

## Research Article

# Aircraft Control Parameter Estimation Using Self-Adaptive Teaching-Learning-Based Optimization with an Acceptance Probability

Yodsadej Kanokmedhakul <sup>1</sup>, Natee Panagant <sup>1</sup>, Sujin Bureerat <sup>1</sup>,  
Nantiwat Pholdee <sup>1</sup> and Ali R. Yildiz <sup>2</sup>

<sup>1</sup>Sustainable Infrastructure Research and Development Center, Department of Mechanical Engineering, Faculty of Engineering, Khon Kaen University, Khon Kaen, Thailand

<sup>2</sup>Department of Mechanical Engineering, Bursa Uludag University, Bursa, Turkey

Correspondence should be addressed to Nantiwat Pholdee; [nantiwat@kku.ac.th](mailto:nantiwat@kku.ac.th)

Received 1 September 2021; Revised 15 November 2021; Accepted 1 December 2021; Published 20 December 2021

Academic Editor: Miaolei Zhou

Copyright © 2021 Yodsadej Kanokmedhakul et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This work presents a metaheuristic (MH) termed, self-adaptive teaching-learning-based optimization, with an acceptance probability for aircraft parameter estimation. An inverse optimization problem is presented for aircraft longitudinal parameter estimation. The problem is posed to find longitudinal aerodynamic parameters by minimising errors between real flight data and those calculated from the dynamic equations. The HANSA-3 aircraft is used for numerical validation. Several established MHs along with the proposed algorithm are used to solve the proposed optimization problem, while their search performance is investigated compared to a conventional output error method (OEM). The results show that the proposed algorithm is the best performer in terms of search convergence and consistency. This work is said to be the baseline for purely applying MHs for aircraft parameter estimation.

## 1. Introduction

Flight control is one of the most important parts in developing a new aircraft or improving an existing one. It is even more crucial when the applications of unmanned aerial vehicles (UAV) have been introduced. Traditionally, aircraft motion is modelled based on the equations of motion or Newton's second law, leading to a system of nonlinear equations. Often, such a system is linearized, resulting in a linear state-space control model. In order to reach the possibly highest performance for flight control, identification of aerodynamic parameters and the aircraft dynamic model needs to be accurate. Although aircraft aerodynamic parameters including aerodynamic, stability, and control derivatives can be evaluated from an empirical model [1], numerical models (a vortex lattice method) [2],

computational fluid dynamics (CFD) [3]), and wind tunnel test, errors between the test data and the real aircraft data are still inevitable. The designed aircraft and the manufactured aircraft are always different, while it is even worse when the aircraft is subject to structural flexibility in real flight. This implies that accurate system identification of the real or manufactured aircraft is always required. In this regard, parameter estimation techniques are necessary and have been one of the most popular research topics in the field of flight dynamics and control. Conventional parameter estimation techniques that can be used to evaluate stability and control derivatives from flight information have been presented [4]. However, the traditional techniques are suitable for estimating stability derivatives in a linear trim condition for a stable and rigid aircraft. They are insufficient for a highly manoeuvrable or unstable aircraft, and also expensive

computation is required for estimating a large number of parameters for a full-order model of an aircraft. Also, some of the more efficient parameter estimation methods have been developed based on an implicated function technique in combination with optimization. These include the Equation Error Method (EEM) [5–8], the Output Error Method (OEM) [9–11], and the Filter Error Method (FEM) [12–15]. However, those methods still require a predefined/initial aircraft model. In this regard, development of the more efficient flight parameter estimation from flight test data without a predefined aircraft model is a challenging topic.

Recently, efficient flight parameter estimation techniques based on artificial intelligence (AI) and optimization tools have been proposed, e.g., artificial neural network (ANN) and fuzzy set theory [16–18]. The combination of AI and MH search has also been presented [19, 20]. However, even all those techniques cannot be used without a predefined aircraft model. ANN and fuzzy require a large amount of training data and also rely on efficient parameter settings. Using standalone MH for such a problem as an inverse optimization problem solver, if successful, could be a good tool for aircraft flight control.

MH (also known as evolutionary algorithms) can be classified as optimization methods which are global and nongradient optimizers. Due to such advantages, they can deal with any kind of design variable, objective, and constraint functions, although they may be less efficient in some cases. They can also explore a Pareto front within a single run, in cases of a multiobjective optimization problem. Therefore, they are at present the most used and popular optimizers for real engineering design optimization problems [21–24]. For optimization applied to an inverse problem, successful use of MHs is reported worldwide, such as a damage detection problem [25, 26], an inverse kinematic design of a robot [27], robot trajectory planning optimization [28], mechanism synthesis [29], and parameter identification of photovoltaic models [30, 31]. For inverse problem optimization of aircraft parameters estimation using MH, to our knowledge studies are rare. Some MHs found being used for solving such a problem are a classical genetic algorithm (GA) [32]. Since investigation on the use of MHs for aircraft control system identification has been limited, it is thus a motivation for this work. Up to the present time, since one of the very first algorithms GA was invented, there have been over a thousand MHs and their variants presented in the literature. A comparative performance study of some established and newly invented MHs on the parameter estimation of an aircraft system is one interesting subject while the more challenging task is to develop a new powerful MH or to improve the performance of an existing MH for such an inverse problem. Among existing MHs, teaching-learning-based optimization (TLBO) is an outstanding MH, which is found to be one of the most powerful algorithms for solving inverse problems through optimization [25, 29–31, 33]. However, its performance on the inverse problem of parameter estimation of an aircraft system has never been tested. Furthermore, although TLBO was found to be one of the best optimizers for

solving inverse problems through optimization, the original version is developed for general optimization problem. Therefore, enhancing the TLBO algorithm based on a novel technique for this specific optimization problem is challenging and will lead to a powerful tool for aircraft parameter estimation.

