

Quadrotor Flight System Design using Collective and Differential Morphing with SPSA and ANN

Oguz Kose*¹, Tugrul Oktay²

Submitted: 03/08/2021 Accepted : 28/09/2021

Abstract: Quadrotor modeling has been done with collective and differential morphing. Quadrotor initial state and morphing states are drawn in the Solidworks program. Newton-Euler approximation was used for quadrotor modeling. The mass and moment of inertia values required for modeling and simulation were obtained from the Solidworks program. Matlab / Simulink environment and state-space model approaches are used for simulations. A simultaneous perturbation stochastic approximation (SPSA) algorithm was used to determine the quadrotor morphing rates. If the morphing state obtained by SPSA is not included in the values obtained from the drawings, here it is provided to find the moments of inertia with the method based on learning by using the data obtained with the Artificial Neural Network(ANN). Proportional Integral Derivative (PID) is used as the quadrotor control algorithm. PID coefficients are also determined by SPSA.

Keywords: Artificial neural network, Collective morphing, Control, Differential morphing, Matlab, PID, Quadrotor, SPSA

This is an open access article under the CC BY-SA 4.0 license.
(<https://creativecommons.org/licenses/by-sa/4.0/>)

1. Introduction

In recent years, with the competition between developed and developing countries, great developments have been observed in the air defense systems and unmanned aerial vehicles(UAV) systems of these countries. As a result of these developments, UAVs have become effective tools for performing complex and strategic tasks. Quadrotor, a type of UAV, is widely used in reconnaissance, surveillance, security, intelligence, and military fields. In addition, the quadrotor is used in many areas such as hobbyist, cinema, agriculture, search and rescue in the civil field. In addition, when quadrotors are used in dangerous areas, they are controlled remotely or autonomously, eliminating the risk of pilots' life and attracting attention with their low costs.

The quadrotor is a four-rotor helicopter designed in a diagonal configuration with two pairs of opposing rotors rotating clockwise and the other pair of rotors rotating counter-clockwise to balance torque. Quadrotors, which are included in the category of rotary-wing unmanned aerial vehicles, do not need any runway since they have vertical take-off and landing(VTOL) features. Since they are 6 degrees of freedom(DOF), they can move in three axes.

Researchers and scientists have done a lot of work on quadrotor control and morphing in recent years. Khatoun et. al[1] presented a comparison between various control methods to be used in the control of a four-rotor unmanned aerial vehicle. It has taken traditional PID and linear-quadratic regulator (LQR) controllers as a control method. He concluded that the PID control provides better stability by bringing the closed-loop poles to the more negative side of the s-plane compared to the LQR control. The

system is stabilized using a PID or LQR controller. It appears that the LQR controller is intact and produces a very low steady-state error with a higher response speed. On the other hand, the PID controller provides a delayed response, making it a poor choice when the system needs fast parameter updates and does not have direct access to all states of the system. Noordin et. al.[2] designed a quadrotor micro aerial vehicle(MAV) flight controller using a PID. Since MAV was small, it was affected by slight perturbation. MAV used Gaussian Noise to minimize perturbation. They created the MAV quadrotor model in the MATLAB program. The personal area network (PAN) was used for low energy consumption. As a result, the MAV quadrotor performed well under PID control and external disturbances. Y. Bai[3] worked on a morphing quadrotor that performs special tasks with its geometry changing during flight. Morphing was performed by changing the length of the front and rear arms for the plus style quadrotor and by changing the arm intersection angles for the x-style quadrotor. It used servo motors to perform the morphing process in the quadrotor. An asymmetric change in quadrotor geometry generates the change in the moment of inertia. As a result, quadrotor control, stability, and performance effect have emerged. PID is preferred as the quadrotor control algorithm. Desbiez et. al.[4] worked on an X-type quadrotor with 380 grams, which they named X-Morf. In his work, the quadrotor was performing the morphing process by changing the angle between the two arms. They applied the Model Reference Adaptive Control (MRAC) law to eliminate uncertainties in the moment of inertia and center of mass during morphing. As a result of the tests, they showed that up to 28.5% of the quadrotor width can be increased or decreased in 0.5 seconds. Kose and Oktay[5] simultaneously discussed the morphing situation for a quadrotor's longitudinal and lateral flight. The quadrotor operated on the principle of lengthening or shortening the arms while its mass remained constant during morphing. The moment of inertia varied

