ISSN 1870-4069

Multi-Objective Assistant for Control Designing with Overshoot Suppression, Robustness, Low Energy Consumption, and Maximum Voltage Level Using Genetic Algorithms

Martín Montes-Rivera, Jesús Rafael Tavarez-Delgado,
Martín Eduardo Rodríguez-Franco

Universidad Politécnica de Aguascalientes,
Mexico

martin.montes@upa.edu.mx,
{mc190001, mc190002}@alumnos.upa.edu.mx

Abstract. The main objective of control theory is to obtain a desired behavior in the controlled variables. There are several control techniques that have been proposed with this goal, but the simpler controllers like the PID are more often used in industry, due to their hand-tuning methods, even when they have several limitations. Alternatively, Artificial Intelligence has become a support for control theory in the designing of controllers without requiring a hard design stage, or several stages for designing a control law. However, their structure and its implementation are so complex that the PID remains being used in around 95% of industrial applications. In this work is proposed a control design assistant based on the automatic generation of controllers with transfer function description that can be programmed in computers or other embedded devices, this assistant uses a proposed multi-objective function that prioritizes candidate solutions with overshoot suppression, robustness, low energy consumption and maximum voltage level. We optimize the proposed multi-objective function with a Genetic Algorithm and test its results using a first order, a second order, and a motor position control system.

Keywords: Intelligent control, generated controller, genetic algorithms.

1 Introduction

The main objective in control theory is to produce a desired behavior in the controlled variables, or system stabilization [1].

Since 1960s modern control theory has developed several techniques in all its branches, including adaptive control, robust control, optimal control, variable structure control, among others [2].

The kind of controller to implement depends on the controlled process, but several techniques are suitable for any specific application, as shown in [3, 4]. The variability of those techniques makes that one must think twice before using a more complex alternative, because it could produce similar results with extra efforts.

Proportional-Integral-Derivative (PID) controllers are among the most popular industry controllers, near to 95% of the industrial control applications use a PID structure [2]. The PID popularity is related to its possibility of compensating several practical processes and the simplicity of the methods designed for tuning its three parameters [2, 5]. However, PID controllers have serious limitations that do not make them the best choice in systems with uncertainties and disturbances, moreover, in several cases, the wrong choosing of the PID gains can result in dangerous behaviors and high energy consumption related to windup effect, derivative kick effect, excessive overshoot and continuous oscillation, among others [2].

On the other hand, new control methods are published each year, and they show better results than those obtained with PID controllers, but they require high knowledge of mathematical modeling, linear algebra, system parametrization and several steps depending on the control technique. In addition, they only guarantee good results under specific situations, and require compensations in the presence of wear or disturbances [1, 2].

Alternatively, control theory has found support on Artificial Intelligence (AI) for designing several methodologies that achieve control without requiring a complex mathematical model, or several stages for designing a control law, like those described in [1, 5–7].

AI also has been applied in control theory using numerical optimization algorithms for tuning the controllers with specific criteria depending on the application. Genetic Algorithms (GAs) and Particle Swarm Optimization (PSO) are common numerical optimizations algorithms used for tuning PID controllers, as shown in [8–10].

On the other hand, adaptive control with model uncertainties based on AI has become a recent focus of attention using reinforced learning and Artificial Neural Networks (ANNs), as shown in [11–14].

Some of the most common criteria in control design are suppression of overshoot, energy efficiency, and disturbance resistance, which are required in several applications like in position controlling of a Direct Current (DC) motor, which is commonly required in robotics. Some of this specifications in control designing are shown in [15–23]. Alternatively, this goals have been used separately with intelligent controllers using different AI techniques, as shown in [7, 8, 11–14, 24–26].

Nowadays, Industry 4.0 is introducing new technological alternatives for improving manufacturing with economic impact and societal progress. The industry 4.0 includes the use of more intelligent robots, cooperative machines, self-decision systems, autonomous solver problems, learning machines, 3D printing, augmented reality, big data analysis, smart city implementations, Internet of Things (IoT), intelligent self-tuning controllers [27].

