

Article

Minimum Energy Control of Quadrotor UAV: Synthesis and Performance Analysis of Control System with Neurobiologically Inspired Intelligent Controller (BELBIC)

Wojciech Giernacki 

Faculty of Automatic Control, Robotics and Electrical Engineering, Institute of Robotics and Machine Intelligence, Poznan University of Technology, ul. Piotrowo 3a, 60-965 Poznan, Poland; wojciech.giernacki@put.poznan.pl

Abstract: There is a strong trend in the development of control systems for multi-rotor unmanned aerial vehicles (UAVs), where minimization of a control signal effort is conducted to extend the flight time. The aim of this article is to shed light on the problem of shaping control signals in terms of energy-optimal flights. The synthesis of a UAV autonomous control system with a brain emotional learning based intelligent controller (BELBIC) is presented. The BELBIC, based on information from the feedback loop of the reference signal tracking system, shows a high learning ability to develop an appropriate control action with low computational complexity. This extends the capabilities of commonly used fixed-value proportional–integral–derivative controllers in a simple but efficient manner. The problem of controller tuning is treated here as a problem of optimization of the cost function expressing control signal effort and maximum precision flight. The article introduces several techniques (bio-inspired metaheuristics) that allow for quick self-tuning of the controller parameters. The performance of the system is comprehensively analyzed based on results of the experiments conducted for the quadrotor model.



Citation: Giernacki, W. Minimum Energy Control of Quadrotor UAV: Synthesis and Performance Analysis of Control System with Neurobiologically Inspired Intelligent Controller (BELBIC). *Energies* **2022**, *15*, 7566. <https://doi.org/10.3390/en15207566>

Academic Editor: Francisco Manzano Agugliaro

Received: 9 September 2022

Accepted: 9 October 2022

Published: 13 October 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: UAV; quadrotor; optimization; minimum energy control; brain emotional learning; BELBIC

1. Introduction

1.1. Background

In recent years, there has been a growing interest in unmanned aerial vehicles (UAVs) [1,2]. Among the various types of UAVs, multi-rotor robots are particularly interesting due to their small size, good flight properties (including the possibility of hovering and flying stably at very low speeds), and relatively low cost [3]. In each of the diverse missions (transportation, agricultural, industrial, photogrammetry, reconnaissance, surveillance, etc.), UAV features such as maximum flight time and smooth, non-overshooting flight trajectories are in demand. These properties determine the safety of control of this inherently unstable and underactuated plant. The appropriate selection of controllers and their proper tuning are of prime importance since they allow the optimal use of highly limited energy resources to generate the appropriate thrust and torques of the particular propulsion units of the UAV.

Nowadays, numerous types of controllers are used in multidimensional UAV control systems [4]. In addition to a number of advanced solutions in which the control system is able to autonomously control the UAV with rapidly changing, time-varying aerodynamic characteristics during flight (briefly characterized in [5]), techniques based on model predictive control (MPC) [6], fuzzy control [7], sliding mode control (SMC) [8], and adaptive fault-tolerant control [9] are widely used. In addition to these techniques, many new ones have appeared [10–13] which are related to advanced intelligent control of nonlinear systems and may be easily adaptable to UAVs. However, the most common commercially available multi-rotor UAVs use solutions based on classical fixed-value feedback controllers of

proportional–derivative (PD) or proportional–integral–derivative (PID) type [14]. They provide good trajectory tracking and do not require a UAV dynamics model.

1.2. Research Motivation

The research motivation was based on the idea that the quality of the above-mentioned fixed-value feedback controllers may be improved by:

- The use of appropriate techniques to optimize their gains.
- Inclusion of fixed-value controllers in the structure of intelligent controllers.

In this article, both improvement solutions are presented. The research was also indirectly inspired by [15], where the SafeOpt algorithm (based on Bayesian optimization) was proposed to solve the problem of automatic adjustment of the controller parameters to ensure a more precise flight. Interesting results are also presented in [16], in which the authors, via an in-flight run of the modified relay feedback test, looked for near-optimal tuning of the quadrotor attitude controllers. In previous articles [17–19], we proposed deterministic optimization methods based on modified zero-order iterative algorithms (Fibonacci-search, golden-search) for in-flight auto-tuning of UAV controllers. These methods of automatic tuning of the gains of fixed-value controllers on the basis of machine learning (iterative learning) algorithms allow, by comparing the obtained values of the cost function for various combinations and of gains, to search for the (locally) optimal gains for specific expectations expressed by the mathematical formula of the cost function. As a result, controllers capable of increasing the tracking precision of the UAV reference trajectory are obtained, and in the case of [19], the energy consumption of the UAV is indirectly optimized by introducing a penalty mechanism for large picks of control signals (included in the optimized cost function value). This mechanism forms the basis for the minimum energy control considerations in this paper. Moreover, in the current article, attention is focused on batch tuning of controllers, as our previous techniques for in-flight tuning of gains of fixed-value UAV controllers do not guarantee stability during the tuning process. Therefore, the in-flight approaches are predefined more for the successive improvement of controller gain in the daily exploitation of drones, and pre-tuning can be performed based on the approach proposed here.

In the synthesis of UAV control systems, in the process of optimizing gains described above, techniques inspired by examples from nature have been used successfully [20]. Naturally, since 2014, when Duan and Li published their book, at least a dozen new and now well-recognized algorithms have been proposed. The most interesting include, among others, the cuttlefish algorithm (CFA) [21], Harris hawks optimization [22], a mayfly optimization algorithm [23], jellyfish search [24], golden eagle optimizer [25], and firebug swarm optimization [26]. In this article, it was decided to use two of them in the synthesis of a UAV autonomous control system with a neurobiologically inspired intelligent controller. They are, respectively: the particle swarm optimization (PSO) and cuttlefish algorithm, used in the author's earlier works, including optimization in nonlinear MPC [27] and auto-tuning of a UAV altitude controller [21]. The preliminary research and the promising results obtained in the previous works were a direct motivation for the selection of these algorithms.

1.3. The State of the Art

Since obtaining perfect accuracy of the nonlinear mathematical model of UAV dynamics is a challenging task, model-free, soft-computing-based controllers are preferred. Especially those that are robust and can deal with real-world environmental complexities and disturbances. Furthermore, the capability of self-learning is expected (with a small computational cost and with simple self-adapting mechanisms). Thus, intelligent controllers (especially PID-type-based) with learning capability are a prime solution to provide appropriate control actions in UAV autonomous low-level control. It is expected that these controllers will have a simple structure and fewer parameters to be tuned than the neural networks and fuzzy equivalents.

Within the universe of model-free neurobiologically inspired intelligent controllers, solutions with low computational complexity are gaining in importance every year—especially those offering a transparent, analytical structure of the control system, as well as reward- and penalty-based learning mechanisms in the controller’s response to external emotional stimuli. Emotional learning is one of the leading psychologically motivated learning strategies (Figure 1), which is then used to produce control actions (emotional response) in the output of intelligent neuromorphic controllers based on the desired and actual system response (sensory input). The emotional learning process is based on emotional evaluations. According to [28], ‘emotions play an essential role in rational decision-making, perception, learning, and a variety of other cognitive functions’. Additionally, emotions help humans survive and react immediately in an emergency or danger. Emotional stimuli (for example, fear, aggression, and anxiety) cause emotional behaviors and the brain’s quick reactions to danger, often far from the complex reasoning and logic. The evaluation of the effect of such an emotional reaction is firmly established in the learning process. Additional actions are not caused only by rational reasoning but are determined and biased by emotions. Thus, emotions may be considered as ‘a tacit expert system’ [29]. In [30], the concept of dividing brain work into rational mind and emotional mind is presented. The latter has a key important feature, which is an extremely fast reaction. Fast emotional processing is possible [31] due to synapses (inhibitory connections) and short pathways in the limbic system of the emotional brain.

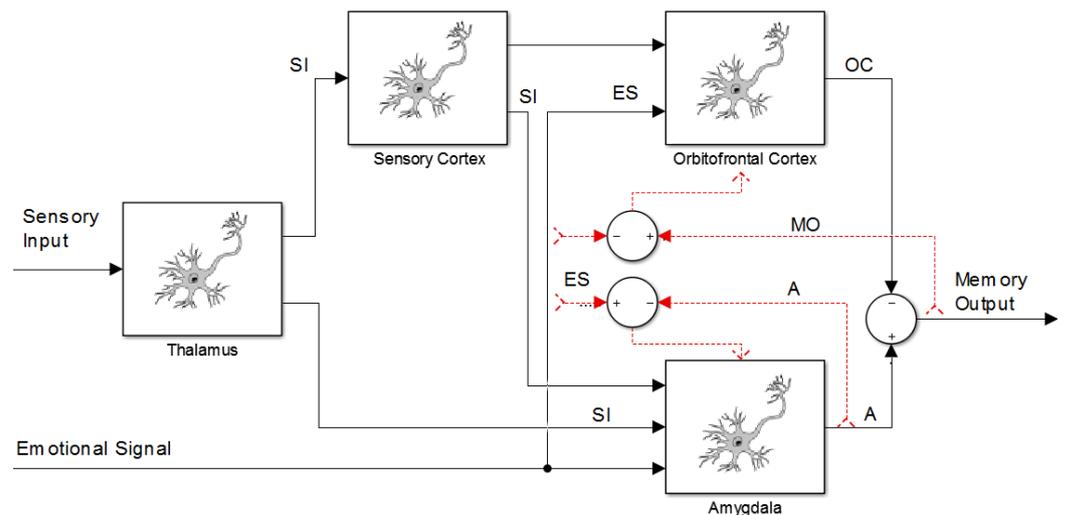


Figure 1. Block diagram of the BEL model (briefly characterized in Section 1.4), proposed by Moren and Balkenius [29], where: SI—sensory input, ES—emotional signal, OC—orbitofrontal cortex, A—amygdala, MO—memory output.

In 2004, Lucas Caro et al., in [32], proposed the idea of a brain emotional learning based intelligent controller (BELBIC), where context processing and an open-loop cognitive amygdala-OFC model created by Moren and Balkenius [29] in 2000 were used (Figure 1). Since 2000, BELBIC controllers have been successfully used, e.g., in developing a new fuzzy neural network by incorporating a BELBIC with fuzzy inference rules [33]. Its performance was evaluated on the model of an inverted double pendulum system. In [34], implementation of the intelligent adaptive controller for an electrohydraulic servo system based on the brain emotional learning (BEL) mechanism was presented. Joao Paulo Coelho et al. adapted BELBIC controllers into two control systems: (a) one with a mathematical model of magneto-rheological (MR) damper [35], and (b) one with a model of a non-collocated three-story building with MR [36], respectively. Lucas Caro et al. applied an intelligent controller to the neurofuzzy microheat exchanger model in [37,38] to control the laboratory overhead travel crane in a model-free and embedded manner. Another interesting paper by Lucas Caro et al. is the control of the speed and flux of induction motors using a BELBIC

controller [39]. The last publication worth mentioning is [40], where fuzzy inference is designed to tune the BELBIC reward function parameter online that is used to control the electrohydraulic actuator.

There is relatively little research on this control strategy in the field of aviation and aerospace. In [41], intelligent autopilot design may be found for a nonlinear model of an autonomous helicopter using an adaptive emotional approach. Valencia and Kim, in [42], used BELBIC to build a control system capable of autonomously operating multiple quadrotors in the leader–follower configuration. Interesting research may be found in the works of Jafari et al., especially [43,44], where real-time flocking control of multi-agent systems in the presence of system uncertainties and dynamic environments and distributed intelligent flocking control of networked multi-UAS were considered, respectively.