^{1*} Gumushane University, School of Applied Sciences, Gumushane, TURKEY, ORCID ID : 0000-0002-8069-8749

² Erciyes University, Department of Aeronautical Engineering, Kayseri, TURKEY, ORCID ID : 0000-0003-4860-2230

* Corresponding Author Email: oguzkose24@gmail.com

during quadrotor morphing. In the study using the state-space model approach, the moments of inertia varied in the state space model according to the morphing state. The amount of morphing the quadrotor would do was determined by simultaneous perturbation stochastic approximation (SPSA). It also used PID as a control algorithm. Kose and Oktay[6, 7] studied the effect of differential morphing on forward and lateral flight using the PID algorithm. In the study, morphing worked according to the principle of simultaneous extension or shortening of the arms. Quadrotor full model and morphing states were drawn in the Solidworks program and moments of inertia that would occur during morphing were determined. PID was used as the control algorithm. Differential morphing was seen to affect forward and lateral flight in the study, which was obtained according to the comparison of the initial state and morphing states according to the design performance criteria. Many parameters of unmanned aerial vehicles can be controlled using artificial intelligence techniques. The use of artificial intelligence techniques causes fast and simple optimization for determining endurance and acceleration of UAV's[8]. Again, the use of artificial intelligence techniques in thrust-torque ratio for rotary-wing UAV's provides great advantages to designers[9]. The use of techniques such as the ANFIS model contributes to the design of more effective UAV's[10].

This article discusses the control of a quadrotor in the collective and differential morphing state. The quadrotor morphing state and controller coefficients were tried to be determined with SPSA. The changing moments of inertia in the case of morphing were obtained by ANN. Moments of inertia are obtained from the models drawn in the Solidworks program. However, when determining the morphing rate with SPSA, if there is an emergence of unavailable data, ANN has been activated here and it has been provided to obtain data that is not available with the learning-based deep learning method. With this method, quadrotor control was realized. With this study, simultaneous use of SPSA and ANN provides faster estimation of unknown parameters. Since unknown parameters are important in quadrotor control, these parameters must be determined quickly and reliably. There are various methods in the literature for estimating PID parameters, but it takes time to obtain them with conventional methods. Here, it is aimed to determine quickly using SPSA. In addition, the rapid determination of the parameters provides rapid response and improvements in the control of the quadrotor control.

2. Dynamic Model and State Space Model

Quadrotor flies with the help of four motors as shown in figure 1. Two of the motors rotate clockwise while the other two rotate counter-clockwise. Four motors rotate at the same speed and if the thrust generated by the motors is greater than the weight of the quadrotor, the quadrotor will make its vertical movement. In pitch, roll, and yaw movements, it is performed by changing the speed of the motors.

Quadrotor is a 6 degrees of freedom (DOF) tool so it contains 6 variables ($x, y, z, \phi, \theta, \psi$) which of them, x, y and z express orientations in space and ϕ, θ, ψ are known as Euler angles. In addition to 6 basic variables, 6 variables are used for the quadrotor speed. These u, v and w are used for linear velocity; p, q and r are used for angular velocity. Accordingly, the basic functions of the variables used for 6 DOF are as follows[11]. In this study, a quadrotor dynamical model has been created using the Newton-Euler approach. According to the Newton-Euler approach, the following opinions are valid[12].

- Quadrotor structure is considered to be rigid and symmetrical,
- The propellers on the quadrotor are solid,
- Quadrotor is directly proportional to the square of its thrust and drag propeller speeds.

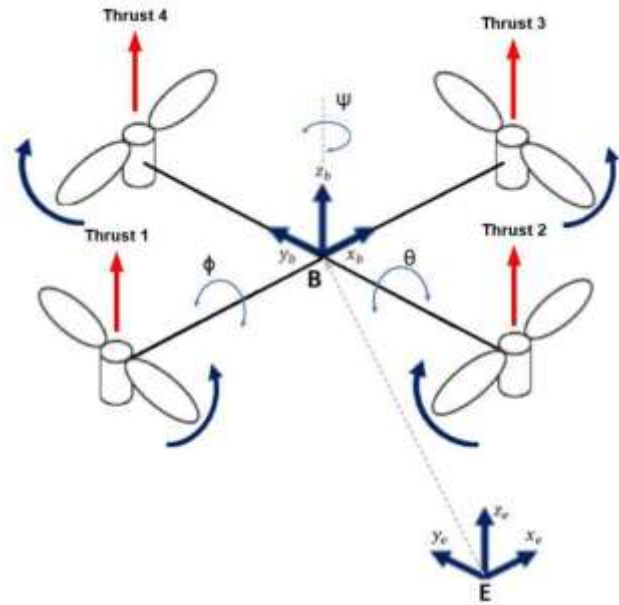


Fig. 1. Quadrotor

A state model approach is used to perform quadrotor modeling in Matlab / Simulink environment. The state-space model forms the basis of modern control theory. The state-space model is considered a set of first-order differential equations[13]. First-order differential equations in this set are obtained by showing inputs and outputs in matrix form. The variables of the state-space model are called state variables.

