Optimization of Fuzzy Logic Quadrotor Attitude Controller - Particle Swarm, Cuckoo Search and BAT Algorithms

Mohamed Siddiq Zatout[1], Amar Rezoug[2*], Abdellah Rezoug[3], Khalifa Baizid[4], and Jamshed Iqbal[5]

[1]Université Djilali Bounaama Khemis Miliana, Aïn Defla, Algeria., zatoutsed@gmail.com
[2*]Laboratoire de Technologie Innovate, Ecole Nationale Superieure de Technologie Diplomatic City, Dergana-Bordj El Kiffan, Algiers, Algeria. amar.rezoug@enst.dz
[3]Department of Computer science, University Mhamed Bougara, Boumerdes Boumerdes, Algeria. abdellah.rezoug@gmail.com
[4]Advanced Industrial Automation, Italian Institute of Technology Genova, Italy.baizid.khelifa@gmail.com
[5]Department of Computer Science and Technology, Faculty of Science and Engineering, University of Hull,HU6 7RX, UK. j.iqbal@hull.ac.uk

Abstract
Bio-inspired optimization algorithms have recently attracted much attention in the control community. Most of these algorithms mimic particular behaviors of some animal species in such a way that allows solving optimization problems. The present paper aims at applying three metaheuristic methods for optimizing Fuzzy Logic Controllers used for quadrotor attitude stabilization. The investigated methods are Particle Swarm Optimization (PSO), BAT algorithm and Cuckoo Search (CS). These methods are applied to find the best output distribution of singleton membership functions of the Fuzzy Controllers. The quadrotor control requires measured responses, therefore, three objective functions are considered: Integral Squared Error, Integral Time-weighted Absolute Error and Integral Time-Squared Error. These metrics allow performance comparison of to compare the controllers in terms of tracking errors and speed of convergence. The simulation results indicate that BAT algorithm demonstrated higher performance than both PSO and CS. Furthermore, BAT algorithm is capable of offering 50% less computation time than CS and 10% less time than PSO. In terms of fitness, BAT algorithm achieved an average of 5% better fitness than PSO and 15% better than Cuckoo Search. According to these results, the BAT-based Fuzzy Controller exhibits superior performance compared with other algorithms to stabilize the quadrotor.

Keywords
Meta-heuristic Fuzzy Logic Quadrotor Attitude Controller Particle Swarm Optimization Cuckoo Search BAT Algorithm.

1 Introduction

Fuzzy Logic Systems (FLS) is an active research topic in automation community and is based on the fuzzy sets theory of L. Zadeh [1]. The first application of FLS in control has been realized by Mamdani in 1975. The basic idea behind a FLS consists of using human experiences to design a Fuzzy Inference System (FIS). The operation of a Fuzzy Logic Controller (FLC) mainly depends on the characteristics of three subsystems: fuzzification, fuzzy rules and defuzzification [2]. Various reported applications show that FLC demonstrates superior performance compared to the classical control algorithms [3-6].

A FLC is based on human's expertise and is easy to design and implement because of its structural simplicity, less computational complexity, and greater control performance [4-6]. Besides, a FLC is very useful for the systems that are hard to model due to their strong inherent nonlinearities and uncertainties [7]. Considering a quadrotor system, scientific literature reports several nonlinear controllers such as, observer-based adaptive fuzzy backstepping [8] model-free control [9], gain-scheduled Proportional Integral Derivative (PID) control [10], fractional-order super twisting sliding mode control [11] and FLC . In this regard, numerous studies have covered various aspects of Unmanned Aerial Vehicle (UAV) systems such as, stability enhancement [3], payload drop missions [4], fault-tolerance [5], visual-based control [6], taxi model [10], path following [11] and quadrotor stabilization [12]. Particularly, the fuzzy control of a quadrotor with or without optimization is reported in [4, 5, 12, 14, 15, 16]. On the other hand, human's expertise does not allow the synthesis of an optimal FLC, therefore; fuzzy-based controllers exhibit limited performance and robustness. The parametric and structural optimization of a FLC makes it possible to achieve suitable performance and robustness. A multitude of fuzzy controller optimization methods have been proposed and applied in the scientific literature [17-25], which can be classified into two categories; the first is the optimization based on analytical methods. A prominent example of this category is the gradient descent algorithm. The second category employs the bio-inspired optimization methods to find the optimal FLC. They are based on population coding and iterative search of the optimal value of a fitness function in this population. In case of FIS, researchers have focused on optimizing Membership Functions (MFs) [26] or fuzzy rules [27] or both of these simultaneously. The commonly used optimization methods include; Genetic Algorithms [27], Ant Colony Optimization (ACO) [17, 24], Particle Swarm Optimization (PSO) [28], Bee Colony Optimization [29] and others [30]. These algorithms have been applied for various UAV systems such as birotor helicopter [21, 23, 24], quadrotor [4, 6, 12, 14, 15, 16, 18, 25], hexacopter [31] and so on.

Few comparative studies of metaheuristic methods have been recently presented owing to novelty of research in this domain and emergence of several algorithmic variants. Studies have addressed the control of a quadrotor UAV system while considering different optimization objectives. PSO-tuned FLC for full autopilot control of a quadrotor dealing with wind disturbances is proposed in [18]. PSO algorithm is utilized to have minimum 4 rules for FLC to improve the controller response. However, the use of PSO has become relatively standard. An approach named as PD-T2-FNN has been proposed in [22] which consists of tuning type-2 Fuzzy Neural Networks (FNN) based on a novel PSO-Sliding Mode Control (SMC) hybrid learning algorithm. PSO has been used for identification of the antecedent part of the T2-FNNs. Another research work [24] proposed optimization of normalized type-2 fuzzy controller parameters based on ACO algorithm which is compared with PSO. Nevertheless, the real-time implementation of these optimized controllers requires advanced computing resources due to several inherent mathematical operations. An Adaptive PSO for optimal Linear Quadratic Regulator (LQR) tracking control of a 2-DoF (Degree of Freedom) laboratory helicopter is proposed in [21], where the adaptive PSO method is used to obtain the elements of Q and R matrices. However, LQR is an optimal controller for linear system whereas a laboratory helicopter presents some nonlinearities and uncertainties.

The present research presents a comparative study of three popular metaheuristic techniques; PSO [28], BAT algorithm [32] and Cuckoo search (CS) [33, 34] which are applied to optimize the distribution of the singleton output MF to control attitude of a quadrotor system. These techniques or their variants have gradually become more popular in scientific community and have successfully been applied to diverse domains owing to their ease in implementation, requirement of few parameters and demand of exceptional performance. As a Genetic Algorithm (GA), PSO has become a de facto standard for comparing bio-inspired methods. CS algorithm was chosen because it has a relatively high convergence speed and a low computational cost compared to several metaheuristics algorithms. BAT algorithm is a latest metaheuristic optimization technique inspired by the echolocation skill of the microbats which directs them based on their foraging behavior. BAT algorithm incorporates major strengths of PSO and Harmony Search (HS) algorithm and thus may give relatively superior performance. The objective of this work is to determine the algorithm that demonstrates superior performance for a quadrotor attitude control. In contrast with another comparative study [35], which applies the optimized fuzzy controller on a brushless motor considering the system dynamics as linear, the present work examines three algorithms under a strong nonlinear multivariable system well known by its control complexity. The fitness functions used are Integral Squared Error (ISE) [30], Integral Time-weighted Absolute Error (ITAE) [35] and Integral Time Squared Error (ITSE). These objective functions are selected to examine precision in case of ISE and robustness in case of ITAE. ITSE permits simultaneous evaluation of precision and speed of convergence. The choice to optimize only the MFs output is based on modeling the input dynamics by two fuzzy variables (error and its change). Hence, the use of Singletons MFs for the outputs addresses performance and robustness with possible limitations. In this regard, the optimization of the output offers improved performance. The minimization of the number of parameters to be optimized is also an important reason considering that the processing time is closely linked with specifications of the equipment used for optimization. This paper sights to present three main contributions: Firstly, a FLC of a quadrotor attitude to improve its speed is detailed. Moreover, the optimization of this controller is based on three metaheuristic algorithms which are PSO, CS and BAT. Secondly, a thorough statistical analysis is performed demonstrating that the FLC based on BAT algorithm is more appropriate for stabilizing a quadrotor than PSO and CS algorithms. Thirdly, the application of the optimized controllers with and without disturbances has proven the superiority of the BAT algorithm in regulation and trajectory tracking modes compared to PSO and CS.The novelty of the proposed work is mentioned below in a point by point manner:

i. Various studies applying FLC on UAV systems are reported (Table I) highlighting metaheuristic optimization (Table II). However, very few of the recently reported works (please refer to Table III) present a rigorous performance comparison among diversified algorithms that lead to a concrete conclusion.

ii. The comprehensive analysis presented in the proposed work involves various design aspects such as, the dynamic model of the quadrotor is derived while taking into account the rotor dynamics, the environment dynamics is also involved in simulation, etc.

iii. Another notable feature of the proposed work is to address both global algorithms (i.e. PSO, CS and BAT) as well as local methods (PEO) using a fuzzy controller with a very clear rules table (Section III.1). In addition these methods are compared with backstepping a PD controllers.

iv. The comparative analysis based on multiple fitness functions (ISE, ITAE and ITSE) makes the proposed work comprehensive leading to a potentially useful conclusion that the fuzzy controller based on BAT algorithm demonstrates superior performance in transient as well as in steady state compared to PSO, CS and PEO algorithms. FLC based BAT is also best to an integral backstepping method.

The remaining of the paper is organized in the following Sections: Section II summarizes the related state-of-the-art, presents the quadrotor dynamic model and briefly describes the related bio-inspired metaheuristics methods. Section III provides a description of the proposed FLC optimized for a quadrotor attitude control. Section IV details the simulation results which are discussed in Section V. Finally, Section VI concludes the paper.

2 Related works and background

2.1 Related works

In order to demonstrate the importance of the research domains pertinent to the present research paper, three review tables are given in this section. Table I describes approaches that apply fuzzy logic in standalone and in combination with other techniques on UAV systems. Table II lists recently reported studies addressing the metaheuristic optimizations of FLCs for UAV systems. Table III illustrates the comparative studies published in last few years with a focus on the nature of the solved problem.

2.2 Quadrotor Dynamic Model

A quadrotor is an under-actuated system composed of four rotors structured in a cross configuration where each symmetric pair of propellers turns in opposite direction. The movement is created by varying speeds in one or several rotors. A quadrotor is a highly nonlinear MIMO (Multiple Input Multiple Output) deterministic system that presents internal parametric and external nonparametric uncertainties with many physical phenomena characterizing its dynamics. It is pertinent to mention here that exact real-world quadrotor dynamics is very difficult to be defined. In order to develop a reasonable dynamic model of the system, several assumptions need to be considered such as:

1. **Assumption 1.** (i) The structure is assumed to be rigid and strictly symmetrical, (ii) The torque is proportional to DC motor voltage and (iii) The quadrotor reference is supposed to be confined with its center of gravity.

2. **Assumption 2.** Quadrotor dynamics is related to the translational positions (x,y,z) and attitude described by the angles (ϕ, θ, ψ). The angles are characterized by the following constraints:

$$-\pi/2 \leq (\phi, \theta) \leq \pi/2 - \pi \leq \psi \leq \pi \quad (1)$$

The coordinate system is composed of a fixed frame O_e and a body frame O_B (Figure 1).

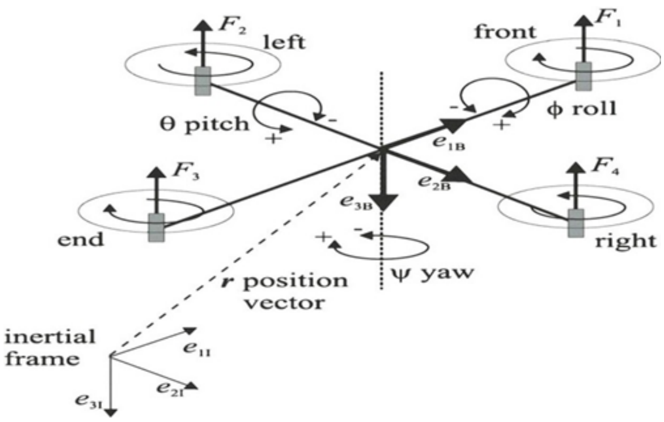


Figure 1. Schema of a quadrotor UAV

Using Newton-Euler formulation (equation 2), dynamics of the quadrotor is written as

$$\begin{cases} m\ddot{\xi} = F_f + F_d + F_g \\ J\dot{\Omega} = -\Omega \wedge J\Omega + \Gamma_f + \Gamma_a + \Gamma_g \end{cases} \quad (2)$$

Using (equation 2), the quadrotor dynamic model can be written as [41],

Name of the approach	Year	Focus	Remarks	Reference
Generic Self-Evolving Neuro-Fuzzy controller	2018	Altitude hexacopter UAV control	Various trajectories are tested	[37]
Uncertain T-S Fuzzy controller	2018	Attitude stabilization of a quadrotor	New generic and relaxed Linear Matrix Inequality (LMI) conditions are proposed	[15]
Generic-controller	2018	Control of hexacopter	Stability of the G-controller is guaranteed using the Lyapunov function. Learning machine namely Generic Evolving Neuro-Fuzzy Inference System is used	[31]
Fuzzy GS-PID Controller	2018	Reducing overshoot of a quadrotor during payload in drop missions	Experimental results are presented	[4]
Adaptive Fuzzy Kalman Fusion Algorithm (AFKF)	2018	Estimation of the quadrotor states and control using adaptive FLC	Precision landing of quadrotor	[13]
Active Fuzzy Fault Tolerant Tracking Control (AFFTTC)	2018	Addressed MIMO unknown nonlinear systems in the presence of unknown actuator faults, sensor failures and external disturbance.	A fuzzy adaptive controller based on back-stepping design is developed	[5]
Cen-NSFLC	2018	Visual Cen-NSFLCs control for trajectory tracking accurate in a real-time for quadrotor	Cen-NSFLCs are compared with PIDs, SFLCs and Tra-NSFLCs	[16]
T–S fuzzy MASs	2018	Sliding-mode control (SMC) problem of Takagi–Sugeno (T–S) fuzzy Multi-Agent Systems (MASs)	A cooperative fuzzy-based dynamical sliding-mode controller is designed and the overall closed-loop T–S fuzzy MAS is constructed	[36]
PID fuzzy controller	2017	Control of attitude and z-axis of quadrotor system	Stability of the PID fuzzy controller is guaranteed	[12]
Fuzzy Cognitive Map (FCM)	2016	FCM is suitable choice for implementing a vision-based intelligent technique	No sensor has been installed on the moving object UAV and the changes in its yaw angle are not available	[6]
Fuzzy sliding mode controller named "AFGS-SMC"	2016	Attitude stabilization of quadrotors with parametric uncertainties and external disturbances	Use of the fuzzy logic to estimate discontinuous control part of a sliding mode controller	[14]

Table 1. Fuzzy logic applied to UAV systems - Recent stat of the art

$$\begin{cases} \ddot{\phi} = \frac{\dot{\theta}\dot{\psi}(I_y-I_z)-K_{ax}\dot{\phi}^2-J_r\dot{\theta}\Omega_r+U_2}{I_x} + \zeta(\phi) \\ \ddot{\theta} = \frac{\dot{\phi}\dot{\psi}(I_z-I_x)-K_{ay}\dot{\theta}^2-J_r\dot{\phi}\Omega_r+U_3}{I_y} + \zeta(\theta) \\ \ddot{\psi} = \frac{\dot{\phi}\dot{\theta}(I_x-I_y)-K_{az}\dot{\psi}^2+U_4}{I_z} + \zeta(\psi) \end{cases} \tag{3}$$

where

$$\begin{cases} U_1 = b(w_1^2+w_2^2+w_3^2+w_4^2) \\ U_2 = bl(w_3^2-w_1^2) \\ U_3 = bl(w_4^2-w_2^2) \\ U_4 = d(w_1^2-w_2^2+w_3^2-w_4^2) \\ \Omega_r = w_1-w_2+w_3-w_4 \end{cases} \tag{4}$$

$\zeta(\phi), \zeta(\theta)$ and $\zeta(\psi)$ represent time-varying external disturbances affecting on roll, pitch and yaw dynamics of the quadrotor respectively, $U_i (i = 1,2,3,4)$ are the control inputs, Ω_r is the total residual speed of rotors, $b, dandl$ are factors corresponding to thrust, drag and lever respectively.

Rotor dynamics

The quadrotor has four fixed pitch rotors and for each rotor, a DC motor is used to actuate the system. As reported in [42], a first-order transfer function given in (equation 5) is sufficient to represent the rotor dynamics considering the reference speed of the propeller and its actual speed.

$$G(s) = \frac{0.936}{0.178s+1} \tag{5}$$

The input to the motor is the armature voltage and its output is the angular velocity. A PD controller controls the rotor system. Introducing the rotor dynamics, the model of the quadrotor essentially becomes more sophisticated.

Name of the approach	Year	Focus	Reference
Dynamic ant colony's labor division (DACLD) model for swarm UAVs	2018	New dynamic environmental stimulus, response threshold and transition probability has been designed and DACLD is proposed for swarm UAVs	[17]
PSO tuned FLC for full autopilot control	2018	PSO tuned FLC for full autopilot control of quadrotor to tackle wind disturbance using bond graph approach	[18]
PD -T2-FNN	2018	FNN is trained using a novel PSO-SMC hybrid learning algorithm	[22]
A novel algorithm identifying optimal flight trajectories for UAVs compliant with environmental constraints	2017	The proposed path planning method is based on an original trajectory modeling coupled with PSO approach	[19]
A novel multi-UAVs coordinated path planning method based on the k-degree smoothing	2016	The k-degree smoothing is the improved ACO	[20]
A novel method for Small Unmanned Helicopter (SUH) system identification based on improved fuzzy PSO	2016	The corresponding aerodynamic parameters of the state-space model of the SUH	[23]
Adaptive PSO for optimal LQR tracking control of 2 DoF laboratory helicopter	2016	Adaptive Particle Swarm Optimization (APSO) method to obtain the elements of Q and R matrices	[21]
ACO fuzzy controller	2014	Optimization of normalization fuzzy parameters	[24]

Table 2. **Metaheuristic optimization of FLCs for UAVs**

Methods compared	Year	Focus	Reference
Firefly Algorithm (FA), Differential Evolution (DE), Artificial Bee Colony (ABC), Harmony Search (HS) and Directed Tabu Search (DTS), Hooke-and-Jeeves (HJ) local	2018	Parameter estimation problem in dynamic systems	[38]
PSO, BCO and BAT	2017	Comparative Study of Type-2 Fuzzyfor the fuzzy controllers	[29]
CS and DE	2016	Solving the constrained optimization problem from selected benchmark functions	[34]
PSO algorithm and its variants	2015	Review of approaches and applications	[28]
BAT algorithm and its variants	2013	Review of approaches and applications	[32]
CS and its variants such as (Discrete binary CS, Binary CS, etc.)	2013	Review of approaches and applications	[33]
ACO and PSO	2012	Optimization of the MFs of FLC for finding the optimal controller for an autonomous wheeled mobile robot	[26]
PSO , Differential Evolution (DE), and Scatter Search	2012	Comparison of GPU-based implementations	[39]
ACO, GA, simulated annealing (SA) and Tabu Search (TS)	2008	Solving Customer Order Scheduling Problem (COSP)	[40]

Table 3. **Review of metaheuristic optimization comparative studies**

2.3 Metaheuristic Methods

This section introduces the related metaheuristic methods, i.e. PSO, CS and BAT algorithms. Mathematical relationships and necessary parameters are given for these algorithms.

2.3.1 PSO Algorithm

PSO algorithm imitates the social behavior of people or animals [43]. It is a stochastic search technique where the potential solutions, called particles move through the problem space until some criteria are met, and it applies to every method (the solution). During the processing time, the updates in positions and velocities of particles are based on the local and global best solutions. PSO algorithm is given by two main equations:

1. *Position of a particle i:* The next position of a particle $x_i(t+1)$ is measured by adding its velocity to the

actual position i.e.

$$x_i(t+1) = x_i(t) + V_i(t+1) \tag{6}$$

2. *Velocity update:* The velocity of a particle i at time instant $'t+1'$ is calculated as

$$V_i(t+1) = V_i(t) + c_1 r_1(t)(x_{besti}(t) - x_i(t))$$
$$+ c_2 r_2(t)(x_{Gbest}(t) - x_i(t)) \tag{7}$$

where, $x_i(t)$ and $V_i(t)$ are the position and the velocity of particle i at time t respectively. c_1 and c_2 are real positive constants and $r_1(t)$ and $r_2(t)$ are random values included in the interval $[0,1]$. $x_{besti}(t)$ is the best position found by particle i and $x_{Gbest}(t)$ is the best position found so far by all the particles in the swarm.

2.3.2 CS Algorithm

CS is an evolutionary metaheuristic approach that was first proposed by Xin. [33, 34]. It simulates the reproduction process of the cuckoos. The CS procedure is based on three main features: (1) Each cuckoo lays one egg at a time. He drops it in a nest that he chooses randomly. (2) The best nests that include eggs (solutions) of good quality build the members of the new generation. (3) The number of valid host nests is fixed. The host bird can detect the foreign cuckoo with a probability $P_a \in [0,1]$. Probability P_a represents the fraction of N nests to be replaced by new nests (with new random solutions at new positions in the search). The quality of a nest (or a solution) is measured according to the fitness function that varies from one problem to another. CS algorithm is characterized by the Lévy flight which is described in below subsection.
In order to generate a new solution $x(t+1)$ for a cuckoo i, Yang and Deb [44] integrated the Lévy flight in the following way

$$x_i(t+1) = x_i(t) + \alpha \oplus L\acute{e}vy(\lambda) \tag{8}$$

where $\alpha > 0$ is the step size and is related to the problem under consideration. In most of the cases, α is assumed to be unity. $\oplus$ is the boolean XOR operator.
Lévy flight was proposed by French mathematician Paul Pierre Lévy [33, 34]. Since its creation, Lévy's flight has given theoretical interpretations to several physical, chemical, biological and natural phenomena. In fact, Lévy's flight makes it possible to model random sequence composed of a large number of steps where the transitions are based on probabilities. In mathematical terminology, Lévy's flight is a random trajectory in which the distance between the steps has a probability distribution that is heavily tailed. The distribution of Lévy is given by (9).

$$L\acute{e}vy \sim u = t^{(-\lambda)}, (1 < \lambda \leq 3) \tag{9}$$

The step length is calculated by (10)

$$s = \frac{u}{|v|^{(1/\beta)}} \tag{10}$$

where u and v are drawn by the normal distribution

$$u \sim N(0, \sigma_u^2), \qquad v \sim N(0, \sigma_v^2)$$

with

$$\sigma_u = \left\{ \frac{\Gamma(1+\beta) sin(\frac{\pi\beta}{2})}{\Gamma\left[\frac{1+\beta}{2}\right] \beta 2^{\frac{(\beta-1)}{2}}} \right\}^{1/\beta}$$

where Γ is the gamma function and $0 < \beta < 2$.