1.4. BELBIC—General Idea and Areas for Improvements

A mathematical model of the limbic system of the human brain (Figure 1) with areas responsible for emotional learning and processing such as the orbitofrontal cortex, the amygdala, the sensory cortex, and the thalamus has been developed in BELBIC controllers with the use of an artificial network with adaptable parameters (adjustable gains)—details of mathematical formulas are provided in Section 2.3. In both cases, i.e., biological and artificial brain models, two networks affecting each other: sensory neural network (SNN) and emotional neural network (ENN) build an internally interconnected system. SNN is used to simulate the brain's orbitofrontal cortex and is responsible for the major output of the BELBIC controller, while the ENN is used to represent the amygdala cortex, and it undergoes stimulation by external factors and has indirect impact on the SNN. This structure of the brain model conforms to Mowrer's cognitive theory of the learning process. The theory states that emotional evaluation (connection of response with stimulus) occurs after association of the stimulus with an emotional consequence. Therefore, emotions in the sensory learning system can be used as constant feedback. They also provide information to evaluate the level of success in applying control actions and to provide a new control.

The emotional response from the emotional brain will appear when input of stimuli from environments will put the state of the limbic system out of balance. This reaction is learning- and adaptation-oriented to provide a higher level of robustness to a constantly changing environment. In the sensorial switching station, which is the thalamus, stimulus inputs are gathered and preprocessed. That sensory data are being properly distributed to the amygdala or to the cerebral cortex (sensory and orbitofrontal cortex). A center for the processing of emotional behavior is the amygdala, which communicates with all other areas of the limbic system. It responds to emotional stimuli, since here, as LeDoux found in 1992 [45], the association between a stimulus and its emotional consequence has a place. The stimulus in the learning process needs to be paired with a primary reinforcer (the reward and penalty), which in the artificial BEL model can be freely chosen. The amygdala interacts with the OFC, which evaluates the response of the amygdala and prevents inappropriate responses based on the context. In the OFC, negative reinforcement signals are being generated. These signals are used to inhibit and mitigate inappropriate signals generated by the amygdala, when there is a difference between the expected prediction of the system and the actual emotional signal received. OFC controls learning extinction in the amygdala to give a proper emotional signal.

In the context of the control theory, using the mimicry of the cognitive functions of the brain, introducing in the control system additional information from the feedback from the control signal to the controller input and the reinforcement critic mechanism, allows for a smooth transition from fixed-value control to the intelligent model-free BELBIC. During the control system synthesis, the emotional signal and sensory input are considered as arbitrary cost functions of signals such as control error, control signal or reference signal—depending on the needs and expectations of the control system designer. It is worth mentioning that finding an appropriate cost function is not a trivial task.

One needs to remember that the BELBIC controller has basically two disadvantages. The first one is similar to fuzzy and neural network controllers—it requires some initial knowledge (expert knowledge) about the control system in its synthesis process. However, unlike these controllers, here, just the functions of sensory input and reward (emotional) signal need to be appropriately arbitrarily chosen by the control system designer. The second drawback is, unfortunately, the difficult design of such controllers during UAV flight, because there is no guarantee of full stability [46]. Therefore, in this article, it was decided to use at the prototyping stage the closed-loop control system model with controllers and plant models—although the BELBIC controller itself is of the “model-free” type. The advantage of this approach is also the possibility of using bio-inspired heuristic optimization methods for shaping transients in flight trajectories.

1.5. Main Contribution

It is proposed to extend the capabilities of the brain emotional learning based intelligent controller in the field of autonomous UAV control, using metaheuristic optimization techniques to conduct the most precise flight of the drone in a minimal energy manner.

The added value of this research is as follows:

- Synthesis of the autonomous drone control system with the BELBIC controller in the proposed controller structures and model of UAV dynamics.
- Formulation of an optimization problem in order to optimize the gains of the BELBIC controller in terms of minimizing the energy expenditure of the UAV flight for selected optimization solvers.
- Evaluation of the performance of the proposed control system by means of numerical experiments, including providing knowledge of whether the proposed method of control can extend the flight time of the drone while increasing the precision of the flight in relation to the fixed-value controller approach.

1.6. Study Outline

The paper is organized as follows: in Section 2, the synthesis of the autonomous control system is presented. Furthermore, here, one may find details of the UAV model. The BELBIC controller paradigm is explained. Section 3 is the most important one, since here the proposed improvements to BELBIC are shown to obtain the minimum energy control of the UAV. The optimization problem is outlined for the chosen sensory input and emotional signal functions. Furthermore, this section contains the necessary details of bio-inspired optimization algorithms used to find the solution of the optimization problem, i.e., gains of BELBIC controllers to autonomously control the UAV. In Section 4, one may find the representative results obtained from the numerical experiments carried out to validate the minimum energy control of the UAV. Extensive performance analysis of the BELBIC-based UAV autonomous control system is shown. Section 5 summarizes the article and future research plans are described. The meaning of symbols used in the paper are found in Table 1.

Table 1. Meaning of symbols used in the paper.

Symbol	Meaning
a_1, a_3, a_3	translational air drag coefficients
a_4, a_5, a_6	aerodynamic friction coefficients
b	thrust constant
c_i, s_i	cognitive, social vectors of PSO algorithm
d	drag factor
$e(t)$	control error
F_d	translational drag force
F_T	thrust force generated by UAV rotors

Table 1. *Cont.*

Symbol	Meaning
g	gravitational acceleration
I	UAV symmetrical inertia matrix
I_R	rotor inertia
J	performance index (cost function in optimization procedure)
k_1, \dots, k_n	gains of BELBIC controller
l	distance between the propulsion unit axis and the UAV's center of the mass
m	UAV mass
p	vector of measured coordinates of UAV position
p_i, v_i	position, velocity vectors of PSO algorithm
R	rotation matrix
t_h	flight time horizon
$u(t)$	control signal (in time domain)
V_i	amygdala i -th gain
W_i	orbitofrontal i -th gain
w	UAV output vector
x_d, y_d, z_d	reference, desired coordinates of UAV position
x, y, z	measured position of the UAV
$y(t)$	output signal
α, β	learning coefficients for the amygdala and orbitofrontal cortex
ϕ_1, ϕ_2	cognition and social constants in PSO algorithm
ϕ_d, θ_d, ψ_d	reference, desired <i>pitch</i> , <i>roll</i> , and <i>yaw</i> angles
Ω	vector of measured UAV angles
\mathcal{BF}	body frame of reference
\mathcal{EF}	Earth frame of reference
τ_x, τ_y, τ_z	roll, pitch, and yaw torques applied to the body of the UAV
Γ	weight coefficient for the control error in J cost function
Ψ	weight coefficient for the control signal in J cost function
λ	inertia weight in PSO algorithm

2. Control System Synthesis

2.1. Autonomous Control of the UAV

Let us consider the autonomous control system of a quadrotor UAV from Figure 2, where four input signals are enough to stabilize all of the drone's six degrees of freedom (expressed by position and orientation vectors in 3D space) and to provide precise tracking of the predefined drone flight path. It is possible, since there are two control loops: (a) position control (outer, slower) and (b) attitude control (inner, faster). In this architecture, input signals may be written as a reference vector:

$$v = [x_d \ y_d \ z_d \ \psi_d]^T, \quad (1)$$

where x_d , y_d , and z_d reference the desired coordinates of UAV position in 3D, and ψ_d the desired rotation around the z -axis. All four reference signals are defined according to the Earth coordinate system $\{\mathcal{EF}\}$ (see Figure 3 for details).

To describe the UAV measured orientation and position in the 3D space during its autonomous flight, two vectors are introduced. The first one, describing the UAV measured position according to $\{\mathcal{EF}\}$, is

$$p = [x, y, z]^T, \quad (2)$$

where x , y , and z are current coordinates of the UAV position in 3D.

The second vector used in the UAV output description is

$$\Omega = [\varphi, \theta, \psi]^T \quad (3)$$

for current orientation, where φ , θ , and ψ are the roll, pitch, and yaw measured angles, respectively.

On the base of Equations (2) and (3), the UAV output may be written as a following vector:

$$w = [p \ \Omega]^T = [x \ y \ z \ \varphi \ \theta \ \psi]^T. \tag{4}$$

Often, first, second, and even third derivatives of a vector’s w elements are used in struggling with UAV stabilization and for desired trajectory tracking. In this research, in every moment of flight, one may use the comparison (difference) between corresponding elements of v and w vectors—known as control errors. These control errors are used next in BELBIC controllers to provide proper control actions, as well as to calculate the cost function value in controller gains optimization in order to find minimum energy control of the UAV—details in Section 3.

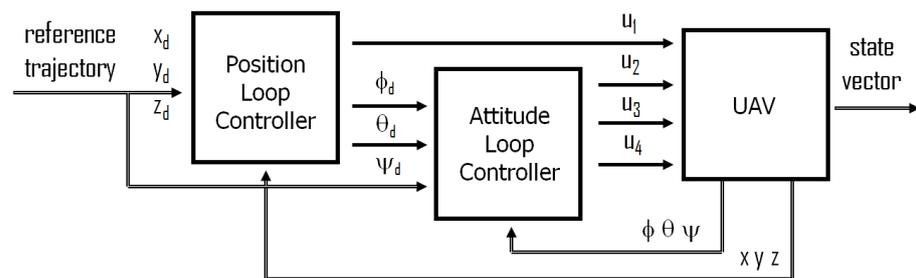


Figure 2. Block diagram for autonomous control of the UAV (thrust and torques u_i for $i = 1, \dots, 4$ are defined in Equation (15)).

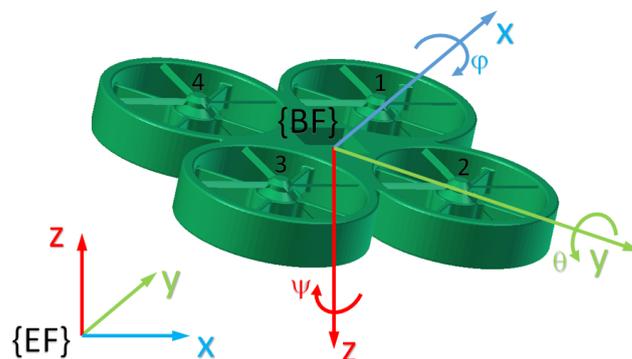


Figure 3. Reference frames related to the X4-flyer II simplified graphics. Left (4) and right (2) propulsion units rotate clockwise, while the front (1) and rear (3) counterclockwise.

2.2. Quadrotor Model

Quadrotor UAVs are currently the most widely used multi-rotor drones. Therefore, let us consider the nonlinear model of such a UAV dynamic, originally published in [47] and adapted here with an additional translational drag effect. It is further assumed in the research that the quadrotor UAV behaves like a rigid body with construction mass accumulated to its geometric center, and it has four rotors with symmetrically distributed propellers at each cross-type frame end.