Quadrotor's longitudinal state-space model:

$$\begin{bmatrix} \dot{x} \\ \dot{z} \\ \dot{u} \\ \dot{w} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -g \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ z \\ u \\ w \\ q \\ \theta \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1/m & 0 \\ 0 & 1/I_y \\ 0 & 0 \end{bmatrix} \begin{bmatrix} f_t \\ \tau_y \end{bmatrix} \quad (1)$$

$$y = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ z \\ u \\ w \\ q \\ \theta \end{bmatrix} \quad (2)$$

Quadrotor's lateral state-space model:

$$\begin{bmatrix} \dot{y} \\ \dot{v} \\ \dot{p} \\ \dot{r} \\ \dot{\phi} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & g & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} y \\ v \\ p \\ r \\ \phi \\ \psi \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1/I_x & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \tau_x \\ \tau_z \end{bmatrix} \quad (3)$$

$$y = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} y \\ v \\ p \\ r \\ \phi \\ \psi \end{bmatrix} \quad (4)$$

3. Collective and Differential Morphing of Quadrotor

In nature, birds change their body shape to make various maneuvers while flying. These shape changes help birds hover or from narrow spaces[14]. In the quadrotor, which is a type of unmanned aerial vehicle, morphing is performed in several types. Morphing in the quadrotor is performed by changing the arm lengths or changing the intersection angles between the arms. There are two types of morphing processes. Active morphing is performed by changing the geometry of the quadrotor during flight and passive morphing by changing the quadrotor geometry before the flight. In this scope of work, active morphing was performed by changing arm lengths on an x-style quadrotor. In addition, in this study, the x-style quadrotor front and rear arms are not lengthened or shortened at the same time. The morphing starting point was taken as 50% of the arm length for the forearms and 10% of the arm length for the back arms. With collective morphing for the forearms, the arms are continuously extended from the starting point to the outside, while differential morphing ensures that the arms are continuously shortened from the starting point to the inner part. In collective morphing, the quadrotor arms lengthen or shorten simultaneously, while in differential morphing, the fore and back arms lengthen or shorten asynchronously. With collective morphing, the forearms move between 25% and 75%, while differential morphing moves between 5% and 20% with the back arm. Since the quadrotor moment arm length changes during collective or differential morphing, the moment of inertia changes[15]. The moments of inertia I_x , I_y , and I_z change during morphing. In this case, since moments of inertia are used in equations of motion, they must be changed in the state space model. 122 cases, including the initial state of quadrotor collective and differential morphing, were drawn in the Solidworks program and moments of inertia were taken from the solid body model according to the corresponding morphing state. Figure 2 shows the initial state and some collective and differential morphing situations. The quadrotor collective and differential morphing states are successive states and state transitions occur while the quadrotor is in the air.