2.3.3 BAT Algorithm

Proposed by Xin-She Yang in 2010 [32], the optimization using the BAT algorithm is inspired by the behavior of a bat. In order to avoid obstacles and to target their prey, bats send forth some pulses to the environment. The returned echo permits possibility to identify various objects in the surroundings. BAT algorithm can be employed to solve continuous optimization problems where the possible solutions can be represented by the geographical positions of the bats. The principle of the algorithm is given below:

- In the space R^n, at a time instant t, each bat i is associated with a position $x_i(t) \in R^n x_i^t$ and it moves with a velocity $V_i(t) \in R^n$ while emitting a pulse of frequency f_{min}.
- At the perception of a prey, frequency, position and velocity parameters are adjusted according to the relationships (11-13):

$$f_i = f_{min} + \beta(f_{max} - f_{min}) \tag{11}$$

$$V_i(t+1) = V_i(t) + f_i(x_i(t) - x_{best}(t)) \tag{12}$$

$$x_i(t+1) = x_i(t) + V_i(t+1) \tag{13}$$

where β is a random vector and x_{best} is the position of the best bat of the group.
In order to search around the best position (x_{best}), Yang [45] has proposed a modification in (11-13) which permits BAT algorithm to consider the solution near the best position. In this regard, a new solution for each bat is generated locally using a random sequence given by (14).

$$x_{new} = x_{best} + \varepsilon A(t) \tag{14}$$

where $A(t) = < A_i(t) >$ is the average loudness of all bats computed at the t^{th} iteration and $\varepsilon \in [-1,1]$ is an uniformly distributed random value. A new solution is accepted if a uniform random number is less than the current loudness A_i and the current fitness value is better than that of global best solution. In order to achieve the balance between exploration and exploitation during the search process, loudness A_i and pulse emission rate r_i should be updated only if the candidate solution is improved as the iterations proceed [46]. In each iteration, the frequency A_i and the amplitude r_i are then updated by (15-16):

$$A_i(t+1) = \alpha A_i(t) \tag{15}$$

$$r_i(t+1) = r_i(0)(1 - e^{\gamma t}) \quad (16)$$

where α and γ are positive real constants satisfying conditions $0 < \alpha < 1$ and $\gamma > 0$.

2.4 The Objective Functions

In this research paper, three criteria are used to examine the performance. These criteria are $ISE, ITAE$ and $ITSE$. The multiplicative term penalizes the error more at the later stages than at the start and therefore effectively reduces the settling time [35]. The square of the error function handles both positive and negative values of the error [35] and consequently guarantees the precision. The quadrotor attitude control needs good precision and response thus highlighting the need of fitness functions [47]. The functions $ISE, ITAE$ and $ITSE$ are given in (17), (18) and (19) respectively:

$$ISE = \int_0^T e^2 dt \quad (17)$$

$$ITAE = \int_0^T (t|e|)dt \quad (18)$$

$$ITSE = \int_0^T te^2 dt \quad (19)$$

In order to provide a thorough comparison, each fitness function is performed for the three algorithms under discussion i.e. PSO, CS and BAT.

3 Control and optimization algorithms design

3.1 Attitude Fuzzy Controller

The main objective, here, is to design a fuzzy optimized controller for the quadrotor attitude stabilization. The idea of applying the attitude control law on the quadrotor is to set the first control U_1 as a constant value and apply other commands (U_2, U_3, U_4) so as to let (φ, θ, ψ) follow their desired values $(\varphi_d, \theta_d, \psi_d)$ (See Figure 2).

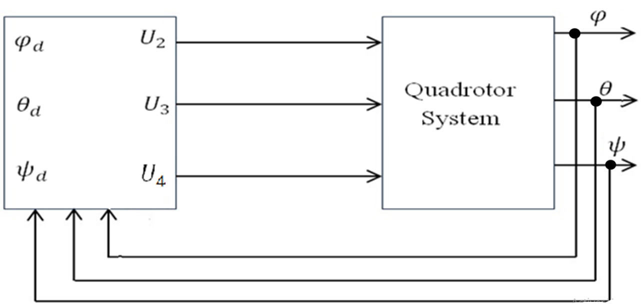
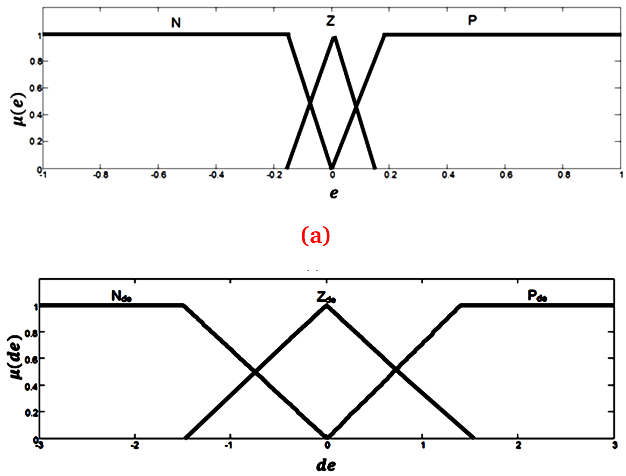


Figure 2. Fuzzy attitude quadrotor control

We considered fuzzy zero order Takagi Sugeno controllers [2]. These controllers are composed of two inputs and one output. The inputs are the error in angular position and its time derivative while the output consists of the velocities that must be applied to each motor of the quadrotor system. The choice of the MFs is based on a typical step response of a second order linear system. The error is divided in three cases i.e. error is positive, negative and around zero. The change in error is related to the tangent under the step response and is also described by three angles positive, negative and around zero (Figure (3). The input and output variables are defined by eleven (11) fuzzy values called: $\{P(Positive)$, $Z(Zero)$, and $N(Negative)\}$ for the error and $(Positive), (Zero), and (Negative)$ for the time derivative of the error and $\{NG(LargeNegative)$, $NM(MediumNegative)$, $Z(Zero)$, $PM(MeanPositive)$, $PG(LargePositive)\}$ for the output. The distributions of the MFs on universes of discourse for all inputs-output are normalized between $[-1, 1]$ except the error change which is between $[-3, 3]$. The used MFs as well as their distributions in universes of discourse are presented in Figures 3-4.



(a)

(b)

Figure 3. MFs of: (a) Error e (b) Derivative of the error de

The rules base:

Two main conditions have been considered to design the rules base: the completeness and the consistency:

- Completeness: 'Rule base' of a fuzzy system is called complete if, for each input vector, there is at least one active fuzzy rule. To ensure this property, the MFs must cover all possible ranges of the input variables. We have used uniformly distributed triangular MFs which satisfy this property.

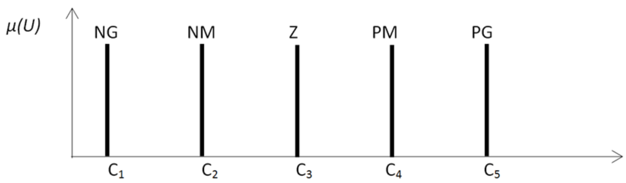


Figure 4. The output MFs

- Consistency: 'Rule base' of a fuzzy system is considered to be inconsistent, if there are two fuzzy rules with the same premise but leading to different conclusions. This consistency avoids contradictions in a 'rule base'. The design rules base used in this research is given in Table IV as an anti-diagonal and anti-symmetrical form. The rule base has been taken based on the relevance with a typical response of a second order system. The essential parameters of this response have been considered which include; start of the response, response around setting time and peak time and response in the vicinity of rise time and overshoot. Moreover, the response around the steady state error is also taken into account.

Figure 5 elaborates the design of the fuzzy rule base given in Table IV. The error indicates presents the difference between the desired response and the measured one. The change in the error represents the velocity and the direction and can bear positive or negative sign. The main idea involves a representative set of situations taken under typical step response of a second order system. Similar approach has been adopted in [50] to design FLC in MATLAB/Simulink environment. Since the error and its derivative can have three values (positive, negative and zero), we have considered a simple rule base corresponding to these three situations for both the error and its change. The output of the fuzzy controller is defined by five MFs in order to have good performances particularly in terms of precision in the response. The overall concept of the chosen rule base is summarized by nine possible combined situations for the error and its change as highlighted in Figure 5.

- If the error is around zero and the error change is also zero, this indicates that the system attained the desired response (situation 9 in Fig. 5).

- If the error or its change is positive and the other input is negative (situations 3 and 7 in Fig. 5), the system is near to the desired response with a positive or a negative velocity. The system is solely guided by its actual inertia to maintain the controlled angle near to the desired one.

- The change in the error is zero in the situations 4, 7 and 9 (see Fig. 5), the control action in this case is related only to the fuzzy value of the error. Due to this reason, if the error is positive, then the system does not always reach the desired angle since a positive control signal is required. On the other hand, if the error goes negative indicating an overshoot situation, then it is important to apply a negative command signal to decrease the error.

- If the error is around zero and the change in error is positive corresponding to the situation 2 in Fig. 5, the system must be decelerated slowly with the negative medium command. On the other hand, if the change in error is negative (situation 6 in Fig. 5), the error becomes positive consequently requiring a positive control action to stabilize the system around the desired angle.

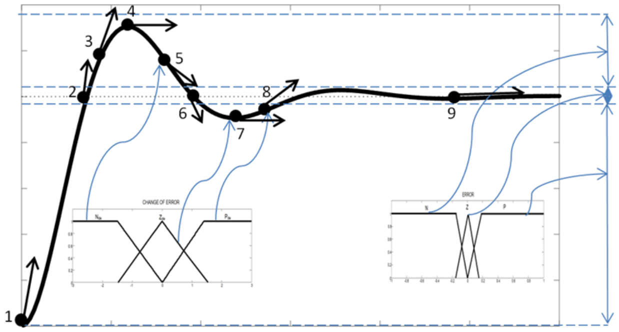


Figure 5. Membership and rules determination

- If the error and it change have their maximum positive or negative values (situations 1 and 5 in Fig. 5), the required actions of applying signals of higher amplitudes. The positive values of error and change in error indicate the situation where the system is very far from the desired angle and the quadrotor should be subjected to a strong positive signal. In contrast, if the error and its change have negative fuzzy values, the system exhibits an overshoot behavior and should be slowed by applying an appropriate signal.

It is evident that if the number of rules is small, there is no necessity to optimize them. The aggregation/implication method of *Zadeh* min-max is used in the present research. Since each rule has a numerical conclusion, so the time consumed by defuzzification procedure is considerably reduced comparing with a *Mamdani*-type controller. In fact, the output of the Takagi *Sugeno* fuzzy controller of zeroth order is given by (20):

$$y(x) = \frac{\sum_{k=1}^{N} \mu_k(x) f_k(x)}{\sum_{k=1}^{N} \mu_k(x)} \tag{20}$$

where $\mu_k(x)$ is the degree to which the i^{th} rule matches the input, N is the number of active outputs and $f_k(x)$ is the value of the control output.

		Error (e)		
		N	Z	P
Error Change	N_{de}	NG	NM	Z
	Z_{de}	NM	Z	PM
	P_{de}	Z	PM	PG

Table 4. Controller rules base

3.2 Metaheuristic Optimization of Fuzzy Controllers

The proposed optimization approach consists of finding the most optimal distribution of the outputs fuzzy controllers. Figure 6 illustrates the proposed optimization strategy to optimize the distribution of the MFs outputs $C1, C2, C3, C4, C5$. The key points are summarized below:

- The fewer number of rules requires no structural optimization. Also, optimization of a small number will not give significant improvement in terms of performance.

- The optimization of output MFs allows investigating the role of optimized distribution of the output singletons MFs to improve the control performance.

- The modeling of input dynamics by two fuzzy variables (error and its change) is supposed to be sufficient, hence; the use of singletons MFs for the outputs generates limited performance and robustness. Therefore, the optimization of the output allows obtaining best performance in terms of control energy.

- The minimization of the number of parameters to be optimized is also an important reason given the fact that the processing time is a function of specifications of the equipment available to carry out the optimization procedure.

The proposed optimization approach can be briefly outlined as:

- In order to calculate the FLC, for the first iteration, all inputs MFs are fixed as shown in Fig. 3. The output MFs $\{C1, C2, C3, C4, C5\}$. of the controller have a random distribution for each controller Roll, Pitch and Yaw (Fig. 4),

- With this first output random distribution, the control law is calculated and applied to the quadrotor. Three fuzzy control signals u_1,u_2,u_3, u_4 are calculated, u_1 has a fixed value,

- Control signals u_1,u_2,u_3, u_4 are sent to the quadrotor actuators and an actual response (ϕ, θ and ψ) is obtained,

- The actual response obtained from the quadrotor is compared with the desired one(ϕ_d,θ_d and θ_d) to calculate the error and consequently, an objective function i.e. the criterion to be minimized (17), (18) or (19), is numerically evaluated,

- The output of the objective function used by the optimization algorithms BAT, PSO, CS, PEO is then modified and is subsequently taken as a new distribution of the $\{C1, C2, C3, C4, C5\}$,

- This operation is repeated until the stop condition (number of iterations) is reached. Results of this algorithm is the best $\{C1, C2, C3, C4, C5\}$ outputs MFs.

4 Application of algorithms

The quadrotor model presented in Subsection II.2 is used in simulation to test and characterize the performance of different algorithms. In order to optimize the fuzzy controller based on the PSO, BAT and CS algorithms, we must first define the inputs and outputs of these algorithms. The input is unique for all the cases and is considered in the criterion to be minimized. Three criteria presented in section III (ISE, ITAE and ITSE) are tested. The output is the distribution of singletons of each fuzzy controller that corresponds to the control of an angle of the quadrotor. The choice of the parameters of the metaheuristic algorithms is an essential task for the improvement of the algorithmic performance. However, this choice requires several experiments because each problem requires a setting of parameters depending on the complexity being addressed and the mean calculation. In this paper, most of the parameters are chosen experimentally. All algorithms start with random distributions of the MFs for each output. The performance of the optimized controllers is compared with two supplementary controllers, which are a PD controller and an integral backstepping controller reported in [51].
The simulation results of the application of PSO, BAT and CS algorithms on the fuzzy controllers for the quadrotor attitude stabilization mainlly include the plots of angles; roll (ϕ), pitch (θ) and yaw (ψ). The initial values of the roll, pitch and yaw angles are set to be 0.5 radians, 0.5 radians and 0 radians respectively, while the desired angles are 0 radians for roll and pitch and 0.5 radians for yaw. The simulation is performed using an intel core i7-2670QR labtop with 2.2 GHz CPU speed and 8 GB RAM. The distributions of output MFs are obtained by minimizing the criteria ISE, ITAE and ITSE, using the three algorithms. For each criterion, the corresponding simulation statistics are presented. In the second part, we will present the curves of the angles obtained by the application of these distributions on the fuzzy controllers.Since the range of the error taken into account in the fuzzy controllers is [-1,+1] rad, the constraints considered in the optimization are :$-1rad \le (\phi, \theta) \le 1$ rad and $-0.5rad \le \psi \le 1.5$.

4.1 Parameters Selection for PSO, BAT and CS

- **PSO:** The optimal distribution of singletons is calculated and is recorded for each iteration using (6) and (7). The acceleration coefficients c1 and c2, generally, range from 0-2. In this work, they are chosen as 2. In order to achieve maximum acceleration, r1 and r2 are considered to have random values with uniform distribution. A population size of 20 particles is used and the number of iterations is selected to be 40.

- **BAT:** BAT algorithm given by (14-16) is characterized by some parameters to run such as; f_{min}, f_{max}, initial amplitude $A(0)$, initial frequency $r(0)$, number of bats and number of iterations. These used parameters are set as follows: $Numberof\,bats = 20$, $A(0) = 0.9$, $r(0) = 0.9$, $\gamma = \sigma = 0.9$ and number of iterations $= 40$. The value of f_{min} has been chosen as zero whereas the choice of $f_{max} = 0.5$ is dictated by iteration-based simulation tests. This choice is adequate keeping in view the interval between f_{min} and f_{max} in the present control method. The change in the distribution at each iteration in-

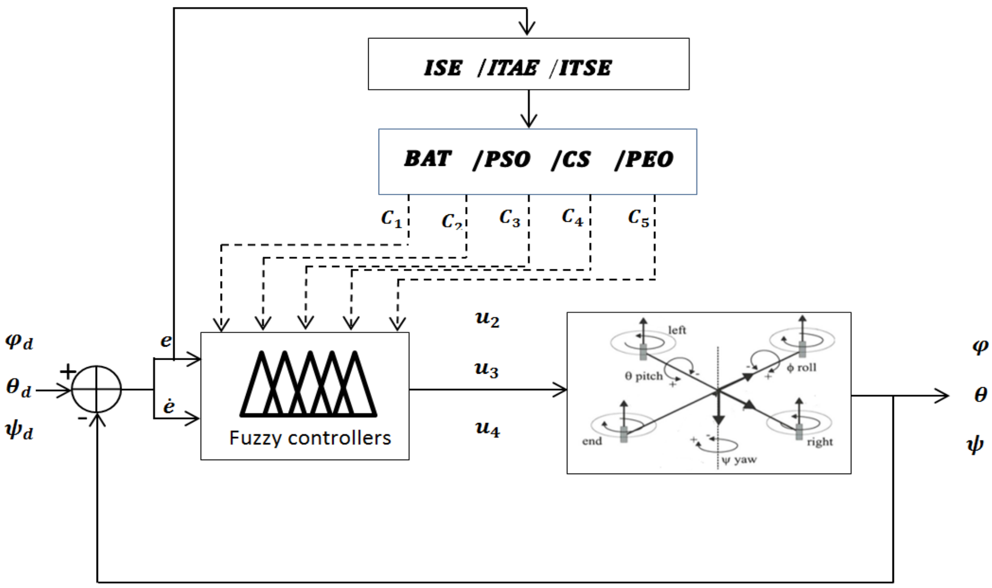


Figure 6. Proposed optimization approach

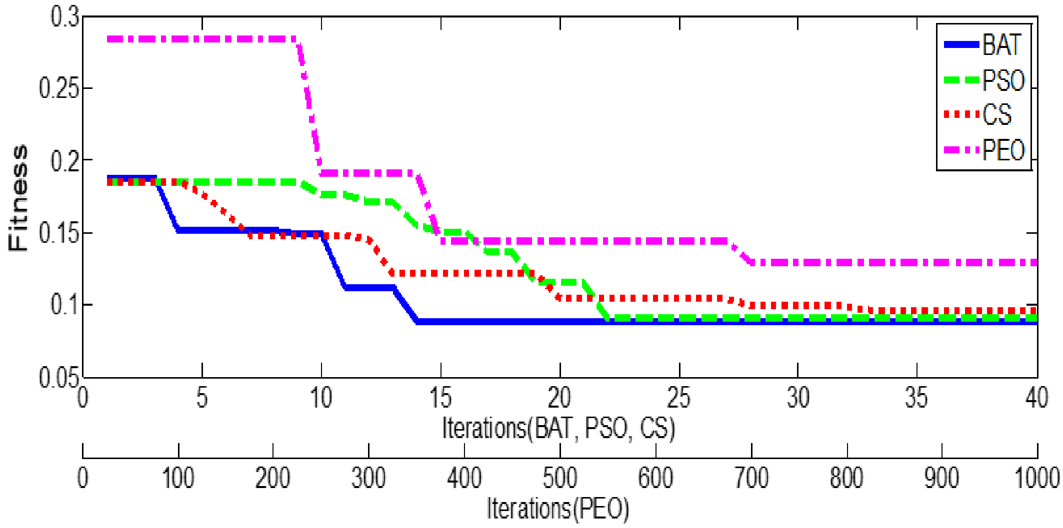
fluences the search procedure. The initial amplitude $A(0)$ decreases with each iteration and is chosen as its standard value of 0.9. The frequency r must have an initial value, which in most cases is 0.9. We have chosen a population size of 20 bats.

- **CS:** The CS parameters are selected arbitrarily. The distribution singletons at each iteration are obtained using Lévy's flight. The initial population contains 20 cuckoos and the algorithms is executed for 40 iterations during each run. The coefficient of the distribution of Lévy's flight follows the range $0 < \beta < 2$, for this reason we have selected $\beta = 1.5$.
- **PEO:** Population Extremal Optimization (PEO) is based on Lévy's flight method and hence uses the CS parameters. PEO is a local search algorithm in which the number of iteration is equal to 1000.

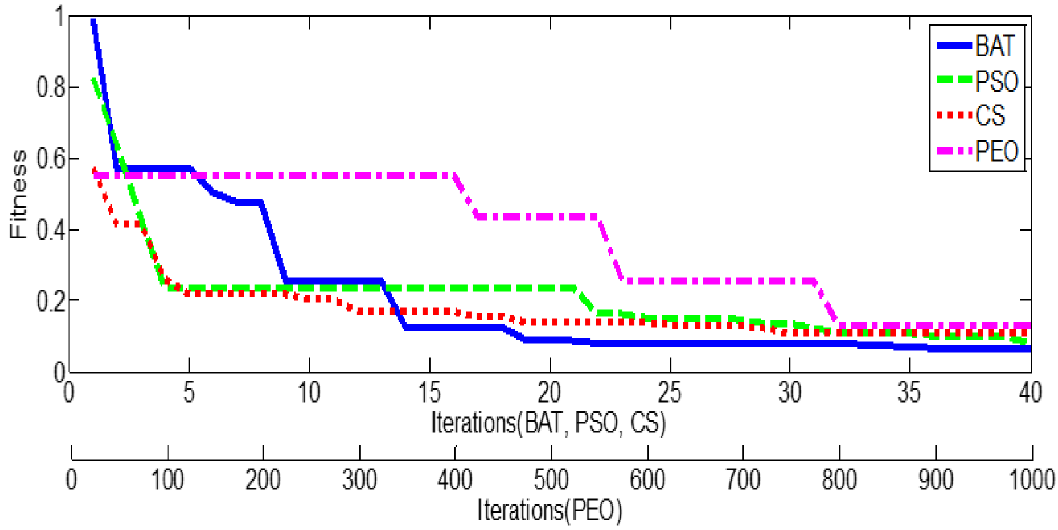
4.2 Application of the Algorithms for the ISE, ITAE and ITSE Optimization

The optimized controller is applied to the quadrotor in regulation mode. Simulation results are obtained after performing five comparative tests using each algorithm to obtain the best results in each case as shown in Figure 7. Five experiments are carried out to avoid the local minimum. The figure illustrates the time evolution of the ISE, ITAE and ITSE criteria corresponding to the three algorithms PSO, CS and BAT. In order to provide compact comparison, these algorithms are compared with a local search method called Population Extremal Optimization (PEO) [48]. In general, extremal optimization principle consists of the elimination of the bad local variables using a generation of new solution by mutation for the selected bad variables and encourage the good ones [49]. The objective of using the PEO algorithm is to compare a local search method with global search methods like PSO, CS and BAT. Specifically, the number of iterations has been selected as 40 for PSO, CS and BAT and 1000 for PEO given the fact that the later most is a local search algorithm. The results showed that BAT algorithm converged in 22^{nd} , 27^{th} and 24^{th} iterations for ISE, ITAE and ITSE respectively in regulation mode (Figure 7). Also, in trajectory tracking mode, BAT converged in 30^{th} , 20^{th} and 15^{th} iterations for ISE, ITAE and ITSE respectively.