Model of the UAV from Figure 3 is based on ‘+’ type layout configuration, where the x -axis of the UAV’s local coordinate system (body frame— $\{BF\}$) coincides with the line drawn from the back propulsion unit to the front one, the y -axis is perpendicular to the right, and the z -axis is looking down according to the right-hand rule, respectively. These axes conventions with regard to the observer’s coordinate system (Earth frame— $\{EF\}$) is the so-called North–East–Down (NED). In the UAV control and measurements, conversions between $\{BF\}$ and $\{EF\}$ are necessary. The rotation matrix $R \in SO3$ from $\{BF\}$ to $\{EF\}$ is

$$R_{ZYX}(\psi, \theta, \varphi) = R_Z(\psi)R_Y(\theta)R_X(\varphi), \tag{5}$$

where $R_Z(\psi)$, $R_Y(\theta)$, and $R_X(\varphi)$, are Euler angle matrices as follows

$$R_Z(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (6)$$

$$R_Y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}, \quad (7)$$

$$R_X(\varphi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & -\sin\varphi \\ 0 & \sin\varphi & \cos\varphi \end{bmatrix}. \quad (8)$$

Using Equations (6)–(8), the matrix $R_{ZYX}(\psi, \theta, \varphi)$ from Equation (5) can be written as

$$R_{ZYX}(\psi, \theta, \varphi) = \begin{bmatrix} c\psi c\theta & c\psi s\theta s\varphi - s\psi c\varphi & c\psi s\theta c\varphi + s\psi s\varphi \\ s\psi c\theta & s\psi s\theta s\varphi + c\psi c\varphi & s\psi s\theta c\varphi - c\psi s\varphi \\ -s\theta & c\theta s\varphi & c\theta c\varphi \end{bmatrix}, \quad (9)$$

where $c = \cos$, and $s = \sin$.

The mathematical model that describes the UAV position may be written according to Newton's second law of motion as

$$m\ddot{p} = -F_g + F_T - F_d, \quad (10)$$

where m —UAV mass, $F_g = [0 \ 0 \ g]^T$ —gravitational force on Earth, g —gravitational acceleration, and $F_T = [0 \ 0 \ T]^T$ —thrust force generated by four rotors, and

$$T = b \sum_{i=1}^4 \omega_i^2, \quad (11)$$

where b —thrust constant and ω_i —rotational speed of the rotor i .

For UAV's airframe, the translational drag force may be written as

$$F_d = [a_1\dot{x} \ a_2\dot{y} \ a_3\dot{z}]^T, \quad (12)$$

where a_1 , a_2 , and a_3 —translational air drag coefficients.

It is now possible to rewrite the Equation (10) for the position of the UAV in the following form:

$$\ddot{p} = -g \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \mathbf{R} \frac{b}{m} \sum_{i=1}^4 \omega_i^2 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} - \frac{1}{m} \begin{bmatrix} a_1\dot{x} \\ a_2\dot{y} \\ a_3\dot{z} \end{bmatrix}, \quad (13)$$

while its orientation is considered according to Euler's rotation equation:

$$I\ddot{\Omega} = -\dot{\Omega} \times I\dot{\Omega} - \sum_{i=1}^4 I_R \left(\dot{\Omega} \times \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right) \omega_i + \begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} + \begin{bmatrix} a_4\dot{\phi}^2 \\ a_5\dot{\theta}^2 \\ a_6\dot{\psi}^2 \end{bmatrix}, \quad (14)$$

where I —symmetrical inertia matrix, I_R —rotor inertia, τ_x , τ_y , and τ_z —roll, pitch, and yaw torques applied to the body of the vehicle, and a_4 , a_5 , and a_6 —aerodynamic friction coefficients.

For the considered quadrotor UAV in ‘+’ type layout configuration, the control input vector is

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} T \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = \begin{bmatrix} -b & -b & -b & -b \\ 0 & -lb & 0 & lb \\ lb & 0 & -lb & 0 \\ d & -d & d & -d \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix}, \tag{15}$$

where l —distance between the propulsion unit axis and the UAV’s center of the mass, and d —drag factor.

After transformations, Equations (13) and (14) formulate the final description of the UAV’s dynamics

$$\begin{cases} \ddot{x} = (\sin\varphi\sin\psi + \sin\theta\cos\varphi\cos\psi)\frac{u_1}{m} - \frac{a_1}{m}\dot{x} \\ \ddot{y} = (-\sin\varphi\cos\psi + \sin\theta\cos\varphi\sin\psi)\frac{u_1}{m} - \frac{a_2}{m}\dot{y} \\ \ddot{z} = -g + \cos\varphi\cos\theta\frac{u_1}{m} - \frac{a_3}{m}\dot{z} \\ \ddot{\varphi} = \left(\frac{I_{yy}-I_{zz}}{I_{xx}}\right)\dot{\theta}\dot{\psi} - \frac{I_R}{I_{xx}}\dot{\theta}\omega_d + \frac{u_2}{I_{xx}} - \frac{a_4}{I_{xx}}\dot{\varphi}^2 \\ \ddot{\theta} = \left(\frac{I_{zz}-I_{xx}}{I_{yy}}\right)\dot{\varphi}\dot{\psi} + \frac{I_R}{I_{yy}}\dot{\varphi}\omega_d + \frac{u_3}{I_{yy}} - \frac{a_5}{I_{yy}}\dot{\theta}^2 \\ \ddot{\psi} = \left(\frac{I_{xx}-I_{yy}}{I_{zz}}\right)\dot{\varphi}\dot{\psi} + \frac{u_4}{I_{zz}} - \frac{a_6}{I_{zz}}\dot{\psi}^2 \end{cases}, \tag{16}$$

where I_{xx} , I_{yy} , and I_{zz} —inertias on the main diagonal of the matrix I , and

$$\omega_d = \omega_2 + \omega_4 - \omega_1 - \omega_3. \tag{17}$$

2.3. BELBIC Controller Design

Let us consider the BELBIC controller structure illustrated in Figure 4, introduced by Lucas Caro in [32], where he adjusted Moren and Balkenius’s computational model of emotional learning in the amygdala (so-called BEL model) [29] to create a feedback mechanism in the closed-loop control system with Sensory Input (SI) function and Emotional Signal (ES) generator—both need to be defined by the user. Since the amygdala is responsible for reinforcement and the orbitofrontal cortex for penalty, one needs to know that the amygdala will never unlearn the emotional response once learned; thus, the orbitofrontal cortex’s role is to inhibit the inappropriate response of the BELBIC controller. The BELBIC, here, is more a control paradigm than a typical controller with fixed structure, but even with this flexibility in the selection of SI and ES, it always operates on two inputs (SI and ES) and one model output (MO), defined as

$$MO = \sum_i A_i - \sum_i OC_i, \tag{18}$$

where i —number of sensory inputs.

From Equation (18), the difference between the provocative amygdala outputs (A_i) and the prohibitive orbitofrontal cortex outputs (OC_i) is calculated. The A_i and OC_i are defined as follows [48]:

$$A_i = V_i SI_i, \tag{19}$$

$$OC_i = W_i \times SI_i, \tag{20}$$

where V_i , W_i —weight parameters (the amygdala and orbitofrontal gains), and SI_i is the i -th sensory input. During control, the weights W_i , V_i are updated according to the following formulas:

$$\Delta V_i = \alpha \times SI_i \times \max\left(0, ES - \sum_i A_i\right), \tag{21}$$

$$\Delta W_i = \beta \times SI_i \times (MO - ES), \tag{22}$$

where α, β —learning coefficients for the amygdala and orbitofrontal cortex (usually between 0 and 1).

The last important equation in the BEBLIC model is the one for the A_{th} signal from the thalamus to the amygdala:

$$A_{th} = V_{th} \times \max(SI_i), \tag{23}$$

where V_{th} —weight parameter.

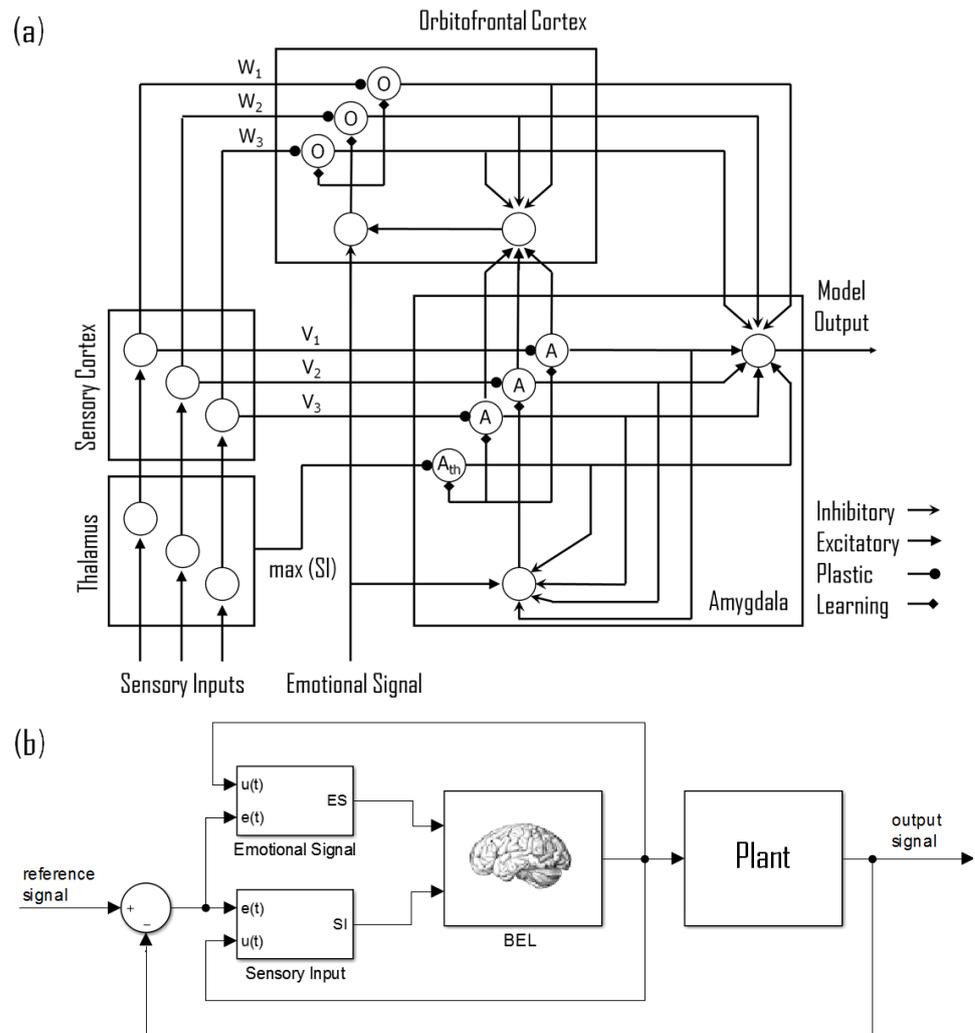


Figure 4. (a) BEL computational model, (b) SISO closed-loop control system with BELBIC controller.

3. Minimum Energy Control of Quadrotor UAV

3.1. Optimization Problem

By introducing the structure of the BELBIC controller from Figure 4 into the autonomous drone control system of Figure 2, the aim is to obtain: (a) stabilization of the system from Figure 5 during UAV flight and (b) to provide minimum energy control. Since, the BELBIC controller is considered as ‘an action selection methodology’ [49], the ES and SI signals need to be properly chosen by the designer. In general, these are functions defined as:

$$ES = \mathcal{G}(e, u, r, y), \tag{24}$$

$$SI = \mathcal{F}(e, u, r, y), \tag{25}$$

where e —control error, u —control signal, r —reference signal, and y —output signal.