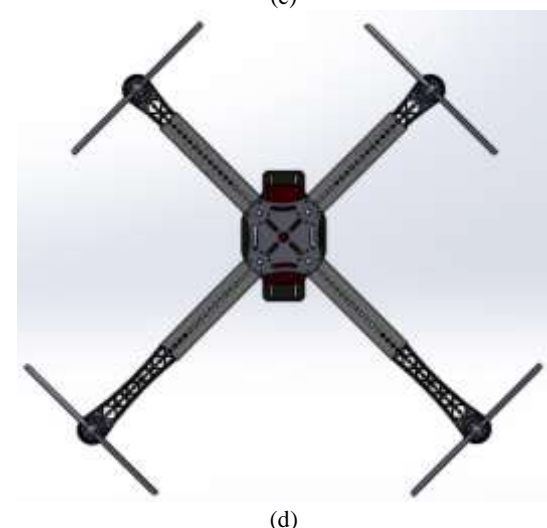
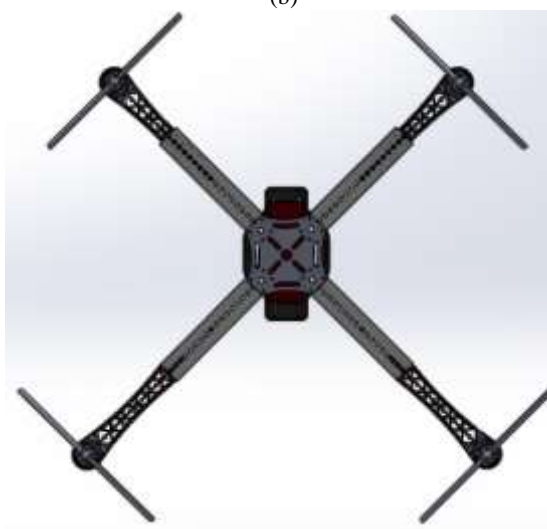
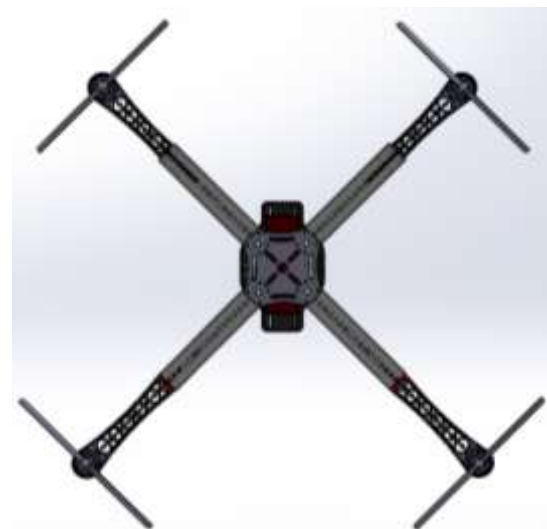
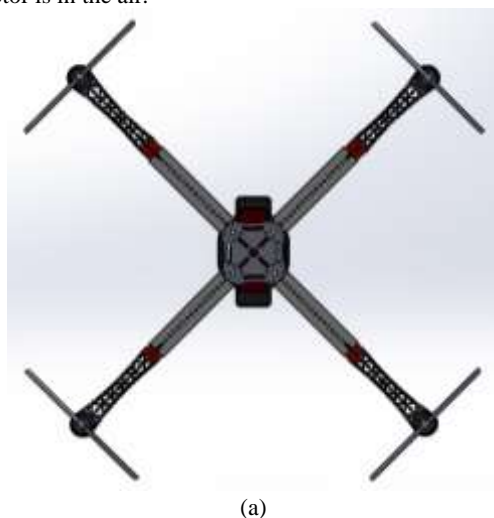


Fig. 2. (a) Initial situation (b) 30% Collective and 15% Differential morphing (c) 45% Collective and 17% Differential morphing (d) 75% Collective and 20% Differential morphing

Moments of inertia varying according to quadrotor morphing conditions are given in Table 1. Since the moments of inertia are taken directly from the models drawn in the Solidworks program, the error rates are not taken into account.

Table 1. Quadrotor mass and moments of inertia according to Fig. 2

	$m(\text{kg})$	$I_x(\text{kg} * \text{m}^2)$	$I_y(\text{kg} * \text{m}^2)$	$I_z(\text{kg} * \text{m}^2)$
Fig. 2 (a)	0.60333	0.08724	0.08745	0.02684
Fig. 2 (b)	0.60333	0.08792	0.08827	0.02835
Fig. 2 (c)	0.60333	0.08692	0.08720	0.02637
Fig. 2 (d)	0.60333	0.08533	0.08546	0.02304

4. SPSA and Artificial Neural Network Design

In this article, SPSA was used as an optimization method. SPSA is an optimization method used in many areas of engineering to solve multivariate optimization problems [16]. When there is a complex relationship between the objective function and optimization variables in optimization problems, analytical calculation of these variables can be difficult or impossible. Random optimization methods such as SPSA are used to overcome this situation. In recent years, a lot of research has been done on such optimization methods. Maryak and Daniel [17] used SPSA, an iterative optimization technique that searches for global optimum, instead of an algorithm that searches for local optimum values. They created two theorems for the SPSA convergence in their work. First, they tried to converge with the noise addition method. The second theorem, on the other hand, tried to converge to the global optimum without noise. Song et. al [18] worked on robust neural network tracking design for nonlinear systems. The neural network uses a closed-loop system to determine nonlinear system function. They used SPSA instead of the backpropagation algorithm to train a neural network. It has been demonstrated that the proposed system is measurable in terms of performance improvement, prevention of weight shifts, fast convergence and robustness against system failure compared to existing systems.