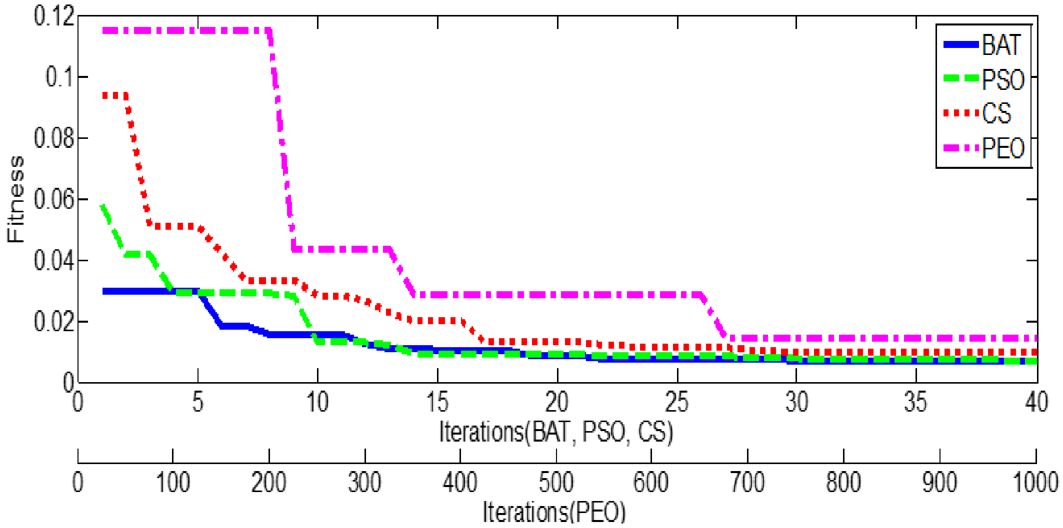
Table V and Table VI present the optimization results after performing PSO, BAT and CS algorithms for three objective functions (ISE, ITAE and ITSE). The optimization experiments have been conducted in regulation mode (Table V) and trajectory tracking mode (Table VI) without involving perturbation terms given in (2). The superior values are illustrated in bold form. Two simulation experiments have been selected which present the best and worst values of the objective functions (ISE, ITAE and ITSE). These values are superior in case of BAT algorithm compared with those in PSO and CS algorithms. Since PEO is a local search algorithm, it is necessary to have an adequate number of iterations (1000 iteration in this case) for its convergence. The average computation time is minimal for BAT algorithm compared to CS, PSO and PEO counterparts. For controlling the quadrotor system, the fitness functions are complex due to couple of reasons; Firstly, the fuzzy Takagi-Sugeno controller (19) is a function of the error and its variation associated with each other in a complex relationship. Secondly, the optimization consists of simultaneously determining 15 parameters (five parameters for each controller), which is not trivial. Moreover, the function complexity permits finding various combinations of parameters corresponding to a same relative fitness value. As confirmed through literature [34], it is possible to find different values of parameters corresponding to the same fitness functions. It can be seen in Table IV and V that large differences in the values of parameters lead to similar values of fitness which suggests the flatness of the criterion function.



(a)

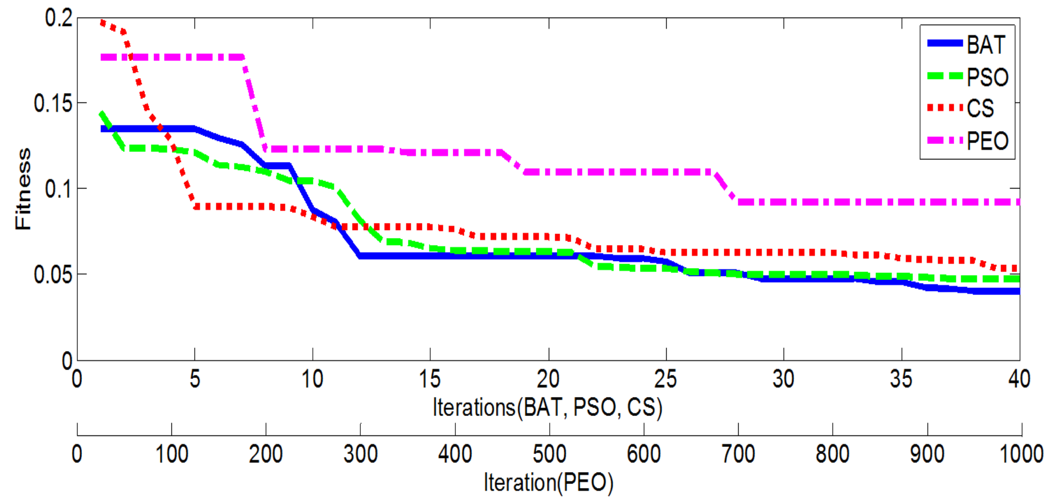


(b)

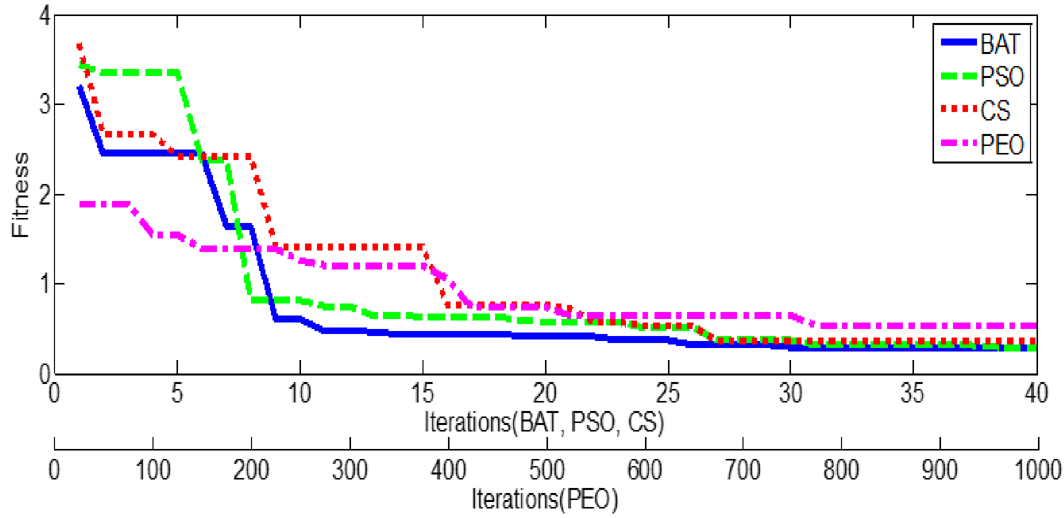


(c)

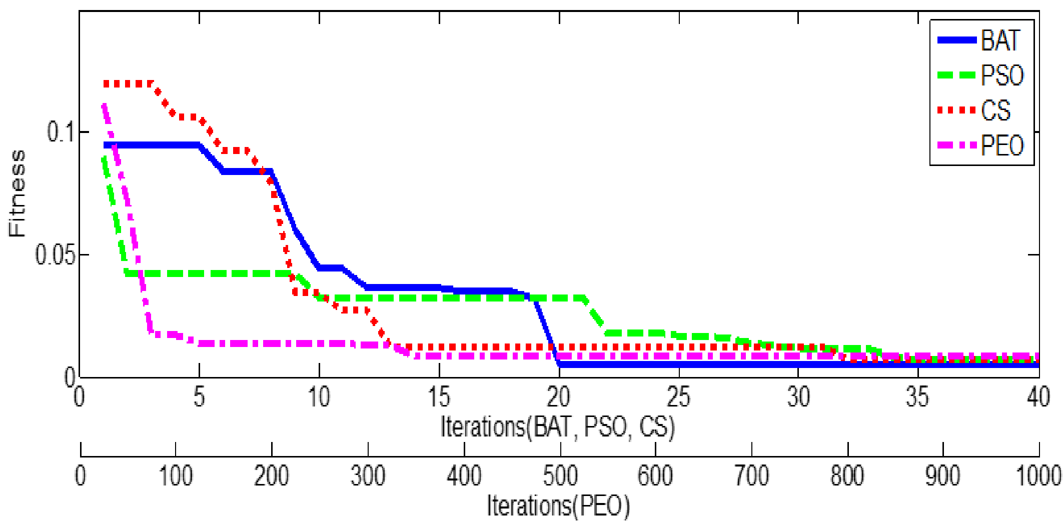
Figure 7. Time evolution of criteria corresponding to BAT, PSO, CS and PEO algorithms: (a) ISE (b) ITAE (c) ITSE



(a)

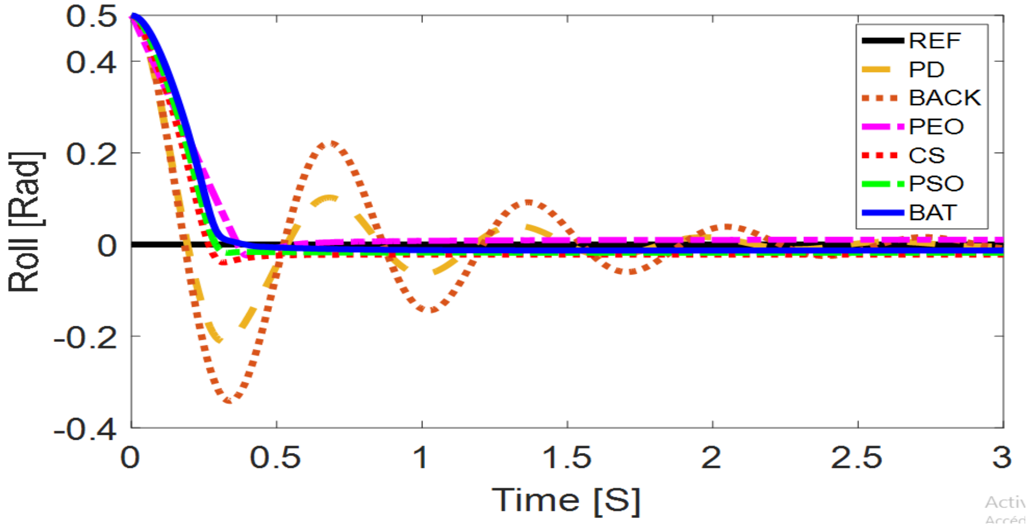


(b)

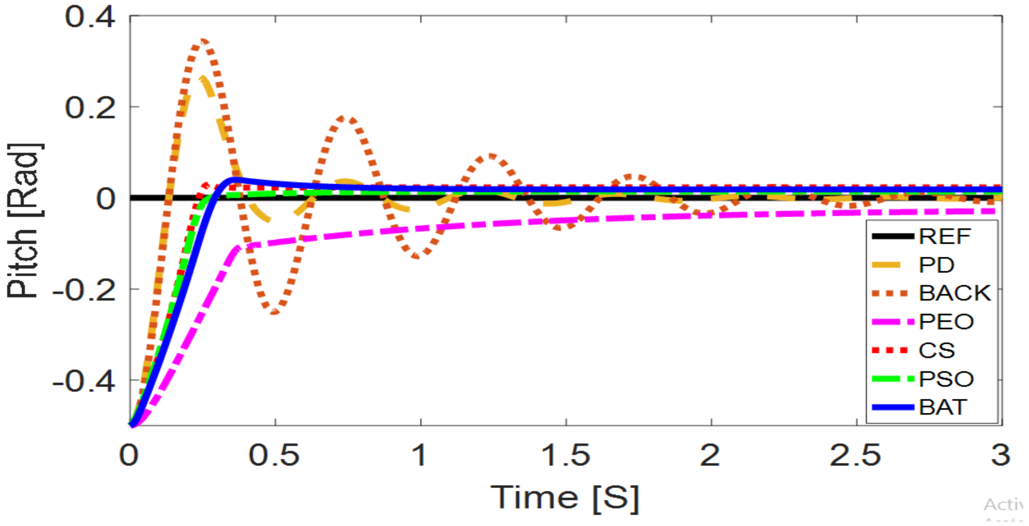


(c)

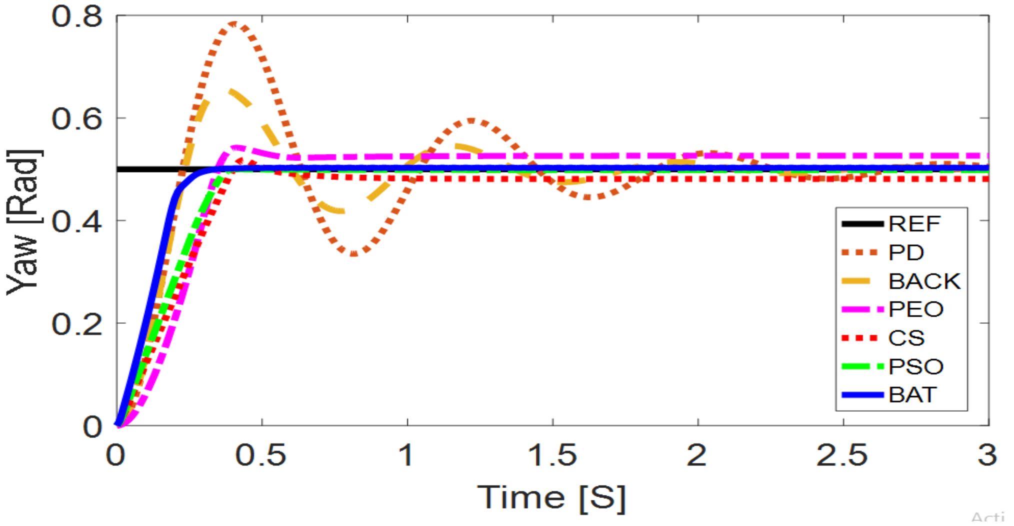
Figure 8. Time evolution of criteria corresponding to BAT, PSO, CS and PEO algorithms in trajectory tracking mode: (a) ISE (b) ITAE (c) ITSE



(a)

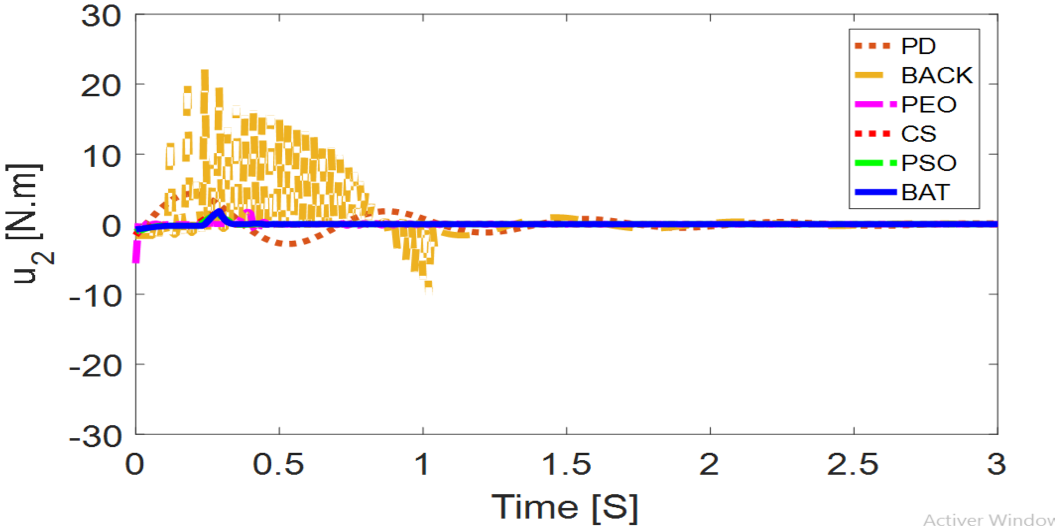


(b)

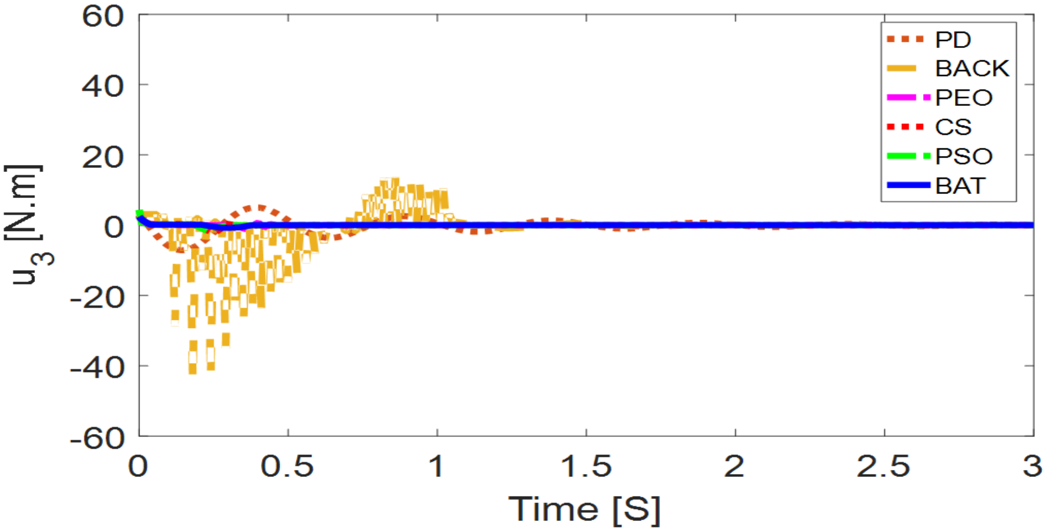


(c)

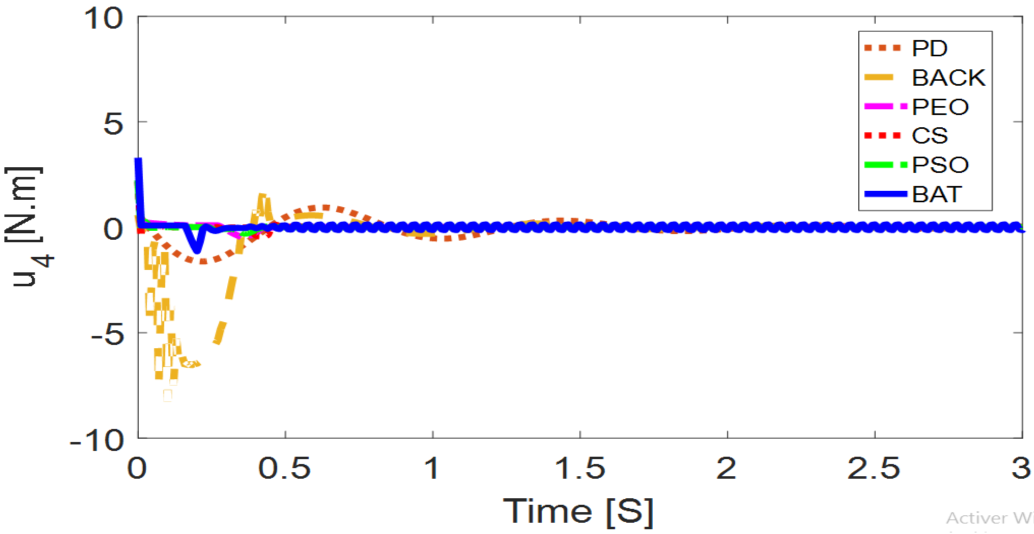
Figure 9. Responses of quadrotor angles using output MFs optimized by ISE: (a) ϕ (b) θ (c) ψ



(a)

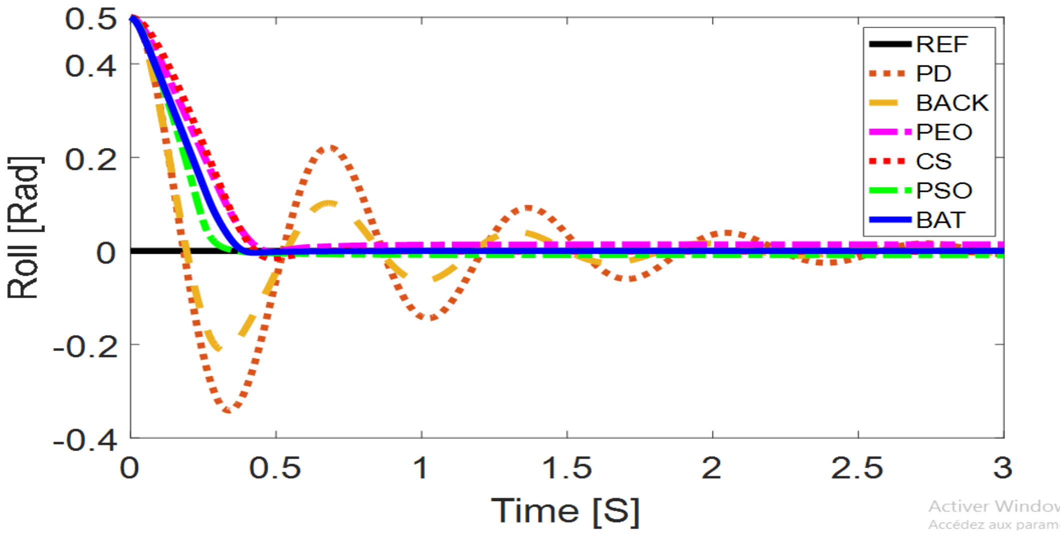


(b)