In this paper we propose a control design assistant where the designer can decide which features prefer for the controller in a specific application by using the multi-objective function gains to prioritize his preferences. Then the controller will be self-generated based on a transfer function. The controller generation is performed off-line, allowing to use the control-designing assistant in special devices with IoT, where input-output response is recorded and send it to a server, where the controller is generated and returned to the device for being applied.

The transfer function schema is selected because this approach is the most popular used in control theory and it has several definitions and techniques that allow to verify behavior and stability of systems, either if it has single or multiple inputs and outputs.

Moreover, a continuous controller based on the transfer function description can be transformed to a discrete transfer function that can be programmed in a computer, an embedded system, a Field-Programmable Gate Array (FPGA) or a microcontroller.

The multi-objective function for this work was selected based on popular goals in different control systems, and it includes overshoot suppression, robustness, low energy consumption, and maximum voltage level.

2 Theoretical Framework: Genetic Algorithm

The Genetic Algorithms (GAs) belong to the family of evolutive algorithms inspired by the selection principles proposed by Darwin. GAs have been used for numerical optimization in several areas since were proposed in the 1950s. Their main operators are population initializing, fitness value calculation, fitness-based selection, crossover, and mutation [28].

The population generation is according to GA's search space (X) using binary string elements $(\mathrm{X} = \mathrm{B}^*)$ or genotypes $b^* \in \mathrm{B}^*/$ required in the crossover and the mutation operations; while phenotype $x \in \mathrm{X}/$ or numerical representation allows getting fitness value calculation. The population (P) depends on the size of the population (S_P), the maximum reachable value (max_V), the minimum reachable value (min_V), and the number of bits for resolution (n_b) [28].

Fitness evaluation maps a numerical value with the objective function $f(\mathrm{x})$ that measures how well adapted is a chromosome in the population [28].

After fitness evaluation, there are different methods for mating pool (M_P) selection, but in this research, we use tournament selection. Because it still being considered a good alternative against noisy data, and it controls the selection pressure with its size (S_T) [28].

Crossover mixes the parents genes with multiple random crossover points for the offspring (O) generation, we choose this method because increases crossover combinations, produces major diversity and evades evading early convergence [42].

After that, we apply mutation operation with the probability (P_M). The mutation is type uniform.

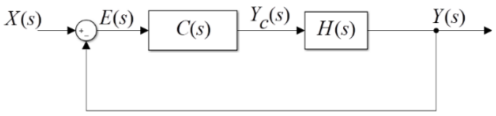

Fig. 1. Block diagram of a SISO feedback controller.

Finally, the population is sorted based on its fitness, and worst adapted elements are deleted to maintain S_P, for ecological stability [28]. Stop condition of GA in this work is the number of generations (N_G).

3 Methodology

The control design assistant proposed optimizes a controller for achieving the desired response in a transfer function description, i.e. the user must determine it either using control theory techniques or the input-output measurements and a computer algorithm. After obtaining it, the dynamic equations of the system described below, evaluate controllers in terms of error, overshoot, energy consumption, a maximum level of voltage, and robustness.

The control diagram in **Fig. 1** shows the control output $Y_c(s)$ and the process variable output $Y(s)$. After analyzing it, the equations (1, 2, 3) are obtained:

$$E(s) = X(s) - Y(s), \tag{1}$$

$$Y(s) = E(s) \cdot C(s) \cdot H(s), \tag{2}$$

$$Y_c(s) = E(s) \cdot C(s). \tag{3}$$

Substituting $Y(s)$ in (1) with (2) and solving for $E(s)$ gives equation (4):

$$E(s) = \frac{X(s)}{1 + C(s) \cdot H(s)}, \tag{4}$$

Taking the $E(s)$ in (3) and substituting with (4), gives $Y_c(s)$ in equation (5), which determines the output of the controller to any input:

$$Y_c(s) = X(s) \frac{C(s)}{1 + C(s) \cdot H(s)}, \tag{5}$$

The $E(s)$ in (2) is substituted with (4), giving the $Y(s)$ in (6), which allows to determine the system output to any input:

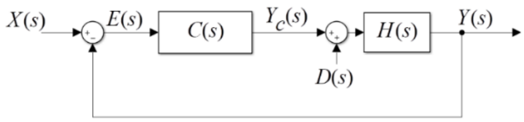

Fig. 2. Block diagram of a SISO feedback controller with disturbances.

$$Y(s) = X(s)\frac{C(s)\cdot H(s)}{1+C(s)\cdot H(s)}\ , \tag{6}$$

A disturbance allows affecting the cost value of controllers that are not robust, as shown in Fig. 2. After analyzing it the equations from (7-10) are obtained:

$$Y(s) = \left[Y_c(s) + D(s)\right]\cdot H(s), \tag{7}$$

$$Y_c(s) = E(s)\cdot C(s), \tag{8}$$

$$E(s) = X(s) - Y(s), \tag{9}$$

$$Y(s) = \left[E(s)\cdot C(s) + D(s)\right]\cdot H(s). \tag{10}$$

Substituting (9) in (10) and simplifying equations gives (11), that represents the output of the system for any input with disturbances:

$$Y(s) = \frac{X(s)\cdot C(s)\cdot H(s)}{1+C(s)\cdot H(s)} + \frac{D(s)\cdot H(s)}{1+C(s)\cdot H(s)}\ , \tag{11}$$

The equation (12) is obtained by substituting (8) with (9) and (11), which represents the controller output for any input with disturbances:

$$Y_c(s) = \frac{X(s)\cdot C(s)}{1+C(s)\cdot H(s)} - \frac{D(s)\cdot C(s)\cdot H(s)}{1+C(s)\cdot H(s)}\ , \tag{12}$$

Having all previous equations, the dynamic behavior of the controlled systems in the presence of specific inputs and disturbances is obtained, and the objective function is given by the dot product of G and $M(\overline{X})$, as shown in equation (13):

$$f(\overline{X}) = G \cdot M(\overline{X}). \tag{13}$$

Where G is an array containing the gains to prioritize the desired characteristics in the controller, $\overline{X}$ are the numerical coefficients generated by the GA, and $M(\overline{X})$ is the multi-objective function array depending on $\overline{X}$.

$M\left(\overrightarrow{X}\right)=\left[h_1(\overrightarrow{X}) \quad h_2(\overrightarrow{X}) \quad h_3(\overrightarrow{X}) \quad h_4(\overrightarrow{X})\right]$ is a four-dimensional array, because it contains five functions used for testing energy consumption $h_1(\overrightarrow{X})$, error suppression with limitation for overshoot, negative responses and steady state error $h_2(\overrightarrow{X})$, maximum voltage level $h_3(\overrightarrow{X})$, and robustness $h_4(\overrightarrow{X})$. The dynamical characteristics in these functions are tested with n samples for testing the outputs $Y(s)$, the inputs $X(s)$ and the controller outputs $Y_c(s)$, with and without disturbances $D(s)$, as described in equations (14-18):

$$h_1(\overrightarrow{X}) = \frac{1}{n}\sum\left|Y_c(s)\right| , \tag{14}$$

$$h_2(\overrightarrow{X}) = \frac{1}{n}\sum\left|Y(s) - X(s)\right| + P . \tag{15}$$

where $P=\alpha\left(p_1+p_2+p_3\right)$ is a limitation for the overshoot, negative response and steady state error, respectively, for the step input, as shown in equation (16):

$$\begin{aligned} p_1 &= \left[\max\left(\left|Y_1(s)\right|\right) > X_1(s)\right]\cdot\max\left(\left|Y_1(s)\right|\right) \\ p_2 &= \left[\min\left(Y_1(s)\right) < 0\right]\cdot\left|\min\left(Y_1(s)\right)\right| \\ p_3 &= \alpha\left|Y_n(s) - X_n(s)\right| \end{aligned} . \tag{16}$$

