Journal of Research and Applications in Mechanical Engineering
ISSN: 2229-2152 (Print); 2697-424x (Online)
(2025) Vol. 13, No. 3, Paper No. JRAME 25-13-041
[DOI: 10.14456/ jrame.2025.41]

Research Article

Many-Objective Optimization of Nonlinear Dynamic Inversion Altitude Controller Using Metaheuristics

N. Ruenruedeepan
N. Pholdee*
N. Panagant
S. Bureerat
Sustainable and Infrastructure Research and Development Center, Department of Mechanical Engineering, Faculty of Engineering, Khon Kaen University, Khon Kaen, 40002, Thailand

Received 10 November 2024
Revised 26 December 2024
Accepted 31 December 2024

Abstract:
Effective control of nonlinear dynamics under varying conditions is challenging in aerospace systems, especially for altitude regulation. Traditional linear controllers like PID and LQR are simple but struggle with complex flight dynamics and changing conditions, such as airspeed and density variations. Nonlinear Dynamic Inversion (NDI) offers a solution by transforming nonlinear dynamics into a linearized form, allowing the use of linear control techniques. However, tuning NDI parameters is complex and time-consuming with traditional methods. Metaheuristic algorithms provide a robust alternative, efficiently exploring large solution spaces for near-optimal tuning. This study compares various metaheuristics for optimizing the parameters of NDI controllers in altitude control, assessing stability, responsiveness, and robustness. Results indicate that Success-History Based Adaptive Differential Evolution (SHADE) with Linear Population Size Reduction (L-SHADE) is the most effective algorithm for NDI controller optimization, delivering optimal control gains across varying conditions.

Keywords: *Nonlinear dynamic inversion controller, Metaheuristic, Altitude control, Many objective optimization.*

1. Introduction

Effective control of systems with nonlinear dynamics and varying operational conditions presents a persistent challenge in modern control engineering, particularly in the field of aerospace [1]. Precise altitude control is crucial for ensuring flight stability, safety, and overall performance, especially when subject to fluctuating environmental conditions such as changes in airspeed and air density [2]. Altitude control serves as a prominent example where nonlinearities in aerodynamic behaviour can significantly affect control performance. Given the wide range of operating conditions, traditional linear control strategies often prove inadequate for maintaining precise altitude regulation [3]. Widely used linear controllers, such as Proportional-Integral-Derivative (PID) controllers [4] and Linear Quadratic Regulators (LQR) [5], are favoured for their simplicity and ease of implementation. However, these controllers frequently struggle to manage highly nonlinear system dynamics effectively. While H_∞ control [6] approaches have been developed to handle system uncertainties and manage some degree of nonlinearity, their performance tends to degrade under extreme nonlinear conditions. Nonlinear Dynamic Inversion (NDI) has emerged as a potential solution for such scenarios, providing a method to linearize nonlinear dynamics by effectively "inverting" them [7].

Nonlinear Dynamic Inversion (NDI) is a control strategy that linearizes nonlinear system dynamics, enabling the application of conventional linear control techniques [8]. This method is particularly valuable in aerospace, where

* Corresponding author: N. Pholdee
E-mail address: nantiwat@kku.ac.th

nonlinearities arise from factors such as aerodynamic forces, changing altitudes, and varying flight speeds. NDI has been effectively employed in the control of high-performance aircraft, offering improved manoeuvrability and stability. A notable example is its use in the control system of the F-16 fighter jet, where NDI has significantly enhanced pitch and roll rate control across a range of operational conditions [9]. Despite its advantages, the successful implementation of NDI depends on precise tuning of control parameters, a task that continues to present substantial challenges.

Optimizing controller parameters in NDI-based systems is essential for achieving high performance, particularly in altitude control. Traditional manual tuning methods, like trial-and-error or expert-guided approaches, are often time-consuming and can possibly fail to yield optimal results. Furthermore, because of the high dimensionality and complexity of nonlinear systems, conventional techniques, such as pole placement, can be insufficient and often demand detailed system modeling. In spacecraft control, for instance, manually tuning attitude control parameters can extend development times and reduce control efficiency. This issue is especially significant in nonlinear systems, where even small changes in the operating environment can markedly affect performance [10].

To address these limitations, metaheuristic optimization methods offer a powerful solution. These techniques are specifically designed to solve complex problems, excelling in handling nonlinearity, multimodal optimization, and uncertain systems. By employing a population-based concept and stochastic operators, metaheuristics enable global exploration and robustness [11–16]. With advancements in computational power, these methods provide automated and intelligent search processes that facilitate the discovery of optimal parameters for nonlinear controllers [17]. This capability ensures superior performance and robustness in complex nonlinear systems, making metaheuristics an indispensable tool for modern control design.

This work focuses on using metaheuristic algorithms to tune controller parameters in NDI-based control systems, aiming to develop an optimal flight controller across the full flight envelope. A multi-objective optimization problem is formulated for NDI flight controller tuning across various flight conditions, with the weighted sum technique employed to scalarize such multiple objectives. Metaheuristics (MHs) such as Particle Swarm Optimization (PSO) [18], Differential Evolution (DE) [19], Self-Adaptive Differential Evolution (JADE) [20], Success-History Based Adaptive Differential Evolution (SHADE) [21], and SHADE with Linear Population Size Reduction (L-SHADE) [22] provide robust solutions for tuning controller parameters by efficiently exploring large solution spaces and identifying near-optimal solutions. These algorithms have proven particularly effective in optimizing control gains for complex systems, as they are capable of handling multiple objective functions and constraints. In the context of altitude control, metaheuristics have been successfully applied to the optimization of controller parameters in aircraft systems, resulting in enhanced stability and overall performance [23].

To our best knowledge, the use of metaheuristic algorithms (MHs) for tuning NDI controller parameters has been minimally explored, particularly in comparative studies focused on altitude control in aerospace systems. This gap provides an opportunity to evaluate various optimization algorithms based on stability, responsiveness, and robustness in nonlinear flight dynamics. Previous studies have shown that the effectiveness of each metaheuristic algorithm for controller tuning can vary depending on system complexity and environmental factors [24].

Therefore, this study conducts a comparative analysis of several metaheuristic algorithms, PSO, DE, JADE, SHADE, and L-SHADE, for tuning NDI controllers in altitude control systems. The research investigates how these optimization techniques affect control performance in terms of stability, robustness, and responsiveness under diverse operational conditions, including changes in airspeed and air density.

2. Formulation of a nonlinear dynamic inversion controller for altitude control design problem

2.1 Aircraft Flight Dynamics used in this work

In this study, a portion of the nonlinear longitudinal flight dynamics model [25] is utilized for the discussion. Forces, velocity, and angles are referenced from the origin at the vehicle's center of mass, as shown in Fig. 1. The system is described in a nonlinear state-variable form as shown in Eq. (1), with the state vector $\boldsymbol{x} = \left\{v_T, \alpha, \theta, q, \delta_e, H, \dot{H}\right\}^T$ and the input $\boldsymbol{u} = \{\delta_e, \delta_t\}^T$.

