Journal of King Saud University – Engineering Sciences xxx (xxxx) xxx

HOSTED BY

Contents lists available at ScienceDirect

Journal of King Saud University – Engineering Sciences

journal homepage: www.sciencedirect.com

Original article

Nonlinear system control analysis and optimization using advanced Pigeon-Inspired optimization algorithm

Mostafa Saad *, Mohammed Abozied Hassan Abozied

Military Technical College, Cairo, Egypt

ARTICLE INFO

Article history:
Received 15 August 2021
Accepted 6 November 2022
Available online xxxx

Keywords:
Nonlinear Missile
Roll Channel
Pitch Channel
Control System Design
Pigeon-Inspired Optimization (PIO)
Numerical Simulation
Adaptive Pigeon-Inspired Optimization (APIO)

ABSTRACT

The Pigeon-Inspired optimization (PIO) algorithm is a novel intelligent optimization algorithm inspired by birds' behavior as their travel. This; behavior modeled to be used for solving many optimization problems in different fields. However; it always suffers from unstable behavior when used with nonlinear; time-varying systems. In; this paper, this algorithm is adapted to calculate the optimum controller gains for roll and pitch channels in a guided tactical missile. The; vehicle model is presented in a nonlinear; form and then shown in a linearized form for the sake of an autopilot design. The PIO; algorithm is supported and accompanied by an adaptive algorithm to determine the initial states and constraints for the PIO algorithm to enhance the behavior of the optimization algorithm to speed up the convergence rate to reach an optimum and feasible solution. Also; an estimation function is incorporated to estimate model parameters variation such as dynamic pressure, stability derivatives, and mass properties. Meanwhile; a comparative analysis is carried out with original PIO and particle swarm optimization algorithms, utilizing a non-linear; model with the presence of noise source and disturbance to ensure the ability of the algorithm to make the autopilot robust and stable against several sources of uncertainties.

1. Introduction

Missile autopilot design is a complex problem due to high dynamics, time-varying model parameters, and nonlinearity. The; recent developments in designing the non-linear missile controller include several approaches such as adaptive control (Kamen et al., 1988); non-linear control (Elnaggar and Khalil 2016); gain scheduling (Shamma et al., 1993); and, robust control(Apkarian et al., 1995). The; most used method for missile non-linear control is the gain scheduling technique. Yet; this method sometimes may not guarantee stability as the optimal gains are calculated for specific operating points then a switching case is utilized to select the gain based on the system parameters. The; switching points and system parameters are the problems of such techniques, as the design is always based on only one or a few system parameters (Shamma et al., 1993). Recently; *meta*-heuristic algorithms have provided a feasible alternative to solve the problems of controller optimization problems. Pigeon-inspired optimization (PIO) is a *meta*-heuristic algorithm (Duan et al., 2014); which is a widely used algorithm in the process of search and optimization tasks in different control systems applications (Zhu and Duan 2015, Alharkan et al., 2020). This; algorithm always aims to locate the optimal solution for high dimensionality and heterogeneity problems (Abadi and Khooban 2015, Nadia et al., 2020). Yet; the main drawback of several *meta*-heuristic algorithms is the increased complexity of finding the optimum solution for time-varying systems that may contain many suboptimal solutions and not differentiable, which results in the absence of transient information (Shayanfar and Gharehchopogh 2018, Kamboj et al., 2020). In; this paper, a modified adaptive Pigeon-inspired optimization algorithm is proposed to calculate the optimum autopilot gains with a time-varying system and achieve a robust time response with adequate stability margins (Mohamed and Duan 2020). The contribution of this research can be summarized as follows:

A modified adaptive PIO algorithm is proposed to design an optimum non-linear controller for a tactical short-range missiles to enhance the controlled system robustness against parameters variation such as dynamic pressure, stability derivatives, and mass properties, the accelerated convergence rate, the reduction of algorithm complexity to meet the constraints of real-time implementation.

* Corresponding author.
E-mail addresses: m_saad@buaa.edu.cn (M. Saad), mohammed.abozied@mtc.edu.eg (M. Abozied Hassan Abozied).
Peer review under responsibility of King Saud University.

ELSEVIER Production and hosting by Elsevier

https://doi.org/10.1016/j.jksues.2022.11.001
1018-3639/

Please cite this article as: M. Saad and M. Abozied Hassan Abozied, Nonlinear system control analysis and optimization using advanced Pigeon-Inspired optimization algorithm, Journal of King Saud University – Engineering Sciences, https://doi.org/10.1016/j.jksues.2022.11.001

Furthermore, a complexity analysis of the proposed algorithm is carried out to investigate the availability of implementation in real time. Also; model parameters variations are estimated based on measured data. Simulation; of the modified algorithm is performed with a nonlinear model, and uncertainties are considered.

2. Vehicle model

2.1. Model equations

Flight dynamics modeling is a crucial task for the designing and implementing the flight controller. Vehicle kinematic and dynamic modeling is always described by the vehicle equation of motion, coordinate frame, and flight envelope based on the vehicle mission. The kinematic model represents the attitude and navigation equations, whereas the dynamic model represents the forces and moments equations (Siouris 2004, Zarchan 2019). Considering; the vehicle's aerodynamic coefficients and mission constraints is essential for more realistic modeling. For; deriving a 6DoF model, the vehicle is commonly treated as a rigid body (Jiang 2012). Otherwise; an infinite degree of freedom should be considered for vehicle rotation, which is hard. The; proposed model in this work can be described as follows:

Force Equations:

$$\begin{cases} \dot{u} = \frac{1}{m}X + \frac{1}{m}T - wq + vr - g\sin(\theta) \\ \dot{v} = \frac{1}{m}Y - ur + wp + g\sin(\phi)\cos(\theta) \\ \dot{w} = \frac{1}{m}Z - vp + uq + g\cos(\phi)\cos(\theta) \end{cases} \tag{1}$$

Moment Equations:

$$\begin{cases} \dot{p} = \frac{L}{I_{xx}} \\ \dot{q} = \frac{M}{I_{yy}} + \frac{I_{zz}-I_{xx}}{I_{yy}}pr \\ \dot{r} = \frac{N}{I_{zz}} + \frac{I_{xx}-I_{yy}}{I_{zz}}pq \end{cases} \tag{2}$$

Attitude Equations:

$$\begin{cases} \dot{\phi} = p + q\tan(\theta)\sin(\phi) + r\tan(\theta)\cos(\phi) \\ \dot{\theta} = q\cos(\phi) - r\sin(\phi) \\ \dot{\psi} = q\sec(\theta)\sin(\phi) + r\sec(\theta)\cos(\phi) \end{cases} \tag{3}$$

Navigation Equations:

$$\begin{cases} \dot{x}_f = u\cos\theta\cos\psi - v(\cos\phi\sin\psi - \sin\phi\sin\theta\cos\psi) \\ \qquad +w(\sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi) \\ \dot{y}_f = u\cos\theta\sin\psi + v(\cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi) \\ \qquad -w(\sin\phi\cos\psi - \cos\phi\sin\theta\sin\psi) \\ \dot{z}_f = -u\sin\theta + v\sin\phi\cos\theta + w\cos\phi\cos\theta \end{cases} \tag{4}$$

2.2. Autopilots construction

Because non-linear state models are challenging to handle, most of the early progress in understanding the dynamics of missiles and the motion stability was derived by studying linear small perturbation equations (Stevens et al., 2015). The state-space models could be obtained as follows:

Roll autopilot State-Space:

$$\begin{bmatrix} \dot{\phi} \\ \dot{p} \\ \dot{\xi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & L_p & L_\xi \\ 0 & 0 & -25 \end{bmatrix} \begin{bmatrix} \phi \\ p \\ \xi \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 25 \end{bmatrix} u_\xi$$
$$\begin{bmatrix} \phi \\ p \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \phi \\ p \\ \xi \end{bmatrix} \tag{5}$$

Pitch autopilot State-Space:

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \\ \dot{\eta} \end{bmatrix} = \begin{bmatrix} \frac{Z_\alpha}{u_0} & 1 & \frac{Z_\eta}{u_0} \\ M_\alpha & M_q & M_\eta \\ 0 & 0 & -25 \end{bmatrix} \begin{bmatrix} \alpha \\ q \\ \eta \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 25 \end{bmatrix} u_\eta$$
$$\begin{bmatrix} a_n \\ q \end{bmatrix} = \begin{bmatrix} l_pM_\alpha - Z_\alpha & l_pM_q & l_pM_\eta - Z_\eta \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ q \\ \eta \end{bmatrix} \tag{6}$$

The missile control system is divided into two categories, roll autopilot, and lateral acceleration autopilot. Roll; autopilot is adopted to maintain the missile roll angle at zero value to facilitate the guidance process. Lateral; autopilot is utilized to perform the guidance commands sent from guidance computer (Blakelock 1991). The roll autopilot block diagram is shown in Fig. 1.

