjective chem-ical reaction optimization algorithm for community detection in complex networks, Int. J. Comput. Intell. Syst. 13 (2020), 524–537.

[34] J.-H. Yi, S. Deb, J.-Y. Dong, A.H. Alavi, G.-G. Wang, An improved NSGA-III algorithm with adaptive mutation operator for Big Data optimization problems, Fut. Gen. Comput. Syst. 88 (2020), 571–585.

[35] J.-H. Yi, L.-N. Xing, G.-G. Wang, J.-Y. Dong, A.V. Vasilakos, A.H. Alavi, L. Wang, Behavior of crossover operators in NSGA-III for large-scale optimization problems, Inf. Sci. 509 (2020), 470–487.

[36] D. Gao, G.-G. Wang, W. Pedrycz, Solving fuzzy job-shop schedul-ing problem using DE algorithm improved by a selection mecha-nism, IEEE Trans. Fuzzy Syst. 28 (2020), 3265–3275.

[37] L.-M. Li, K.-D. Lu, G.-Q. Zeng, L. Wu, M.-R. Chen, A novel real-coded population-based extremal optimization algorithm with polynomial mutation: a non-parametric statistical study on continuous optimization problems, Neurocomput. 174 (2016), 577–587.

[38] G.-Q. Zeng, X.-Q. Xie, M.-R. Chen, J. Weng, Adaptive population extremal optimization-based PID neural network for multivari-able nonlinear control systems, Swarm Evol. Comput. 44 (2019), 320–334.

[39] M.-R. Chen, G.-Q. Zeng, K.-D. Lu, A many-objective population extremal optimization algorithm with an adaptive hybrid muta-tion operation, Inf. Sci. 498 (2019), 62–90.

[40] G.-G. Wang, Y. Tan, Improving metaheuristic algorithms with information feedback models, IEEE Trans. Cybern. 49 (2019), 542–555.

[41] Z.-M. Gu, G.-G. Wang, Improving NSGA-III algorithms with information feedback models for large-scale many-objective optimization, Fut. Gen. Comput. Syst. 107 (2020), 49–69.

[42] Y. Zhang, G.-G. Wang, K.-Q. Li, W.-C. Yeh, M.-W. Jian, J.-Y. Dong, Enhancing MOEA/D with information feedback models for large-scale many-objective optimization, Inf. Sci. 522 (2020), 1–16.

[43] R.-E. Precup, R.-C. David, E.M. Petriu, A.-I. Szedlak-Stinean, C.-A. Bojan-Dragos, Grey wolf optimizer-based approach to the tuning of PI-fuzzy controllers with a reduced process parametric sensitivity, IFAC-PapersOnLine. 49 (2016), 55–60.

[44] R.-E. Precup, R.-C. David, E.M. Petriu, Grey wolf optimizer algorithm-based tuning of fuzzy control systems with reduced parametric sensitivity, IEEE Trans. Ind. Electron. 64 (2017), 527–534.

[45] R.-C. David, R.-E. Precup, S. Preitl, A.-I. Szedlak-Stinean, R.-C. Roman, E.M. Petriu, Design of low-cost fuzzy controllers with reduced parametric sensitivity based on whale optimization algorithm, in Proceeding of 2020 IEEE World Congress on Computational Intelligence, Glasgow, UK, 2020, pp. 1–6.

[46] R.-C. Roman, R.-E. Precup, E.M. Petriu, F. Dragan, Combination of data-driven active disturbance rejection and Takagi-Sugeno fuzzy control with experimental validation on tower crane systems, Energies. 12 (2019), 1548.

[47] S. Preitl, R.-E. Precup, On the algorithmic design of a class of control systems based on providing the symmetry of open-loop Bode plots, Sci. Bull. UPT Trans. Autom. Control Comput. Sci. 41 (1996), 47–55.

[48] S. Preitl, R.-E. Precup, An extension of tuning relations after symmetrical optimum method for PI and PID controllers, Automatica. 35 (1999), 1731–1736.

[49] S. Galichet, L. Foulloy, Fuzzy controllers: synthesis and equivalences, IEEE Trans. Fuzzy Syst. 3 (1995), 140–148.