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\boldsymbol{u} \qquad (1)$$

or

$$\begin{bmatrix}\dot{v}_T\\ \dot{\alpha}\\ \dot{\theta}\\ \dot{q}\\ \dot{\delta}_e\\ \dot{\delta}_t\\ \dot{H}\\ \ddot{H}\end{bmatrix} = \begin{bmatrix} \frac{F_T(v_T,\delta_t)\cos\alpha-\overline{q}(v_T)SC_D(\alpha)-W\sin\gamma(\alpha,\theta)}{m}\\ \frac{-F_T(v_T,\delta_t)\sin\alpha-\overline{q}(v_T)SC_L(\alpha)+W\cos\gamma(\alpha,\theta)}{mv_T+q}\\ q\\ \frac{M(v_T,\alpha,q,\delta_e)}{I_{yy}}\\ -20.2\delta_e\\ -20.2\delta_t\\ \dot{H}\\ \frac{F_T(v_T,\delta_t)\sin\theta-\overline{q}(v_T)SC_D(\alpha)\sin\gamma(\alpha,\theta)-W+\overline{q}(v_T)SC_L(\alpha)\cos\gamma(\alpha,\theta)}{m}\end{bmatrix} + \begin{bmatrix}0 & 0\\ 0 & 0\\ 0 & 0\\ 0 & 0\\ 20.2 & 0\\ 0 & 20.2\\ 0 & 0\\ 0 & 0\end{bmatrix}\begin{bmatrix}\delta_e\\ \delta_t\end{bmatrix} \tag{2}$$

The equations for calculating other variables are as follows:

$$F_T(v_T,\delta_t) = (338.02 + 1.5651v_T(t) - 0.00884v_T(t)^2)\delta_t(t) \tag{3}$$

$$\overline{q}(v_T) = \frac{\rho v_T(t)^2}{2} \tag{4}$$

$$C_L(\alpha) = C_{L_0} + C_{L_\alpha}\alpha(t) \tag{5}$$

$$C_D(\alpha) = C_{D_0} + \frac{1}{\pi eAR}C_L(\alpha)^2 \tag{6}$$

$$\gamma(\alpha,\theta) = \theta(t) - \alpha(t) \tag{7}$$

$$C_M(\alpha,\delta_e) = C_{M_0} + C_{M_\alpha} + C_{M_{\delta_e}}\delta_e(t) \tag{8}$$

$$M(v_T,\alpha,q,\delta_e) = \overline{q}(v_T)S\overline{c}\left(C_M(\alpha,\delta_e) + \frac{\overline{c}}{2v_T(t)}C_{M_q}q(t)\right) \tag{9}$$

The parameters $W,\ m,\ I_{yy},\ S,\ \overline{c},\ e, and\ AR$ represent the aircraft's weight, mass, moment of inertia about the lateral axis, wing reference area, mean aerodynamic chord, Oswald efficiency factor, and aspect ratio, respectively. The terms $F_T,\ \overline{q},\ C_L,\ C_D,\ \gamma,\ C_M,$ and M correspond to nonlinear thrust force, dynamic pressure, lift coefficient, drag coefficient, angle of attack, pitching moment coefficient, and pitching moment, respectively.

Additionally, aerodynamic parameters are defined as:

C_{L_0}: Lift coefficient at zero angle of attack.
C_{D_0} and C_{M_0} : Drag coefficient and pitching moment coefficient at zero lift.
C_{L_α} and C_{M_α} : Derivatives of the lift coefficient and pitching moment coefficient with respect to the angle of attack.
$C_{M_{\delta_e}}$ and C_{M_q}: Derivatives of the pitching moment coefficient with respect to elevator deflection and pitch rate, respectively.

The constant values used in this study were derived from small airplane models, specifically the Cessna 172, as presented by Brian L. Stevens and his team in the book Aircraft Control and Simulation [21], as follows:

$W = 2300$(lb), $m = 71.4286$(slugs), $I_{yy} = 2094$(slug-ft^2), $S = 175$ (ft^2), $\overline{c} = 4.89$(ft), and $\frac{1}{\pi eAR} = 0.053$

The aerodynamic derivative can be expressed as follows:

$C_{m_q} = -12.0$(per rad/s), $C_{L_0} = 0.25$, $C_{L_\alpha} = 4.58$(per rad), $C_{D_0} = 0.038$,
$C_{M_0} = 0.015$, $C_{M_\alpha} = -0.75$(per rad), and $C_{M_{\delta_e}} = -0.9$

The state evolution and output $\boldsymbol{y}$ are defined by arbitrary functions of the current states in Eq. (10).

$$\boldsymbol{y} \;=\; \boldsymbol{h}(\boldsymbol{x}) \tag{10}$$

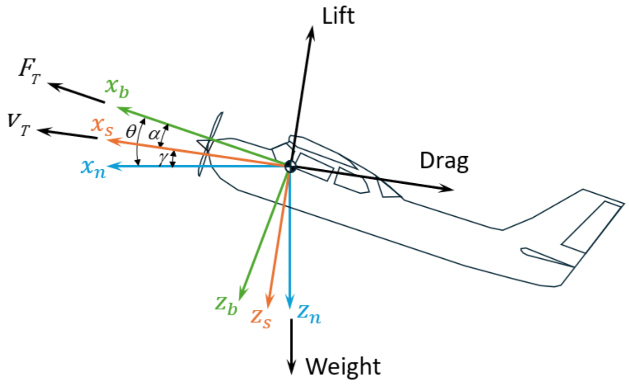

Fig. 1. The notation for forces, velocity, and angles: The coordinates' origin is at the centre of the vehicle's mass.