The lateral autopilot contains two autopilots, normal, and lateral acceleration autopilots. As; the missile is an axis-symmetric airframe, both autopilots are identical (Blakelock 1991). The normal acceleration autopilot could be studied alone instead of studying both autopilots (will be denoted later as pitch autopilot). The; block diagram of the normal acceleration autopilot is shown in Fig. 2.

3. Pigeon inspired optimization (PIO) algorithm

The PIO algorithm is one of the bio-inspired swarm intelligence optimization algorithms which depend on their formulation of the behavior of pigeons' flocks (Mora et al., 2004). Several modifications are proposed for enhancing the PIO algorithm convergence by changing the tuning parameters or implementation procedure (DUAN et al., 2015). The predator–prey pigeon-inspired optimization (PPPIO) is proposed for integrated mission planning and controller design (Mohamed et al., 2017). Furthermore; Bloch quantum-behaved pigeon-inspired optimization (BQPIO) is a hybrid algorithm that integrating the PIO with quantum theory (Li and Duan 2014). PIO; is also utilized to tackle the air robot path planning optimization problem investigated through comparative experiments (Zhang et al., 2015). PIO; is also modified by integration with control parameterization and time discretization (CPTD) technique to handle UAV formation problems (Zhang et al., 2014). PIO algorithm could be extended in its applicability to other applications like DC motors control (Rajesh and Deepa 2020); spacecraft (Zhang and Duan 2015); control of landing systems (Deng and Duan 2016); and gliding trajectory (Zhao and Zhou 2015). Also, the PIO algorithm could be applied in the field of flying vehicle guidance to provide an optimum trajectory for vehicle flight. The PIO; algorithm is based on the map and compass operator and landmark models. *Map and compass operator* models are based on the earth's magnetic field and the sun with which the pigeons are utilized to find a flock in its long travel. For; pigeon, i with position vector X_i and speed vector V_i, its position and velocity in a D-dimension search space will be updated in each iteration. For pigeon i, its new position vector X_i and velocity vector V_i at the t^{th} iteration could be calculated by (7) and (8).

$$V_i(t) = V_i(t-1)\cdot e^{-Rt} + rand\cdot(X_g(t) - X_i(t-1)) \tag{7}$$

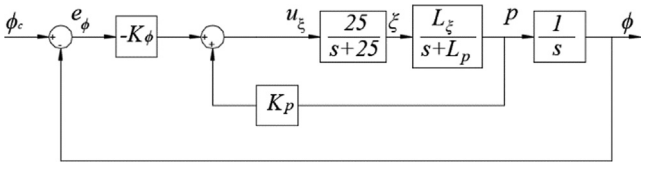


Fig. 1. Block diagram of roll autopilot.

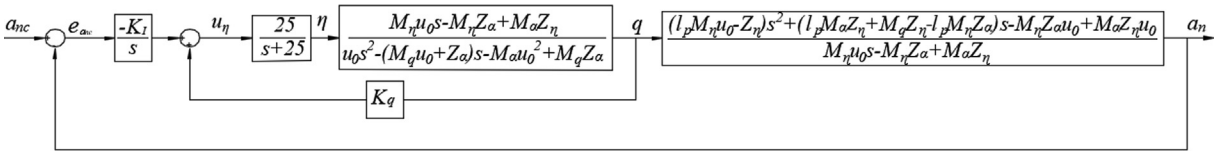


Fig. 2. Pitch autopilot block diagram.

$$X_i(t) = X_i(t-1) + V_i(t) \tag{8}$$

where R is the map and compass factor, *rand* is a random number, X_g is the current global best position, and X_g could be calculated by comparing the positions among all the pigeons.

Landmark operator is based on landmarks with which the pigeons use the memorized landmarks to find their way and fly back to their nest. This; reduces the number of pigeons in each new population by half. This; operator represents that the pigeons are still far from their destination and assumes they are not familiar with the landmarks. Let; $X_c(t)$ be the center of half of the pigeon's position at the t^{th} iteration, then suppose every pigeon could fly in a straight way to the destination X_c. The; position updating rule for pigeon i at the t^{th} iteration could be given by:

$$N_{pl}(t) = \frac{N_p(t-1)}{2} \tag{9}$$

$$X_c = \frac{\sum_{N_{pl}} X_i(t).fitness(X_i(t))}{\sum_{N_{pl}} fitness(X_i(t))} \tag{10}$$

$$X_i(t) = X_i(t-1) + rand.(X_c(t) - X_i(t-1)) \tag{11}$$

fitness (Xi(t)) is the quality of the individual pigeon. For the minimum optimization problems, we can choose:

$$fitness(X_i(t)) = \begin{cases} \frac{1}{f_{\min}(X_i(t))+\varepsilon} & \text{for minimization problems} \\ f_{\max}(X_i(t)) & \text{for maximization problems} \end{cases}$$

For; each pigeon, the optimal position of the Nc^{th} iteration can be denoted with X_p, and X_p = min (X_{i1}, X_{i2}, …, X_{iNc}). The role of the PIO algorithm should be identified. Both; controllers are used to control the vehicles by multiplying the feedback signals by certain gains. Because the missile model is a time-variant, these gains change with time. The; role PIO algorithm is to compute the optimum value for gains that will be used to control the missile motion. Fig. 3 shows the block diagram, which clarifies the role of the PIO algorithm in missile control operations.

3.1. Fitness function calculation

3.1.1. Fitness function calculation for pitch autopilot

The control law is formulated to provide a control action as:

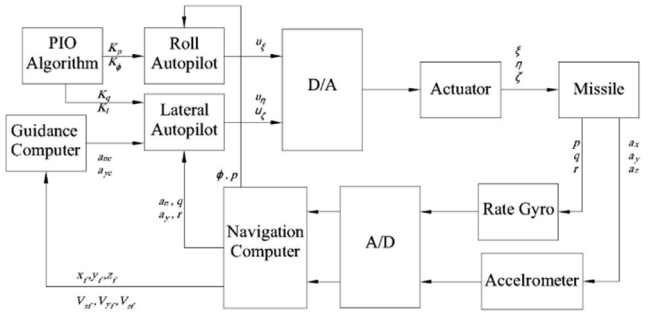


Fig. 3. Role of PIO algorithm in vehicle control.

$$u_\eta(\mathrm{n}) = -K_q \cdot q(n) + K_I \sum_{k=0}^{n} e_{a_n}(k)T \tag{12}$$

Then, as the minimal objective function, use temporal integration of the absolute square error value to minimize the control effort as much as feasible. The objective function is added to the control input square and the pitch rate square. The; following is the best parameter selection:

$$J = \int_0^\infty (k_1.q^2 + k_1.u_\eta^2 + k_3 e_{a_n}^2)dt \tag{13}$$

Where k_1, k_2, *and* k_3 represent the weights. The; rise time is added, as shown in Eq.(14), to improve the response that leads to decrease the settling time, which provides a faster response. The; punishing function is modified to prevent the overshoot. Overshoot; is incorporated into the optimum index once it arises. Then, the optimal indexes are as follows:

$$J = \begin{cases} \int_0^\infty (k_1.q^2 + k_1.u_\eta^2 + k_3 e_{a_n}^2)dt + k_4.t_r : e_{a_n}(t) < 0 \\ \int_0^\infty (k_1.q^2 + k_1.u_\eta^2 + k_3 e_{a_n}^2 + k_5.|e_{a_n}(t)|)dt + k_4.t_r : e_{a_n}(t) \geqslant 0 \end{cases}$$

Where t_r is the rise time, k_4 and k_5 are weighting parameters. Setting the weights to be [k_1 = 1, k_2 = 0.1, k_3 = 4, k_4 = 100, k_5 = 4].

3.1.2. Fitness function calculation for roll autopilot

The same procedure illustrated in the previous section is utilized for the roll channel. The; control law which is formulated to provide the control action is as follows:

$$u_\xi(\mathrm{n}) = K_p \cdot p(n) + K_\phi \mathrm{e}_\phi(\mathrm{n}) \tag{15}$$

The optimal parameter selection is:

$$J = \int_0^\infty (k_1 p^2 + k_1.u_\xi^2 + k_3 \mathrm{e}_\phi^2)dt \tag{16}$$

The optimal indexes are:

$$J = \begin{cases} \int_0^\infty (k_1.p^2 + k_1.u_\xi^2 + k_3 e_\phi^2)dt + k_4.t_r : e_\phi(t) < 0 \\ \int_0^\infty (k_1.p^2 + k_1.u_\xi^2 + k_3 e_\phi^2 + k_5.|e_\phi(t)|)dt + k_4.t_r : e_\phi(t) \geqslant 0 \end{cases} \tag{17}$$

where t_r is the rise time, k_4 and k_5 are weighting parameters. Setting the weights to be [k_1 = 1, k_2 = 0.1, k_3 = 4, k_4 = 200, k_5 = 4].