Having $X_{s_i}(s)$ and $Y_{s_i}(s)$ as the inputs and outputs in the step response, we have also $X_{s_n}(s)$ and $Y_{s_n}(s)$ as the final inputs and outputs of the step response after n samples:

$$h_3(\overrightarrow{X}) = \max\left(\left|Y_c(s)\right|\right)\cdot\left[1+\alpha\cdot\left(\max\left(\left|Y_c(s)\right|\right) > V\right)\right], \tag{17}$$

where $\alpha\cdot\left(\max\left(\left|Y_c(s)\right|\right) > V\right)$ is a limitation for responses with values of control greater that the maximum allowed voltage $\left(V\right)$:

$$h_4(\overrightarrow{X}) = \frac{1}{n}\sum\left|X_1(s) - Y_D(s)\right| . \tag{18}$$

$Y_D(s)$ as $Y(s)$ is obtained in the presence of $D(s)$ disturbances, as shown in equation (11).

Table 1. Training parameters used for GA.

Algorithm	Input Parameters
GA	S_T: 300, S_P : 300, N_G :3000, P_M :0.15, $n_b = 14, resol = 0.001$

Table 2. Description and transfer function of the testing systems.

1st Order	2nd Order
$\frac{Y(s)}{X(s)} = \frac{1}{s+1}$	$\frac{Y(s)}{X(s)} = \frac{1}{s^2+s+1}$
Motor Position	
$\frac{Y(s)}{X(s)} = \frac{2.3693244195\text{e-}5}{1.162653\text{e-}9s^3 + 1.359159\text{e-}6s^2 + 0.00039687s}$	

Table 3. GA generated controllers with the proposed multi-objective function.

1st order	$\frac{8.796^{12}s^3+2.831^{16}s^2+1.428^{17}s+7.193^{16}}{7.768^{16}s^3+1.161^{17}s^2+1.399^{17}s+8.796^{12}}$	2nd order	$\frac{7.921^{13}s^3+9.171^{16}s^2+1.441^{17}s+7.210^{16}}{1.491^{16}s^3+5.621^{16}s^2+1.261^{17}s+8.810^{12}}$
Motor position control	$\frac{8.810^{12}s^3+1.131^{17}s^2+7.151^{16}s+7.661^{16}}{8.971^{16}s^3+1.261^{17}s^2+1.351^{17}s+5.411^{16}}$		

4 Results

4.1 Design of Experiment

We use Matlab™ for training process and simulation with the GA programmed according to section 2. The computer used was a Windows 10 desktop with processor Intel(R) Core™ i7-6700 CPU 3.40GHz, 16.0 GB RAM.

The control-designing assistant in this research it was tested in a first order, a second-order, and a position control DC motor system.

The GA optimized eight coefficients, four for the numerator and four for the denominator of the transfer function, allowing to obtain up a third-order transfer function. The number of iterations was set depending on the processing time, the GA used 3000 iterations with a total processing time of 12.5 minutes.

The optimizing algorithm has several numerical parameters, they were determined using cross-validation 80% training and 20% for testing rapid convergence and good results. The determined parameters are shown in Table 1.

The algorithm ran three times, one for each transfer function in the testing systems described in Table 2.

Four different inputs were used for measuring the quality of each controller, these inputs include step response, ramp response, variable response, and step response with disturbances. The maximum voltage accepted was 5 volts, similarly as if the motor operates with TTL conditions.

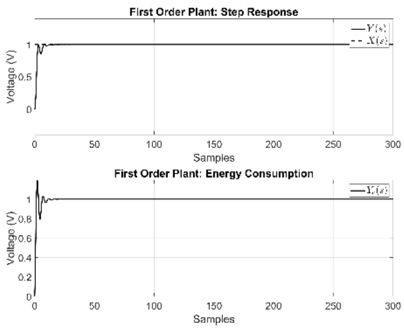

Fig. 3. GA controller with step response in a first-order system.

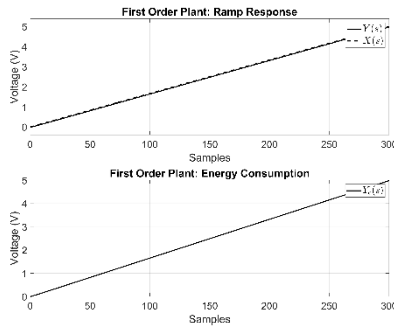

Fig. 4. GA controller with ramp response in a first-order system.