Multivariate random optimization methods are used in the control and analysis of many problems. In optimization problems, it is necessary to use an optimization algorithm that iteratively searches for the solution of the problem. Therefore, SPSA optimization method is useful for multivariate optimization problems. SPSA is an optimization algorithm that uses only objective function measurements to search for solutions. In the SPSA optimization process, it is the step-by-step process of changing the adjustable parameters from the first estimated value to the minimum value of the objective function.

In this article, SPSA has been used to determine a total of 8 parameters: longitudinal flight PID coefficients (K_p , K_i , and K_d), lateral flight PID coefficients (K_p , K_i , and K_d), collective morphing ratio and differential morphing ratio. SPSA is a fast algorithm in terms of providing optimum and reliable results. It uses the cost function for this. The cost function is shown in Equation 5. However, since longitudinal and lateral flights are considered separately in this study, the cost function is decomposed as shown in Equation 6 and 7. In addition, the total cost index can be shown as a percentage as in Equation 8.

$$J = T_{rt} + T_{st} + OS \quad (5)$$

$$J_{long} = T_{rt_{long}} + T_{st_{long}} + OS_{long} \quad (6)$$

$$J_{lat} = T_{rt_{lat}} + T_{st_{lat}} + OS_{lat} \quad (7)$$

$$\%J_{tot_i} = J_{long_i} + \frac{J_{long_0}}{J_{lat_0}} + J_{lat_i} \quad (8)$$

where T_{rt} is rise time, $T_{rt_{long}}$ and $T_{rt_{lat}}$ are the rise time for

longitudinal and lateral flight. T_{st} is settling time, $T_{st_{long}}$ and $T_{st_{lat}}$ are the settling time for longitudinal and lateral flight. OS is the overshoot value, OS_{long} and OS_{lat} are the overshoot value for longitudinal and sideways flight. J value gives the cost index, while J_{long} and J_{lat} give the longitudinal and lateral cost index. $\%J_{tot_i}$ is total cost index of i. iteration. J_{long_i} and J_{lat_i} are the longitudinal and lateral cost index of i. iteration. J_{long_0} and J_{lat_0} the longitudinal and lateral cost index of initial iteration

Optimization is the process of obtaining the most appropriate solution by providing certain constraints in line with the given purpose or objectives. Mathematical optimization deals with the problem of finding numerically minimums (or maximums or zeros) of a function. In this context, the function is called cost function. In this study, design performance criteria were used to obtain the cost function. Design performance criteria consist of system parameters such as rise time, settling time and overshoot. Of these parameters, rise time is the time required for the response to rise from 10% to 90%, the settling time is the time required for the response curve to reach and stay within a range about the final value of size specified by absolute percentage of the final value and overshoot is the maximum peak value of the response curve measured from unity. The values of the design performance criteria for longitudinal and lateral flight are added together to give the cost index for longitudinal and lateral flight. For the total cost index, the sum of the design performance criteria obtained for the flight is taken. For the total cost index in each iteration, the ratio of the longitudinal and lateral flight values in the relevant iteration and the longitudinal and lateral flight values in the first iteration is summed.

A neural network represents the human brain mechanism. Since the brain consists of many neurons, there are many node connections in the neural network. Neural networks mimic the connection of neurons using weight values.

Except for 122 morphing cases obtained in the Solidworks program, SPSA can estimate different values within limits. In this case, moments of inertia will not be defined in the optimization. Artificial neural network (ANN) has been used to overcome this situation. ANN are systems that can learn about events and determine how to react to environmental factors, using examples made by people. Similar to the human brain, ANN is successfully applied to topics such as learning, generalization, classification, characterization and association.

In the morphing state, the arms lengthen and shorten while the quadrotor mass remains constant. 122 morphing states, including the quadrotor initial state, were drawn in Solidworks. Moment of inertia values of these drawings are taken to be used in ANN. While 70% of this data was used for training, 30% was used for validation. As shown in Figure 3, a two-layer neuron structure was created for the ANN. The ANN takes the collective and differential morphing rates as input and gives these values to the trained network and gives the moments of inertia as output. In this case, the ANN has a structure with 2 inputs and 3 outputs. ANN learning rate was determined as 70%. In the optimization, the arm length value determined by SPSA is sent to the ANN structure and the most appropriate moment of inertia values for the determined arm length are determined.