2.2 Nonlinear dynamic inversion controller architecture

The nonlinear dynamic inversion controller is given by

$$\boldsymbol{u} \;=\; G^{-1}(\boldsymbol{x})[-F(\boldsymbol{x}) \;+\dot{r} + \;K_y(r-y)] \tag{11}$$

In this work, we defined input $\boldsymbol{u} = \{\delta_e\}$ and $\dot{r} = 0$. This controller is depicted in Fig. 2. It requires state feedback as $\boldsymbol{x} = \{v_T, \alpha, \theta, q, \delta_e\}^T$. The output is selected as

$$y = n_{\mathrm{zp}} + K_q q \equiv h(x) \tag{12}$$

The K_q parameter is constant. It was argued that the n_{zp} cue would dominate at high velocities, and at slower approaches, velocities q would dominate. The normal acceleration at the pilot's station is

$$n_{\mathrm{zp}} = n_z + \; 15M/gI_{yy} \tag{13}$$

With M the pitching moment. The normal acceleration is given by

$$n_z \;=\; \bar{q}S(C_L \,cos\,\alpha + \;C_D \,sin\,\alpha)/mg - cos\,\theta \tag{14}$$

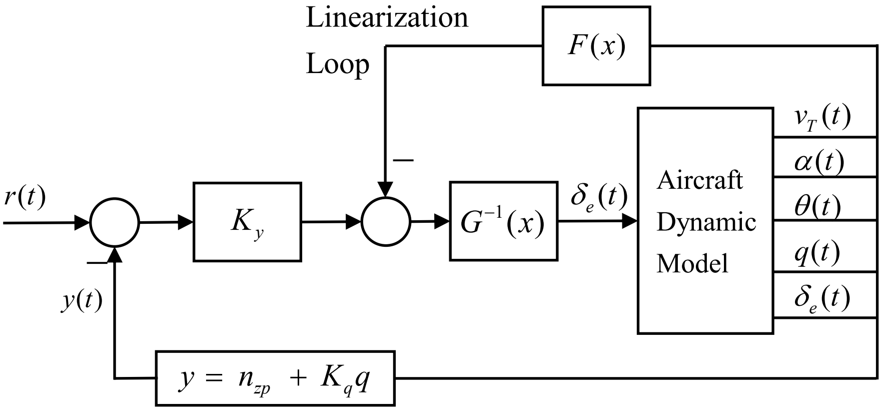

Fig. 2. Nonlinear dynamic inversion controller.

We aim to apply (11) to compute the dynamic inversion controller. To do this, we must determine

$$F(\boldsymbol{x}) = \frac{\partial h}{\partial x}\boldsymbol{f}(\boldsymbol{x}), G(\boldsymbol{x}) = \frac{\partial h}{\partial x}\boldsymbol{g}(\boldsymbol{x}) \tag{15}$$

The functions $\boldsymbol{f}(\boldsymbol{x}), \boldsymbol{g}(\boldsymbol{x})$ are defined as:

$$\boldsymbol{f}(\boldsymbol{x}) = \begin{bmatrix} \frac{F_T(v_T,\delta_t)\cos\alpha - \overline{q}(v_T)SC_D(\alpha) - W\sin\gamma(\alpha,\theta)}{m} \\ \frac{-F_T(v_T,\delta_t)\sin\alpha - \overline{q}(v_T)SC_L(\alpha) + W\cos\gamma(\alpha,\theta)}{mv_T+q} \\ q \\ \frac{M(v_T,\alpha,q,\delta_e)}{I_{yy}} \\ -20.2\delta_e \end{bmatrix}, \boldsymbol{g}(\boldsymbol{x}) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 20.2 \end{bmatrix} \tag{16}$$

Finding $\partial h/\partial \boldsymbol{x}$ is tedious and the results are as follows.

$$\frac{\partial h}{\partial x} \equiv \frac{\partial y}{\partial x} = \frac{\partial n_z}{\partial x} + \frac{15}{gI_{yy}}\frac{\partial M}{\partial x} + \boldsymbol{k} \tag{17}$$

or

$$\frac{\partial h}{\partial x} = \begin{bmatrix} \frac{\partial n_z}{\partial v_T} \\ \frac{\partial n_z}{\partial \alpha} \\ \frac{\partial n_z}{\partial \theta} \\ \frac{\partial n_z}{\partial q} \\ \frac{\partial n_z}{\partial \delta_e} \end{bmatrix}^T + \frac{15}{gI_{yy}} \begin{bmatrix} \frac{\partial M}{\partial v_T} \\ \frac{\partial M}{\partial \alpha} \\ \frac{\partial M}{\partial \theta} \\ \frac{\partial M}{\partial q} \\ \frac{\partial M}{\partial \delta_e} \end{bmatrix}^T + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ K_q \end{bmatrix}^T \tag{18}$$

First, $\partial n_z/\partial \boldsymbol{x}$ is given as

$$\begin{aligned} \frac{\partial n_z}{\partial v_T} &= \frac{\rho v_T S(C_L\cos\alpha + C_D\sin\alpha)}{mg} \\ \frac{\partial n_z}{\partial \alpha} &= \frac{\overline{q}S[(C_D + 4.58)\cos\alpha - 0.515C_L\sin\alpha]}{mg} \\ \frac{\partial n_z}{\partial \theta} &= 0, \frac{\partial n_z}{\partial q} = 0, \frac{\partial n_z}{\partial \delta_e} = 0 \end{aligned} \tag{19}$$

Next, $\partial M/\partial \boldsymbol{x}$ is given by

$$\begin{aligned} \frac{\partial M}{\partial v_T} &= \rho v_T S\bar{c}C_M + \frac{S\bar{c}^2}{4}\rho q C_{mq} \\ \frac{\partial M}{\partial \alpha} &= -0.75\bar{q}\bar{c}S \\ \frac{\partial M}{\partial \theta} &= 0 \\ \frac{\partial M}{\partial q} &= \frac{S\bar{c}^2\bar{q}}{2v_T}C_{mq} \\ \frac{\partial M}{\partial \delta_e} &= -0.9\bar{q}\bar{c}S \end{aligned} \tag{20}$$

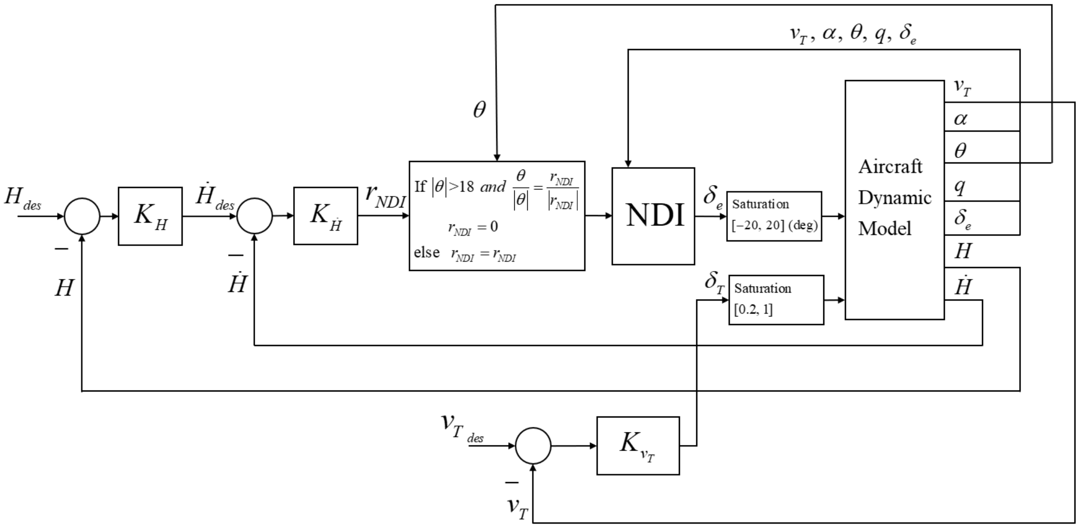

Fig. 3. Overview of the altitude controller architecture.