As a result, this work proposes an efficient MH algorithm called “self-adaptive teaching-learning-based optimization with acceptance probability (SaTLBO-AP)” for solving an inverse problem of aircraft parameter estimation. The proposed algorithm is based on using TLBO as the main algorithm in combination with a diversity archive. The self-adaptive scheme exploits the acceptance probability used in simulated annealing for the balance between diversification and intensification. The inverse optimization problem for longitudinal flight parameter identification of the HANSA-3 aircraft [20] is proposed. The optimization problem is then solved by the proposed algorithm along with several newly invented and well-established MHs, including Ant Lion Optimizer (ALO) [34], Whale Optimization Algorithm (WOA) [35], Salp Swarm Algorithm (SSA) [36], Moth-Flame Optimization (MFO) [37], Grey Wolf Optimization (GWO) [38], Grasshopper Optimization Algorithm (GOA) [39], Dragonfly Algorithm (DA) [40], Water Cycle Algorithm (WCA) [41], Multi-Verse Optimizer (MVO) [42], Sine Cosine Algorithm (SCA) [43], Monarch Butterfly Optimization (MBO) [44], Slime Mould Algorithm (SMA) [45], Elephant Herding Optimization (EHO) [46], Artificial Bee Colony Algorithm (ABC) [47], Self-adaptive Differential Evolution Algorithm (SaDE) [48], Improved Teaching-Learning-Based Optimization (ITLBO) [31], and the original Teaching-Learning-Based Optimization (TLBO) [49]. The results obtained are compared and discussed.

The rest of this paper includes sections on the proposed algorithm, SaTLBO-AP, formulation of the inverse optimization problem for longitudinal flight, parameter estimation, numerical experiments, results and discussion, and conclusions.

## 2. Formulation of Inverse Optimization Problem

Rigid aircraft flight dynamics are governed by the equations of motion or Newton’s second law. The rigid aircraft has 6 degrees of freedom with 3 for translations and the other 3 for rotations. The conventional north-east-down coordinates can be used as an inertial reference frame. It is also convenient to use the body axes as shown in Figure 1. The equations of motion lead to a nonlinear flight dynamic model. In order to simplify the model, a small perturbation approach is employed to linearize the model. Then, with the left right symmetry of an aircraft, the model can be separated into a longitudinal and lateral/directional motions. This leads to easy to handle aircraft dynamic and control models.

In order to examine the performance of the proposed parameter estimation approach, parameter estimation of the longitudinal flight control model is presented. The nonlinear flight control model for longitudinal motion of a conventional aircraft used in this study can be expressed as [4]

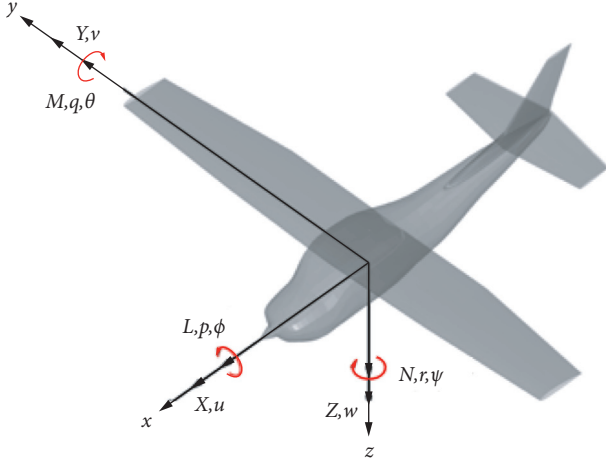


FIGURE 1: Aircraft coordinate systems.

$$C_D = C_{D_0} + C_{D_\alpha} \alpha + C_{D_{\delta e}} \delta e, \quad (1)$$

$$C_L = C_{L_0} + C_{L_\alpha} \alpha + C_{L_q} \left( \frac{\bar{q}c}{2V} \right) + C_{L_{\delta e}} \delta e, \quad (2)$$

$$C_m = C_{m_0} + C_{m_\alpha} \alpha + C_{m_q} \left( \frac{\bar{q}c}{2V} \right) + C_{m_{\delta e}} \delta e, \quad (3)$$

$$\dot{V} = - \left( \frac{\bar{q}S}{m} \right) C_D + g \sin(\alpha - \theta) + \left( \frac{F_{\text{eng}}}{m} \right) \cos(\alpha), \quad (4)$$

$$\dot{\alpha} = - \left( \frac{\bar{q}S}{mV} \right) C_L + q + \left( \frac{g}{V} \right) \cos(\alpha - \theta) - \left( \frac{F_{\text{eng}}}{mV} \right) \sin(\theta), \quad (5)$$

$$\dot{\theta} = q. \quad (6)$$

The aerodynamic parameters  $\Theta = \{C_{D_0}, C_{D_\alpha}, C_{D_{\delta e}}, C_{L_0}, C_{L_\alpha}, C_{L_q}, C_{L_{\delta e}}, C_{m_0}, C_{m_\alpha}, C_{m_q}, C_{m_{\delta e}}\}$  are assumed unknown and to be identified.

To apply an inverse optimization problem for longitudinal motion parameter estimation, the design problem is posed to find a set of aerodynamic parameters, in order to minimise errors between the longitudinal response of a real aircraft and the calculated response from the longitudinal dynamic equations. The optimization problem can be expressed as

$$\min : f(x) = \sum_{i=1}^l \sum_{t=t_0}^{t_{\text{end}}} \frac{|r_{\text{real},i}(t) - r_{\text{estimate},i}(t)|}{|r_{\text{real},i}(t)|}. \quad (7)$$

Subject to

$$L_b \leq x \leq U_b, \quad (8)$$

where  $x = \Theta$  is a vector of design variables having  $L_b$  and  $U_b$  as the lower and upper bounds. The details of the design variables are shown in Table 1. The parameters  $r_{\text{real},i}$  and  $r_{\text{estimate},i}$  are

TABLE 1: Design variables with lower and upper bounds.

	$C_{D_0}$	$C_{D_\alpha}$	$C_{D_{\delta e}}$	$C_{L_0}$	$C_{L_\alpha}$	$C_{L_q}$	$C_{L_{\delta e}}$	$C_{m_0}$	$C_{m_\alpha}$	$C_{m_q}$	$C_{m_{\delta e}}$
$L_b$	0	0	0	0	0	0	0	0	-5	-50	-5
$U_b$	5	1	5	5	50	200	5	1	0	0	0

respectively the  $i^{\text{th}}$  real and estimated longitudinal motion parameter time responses, whereas  $l$  is the number of longitudinal parameters considered. Four longitudinal parameters include  $V$ ,  $\alpha$ ,  $\theta$ , and  $q$ . The parameters  $t_0$  and  $t_{\text{end}}$  are the initial and final times of the simulation, respectively.

As there are several types of physical parameters with totally different units, e.g., aircraft velocity and angular position, each estimation error in (7) is therefore normalised before being summed up. This step can be easily achieved by dividing by the absolute value of the real time response.

