Bursa Uludağ University Journal of The Faculty of Engineering, Vol. 27, No. 2, 2022 RESEARCH

DOI: 10.17482/uumfd.1090766

PID TUNING WITH UP-TO-DATE METAHEURISTIC ALGORITHMS

Fahri VATANSEVER[*]
Emre HACIİSKENDEROĞLU

Received: 21.03.2022; revised: 19.04.2022; accepted: 23.05.2022

Abstract: Control of systems is very important in applications. For this purpose appropriate controllers need to be designed. PIDs are the most popular controllers and there are traditional methods for their design. In recent years, metaheuristic algorithms also have been used to tuning the PID coefficients. In this study, an interactive graphical user interface program was designed, which makes the design of PID type controllers with six up-to-date metaheuristic algorithms according to different performance criteria. The controller coefficients can be tuned easily, quickly and effectively with this software tool that performs single or comparative designs, provides numerical and graphical solutions, and enables detailed analysis and synthesis.

Keywords: Control systems, PID, metaheuristic algorithms, computer aided engineering.

Güncel Metasezgisel Algoritmalarla PID Ayarlama

Öz: Uygulamalarda sistemlerin kontrolü son derece önemlidir. Bu amaçla uygun denetleyicilerin tasarlanması gerekmektedir. En popüler denetleyicilerin başında PID'ler gelmektedir ve bunların tasarımı için geleneksel yöntemler mevcuttur. Son yıllarda PID katsayılarının ayarlanması için metasezgisel algoritmalardan da faydalanılmaktadır. Gerçekleştirilen çalışmada PID türü denetleyicilerin tasarımını, farklı performans kriterlerine göre altı güncel metasezgisel algoritma ile yapan etkileşimli grafiksel kullanıcı arayüz programı tasarlanmıştır. Tekli veya karşılaştırmalı tasarımlar gerçekleştiren, sayısal ve grafiksel çözümler sunan, ayrıntılı analiz ve sentezlere olanak sağlayan bu yazılım aracıyla denetleyici katsayılarının ayarlanması kolay, hızlı ve etkin şekilde yapılabilmektedir.

Anahtar Kelimeler: Kontrol sistemleri, PID, metasezgisel algoritmalar, bilgisayar destekli mühendislik.

1. INTRODUCTION

One of the fundamental structures of the systems is the controllers. These structures ensure that the related systems operate in the desired properties. For this purpose, controllers with very different structures and properties have been developed, and the analysis and design of these circuits is very important. Traditional PID (proportional-integral-derivative) is the popular controller that frequently used in the industrial area. These controllers, which aim to reduce the error by performing multiplication, integral and derivative functions, and thus keeping the system response at the desired value, are mostly used as P, PI, PD and PID combinations.

The coefficients of PID controllers can be obtained both analytically and experimentally. The most known and used methods are Ziegler-Nichols (Ziegler and Nichols, 1942), Cohen-

[*]Bursa Uludağ University, Faculty of Engineering, Electrical-Electronics Eng. Dept. 16059 Bursa/Türkiye
Corresponding author: Fahri Vatansever (fahriv@uludag.edu.tr)

Coon, Chien Hrones Reswick, Wang-Juang-Chan (Golnaraghi and Kuo, 2009; Nise, 2015; Xue et al., 2007). By using time or frequency responses, appropriate controller coefficients can be determined according to different criteria in these methods.

In parallel with the developments in technology, computer aided designs also have gained wide application areas. There are many studies in the literature on the use of evolutionary, heuristic and metaheuristic algorithms in controller design (Abushawish et al., 2020; Oladipo et al., 2020; Rodríguez-Molina et al., 2020).

In this study, a software tool was designed to tuning PID coefficients with up-to-date metaheuristic algorithms. This tool calculates (tunes) the P, PI, PD or PID controller coefficients determined for user-defined feedback system with the desired up-to-date metaheuristic algorithm (Aquila optimizer (Abualigah et al., 2021b), Archimedes optimization (Hashim et al., 2021), Arithmetic optimization (Abualigah et al., 2021a), Bald eagle search optimization (Alsattar et al., 2020), Seagull optimization (Dhiman and Kumar, 2019), Sparrow search algorithm (Xue and Shen, 2020)) according to the selected performance criteria (IAE, ISE, ITAE, ITSE). With this software tool, which generates many numerical and graphical results for single and comparative designs, controller tunings can be made easily, quickly and effectively.

This paper is organized as follows: In Section 2, PID controllers and used metaheuristic algorithms are summarized. In Section 3, designed software tool with applications are given. Finally, Section 4 contains conclusions.

2. MATERIALS AND METHODS

2.1. PID Controllers

The basic block diagram of a feedback system is given in Fig. 1. PIDs are one of the most used classical/traditional controllers in applications. The general structure and equations of the PID controller are given in Table 1, and its effects on the closed-loop system response are given in Table 2. In addition, the fundamental performance criteria used in the design of control systems are summarized in Table 3 (CTMS, 2022; Matlab, 2021; Vatansever and Sen, 2013).

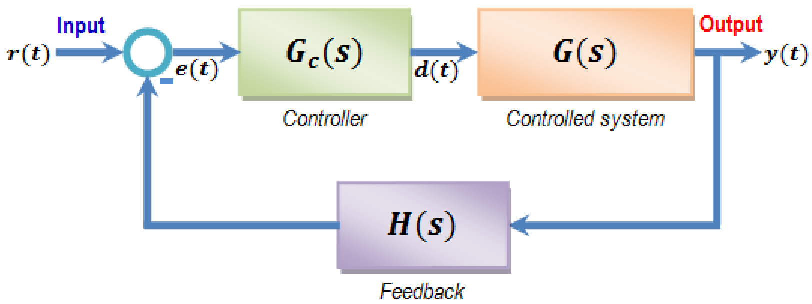


Figure 1:
Block diagram of the classical feedback system

2.2. Used Metaheuristic Algorithms

The up-to-date metaheuristic algorithms used in the study are summarized in Table 4 (Abualigah et al., 2021a; Abualigah et al., 2021b; Alsattar et al., 2020; Dhiman and Kumar, 2019; Hashim et al., 2021; Xue and Shen, 2020).