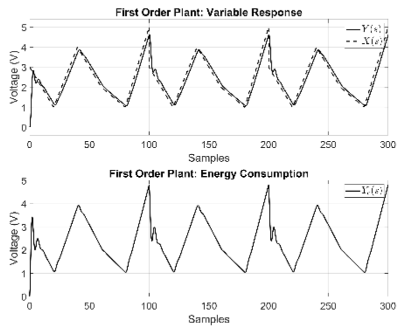

Fig. 5. GA controller with variable input in a first-order system.

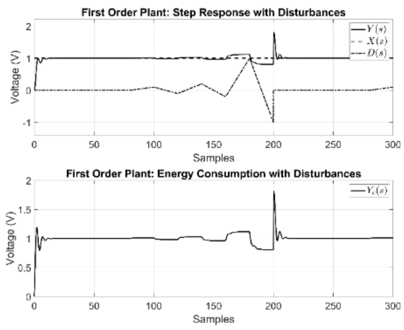

Fig. 6. GA controller with step response and disturbances in a first- order system.

Fig. 7. GA controller with step response and second-order system.

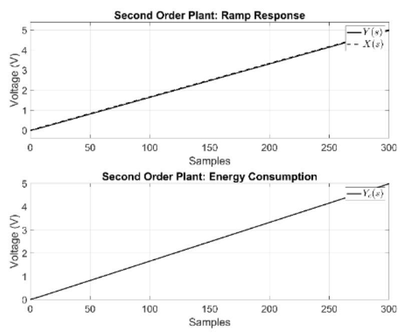

Fig. 8. GA controller with ramp response in a second-order system.

4.2 Results of the Generated Controllers

This section shows the generated controllers (Table 3) and the control response in each test. The fist-order and second order responses show: 0 steady-state error, reference following, no overshoot, they maintain controller output below the 5V limit, robustness and show low energy consumption decreasing the controllers' outputs when the reference is reached. Like it is show from Fig. 3 to Fig. 10.

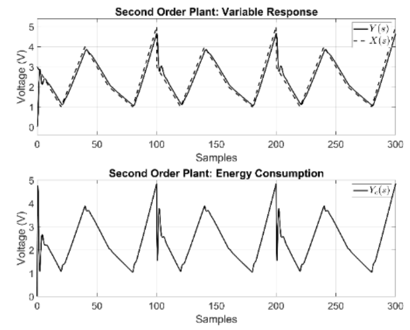

Fig. 9. GA controller with variable input in a second-order system.

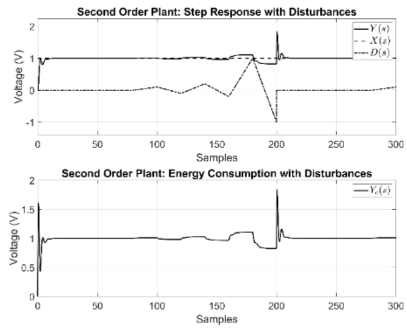

Fig. 10. GA controller with step response and disturbances in a second-order system.

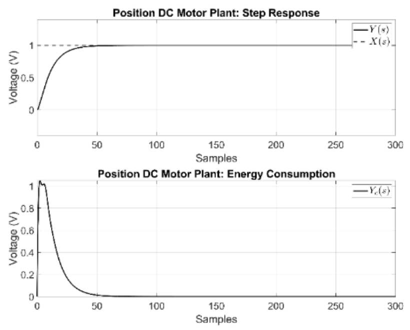

Fig. 11. GA controller with step response in a motor positioning system.