In this study, the real time response from a flight test is simulated using the flight model of the HANSA-3 aircraft as shown in Table 2, while the target values of aerodynamic parameters  $\{C_{D_0}, C_{D_\alpha}, C_{D_{\delta e}}, C_{L_0}, C_{L_\alpha}, C_{L_q}, C_{L_{\delta e}}, C_{m_0}, C_{m_\alpha}, C_{m_q}, C_{m_{\delta e}}\}$  are shown in Table 3. The aircraft flight data are simulated with the elevator deflection of a 3-2-1-1 step input. The simulation is performed for a six-second time length ( $t$ ) with a time step ( $\Delta t$ ) of 0.025 second. Gaussian noise with zero mean is added to the system response with the degree of noise at 0%, 5%, and 10% with respect to the amplitude of the time response. The state time response of the longitudinal motion  $\mathbf{r} = \{V, \alpha, \theta, q\}^T$  can be numerically solved based on (9), while  $\dot{\mathbf{r}}$  can be calculated based on (1)–(6). The state time response used as real flight data is shown in Figure 2.

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \int_t^{t+\Delta t} \dot{\mathbf{r}}(t) dt. \quad (9)$$

### 3. Self-Adaptive Teaching-Learning Based Optimization with an Acceptance Probability

A TLBO is a simple but efficient MH proposed by Rao et al. in 2011 [49]. The algorithm was inspired by the behaviour of teaching and learning in a class. The main search procedure of the TLBO consists of population initialisation, reproduction, and selection, while, at the reproduction process, there are two main phases called the teaching and learning phases. Each individual in the population is first updated in the teaching phase with the relation.

$$x_{\text{teaching}}^i = x^i + \text{rand} \cdot d \{x_{\text{teacher}} - (T_F x_{\text{mean}})\}, \quad (10)$$

where  $x^i$  is the  $i^{\text{th}}$  individual in the population.  $x_{\text{teacher}}$  is the best individual.  $x_{\text{mean}}$  is mean value of other members in the population.  $\text{rand} = a$  uniform random number in the range of [0, 1].  $T_F$  = teaching factor, which can be either 1 or 2 at random.

TABLE 2: Flight condition and aircraft geometry of HANSA-3.

Variable	Value
Mean aerodynamic chord ( $\bar{c}$ )	1.21 m
Wingspan ( $b$ )	10.84 m
Reference wing area ( $S$ )	12.47 m <sup>2</sup>
Mass ( $m$ )	758 kg
True air speed ( $V$ )	52 m/s
Moment of inertia ( $I_y$ )	925 kgm <sup>2</sup>
Engine thrust ( $F_{eng}$ )	1136 N

TABLE 3: The target value of aerodynamic parameters.

Parameter	$C_{D_0}$	$C_{D_\alpha}$	$C_{D_{\delta e}}$	$C_{L_0}$	$C_{L_\alpha}$	$C_{L_q}$	$C_{L_{\delta e}}$	$C_{m_0}$	$C_{m_\alpha}$	$C_{m_q}$	$C_{m_{\delta e}}$
True value	0.036	0.061	0.152	0.23	4.886	37.259	0.376	0.091	-0.412	-8.792	-0.735

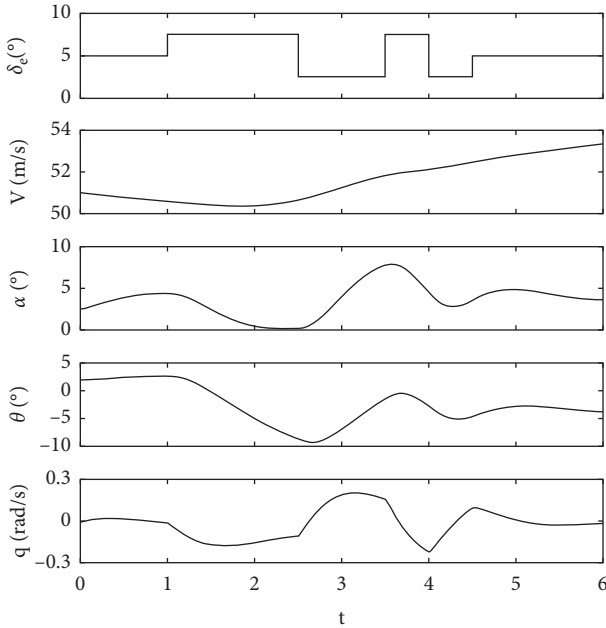


FIGURE 2: The state time response used as real flight data.

The offspring and parents are then put together while the best of them is selected and sent to the learner phase. In the learner reproduction phase, a particular offspring can be created as

$$\mathbf{x}_{\text{learner}}^i = \begin{cases} \mathbf{x}^i + \text{rand}(\mathbf{x}_{i1} - \mathbf{x}_{i2}), & \text{if } f(\mathbf{x}_{i1}) < f(\mathbf{x}_{i2}), \\ \mathbf{x}^i + \text{rand}(\mathbf{x}_{i2} - \mathbf{x}_{i1}), & \text{if } f(\mathbf{x}_{i2}) < f(\mathbf{x}_{i1}), \end{cases} \quad (11)$$

where  $x_{i1}$  and  $x_{i2}$  are two randomly selected individuals in the population. The greedy selection is then performed in the same manner as with the teaching phase. The computational steps are given in Algorithm 1.

From Algorithm 1, the original TLBO teaching phase is performed using one teacher (the current best solution), while the learning phase is performed exploiting two randomly selected students (individuals). This, to some extent,

leads to limitation in TLBO search exploration and exploitation. Therefore, this work proposed an improved version of TLBO by introducing several numerical schemes. For the proposed self-adaptive teaching-learning based optimizer with acceptance probability algorithm, both teaching and learner phases are upgraded. Multiple teachers are assigned in the teacher phase while a three-student learning scheme is added to the learner phase to enhance its convergence rate. With several new numerical schemes being added, some control parameters are exploited in the new algorithm. While the original TLBO is said to be a derivative-free MH, the proposed algorithm applies self-adaptive strategies to the added control parameters.