[50] I. Škrjanc, S. Blažič, Predictive functional control based on fuzzy model: design and stability study, J. Intell. Robot. Syst. 43 (2005), 283–299.

[51] R.-E. Precup, M.L. Tomescu, Stable fuzzy logic control of a general class of chaotic systems, Neural Comput. Appl. 26 (2015), 541–550.

[52] J. Vaščák, J. Hvizdoš, Vehicle navigation by fuzzy cognitive maps using sonar and RFID technologies, in Proceeding of IEEE 14th International Symposium on Applied Machine Intelligence and Informatics, Herľany, Slovakia, 2016, pp. 75–80.

[53] I. Dzitac, F.G. Filip, M.-J. Manolescu, Fuzzy logic is not fuzzy: world-renowned computer scientist Lotfi A. Zadeh, Int. J. Comput. Communic. Control. 12 (2017), 748–789.

[54] Z.C. Johanyák, A modified particle swarm optimization algorithm for the optimization of a fuzzy classification subsystem in a series hybrid electric vehicle, Tech. Vjes. Tech. Gaz. 24 (2017), 295–301.

[55] O. Galluppi, S. Formentin, C. Novara, S.M. Savaresi, Multivariable D2-IBC and application to vehicle stability control, ASME J. Dyn. Syst. Meas. Control. 141 (2019), 1–12.

[56] Modular Servo System, User's Manual, Inteco Ltd., Krakow, Poland, 2007.

[57] P. Baranyi, P. Korondi, R.J. Patton, H. Hashimoto, Trade-off between approximation accuracy and complexity for TS fuzzy models, Asian J. Control. 6 (2004), 21–33.

[58] R.-E. Precup, S. Preitl, Development of fuzzy controllers with non-homogeneous dynamics for integral-type plants, Electr. Eng. 85 (2003), 155–168.

[59] P Angelov, Outside the box: an alternative data analytics framework, J. Autom. Mob. Robot. Intell. Syst. 8 (2014), 29–35.

[60] R. Andoga, L. Főző, J. Judičák, R. Bréda, S. Szabo, R. Rozenberg, M. Džunda, Intelligent situational control of small turbojet engines, Int. J. Aerosp. Eng. 2018 (2018), 8328792.

[61] M. Evagoras, K.M. Deliparaschos, E. Kalyvianaki, A.C. Zolotas, T. Charalambous, Robust dynamic CPU resource provisioning in virtualized servers, IEEE Trans. Services Comput. (2020).

[62] B. Xiao, H.-K. Lam, Y. Yu, Y.-D. Li, Sampled-data output-feedback tracking control for interval type-2 polynomial fuzzy systems, IEEE Trans. Fuzzy Syst. 28 (2020), 424–433.

[63] R.-C. Chi, Y. Hui, B. Huang, Z.-S. Hou, Adjacent-agent dynamic linearization-based iterative learning formation control, IEEE Trans. Cybern. 50 (2020), 4358–4369.

[64] R.-E. Precup, R.-C. Roman, T.-A. Teban, A. Albu, E.M. Petriu, C. Pozna, Model-free control of finger dynamics in prosthetic hand myoelectric-based control systems, Stud. Informat. Control. 29 (2020), 399–410.

[65] G.-G. Wang, S. Deb, Z.-H. Cui, Monarch butterfly optimization, Neural Comput. Appl. 31 (2019), 1995–2014.

[66] G.-G. Wang, S. Deb, L. Dos Santos Coelho, Earthworm optimisation algorithm: a bio-inspired metaheuristic algorithm for global optimisation problems, Int. J. Bio-Insp. Comput. 12 (2018), 1–22.

[67] G.-G. Wang, S. Deb, L. Dos Santos Coelho, Elephant herding optimization, in Proceeding of 3rd International Symposium on Computational and Business Intelligence (ISCBI), Bali, Indonesia, 2015, pp. 1–6.

[68] G.-G. Wang, Moth search algorithm: a bio-inspired metaheuristic algorithm for global optimization problems, Memetic Comput. 10 (2018), 151–164.

[69] S.-I. Bejinariu, R. Luca, H. Costin, Nature-inspired algorithms based multispectral image fusion, in Proceeding of 9th International Conference and Exposition on Electrical and Power Engineering, Iasi, Romania, 2016, pp. 10–15.