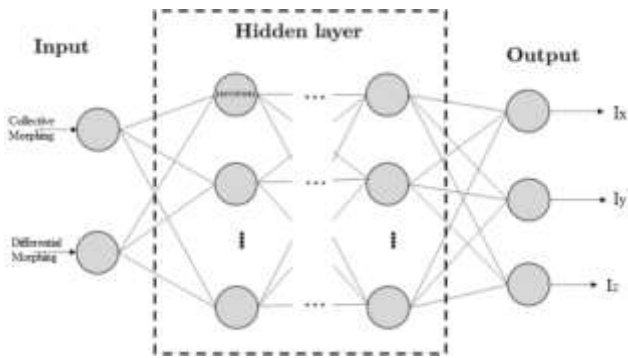


Fig. 3. ANN structure

In Figure 3, the first rounded segments represent the input layer. The task of the input layer is to transmit the values it receives to the next node. The rightmost nodes represent the output layer. The value of these nodes represents the final output of the neural network. The intermediate layers between the input and output nodes are called hidden layers. It is not possible to access these layers from outside the neural network.

5. Result and Discussions

In this article, optimization and simulation have been done in Matlab / Simulink environment. Longitudinal and lateral flight simulations were performed separately for collective and differential morphing. Quadrotor trajectory tracking is taken for longitudinal and lateral flight with 1 degree and $\pi/2$ degree order. Collective morphing, differential morphing, cost index parameters, longitudinal and lateral flight PID parameters are given below in graphs. Analytical calculation of moments of inertia depending on the morphing state optimized with SPSA is a difficult task. The estimated moments of inertia at each iteration with the ANN are as follows.

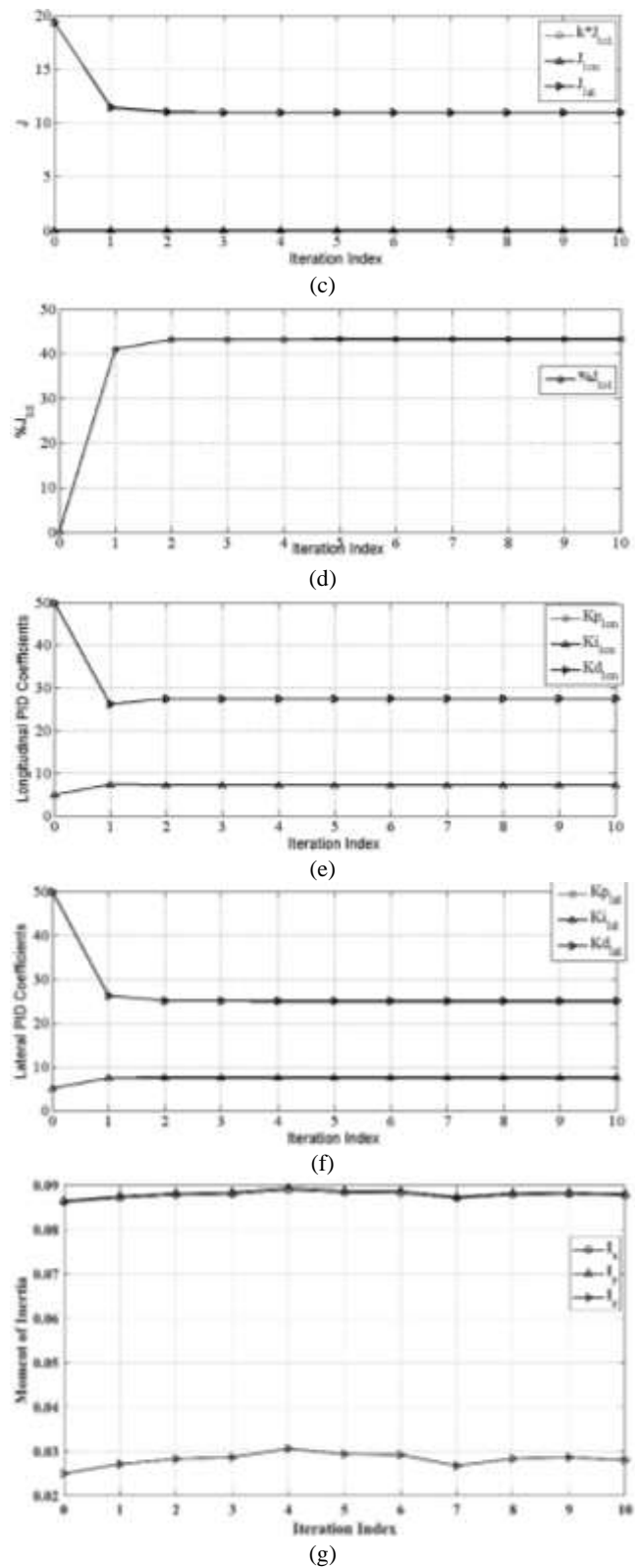
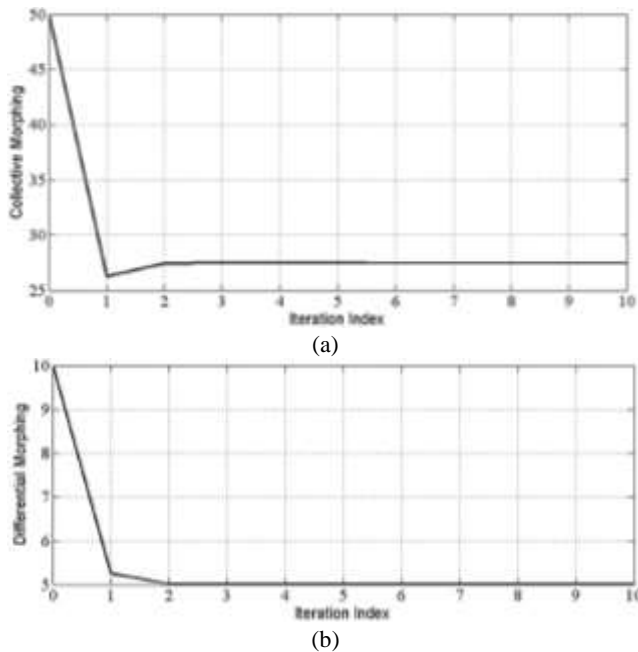