In the teaching phase, a diversity archive is used to keep some promising solutions which have good balance between exploration and exploitation. Those solutions are assigned as teachers. The archive is created and updated using the nondominated sorting technique to classify solutions to the archive. The nondominated sorting is operated based on simultaneously minimising the original objective function ( $f(x)$ ) and the diversity objective function ( $f_D(x)$ ) [29]. The diversity function values are calculated based on a combination of the population in the current iteration and the populations from a few previous iterations. Then, the diversity function of those individuals can be computed as

$$f_D(x_i) = w_1 f(x_i) + w_2 f_2(x_i), \quad (12)$$

where  $w_1$  and  $w_2$  are weighting coefficients with the condition  $w_1 + w_2 = 1$ , while  $w_1$  is generated randomly within the range of  $[0,1]$ . The function  $f_2$  can be calculated based on (11)

$$f_2(x_i) = \sum_{j=1}^{n_p} \frac{1}{\max(0.0001, D_{ij})}. \quad (13)$$

The variable  $D_{ij}$  is a Eulerian distance between individuals  $i$  and  $j$ , while  $\max(0.0001, D_{ij})$  is the maximum value between 0.0001 and  $D_{ij}$  which is used to avoid a singularity occurring in the calculation.  $n_p$  is the number of individuals in the pool. (11) is still used for the teaching reproduction phase, but the teacher can be selected between the best solution and those in the diversity archive with a given

Input: Maximum iteration number ( $maxiter$ ), population size ( $n_p$ ).  
Output:  $x_{best}$ ,  $f_{best}$   
Main algorithm  
(1) Initialise population and their objective function values.  
(2) For  $i = 1$  to  $maxiter$ .  
(2.1) Identify the best solution,  $x_{best}$ ,  $f_{best}$  and define  $x_{teacher} = x_{best}$ .  
(Teacher Phase)  
For  $j = 1$  to  $n_p$   
(2.2) Update the population using (10).  
(2.2.1) Evaluate the objective function value.  
(2.2.2) Perform greedy selection.  
End  
(Learner Phase)  
For  $j = 1$  to  $n_p$   
(2.3) Update the population using (11).  
(2.3.1) Evaluate the objective function value.  
(2.3.2) Perform greedy selection.  
End  
(3) End

ALGORITHM 1: TLBO.

probability. The selection of the  $x_{teacher}$  is performed based on a probability of selection which can be expressed as

$$x_{teacher} = \begin{cases} x_{best}, & \text{if } \text{rand} < p_T, \\ x_{D,rand}, & \text{otherwise,} \end{cases} \quad (14)$$

where  $p_T$  is the probability of selecting the best solution, while  $x_{D,rand}$  is an individual randomly selected in the diversity archive (archive of nondominated solutions obtained from  $f_D$  and  $f_2$ ). The probability of selecting the best solution for the teaching reproduction is made self-adaptive based on the accumulated data on each optimization run. Three subintervals for generating  $p_T$  are defined as [0.4, 0.5], [0.5, 0.6], and [0.6, 0.7] where selection of the intervals is carried out using a roulette wheel selection technique. For example, if subinterval one is selected, the value of  $p_T$  is generated as

$$p_T = 0.4 + \text{rand}(0.5 - 0.4), \quad (15)$$

where  $\text{rand} \in [0,1]$  is a uniform random number. The  $j$ -th subinterval has the probability of being selected as  $p_{wj}$ , which can be computed from

$$p_{wj} = \frac{p_{T\_success,j}}{p_{T\_success,j} + p_{T\_fail,j}}. \quad (16)$$

Initially, two  $1 \times 3$  vectors  $p_{T\_success}$  and  $p_{T\_fail}$  whose elements are all zeroes are created. During the teaching reproduction, if the  $j$ -th subinterval is used to generate the value of  $p_T$  and the reproduced offspring is better or as good as its parent, the value of  $p_{T\_success,j}$  should be increased by adding a point to its  $j$ -th element. On the other hand, if it fails to surpass its parent, the value of  $p_{T\_fail,j}$  should be increased by one point. With such a concept, the value of  $p_{wj}$  depends on the history of its use being either a success or a failure. Nevertheless, counting only success or failure can

possibly lead to local optimum traps of the evolution of  $p_{wj}$ ; as a result, the acceptance probability concept is used similarly to the Boltzmann's probability employed in simulated annealing. As a result, in cases that the  $j$ -th subinterval is used leading to an offspring  $x_{teaching}^i$ , the updating scheme for both  $p_{T\_success,j}$  and  $p_{T\_fail,j}$  can be expressed as

$$\begin{aligned} f(x_{teaching}^i) &\leq f(x^i), \\ p_{T\_success,j} &= p_{T\_success,j} + 1, \\ &\text{Else,} \\ &\text{If } \text{rand} < p_{acc}, \\ p_{T\_success} &= p_{T\_success,j} + 0.5, \\ &\text{Else,} \\ p_{T\_fail,j} &= p_{T\_fail,j} + 1, \end{aligned} \quad (17)$$

where  $p_{acc}$  is an acceptance probability set with a high value initially and reduced as the optimization run progresses. Although it has failed, the  $p_{T\_success,j}$  still has a chance to increase its score to 0.5 if the acceptance probability is passed. In this work, simple probability scheduling as displayed in Figure 3 is used, where  $t_{MAX}$  is the maximum iteration number.

For the learner phase, the 2-student learning in (11) is still used along with the 3-student learning strategy [33]. The selection of the two learning strategies relies on the probability of choosing the 2-student learning defined as  $p_L$ . The 3-student learning is achieved by randomly selecting three individuals in the current population, then the search direction is computed in such a way that two other students are directed towards the best student. The new learner reproduction can then be written as



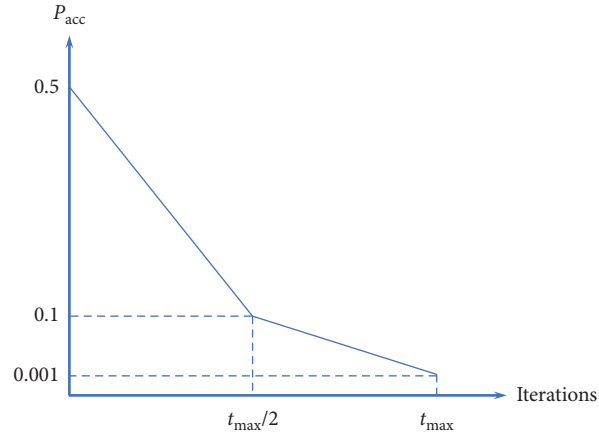


FIGURE 3: Acceptance probability scheduling.

$$x_{\text{learner}}^i = \begin{cases} \text{Eq. (10)}, & \text{if rand} < p_L, \\ x^i + \text{rand}(x_{i3} - x_{i1}) + \text{rand}(x_{i3} - x_{i2}), & \text{otherwise,} \end{cases} \quad (18)$$

where  $x_{i1}$ ,  $x_{i2}$ ,  $x_{i3}$  are randomly selected from the current population and the last one is the best of them. The variable  $p_L$  is generated in a similar way to  $p_T$ . That means there are three subintervals for randomly generating  $p_L$ , while the  $1 \times 3$  vectors  $p_{L\_success}$  and  $p_{L\_fail}$  are used to memorize the successful and failed records of using the  $j$ -th subinterval. Similarly, the probabilities for the roulette wheel selection are computed using (16).