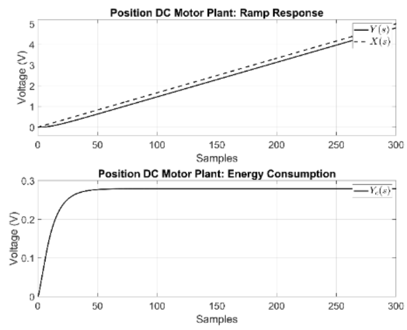

Fig. 12. GA controller with ramp response in a motor positioning system.

The motor response also gets a steady-state error equal to 0, achieving the secondary goals: without overshoot, voltage under the desired maximum, lower energy consumption and robustness. Like shown from **¡Error! No se encuentra el origen de la referencia.** to Fig. 14.

The reached cost values for each solution obtained according to equation (13) are 0.4922 for the first-order system, 0.4612 for the second order system, and 0.6653 for the motor position controller.

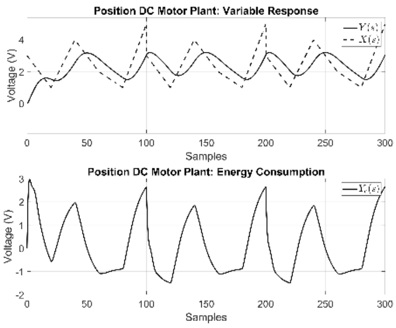

Fig. 13. GA controller with variable input in a motor positioning system.

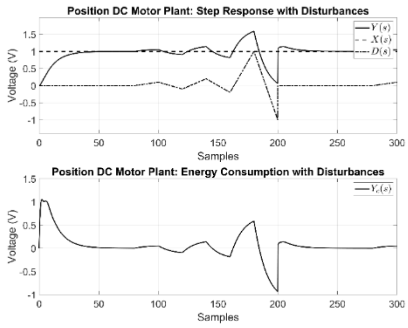

Fig. 14. GA controller with step response and disturbances in a motor positioning system.

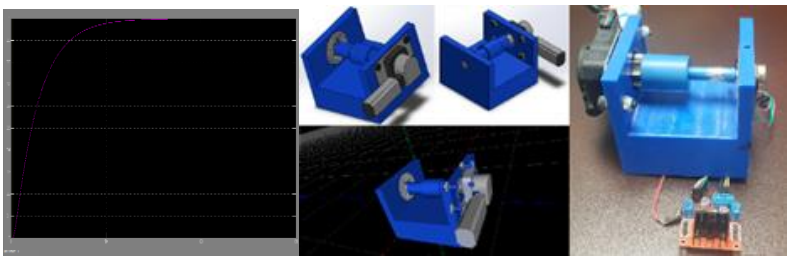

Fig. 15. 3D simulation of motor with VRML and Simulink 3D animation.

DC motor is simulated using Simulink 3D animation, based on the real motor that will be used for this application like is shown in Fig. 15.

5 Conclusions

In this work presented a control design assistant based on a transfer function that is automatically generated for satisfying a proposed multi-objective cost function. This cost function identifies the quality of the candidate solutions according to desired characteristics, which include reducing the steady-state error, use of energy, overshoot, maintenance of control voltage under the maximum accepted value, and robustness.

The multi-objective function was tested using four different control processes: a position control for a DC motor, a first order, and a second-order systems. The inputs applied to test them were step, ramp, variable and variable with disturbances.

We optimized the multi-objective function with Genetic Algorithms obtaining a cost value of 8.9253 and an average processing time of 0.25 seconds per iteration or 12.5 minutes. The GA controllers show steady-state error equal to 0, they maintain a 5V desired maximum voltage, they remain without overshoot, with low energy consumption and robustness.

The control design assistant proposed can generate controllers for different processes satisfying the specific control characteristics, like it is shown this work.

5.1 Future Work

In this work, the control design assistant has been tested under simulation, and the controllers were applied in the computer used for optimizing the control structure.

The next stage of research will include the device development for acquiring the input-output signals and sending them to server for generating the controller with a GA, and then send back the controller to the device for being applied.