Table 1. The fundamental equations of PID controller

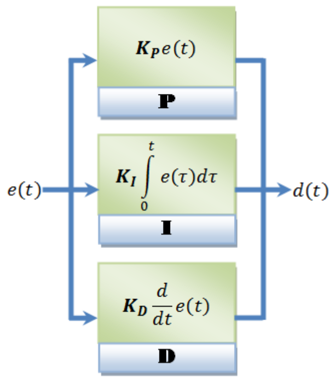
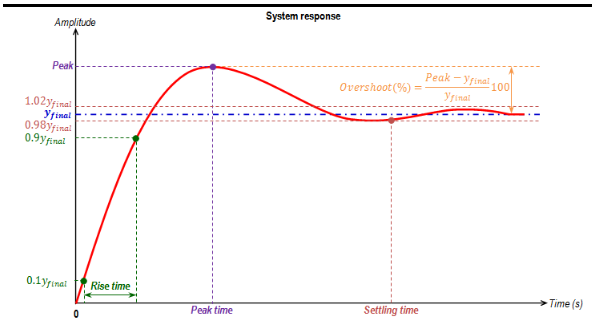
	Continuous-time	$d(t) = K_P e(t) + K_I \int_0^t e(\tau)d\tau + K_D \frac{d}{dt}e(t)$ $d(t) = K_p \left(e(t) + \frac{1}{T_i}\int_0^t e(\tau)d\tau + T_d \frac{d}{dt}e(t) \right)$ K_P: Proportional coefficient , K_I: Integral coef. , K_D: Derivative coef. T_i: Integral time , T_d: Derivative time
	Discrete-time	$d[n] = K_c \left(e[n] + \frac{T_s}{T_i}\sum_{i=0}^{n} e[i] + \frac{T_d}{T_s}(e[n] - e[n-1]) \right)$ T_s: Sampling time
	Transfer function	$G_c(s) = K_P + \frac{K_I}{s} + K_D s = \frac{K_D s^2 + K_P s + K_I}{s}$ $G_c(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s\right)$

Table 2. The general effect of PID parameters on a closed-loop system response



	K_P	K_I	K_D
Overshoot	Increase	Increase	Decrease
Rise time	Decrease	Decrease	Small effect
Settling time	Small effect	Increase	Decrease
Steady-state error	Decrease	Decrease	No effect

Table 3. Performance criteria (indices/indexes)

Performance criteria	Equation	MATLAB code

Integral of absolute error	$IAE = \int_0^\infty \lvert e(t) \rvert dt$	`time=0:a:b; [y t]=step(system,time);`
		`for i=1:length(t)`
		`    error(i)=1-y(i);`
		`end`
		`IAE=sum(abs(error));`

Integral of square error	$ISE = \int_0^\infty e^2(t)dt$	`time=0:a:b; [y t]=step(system,time);`
		`for i=1:length(t)`
		`    error(i)=1-y(i);`
		`end`
		`error=error*error'; ISE=sum(error);`

Integral of time-absolute error	$ITAE = \int_0^\infty t\lvert e(t) \rvert dt$	`time=0:a:b; [y t]=step(system,time);`
		`for i=1:length(t)`
		`    error(i)=abs(1-y(i))*t(i);`
		`end`
		`ITAE=sum(error);`

Integral of time-square error	$ITSE = \int_0^\infty te^2(t)dt$	`time=0:a:b; [y t]=step(system,time);`
		`for i=1:length(t)`
		`    error(i)=((1-y(i))^2)*t(i);`
		`end`
		`ITSE=sum(error);`