3.2. Steps of Pigeon-Inspired optimization implementation

The implementation of the optimization procedure is shown as a flowchart in Fig. 4.

4. Adaptive Pigeon-Inspired optimization (APIO) algorithm

A novel adaptive parametric setting is introduced by the Adaptive Pigeon-Inspired Optimization method (abbreviated as APIO). This; algorithm adapts the initial location of the population agents with the vehicle model to enhance the algorithm convergence by bringing the pigeons closer to the optimal solution (Nadia et al., 2020). Furthermore; the required effort to find the optimal solution is reduced to meet the real-time implementation constraints.

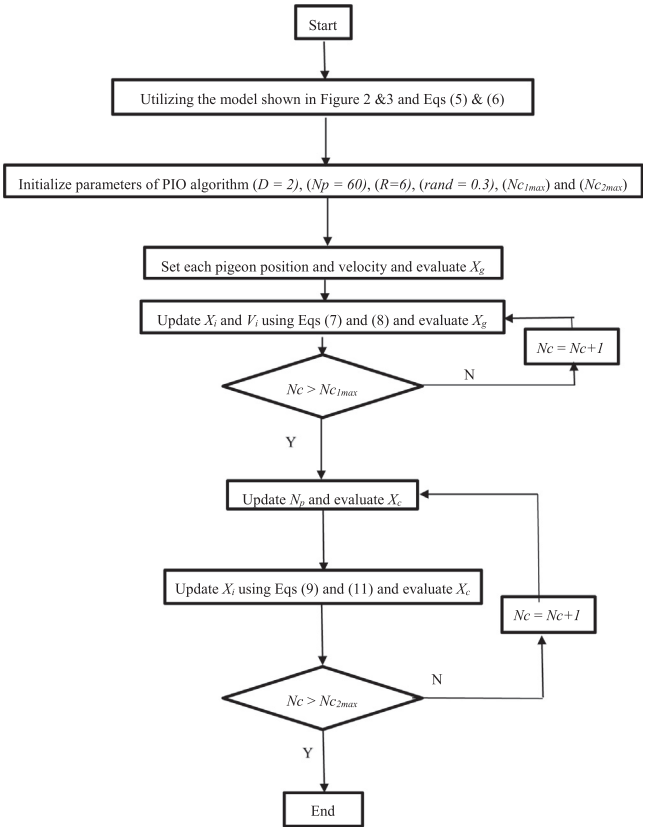


Fig. 4. PIO Implementation Procedure.

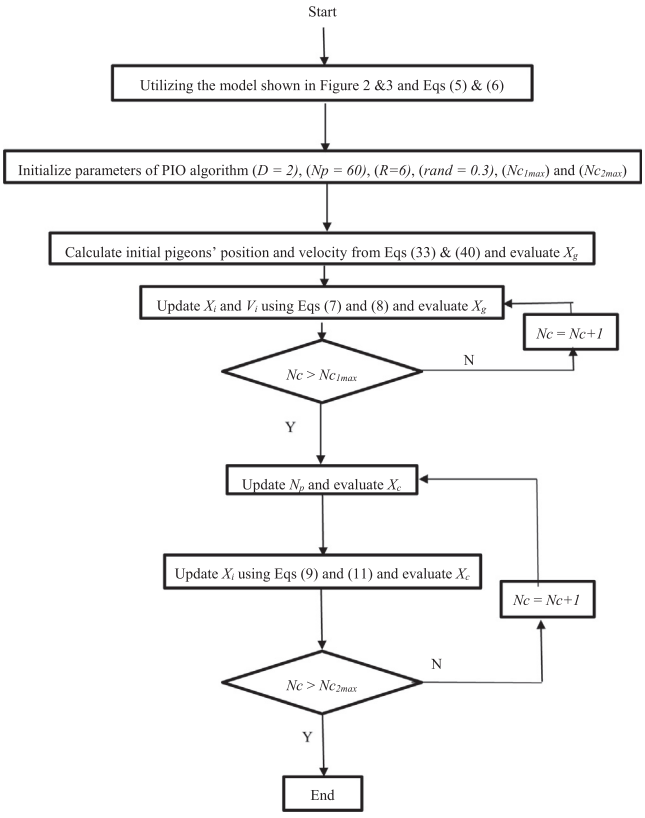


Fig. 5. Procedure of APIO algorithm.

Moreover; it generates pigeon position boundaries based on stability bounds calculated via various stability tests, the parameters of map and compass operator equations could be adapted to allow the pigeons' location to encompass the whole convergence field to find the local best solution. The; pigeons are placed in a starting location as in Eq.(18) to begin the algorithm initialization as follows:

$$X_i(0) = \begin{cases} K_l + (i-1)K_{step1} : 1 \leqslant i \leqslant \frac{N_p}{2} \\ K_{initial} + iK_{step2} : \frac{N_p}{2} + 1 \leqslant i \leqslant N_p \end{cases} \tag{18}$$

where

$$K_{step1} = \frac{K_{initial} - K_l}{\frac{N_p}{2} - 1}, K_{step2} = \frac{K_u - K_{initial}}{\frac{N_p}{2}} \tag{19}$$

Initializing the velocity with zero at each new cycle, so:

$$V_i(0) = 0 : 1 \leqslant i \leqslant N_p \tag{20}$$

4.1. Roll autopilot adaptation

In roll autopilot, the input signal is u_ξ, the state vector is $[\varphi, p, \xi]^T$, the command signal is φ_c. From; Fig. 1, the input signal will be:

$$u_\xi = K_p p + K_\phi \phi - K_\phi \phi_c = [K_\phi \quad K_p \quad 0] \begin{bmatrix} \phi \\ p \\ \xi \end{bmatrix} + [-K_\phi][\phi_c]$$

$$= K_1 X + K_2 u_c \tag{21}$$

The gains K_p and K_φ will be initialized as:

$$K_p = \frac{1}{L_\xi}\left(-L_p - \frac{1}{25}\left(50\omega_n\zeta - 2\omega_n\zeta L_p - 4\omega_n^2\zeta^2 + \omega_n^2\right)\right)$$
$$K_\phi = \frac{-\omega_n^2}{25L_\xi}\left(25 - L_p - 2\omega_n\zeta\right) \tag{22}$$

The gain bounds are investigated, utilizing Routh-Hurwitz criterion. The; resulting gain bounds are as follows:

$$K_p : 0 \to \infty$$
$$K_\phi : 0 \to (25 - L_p)\left(K_p + \frac{L_p}{L_\xi}\right) \tag{23}$$

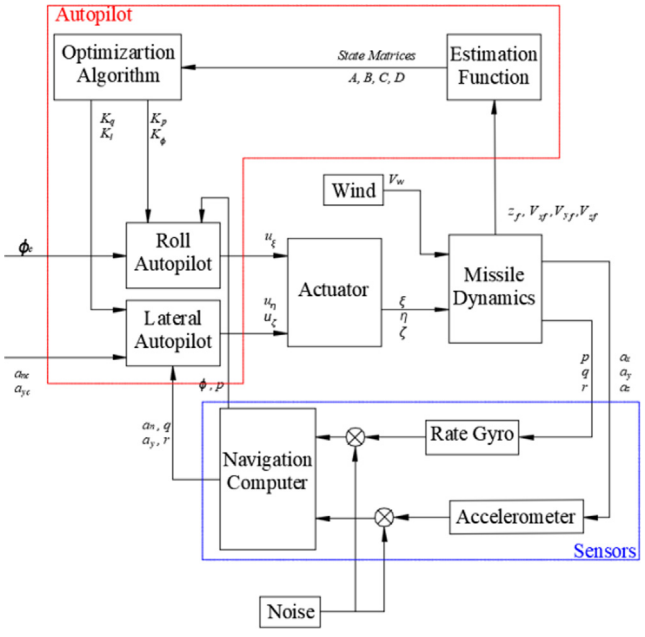
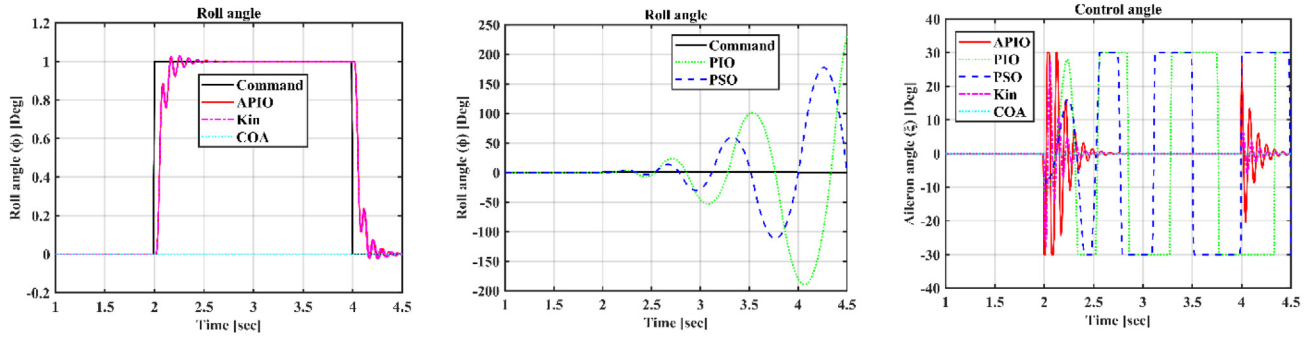