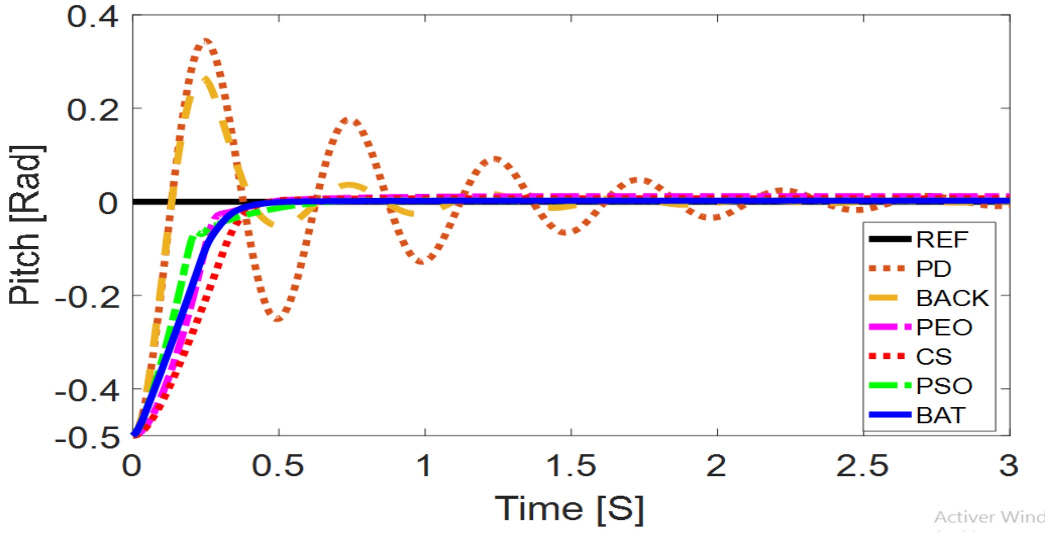


(c)

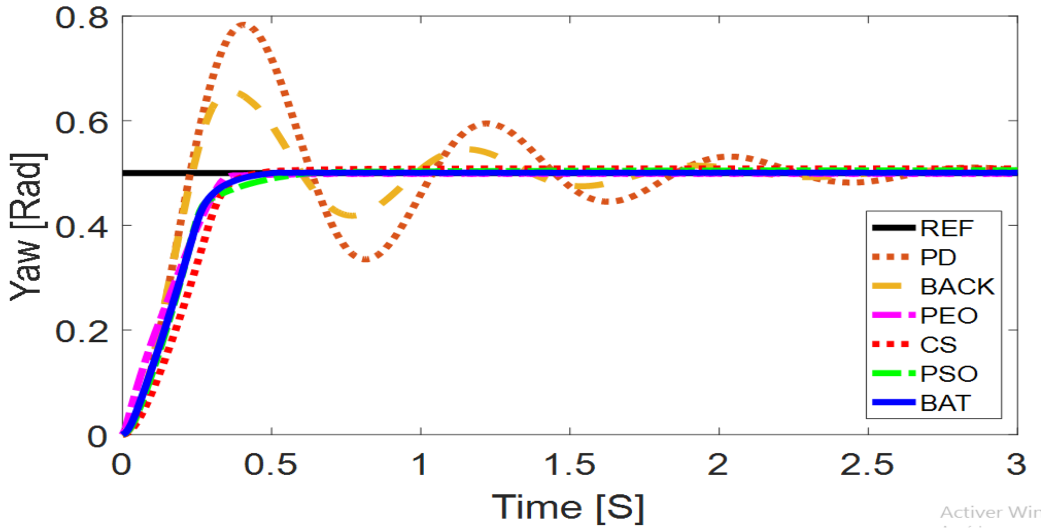
Figure 10. Control efforts of quadrotor angles using output MFs optimized by ISE: (a) ϕ (b) θ (c) ψ



(a)

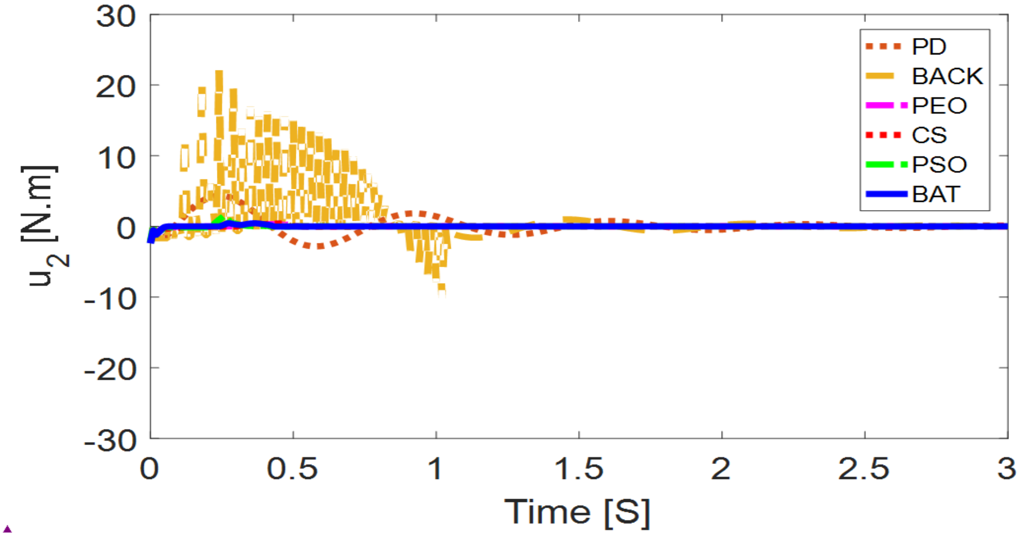


(b)

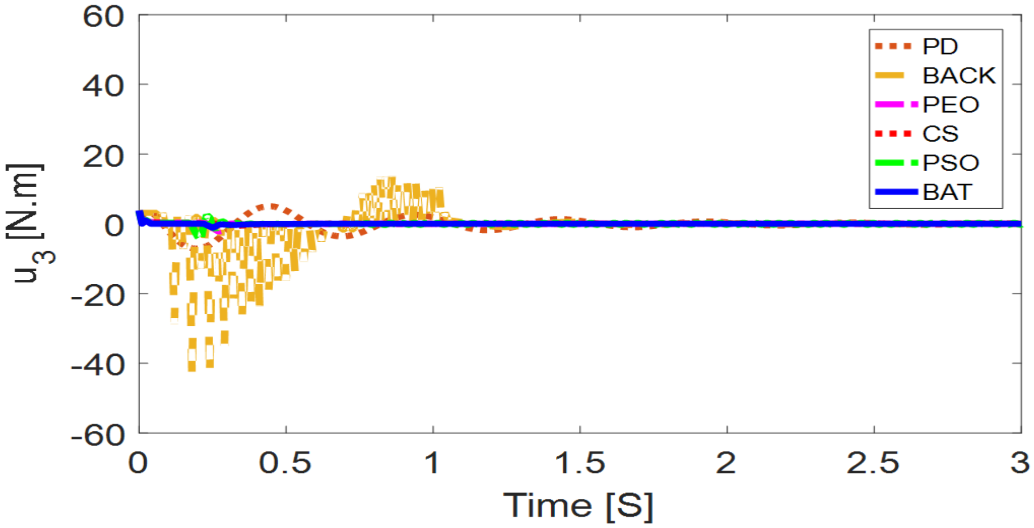


(c)

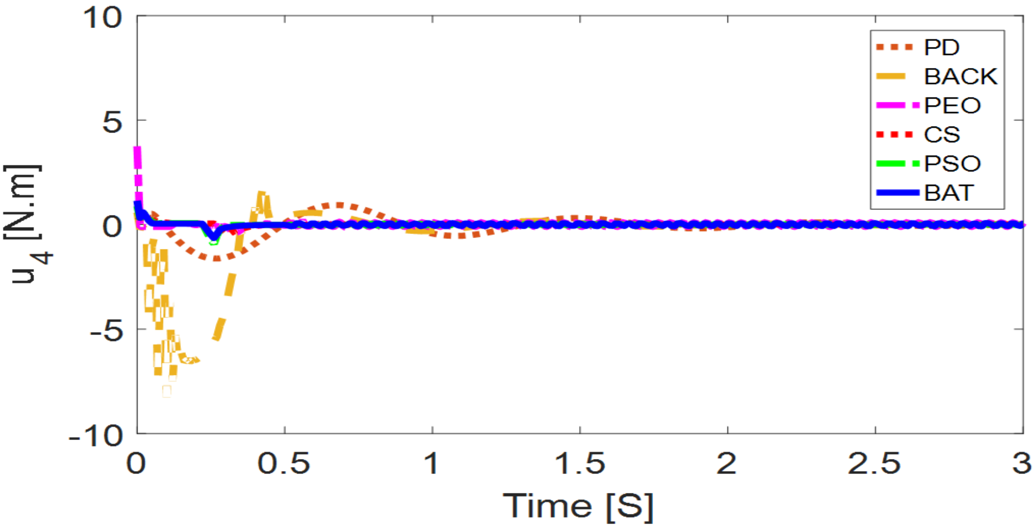
Figure 11. **Responses of quadrotor angles using output MFs optimized by ITAE: (a) ϕ (b) θ (c) ψ**



(a)

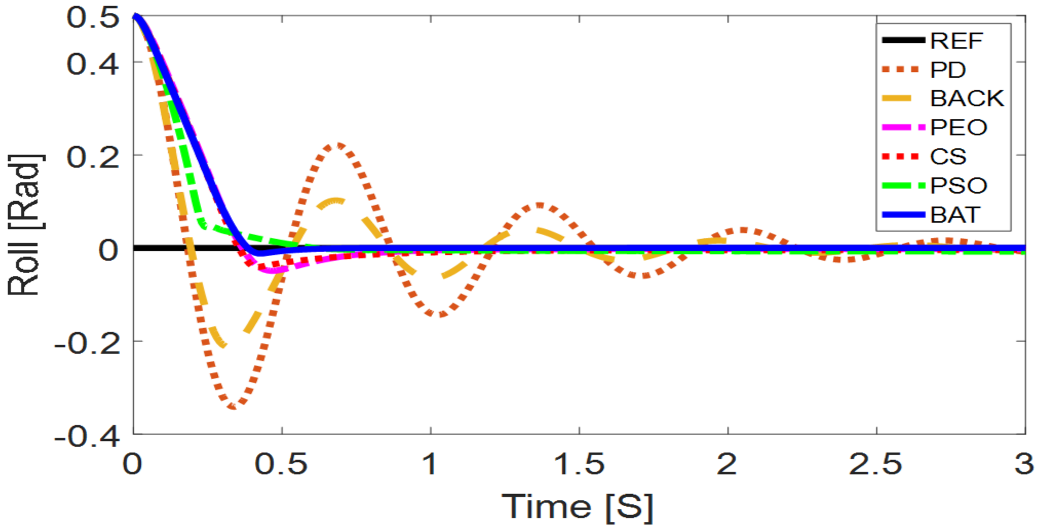


(b)

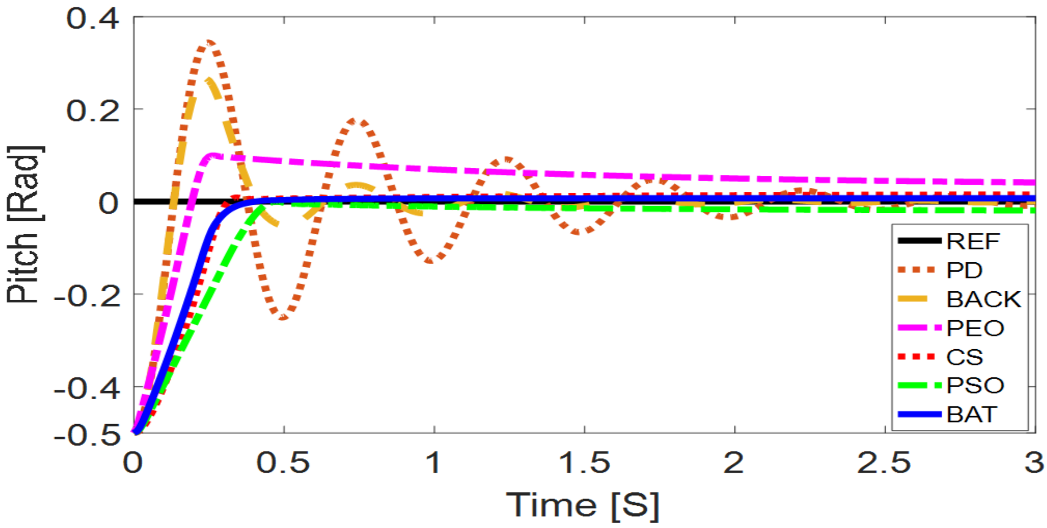


(c)

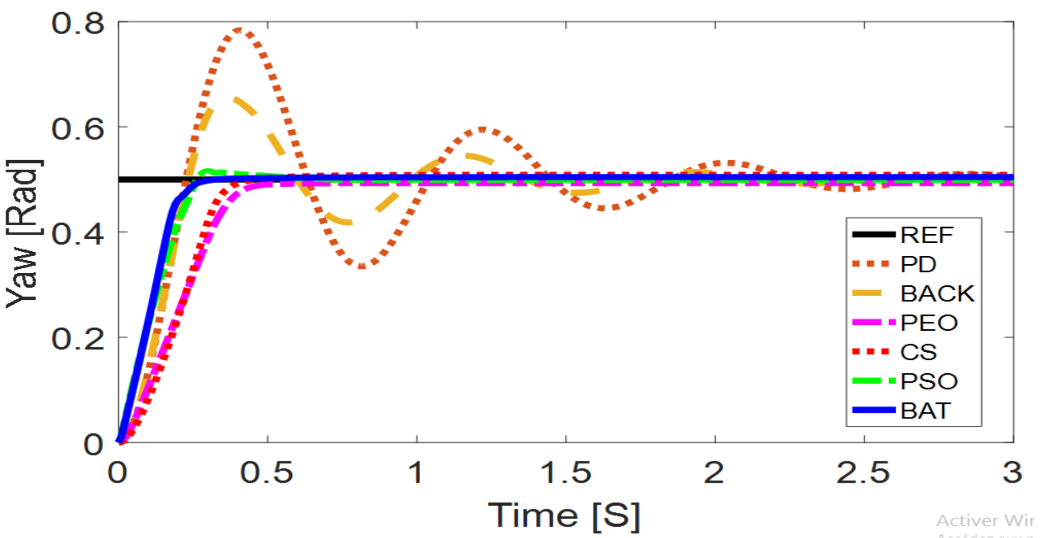
Figure 12. **Control efforts of quadrotor angles using output MFs optimized by ITAE: (a) ϕ (b) θ (c) ψ**



(a)



(b)



(c)

Figure 13. **Responses of quadrotor angles using output MFs optimized by ITSE: (a) ϕ (b) θ (c) ψ**

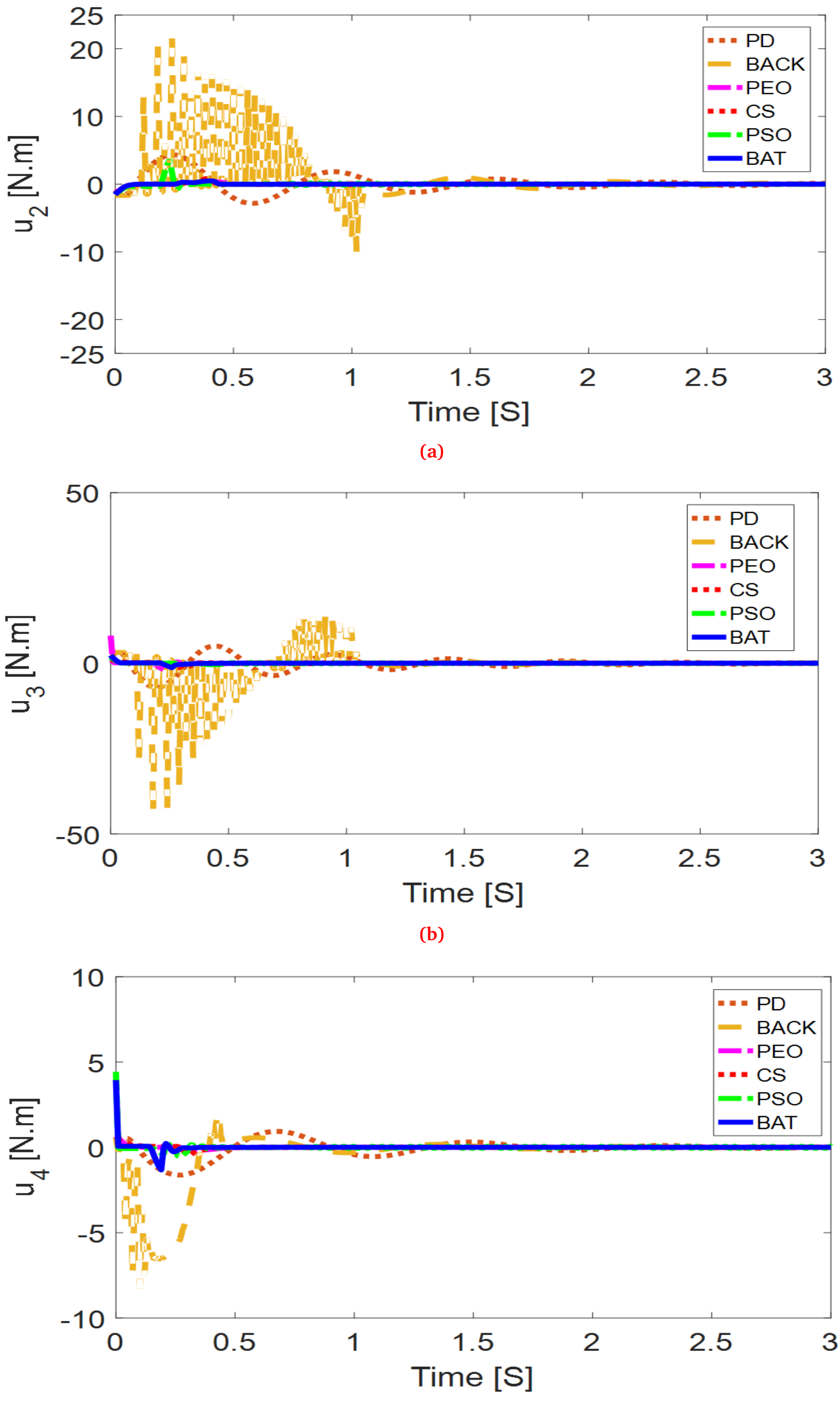


Figure 14. Control efforts of quadrotor angles using output MFs optimized by ITSE: (a) ϕ (b) θ (c) ψ

ISE										
		N_g	N_m	Z	P_m	P_g	Best	Medium	Worst	Time(s)
BAT	ϕ	-2.6575	-0.7378	-0.2320	2.4333	4.5670	**0.0890**	0.0895	0.1111	**1446**
	θ	-1.9784	-0.8416	0.1180	2.4920	2.7363				
	ψ	-6.3185	-1.8812	0.0821	3.2954	8.3622				
PSO	ϕ	-1.1419	-1.0930	-0.2199	1.6349	3.2869	0.0910	0.1034	0.1176	1622
	θ	-4.1912	-2.1897	0.2275	4.3828	5.2394				
	ψ	-4.3356	-1.0985	-0.0249	2.2204	3.8422				
CS	ϕ	-1.7800	-1.5771	-0.2291	1.3294	3.0320	0.0952	0.1092	0.1294	2806
	θ	-6.8459	-1.6631	0.3046	2.8121	3.2061				
	ψ	-1.7119	-1.1955	-0.3108	2.1426	2.1660				
PEO	ϕ	-9.0438	-5.5675	0.4179	0.7069	11.622	0.1296	0.1432	0.1466	1782
	θ	-9.6840	-8.0357	-0.1388	0.7380	9.106				
	ψ	-1.2958	-0.4464	0.0946	0.2487	9.6693				
ITAE										
		N_g	N_m	Z	P_m	P_g	Best	Medium	Worst	Time(s)
BAT	ϕ	-2.9742	-2.3939	0.0006	1.5876	3.2798	**0.0660**	0.0839	0.1083	**1398**
	θ	-4.7722	-3.2269	0.0517	3.8716	7.8843				
	ψ	-19.1999	-1.3784	0.0464	1.1444	2.214				
PSO	ϕ	-3.6573	-2.2501	-0.1389	2.2450	3.6834	0.0819	0.0844	0.0937	1603
	θ	-11.547	-8.6925	0.3164	3.5046	4.9529				
	ψ	-5.5697	-1.8934	0.0700	0.8830	7.1436				
CS	ϕ	-1.1947	-0.6325	-0.0077	0.8444	2.7686	0.1077	0.1302	0.1475	3064
	θ	-1.6655	-1.4598	0.0780	0.6682	1.2969				
	ψ	-0.9497	-0.8636	0.0534	0.3954	1.1319				
PEO	ϕ	-1.2698	-1.0269	0.1051	1.0841	8.7210	0.1340	0.1388	0.1582	1695
	θ	-6.2907	-3.4333	0.2998	0.7615	2.0024				
	ψ	-12.259	-1.5727	-0.1148	3.7499	8.928				
ITSE										
		N_g	N_m	Z	P_m	P_g	Best	Medium	Worst	Time(s)
BAT	ϕ	-1.6931	-1.5276	0.0037	1.0686	3.8598	**0.0064**	0.0093	0.0099	**1430**
	θ	-2.7691	-2.2412	0.1156	2.3560	4.5540				
	ψ	-2.2445	-1.9483	0.0584	3.9401	4.8000				
PSO	ϕ	-1.8713	-1.4582	-0.3217	5.8608	10.491	0.0071	0.0099	0.0106	1588
	θ	-9.1344	-1.0687	-0.2727	1.706	12.207				
	ψ	-1.1258	-0.8304	-0.0716	4.4275	8.3682				
CS	ϕ	-4.2465	-1.4388	-0.0127	0.4265	4.2587	0.0092	0.0145	0.0199	3126
	θ	-12.221	-1.5471	0.1955	1.0594	2.899				
	ψ	-25.560	-0.0818	-0.0167	1.3547	1.5723				
PEO	ϕ	-6.1276	-1.3008	-0.0079	0.5691	0.9104	0.0085	0.0166	0.0179	1706
	θ	-3.3234	-0.5194	0.1562	8.1039	10.514				
	ψ	-3.6146	-0.8501	-0.0330	0.6688	1.950				

Table 5. Optimal values of tuning parameters of the FLC with BAT, PSO, CS and PEO algorithms in regulation mode

Table VIII presents results of some performance parameters such as, Rise Time (RT), Settling Time (ST), Over-Shoot (OS) and Steady State Error (SSE) achieved using the optimized controllers.