Table 4. The used metaheuristic algorithms

	Year	Inspiration	Basic knowledge	Pseudo-code
Aquila optimizer	2021	The behaviours of a bird namely Aquila.	The Aquila optimization algorithm steps (the process of catching the prey) are represented in four methods: ❖ The first method (expanded exploration): Selecting the search space by high soar with the vertical stoop. ❖ The second method (narrowed exploration): Contour flight with short glide attack. ❖ The third method (expanded exploitation): Low flight with slow descent attack. ❖ The fourth method (narrowed exploitation): Swooping by walk and grab prey.	▪ Initialization ▪ While (stopping criterion) ○ Calculate the fitness values ○ Determine the best value ➢ For each candidate solution • Update the mean value of the current solution • Update the parameters • Update position (solution) with appropriate search strategies (expanded and narrowed exploration, expanded and narrowed exploitation) ➢ end for end while ▪ Return the best solution
Archimedes optimization	2021	The physics law called Archimedes' principle.	The Archimedes optimization algorithm mimics the principle of upward buoyancy applied to an object fully or partially immersed in fluid. This state is proportional to weight of the displaced fluid. In this algorithm the population's individuals are the objects of different weights, volumes, accelerations and positions immersed in fluid. In algorithm's steps, the density and volume of every object are updated. In addition, if the object collides with the neighbouring object, its acceleration is also updated. The new position of the object is determined by the updated density, volume and acceleration.	▪ Initialization ▪ Calculate the fitness values and determine best value ▪ While (stopping criterion) ○ For each object ➢ Update parameters (density, volume, transfer and density decreasing factors) ➢ Update acceleration and normalize acceleration, update position / *Exploration phase* / ➢ Update acceleration and normalize acceleration, update direction flag, update position / *Exploitation phase* / ○ end for ○ Calculate the fitness values for each object and determine the best value end while ▪ Return the best solution
Arithmetic optimization	2021	The using main arithmetic operators (addition, subtraction, multiplication and division) for solving mathematical problems.	The Arithmetic optimization algorithm steps are represented in three phases like many other methods: ❖ Initialization phase: Creating the set of candidate solutions. ❖ Exploration phase: Randomly exploring the search space in several regions and approaching finding a better solution based on Division and Multiplication. ❖ Exploitation phase: Exploring the search area deeply on several dense regions and approaching finding a better solution based on Subtraction and Addition.	▪ Initialization ▪ While (stopping criterion) ○ Calculate the fitness values and find the best solution ○ Update Math Optimizer Accelerated and Math Optimizer Probability ○ For each solution ➢ For each position ✓ Generate random values ✓ Apply relevant operators (/ or *) according to condition and update solution with related rules / *Exploration phase* / ✓ Apply relevant operators (- or +) according to condition and update solution with related rules / *Exploitation phase* / ➢ end for ○ end for end while ▪ Return the best solution
Bald eagle search optimization	2020	The hunting strategy (intelligent social behaviour) of bald eagles as they search for fish.	In the bald eagle search optimization algorithm, hunting takes place in three stages: ❖ The first stage (selecting space): Selecting the space with the most prey (in terms of amount of food). ❖ The second stage (searching in space): Prey search movements (different directions within a spiral space for acceleration) within the selected space. ❖ The third stage (swooping): Swinging from the best position identified to their target prey.	▪ Initialization ▪ Calculate the fitness values of initial point ▪ While (stopping criterion) ○ For each point / *Select stage* / ➢ Identify and select the best area for food ○ end for ○ For each point / *Search stage* / ➢ Evaluate the new position for hunting ○ end for ○ For each point / *Swooping stage* / ➢ Use the new position to swoop towards the prey ○ end for end while ▪ Return the best solution

Seagull optimization	2019	The migration and attacking behaviours of a seagull.	The seagull optimization algorithm steps are represented in two stages: ❖ The migration (exploration) stage: The group of seagulls moves satisfying three conditions (avoiding the collisions, movement towards best neighbour's direction, remain close to the best search agent). ❖ The attacking (exploitation) stage: Attack the prey with spiral movements in the air.	▪ Initialization ▪ While (stopping criterion) ○ Calculate the fitness values of each search agent ○ Generate random parameters / *Migration behaviour* / ○ Generate the spiral behaviour during migration / *Attacking behaviour* / ○ Calculate the distance ○ Compute related planes ○ Update position of search agent end while ▪ Return the best solution
Sparrow search algorithm	2020	The group wisdom, foraging and anti-predation behaviours of sparrows.	The bionic principles (foraging strategy ant the anti-predation behaviours) of Sparrow Search Algorithm are as follows: ❖ The foraging strategy: There are two behaviour strategies (the discoverer's and the joiner's). As the leader of the population, the discoverer discovers the foraging area and specific direction for all. The joiner usually follows the discoverer. In addition, to increase their predation rate, some joiners will follow the discoverer for food. ❖ Anti-predation behaviour: The population conducts anti-predation behaviours in case of danger.	▪ Initialization ▪ While (stopping criterion) ○ Rank the fitness values, find the current best and worst individual ○ Generate alarm value ○ For each producer ➢ Update the sparrow's location ○ end for ○ For each sparrow ➢ Update the sparrow's location ○ end for ○ For each sparrow who perceive the danger ➢ Update the sparrow's location ○ end for ○ Get the current new location and update it, if necessary end while ▪ Return the best solution

3. DESIGNED TOOL and APPLICATIONS

In this study, a software tool was designed using MATLAB App Designer (Matlab, 2021). The main screen of this tool is given in Fig. 2. On this screen, when type of controller is determined by checking the related checkbox, the block diagram of the system is displayed and the required values (transfer function coefficients of controlled system and feedback) are entered. PID coefficients are tuned/calculated by selecting the metaheuristic algorithm and performance criteria. When the "Metaheuristic algorithm" type is selected, the "Parameters" window appears and the algorithm parameters are entered or changed. Finally, the performance criterion is selected. As a result of the analysis, time domain response (step response) - together with their parameters (maximum overshoot; peak, rise and settling times; steady state response and error) - is displayed on the screen. The basic flowchart of this process is given in Fig. 3.

As a first application, PI controller design which will be performed for the system with transfer function is given in Eq. (1) (Matlab, 2021). The steps of the design process using Arithmetic Optimization according to ITAE performance criteria are given in Fig. 4. The selection of controller type, the input of controlled system and feedback system transfer function coefficients, and selection of performance criteria are shown in Fig.4a. Entering upper and lower bounds/limits with the "Bounds" option in the "Settings" menu or the "Bounds" toolbar icon is given in Fig. 4b. The entering or changing of algorithm parameters by using the "Algorithm parameters" option in the "Settings" menu or by selecting from the "Metaheuristic algorithm" drop-down box is given in the Fig. 4c. The result screen including the designed controller coefficients, reference and system step response graphs and parameters is shown in Fig. 4d.