The altitude controller for this study utilizes the control architecture depicted in Fig. 3. Form Fig. 2. and Fig. 3. showing control gains are $K_{v_T}, K_H, K_{\dot{H}}, K_y, K_q$.

2.3 Controller design optimization problem

The optimization problem is formulated to identify all feasible controllers, as shown in Fig. 3., for flight control conditions involving an altitude climb to a reference of $H = +10(\text{ft})$ from the initial state over a time interval of 0 to 10 seconds, followed by a descent to a reference of $H = -10(\text{ft})$ from 10 to 20 seconds. This process is evaluated across varying speed and air density conditions. The design seeks to find $\boldsymbol{x} = \{K_{v_T}, K_H, K_{\dot{H}}, K_y, K_q\}^{\boldsymbol{T}}$ subject to bound constraints $\{0 \quad 0 \quad 0 \quad 0 \quad 0\}^T \leq \boldsymbol{x} \leq \{100 \quad 10 \quad 10 \quad 100 \quad 10\}^{\boldsymbol{T}}$.

The objective is to minimize specific functions for optimal altitude reference tracking across four initial trim conditions, as illustrated in Fig. 4. Two objective functions, detailed in Eqs. (21) – (22), are proposed to consolidate many objectives into a single formulation. These objectives are compared to identify the most efficient controller obtained through the optimization process. The proposed optimization problem is formulated as follows:

Objective function Case I

$$min\colon \boldsymbol{f}(\mathbf{x}) = \max\left(f_1(\mathbf{x}), f_2(\mathbf{x}), f_3(\mathbf{x}), f_4(\mathbf{x}) \right) \tag{21}$$

Objective function Case II

$$min\colon \boldsymbol{f}(\mathbf{x}) = 0.25 f_1(\mathbf{x}) + 0.25 f_2(\mathbf{x}) + 0.25 f_3(\mathbf{x}) + 0.25 f_4(\mathbf{x}) \tag{22}$$

Subjected to:

$g_1(x) = |\theta(t)| \leq 20(deg)$,
$g_2(x) = max(H(t)) \leq 10(\text{ft})$,
$g_3(x) = min(H(t)) \geq 0(\text{ft})$

where

$$f_1(\mathbf{x}) = \int_{t=0}^{t=10} t e^2(t) dt + \int_{t=10}^{t=20} (t-10) e^2(t) dt$$

: Initial trim condition $v_T = 90$ (ft/s) and ρ at sea level

$$f_2(\mathbf{x}) = \int_{t=0}^{t=10} te^2(t)dt + \int_{t=10}^{t=20} (t-10)e^2(t)dt$$

: Initial trim condition $v_T = 120$ (ft/s) and ρ at 15,000 (ft) above sea level

$$f_3(\mathbf{x}) = \int_{t=0}^{t=10} te^2(t)dt + \int_{t=10}^{t=20} (t-10)e^2(t)dt$$

: Initial trim condition $v_T = 180$ (ft/s) and ρ at sea level

$$f_4(\mathbf{x}) = \int_{t=0}^{t=10} te^2(t)dt + \int_{t=10}^{t=20} (t-10)e^2(t)dt$$

: Initial trim condition $v_T = 200$ (ft/s) and ρ at 15,000 (ft) above sea level

The functions $f_1(\mathbf{x})$ - $f_4(\mathbf{x})$ represent the integral time-squared error of the altitude response compared to the altitude reference across four initial flight trim conditions, as shown in Fig. 4. This formulation aims for optimal altitude reference tracking over a time domain of 0 to 20 seconds, with a time step of 0.01 seconds. The constraint $g_1(x)$ ensures that the pitch angle does not exceed the aircraft's operational limits, while $g_2(x)$ and $g_3(x)$ accuracy in reference tracking.

The penalty function used to handle these constraints in this work is expressed as follows:

$$f_p(x) = \begin{cases} f(x) & If \ \max(\boldsymbol{g}(\boldsymbol{x})) < 0 \\ f(x) + 1e^3 \sum_{i=1}^{3} \max(0, g_i(\boldsymbol{x})), & \text{otherwise} \end{cases}$$

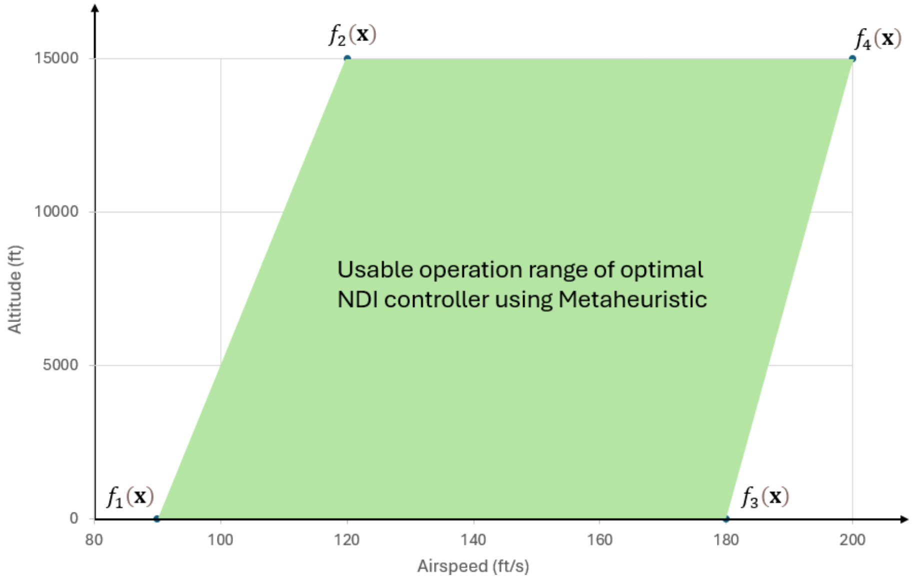

Fig. 4. Usable operation range of optimal NDI controller using Metaheuristics.