The search process of SaTLBO-AP starts with initialising a population, a schedule of  $p_{acc}$ , and the initial (zero) sets of  $p_{T\_success}$ ,  $p_{T\_fail}$ ,  $p_{L\_success}$  and  $p_{L\_fail}$ . After objective functions of the current population are evaluated, the diversity archive is created. Then, the reproduction process with the teaching and learner phases is performed, and the  $p_{T\_success}$ ,  $p_{T\_fail}$ ,  $p_{L\_success}$  and  $p_{L\_fail}$  sets and the parameter  $p_{acc}$  are updated. The search process is repeated until the termination criterion is satisfied. The computational steps of the proposed SaTLBO-AP are shown in Algorithm 2.

#### 4. Numerical Experiment

To examine the search performance of the proposed algorithm for solving aircraft parameter estimation, the proposed aircraft longitudinal parameter estimation problem in Section 2 was used. The random noise at 0%, 5%, and 10% levels is added into the real flight data, leading to the three cases of the test problem. A number of established MHs along with the proposed algorithm are used to solve such a problem. The MHs used in this study include the following:

Ant Lion Optimizer (ALO) [33], Dragonfly Algorithm (DA) [40], Grasshopper Optimization Algorithm (GOA) [39], Grey Wolf Optimization (GWO) [38], Moth-Flame Optimization Algorithm (MFO) [37], Multi-Verse Optimizer (MVO) [42], Sine Cosine Algorithm (SCA) [43], Salp Swarm Algorithm (SSA) [36], Water Cycle Algorithm (WCA) [41], Whale Optimization Algorithm (WOA) [35], Monarch Butterfly Optimization (MBO) [44], Slime Mould Algorithm (SMA) [45], Elephant Herding

Optimization (EHO) [46], Artificial Bee Colony Algorithm (ABC) [47], Self-adaptive Differential Evolution Algorithm (SaDE) [48], Teaching-Learning Based Optimization (TLBO) [49], Improved Teaching-Learning-Based Optimization (ITLBO) [31], and the proposed algorithm (SaTLBO-AP).

Each optimizer is used to solve the problems for 20 independent runs. The population size and maximum number of iterations are set to be 200 and 250, respectively. For any optimizer using a different population size, they will be terminated at the same number of function evaluations (FEs) of  $200 \times 250 = 50,000$  FEs.

In addition, the conventional OEM has been used to compare with the MH approach. The OEM used in this study is based on the default code from the System Identification Program for Aircraft (SIDPAC) from NASA [50]. Due to the requirement of a predefined initial solution of OEM which is difficult to determine in the real situation, 20,000 initial solutions are generated based on the Latin hypercube sampling in this study. As a result, 20,000 runs from 20,000 initial solutions are performed for the OEM and the results obtained are discussed and compared with the proposed MH. In cases that the solution cannot converge, the OEM search process is terminated at 3,000,000 FEs. It should be noted that the numerical experiment is conducted via MATLAB 2020a with AMD Ryzen 9 5950  $\times$  16-Core Processor 3.40 GHz with 32 GB of ram.

#### 5. Results and Discussion

Having performed 20 independent runs of all MHs for solving the three cases of the optimization for aircraft control parameter estimation, the results are reported in Table 4. Table 4 shows the root mean square error (RMSE) between the longitudinal response of the real aircraft and the calculated response from the longitudinal dynamic equations at the optimum points. The mean values of the RMSE

Input: Maximum iteration number ( $maxiter$ ), population size ( $n_p$ ).  
Output:  $x_{best}$ ,  $f_{best}$ .  
Main algorithm  
(1) Initialise a set of population,  $p_{L\_success}$ ,  $p_{T\_success}$ ,  $p_{T\_fail}$ ,  $p_{L\_fail}$  and  $p_{acc}$ .  
(2) For  $i = 1$  to  $maxiter$ .  
(2.1) Identify the best solution,  $x_{best}$ ,  $f_{best}$  and define  $x_{teacher} = x_{best}$ .  
(Teacher Phase)  
For  $j = 1$  to  $n_p$   
(2.2) Generate  $p_T$  based on  $p_{T\_success, j}$ ,  $p_{T\_fail, j}$ .  
(2.3) Update the population using (10) based on the  $x_{teacher}$  from (14).  
(2.2.1) Evaluate the objective function value.  
(2.2.2) Perform greedy selection.  
(2.2.3) Update  $p_{T\_success, j}$ ,  $p_{T\_fail, j}$  using (17).  
End  
(Learner Phase)  
For  $j = 1$  to  $n_p$   
(2.4) Generate  $p_L$  based on  $p_{L\_success, j}$ ,  $p_{L\_fail, j}$ .  
(2.5) Update the population using (18).  
(2.5.1) Evaluate the objective function value.  
(2.5.2) Perform greedy selection.  
(2.5.3) Update  $p_{L\_success, j}$ ,  $p_{L\_fail, j}$  based on (17).  
End  
Update  $p_{acc}$   
(3) End

ALGORITHM 2: SaTLBO-AP.

TABLE 4: Best, worst, average, standard deviation, and Friedman test of the RMSE at 50,000 function evaluations of 20 individual runs, with and without added noise