$$G(s) = \frac{1}{s^3+3s^2+3s+1} \tag{1}$$

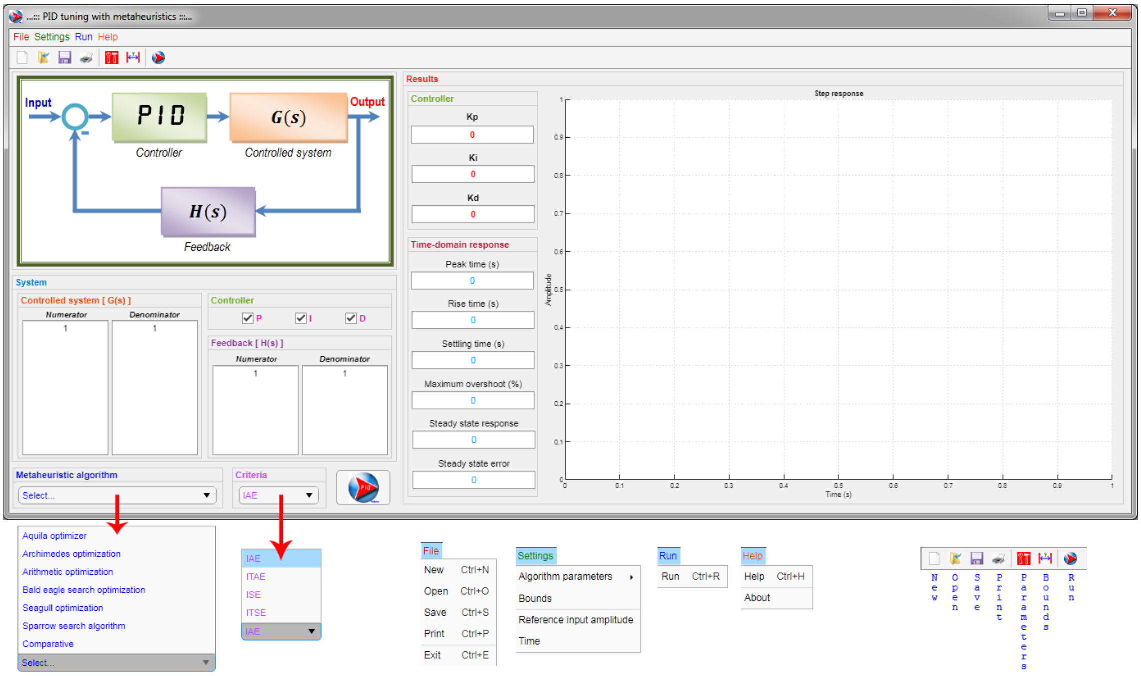


Figure 2:
The main screen of the designed software tool

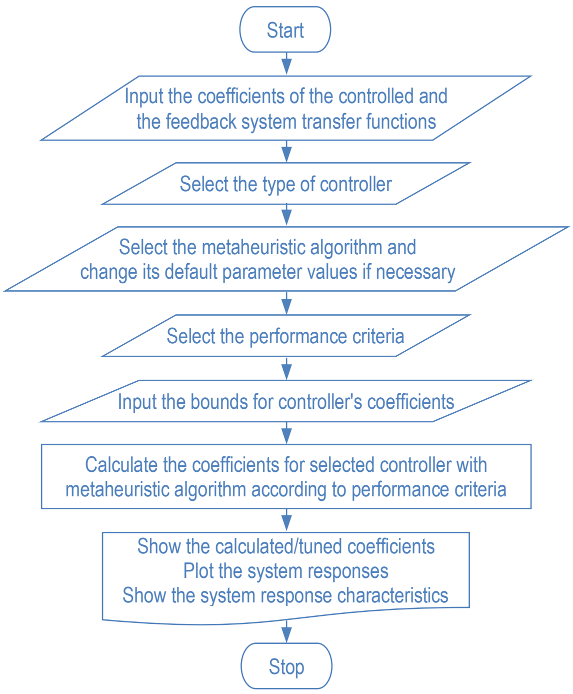
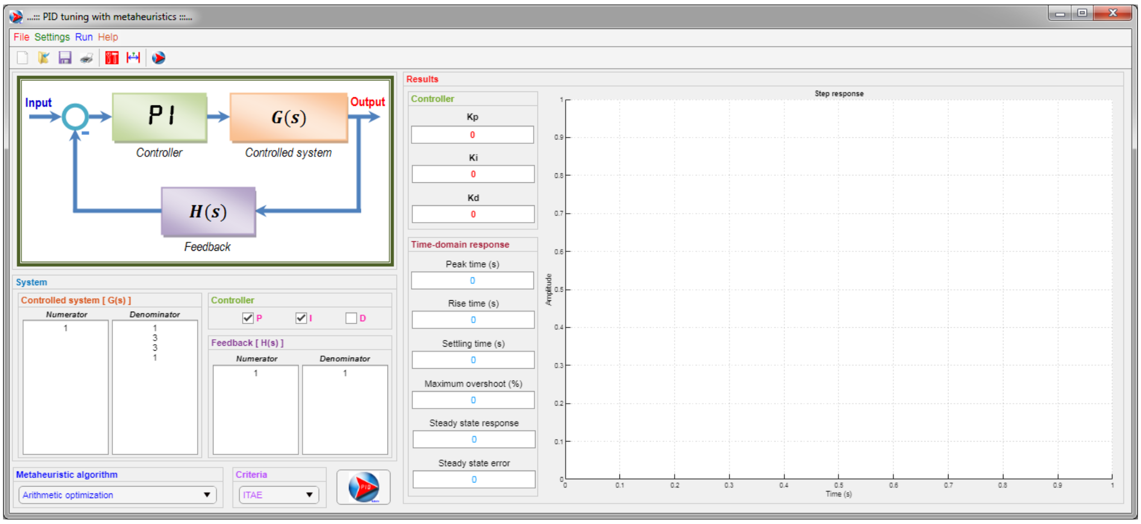
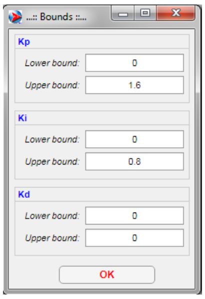