Fig. 4. (a) Collective morphing (b) Differential morphing (c) Cost (d) Relative cost (e) Longitudinal PID (f) Lateral PID (g) Moment of Inertia

In the simulation, the quadrotor successfully followed the trajectory given for longitudinal and lateral flight.

In this study, UAV control, which has an important place in aviation, is discussed. A UAV control with morphing is provided by changing the arm lengths by going beyond the traditional methods. The control algorithm, which is one of the main difficulties in UAV control, is combined with SPSA and ANN to obtain the most optimum control parameters (PID coefficients). In

this respect, SPSA and ANN application has shown that it can be successfully applied in morphing UAVs.

6. Conclusion

In this article, longitudinal and lateral flight autopilot system is discussed with collective and differential morphing. Active morphing was chosen as the morphing type. SPSA was used to obtain the optimal morphing parameters and the PID coefficients required for both longitudinal and lateral flight depending on these parameters. ANN was used to obtain the moment of inertia parameters in case the morphing rate was determined except for the morphing cases drawn in the Solidworks program.

Using SPSA, the cost index has been improved by 43% compared to the initial situation. The improvement of the cost index has given good performance for both longitudinal and lateral flight. PID coefficients were selected as $P = 50$, $I = 5$, and $D = 50$ for both longitudinal and lateral flight in the initial state. With SPSA, the optimum PID coefficients for longitudinal flight were determined as $P = 27.5$, $I = 7.25$ and $D = 27.5$. For lateral flight, $P = 25$, $I = 7.5$ and $D = 24.9$. Morphing rates and PID coefficients obtained with SPSA did not show catastrophic behaviour in simulations. The given trajectories were followed successfully, depending on the parameters obtained. The collective morphing of the forearms of the x-type quadrotor and the differential morphing of the rear arms have been successfully applied.

Morphing rates were determined as 50% for the collective and 10% for the differential in the initial condition without SPSA. The PID coefficients for longitudinal and lateral flight are $P=50$, $I=10$ and $D=50$. In this case, there was no improvement in the cost index value. In the first iteration with SPSA, improvements were obtained in both cost index value and PID coefficient values. In this context, when the initial state and the SPSA final iteration state are compared, the quadrotor control for both longitudinal and lateral flight has been successfully achieved and all other parameters have been obtained quickly and reliably.

Acknowledgements

This work was supported by the Research Fund of Erciyes University Scientific Research Projects (BAP) Coordination Unit under Project Number: FBA-2020-10102.