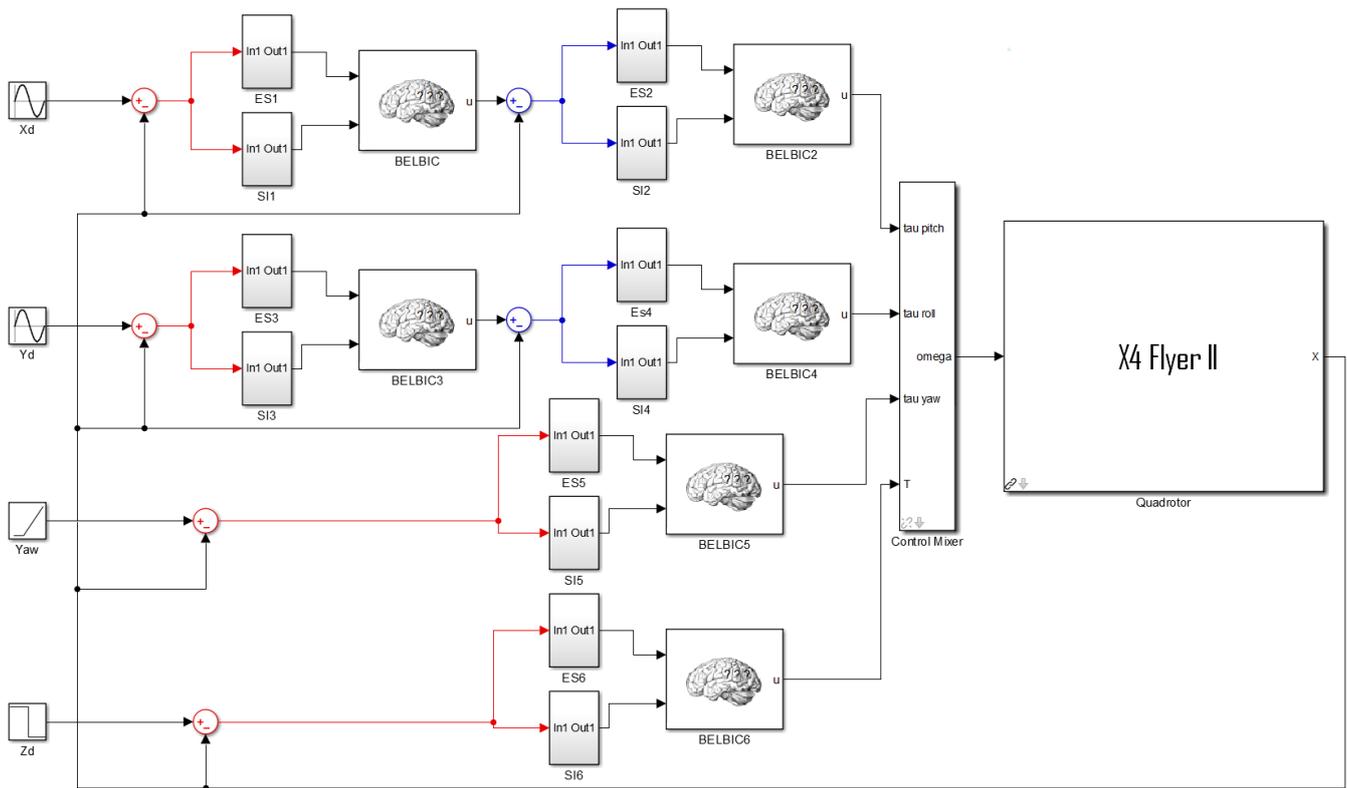


Figure 5. MATLAB-based block diagram of the UAV autonomous control system with BELBIC controllers (inputs: UAV's desired position in x_d , y_d , and z_d axis and desired yaw angle ψ_d ; while output: UAV state vector of current position and orientation).

Since UAV autonomous control systems use at most six separate controllers (usually PD or PID type), it is proposed to use inner and outer control loops (see Figure 5), following functions ES_l and SI_l for $l = \{1, \dots, 6\}$ in their BELBIC counterparts

$$ES_l = k_{l1}e_l(t) + k_{l2} \int_0^{t_h} e_l(t)dt + k_{l3} \frac{d}{dt}e_l(t), \quad (26)$$

$$SI_l = k_{l4}e_l(t) + k_{l5} \frac{d}{dt}e_l(t), \quad (27)$$

where t_h —time horizon of control, and k_{l1}, \dots, k_{l5} , are positive gains of ES_l and SI_l functions of l -th PID- and PD-type controllers based on $e_l(t)$ tracking error.

Based on its high flexibility, the PID-type controller was chosen for ES_i , and since UAV as an unstable plant is often approximated by a linear double-integrating term that can be counteracted by the derivative term of simple PD-type controller, this structure was proposed for SI_i . Furthermore, according to the control theory, in the autonomous control system of the UAV, in Formulas (26) and (27), gains k_{l1} and k_{l4} adjust the UAV settling time, gains k_{l3} and k_{l5} reduce the overshoot, and k_{l2} determines the steady-state error, respectively.

The ES function reflects information about the deterioration of the control quality, i.e., when l -th control error increases, it mimics a negative emotion in BEL. Thus, the l -th BELBIC controller will work via the l -th SI according to the Algorithm 1 to provide proper control signal for the UAV. To force the BELBIC to work more energy efficiently and precisely, optimization mechanisms can be easily adapted. Accordingly, the optimization task for minimum energy control of the UAV is formulated as follows:

- Goal: Ensure the smallest tracking errors during the UAV flight:

$$e_p = p_d - p, \quad (28)$$

$$e_\psi = \psi_d - \psi, \quad (29)$$

at lowest possible energy effort.

- Cost function (performance index) $J(t)$:

$$J(t) = \int_0^{t_h} (\Gamma|e(t)| + \Psi|u(t)|)dt, \quad (30)$$

where Γ and Ψ —weight coefficients for the control error and the control signal of a particular controller, respectively.

- Optimization problem formalism:

$$\begin{aligned} \min_{k_1, k_2, \dots, k_N} \quad & J(t) = \int_0^{t_h} (\Gamma|e(t)| + \Psi|u(t)|)dt, \\ \text{s.t.} \quad & 0 \leq k_1 \leq k_1^{max} \\ & 0 \leq k_2 \leq k_2^{max} \\ & \dots \\ & 0 \leq k_N \leq k_N^{max} \end{aligned} \quad (31)$$

where $k_1^{max}, k_2^{max}, \dots$, and k_N^{max} are predefined by designer upper bounds of ranges where the optimizer explores the search space for optimal gains of N controller parameters.

Gains k_1 and k_5 for each of the BELBIC controllers may be found using bio-inspired optimization algorithms.

Remark 1. Γ and Ψ are used to profile the UAV output signals in a meaning of energy efficient flights, that is, penalizing by using larger values of Ψ will cause more smooth flight characteristics, avoiding large control signal and aggressive controller work, and thus the flight time will be extended.

Algorithm 1 The BELBIC-inspired algorithm for UAV control

- 1: **Variables initialization** Set: $V_i = 0, V_{th} = 0, W = 0$, for $i = 0, \dots, 6$
 - 2: **Define** $ES_i = \text{cost function}$, for $i = 0, \dots, 6$
 - 3: **for** Each iteration $t = t_s$ **do**
 - 4: **for** Each control inputs l **do**
 - 5: Compute $ES_l = k_{l1}e_l(t) + k_{l2} \int_0^{t_h} e_l(t)dt + k_{l3} \frac{d}{dt} e_l(t)$
 - 6: Compute $SI_l = k_{l4}e_l(t) + k_{l5} \frac{d}{dt} e_l(t)$
 - 7: Compute $A_l = V_l SI_l$
 - 8: Compute $OC_l = W_l SI_l$
 - 9: Compute $A_{th} = V_{th} \max(SI_l)$
 - 10: Compute MO_l
 - 11: Update V_l
 - 12: Update W_l
 - 13: Update V_{th}
 - 14: **end for**
 - 15: **end for**
-

3.2. Bio-Inspired Optimization Algorithms

In optimization tasks, where an approach based on a control system model is possible, nature- and bio-inspired algorithms have been used successfully for years [20]. By means of numerical calculations, they allow the J index to be calculated for a large number of combinations of controller gains. On the basis of the state of the art, it was decided to use two: the “classical”, well-known particle swarm optimization (PSO) and the “rising”

cuttlefish algorithm (CFA). In this paper, only the mathematical formulas for both are presented. For more details, see [50,51].

3.2.1. Particle Swarm Optimization

The PSO algorithm mimics the behavior of a group of animals that live in flocks and communicate with each other, e.g., to find the best food supplies. Each particle in the optimization method is treated as a set of controller gains. The PSO algorithm starts with an initial set of particles and, by the movement of these particles, explores the constrained search space of size m . The movement of each i particle in the subsequent iteration of the PSO algorithm is determined by its individual and social behavior. The velocity of the particle $v_i = [v_{i1}, v_{i2}, \dots, v_{im}]^T$ and its position $x_i = [x_{i1}, x_{i2}, \dots, x_{im}]^T$ in the t iteration can be updated according to the following equations

$$v_i(t+1) = v_i(t) + \phi_1 c_i(t) + \phi_2 s_i(t), \quad (32)$$

$$x_i(t+1) = x_i(t) + v_i(t+1), \quad (33)$$

where ϕ_1 and ϕ_2 are cognition and social constants to explore the search space. They are usually chosen as values from 0 to 2 to establish the proper balance between cognitive (c_i) and social (s_i) oriented exploration. Vectors $c_i = [c_{i1}, c_{i2}, \dots, c_{im}]^T$ and $s_i = [s_{i1}, s_{i2}, \dots, s_{im}]^T$ are defined as

$$c_i(t) = p_i(t) - x_i(t), \quad (34)$$

$$s_i(t) = g_i(t) - x_i(t), \quad (35)$$

where the vector $p_i = [p_{i1}, p_{i2}, \dots, p_{im}]^T$ is the best position obtained from the particle i until the current iteration t , and the vector $g_i = [g_{i1}, g_{i2}, \dots, g_{im}]^T$ is the best position of all particles in iteration t .

In 2001, Eberhart and Kennedy proposed in [50] the modification of their PSO algorithm (32)–(35) by introducing an additional inertia weight λ for a better convergence of the algorithm to the optimum; thus, in the equation:

$$v_i(t+1) = \lambda(t)v_i(t) + \phi_1 c_i(t) + \phi_2 s_i(t) \quad (36)$$

higher values of λ provide more social (global) exploration and smaller, more cognitive (local) exploration in the available search space, respectively.