From values of %OS given in Table VII, we can notice that the maximum values are offered by the PEO and CS algorithms, however these values are still in acceptable range. Therefore, further performance comparison includes calculations of mean values of other transient parameters and SSE as given in (21-23) .

$$RT_{mean} = \frac{(RT_{ISE} + RT_{ITAE} + RT_{ITSE})}{3} \tag{21}$$

$$ST_{mean} = \frac{(ST_{ISE} + ST_{ITAE} + ST_{ITSE})}{3} \tag{22}$$

$$SSE_{mean} = \frac{(SSE_{ISE} + SSE_{ITAE} + SSE_{ITSE})}{3} \tag{23}$$

where RT_{mean}, ST_{mean} and SSE_{mean} are respectively the mean values of performance metrics RT, ST and SSE using ISE, ITAE and ITSE. Obtained results are summarized in Table IX.

It can be seen that BAT algorithm offers the best performance (shown in bold) and CS and PSO cannot compete

		N_g	N_m	Z	P_m	P_g	Best	Medium	Worst	Time(s)
ISE										
BAT	ϕ	-4.2380	-4.0518	-0.1773	4.9948	7.8007	**0.0403**	0.0666	0.0865	**1723**
	θ	-5.6264	-1.3738	-0.0859	4.8518	30.2715				
	ψ	-8.1547	-1.2549	0.0274	4.3221	12.9975				
PSO	ϕ	-10.686	-3.6258	-0.0302	1.4376	3.9623	0.0475	0.0603	0.0909	1904
	θ	-3.6467	-3.1171	-2.2500	9.3346	18.3859				
	ψ	-10.4762	-4.7786	0.1324	3.0861	8.7258				
CS	ϕ	-6.2359	-5.8333	0.3056	1.1822	7.6763	0.0535	0.0791	0.0896	3925
	θ	-4.5420	-1.0922	-0.1348	0.5412	4.3258				
	ψ	-2.0409	-1.3905	0.1516	2.5016	3.1208				
PEO	ϕ	-15.263	-3.9231	0	4.3264	10.200	0.0921	0.107	0.1190	2210
	θ	-0.4698	-0.4032	0.0867	0.1849	2.9273				
	ψ	-5.3304	-1.0375	-0.2368	0.3990	10.1084				
ITAE										
		N_g	N_m	Z	P_m	P_g	Best	Medium	Worst	Time(s)
BAT	ϕ	-10.9092	-1.8654	-0.2535	7.6460	31.736	**0.2765**	0.3746	0.3947	**1800**
	θ	-3.3619	-2.2317	-0.2550	6.2953	8.187				
	ψ	-6.7041	-5.0994	0.0699	2.0404	2.0991				
PSO	ϕ	-21.9113	-3.4139	0.0360	1.0073	1.4765	0.2863	0.3916	0.4195	1896
	θ	-6.9647	-5.8079	0.0372	1.8132	2.7990				
	ψ	-11.1037	-1.9044	0.1115	0.3171	2.2307				
CS	ϕ	-12.6897	-7.5886	0.2216	1.8588	7.2020	0.3579	0.4960	0.5167	3621
	θ	-4.1795	-0.3651	-0.0768	2.4803	3.1174				
	ψ	-1.2485	-1.2349	-0.2516	4.0880	9.8809				
PEO	ϕ	-4.3315	-2.5050	0.0201	0.5903	0.7773	0.5458	0.5888	0.6548	2236
	θ	-1.7094	-0.2596	-0.0442	0.4995	4.5958				
	ψ	-1.9557	-0.1520	-0.0022	0.9450	2.6988				
ITSE										
		N_g	N_m	Z	P_m	P_g	Best	Medium	Worst	Time(s)
BAT	ϕ	-11.9873	-6.5219	0.2100	3.5536	11.693	**0.0051**	0.0074	0.0085	**1900**
	θ	-11.2445	-7.8307	0.8329	1.3004	5.0401				
	ψ	-6.3045	-0.1158	-0.0212	2.4890	8.0757				
PSO	ϕ	-1.3645	-0.8586	-0.0310	1.6132	2.6919	0.0069	0.0086	0.0090	2274
	θ	-6.8461	-0.7587	-0.0316	0.5797	15.5532				
	ψ	-1.3103	-0.4037	-0.0174	2.2969	4.5436				
CS	ϕ	-2.6687	-1.7225	0.0519	0.3826	1.0869	0.0073	0.0079	0.0100	3852
	θ	-3.0136	-2.3838	0.0544	1.4889	1.4970				
	ψ	-1.6863	-0.7100	0.0464	1.9283	12.2456				
PEO	ϕ	-1.1979	-0.9245	0.0284	0.7254	0.9516	0.0081	0.0108	0.0127	2409
	θ	-0.3742	-0.3283	-0.0231	0.9583	1.4421				
	ψ	-1.4376	-0.8663	-0.0085	1.1838	1.6208				

Table 6. Optimal values of tuning parameters of FLC with BAT, PSO, CS and PEO algorithms in trajectory tracking mode

with it in most cases. The best results offered by BAT algorithm are more evident in case of ITAE, which can optimize the transient behavior as well as steady state response. PSO is the second best optimizer demonstrating best performance in some cases, specially corresponding to ITSE. However, even in this case, the results given by BAT algorithm are very near to that of PSO. In terms of precision in performance, as indicated by the mean values of SSE, BAT algorithm demonstrates superiority compared to the other algorithms closely followed by PSO algorithm. PEO stands as the last algorithm from performances point of view. We can conclude that PEO is not a good choice for this type of fuzzy controllers used in quadrotor systems.

4.3 Robustness Tests

The robustness of the proposed optimized controllers has been tested in the presence of time-varying external disturbances as given in (24).

$$\begin{cases} \zeta(\phi) = -0.06230sin(0.5t + 50) + ds1 \\ \zeta(\theta) = -0.09345sin(0.8t + 30) + ds2 \\ \zeta(\psi) = -0.05600cos(t + 20) + ds3 \end{cases} \tag{24}$$

ISE					
	Algo	RT (sec)	ST (sec)	OS (%)	SSE
ϕ	BAT	0.2054	0.2975	3.0301	0.01455
	PSO	0.1980	0.2747	3.2908	0.01505
	CS	0.1885	0.3340	7.3102	0.02497
	PEO	0.2886	0.3467	4.4422	0.005
θ	BAT	0.2120	0.6715	8.4956	0.01480
	PSO	0.1823	0.2367	3.0683	0.01490
	CS	0.1688	0.2196	6.1501	0.0250
	PEO	1.3642	2.6615	0	0.03021
ψ	BAT	0.1727	0.2387	1.3490	0.0035
	PSO	0.2792	0.3464	0	0.0050
	CS	0.3215	0.4037	0	0.0155
	PEO	0.2291	0.4936	8.9696	0.0250
ITAE					
	Algo	RT (sec)	ST (sec)	OS (%)	SSE
ϕ	BAT	0.2682	0.3460	0.7742	0.0029
	PSO	0.2063	0.2819	1.2305	0.0050
	CS	0.2904	0.3988	3.2354	0.0050
	PEO	0.3067	0.4055	0.4552	0.0150
θ	BAT	0.2533	0.3368	1.0538	0.0043
	PSO	0.2047	0.3202	1.5297	0.0040
	CS	0.2691	0.3811	1.1737	0.0050
	PEO	0.1965	0.2993	1.3847	0.0049
ψ	BAT	0.2340	0.3358	1.0905	0.0007
	PSO	0.2292	0.3885	1.1154	0.0045
	CS	0.2464	0.3502	1.2071	0.0052
	PEO	0.2666	0.3200	0.2184	0.0035
ITSE					
	Algo	RT (sec)	ST (sec)	OS (%)	SSE
ϕ	BAT	0.2729	0.3536	2.2974	0.0045
	PSO	0.1819	0.3853	1.1751	0.0037
	CS	0.2619	0.5979	8.5539	0.0048
	PEO	0.2667	0.6312	9.5000	0.0048
θ	BAT	0.2218	0.3011	1.2129	0.0047
	PSO	0.3208	0.4057	0.1320	0.0050
	CS	0.2093	0.2877	3.0420	0.014
	PEO	0.1812	0.2661	6.2559	0.0063
ψ	BAT	0.1604	0.2243	1.3788	0.0042
	PSO	0.1968	0.2405	2.8998	0.0043
	CS	0.2677	0.3287	3.0186	0.0049
	PEO	0.2981	0.4001	0	0.0050

Table 7. Performance parameters in transient and in steady state using ISE, ITAE and ITSE

where $ds1$, $ds2$ and $ds3$ are uniformly distributed random signals over the interval [-0.05, 0.05]. $\zeta(\phi)$, $\zeta(\theta)$ and $\zeta(\psi)$ are time-varying external disturbances affecting roll, pitch and yaw respectively. The disturbances considered include nonlinear dynamics as well as noise. Figure 15 shows the disturbances corresponding to the three angles for testing and comparing the designed controllers in regulation and trajectory tracking modes. Simulation results in both modes are illustrated in Figures 16-27 with the discussion presented in Section V.

Fitness function	Algorithm	RT (sec)	ST (sec)	SSE
ISE	BAT	**0.196**	0.402	**0.010**
	PSO	0.219	**0.285**	0.011
	CS	0.226	0.319	0.021
	PEO	0.627	1.167	0.020
ITSE	BAT	0.251	0.339	**0.002**
	PSO	**0.213**	**0.330**	0.004
	CS	0.268	0.376	0.005
	PEO	0.256	0.341	0.007
ITAE	BAT	**0.218**	**0.293**	**0.004**
	PSO	0.233	0.343	0.004
	CS	0.246	0.404	0.007
	PEO	0.248	0.432	0.005

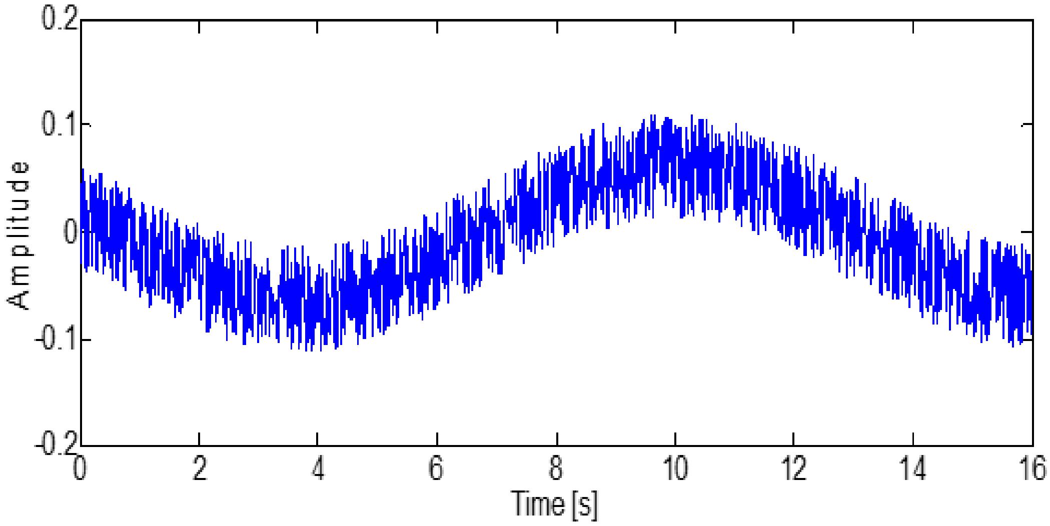
Table 8. Means of performance parameters in transient and in steady state using ISE, ITAE and ITSE

5 Discussion

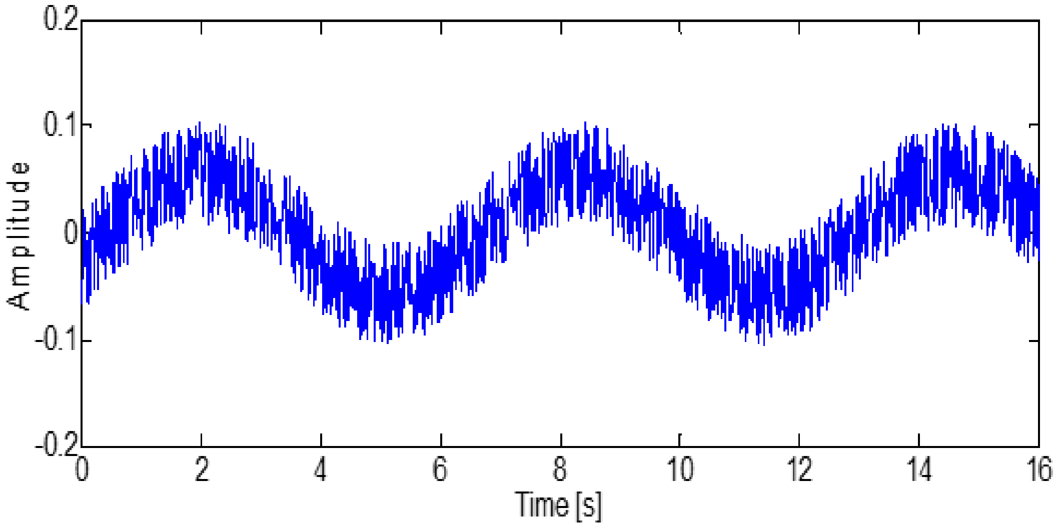
According to Tables V and VI, the worst result in terms of minimum convergence speed is demonstrated by CS algorithm in all cases while the minimum fitness value is offered by BAT algorithm. In terms of processing time, PSO, PEO and BAT algorithms offered nearly the same time. However, CS algorithm consumed more processing time. BAT algorithm demonstrated an optimal convergence time with a fast processing time. Moreover, it is also able to reach the minimum value of the fitness functions. PSO algorithm involved a reasonable processing while demonstrating an average value of the fitness. The worst of the four algorithms is the CS, which offers longer processing time and slower convergence. The highest of minimum fitness function is given by PEO and CS algorithms. Based on that, we can confirm that BAT algorithm demonstrated relatively superior performance compared with PSO, PEO and CS algorithms. In overall, BAT algorithm gives uniform MFs distributions in case of tuning in regulation and trajectory tracking modes.

Quantitatively, BAT algorithm demonstrated 50% less computation time compared to CS algorithm and 10% less than PSO and PEO algorithms. In terms of fitness, BAT algorithm over-performs than PSO, CS and PEO algorithms by a factor of 5%, 15‘% and 50% respectively. According to these results, the BAT-based FLC is found to be superior to the other algorithms.

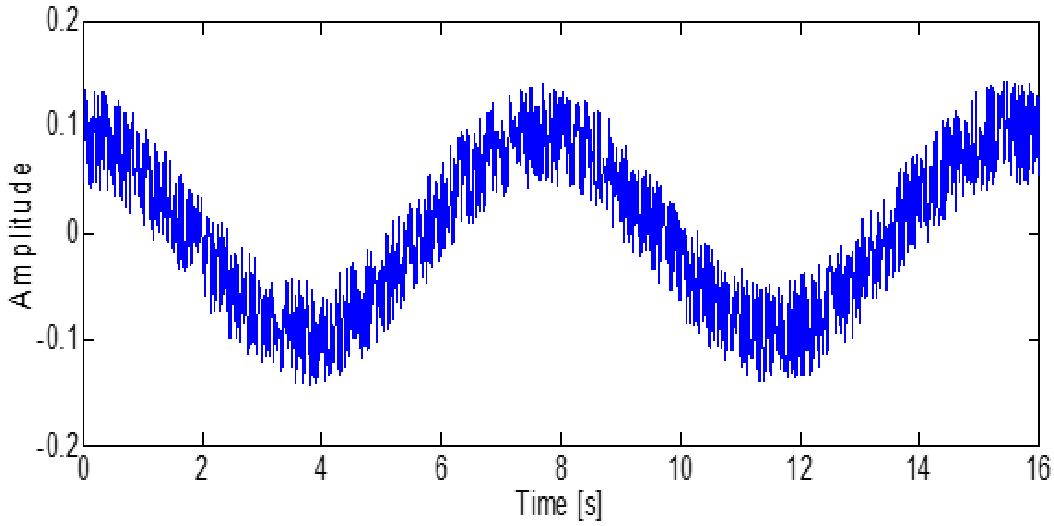
In order to find the appropriate and optimized controller, PSO, CS, PEO and BAT algorithms have been implemented on the quadrotor controller in software. Simulation results of applying optimized FLC are shown in Figures 9-14. From results of ISE-based optimized FLC (see Figure 8), it is clear that the accuracy offered by BAT algorithm is superior than demonstrated by the other three algorithms. This is consistent with results shown in Table VII, where in most cases, BAT algorithm showed better results in terms of RT, ST and OS. Figure 10 illustrates the results of the convergence speed of the controllers, as indicated by the ITAE criteria. It is evident that BAT algorithm was able to significantly accelerate the convergence. The results of ITSE fuzzy based PSO, PEO, CS and BAT algorithms



(a)



(b)



(c)

Figure 15. **External disturbances affecting the attitude dynamics of the quadrotor: (a) ϕ (b) θ (c) ψ**

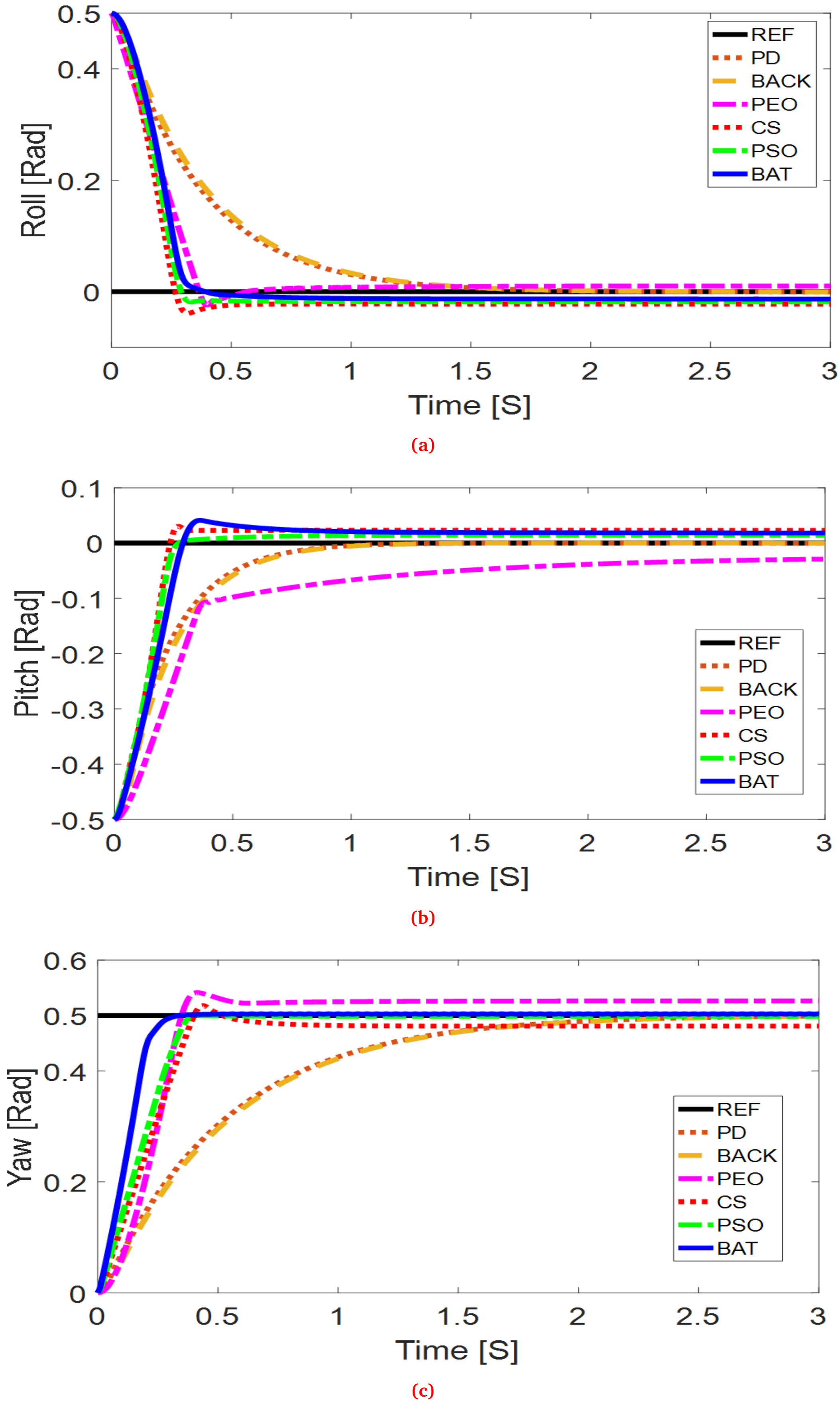
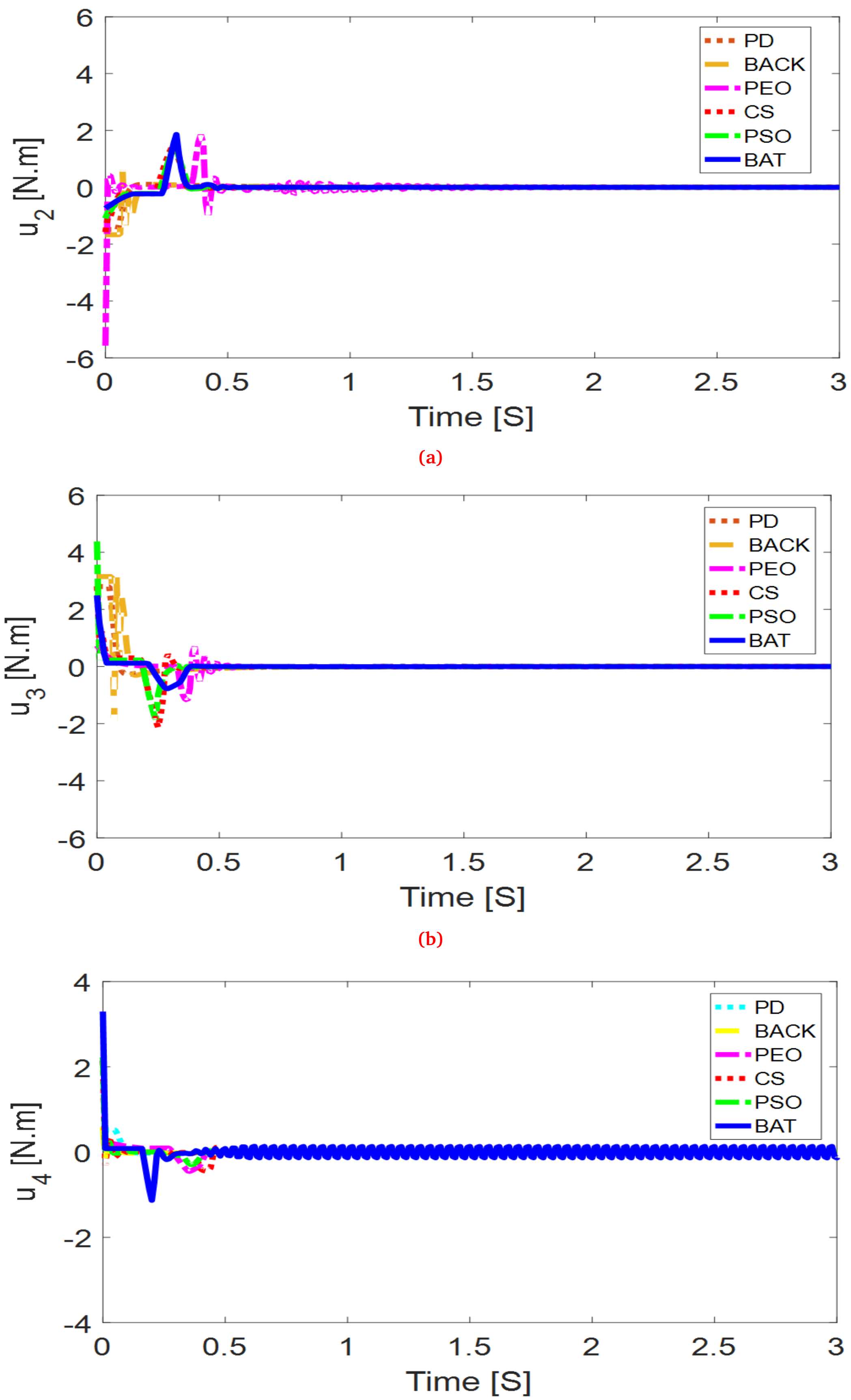