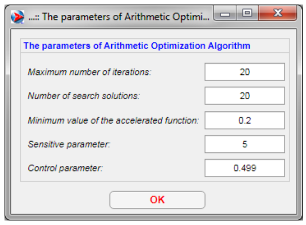
Figure 3:
The basic flowchart for process in the designed software tool



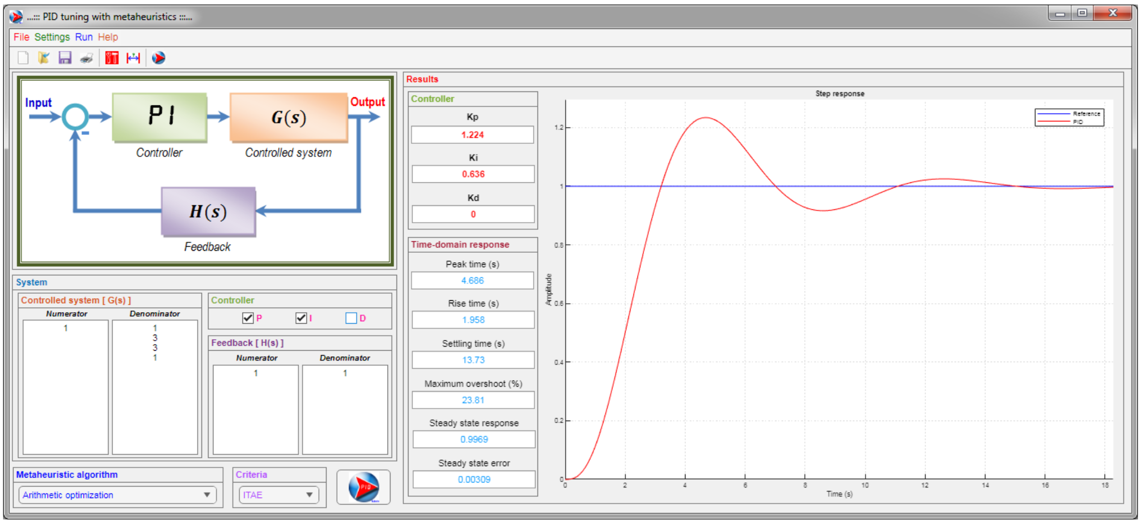
(a)



(b)



(c)

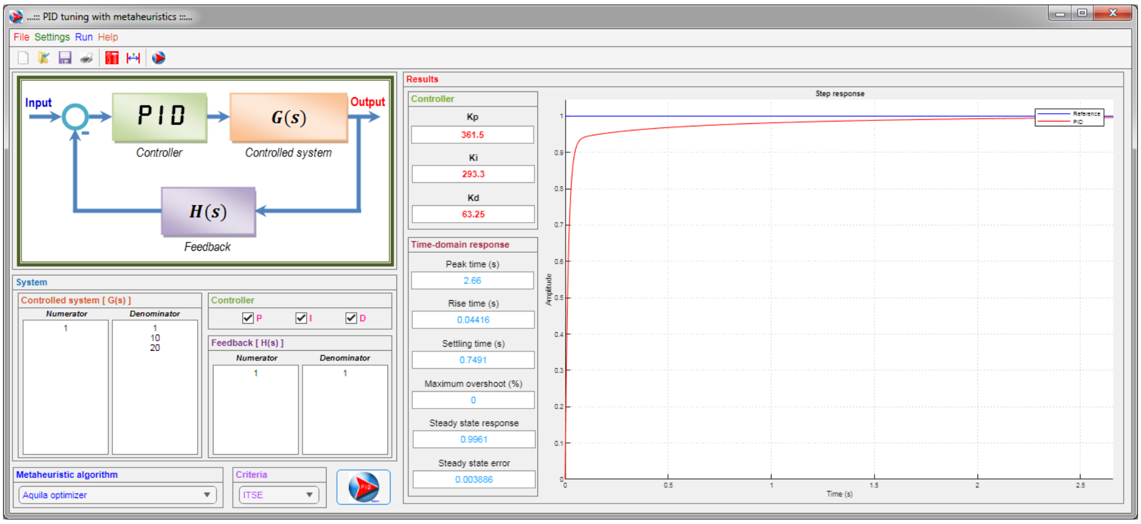


(d)

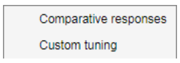
Figure 4:
The steps of the design process for first application

As a second application, PID controller design will be performed for the system with transfer function given in Eq. (2) (CTMS, 2022). The design is realized by using Aquila optimizer with ITSE criteria. The result screen is shown in Fig. 5a. Also a submenu (Fig. 5b) is opened by right-clicking on the step response graph. If "Comparative responses" is selected, system responses with and without controller are shown in a separate window (Fig. 5c) comparatively. Thus, the effect of the designed controller on the system can be clearly seen. If "Custom tuning" is selected, the screen in which the relevant coefficient values can be changed appears (Fig. 5d). In this window where individual or consecutive comparative responses can be observed, the effects of the coefficients on the system response can be examined (Fig. 5e).