3. Numerical Experiment

Two numerical experiments are proposed in this study. Firstly, consolidate multiple objectives problem into a single formulation problem as detailed in objective function Case I and Case II are compared. The Success-History Based Adaptive Differential Evolution with Linear Population Size Reduction (L-SHADE) [22] is employed for process optimization of those two objective problem and performance of the flight controller. Then, an object problem is used to evaluate the performance of MHs. For the MHs performance investigation five state of the art MHs are used. The five MHs used in this study and their optimization parameter settings are detailed as follows:

1. Particle swarm optimisation (PSO) [18]:
   - Starting inertia weight (W_{st}) and ending inertia weight (W_{en}) are set to 0.5 and 0.01, respectively while the balance between exploration and exploitation is achieved by controlling the influence of a particle's previous velocity on its current movement.
   - A cognitive learning factor (C_1) is set to 2.8. The parameter determines the influence of a particle's personal best on its movement, reflecting its tendency to learn from its own experience.
   - A social learning factor (C_2) is set to 1.3. The parameter governs the influence of the global best on a particle's movement, representing its tendency to follow the collective knowledge of the swarm.

2. Differential evolution (DE) [19]:
   - The DE/best/2/bin strategy was used.
   - A scaling factor (F) is set to 0.5, it controls the magnitude of the differential mutation operator.
   - A crossover rate (CR) and the probability of choosing elements of mutant vectors (PC) are set to 0.7 and 0.8, respectively. These parameters determine the likelihood of each component of the trial vector being inherited from the mutant vector.

3. Self Adaptive Differential Evolution (JADE) [20]:
   - The optimisation parameters are self-adaptive.

4. Success-History Based Adaptive Differential Evolution (SHADE) [21]:
   - The optimisation parameters are self-adaptive.

5. SHADE with Linear Population Size Reduction (L-SHADE) [22]:
   - The optimisation parameters are self-adaptive.

Each algorithm is used to solve the proposed design problem for 30 optimization runs while the maximum number of iterations and the population size are both set at 50. Each run consists of a total of 2,500 evaluations.

4. Results and Discussion

4.1 Evaluation of consolidating many objectives into a single formulation for Case I and Case II

After 30 optimization runs using L-SHADE for solving Case I and Case II, the comparative efficiency results for each case are presented in Figs. 5-6. and Table 1. Fig. 5. shows the search history of minimizing penalty functions $f_{p1}(\mathbf{x}), f_{p2}(\mathbf{x}), f_{p3}(\mathbf{x})$, and $f_{p4}(\mathbf{x})$ as a function of evaluations for both Case I and Case II. The figure shows that Case II achieved better initial minimization than Case I approximately 650 function evaluations. However, after 1,000 function evaluations, both cases converge to a similar minimum value. These results indicate that, for NDI-based flight controller optimal tuning, consolidating many objectives into a single formulation for Case I and Case II does not affect the final solution after 2,500 function evaluations. The final solutions obtained based on Case I and Case II, presented in Table 1, are insignificantly different. Case I produces slightly better results for $f_{p1}(\mathbf{x})$ and $f_{p2}(\mathbf{x})$, while Case II yields slightly better values for $f_{p3}(\mathbf{x})$ and $f_{p4}(\mathbf{x})$. The steady-state errors in both cases are very small, with Case I demonstrating slightly better performance.

Fig. 6. illustrates the height reference tracking for varying initial conditions of the controller designed with Case I compared to Case II. The figure clearly shows that the two lines overlap perfectly. These results indicate that Case I and Case II produce equivalent design outcomes. Fig. 6. also demonstrates that the NDI controller, designed using L-SHADE, effectively performs height reference tracking across varying initial flight trim conditions, as illustrated in Fig. 4. Fig. 7. presents the aircraft's dynamic simulation states from 0 to 20 seconds, confirming that the results comply with the specified constraint $|\theta(t)| \leq 20(deg)$ and saturation limits $|\delta_e| \leq 20(deg)$, and $0.2 \leq \delta_T \leq 1$.

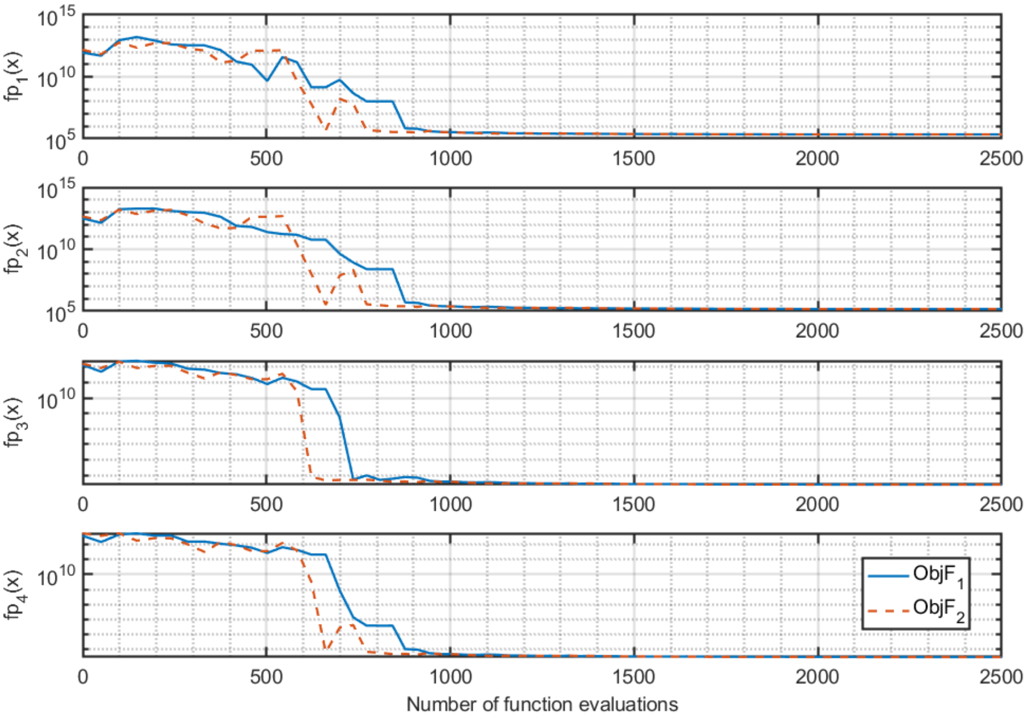

Fig. 5. Mean search history of 30 optimization runs for minimizing the penalty function of objective function: Case I vs. Case II (Logarithmic Scale)

Table 1: The design variables, objective penalty functions, constraints, and steady-state error of the best run were minimized using objective function Case I and Case II (Bold indicates the best value).