Figure 16. **Responses of quadrotor angles in regulation mode using optimized MFs by ISE in the presence of disturbance: (a) ϕ (b) θ (c) ψ**

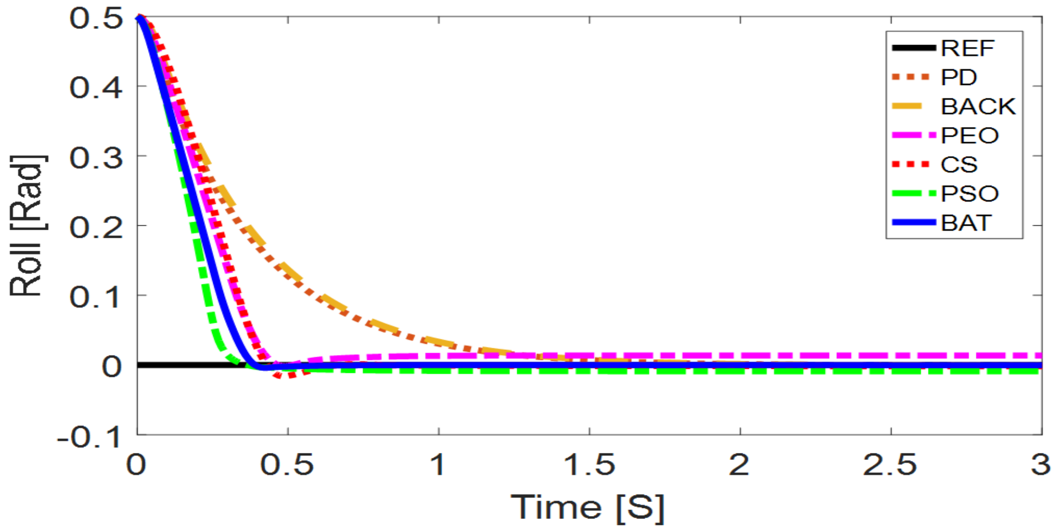


(a)

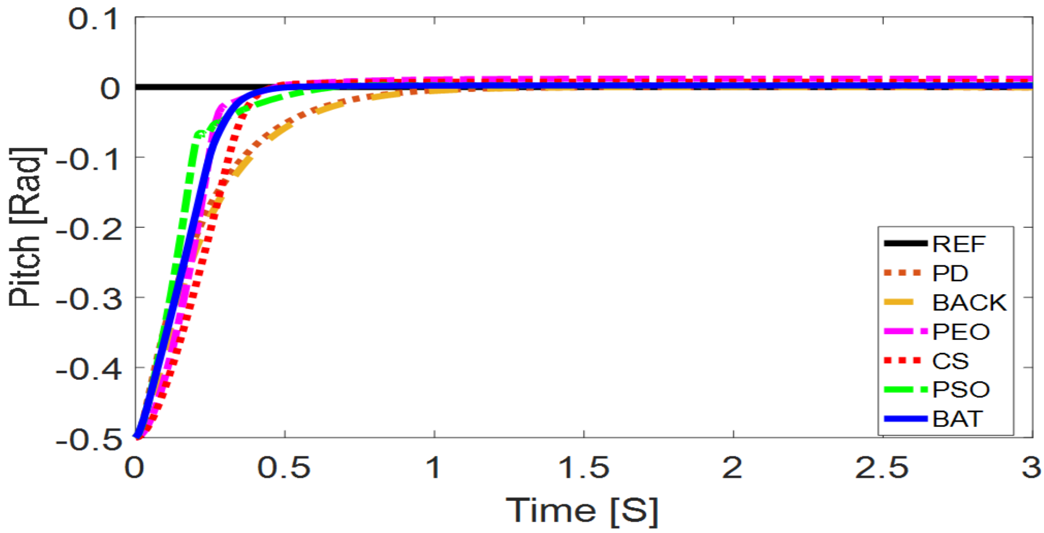
(b)

(c)

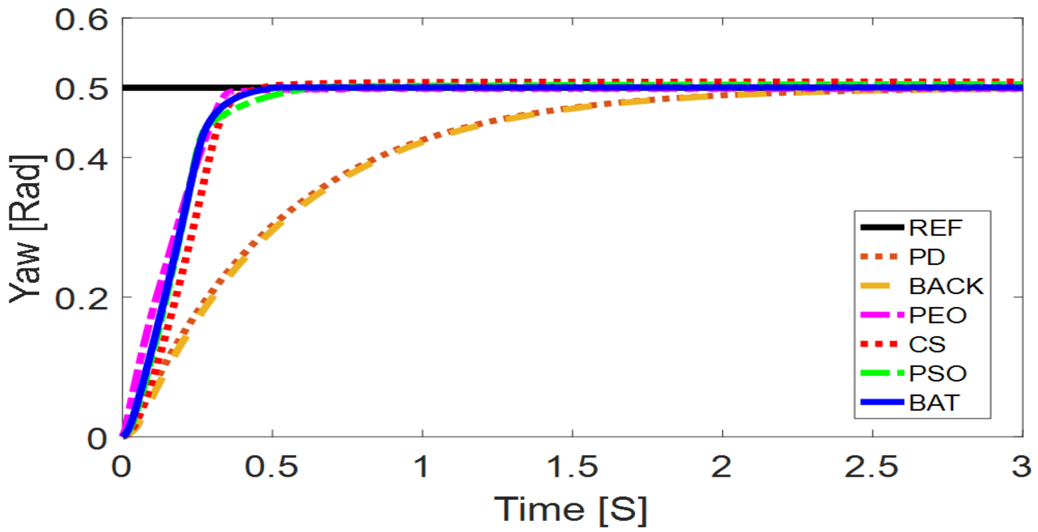
Figure 17. Control efforts of quadrotor angles in regulation mode using optimized MFs by ISE in the presence of disturbance: (a) ϕ (b) θ (c) ψ



(a)

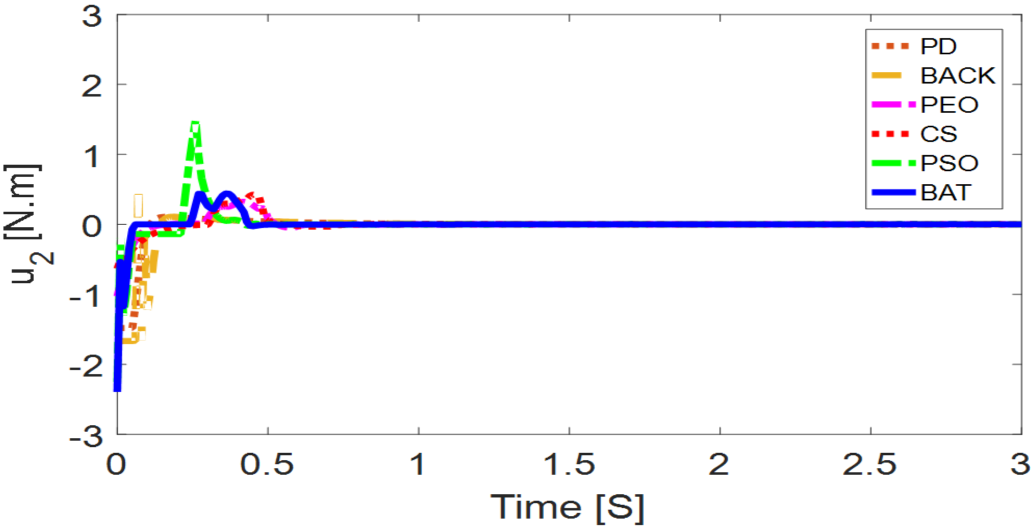


(b)

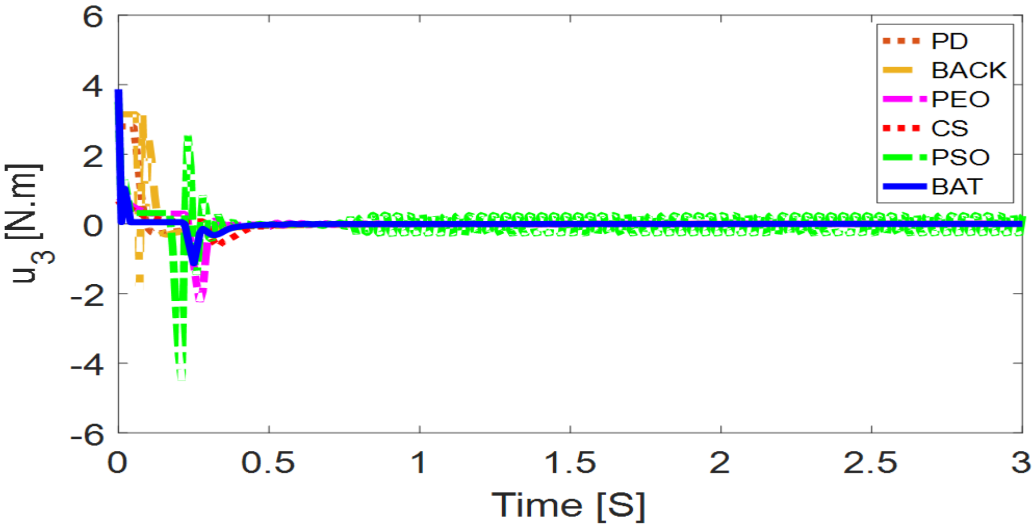


(c)

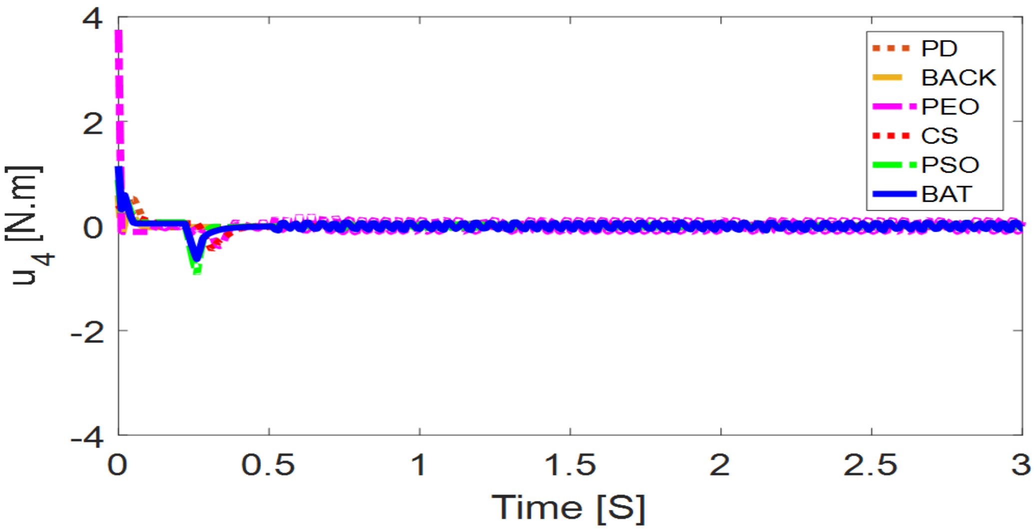
Figure 18. **Responses of quadrotor angles in regulation mode using optimized MFs by ITAE in the presence of disturbance: (a) ϕ (b) θ (c) ψ**



(a)

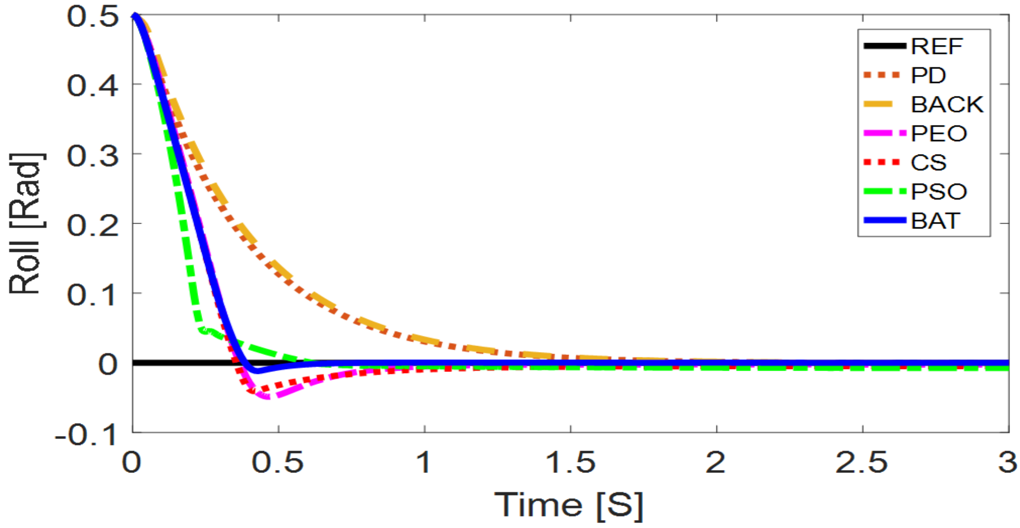


(b)

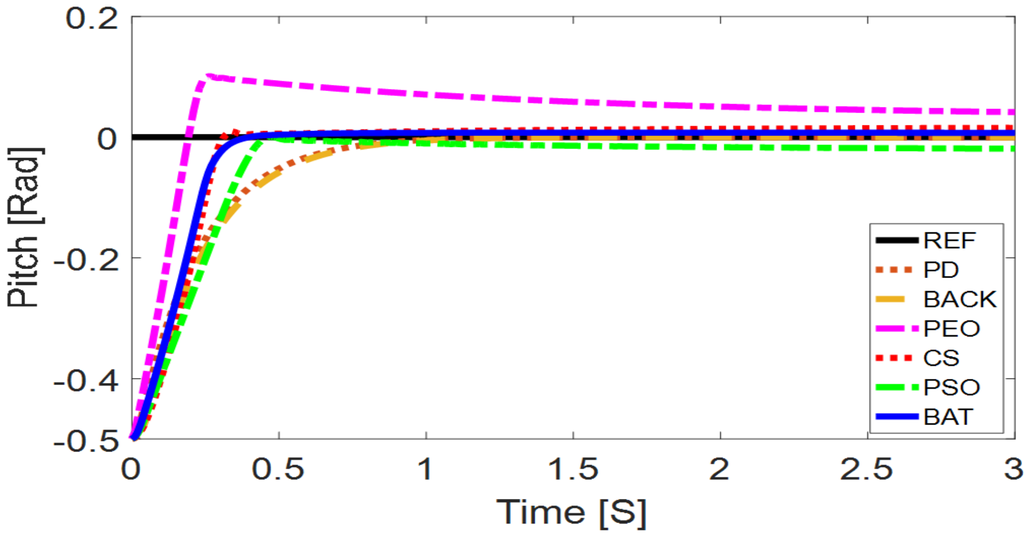


(c)

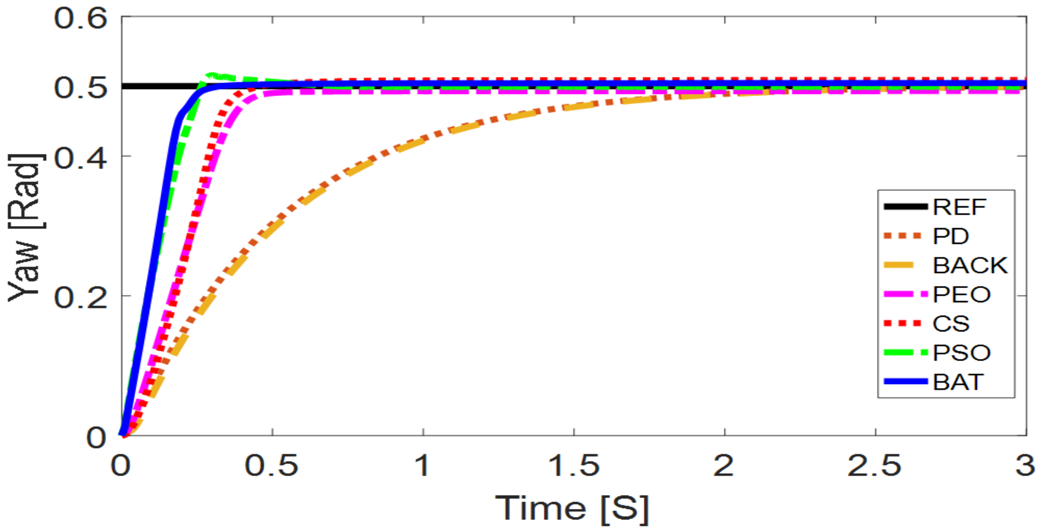
Figure 19. Control efforts of quadrotor angles in regulation mode using optimized MFs by ITAE in the presence of disturbance: (a) ϕ (b) θ (c) ψ



(a)

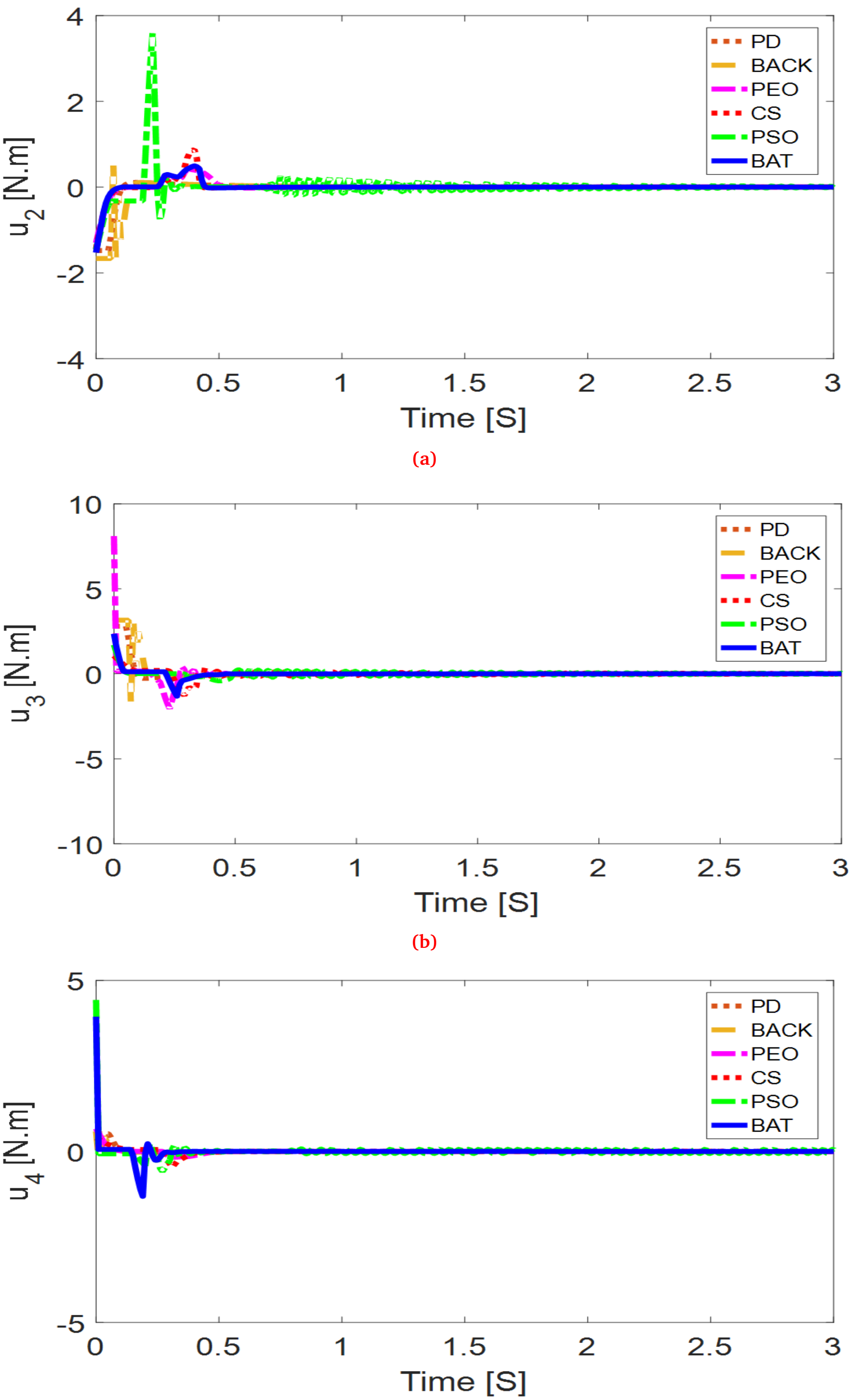


(b)



(c)