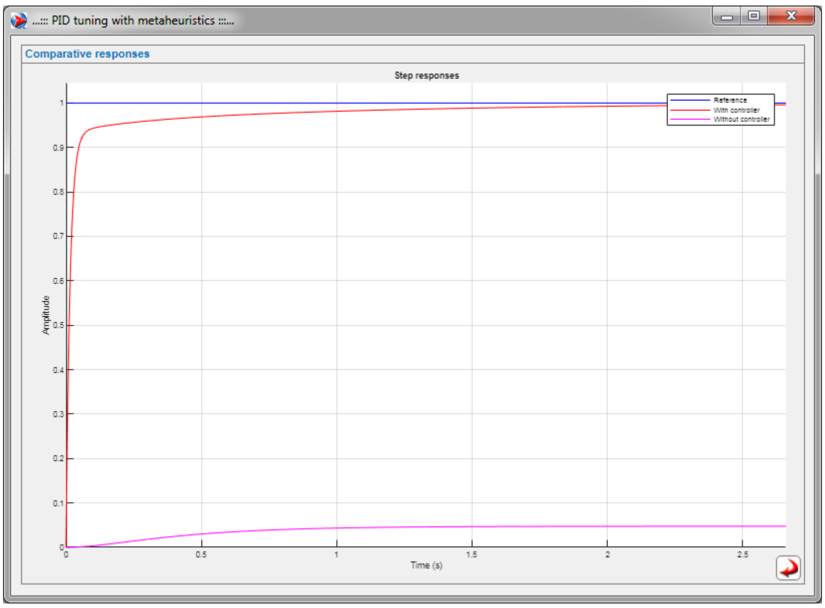
$$G(s) = \frac{1}{s^2+10s+20} \tag{2}$$



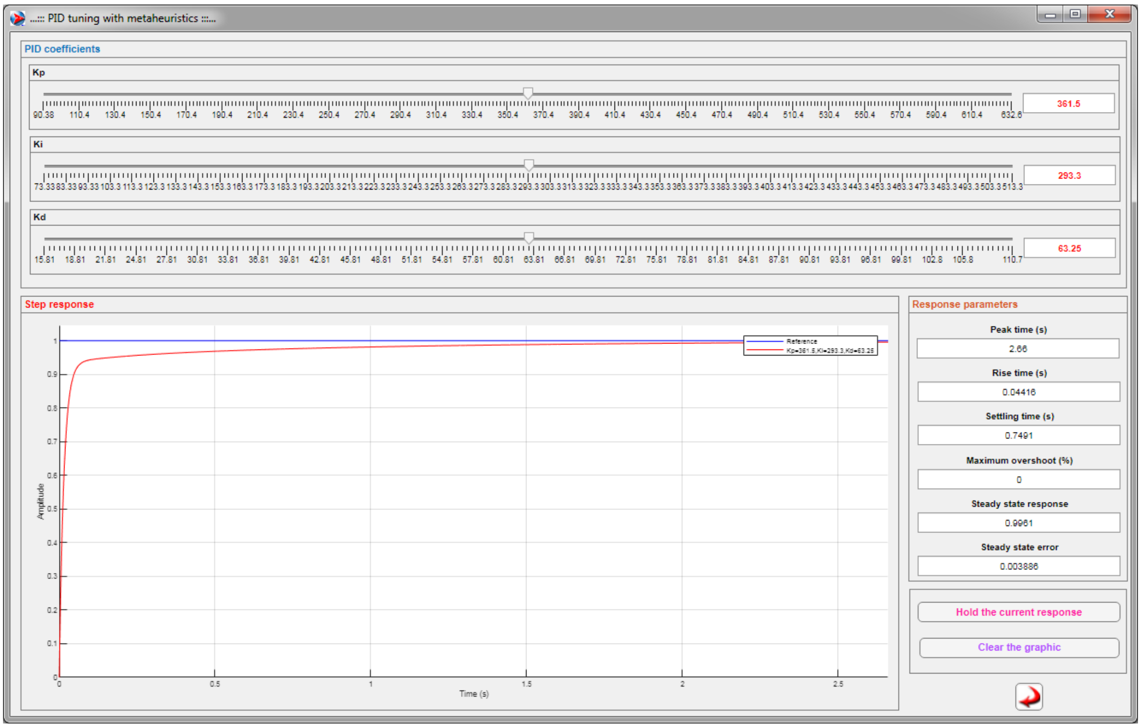
(a)



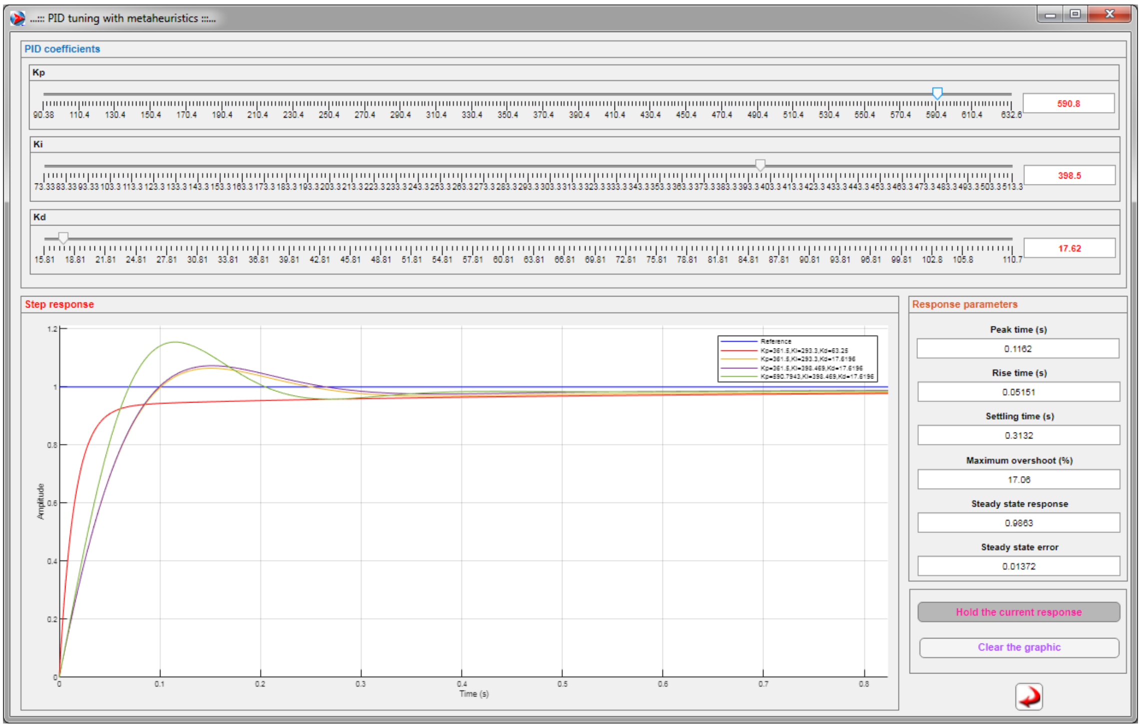
(b)