Design Variables	$\mathbf{ObjF_1}$	$\mathbf{ObjF_2}$
x_1	1.0000E+02	1.0000E+02
x_2	5.2953E-01	5.2853E-01
x_3	8.5115E-02	8.7899E-02
x_4	1.0000E+02	9.9996E+01
x_5	1.7714E+00	1.8345E+00
Penalty functions		
$f_{p1}(x)$	**2.1916E+05**	2.1925E+05
$f_{p2}(x)$	**1.4801E+05**	1.4803E+05
$f_{p3}(x)$	2.5090E+04	**2.4987E+04**
$f_{p4}(x)$	3.1920E+04	**3.1737E+04**
Constraints		
$g_1(x)$	-7.2072E-02	-7.1912E-02
$g_2(x)$	-8.0751E-04	-8.8208E-04
$g_3(x)$	8.4399E-06	8.4399E-06
Steady-state error		
e_{ss} at 10s on $v_T = 90$ and ρ at 0 ft	0.0113 ft	0.0127 ft
e_{ss} at 20s on $v_T = 90$ and ρ at 0 ft	0.0132 ft	0.0149 ft
e_{ss} at 10s on $v_T = 120$ and ρ at 15,000 ft	0.0087 ft	0.0102 ft
e_{ss} at 20s on $v_T = 120$ and ρ at 15,000 ft	0.0081 ft	0.0094 ft
e_{ss} at 10s on $v_T = 180$ and ρ at 0 ft	0.0061 ft	0.0070 ft
e_{ss} at 20s on $v_T = 180$ and ρ at 0 ft	0.0061 ft	0.0071 ft
e_{ss} at 10s on $v_T = 200$ and ρ at 15,000 ft	0.0074 ft	0.0084 ft
e_{ss} at 20s on $v_T = 200$ and ρ at 15,000 ft	0.0074 ft	0.0085 ft

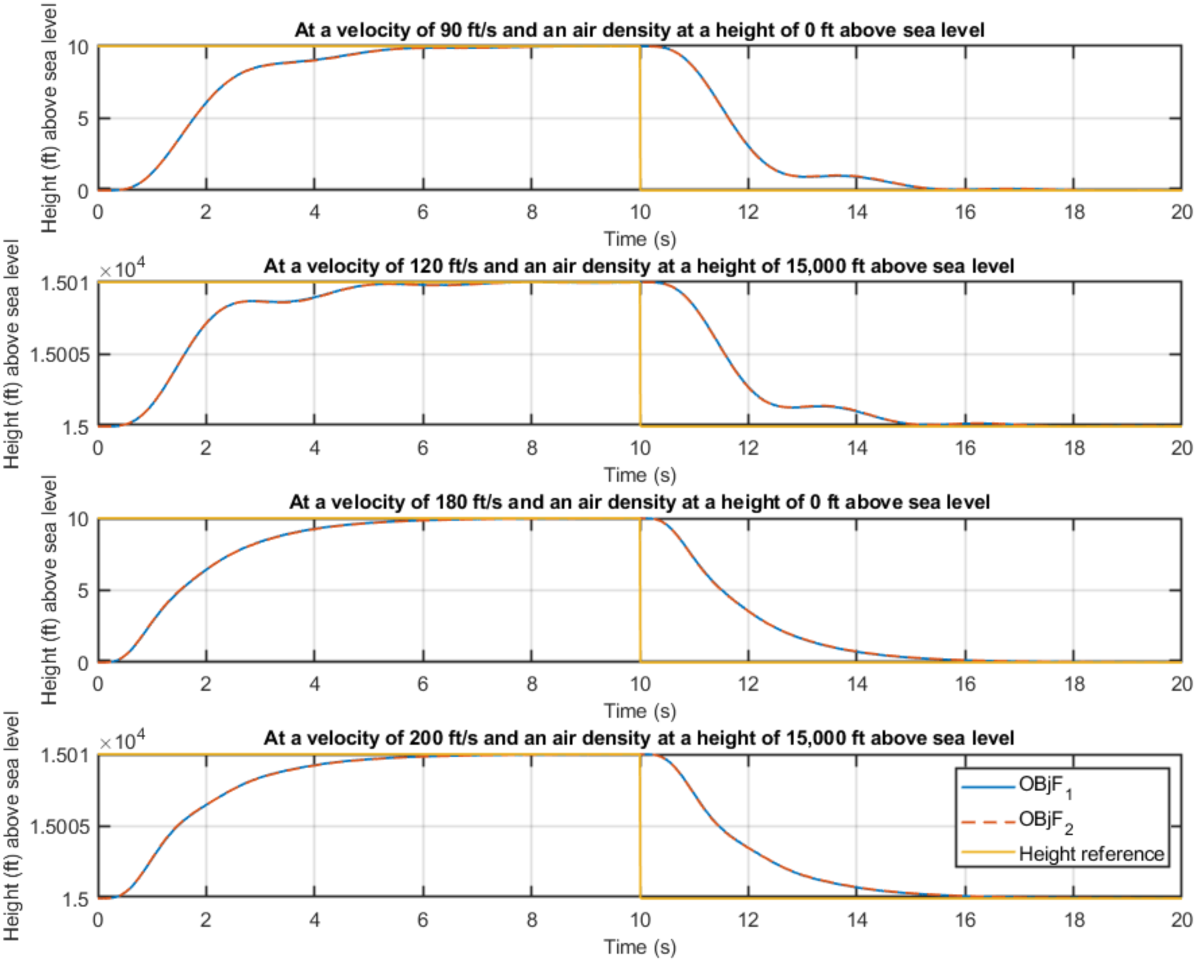

Fig. 6. Height reference tracking for various initial conditions of best solution at 2,500 function of evaluations for controller designed using objective function Case I compared with objective function Case II.

4.2 Performance investigation of MHs

To evaluate performance of several MHs on solving NDI-base flight control optimization design, many objective problems which consolidated to single objective in Case II is applied. After perform 30 optimization runs on solving the problem, the results were compared based on the objective function values presented in Table 2. The mean value of the objective function was used to evaluate the convergence rate and consistency of the algorithms, while the number of successful runs was used to measure their search consistency. In the event that two algorithms had the same number of successful runs, their consistency performance was evaluated using the standard deviation (STD) of the objective function values. Only the algorithm that found the feasible solution at least twice was allowed to have mean and STD values. From Table 2, all metaheuristics (MHs) were able to obtain 30 feasible solutions. L-SHADE performed best across all metrics, including Best, Worst, Mean, Standard Deviation (STD), and Friedman ranking, while SHADE and JADE ranked second and third, respectively.

Fig. 8. illustrates the average search history of the metaheuristics (MHs) from Table 2. At the beginning of the optimization process, Differential Evolution (DE) performed the worst, while SHADE became trapped in a local optimum. In contrast, JADE, SHADE, and L-SHADE continued to improve. JADE was the fastest, followed by L-SHADE and SHADE in second and third places, respectively. By the end of the optimization process, L-SHADE emerged as the best performer, with SHADE and JADE in second and third places, respectively.