3.2.2. Cuttlefish Algorithm

Cuttlefish, in danger, can very quickly change color to be as invisible as possible in the water environment or, contrarily, become stunningly visible. This behavior is mimicked in the cuttlefish optimization algorithm, where the color change mechanism (based on the *reflection* and *visibility* processes) is used to solve optimization tasks. In nature, all the colors and patterns on the skin of cuttlefish come from reflected light from different layers of cells, which are stacked together. These mirror-like cells are chromatophores, iridophores, and leucophores. In the first effect, *reflection*, light can be reflected from cells in six combinations. In the second effect, which is *visibility*, the cuttlefish try to mimic the patterns of their water environment. In CFA, that is the difference between the best and current solutions of the optimization task. Using the effects of *reflection* and *visibility*, as well as the division of cells into four groups, the CFA algorithm (Figure 6) explores the search space of cells. Groups no. 1 and 4 are used for the local search, while no. 2 and 3 are used for the global search. All groups share the best solution and work independently. In the case considered in the article, each cell represents a particular combination of controller gains, and a new solution (*newP*) is calculated in every iteration t of the CFA algorithm according to the following equation

$$newP = reflection + visibility. \quad (37)$$

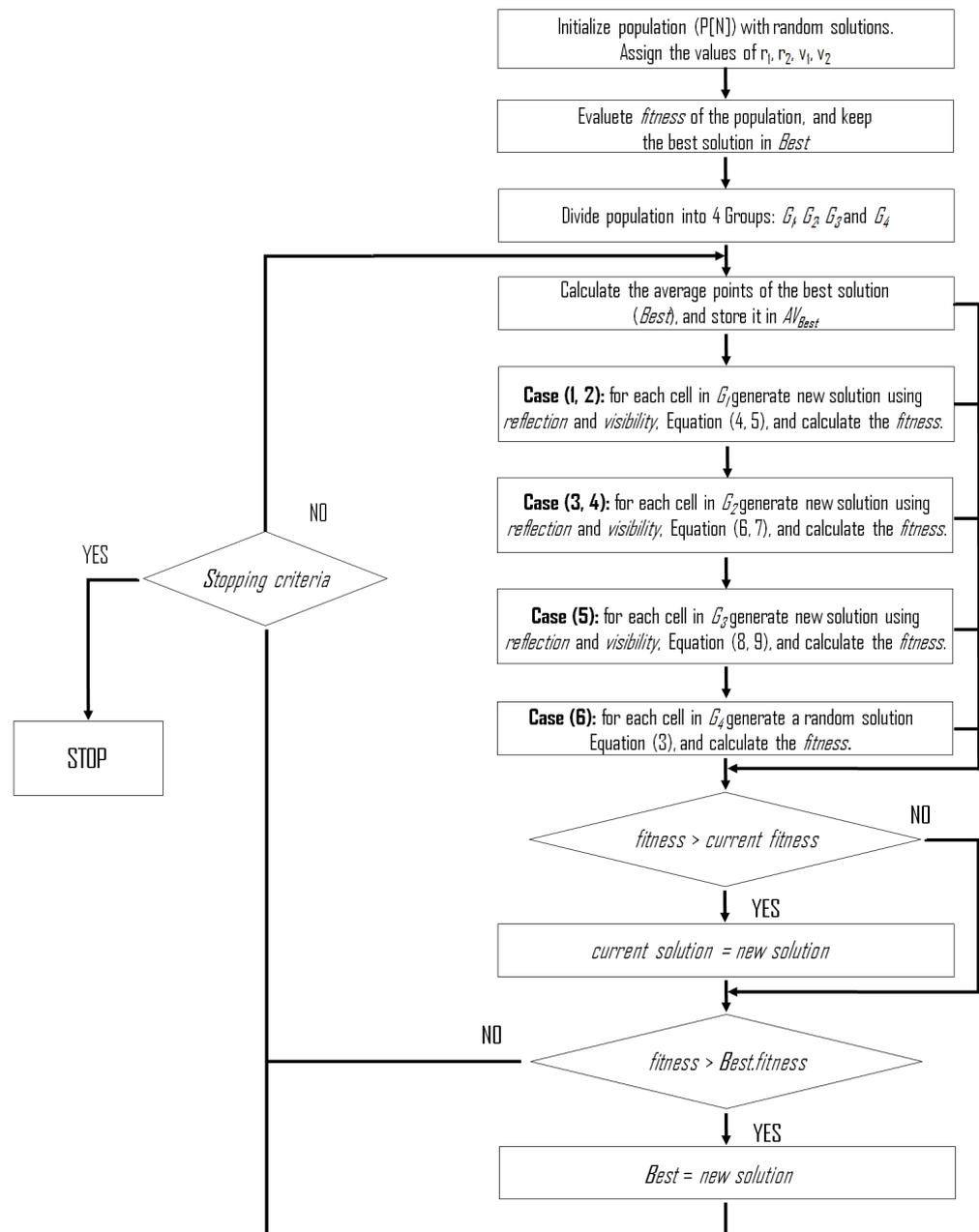


Figure 6. The CFA algorithm [51].

To start the CFA algorithm (Figure 6), a population P (cells) of N initial solutions $P = cells = \{points_1, points_2, \dots, points_N\}$ is spread over d size search space at random positions ($points$) using:

$$P[i].points[j] = random \cdot (upperLimit - lowerLimit) + lowerLimit, \quad (38)$$

$$i = 1, 2, \dots, N; j = 1, 2, \dots, d$$

where $upperLimit$ and $lowerLimit$ are the upper and lower limits in the problem domain, since $random$ is a value between 0 and 1.

In the CFA algorithm, a single cell in the population is represented by $points_i$. It is also associated with two values: $fitness$ and a vector of continuous values of dimension d . $Best$ keeps the best solution, while $AVBest$ stores the calculated average of $Points$. These metrics are used in calculations performed in each of six cases for four cell groups:

- Cases no. 1 and 2 for Group G1:

$$Reflection[j] = R \cdot G_1[i].Points[j] \quad (39)$$

$$Visibility[j] = V \cdot (Best.Points[j] - G_1[i].Points[j]) \quad (40)$$

- Cases no. 3 and 4 for Group G2:

$$Reflection[j] = R \cdot Best.Points[j] \quad (41)$$

$$Visibility[j] = V \cdot (Best.Points[j] - G_2[i].Points[j]) \quad (42)$$

- Case 5 for Group G3:

$$Reflection[j] = R \cdot Best.Points[j] \quad (43)$$

$$Visibility[j] = V \cdot (Best.Points[j] - AV_{Best}) \quad (44)$$

- Case 6 for Group G4—Equation (38), where i — i -th cell of group G1, $Points[j]$ — j -th point of i -th cell, R —degree of reflection, and V —degree of visibility.

The values of R and V are calculated according to the following equations

$$R = random \cdot (r_1 - r_2) + r_2, \quad (45)$$

$$V = random \cdot (v_1 - v_2) + v_2, \quad (46)$$

where $random()$ —function to generate random numbers between (0, 1), while r_1 , r_2 , v_1 , and v_2 —constant values that determine the stretch interval of the chromatophores cells and the visibility degree interval of the final view of the pattern, respectively.

4. Simulation Tests

4.1. Simulation Environment

For the performance analysis of the proposed control system, due to the number of repetitions necessary to determine the best gains of particular BELBIC controllers, the possibilities of computer simulation were used. For this purpose, a drone model widely recognized and validated by the UAV community was selected, that is, the X4-flyer II proposed by Paul Pounds et al. in [52]. This quadrotor UAV was built at the Australian National University. Its important construction and dynamical parameters are summarized in Table 2. The X4-flyer II model, as well as BELBIC controllers, were implemented with the use of open source software, i.e., *Robotics Toolbox* created by Peter Corke et al. [53] for MATLAB/Simulink. In this environment, elements of *Brain Emotional Learning Toolbox* [54] were also implemented and optimization algorithms (PSO and CFA) were integrated.

Table 2. Parameters of the UAV quadrotor X4-flyer II (in SI units) [52,53].

Parameter	Symbol & Value
UAV mass	$m = 4$
Rotational inertia matrix	$J = \text{diag}([I_{xx} I_{yy} I_{zz}]), I_{xx} = I_{yy} = 0.082, I_{zz} = 0.149$
Height of rotors above CoG	$h = 0.007$
Length of flyer arms	$d = 0.315$
Number of blades per rotor	$n = 2$
Rotor radius	$r = 0.165$
Blade chord	$c = 0.018$
Flapping hinge offset	$e = 0.0$
Rotor blade mass	$M_b = 0.005$
Estimated hub clamp mass	$M_c = 0.010$
Blade root clamp displacement	$e_c = 0.004$
Non-dim. thrust coefficient	$C_t = 0.0048$
Lift slope gradient	$a = 5.5$

All simulation tests were carried out using a Dell Inspiron 3543 laptop, with an Intel Core i5-5200U CPU@2.2 GHz processor, with 8 GB RAM memory under the 64-bit Windows 8.1 operating system. For the calculations, the MATLAB/Simulink 2016a was used.

The selected, most important, representative, and interesting research results regarding the tuning process of BELBIC controllers to provide minimum energy control are presented in the following subsections.

4.2. Experiment No. 1: Preliminary Adjustment

The autonomous control system of the UAV in Figure 2 is considered with the dynamical model of X4-flyer II and the parameters of Table 2. Nominal PD-type controllers from [53] are used to stabilize angles ϕ_d , θ_d , and ψ_d , while the movement of the drone in x , y , and z are controlled by the BELBIC type. In the preliminary stage of research, the trial-and-error tuning approach is usually the first choice. This approach allows one to acquire ‘expert knowledge’ and to know the useful gain ranges. One needs to remember that in the case of each BELBIC controller (Equations (26) and (27)), there are ten different parameters that need to be tuned, i.e., K_1 – K_5 , α , β , V_i , W_i , and V_{th} , and this is not a trivial task.

The results of an interesting example of BELBIC position controllers’ pre-tuning are shown in Figure 7 and on the AeroLab website (animated, recorded flight trajectories from the simulation tests discussed in the article are available at <http://www.uav.put.poznan.pl> and <https://youtu.be/iVDeJbMYIQQ>, accessed on 20 May 2022). They were obtained in the mission, where the UAV starts from the initial position $(-1, 0, 0)$, rises to 1.5 m, and flies to draw a square-shaped shape within the time horizon of 20 s. For the X and Y axes, the same controller gains were used due to the symmetric construction of the drone.

With respect to Figure 7, an interesting effect is visible. The drone accelerates rapidly, but there are overshoots and it takes a long time to obtain the expected precision around the control waypoints. It can be seen that BELBIC-type position controllers are over-reactive and force a change in the altitude of the drone instead of its tilt only when moving forward and sideways.

4.3. Experiment No. 2: PSO-Based Gains Selection vs. Path-Tracking Precision (Altitude Controller)

In experiment no. 2, toward minimum energy control in the problem of proper gains selection, research was conducted to give the answer to the question: *How do the gains of the designed BELBIC controllers determine their performance in reference path tracking?*

It was decided to implement the following modifications regarding the configuration of the setup from Experiment No. 1:

- Only the BELBIC-type altitude controller is analyzed, the rest are native PD-type controllers with the gains from [53].
- In the reference trajectory, a circle was introduced in place of the square shape (avoiding sudden moments of position switching at this stage of gains selection).
- During flight, the UAV aims to rotate simultaneously around the Z -axis of $\{\mathcal{E}\mathcal{F}\}$.
- The integral of the absolute error (IAE) was introduced as a measure of flight performance assessment:

$$IAE = \int_0^{t_h} |e(t)| dt, \quad (47)$$

- The PSO algorithm was used to search for the optimal gains of the BELBIC controller according to Table 3.

Table 3. Parameters for tuning the BELBIC-type altitude controller using the PSO algorithm.