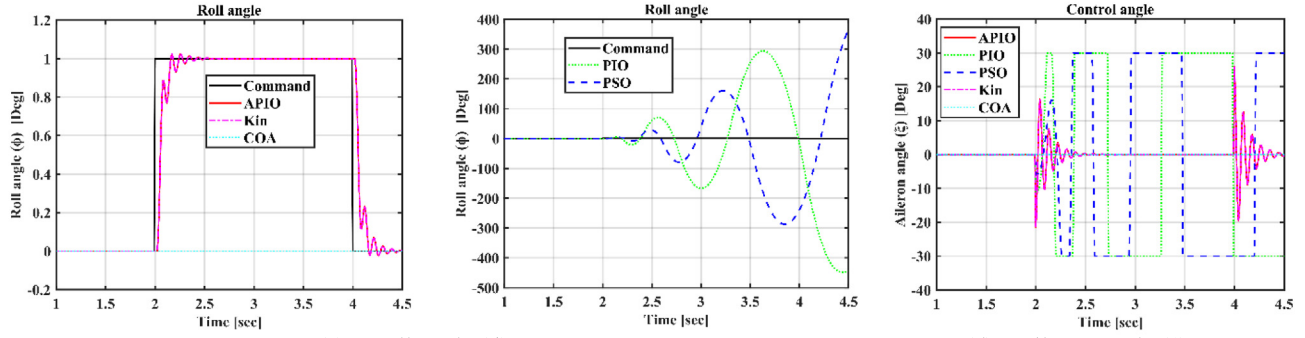
Fig. 6. Block diagram of the vehicle.

Point 1:



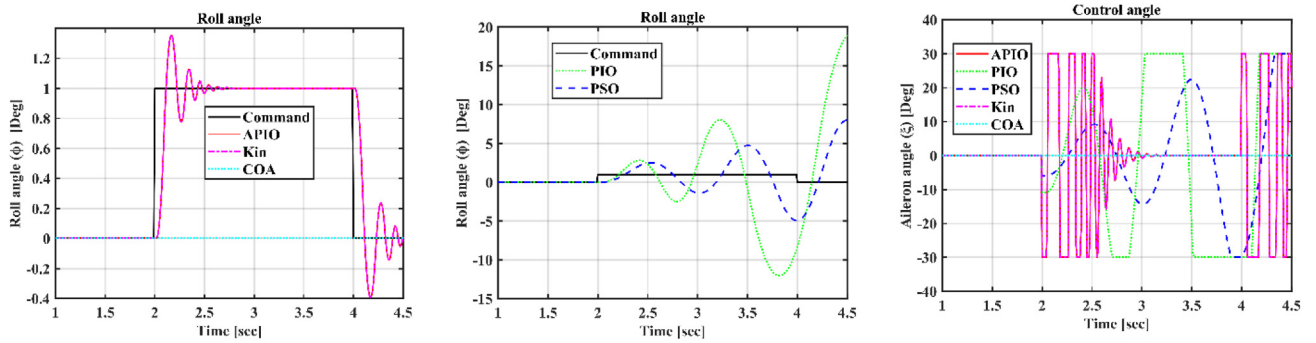
(a) Roll angle (ϕ) response (b) Aileron angle (ξ) response

Point 2:



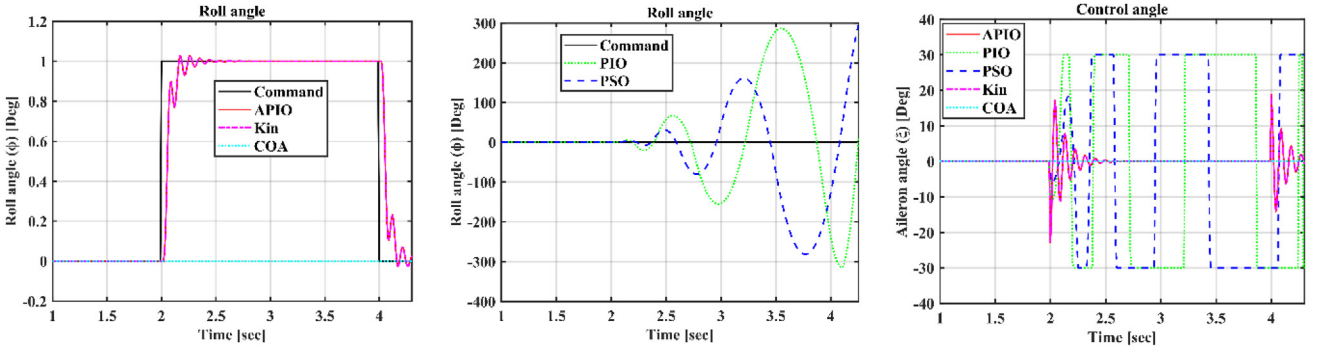
(c) Roll angle (ϕ) response (d) Aileron angle (ξ) response

Point 3:



(e) Roll angle (ϕ) response (f) Aileron angle (ξ) response

Point 4:



(g) Roll angle (ϕ) response (h) Aileron angle (ξ) response

Fig. 7. Roll Channel Results in case of absence of noise and disturbance.

The pigeon's position and velocity initialization are as in the following equation:

$$X_i(0)=\begin{cases}\begin{bmatrix}\frac{5+0.8i}{29L_\xi}\left(-L_p-\frac{1}{25}\left(50\omega_n\zeta-2\omega_n\zeta L_p-4\omega_n^2\zeta^2+\omega_n^2\right)\right)\\ \frac{-\omega_n^2(i-1)}{725L_\xi}(25-L_p-2\omega_n\zeta)+\frac{3-0.1i}{29}(25-L_p)\left(K_p+\frac{L_p}{L_\xi}\right)\end{bmatrix}:1\leqslant i\leqslant\frac{N_p}{2}\\ \begin{bmatrix}\frac{30+0.8i}{30L_\xi}\left(-L_p-\frac{1}{25}\left(50\omega_n\zeta-2\omega_n\zeta L_p-4\omega_n^2\zeta^2+\omega_n^2\right)\right)\\ 0.03i(25-L_p)\left(K_p+\frac{L_p}{L_\xi}\right)+\frac{(i-30)\omega_n^2}{750L_\xi}(25-L_p-2\omega_n\zeta)\end{bmatrix}:\frac{N_p}{2}+1\leqslant i\leqslant N_p\\ V_i(0)=0:1\leqslant i\leqslant N_p\end{cases} \tag{24}$$

4.2. Pitch autopilot adaptation

In lateral autopilot, the input signal is u_η; the state vector is $[\alpha,q,\eta,\lambda]^T$ where λ is an introduced variable representing the integration of the error signal and the command signal is a_{nc}. From; Fig. 2, the input signal will be:

$$\begin{aligned}u_\eta &= K_q q-K_I\int e_{a_n}dt=K_q q-K_I\lambda\\ &=\begin{bmatrix}0 & K_q & 0 & -K_I\end{bmatrix}\begin{bmatrix}\alpha\\ q\\ \eta\\ \lambda\end{bmatrix}+[0][a_{nc}]=K_1X+K_2u_c\end{aligned} \tag{25}$$

The initial position and velocity of the pigeons will be as follows:

$$X_i(0)=\begin{cases}\begin{bmatrix}\frac{5+0.8i}{29}(K_q)_{initial}\\ \frac{(i-1)}{29}(K_I)_{initial}+\frac{3-0.1i}{29(T_{22}-A_1T_{12})}(A_1B_1-D_1-(T_{21}-A_1T_{11})K_q)\end{bmatrix}:1\leqslant i\leqslant\frac{N_p}{2}\\ K_{initial}+\begin{bmatrix}\frac{30+0.8i}{30}(K_q)_{initial}\\ \frac{0.03i}{(T_{22}-A_1T_{12})}(A_1B_1-D_1-(T_{21}-A_1T_{11})K_q)-\frac{30-i}{30}(K_I)_{initial}\end{bmatrix}:\frac{N_p}{2}+1\leqslant i\leqslant N_p\\ V_i(0)=0:1\leqslant i\leqslant N_p\end{cases} \tag{26}$$

4.3. Steps of adaptive Pigeon-Inspired optimization implementation

The implementation procedure of the proposed optimization algorithm is presented in the flowchart, as shown in Fig. 5.