Algorithms	Without noise					Noise 5%					Noise 10%				
	Worst	Best	Mean	Std	FR	Worst	Best	Mean	Std	FR	Worst	Best	Mean	Std	FR
ALO	9.0037	0.1529	2.7785	3.3000	8.95	10.2319	0.2295	1.7926	2.7769	7.8	9.0889	0.4015	1.1421	2.1683	5.15
DA	25.1420	0.2640	10.0351	6.6882	14.45	28.0098	0.3375	11.9031	8.3283	14.45	16.8180	0.4513	5.9119	4.8094	13.05
GOA	28.0936	0.9277	18.0601	6.7818	16.85	28.9044	0.8945	20.7342	7.0532	17.2	26.8451	9.1712	20.9930	5.5621	17.35
GWO	1.2739	0.1994	0.3928	0.3098	5.6	1.0688	0.2304	0.4032	0.2196	5.2	0.8363	0.4266	0.5077	0.1188	5.3
MFO	9.2415	0.1832	2.6290	2.7619	9.55	4.6958	0.2581	1.6013	1.5389	8.8	9.4441	0.4307	1.5822	2.1702	8.4
MVO	29.4382	5.4741	17.8161	6.6723	16.6	25.0409	1.6569	14.0516	6.9486	16.3	29.7928	9.0117	16.8057	5.0023	16.8
SCA	8.8118	0.2209	1.4037	1.9144	8.9	3.1936	0.3810	1.4519	0.8058	10.4	2.9651	0.5678	1.3645	0.7551	10.7
SSA	0.8091	0.3462	0.6309	0.1288	7.75	0.7704	0.5509	0.6597	0.0601	7.75	0.9014	0.5689	0.7121	0.0786	8.4
WCA	0.3967	0.0119	0.1100	0.0960	2.9	0.3726	0.1909	0.2217	0.0470	2.65	0.5814	0.3943	0.4338	0.0523	3.1
WOA	24.2830	0.2409	7.9811	8.2753	11.95	24.2636	0.3772	8.9912	7.3915	14.15	20.2952	0.5557	7.1401	6.0350	13.8
MBO	22.7703	0.4050	3.3462	5.5273	10.85	15.0105	0.6030	2.0042	3.1841	10.5	26.3406	0.5751	6.9586	8.7916	13
SMA	11.2174	0.3254	2.1315	2.3528	10.8	13.6034	0.4805	2.4623	3.1702	11.3	25.2341	0.7177	3.3992	6.2542	11.1
EHO	4.0977	0.7582	1.8700	0.7765	11.25	3.4694	0.4009	1.3911	0.7981	10.55	1.1310	0.9659	1.0707	0.0423	10.85
ABC	16.4939	2.3997	7.5099	4.2127	14.6	15.2516	0.9968	5.5834	4.6919	13.5	8.6376	1.0979	3.9645	2.6187	13.35
SaDE	0.1521	0.0028	0.0488	0.0362	2.25	0.2123	0.1919	0.1975	0.0060	2.4	0.4879	0.3977	0.4253	0.0222	2.95
TLBO	4.3630	0.1490	1.7028	1.2234	10.15	6.2615	0.3263	1.6474	1.4422	10.4	3.0237	0.4984	1.3931	0.7821	10.5
ITLBO	1.5246	0.0174	0.4906	0.3604	6.35	2.5530	0.2082	0.5442	0.5402	6.45	1.1887	0.4174	0.5755	0.1820	6.05
SaTLBO-AP	<b>0.0369</b>	<b>0.0024</b>	<b>0.0195</b>	<b>0.0083</b>	<b>1.25</b>	<b>0.1968</b>	<b>0.1899</b>	<b>0.1914</b>	<b>0.0017</b>	<b>1.2</b>	<b>0.4043</b>	<b>0.3960</b>	<b>0.3994</b>	<b>0.0016</b>	<b>1.15</b>

\*FR is the Friedman test score; lower is better.

(Mean) are used to measure the MHs search convergence while the standard deviation values (Std) are used to measure MH search consistency. The results show that, for each case of the design problem, the proposed SaTLBO-AP algorithm is the best performer based on the Friedman ranking, search convergence, and search consistency, while

the second best and third best algorithms are SaDE and WCA, respectively.

Figure 4 shows the search history of the top four algorithms as plots of iterations versus average objective function values. It was found that WCA shows the fastest convergence while SaDE is the slowest from the beginning.

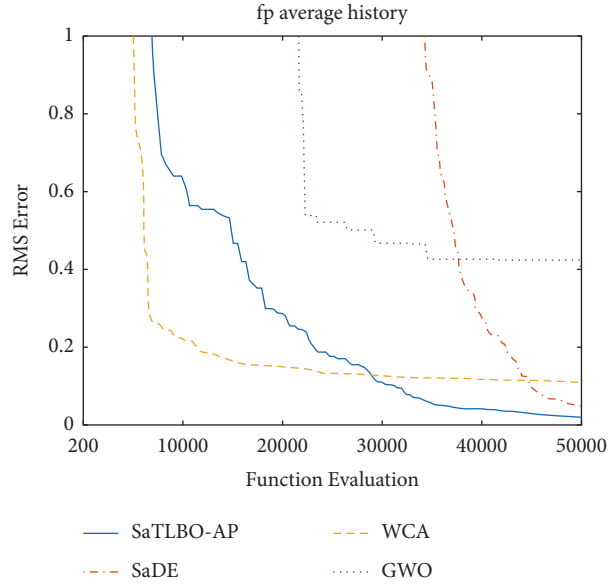


FIGURE 4: Average fitness RMSE values of 20 individual runs without noise from the top 4 algorithms.

TABLE 5: Top 10 best RMSE obtained from 20,000 initial solutions of data with and without noise obtained from OEM.

No.	Without noise	Noise 5%	Noise 10%
1	3.32E-15	0.1896	0.3911
2	1.16E-14	0.4020	0.3911
3	0.4325	0.5277	0.3911
4	0.6074	0.8364	0.3911
5	0.7240	2.4400	0.3911
6	0.9079	2.5514	0.4594
7	0.9877	3.5373	0.4717
8	1.9814	3.5979	0.5643
9	2.1582	4.0832	0.6481
10	3.0534	5.1517	1.9992
Best RMSE obtained from OEM	<b>3.32 E - 15</b>	<b>0.1896</b>	<b>0.3911</b>
Best RMSE obtained from SaTLBO-AP	0.0024	0.1899	0.3960
Average RMSE of the top 10 best obtained from OEM	1.0853	2.3317	0.6098
Average RMSE from SaTLBO-AP	<b>0.0195</b>	<b>0.1914</b>	<b>0.3994</b>

Afterwards, WCA gets trapped at a local optimum after about 10,000 FEs. The proposed SaTLBO-AP seems to converge slower than the WCA initially, however, after 30,000 FEs, the proposed SaTLBO-AP is superior to the WCA and steadily approaches the global optimum. At the end of the optimization run, SaDE managed to reach near the global minimum point, but is still behind SaTLBO-AP. It can be concluded that the proposed SaTLBO-AP is obviously the best algorithm for this problem, is the most robust, and has good balance between search intensification and diversification.