	K_1	K_2	K_3	K_4	K_5	α	β	V	V_{th}	A
<i>min</i>	0.01	0.01	0.01	0.01	0.01	0.01	0.01	1	1	0.001
<i>max</i>	700	700	700	100	100	0.1	0.1	const	const	const
<i>best</i>	699.99	0.01	256.62	32.31	11.28	0.09	0.01	const	const	const

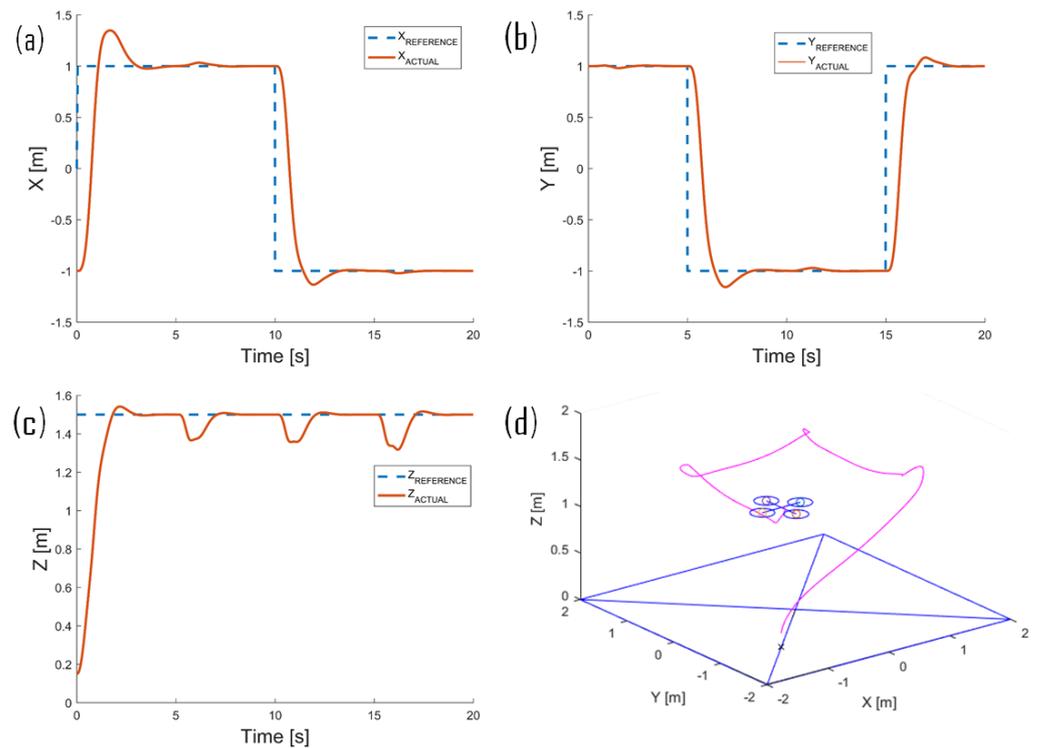


Figure 7. Experiment No. 1: An exemplary test of the effectiveness of square-shaped path tracking for the X-4 Flyer II drone model in a system with BELBIC controllers tuned by trial and error: (a–c) reference (desired) and actual (measured) positions of the UAV on the X, Y, and Z axes, (d) flight trajectory in 3D.

In the search for optimal gains, a swarm consisting of 20 particles was used. The values of K_1 , K_2 , and K_3 were changed with a step equal to 50, and the PSO algorithm was repeated three times in each case to average the IAE value obtained for the best result, as shown in Figure 8, where the function $IAE = f(K_i)$ (for $i = 1, \dots, 3$) is presented.

In experiment No. 2, the PSO algorithm was initialized 45 times, and 289,660 combinations of the BELBIC altitude controller gains were checked. As shown in Figure 8, good performance close to optimal solution is obtained by the limits of K_i equal to 400. Since in BELBIC we are based on the PID structure in ES, higher actuation (through K_1 and K_3) is favored by a better response of the controller (see Figure 9), and theoretically better precision can be achieved by proper selection of other parameters. However, it should be remembered that in the closed-loop autonomous control system of the UAV, the control signal of the BELBIC controller is saturated to protect the propulsion units, and a high value of the control signal will simply be saturated.

For further synthesis and performance analysis of the BELBIC-based energy-saving control, a maximum gains limit of 400 was selected. For this limit, the results obtained with an exemplary combination of gains from Table 4 are shown in Figure 10, where the altitude controller works dynamically and generates just minimal overshoot, which in the considered case is still desirable, since in subsequent experiments additional restrictions in the form of a penalty function will be imposed on the control signal, due to which it will be possible to slow down the controller's emotional response (and thus reject the overshoot) to ensure the expected tracking precision.

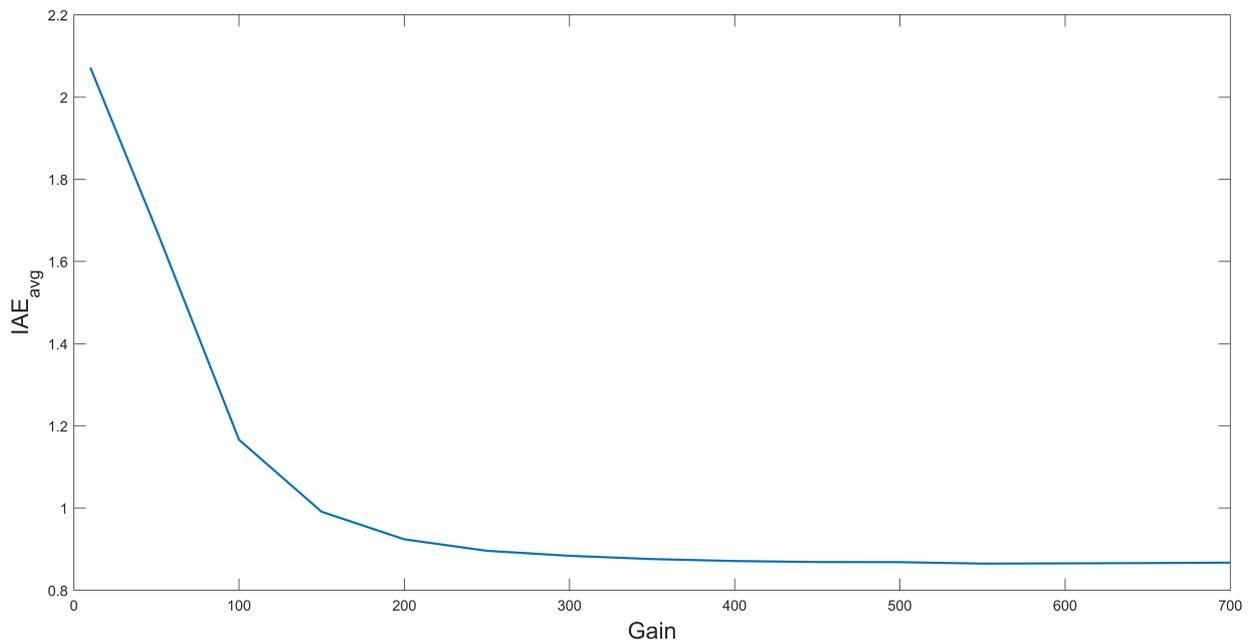


Figure 8. Experiment No. 2: Function $IAE = f(K_i)$ (for $i = 1, \dots, 3$).

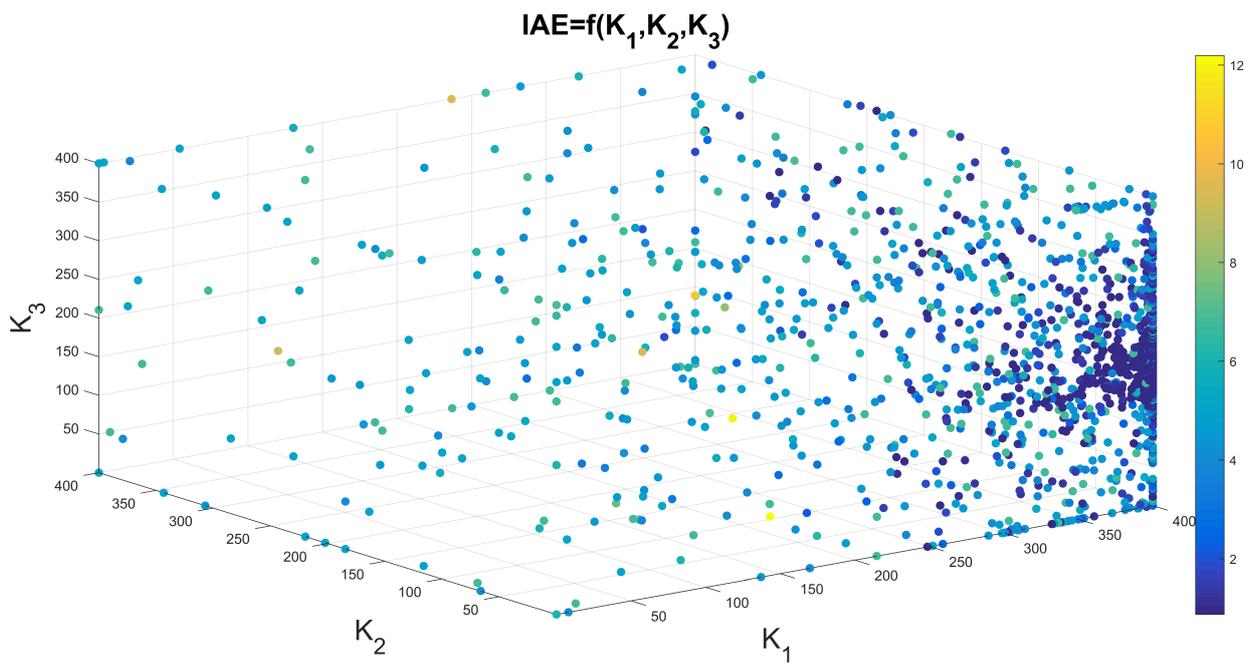


Figure 9. Experiment No. 2: Function $IAE = f(K_1, K_2, K_3)$ for gains limited to the value of 400 and $\alpha = 0.1, \beta = 0.01$.

Table 4. Parameters of the BELBIC altitude controller used in Experiment No. 2.

	K_1	K_2	K_3	K_4	K_5	α	β	V	V_{th}	A
value	399.9993	182.3310	31.4309	11.3261	0.0999	0.01	0.01	1	1	0.001

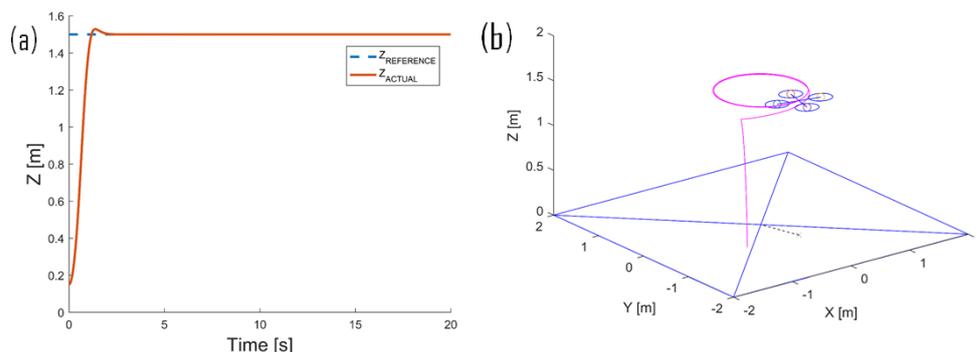


Figure 10. Experiment No. 2: Test of the performance of circle-shaped path tracking for the X-4 Flyer II drone model in a system with BELBIC altitude controller: (a) reference (desired) and actual (measured) positions of the UAV in the Z axes and (b) flight trajectory in 3D.