4.4. Complexity of adaptive Pigeon-Inspired optimization

The input length does not vary between iterations with the map and compass operators, and data is handled as an array structure. As; a result, the map and compass operator's complexity on a single iteration is $O(DNp)$, because the algorithm's input length is the number of pigeons. N_p; is multiplied by D, which is the search space. Regarding; the landmark operator, as the input length and data are handled as an array, a quick sorting for the input is potentially affects the time complexity to the worst-case condition. The; complexity of the landmark operator in one iteration is $O(DN_p + {N_p}^2)$. Since; the number of iterations is N_{c1max} for the map and compass operator and N_{c1max} for landmark operator, the overall algorithm complexity can be calculated as the sum of the previous operator's complexities as $O(DN_p\,N_{c1max}+(DN_p+{N_p}^2)N_{c2max})$. The; iteration process' computational complexity does not vary in the APIO algorithm; the initialization phase complexity is only added. The initialization; procedure has a computational complexity of $(3D + 2DN_p)$ to calculate the initial value and upper and lower limits. Moreover; the calculation of initial pigeons' position and velocity are included. Therefore; the computational complexity of APIO algorithm is $O(DN_p\,N_{c1max} + (DN_p + {N_p}^2)\,N_{c2max} + 3D + 2DN_p)$.

5. Simulation

5.1. Simulation construction

To begin the building of the simulation, all the tools required to embark on the six degrees of freedom simulation must be assembled. Model; equations in (1), (2), (3), and (4), autopilots for the pitch and roll channels were engaged with PIO, PSO, and APIO algorithms and without utilizing the adaptation algorithm alone with performing optimization with initial values of gains (Kin) all together are to engage. Coyote; Optimization Algorithm (COA) (Pierezan et al., 2019); for Global Optimization a state-of-the-art optimization technique is involved to compare APIO behavior with the COA technique. This; part is devoted to evaluating the designed autopilots using a 6-DOF flight simulation model and constructing a simulation on a Matlab environment, including the feedback sensors, sensor noise, atmospheric disturbance, actuators, and autopilots. The; simulation is performed using Matlab/Simulink 2017b environment to achieve optimization and control. Fig. 6; shows the block diagram of the simulation with the module blocks.

The Dynamic Pressure increases at the beginning of the flight due to fuel consumption till (t = 13.05 *sec*), then decreases due to fuel burnout till the peak, and then increases at decent. The; wind velocity is determined in an inertial frame and transformed to the body frame, then added to the vehicle velocity. The; resultant force of the airframe is calculated due to the resultant velocity. The; wind model is shown in (27), with a wind velocity vector of value:

$$\vec{V}_w=\begin{bmatrix}5.66\\ 5.66\\ 0\end{bmatrix}[m/s]\Rightarrow\begin{bmatrix}V_{wx}\\ V_{wy}\\ V_{wz}\end{bmatrix}_f=\begin{cases}\vec{V}_w(1-\cos(\frac{\pi(t-15)}{2})):15\leqslant t\prec 16\\ \vec{V}_w\ :16\leqslant t\prec 19\\ \vec{V}_w(1-\cos(\frac{\pi(20-t)}{2})):19\leqslant t\prec 20\\ \vec{0}\ :otherwise\end{cases} \tag{27}$$

Noise block introduces a uniform noise added with the feedback sensors output. The; noise model of bounds ± 0.014 and zero mean value and variance (6.54×10^{-5}).

5.2. Simulation application

To apply the proposed algorithms to the vehicle in its nonlinear form model with its complexity and stochastic calculation, the algorithm should be used by selecting specific points at the dynamic pressure response shown in Fig. 7 and applying the algorithm to reach optimum gains for the autopilot. Depending; on this, the selection will be based on choosing two points during the action of rocket propulsion and two points after shut-off.

Table 1
Values of stability derivatives at checkpoints.

Point	t	L_p	L_ξ	Z_α	M_α	Z_η	M_η	M_q	u_o	l_p
1	6	−71	−4.965	−1.696	−2.7919	−0.984	−1.997	−0.1834	469	2.218
2	13	−74.1	−14.395	−6.091	−6.751	−2.0933	−3.3076	−0.188	1053	1.971
3	90	−0.583	−0.0689	−0.0203	−0.0238	−0.00978	−0.01533	−0.00127	692	1.878
4	170	−35.056	−6.303	−2.213	−2.447	−0.7607	−1.1943	−0.0798	933	1.878

The; two most convenient points during rocket motor action at the beginning of control system action are at the six second and shut-off. After; the peak point is selected near the flight ends. Therefore; the checkpoints will be at times (6 *sec* – 13 *sec* – 90 *sec* – 170 *sec*). Table 1; gives the value of stability derivatives at these points.

5.3. Simulation results

Applying unit step input signal during 2 s for both commanded roll angle and a_n investigation of response for two cases; absence of noise and wind and existence of noise and wind. Simulating; the vehicle with the wind effect, as shown in Fig. 7 and Fig. 8 for roll

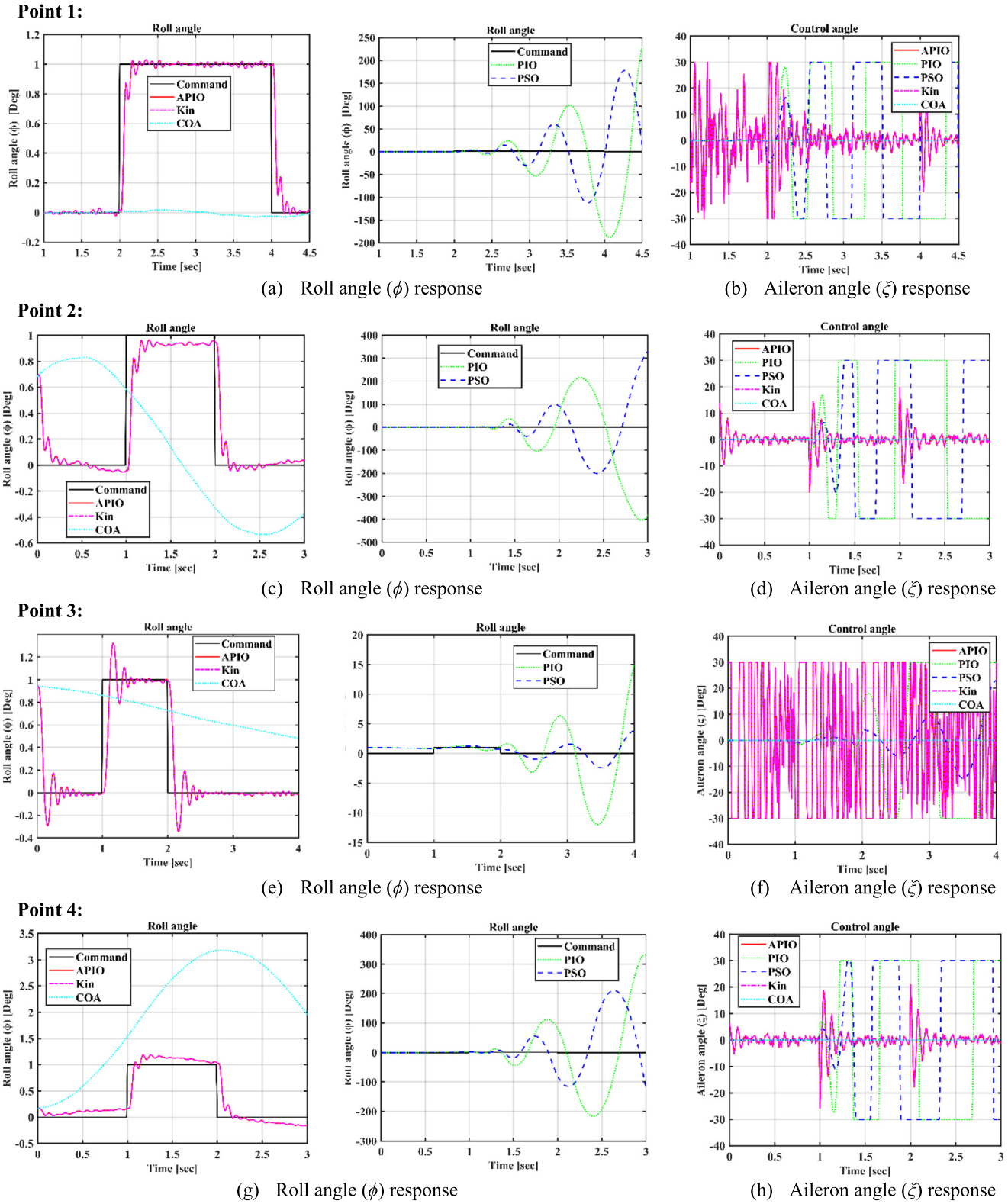
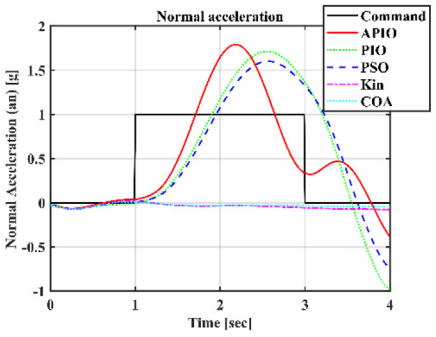
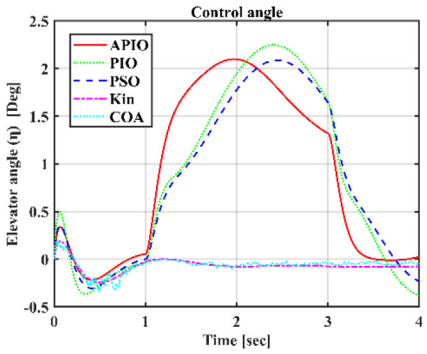