Figure 20. **Responses of quadrotor angles in regulation mode using optimized MFs by ITSE in the presence of disturbance: (a) ϕ (b) θ (c) ψ**

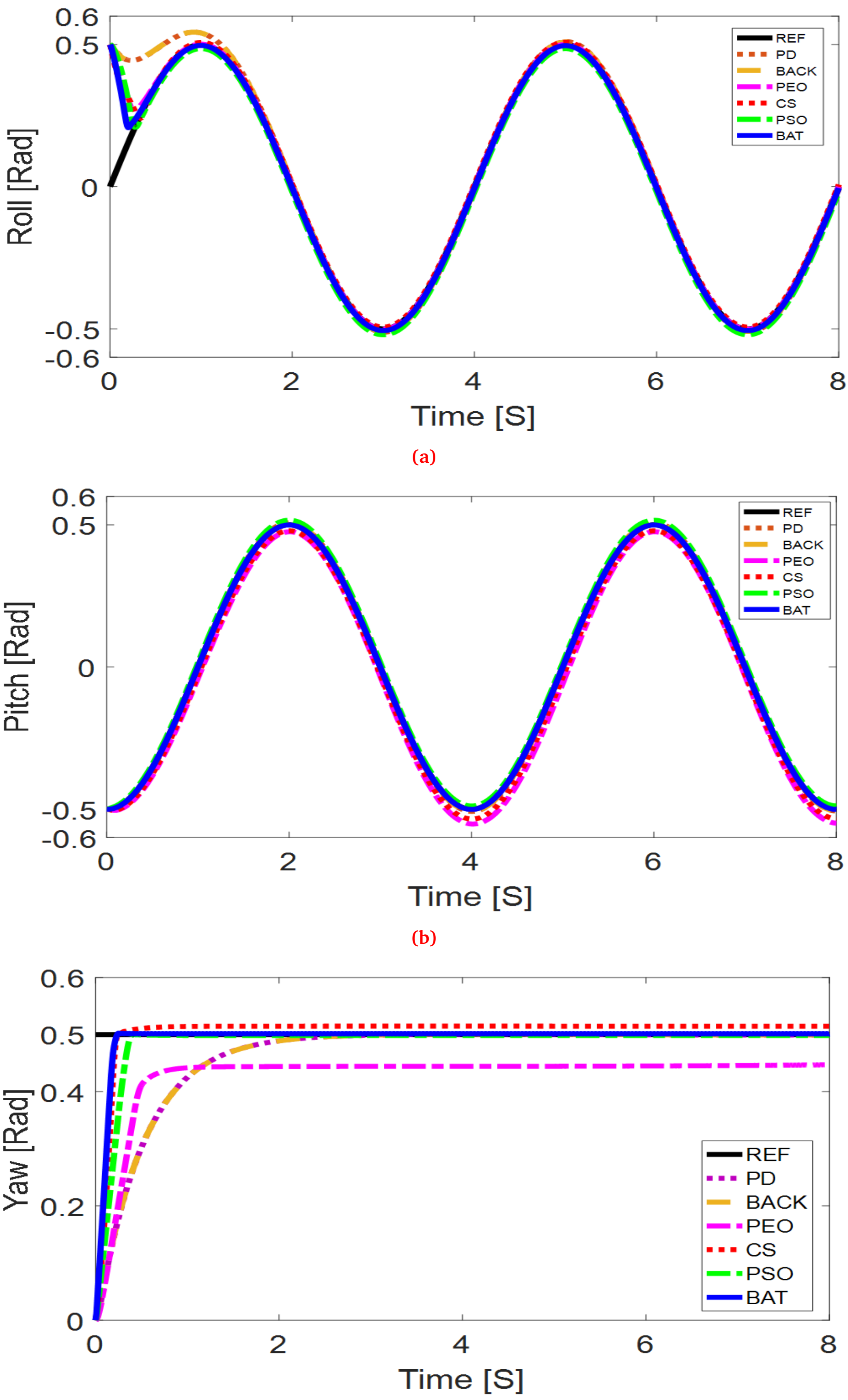


(a)

(b)

(c)

Figure 21. **Control efforts of quadrotor angles in regulation mode using optimized MFs by ITSE in the presence of disturbance: (a) ϕ (b) θ (c) ψ**

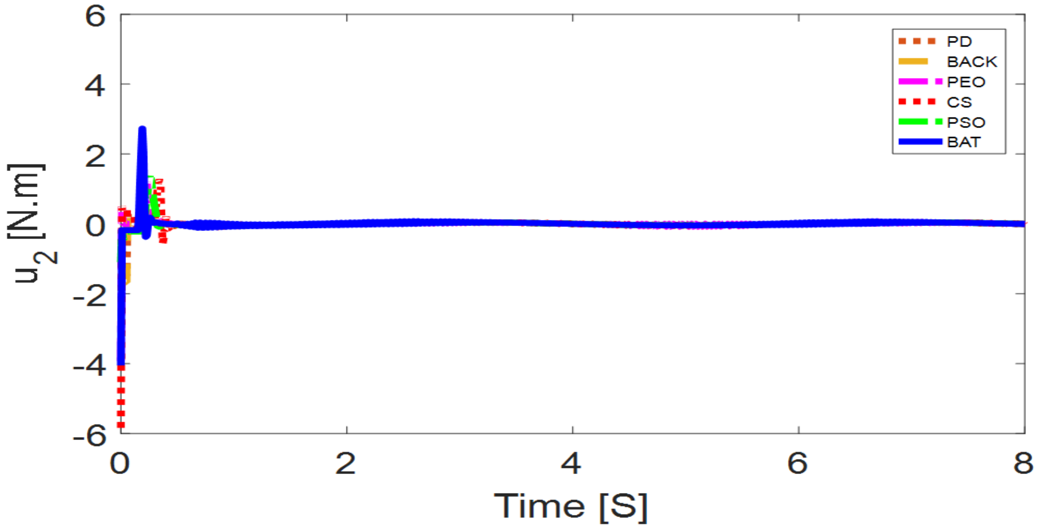


(a)

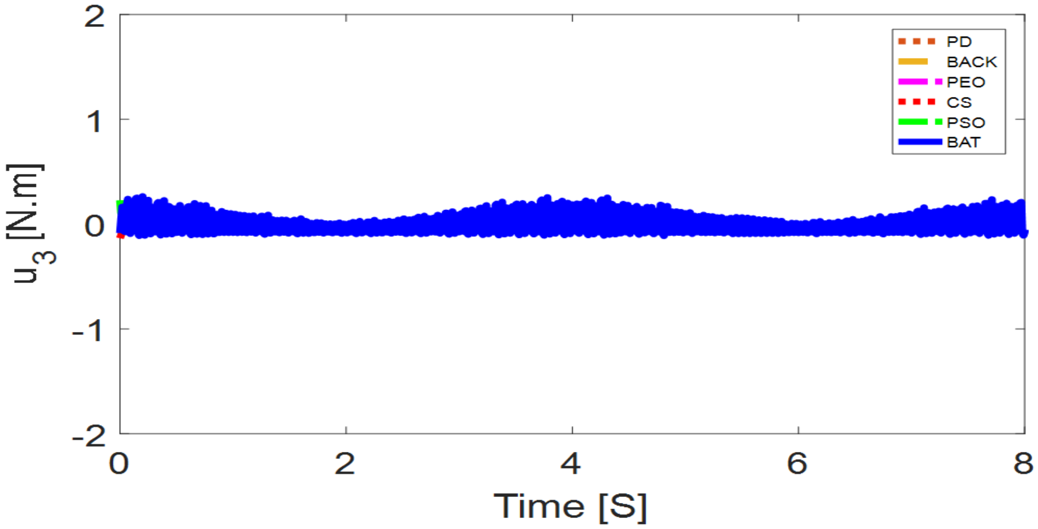
(b)

(c)

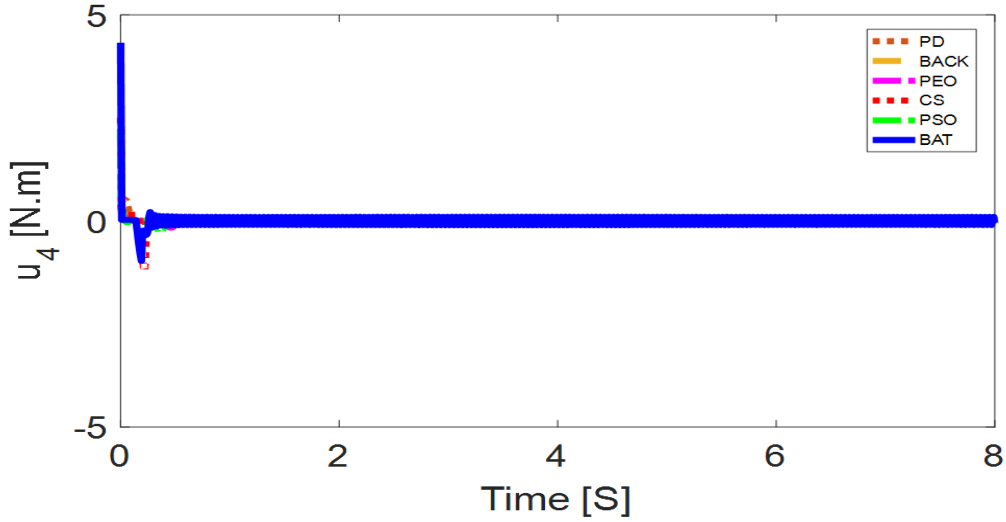
Figure 22. **Responses of quadrotor angles using optimized MFs by ISE in the presence of disturbance: (a)** ϕ **(b)** θ **(c)** ψ



(a)

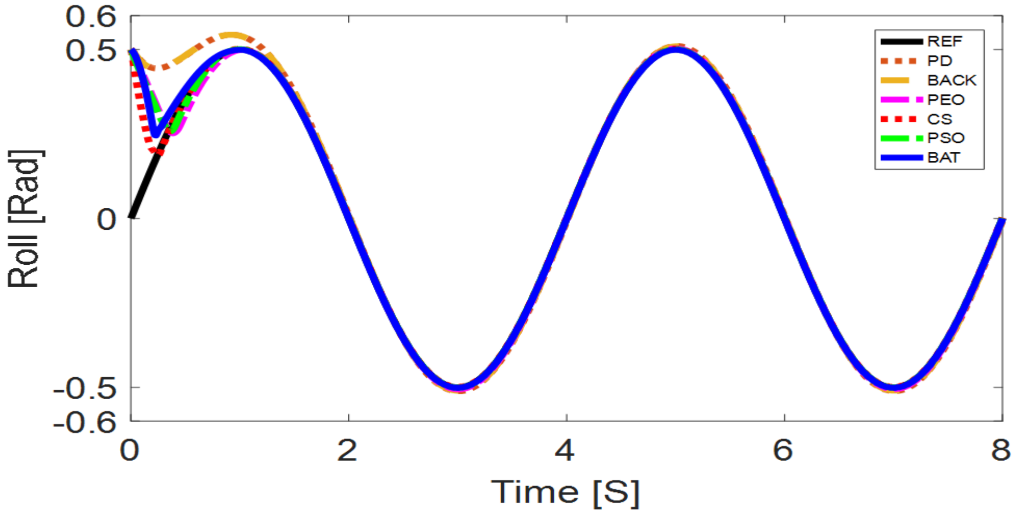


(b)

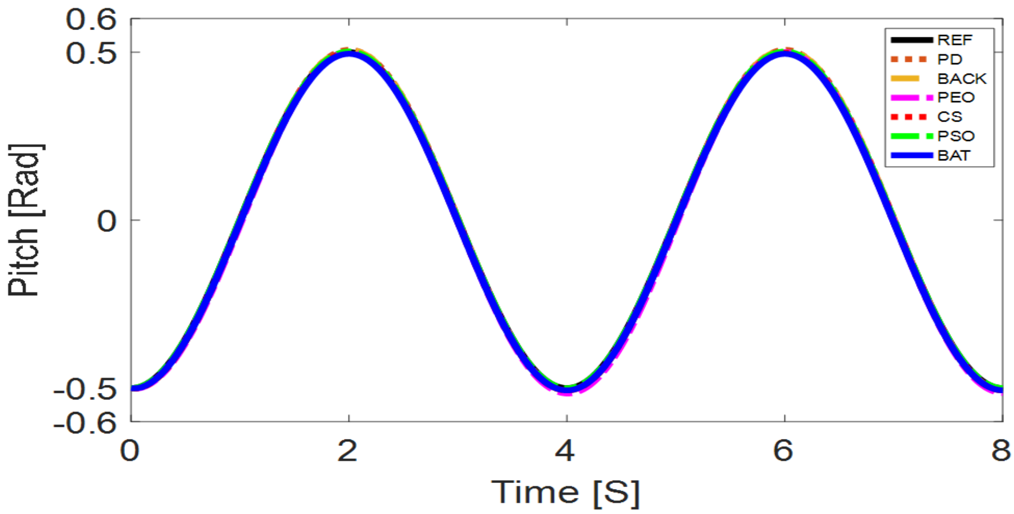


(c)

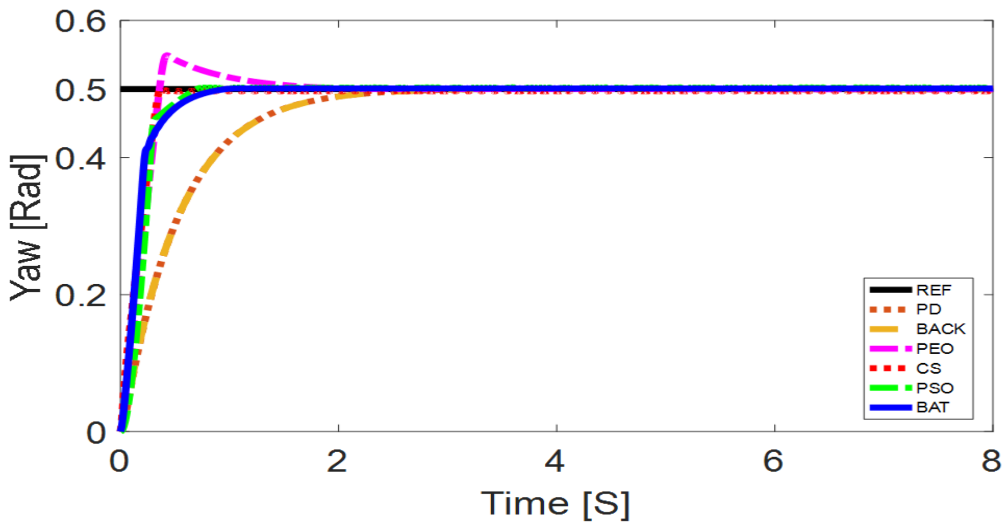
Figure 23. **Control efforts of quadrotor angles using optimized MFs by ISE in the presence of disturbance: (a)** ϕ **(b)** θ **(c)** ψ



(a)

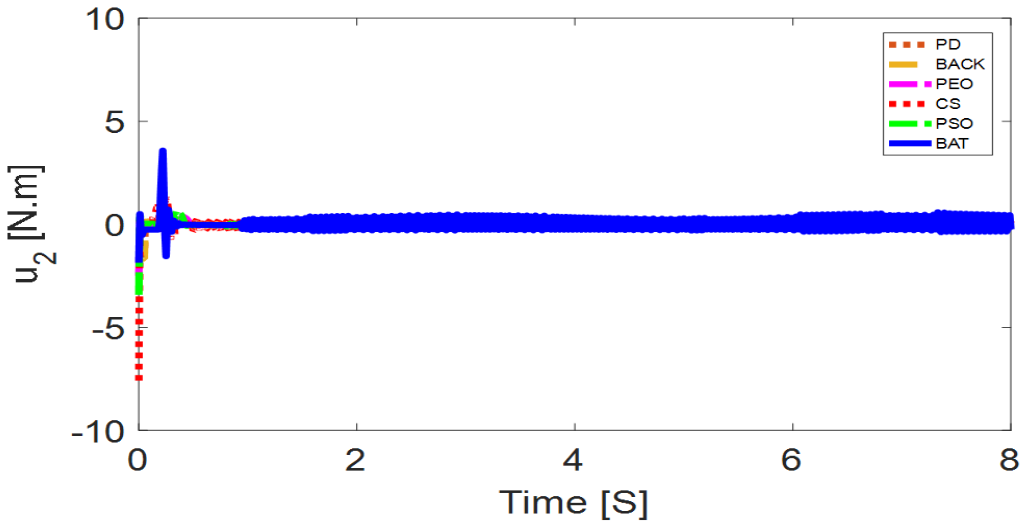


(b)

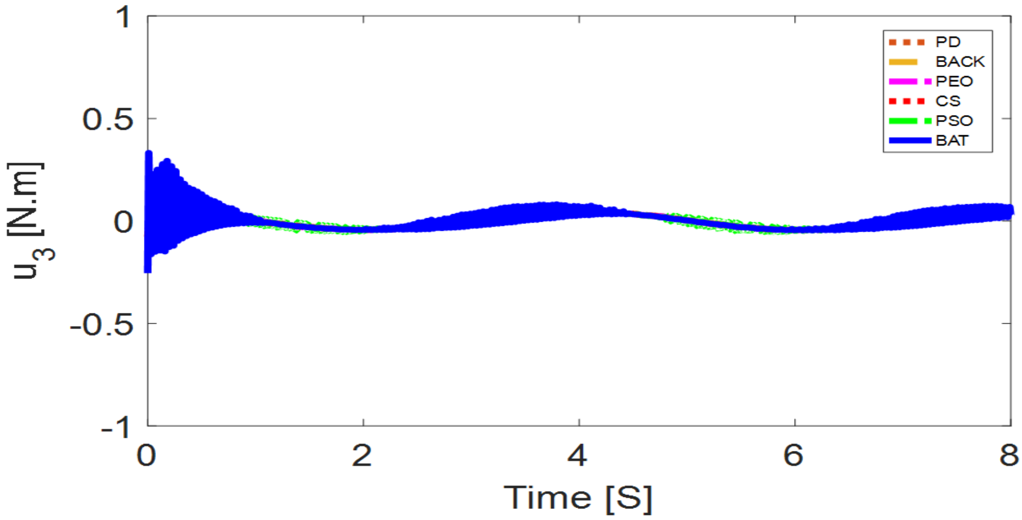


(c)

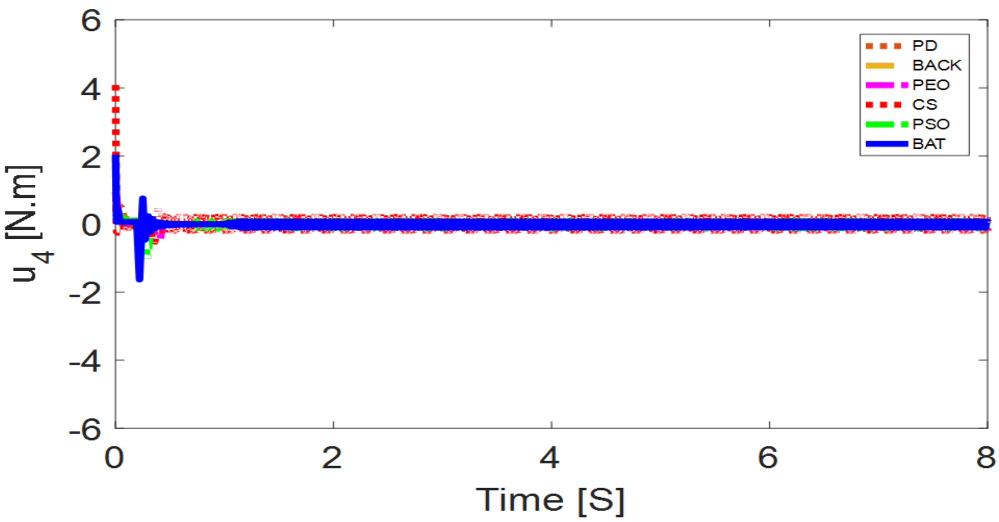
Figure 24. **Responses of angles quadrotor in trajectory tracking mode using optimized MFs by ITAE in the presence of disturbance: (a) ϕ (b) θ (c) ψ**



(a)

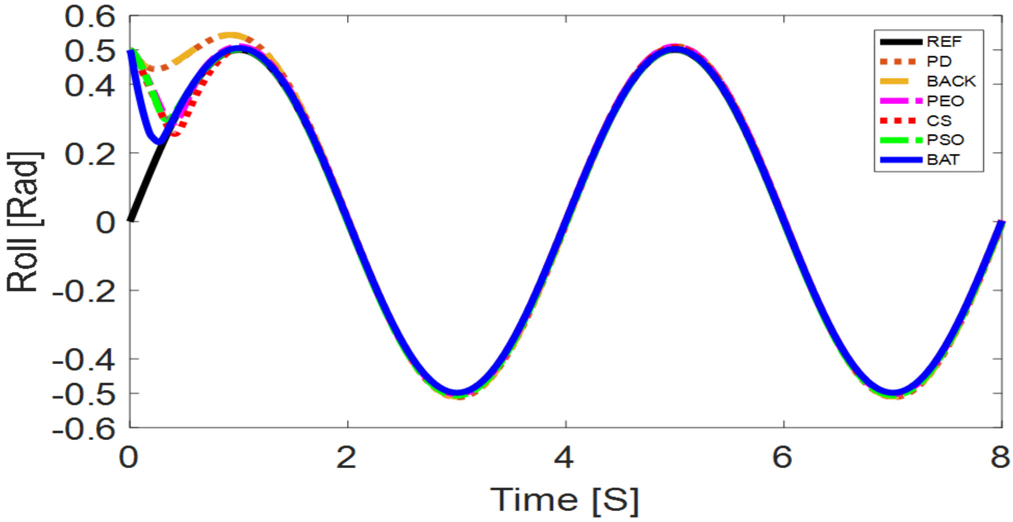


(b)