4.4. Experiment No. 3: Gain Tuning for Minimum Energy Control of the UAV (Altitude Controller)

In the next experiment considered, attention was focused on the first 5 s of the UAV flight, where how savings in generating the control signal affect the quality of the reference path tracking was analyzed. In the optimization process, the smallest value of the cost function of the Equation (30) was searched by increasing the value Ψ (see Table 5), which is a penalty for too large control signals. To be able to compare the results of the experiments, the integral of absolute error (IAE) is analyzed as a measure of control precision and the integral of absolute of the control signal (IAU) as the equivalent of the energy expenditure in this (altitude control) part of the drone control system. Analyzing the results shown in Figure 11, it can be seen that, depending on the expectations expressed by the value of Ψ , using the optimization algorithm, one can search for the gains of the altitude controller that will provide a slower flight profile (with a smoother shape), which is desirable, for example, during video recordings with the use of a drone. When comparing the results for $\Psi = 0$ and $\Psi = 0.005$, there is a difference in the output signal response by 0.2 s (IAE deteriorated by 14.19%), and energy expenditure is reduced by 32.16%. In the case of the highest difference in the value of Ψ , the obtained value of IAU is double. It seems intuitive to introduce a mechanism that allows controllers to modify/switch gains depending on the needs or type of particular drone mission (agile maneuvers, cargo, filming, etc.).

Table 5. Results of the tuning of the BELBIC-type altitude controller using the PSO algorithm ($\Psi = var$, $\Gamma = 1$).

	K_1	K_2	K_3	K_4	K_5	α	β	IAE	IAU
<i>min</i>	0.01	0.01	0.01	0.01	0.01	0.01	0.001	—	—
<i>max</i>	400	5	200	100	50	1.0	0.01	—	—
$\Psi = 0.000$	399.99	0.06	156.55	91.76	30.15	0.998	0.001	0.895	167.0
$\Psi = 0.001$	298.49	0.01	159.35	47.79	19.08	0.038	0.001	0.901	164.0
$\Psi = 0.002$	264.44	0.01	185.18	46.65	20.19	0.023	0.001	0.937	138.0
$\Psi = 0.003$	259.92	0.01	199.99	44.99	20.46	0.020	0.001	0.959	128.6
$\Psi = 0.004$	209.15	0.02	162.11	46.42	22.65	0.046	0.003	1.003	126.8
$\Psi = 0.005$	215.31	0.01	199.99	46.40	24.08	0.012	0.001	1.022	113.3
$\Psi = 0.006$	197.87	0.01	199.99	44.99	25.74	0.010	0.002	1.061	105.6
$\Psi = 0.007$	263.42	0.01	112.22	41.01	28.31	0.999	0.001	1.083	108.1
$\Psi = 0.008$	121.44	4.70	86.730	85.22	38.19	0.984	0.010	1.138	101.7
$\Psi = 0.009$	157.10	0.02	187.15	33.15	25.29	0.010	0.004	1.251	89.93

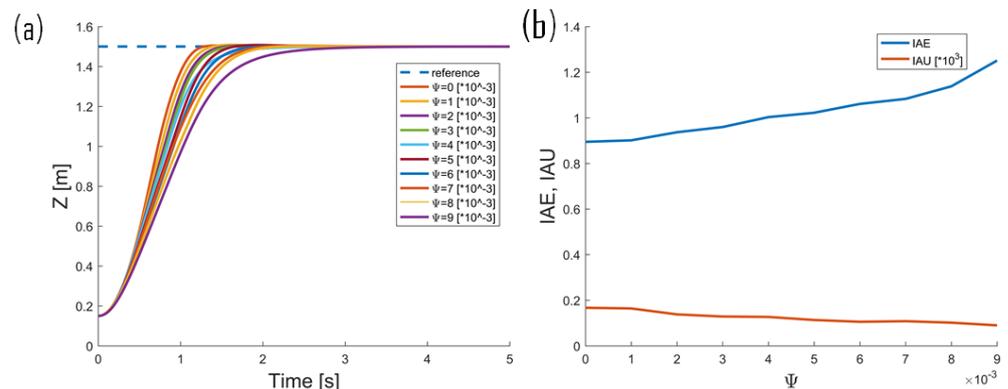


Figure 11. Experiment No. 3: (a) $Z = f(t)$ [m] of the X-4 Flyer II drone model in a system with BELBIC altitude controller in 5 s flight time horizon for $\Psi = var$, and $\Gamma = 1$, (b) IAE and IAU values.

4.5. Experiment No. 4: Performance Evaluation of Position Controllers

A similar experiment was conducted as before for the square-shaped flight profile with position controllers in a time horizon of 10 s. The most important obtained results are summarized in Table 6. It can be noticed that the use of the CFA algorithm to optimize the PID controller gains improved the flight performance in the Z-axis, which in turn had a positive effect on the precision of the drone positioning in the X and Y axes, with a slightly worse tracking of ψ changes. The last interesting case is the fourth one in the Table 6, where the minimum energy control of the yaw angle was obtained with the imposed penalty for the too large control signal of the BELBIC-type controller. A slight slowing down of the rotation angle changes interferes with the results for the X and Y axes. The slower turning of the aircraft minimally deteriorates the tracking (higher IAE value recorded).

Table 6. Results obtained for the square-shaped flight profile with different position controllers in a time horizon of 10 s.

Ψ Value	Z-Axis	ψ -Axis	X, Y Axes	IAE (Z)	IAE (ψ)	IAE (X,Y)
0	PD	PD	PD and PD	2.608	1.642	9.170
0	PID (PSO)	PD	PD and PD (PSO)	1.222	1.801	3.751
0	PID (CFA)	PD	PD and PD (PSO)	1.091	1.838	3.519
3×10^{-3}	PID (CFA)	BELBIC (PSO)	PD and PD (PSO)	1.111	1.810	4.032

5. Conclusions

The minimum energy fine-tuning control methodology is proposed for the predefined quadrotor UAV path-tracking task. The autonomous cascade control system with the nonlinear six DoF mathematical model of the X4-flyer II drone and neurobiologically inspired intelligent controller is used to find the best possible gains that will provide a good tracking quality with the lowest possible control signal effort. The synthesis of the control system is presented for the chosen BELBIC-type controller structures, bio-inspired optimizers, cost functions, and gains ranges (on the basis of expert knowledge). The performance analysis of the proposed control method is validated on two simple flight missions. By means of numerical experiments, new knowledge is provided, i.e., how long for altitude and X, Y axes control tasks; it is possible to extend the flight time of an unmanned aircraft (while increasing the precision of tracking) using brain emotional learning based intelligent controllers in the proposed shape. path tracking task

The results presented from five selected experiments illustrate the potential of the proposed fine-tuning BELBIC-based control methodology to be applied to very demanding hardware applications with limited energy sources, such as the one that is the next target for real-world scenario controller applications in our fault-tolerant Falcon V5 drone, which is a coaxial X8 quadrotor (details in [55]). It is also planned to verify the performance of the

solution on our flapping-wing microdrones. Furthermore, in parallel, comparative studies of the tracking quality of the proposed solution are conducted with the optimal regulators based on the linear quadratic regulator (LQR), coefficient diagram method (CDM), dynamic pole motion (DPM) approach [56], and State-Dependent Riccati Equation (SDRE) technique.

Funding: This research was funded by the National Science Centre—NCN (Poland), under the program MINIATURA 4, grant number: 2020/04/X/ST7/00357.

Data Availability Statement: The numerical results gathered and analyzed during the current study are available in: <https://chmura.put.poznan.pl/s/8hzdGFFDfOnMXq>, accessed on 20 May 2022. The data are shared after contact with the author (password needed).

Acknowledgments: The author would like to thank Youmin Zhang for his support during the research and work on the article.

Conflicts of Interest: The author declares no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

BEL	Brain Emotional Learning
BELBIC	Brain Emotional Learning Based Intelligent Controller
CDM	Coefficient Diagram Method
CFA	Cuttlefish Algorithm
CoG	Center of Gravity
ENN	Emotional Neural Network
IAE	Integral of the Absolute Error
LQR	Linear Quadratic Regulator
ES	Emotional Signal
MO	Model Output
MPC	Model Predictive Control
MR	Magneto-Rheological (Damper)
OFC	Orbitofrontal Cortex
PD	Proportional–Derivative (Controller)
PID	Proportional–Integral–Derivative (Controller)
PSO	Particle Swarm Optimization
SI	Sensory Input
SISO	Single-Input Single-Output
SMC	Sliding Mode Control
SNN	Sensory Neural Network
SDRE	State-Dependent Riccati Equation
UAV	Unmanned Aerial Vehicle

References

- Allaire, F.J.; Labonté, G.; Tarbouchi, M.; Roberge, V. Recent advances in unmanned aerial vehicles real-time trajectory planning. *J. Unmanned Veh. Syst.* **2019**, *7*, 259–295. [\[CrossRef\]](#)
- Dong, Y.; Tao, J.; Zhang, Y.; Lin, W.; Ai, J. Deep Learning in Aircraft Design, Dynamics, and Control: Review and Prospects. *IEEE Trans. Aerosp. Electron. Syst.* **2021**, *57*, 2346–2368. [\[CrossRef\]](#)
- Amin, R.; Aijun, L.; Shamshirband, S. A review of quadrotor UAV: Control methodologies and performance evaluation. *J. Unmanned Veh. Syst.* **2016**, *10*, 87–103. [\[CrossRef\]](#)
- Kangunde, V.; Jamisola, R.S., Jr.; Theophilus, E.K. A review on drones controlled in real-time. *Int. J. Dyn. Control* **2021**, *9*, 1832–1846. [\[CrossRef\]](#) [\[PubMed\]](#)
- Michailidis, M.G.; Rutherford, M.J.; Valavanis, K.P. A Survey of Controller Designs for New Generation UAVs: The Challenge of Uncertain Aerodynamic Parameters. *Int. J. Control Autom. Syst.* **2020**, *18*, 801–816. [\[CrossRef\]](#)
- Nascimento, T.; Saska, M. Fast nonlinear model predictive control for very-small aerial vehicles. In Proceedings of the 2020 International Conference on Unmanned Aircraft Systems (ICUAS), Athens, Greece, 1–4 September 2020; pp. 523–528. [\[CrossRef\]](#)
- Erginer, B.; Altuğ, E. Design and implementation of a hybrid fuzzy logic controller for a quadrotor VTOL vehicle. *Int. J. Control Autom. Syst.* **2012**, *10*, 61–70. [\[CrossRef\]](#)