References

1. Rivera, M.M., Ramos, M.P., Mora, J.L.O.: Automatic Generator of Decoupling Blocks Using Genetic Programming. In: New Trends in Networking, Computing, E-learning, Systems Sciences, and Engineering, pp. 281–290, Springer (2015) https://doi.org/10.1007/978-3-319-06764-3.
2. Hou, Z.S., Wang, Z.: From model-based control to data-driven control: Survey, classification and perspective. Inf. Sci. (Ny)., 235, pp. 3–35 (2013) https://doi.org/10.1016/j.ins.2012.07.014.
3. Kazmierkowski, M.P., Malesani, L.: Current control techniques for three-phase voltage-source pwm converters: A survey. IEEE Trans. Ind. Electron, 45, pp. 691–703 (1998) https://doi.org/10.1109/41.720325.
4. Wirasingha, S.G., Emadi, A.: Classification and review of control strategies for plug-in hybrid electric vehicles (2011) https://doi.org/10.1109/TVT.2010.2090178.
5. Hernandez-Barragan, J., Rios, J.D., Alanis, A.Y., Lopez-Franco, C., Gomez-Avila, J., Arana-Daniel, N.: Adaptive Single Neuron Anti-Windup PID Controller Based on the Extended Kalman Filter Algorithm. Electronics, 9, 636 (2020) https://doi.org/10.3390/electronics9040636.
6. Ferdaus, M.M., Pratama, M., Anavatti, S.G., Garratt, M.A., Lughofer, E.: PAC: A novel self-adaptive neuro-fuzzy controller for micro aerial vehicles. Inf. Sci. (Ny)., 512, pp. 481–505 (2020) https://doi.org/10.1016/j.ins.2019.10.001.
7. Hua, C., Wu, S., Guan, X.: Stabilization of t-s fuzzy system with time delay under sampled-data control using a new looped-functional. IEEE Trans. Fuzzy Syst., 28, pp. 400–407 (2020) https://doi.org/10.1109/TFUZZ.2019.2906040.
8. Silva, P.R., Abreu, I.S., Forte, P.A., Do Amaral, H.M.C.: Genetic Algorithms for Satellite Launcher Attitude Controller Design. Intel. Artif., 22, pp. 150–161 (2019) https://doi.org/10.4114/intartif.vol22iss63pp150-161.
9. Nekoui, M.A., Pakzad, M., Pakzad, S.: Optimal fractional order PID controllers design based on genetic algorithm for time delay systems. In: 19th International Symposium on Power Electronics, pp. 1–6. Institute of Electrical and Electronics Engineers Inc. (2017) https://doi.org/10.1109/PEE.2017.8171685.
10. Xiang, Z., Shao, X., Wu, H., Ji, D., Yu, F., Li, Y.: An adaptive integral separated proportional–integral controller based strategy for particle swarm optimization. Knowledge-Based Syst., 195, 105696 (2020) https://doi.org/10.1016/j.knosys.2020.105696.
11. Yang, Y., Guo, Z., Xiong, H., Ding, D.W., Yin, Y., Wunsch, D.C.: Data-Driven Robust Control of Discrete-Time Uncertain Linear Systems via Off-Policy Reinforcement

Learning. IEEE Trans. Neural Networks Learn. Syst., 30, pp. 3735–3747 (2019) https://doi.org/10.1109/TNNLS.2019.2897814.