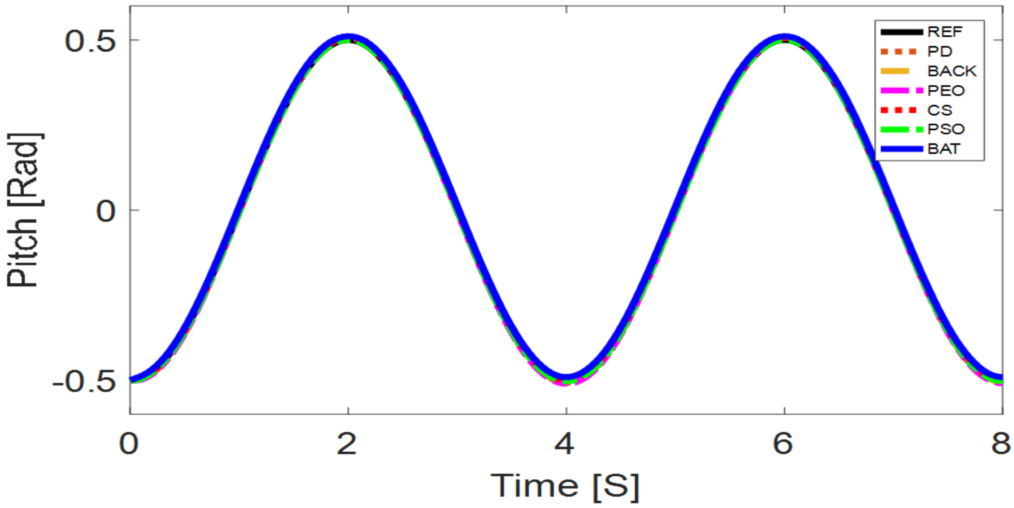


(c)

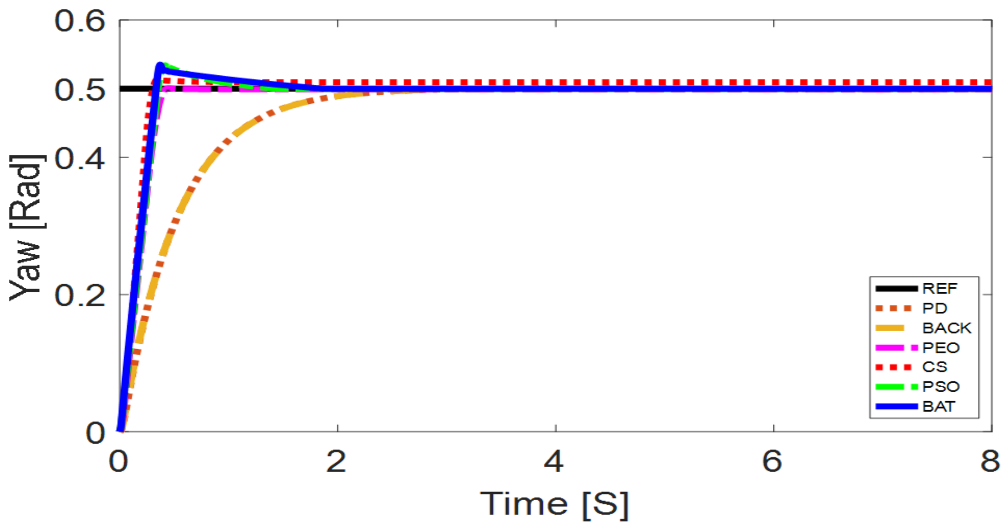
Figure 25. **Control efforts of angles quadrotor in trajectory tracking mode using optimized MFs by ITAE in the presence of disturbance: (a) ϕ (b) θ (c) ψ**



(a)



(b)



(c)

Figure 26. **Responses of quadrotor angles in trajectory tracking mode using optimized MFs by ITSE in the presence of disturbance: (a) ϕ (b) θ (c) ψ**

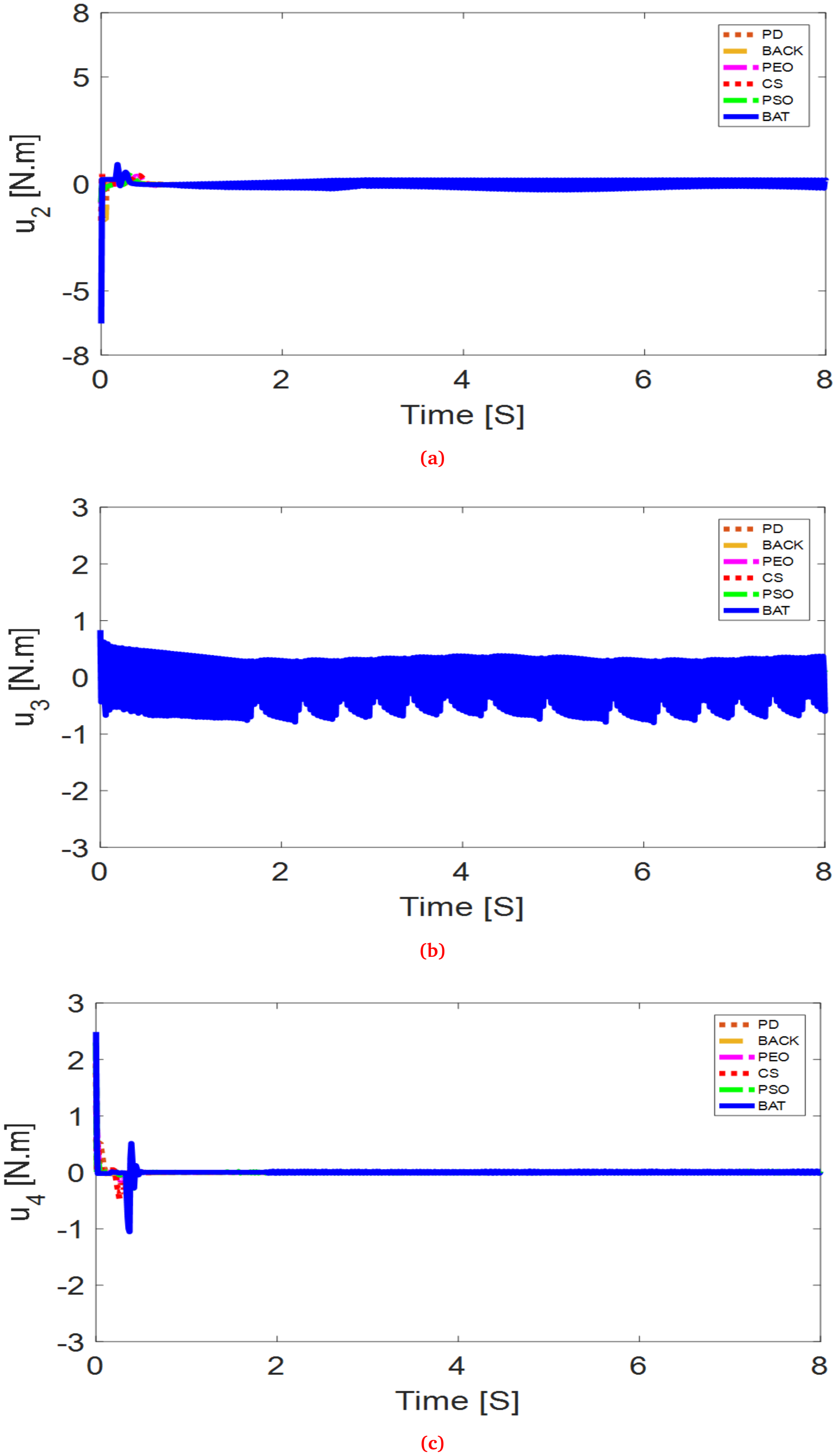


Figure 27. Control efforts of quadrotor angles in trajectory tracking mode using optimized MFs by ITSE in the presence of disturbance: (a) ϕ (b) θ (c) ψ

(Figure 13 and 20) demonstrate that the best response is offered by BAT algorithm compared with those of the other optimized controllers. Figure 13 indicates the compromise between speed and accuracy. e.g. CS algorithm offered best speed in case of roll control however, with deteriorated accuracy. Same remark holds true for pitch control using PSO. In contrast, in case of BAT algorithm, this compromise is taken into account. Comparing the performance of the optimized fuzzy controller with that of PD and backstepping controllers confirmed the superior performance exhibited by BAT based fuzzy controller. PD and backstepping showed a significant overshoot while requiring more settling time. The existence of oscillations in all the control signals of PD and Integral backstepping is also remarked. Results of both modes in the presence of disturbances are illustrated in Figures 13-18. These results dictate that the designed controllers are robust against perturbations and noise. BAT-based FLC gave the best results compared to FLC algorithms based on CS, PSO and PEO in majority of the responses. BAT-based FLC also demonstrated superior robustness compared with the integral backstepping and PD control counterparts. Some results of BAT-based FLC present usual oscillations around the desired responses (Figures 16c, 18c, 20c, 22c,24c and 26c), indicating the difficulty to control the quadrotor yaw angle.

6 Conclusion

The work presented in this paper consists of metaheuristic optimization of FIS to control a quadrotor system. The objective was to determine the advantages of the metaheuristic algorithms for different objective functions and to evaluate the best metaheuristic to control attitude of a quadrotor in the presence of uncertainties and disturbances. The outputs MFs distributions of the fuzzy controllers have been optimized by three metaheuristic algorithms; PSO, PEO, BAT and CS. The dynamic quadrotor model is used in simulation to test the developed commands. The application of the optimized FLC using BAT, CS, PEO and PSO and comparative results revealed that BAT algorithm has offered better performance compared with PSO, PEO and CS algorithms.

In near future, we plan to realize the algorithms under discussion on a real quadrotor platform. Although the present work considers the controller design as the single problem, however, it is planned to use multi-objective methods by choosing non-contradictory objective functions for a highly nonlinear and multivariable system like a quadrotor. Another possible avenue is to focus on BAT and its variants to control a quadrotor attitude with the final objective of designing a full-fledge trajectory tracking system of a quadrotor.

Data Availability Statement (DAS)

The authors confirm that the data supporting the findings of this study are available within the article.

Disclosure statement

No potential conflict of interest was reported by the authors.

References

1. S. A. Ajwad, J. Iqbal, M. I. Ullah, A. Mehmood, A systematic review of current and emergent manipulator control approaches, Frontiers of Mechanical Engineering, Springer, 2015; 10(2):198-210.

2. S. A. Ajwad, M. I. Ullah, B. Khelifa, and J. Iqbal, A comprehensive state-of-the-art on control of industrial articulated robots, Journal of the Balkan Tribological Association, 2014; 20(4): 499-521.

3. S. Ullah, A. Mehmood, Q. Khan, S. Rehman, J. Iqbal, Robust integral sliding mode control design for stability enhancement of underactuated quadcopter, International Journal of Control, Automation and Systems, Springer, 2020; https://doi.org/10.1007/s12555-019-0302-3

4. Henderson V. Gue, Alvin Y. Chua, Development of a fuzzy GS-PID controlled quadrotor for payload drop missions, Journal of Telecommunication, Electronic and Computer Engineering, 2018;10:1-5.

5. Bounemeur, M. Chemachema, N. Essounbouli, Indirect adaptive fuzzy fault-tolerant tracking control for MIMO nonlinear systems with actuator and sensor failures, ISA Transactions, 2018;79: 45-61.

6. Amirkhani, M. Shirzadeh, E. I. Papageorgiou, M. R. Mosavi, Visual-based quadrotor control by means of fuzzy cognitive maps, ISA Transactions, 2016;60:128-142.

7. O. Tutsoy, D. E. Barkana, H. Tugala, Design of a completely model free adaptive control in the presence of parametric, non-parametric uncertainties and random control signal delay, ISA Transactions, 2018;76:67-77.

8. K. Wang, C. Hua, J. Chen and M. Cai (2020) Dual-loop integral sliding mode control for robust trajectory tracking of a quadrotor, International Journal of Systems Science, 51:2, 203-216, DOI: 10.1080/00207721.2019.1622815.

9. Z. Li, X. Ma & Y. Li (2020) Robust trajectory tracking control for a quadrotor subject to disturbances and model uncertainties, International Journal of Systems Science, 51:5, 839-851, DOI: 10.1080/00207721.2020.1746430.

10. M. Wasim, M. Ullah, J. Iqbal, Gain-scheduled proportional integral derivative control of taxi model of unmanned aerial vehicles, Revue Roumaine des Sciences Techniques-Serie Electrotechnique et Energetique, 2019; 64(1):75-80.

11. M. Labbadi, Y. Boukal, M. Cherkaoui, (). Path following control of quadrotor UAV with continuous fractional-order super twisting sliding mode. Journal of Intelligent & Robotic Systems, 2020; 100(3), 1429-1451.

12. N. Cherrat, H. Boubertakh, H. Arioui, Adaptive fuzzy PID control for a quadrotor stabilization, In Proceedings of the IOP Conference Series: Materials Science and Engineering, Thailand, February 2018.

13. M. K. Al-Sharman, B. J. Emran, M. A. Jaradat, H. Najjaran, R. Al-Husari, Y. Zweiri, Precision landing using an adaptive fuzzy multi-sensor data fusion architecture, Applied Soft Computing, 2018;69:149-164.

14. Y. Yang, Y. Yan, Attitude regulation for unmanned quadrotors using adaptive fuzzy gain-scheduling sliding mode control, Aerospace Science and Technology, 2016;54:208-217.

15. Cherifi, K. Guelton, L. Arcese, Uncertain T-S model-based robust controller design with D-stability constraints—A simulation study of quadrotor attitude stabilization, Engineering Applications of Artificial Intelligence, 2018;67:419-429.

16. Sarabakha, E. Kayacan, C. Wagner, R. John and J. M. Garibaldi, Input uncertainty sensitivity enhanced nonsingleton fuzzy logic controllers for long-term navigation of quadrotor UAVs, IEEE/ASME Transactions on Mechatronics, 2018; 23(2):725-734.

17. H. Wu, H. Li, R. Xiao, J. Liu, Modeling and simulation of dynamic ant colony's labor division for task allocation of UAV swarm, Physica A: Statistical Mechanics and its Applications, 2018;491:127-141.

18. V. Mohammadi, S. Ghaemi, H. Kharrati, PSO tuned FLC for full autopilot control of quadrotor to tackle wind disturbance using bond graph approach, Applied Soft Computing, 2018;65:184-195.

19. L. Blasi, S. Barbato, E. D'Amato, A mixed probabilistic–geometric strategy for UAV optimum flight path identification based on bit-coded basic manoeuvres, Aerospace Science and Technology, 2017;71:1-11.

20. L. Huang, H. Qu, P. Ji, X. Liu, Z. Fan, A novel coordinated path planning method using k-degree smoothing for multi-UAVs, Applied Soft Computing, 2016;48:182-192.

21. E. V. Kumar, G. S. Raaja, J. Jerome, Adaptive PSO for optimal LQR tracking control of 2 DoF laboratory helicopter, Applied Soft Computing, 2016;41:77-90.

22. E. Camci, D. R. Kripalani, L. Ma, E. Kayacan, M. A. Khanesar, An aerial robot for rice farm quality inspection with type-2 fuzzy neural networks tuned by particle swarm optimization-sliding mode control hybrid algorithm, Swarm and Evolutionary Computation, 2018;41:1-8.

23. Q. Bian, K. Zhao, X. Wang, R. Xie, System Identification Method for Small Unmanned Helicopter Based on Improved Particle Swarm Optimization, Journal of Bionic Engineering, 2016;13(3):504-514.

24. Rezoug, Z. Achour, M. Hamerlain, Ant colony optimization of type-2 fuzzy helicopter controller. 2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014), Bali, 2014, pp. 1548-1553.

25. V. Mohammadia S. Ghaemib H. Kharratic "PSO tuned FLC for full autopilot control of quadrotor to tackle wind disturbance using bond graph approach" Applied Soft Computing, 2018; 65:184-195.

26. O. Castillo, R. Martínez-Marroquín, P. Melin, F. Valdez, J. Soria, Comparative study of bio-inspired algorithms applied to the optimization of type-1 and type-2 fuzzy controllers for an autonomous mobile robot, Information Sciences, 2012;192:19-38.

27. Sánchez, P. Melin, Hierarchical Genetic Algorithms for Type-2 Fuzzy System Optimization Applied to Pattern Recognition and Fuzzy Control. Studies in Computational Intelligence Recent Advances on Hybrid Approaches for Designing Intelligent Systems, 2014; 547:19–35.

28. Y. Zhang, S. Wang, G.Ji, A comprehensive survey on particle swarm optimization algorithm and its applications, Mathematical Problems in Engineering, 2015;2015:1-38.

29. F. Olivas, L. Amador-Angulo, J. Perez, C. Caraveo, F. Valdez, O. Castillo, Comparative study of type-2 fuzzy particle swarm, bee colony and BAT algorithms in optimization of fuzzy controllers. Algorithms, 2017; 10 (3): 1-10.

30. X. S. Yang,Nature-inspired optimization algorithms, Elsevier (2014).

31. M. Ferdaus, M. Pratama, S. G. Anavatti, M. A. Garratt, Y. Pan, Generic evolving self-organizing neuro-fuzzy control of bio-inspired unmanned aerial vehicles,arXivpreprint arXiv:1802.00635, 2018.

32. X. S. Yang, BAT algorithm: literature review and applications. International Journal of Bioinspired Computation, 2013; 5(3):141–149.

33. Fister, X. S.Yang, D. Fister, I. Fister, Cuckoo Search: A brief literature review. Yang XS. (eds) Cuckoo Search and Firefly Algorithm.In Cuckoo search and firefly algorithm (pp.49–62).Berlin:Springer.

34. M.I. Solihin, M.F. Zanil, Performance comparison of Cuckoo search and differential evolution algorithm for constrained optimization, Int. Enginering Research and Innovation Symposium, 2016;160(1):1-7.

35. K. Premkumar, B.V. Manikandan, BAT algorithm optimized fuzzy PD based speed controller for brushless direct current motor, International Journal of Engineering Science Technology, 2016;19:818-840.

36. Z. Z. Yang, S.Z. Zhang, W. Yan, New results on sliding-mode control for takagi–sugeno fuzzy multiagent systems,IEEE Transactions on Cybernetics, 2018; 99:1-13.

37. M. M. Ferdaus, M. Pratama, S. G. Anavatti, M. A. Garratt, A generic self-evolving neuro-fuzzy controller based high-performance hexacopter altitude control system, IEEE International Conference on Systems, Man, and Cybernetics, 2018; 2784-2791.

38. G. C. V. Ramadas, E. M. G. P. Fernandes, A. M. V. Ramadas, A. M. A. C. Rocha, M. F. P. Costa, "On metaheuristics for solving the parameter estimation problem in dynamic systems: a comparative study," Journal of Optimization, 2018; 2018:1–21.

39. Y. S. G. Nashed, P. Mesejo, R. Ugolotti, J. Dubois-Lacoste, S. Cagnoni, A comparative study of three GPU-based metaheuristics, Lecture Notes in Computer Science Parallel Problem Solving from Nature - PPSN XII, 2012;7492: 398–407.

40. Ö. Hazır, Y. Günalay, E. Erel, Customer order scheduling problem: a comparative metaheuristics study, International Journal of Advanced Manufacturing Technology, 2007; 37(5-6):589–598.

41. K. Alexis, G. Nikolakopoulos, and A. Tzes, Model predictive quadrotor control: attitude, altitude and position experimental studies, IET Control Theory & Applications, 2012 6(12): 1812-1827.

42. S. Bouabdallah, Design and control of quadrotors with application to autonomous flying. No.3727, PhD Thesis, Faculty of Science and Technology of Engineer, Micro-Technique Section, EPFL, 2007.

43. K. Baizid, A, Yousnadj, A. Meddahi, R. Chellali, J. Iqbal, Time scheduling and optimization of industrial robotized tasks based on Genetic Algorithms, Robotics and Computer Integrated Manufacturing, 2015; 34:140-150.

44. X. S. Yang, S. Deb, Cuckoo search via Levy flights, Lecture Notes in Nature & Biologically Inspired Computing, NaBIC 2009 World Congress, 2009: 210–214.

45. X. S., Yang, A New Metaheuristic Bat-Inspired Algorithm. In J. González, D. Pelta, C. Cruz, G. Terrazas & N. Krasnogor (Eds.), Nature Inspired Cooperative Strategies for Optimization (NICSO 2010), Springer Berlin Heidelberg. 2010. 284: pp. 65-74.

46. W. Kongkaew, Bat algorithm in discrete optimization: A review of recent applications, Songklanakarin J. Sci. Technol. 2017; 39 (5): 641-650.

47. S. G. Khan, S. Bendoukha, W. Naeem, J. Iqbal, Experimental validation of an integral sliding mode-based LQG for the pitch control of a UAV-mimicking platform, Advances in Electrical and Electronics Engineering, 2019; 17(3):275-284.

48. Y-Z. Lu, Y-W. Chen, M-R. Chen, P. Chen and G-Q. Zeng Extremal Optimization Fundamentals, Algorithms, and Applications, Chemical Industry Press. Published by Taylor & Francis, 2016.

49. G. Q. Zeng, J. Chen, Y-X. Dai, L-M. Li, C.-W. Zheng, Min-Rong Chen, Design of fractional order PID controller for automatic regulator voltage system based on multi-objective extremal optimization, Neurocomputing, 2015; 160: 173-184.

50. H. Altas, A. M. Sharaf, A generalized direct approach for designing fuzzy logic controllers in MATLAB/Simulink GUI environment. International Journal of Information Technology and Intelligent Computing, 2007, 1(4): 1-27.

51. Al Younes, Y., Drak, A., Noura, H., Rabhi, A., and El Hajjaji, A. (2016). Robust model-free control applied to a quadrotor UAV. Journal of Intelligent and Robotic Systems, 84(1), 37-52.

7 Appendix

Symbol	Nomenclature
ξ	Quadrotor's positions vector
m	Total mass
Ω	Angular velocity expressed in the fixed frame
F_f	Total force generated by the four rotors
F_d	Drag forces
F_g	Gravitational force
Γ_f	Moment caused by thrust and drag forces
Γ_a	Moment resulting from aerodynamic friction
Γ_g	Moment caused by gravitational force
J	Symmetrical inertia matrix
K_{ax},K_{ay},K_{az}	coefficients of aerodynamic friction
I_x,I_y,I_z,	inertia moments
J_r	rotor inertia

Table 9. Quadrotors' nomenclature

The quadrotor's physical parameters are

$m = 1.5\ kg$; $g = 9.81\ N \times m$; $l = 0.45\ m$;
$b = 192.3208 \times 10^{-7}\ N \times S^2$;
$d = 4.003 \times 10^{-7}\ N \times S^2$;
$J_r = 6.01 \times 10^{-5}\ kg \times m^2$;
$I_x = 15.646 \times 10^{-3}\ kg \times m^2$;
$I_y = 15.646 \times 10^{-3}\ kg \times m^2$;
$I_z = 17.175 \times 10^{-3}\ kg \times m^2$;
$k_{ax} = 5.5670 \times 10^{-4}\ kg/S$;
$k_{ay} = 5.5670 \times 10^{-4}\ kg/S$;
$k_{az} = 6.3540 \times 10^{-7}\ kg/S$.