References

- [1] S. Khatoon, M. Shahid, Ibraheem, and H. Chaudhary, "Dynamic modeling and stabilization of quadrotor using PID controller," *Proc. 2014 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2014*, pp. 746–750, 2014, doi: 10.1109/ICACCI.2014.6968383.
- [2] A. Noordin, M. A. M. Basri, and Z. Mohamed, "Simulation and experimental study on pid control of a quadrotor MAV with perturbation," *Bull. Electr. Eng. Informatics*, vol. 9, no. 5, pp. 1811–1818, 2020, doi: 10.11591/eei.v9i5.2158.
- [3] Y. Bai, "Control and simulation of morphing quadcopter," 2017.
- [4] A. Desbief *et al.*, "X-Morf: a crash-separable quadrotor that morphs its X-geometry in flight To cite this version: HAL Id: hal-01644528," 2017.
- [5] O. Kose and T. Oktay, "Simultaneous quadrotor autopilot system and collective morphing system design," *Aircr. Eng. Aerosp. Technol.*, 2020, doi: 10.1108/AEAT-01-2020-0026.
- [6] O. Kose and T. Oktay, "Investigation of the Effect of Differential Morphing on Forward Flight by Using PID Algorithm in

- Quadrotors," *J. Aviat.*, vol. 4, no. 1, pp. 15–21, 2020.
- [7] O. Kose and T. Oktay, "Investigation of the Effect of Differential Morphing on Lateral Flight by Using PID Algorithm in Quadrotors," *Eur. J. Sci. Technol.*, no. 18, pp. 636–644, 2020, doi: 10.31590/ejosat.702727.
- [8] M. Konar, "Simultaneous determination of maximum acceleration and endurance of morphing UAV with ABC algorithm-based model," *Aircr. Eng. Aerosp. Technol.*, vol. 92, no. 4, pp. 579–586, 2020, doi: 10.1108/AEAT-11-2019-0229.
- [9] M. Konar, A. Turkmen, and T. Oktay, "Improvement of the thrust-torque ratio of an unmanned helicopter by using the ABC algorithm," *Aircr. Eng. Aerosp. Technol.*, vol. 92, no. 8, pp. 1133–1139, 2020, doi: 10.1108/AEAT-03-2020-0057.
- [10] M. Konar, "Redesign of morphing UAV's winglet using DS algorithm based ANFIS model," *Aircr. Eng. Aerosp. Technol.*, vol. 91, no. 9, pp. 1214–1222, 2019, doi: 10.1108/AEAT-09-2018-0255.
- [11] M. Walid, N. Slaheddine, A. Mohamed, and B. Lamjed, "Modelling, identification and control of a quadrotor UAV," *2018 15th Int. Multi-Conference Syst. Signals Devices, SSD 2018*, no. October, pp. 1017–1022, 2018, doi: 10.1109/SSD.2018.8570512.
- [12] H. talla M. ElKholy, "Dynamic Modeling and Control of a Quadrotor Using Linear and Nonlinear Approaches," 2014.
- [13] T. Oktay and S. Coban, "Simultaneous longitudinal and lateral flight control systems design for both passive and active morphing TUAUs," *Elektron. ir Elektrotehnika*, vol. 23, no. 5, pp. 15–20, 2017.
- [14] T. OKTAY and S. Coban, "Lateral Autonomous Performance Maximization of Tactical Unmanned Aerial Vehicles by Integrated Passive and Active Morphing," *Int. J. Adv. Res. Eng.*, vol. 3, no. 1, p. 1, 2017, doi: 10.24178/ijare.2017.3.1.01.
- [15] O. Kose and T. Oktay, "Non Simultaneous Morphing System Desing for Quadrotors," no. 16, pp. 577–588, 2019, doi: 10.31590/ejosat.569785.
- [16] J. C. Spall, "M An Overview of the Simultaneous Perturbation Method for Efficient Optimization," 1998.
- [17] J. L. Maryak and C. C. Daniel, "Global Random Optimization by Simultaneous Perturbation Stochastic Approximation," 2001, vol. 1, no. Section 2, pp. 756–762.
- [18] J. N. Song, Qing, James C. Spall, Yeng Chai Soh, "Robust Neural Network Tracking Controller Using Simultaneous Perturbation Stochastic Approximation," in *IEEE TRANSACTIONS ON NEURAL NETWORKS*, 2008, pp. 817–835.