8. Besnard, L.; Shtessel, Y.B.; Landrum, B. Quadrotor vehicle control via sliding mode controller driven by sliding mode disturbance observer. *J. Frankl. Inst.* **2012**, *349*, 658–684. [[CrossRef](#)]
9. Wang, B.; Yu, X.; Mu, L.; Zhang, Y. Disturbance observer-based adaptive fault-tolerant control for a quadrotor helicopter subject to parametric uncertainties and external disturbances. *Mech. Syst. Signal Process.* **2019**, *120*, 727–743. [[CrossRef](#)]
10. Zhao, Y.; Zhang, H.; Chen, Z.; Wang, H.; Zhao, X. Adaptive neural decentralised control for switched interconnected nonlinear systems with backlash-like hysteresis and output constraints. *Int. J. Syst. Sci.* **2022**, *53*, 1545–1561. [[CrossRef](#)]
11. Li, Y.; Niu, B.; Zong, G.; Zhao, J.; Zhao, X. Command filter-based adaptive neural finite-time control for stochastic nonlinear systems with time-varying full-state constraints and asymmetric input saturation. *Int. J. Syst. Sci.* **2022**, *53*, 199–221. [[CrossRef](#)]
12. Liu, S.; Zhang, L.; Niu, B.; Zhao, X.; Ahmad, A. Adaptive neural finite-time hierarchical sliding mode control of uncertain under-actuated switched nonlinear systems with backlash-like hysteresis. *Inf. Sci.* **2022**, *599*, 147–169. [[CrossRef](#)]
13. Chang, X.; Liu, L.; Shen, M. Resilient Control Design for Lateral Motion Regulation of Intelligent Vehicle. *IEEE/ASME Trans. Mechatron.* **2019**, *24*, 2488–2497. [[CrossRef](#)]
14. Kim, J.; Gadsden, S.A.; Wilkerson, S.A. A Comprehensive Survey of Control Strategies for Autonomous Quadrotors. *J. Frankl. Inst.* **2020**, *43*, 3–16. [[CrossRef](#)]
15. Berkenkamp, F.; Schoellig, A.P.; Krause, A. Safe controller optimization for quadrotors with Gaussian processes. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 491–496. [[CrossRef](#)]
16. Chehadeh, M.S.; Boiko, I. Design of rules for in-flight non-parametric tuning of PID controllers for unmanned aerial vehicles. *J. Frankl. Inst.* **2019**, *356*, 474–491. [[CrossRef](#)]
17. Giernacki, W.; Horla, D.; Saska, M. In-flight Efficient Controller Auto-tuning using a Pair of UAVs. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 24 October–24 January 2020; pp. 1300–1307. [[CrossRef](#)]
18. Giernacki, W.; Horla, D.; Baca, T.; Saska, M. Real-Time Model-Free Minimum-Seeking Autotuning Method for Unmanned Aerial Vehicle Controllers Based on Fibonacci-Search Algorithm. *Sensors* **2019**, *19*, 312. [[CrossRef](#)] [[PubMed](#)]
19. Giernacki, W. Iterative Learning Method for In-Flight Auto-Tuning of UAV Controllers Based on Basic Sensory Information. *Appl. Sci.* **2019**, *9*, 648. [[CrossRef](#)]
20. Duan, H.; Li, P. *Bio-Inspired Computation in Unmanned Aerial Vehicles*; Springer: Berlin/Heidelberg, Germany, 2014. [[CrossRef](#)]
21. Giernacki, W. Cuttlefish Optimization Algorithm in Autotuning of Altitude Controller of Unmanned Aerial Vehicle (UAV). In Proceedings of the ROBOT 2017: Third Iberian Robotics Conference, Seville, Spain, 22–24 November 2017; pp. 841–852. [[CrossRef](#)]
22. Heidari, A.; Mirjalili, S.; Faris, H.; Aljarah, I.; Mafarja, M.; Chen, H. Harris hawks optimization: Algorithm and applications. *Future Gener. Comput. Syst.* **2019**, *97*, 849–872. [[CrossRef](#)]
23. Zervoudakis, K.; Tsafarakis, S. A mayfly optimization algorithm. *Comput. Ind. Eng.* **2020**, *145*, 1–23. [[CrossRef](#)]
24. Chou, J.; Truong, D. A novel metaheuristic optimizer inspired by behavior of jellyfish in ocean. *Appl. Math. Comput.* **2021**, *389*, 125535. [[CrossRef](#)]
25. Mohammadi-Balani, A.; Nayeri, M.; Azar, A.; Taghizadeh-Yazdi, M. Golden eagle optimizer: A nature-inspired metaheuristic algorithm. *Comput. Ind. Eng.* **2021**, *152*, 107050. [[CrossRef](#)]
26. Noel, M.; Muthiah-Nakarajan, V.; Amali, G.; Trivedi, A. A new biologically inspired global optimization algorithm based on firebug reproductive swarming behaviour. *Expert Syst. Appl.* **2021**, *183*, 115408. [[CrossRef](#)]
27. Zietkiewicz, J.; Koziarski, P.; Giernacki, W. Particle swarm optimisation in nonlinear model predictive control; comprehensive simulation study for two selected problems. *Intell. Autom. Soft Comput.* **2020**, *97*, 2623–2639. [[CrossRef](#)]
28. Picard, R. *Affective Computing*; MIT Press: Cambridge, MA, USA, 2000.
29. Moren, J.; Balkenius, C. A Computational Model of Emotional Learning in the Amygdala. In Proceedings of the 6th International Conference on the Simulation of Adaptive Behavior, Paris, France, 11–17 September 2000; MIT Press: Cambridge, MA, USA, 2000; pp. 1–9.
30. Goleman, D. *The Brain and Emotional Intelligence: New Insights*; More Than Sound: Florence, MA, USA, 2011.
31. Lofti, E.; Akbarzadeh, M. Adaptive brain emotional decayed learning for online prediction of geomagnetic activity indices. *Neurocomputing* **2014**, *126*, 188–196. [[CrossRef](#)]
32. Lucas, C.; Shahmirzadi, D.; Sheikholeslami, N. Introducing BELBIC: Brain emotional learning based intelligent control. *Intell. Autom. Soft Comput.* **2004**, *10*, 11–22. [[CrossRef](#)]
33. Lin, C.M.; Chung, C.C. Fuzzy Brain Emotional Learning Control System Design for Nonlinear Systems. *Int. J. Fuzzy Syst.* **2015**, *17*, 117–128. [[CrossRef](#)]
34. Sadeghieh, A.; Roshanian, J.; Najafi, F. Implementation of an Intelligent Adaptive Controller for an Electrohydraulic Servo System Based on a Brain Mechanism of Emotional Learning. *Int. J. Adv. Robot. Syst.* **2012**, *9*, 1–12. [[CrossRef](#)]
35. Cesar, M.; Coelho, J.; Goncalves, J. Evolutionary-Based BEL Controller Applied to a Magneto-Rheological Structural System. *Actuators* **2018**, *7*, 29. [[CrossRef](#)]
36. Cesar, M.; Coelho, J.; Goncalves, J. Semi-Active Vibration Control of a Non-Collocated Civil Structure using Evolutionary-Based BELBIC. *Actuators* **2019**, *8*, 43. [[CrossRef](#)]

37. Rouhani, H.; Jalili, M.; Araabi, B.; Eppler, W.; Lucas, C. Brain emotional learning based intelligent controller applied to neurofuzzy model of micro-heat exchanger. *Expert Syst. Appl.* **2007**, *32*, 911–918. [[CrossRef](#)]
38. Jamali, M.; Dehyadegari, M.; Arami, A.; Lucas, C.; Navabi, Z. Real-time embedded emotional controller. *Neural Comput. Appl.* **2010**, *19*, 13–19. [[CrossRef](#)]
39. Arab Markadeh, G.; Daryabeigi, E.; Lucas, C.; Rahman, M. Speed and Flux Control of Induction Motors Using Emotional Intelligent Controller. *IEEE Trans. Ind. Appl.* **2011**, *47*, 1126–1135. [[CrossRef](#)]
40. Nahian, S.; Truong, D.; Ahn, K. A self-tuning brain emotional learning-based intelligent controller for trajectory tracking of electrohydraulic actuator. *IEEE Trans. Ind. Appl.* **2014**, *228*, 461–475. [[CrossRef](#)]
41. Mokhtari, A.; Nikkhah, A.; Parvar, M.; Novinzadeh, A. Intelligent Auto pilot Design for a Nonlinear Model of an Autonomous Helicopter by Adaptive Emotional Approach. *J. Aerosp. Sci. Technol.* **2012**, *9*, 33–43.
42. Valencia, D.; Kim, D. Trajectory Tracking Control for Multiple Quadrotors Based on a Neurobiological-Inspired System. In Proceedings of the 2019 Third IEEE International Conference on Robotic Computing (IRC), Naples, Italy, 25–27 February 2019; pp. 465–470. [[CrossRef](#)]
43. Jafari, M.; Xu, H.; Garcia Carrillo, L. A biologically-inspired reinforcement learning based intelligent distributed flocking control for Multi-Agent Systems in presence of uncertain system and dynamic environment. *IFAC J. Syst. Control* **2020**, *13*, 100096. [[CrossRef](#)]
44. Jafari, M.; Xu, H. Biologically-Inspired Intelligent Flocking Control for Networked Multi-UAS with Uncertain Network Imperfectionst. *Drones* **2018**, *2*, 33. [[CrossRef](#)]
45. LeDoux, J. *The Amygdala: Neurobiological Aspects of Emotion, Memory, and Mental Dysfunction*; Wiley-Liss: New York, NY, USA, 1992; pp. 339–351.
46. Jafarzadeh, S.; Mirheidari, R.; Reza, M.; Motlagh, J.; Barkhordari, M. Intelligent Autopilot Control Design for a 2-DOF Helicopter Model. *Int. J. Comput. Commun. Control (IJCCC)* **2008**, *3*, 337–342.
47. Voos, H. Nonlinear control of a quadrotor micro-UAV using feedback-linearization. In Proceedings of the 2009 IEEE International Conference on Mechatronics, Malaga, Spain, 14–17 April 2009; pp. 1–6. [[CrossRef](#)]
48. Aghaee, S.; Lucas, C.; Zadeh, K. Applying Brain Emotional Learning Based Intelligent Controller (Belbic) to Multiple-Area Power Systems. *Asian J. Control* **2012**, *14*, 1580–1588. [[CrossRef](#)]
49. Jafari, M.; Xu, H.; Garcia Carrillo, L. A neurobiologically-inspired intelligent trajectory tracking control for unmanned aircraft systems with uncertain system dynamics and disturbance. *Trans. Inst. Meas. Control* **2019**, *41*, 417–432. [[CrossRef](#)]
50. Eberhard, R.; Shi, Y.; Kennedy, J. *Swarm Intelligence*; Academic Press: London, UK, 2001.
51. Eesa, A.; Brifcani, A.; Z., O. A New Tool for Global Optimization Problems—Cuttlefish Algorithm. *Int. J. Comput. Inf. Eng.* **2014**, *8*, 1235–1239.
52. Pounds, P.; Mahony, R.; Corke, P. Modelling and control of a large quadrotor robot. *Control Eng. Pract.* **2010**, *18*, 691–699. [[CrossRef](#)]
53. Corke, P. *Robotics, Vision and Control. Fundamental Algorithms in MATLAB. Second, Completely Revised, Extended and Updated Edition*; Springer International Publishing: Cham, Switzerland, 2017. [[CrossRef](#)]
54. Coelho, J.; Pinho, T.; Boaventura-Cunha, J.; de Oliveira, J. A new brain emotional learning Simulink® toolbox for control systems design. *IFAC-PapersOnLine* **2017**, *50*, 16009–16014. [[CrossRef](#)]
55. Bondyra, A.; Kołodziejczak, M.; Kolikowski, R.; Giernacki, W. An Acoustic Fault Detection and Isolation System for Multirotor UAV. *Energies* **2022**, *15*, 3955. [[CrossRef](#)]
56. Song, K.Y.; Behzadfar, M.; Zhang, W. A Dynamic Pole Motion Approach for Control of Nonlinear Hybrid Soft Legs: A Preliminary Study. *Machines* **2022**, *10*, 875. [[CrossRef](#)]