Fig. 8. Roll Channel Results in case of the existence of noise and disturbance.

Point 1:

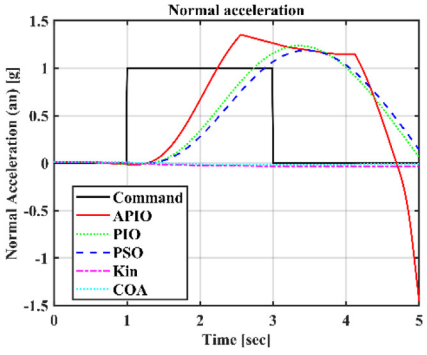


(a) Normal acceleration (a_n) response

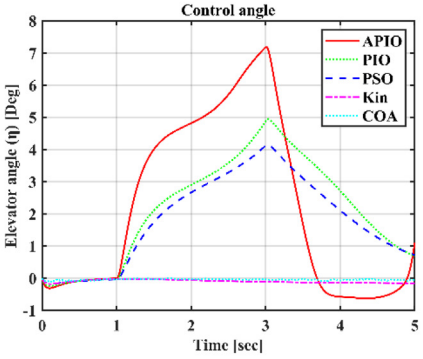


(b) Elevator angle (η) response

Point 2:

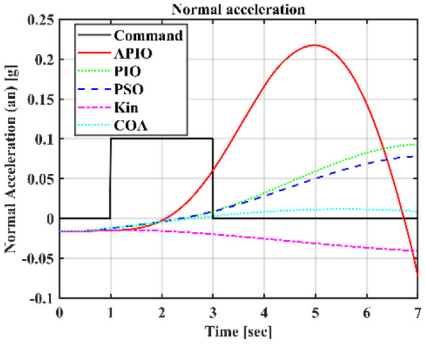


(c) Normal acceleration (a_n) response

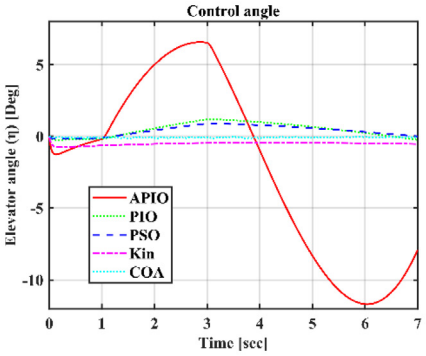


(d) Elevator angle (η) response

Point 3:

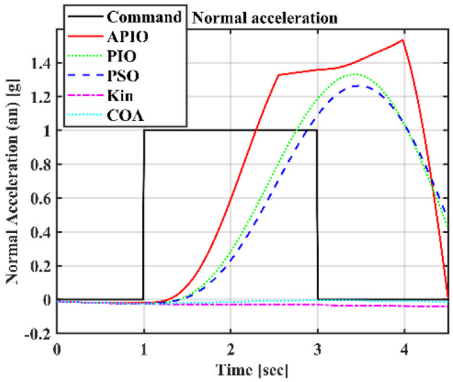


(e) Normal acceleration (a_n) response

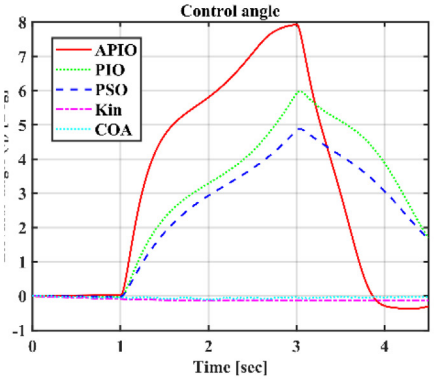


(f) Elevator angle (η) response

Point 4:



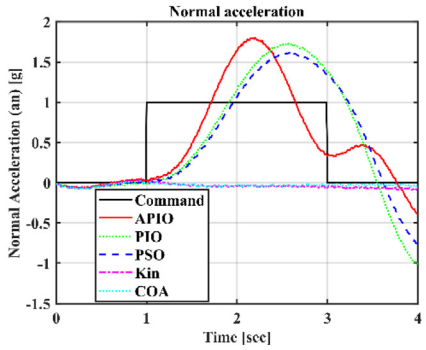
(g) Normal acceleration (a_n) response



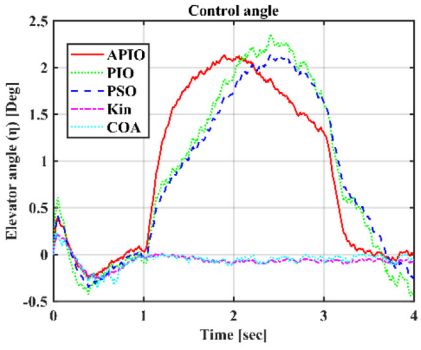
(h) Elevator angle (η) response

Fig. 9. Pitch Channel Results in case of absence of noise and disturbance.

Point 1:

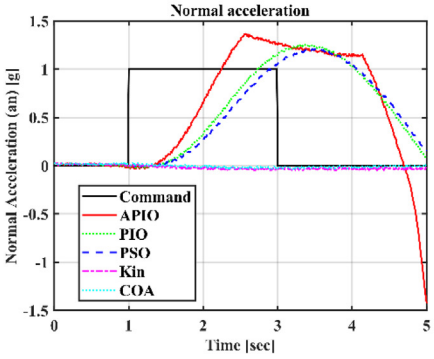


(a) Normal acceleration (a_n) response

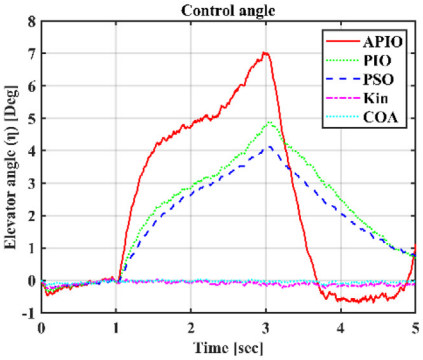


(b) Elevator angle (η) response

Point 2:

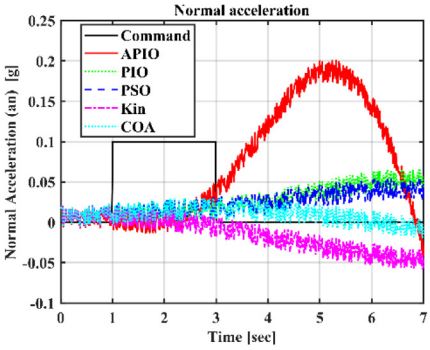


(c) Normal acceleration (a_n) response

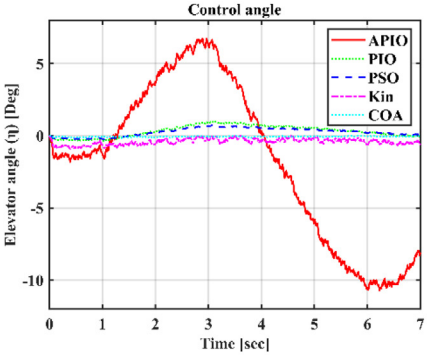


(d) Elevator angle (η) response

Point 3:

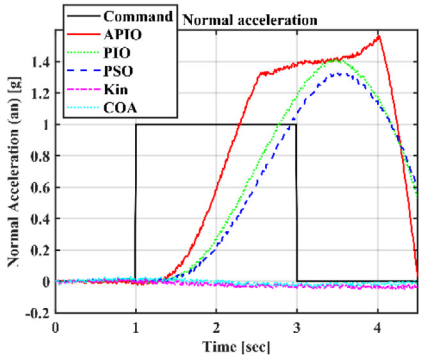


(e) Normal acceleration (a_n) response

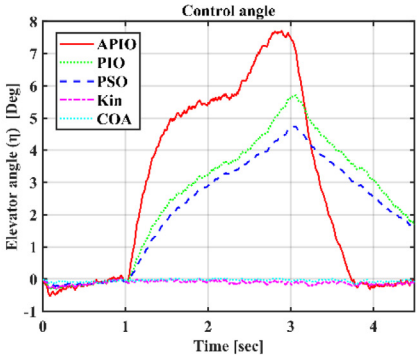


(f) Elevator angle (η) response

Point 4:



(g) Normal acceleration (a_n) response



(h) Elevator angle (η) response

Fig. 10. Pitch Channel Results in case of the existence of noise and disturbance.

autopilot and Fig. 9 and Fig. 10 for pitch autopilot. The; input signal for roll autopilot is a roll angle square signal with amplitude (1 [deg]) for two seconds, and for pitch autopilot is the normal acceleration square signal with amplitude (1 [g]) for two seconds except at point 3, which will be of amplitude (0.1 [g]) for two seconds.

(a) APIO algorithm in Fig. 7(a), (c), (e), and (g) successfully provided the required performance as the autopilot tracked the roll angle command released from the roll controller with a rise time about 0.15 *sec*, in contrast, both PIO and PSO failed to make the autopilot conduct a stable performance for roll angle.
(b) In Fig. 7(b), (d), (f), and (h), APIO produced a control angle within angle limits without exceeding limits except for point 3 due to lower density at the trajectory peak, and in turn, the dynamic pressure will be lower, causing the rolling moment damping derivative L_p to be low, and the response produces high overshoot (about 32 %) and APIO consumes a great effort to track the commanded signal.
(c) Applying the adaptation algorithm alone without optimization in the simulation of initial gains (Kin) gives almost the same performance in all parameters, which reflects that optimization has not presented a significant effect on the gains calculated from the adaptation algorithm.
(d) APIO applies a control angle to track the commanded signal, whereas, both PIO and PSO are attempting to apply control angle to track the input signal with no use till reaching a control angle limits.
(a) In Fig. 8(a), the wind disturbance effect has not started, but the noise alone has no noticeable effect on roll angle response due to the roll channel low-frequency nature and a remarkable fluctuation in control angle response at point 1 and 3, i.e., at low dynamic pressure in Fig. 8(b) and (f).
(b) In Fig. 8(d) and (h), the noise causes slight fluctuations in the control angle due to the high dynamics of the vehicle, and these fluctuations appear clearly in the control angle response resulting from the APIO algorithm. Which; means that the high dynamics of the vehicle repel noise affecting the sensor readings.
(c) In Fig. 8(b) and (g), the response of APIO tracked the command signal but failed to reach a steady-state response during the action of the control signal due to the wind effect.
(d) In Fig. 8(a), (c), (e), and (g), noise and wind exerted on both PIO and PSO enlarged the fluctuation in response, which resulted from an increase of reduction in stability due to noise sensitivity and the existence of wind.
(a) In Fig. 9(a), (c), (e), and (g), APIO was tracking reference commanded signal faster than both PIO and PSO, i.e., rise time resulted from APIO is the least but with greater overshoot, revealing that the adaptation algorithm has improved the performance also in pitch autopilot.
(b) Also, applying only initial values for gains (Kin) results in a slow response for the normal acceleration, and this reveals that there were other optimum values for pitch autopilot gains inside the region of stability and herein comes the role of the APIO algorithm that utilized optimization to reach this solution.
(c) In Fig. 9(b), (d), (f), and (h), APIO applied a greater control angle to perform a faster response, but this resulted in a high overshoot (25 %-100 %) at all points.
(a) In Fig. 10(a), (c), and (g), noise and disturbance did not present any noticeable change in the response of normal acceleration or other parameters except at point 3 (Fig. 11(e)) due to low dynamic pressure at the peak.
(b) In Fig. 10(a), applying only initial values for gains (Kin) is still resulting a slow response for the normal acceleration, and this reveals that there were other optimum values for pitch autopilot gains inside the region of stability, and herein comes the role of APIO algorithm that utilized optimization to reach this solution.
(c) In Fig. 10(b), (d), (f) and (h), the noise effect is remarkable in the form of a high-frequency fluctuations in the control angle response.

Statistical analysis is carried out utilizing the Friedman test (Derrac et al., 2014); for analyzing the error between commanded and tracked roll and pitch angles. It; involves a variance-based analysis by ranking the test data, as the lower rank indicates higher performance. The; main test evaluation parameters are the p-value to demonstrate the difference between the utilized testing data and the mean rank, which measures the performance level of the utilized datasets candidates.

A comparative statistical analysis is accomplished between the proposed (APIO) algorithm and PIO, PSO, Kin, and COA algorithms

Table 2
Statistics and Mean ranking coefficient of Friedman's test for each algorithm.

	Without disturbance			With disturbance	
	Method	Mean Rank		Method	Rank
Point 1	APIO	2. 886	Point 1	APIO	2. 876
p-value = 0.00632	Kin	2.9	p-value = 0.02095	Kin	2. 936
	PIO	3.126		PIO	3.112
	PSO	2.9		PSO	2.936
	COA	3.188		COA	3.14
Point 2	APIO	2.792	Point 2	APIO	2.755
p-value = 0.0086	Kin	2.934	p-value = 0.00068	Kin	3.0175
	PIO	2.898		PIO	3.0675
	PSO	3.188		PSO	2.755
	COA	3.168		COA	3.405
Point 3	APIO	2.924	Point 3	APIO	1.335
p-value = 0.00509	Kin	3.03	p-value = 0.00028	Kin	3.5825
	PIO	2.986		PIO	3.2925
	PSO	2.924		PSO	3.385
	COA	3.136		COA	3.383
Point 4	APIO	2.6267	Point 4	APIO	1.81
p-value = 0.00025	Kin	3.0289	p-value = 0.00037	Kin	2.9425
	PIO	2.86		PIO	3.3575
	PSO	3.2422		PSO	3.445
	COA	3.242		COA	3.445

utilizing the presented non-linear missile model for roll and pitch channels without considering the action of control signals. The; test was applied to the four points of the trajectory previously selected, and the results are summarized in Table 2. The; results of the pitch and roll channels are augmented for real evaluation due to the coupling between both channels. As; shown in Table 2, the (p-value) is less than 0.05, which indicates a higher degree of difference between the utilized datasets; particularly in point-4, the (p-value) is 0.00025 and 0.00037 for the nominal and disturbed trajectory, respectively. Further; the mean rank demonstrates the proposed algorithm's performance for accurately tracking the demanded trajectory in the roll and pitch channels. The; results of PSO and Kin algorithms are close to the results of the proposed algorithm by a difference of around 0.014 at point 1 and point 2, but it presents a lower performance at point 3 and point 4 by a difference of around 0.4 due to high dynamics as well as the degree of disturbance which reflects the proposed algorithm efficiency and fast convergence rate for selecting the optimal gains for the roll and pitch autopilots.

Finally, the APIO algorithm has accomplished the required task, but like pigeon-inspired optimization, APIO has some drawbacks in the application. APIO; has treated the disadvantages of PIO, but these disadvantages still exist. These; appear in the complex implementation of the APIO algorithm despite the complexity analysis. Also; like PIO and APIO could be trapped into absolute minima or maxima, and the iterations consume too much time till ending. APIO; has treated the drawback of the PIO algorithm in the shortage of information of PIO about the system to get complete details about the linear model of the system obtained and acquired from the APIO algorithm.