12. Zhang, M., Gan, M.G.: Data-Driven Adaptive Optimal Control for Linear Systems with Structured Time-Varying Uncertainty. IEEE Access, 7, pp. 9215–9224 (2019) https://doi.org/10.1109/ACCESS.2019.2891575.
13. Sami, A., Memon, A.Y.: Robust optimal control of continuous time linear system using reinforcement learning. In: ANZCC 2018-2018 Australian and New Zealand Control Conference, pp. 154–159, Institute of Electrical and Electronics Engineers Inc. (2019) https://doi.org/10.1109/ANZCC.2018.8606607.
14. Wang, Y., Velswamy, K., Huang, B.: A Novel Approach to Feedback Control with Deep Reinforcement Learning. IFAC-PapersOnLine, 51, pp. 31–36 (2018) https://doi.org/10.1016/j.ifacol.2018.09.241.
15. Keadtipod, P., Banjerdpongchai, D.: Supervisory Robust Model Predictive Control for Linear Systems Subject to Parametric Uncertainty and Application to Level Control Process. In: 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), pp. 680–683 (2018) https://doi.org/10.1109/ECTICon.2018.8619987.
16. Duan, L., Yu, X., Jin, S., Hou, Z., Bu, X.: Feedforward and Feedback Model Free Adaptive Iterative Learning Control with Application to a Linear Motor System. In: Chinese Control Conference (CCC), pp. 2564–2569 (2019) https://doi.org/10.23919/ChiCC.2019.8866628.
17. Zhang, K., Zhou, B.: Parametric Lyapunov Equation Based Event-Triggered and Self-Triggered Control of Discrete-Time Linear Systems. In: IEEE Conference on Control Technology and Applications (CCTA), pp. 866–869 (2019) https://doi.org/10.1109/CCTA.2019.8920623.
18. Demirel, B., Ghadimi, E., Quevedo, D.E., Johansson, M.: Optimal Control of Linear Systems With Limited Control Actions: Threshold-Based Event-Triggered Control. IEEE Trans. Control Netw. Syst., 5, pp. 1275–1286 (2018) https://doi.org/10.1109/TCNS.2017.2701003.
19. Borgers, D.P., Dolk, V.S., Heemels, W.P.M.H.: Riccati-Based Design of Event-Triggered Controllers for Linear Systems With Delays. IEEE Trans. Automat. Contr., 63, pp. 174–188 (2018) https://doi.org/10.1109/TAC.2017.2713047.
20. Otsuka, N., Kakehi, D.: Stabilization of Continuous-Time Switched Linear Systems Under Arbitrary Switching Via Reduced-Order Switched Observers. In: UKACC 12th International Conference on Control (CONTROL), pp. 223 (2018) https://doi.org/10.1109/CONTROL.2018.8516855.
21. Zhang, S., Wang, Q., Liang, J., Dong, C.: Model reference adaptive sliding mode control for switched linear systems. In: Chinese Automation Congress (CAC), pp. 4034–4038 (2017) https://doi.org/10.1109/CAC.2017.8243486.
22. Yuan, S., De Schutter, B., Baldi, S.: Adaptive Tracking Control of Switched Linear Systems Using Mode-Dependent Average Dwell Time. In: Eighth International Conference on Information Science and Technology (ICIST), pp. 1–7 (2018) https://doi.org/10.1109/ICIST.2018.8426176.
23. M.A., A., N.S., D., Rajanna, K.: Application of EMPC for precise position control of DC-motor system with Backlash. Control Eng. Pract., 100, 104422 (2020) https://doi.org/10.1016/j.conengprac.2020.104422.
24. Kluska, J., Zabinski, T.: PID-Like adaptive fuzzy controller design based on absolute stability criterion. IEEE Trans. Fuzzy Syst. (2020) https://doi.org/10.1109/TFUZZ.2019.2908772.

25. Lv, X., Fei, J., Sun, Y.: Fuzzy PID Controller Design for Uncertain Networked Control Systems Based on T–S Fuzzy Model with Random Delays. Int. J. Fuzzy Syst., 21, pp. 571–582 (2019) https://doi.org/10.1007/s40815-018-0583-3.
26. Annisa Jamali; Intant Z. Mat Darus: Intelligen Evolutionary Controller for Flexible Robotic Arm. In: Journal of Physics: Conference Series, pp. 1–14, IOP Publishing (2020) https://doi.org/10.1088/1742-6596/1500/1/012020.
27. Oztemel, E., Gursev, S.: Literature review of Industry 4.0 and related technologies (2020) https://doi.org/10.1007/s10845-018-1433-8.
28. Weise, T.: Global optimization algorithms-theory and application. Self-published. 2 (2009)
29. Zhang, J., Kang, M., Li, X., Liu, G.Y.: Bio-inspired genetic algorithms with formalized crossover operators for robotic applications. Front. Neurorobot (2017) https://doi.org/10.3389/fnbot.2017.00056.