Table 5 shows the top 10 best solutions obtained from 20,000 initial solution of OEM. From the table, it is seen that for the parameter estimation problem without noise, there are only 7 of 20,000 initial solutions where the RMSEs are lower than 1. There are only 4 out of 20,000 and 9 of 20,000 initial solutions with RMSE lower than 1 for the cases of 5%

and 10% noise, respectively. When comparing the best obtained solution based on RMSE of the OEM and the proposed SaTLBO-AP, the OEM seems to have a slightly better RMSE for all cases, due to the use of a gradient-based optimizer. Nevertheless, such solutions can be found only 4–9 times out of 20,000 trials. When comparing the average RMSE obtained from 20 independent runs of the proposed SaTLBO-AP and average RMSE obtained from the top 10 best results classified from 20,000 solutions by using OEM, it is clearly seen that the proposed SaTLBO-AP is better for all cases. Although OEM is advantageous in terms of convergence rate with the use of gradient information, it requires a good predefined initial solution in order to obtain an acceptable solution, which is difficult to identify in a real situation. Based on this study, the possibility to obtain good results is lower than 10/20,000. However, for the proposed SaTLBO-AP based on solving the proposed inverse



TABLE 6: Hansa-3 aerodynamics coefficient simulated without noise, noise 5% and 10%.

True value	Without noise			Noise 5%			Noise 10%			
	SaTLBO-AP	SaDE	WCA	SaTLBO-AP	SaDE	WCA	SaTLBO-AP	SaDE	WCA	
$C_{D_0}$	0.036	0.0367 (0.0079)	0.0368 (0.0086)	0.0437 (0.0253)	0.038 (0.009)	0.0463 (0.0058)	0.0544 (0.0227)	0.0403 (0.0089)	0.023 (0.0076)	0 (0.0172)
$C_{D_\alpha}$	0.061	0.0587 (0.0558)	0.0511 (0.054)	0.0183 (0.2957)	0.0703 (0.0544)	0.0433 (0.0456)	0.0002 (0.273)	0.0368 (0.0566)	0.1879 (0.1109)	0.8955 (0.3237)
$C_{D_{\delta e}}$	0.152	0.1454 (0.0532)	0.1486 (0.0721)	0.0972 (0.2322)	0.1366 (0.0638)	0.065 (0.0511)	0 (0.2145)	0.1611 (0.0698)	0.2274 (0.0868)	0 (0.2046)
$C_{L_0}$	0.23	0.2297 (0.0037)	0.2302 (0.0077)	0.2384 (0.1389)	0.2333 (0.015)	0.2193 (0.0173)	0.126 (0.1284)	0.2416 (0.0178)	0.2005 (0.0504)	0.0001 (0.1133)
$C_{L_\alpha}$	4.886	4.9015 (0.0295)	4.8831 (0.1079)	4.938 (0.814)	5.053 (0.066)	5.1848 (0.0597)	5.3335 (0.7129)	4.846 (0.1371)	4.1228 (0.5722)	5.5128 (1.0446)
$C_{L_q}$	37.259	37.1573 (1.2757)	37.281 (2.1063)	34.3262 (56.1791)	39.5509 (2.9128)	41.9119 (2.2761)	48.286 (37.4155)	43.377 (6.5266)	57.2311 (12.6068)	87.3309 (50.7457)
$C_{L_{\delta e}}$	0.376	0.3755 (0.0406)	0.376 (0.0672)	0.2488 (1.2566)	0.1776 (0.1403)	0.2639 (0.1831)	1.2234 (1.0458)	0.0033 (0.1661)	1.0604 (0.3744)	2.1766 (0.9867)
$C_{m_0}$	0.091	0.091 (0.0002)	0.091 (0.0008)	0.0906 (0.0121)	0.0922 (0.0005)	0.0944 (0.001)	0.0918 (0.0108)	0.1035 (0.0009)	0.0969 (0.0054)	0.1037 (0.0392)
$C_{m_\alpha}$	-0.412	-0.4132 (0.0014)	-0.4119 (0.0037)	-0.4151 (0.1517)	-0.4329 (0.004)	-0.4186 (0.0082)	-0.4239 (0.0933)	-0.4652 (0.0118)	-0.3811 (0.0456)	-0.517 (0.2)
$C_{m_q}$	-8.792	-8.758 (0.0676)	-8.8043 (0.2355)	-8.6207 (3.3145)	-8.6012 (0.1825)	-8.8541 (0.3182)	-8.1431 (2.143)	-10.2257 (0.4649)	-10.221 (1.3053)	-7.5769 (10.9678)
$C_{m_{\delta e}}$	-0.735	-0.7344 (0.0019)	-0.7355 (0.0093)	-0.7288 (0.0954)	-0.73 (0.0063)	-0.763 (0.0093)	-0.7343 (0.073)	-0.8156 (0.0153)	-0.8112 (0.055)	-0.7767 (0.37)

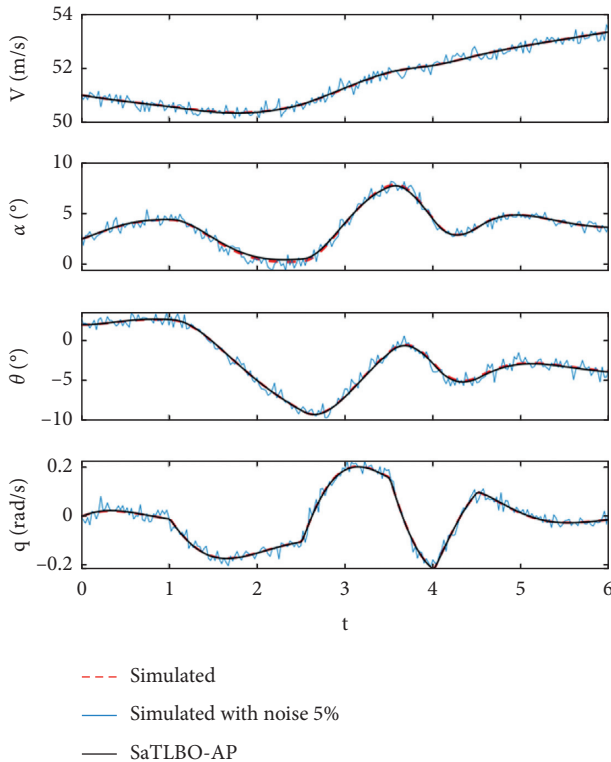


FIGURE 5: Comparison between simulated data with noise 5% and without noise from SaTLBO-AP best result.