(c)



(d)



(e)

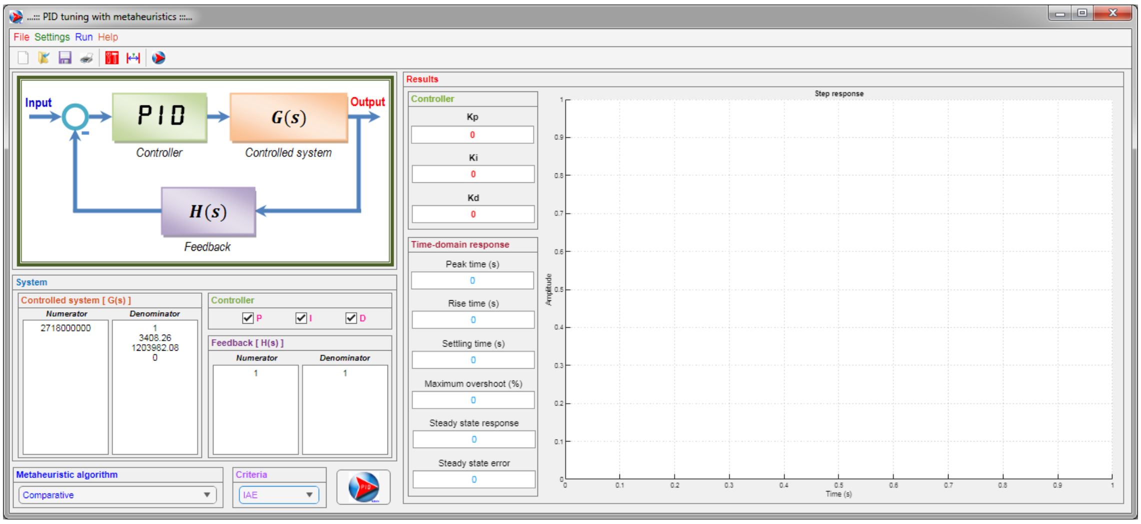
Figure 5:

The screenshots for second application

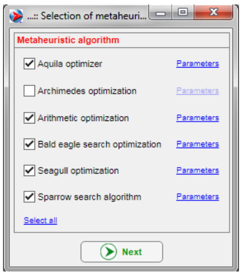
As a third application, comparative PID controller design will be performed for the system with transfer function given in Eq. (3) (Golnaraghi and Kuo, 2009).

$$G(s) = \frac{2718000000}{s(s+400.26)(s+3008)} \tag{3}$$

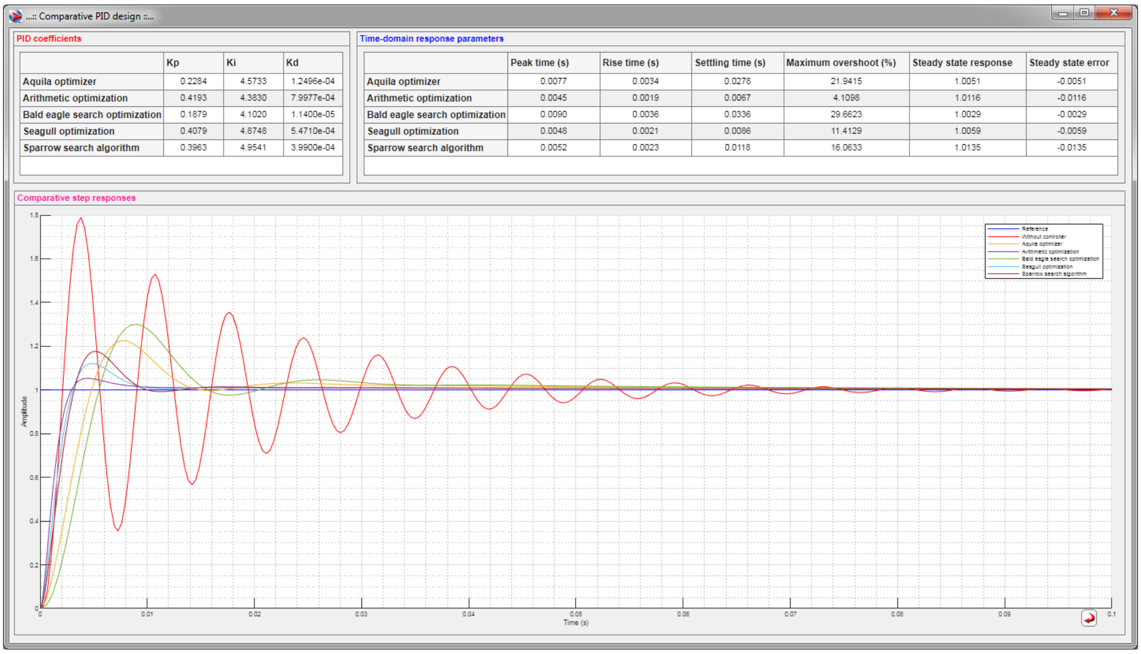
The steps of the design process using selected metaheuristic algorithms to IAE performance criteria are given in Fig. 6. When the "Comparative" option is selected in the "Metaheuristic algorithm" menu, the selection window is opened and the algorithms to be compared are selected (Fig. 6b). After determining the algorithms to be used and adjusting the parameters if it is desired, the window with the comparative results appears (Fig. 6c). In this window, the controller coefficients and the unit step response parameters of the systems with these controllers are given in tables. Also comparative step responses are provided.