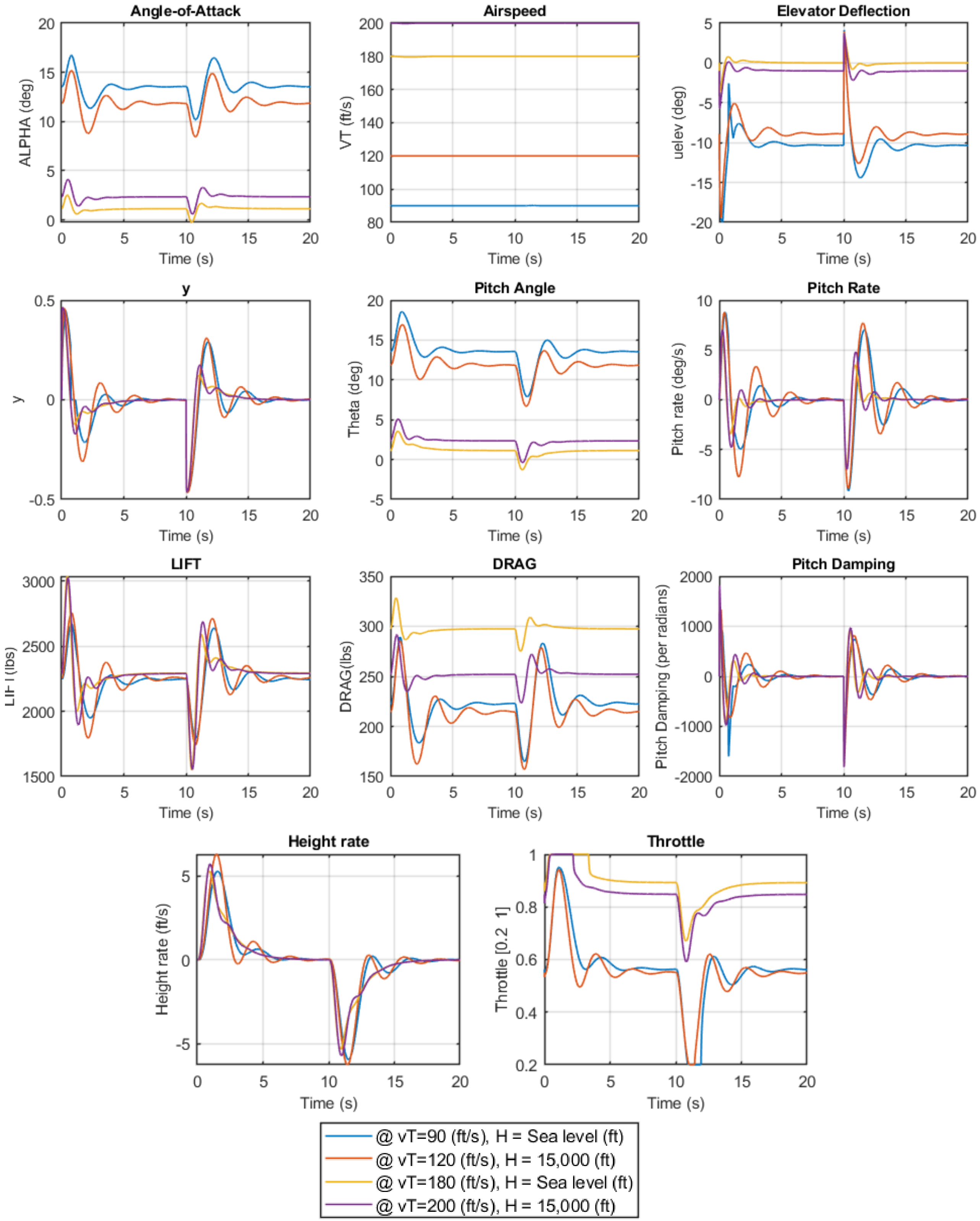

Fig. 7. The aircraft's dynamic simulation states from 0 to 20 seconds of the best solution of L-SHADE minimizes minimizing objective function Case II.

Table 2: The statistical comparison includes the best, worst, mean, standard deviation (STD), and Friedman ranking (Fr) scores, based on the 30 runs minimizing objective function Case II (Bold indicates the best value).

Algorithms	Worst	Best	Mean	STD	No.sr*	Fr*
PSO	2.452E+08	1.136E+05	1.308E+08	1.243E+08	30	4.03
DE	4.327E+12	2.650E+07	1.457E+11	7.897E+11	30	4.97
JADE	1.189E+05	1.072E+05	1.099E+05	2.338E+03	30	2.63
SHADE	1.151E+05	1.066E+05	1.093E+05	1.803E+03	30	2.37
LSHADE	**1.061E+05**	**1.060E+05**	**1.060E+05**	**1.053E+01**	30	**1.00**

No.sr* = No. of successful runs , Fr* = The mean Friedman ranking scores

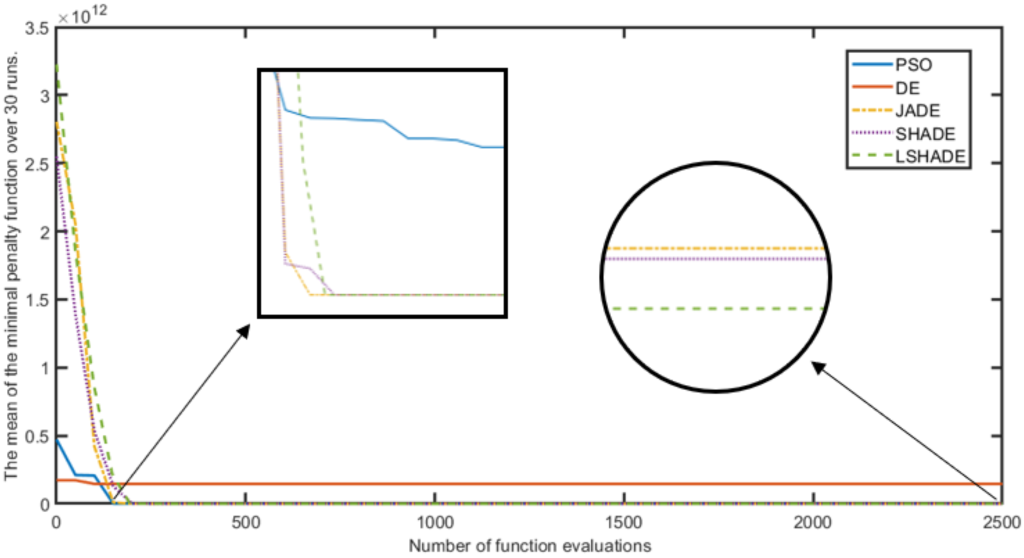

Fig. 8. The search history averaged over 30 runs.