6. Conclusion

In this paper, a tactical missile control system is designed, and the optimum values of controller gains are calculated using an adaptive modified PIO algorithm based on a parametric setting to enhance the algorithm robustness against model parametric variations and speed up the convergence rate and finally, to reduce the algorithm complexity, reduce the required execution time and to meet the constraints of real-time implementation. Furthermore; a comparative analysis is carried out for the proposed algorithm with the original PIO, Kin, COA, and PSO algorithms utilizing the presented non-linear missile model for roll and pitch channels without considering the action of control signals. Model; parameters variation with time is handled by incorporating an estimation function into the original PIO algorithms to overcome the problems of dealing with time-varying systems. PIO; and PSO algorithms both performed unstable behavior with non-linear vehicle model in roll autopilot response. In contrast, the proposed APIO algorithm and simulation of adaptation algorithm without optimization both have performed a stable and robust performance with a rise time around 0.15 *sec*. The; existence of noise and disturbance did not present a remarkable effect on roll angle response and normal acceleration response. PIO; algorithm, PSO algorithm, and APIO algorithm have performed a stable behavior in normal acceleration response when engaged with pitch autopilot. In contrast, the adaptation algorithm, when engaged alone with pitch autopilot, didn't present the required performance characteristics, and failed to reach the commanded normal acceleration. This; clarifies the importance of the optimization algorithm in calculating optimum gains for the normal acceleration control in pitch autopilot. APIO; algorithm, when engaged with pitch autopilot, provided a faster performance than PIO, PSO, and COA algorithms, as the rise time for the APIO algorithm is less than the rise time for the others by around 30 % enhancement. Although; higher overshoot occurring in normal acceleration response around 25 % higher in APIO algorithm, but its effect can be negligible according to the system stability constraints.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Abadi, D.N.M., Khooban, M.H., 2015. Design of optimal Mamdani-type fuzzy controller for nonholonomic wheeled mobile robots. J. King Saud Univ.-Eng. Sci. 27 (1), 92–100. https://doi.org/10.1016/j.jksues.2013.05.003.

Alharkan, I., Saleh, M., Ghaleb, M.A., et al., 2020. Tabu search and particle swarm optimization algorithms for two identical parallel machines scheduling problem with a single server. J. King Saud Univ.-Eng. Sci. 32 (5), 330–338. https://doi.org/10.1016/j.jksues.2019.03.006.

Apkarian, P., Gahinet, P., Becker, G., 1995. Self-scheduled H∞ control of linear parameter-varying systems: a design example. J. Automatica. 31 (9), 1251–1261. https://doi.org/10.1016/0005-1098(95)00038-X.

Blakelock, J.H., 1991. Automatic control of aircraft and missiles. John Wiley & Sons.

Deng, Y., Duan, H., 2016. Control parameter design for automatic carrier landing system via pigeon-inspired optimization. J. Nonlinear Dyn. 85 (1), 97–106. https://doi.org/10.1007/s11071-016-2670-z.

Derrac, J., García, S., Hui, S., et al., 2014. Analyzing convergence performance of evolutionary algorithms: a statistical approach. Inf. Sci. 289, 41–58. https://doi.org/10.1016/j.ins.2014.06.009.

Duan, H., Qiao, P., 2014. Pigeon-inspired optimization: a new swarm intelligence optimizer for air robot path planning. Int. J. intell. Comput. Cybern. 7 (1), 24–37. https://doi.org/10.1108/IJICC-02-2014-0005.

Duan, H., Qiu, H., Fan, Y., 2015. Unmanned aerial vehicle close formation cooperative control based on predatory escaping pigeon-inspired optimization. J. Scientia Sinica Technologica. 45 (6), 559–572. https://doi.org/10.1360/N092015-00125.

Elnaggar, A., Khalil, K., 2016. The response of nonlinear controlled system under an external excitation via time delay state feedback. J. King Saud Univ.-Eng. Sci. 28 (1), 75–83. https://doi.org/10.1016/j.jksues.2014.01.003.

Jiang, J., 2012. Guidance Principle of Missiles. Beijing University of Aero. and Astr Press, Beijing, China.

Kamboj, V.K., Nandi, A., Bhadoria, A., et al., 2020. An intensify Harris Hawks optimizer for numerical and engineering optimization problems. Appl. Soft Comput. 89, 106018. https://doi.org/10.1016/j.asoc.2019.106018.

Kamen, E., Bullock, T., Song, C.-H., 1988. Adaptive control applied to missile autopilots. In: 1988 American Control Conference, IEEE.

Li, H., Duan, H., 2014. Bloch quantum-behaved Pigeon-inspired optimization for continuous optimization problems. In: Proceedings of 2014 IEEE Chinese Guidance, Navigation and Control Conference, IEEE.

Mohamed, M.S., Duan, H., 2020. Flight control system design using adaptive pigeon-inspired optimisation. Int. J. Bio-Inspired Comput. 16 (3), 133–147. https://doi.org/10.1504/IJBIC.2020.111266.

Mohamed, M.S., Duan, H., Fu, L., 2017. Flying vehicle longitudinal controller design via prey-predator pigeon-inspired optimization. In: 2017 IEEE Symposium Series on Computational Intelligence (SSCI), IEEE.

Mora, C.V., Davison, M., Wild, J.M., et al., 2004. Magnetoreception and its trigeminal mediation in the homing pigeon. Nature 432 (7016), 508–511. https://doi.org/10.1038/nature03077.

Nadia, A.-R., Isa, N.A.M., Desa, M.K.M., 2020. Efficient single and dual axis solar tracking system controllers based on adaptive neural fuzzy inference system. J. King Saud Univ.-Eng. Sci. 32 (7), 459–469. https://doi.org/10.1016/j.jksues.2020.04.004.

Pierezan, J., Maidl, G., Yamao, E.M., et al., 2019. Cultural coyote optimization algorithm applied to a heavy duty gas turbine operation. Energy Convers. Manage. 199, 111932. https://doi.org/10.1016/j.enconman.2019.111932.

Rajesh, R., Deepa, S., 2020. Design of direct MRAC augmented with 2 DoF PIDD controller: an application to speed control of a servo plant. J. King Saud Univ.-Eng. Sci. 32 (5), 310–320. https://doi.org/10.1016/j.jksues.2019.02.005.

Shamma, J.S., Cloutier, J.R., 1993. Gain-scheduled missile autopilot design using linear parameter varying transformations. J. Guidance, Control. Dyn. 16 (2), 256–263. https://doi.org/10.2514/3.20997.

Shayanfar, H., Gharehchopogh, F.S., 2018. Farmland fertility: a new metaheuristic algorithm for solving continuous optimization problems. Appl. Soft Comput. 71, 728–746. https://doi.org/10.1016/j.asoc.2018.07.033.

Siouris, G.M., 2004. Missile guidance and control systems. Springer Science & Business Media.

Stevens, B.L., Lewis, F.L., Johnson, E.N., 2015. Aircraft control and simulation: dynamics, controls design, and autonomous systems. John Wiley & Sons.

Zarchan, P., 2019. Tactical and Strategic Missile Guidance. American Institute of Aeronautics and Astronautics Inc..

Zhang, S., Duan, H., 2015. Gaussian pigeon-inspired optimization approach to orbital spacecraft formation reconfiguration. Chinese J. Aeronautics. 28 (1), 200–205. https://doi.org/10.1016/j.cja.2014.12.008.

Zhang, X., Duan, H., Yang, C., 2014. Pigeon-inspired optimization approach to multiple UAVs formation reconfiguration controller design. In: Proceedings of 2014 IEEE Chinese Guidance, Navigation and Control Conference, IEEE.

Zhang, B., Duan, H., 2015. Three-dimensional path planning for uninhabited combat aerial vehicle based on predator-prey pigeon-inspired optimization in dynamic environment. J IEEE/ACM Trans. Comput. Biol. Bioinformatics 14 (1), 97–107. https://doi.org/10.1109/TCBB.2015.2443789.

Zhao, J., Zhou, R., 2015. Pigeon-inspired optimization applied to constrained gliding trajectories. J. Nonlinear Dyn. 82 (4), 1781–1795. https://doi.org/10.1007/s11071-015-2277-9.

Zhu, W., Duan, H., 2015. Predator-prey biogeography-based optimization for parameters identification of UCAV flight control system. Aircraft Eng. Aerospace Technol.: Int. J. 87 (3), 249–259. https://doi.org/10.1108/AEAT-06-2013-0112.