(a)



(b)



	Kp	Ki	Kd
Aquila optimizer	0.2284	4.5733	1.2496e-04
Arithmetic optimization	0.4193	4.3830	7.9977e-04
Bald eagle search optimization	0.1879	4.1020	1.1400e-05
Seagull optimization	0.4079	4.8748	5.4710e-04
Sparrow search algorithm	0.3963	4.9541	3.9900e-04

	Peak time (s)	Rise time (s)	Settling time (s)	Maximum overshoot (%)	Steady state response	Steady state error
Aquila optimizer	0.0077	0.0034	0.0278	21.9415	1.0051	-0.0051
Arithmetic optimization	0.0045	0.0019	0.0067	4.1098	1.0116	-0.0116
Bald eagle search optimization	0.0090	0.0036	0.0336	29.6623	1.0029	-0.0029
Seagull optimization	0.0048	0.0021	0.0086	11.4129	1.0059	-0.0059
Sparrow search algorithm	0.0052	0.0023	0.0118	16.0633	1.0135	-0.0135

(c)

Figure 6:

The screenshots for third application

4. CONCLUSIONS

In this study, an interactive software tool was carried out for tuning PID coefficients with up-to-date metaheuristic algorithms. With this tool, the desired P, PI, PD or PID controller designs for the feedback system were defined by the user can be made with selected metaheuristic algorithms and under specified performance criteria. Also with developed software tool that provides numerical and graphical results, single and comparative designs can be made quickly, easily and effectively. In addition, both the effects of algorithm parameters on PID tuning and controllers on the system responses can be clearly observed.

CONFLICT OF INTEREST

Authors approves that to the best of his knowledge, there is not any conflict of interest or common interest with an institution/organization or a person that may affect the review process of the paper.

AUTHOR CONTRIBUTION

Fahri Vatansever: Conceptualization, investigation, methodology, software, validation, visualization, writing-original draft, writing-review & editing.

Emre Hacıiskenderoğlu: Resources, validation, writing-review & editing.

REFERENCES

1. Abualigah, L., Diabat, A., Mirjalili, S., Elaziz, M.A., Gandomi, A.H. (2021a) The Arithmetic optimization algorithm, *Computer Methods in Applied Mechanics and Engineering*, 376, Art no. 113609. doi: https://doi.org/10.1016/j.cma.2020.113609

2. Abualigah, L., Yousri, D., Elaziz, M.A., Ewees, A.A., Al-qaness, M.A.A., Gandomi, A.H. (2021b) Aquila optimizer: A novel meta-heuristic optimization algorithm, *Computers & Industrial Engineering*, 157, Art no. 107250. doi: https://doi.org/10.1016/j.cie.2021.107250

3. Abushawish, A., Hamadeh, M., Nassif, A.B. (2020) PID Controller gains tuning using metaheuristic optimization methods: A survey, *International Journal of Computers*, 14, 87-95. doi: http://doi.org/10.46300/9108.2020.14.14

4. Alsattar, H.A., Zaidan, A.A., Zaidan, B.B. (2020) Novel meta-heuristic bald eagle search optimisation algorithm, *Artificial Intelligence Review*, 53, 2237-2264. doi: https://doi.org/10.1007/s10462-019-09732-5

5. Control Tutorials for MATLAB and Simulink (CTMS), (2022). Access address: https://ctms.engin.umich.edu/CTMS/index.php?example=Introduction&section=ControlPID (Accessed in: 01.02.2022)

6. Dhiman, G., Kumar, V. (2019) Seagull optimization algorithm: Theory and its applications for large-scale industrial engineering problems, Knowledge-Based Systems, 165, 169-196. doi: https://doi.org/10.1016/j.knosys.2018.11.024

7. Golnaraghi, F., Kuo, B.C (2009) *Automatic Control Systems*, 9th ed., John Wiley & Sons, USA.

8. Hashim, F.A., Hussain, K., Houssein, E.H., Mabrouk, M.S., Al-Atabany, W. (2021) Archimedes optimization algorithm: a new metaheuristic algorithm for solving optimization problems, *Applied Intelligence*, 51, 1531-1551. doi: https://doi.org/10.1007/s10489-020-01893-z

9. MATLAB, (2021). The MathWorks Inc. https://www.mathworks.com/

10. Nise, N.S. (2015) *Control Systems Engineering*, 7th ed., John Wiley & Sons, USA.

11. Oladipo, S., Sun, Y., Wang, Z. (2020) Optimization of PID controller with metaheuristic algorithms for DC motor drives: Review, *International Review of Electrical Engineering (I.R.E.E.)*, 15(5), 352-381. doi: https://doi.org/10.15866/iree.v15i5.18688

12. 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, Art no. 106342. doi: https://doi.org/10.1016/j.asoc.2020.106342

13. Vatansever, F., Sen, D. (2013) Design of PID controller simulator based on genetic algorithm, *Uludağ University Journal of the Faculty of Engineering*, 18(2), 7-18.

14. Xue, D., Chen, Y.Q., Atherton, D.P. (2007) *Linear Feedback Control (Analysis and Design with MATLAB)*, SIAM, USA.

15. Xue, J., Shen, B. (2020) A novel swarm intelligence optimization approach: sparrow search algorithm, *Systems Science & Control Engineering*, 8(1), 22-34. doi: https://doi.org/10.1080/21642583.2019.1708830

16. Ziegler, J.G., Nichols, N.B. (1942) Optimum settings for automatic controllers, *Transactions of the A.S.M.E.*, 64, 759-768.