5. Conclusion

This study explores the successful application of several metaheuristic algorithms (MHs) in designing a nonlinear dynamic inversion (NDI) controller for altitude control. The optimization challenge centers on selecting control gains that maximize performance while maintaining stability and adhering to control handling constraints. To simplify the many objective optimization problem, it is reformulated into two single-objective cases—Case I and Case II. The results show that both cases can be used effectively to design the NDI controller with comparable efficiency, although Case II achieves a slightly better optimum result. Additionally, a range of MHs is employed to tackle this problem, with their performance rigorously assessed. Among the algorithms tested, L-SHADE is identified as the most efficient, demonstrating superior convergence and consistency. A key contribution of this work is that the application of effective metaheuristics enables even novice designers to engage in NDI controller synthesis successfully. This research lays the groundwork for further exploration of straightforward and efficient approaches to NDI controller design using many-objective MHs, with future studies planned to extend these methods to the design of Incremental Nonlinear Dynamic Inversion (INDI) controllers.

Acknowledgement

The authors are grateful for financial support by the Research Fund of the Faculty of Engineering, Khon Kaen University under the Research Scholarship for Ph.D. Students project under Contract No. Ph.D.-001/2565 and the National Research Council of Thailand, Grant No. N41A670974.

Declaration of generative AI in scientific writing: The authors used ChatGPT for grammar corrections and readability improvement of the manuscript. Afterwards, the authors reviewed and edited the content and take full responsibility for the content of the publication.

References

[1] Suiçmez EC, Kutay AT. Full envelope nonlinear flight controller design for a novel electric VTOL (eVTOL) air taxi. The Aeronautical Journal. 2024;128(1323):966–93.

[2] The influence of the operational altitude variation on flight controller design on combat aircraft configuration. AIAA Aviation Forum and ASCEND Co-located Conference Proceedings [Internet]. [cited 2024 Nov 2]. Available from: https://arc.aiaa.org/doi/10.2514/6.2024-4584

[3] Elkhatem AS, Engin SN. Enhancing performance and stability of gain-scheduling control system using evolutionary algorithms: A case study on transport aircraft. Expert Systems with Applications. 2023;213:118859.

[4] Ruenruedeepan N, Champasak P, Panagant N, Pholdee N, Bureerat S. Fixed-structure heading-autopilot controller design using meta-heuristics. Engineering & Applied Science Research. 2024;51(1):34–41.

[5] Kanokmedhakul Y, Pholdee N, Bureerat S, Panagant N. LQR aircraft pitch controller design for handling disturbance using differential evolution. Journal of Research and Applications in Mechanical Engineering. 2019;7(2):145–53.

[6] Kanokmedhakul Y, Bureerat S, Panagant N, Radpukdee T, Pholdee N, Yildiz AR. Metaheuristic-assisted complex H-infinity flight control tuning for the Hawkeye unmanned aerial vehicle: A comparative study. Expert Systems with Applications. 2024;248:123428.

[7] Adaptive nonlinear flight control of STOL-aircraft based on incremental nonlinear dynamic inversion. AIAA Aviation Forum [Internet]. [cited 2024 Nov 2]. Available from: https://arc.aiaa.org/doi/10.2514/6.2018-3257

[8] Zou Q, Mu X, Li H, Huang R, Hu H. Robust active suppression for body-freedom flutter of a flying-wing unmanned aerial vehicle. Journal of the Franklin Institute. 2021;358(5):2642–60.

[9] Kim C, Ji C, Koh G, Choi N. Review on flight control law technologies of fighter jets for flying qualities. International Journal of Aeronautical and Space Sciences. 2023;24(1):209–36.

[10] Abdallah AM, Newman BA, Omran AM. Measuring aircraft nonlinearity across aerodynamic attitude flight envelope. In: AIAA Atmospheric Flight Mechanics (AFM) Conference [Internet]. American Institute of Aeronautics and Astronautics; [cited 2024 Dec 22]. Available from: https://arc.aiaa.org/doi/abs/10.2514/6.2013-4985

[11] Yang L, Shami A. On hyperparameter optimization of machine learning algorithms: Theory and practice. Neurocomputing. 2020;415:295–316.

[12] Pătrăușanu A, Florea A, Neghină M, Dicoiu A, Chiș R. A systematic review of multi-objective evolutionary algorithms optimization frameworks. Processes. 2024;12(5):869.

[13] Champasak P, Panagant N, Bureerat S, Pholdee N. Investigation on the performance of meta-heuristics for solving single objective conceptual design of a conventional fixed wing unmanned aerial vehicle. Journal of Research and Applications in Mechanical Engineering. 2022;10(1):22–10.

[14] Sleesongsom S, Bureerat S. Optimal synthesis of four-bar linkage path generation through evolutionary computation. Journal of Research and Applications in Mechanical Engineering. 2015;3(2):46–53.

[15] Chattunyakit S, Kobayashi Y, Emaru T. PSO-based leg-loss identification method for legged robots. Journal of Research and Applications in Mechanical Engineering. 2016;4(1):46–57.

[16] Leeprechanon N, Phonrattanasak P. Optimal placement of wind farm on the power system topology. Journal of Research and Applications in Mechanical Engineering. 2011;1(1):13–7.

[17] Mostafa E, Aly HA, Elliethy A. Accelerating genetic optimization of nonlinear model predictive control by learning optimal search space size. arXiv [Internet]. 2023 [cited 2024 Nov 26]. Available from: http://arxiv.org/abs/2305.08094

[18] Schutte JF, Groenwold AA. Sizing design of truss structures using particle swarms. Structural and Multidisciplinary Optimization. 2003;25(4):261–9.

[19] Gao Z, Xiao T, Fan W. Hybrid differential evolution and Nelder–Mead algorithm with re-optimization. Soft Computing. 2011;15(3):581–94.

[20] Zhang J, Sanderson AC. JADE: Self-adaptive differential evolution with fast and reliable convergence performance. IEEE Congress on Evolutionary Computation. 2007;2251–8.

[21] Tanabe R, Fukunaga A. Success-history based parameter adaptation for differential evolution. 2013;71–8.

[22] Tanabe R, Fukunaga A. Improving the search performance of SHADE using linear population size reduction. 2014.

[23] Joseph SB, Dada EG, Abidemi A, Oyewola DO, Khammas BM. Metaheuristic algorithms for PID controller parameters tuning: Review, approaches and open problems. Heliyon [Internet]. 2022 [cited 2024 Nov 2];8(5). Available from: https://www.cell.com/heliyon/abstract/S2405-8440(22)00687-9

[24] Rodríguez-Molina A, Mezura-Montes E, Villarreal-Cervantes MG, Aldape-Pérez M. Multi-objective meta-heuristic optimization in intelligent control: A survey on the controller tuning problem. Applied Soft Computing. 2020;93:106342.

[25] Stevens BL, Lewis FL, Johnson EN. Aircraft Control and Simulation, 3rd ed. Dynamics, Controls Design and Autonomous Systems.