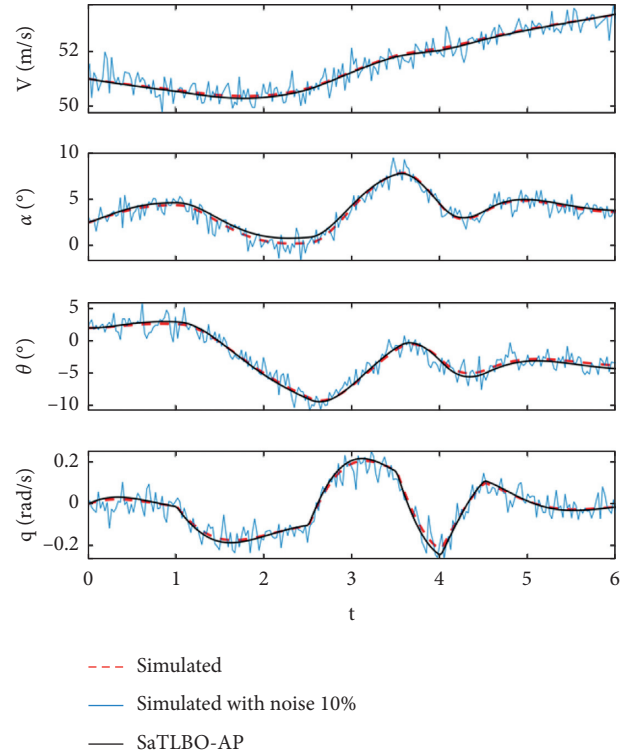


FIGURE 6: Comparison between simulated data with noise 10% and without noise from SaTLBO-AP best result.

optimization problem of aircraft parameter estimation, an acceptable solution can be obtained for all trials without the barrier of predefining an initial solution.

Table 6 shows the comparison between the aerodynamic coefficients and derivatives obtained from using the three best MHs, i.e., SaTLBO-AP, SaDE and WCA and the target values.

The best obtained values (best run) and standard deviation (in parentheses) are presented in the table for each aerodynamic parameter. The results reveal that the aerodynamic coefficients and derivatives obtained from the proposed SaTLBO-AP and SaDE are close to the target values for all cases. Most of the standard deviation values for all aerodynamic coefficients and derivatives obtained from SaTLBO-AP are the lowest for all cases. The output responses obtained from the best run of the proposed SaTLBO-AP compared with target output responses are plotted in Figures 5 and 6.

Overall, it was found that the proposed SaTLBO-AP is the best performer for the case studies of aircraft parameter estimation. Applying the diversity technique, parameter self-adaptation and the acceptance probability concept can increase the MH search performance. The obtained values of the aerodynamic coefficients and derivatives may be slightly different from the actual or desired values; however, if a robust controller is used, such uncertainties can be coped with.

## 6. Conclusions

In this work, a new self-adaptive TLBO is proposed for aircraft parameter estimation. The method is based on integrating a diversity archive and a 3-student learning scheme into the teaching and learner phases respectively. A new self-adaptive strategy with the use of an acceptance probability is proposed. An inverse optimization problem is presented for aircraft longitudinal parameters estimation. The problem is posed to find longitudinal aerodynamic parameters by minimising errors between real flight data and those calculated from the dynamic equations. Several established MHs along with the proposed algorithm are used to solve the proposed optimization problem. The results shown that the top three best algorithms are SaTLOB-AP, SaDE, and WCA, while the proposed SaTLBO-AP is the best performer in both search convergence and consistency. When comparing the proposed SaTLBO-AP with the conventional OEM, the OEM gives better results if a good initial solution is used. Nevertheless, based on this study, only 1 of 20,000 trials for initial solutions of OEM leads to slightly better results than the proposed SaTLBO-AP, while the proposed SaTLBO-AP can obtain acceptable solutions without worrying about an initial guess. However, the proposed algorithm is shown to be suitable for the inverse problem of aircraft parameter estimation, while its performance on other inverse problems still needs investigation. This work is proposed to be the baseline of an investigation that only applies MHs for aircraft parameter estimation. For future work, performance enhancement of the proposed MH for solving the problem and for full model aircraft system identification should be studied.

## Nomenclature

$C_D, C_L, C_m$ :	Drag, lift, pitching moment coefficients
$C_{D_0}, C_{L_0}, C_{m_0}$ :	Trim drag, lift, pitching moment coefficients
$p, q, r$ :	Angular velocities in roll, pitch, and yaw axes
$\phi, \theta, \psi$ :	Roll, pitch, and yaw angles
$u, v, w$ :	Airspeed components in x, y, and z axes
$I_x, I_y, I_z$ :	Moment of inertia about x, y, and z axes

$\Theta$ :	Vector of unknown parameters
$\alpha$ :	Angle of attack
$C_{D\alpha}, C_{D\alpha_e}$ :	$\partial C_D / \partial \alpha \partial C_D / \partial \delta_e$
$L, M, N$ :	Rolling, pitching, and yawing moments
$\bar{q}$ :	Dynamic pressure
$\rho$ :	Air density
$V$ :	True airspeed
$F_{eng}$ :	Engine thrust.

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

The authors are grateful for the support from the Royal Golden Jubilee Ph.D. program (grant no. PHD/0153/2561) and the National Research Council of Thailand (NRCT5-RSA63003-06).

## References

- [1] W. Blake, *Prediction of Fighter Aircraft Dynamic Derivatives Using Digital Datcom*, 1985.
- [2] M. Drela and H. Youngren, "Athena vortex lattice (AVL)," *Computer Software. AVL*, vol. 4, 2008.
- [3] J. D. Anderson and J. Wendt, *Computational Fluid Dynamics*, Springer, Berlin, Germany, 1995.
- [4] V. Klein and E. A. Morelli, *Aircraft System Identification: Theory and Practice*, American Institute of Aeronautics and Astronautics, Reston, VA, USA, 2006.
- [5] N. K. Peyada, A. Sen, and A. K. Ghosh, "Aerodynamic characterization of HANSA-3 aircraft using equation error, maximum likelihood and filter error methods," *Lecture Notes in Engineering and Computer Science*, vol. 2169, no. No. 1, pp. 1902–1907, 2008.
- [6] E. A. Morelli, "Practical aspects of the equation-error method for aircraft parameter estimation," in *Proceedings of the Collection of Technical Papers - 2006 Atmospheric Flight Mechanics Conference*, vol. 1, pp. 345–362, Colorado, CA, USA, August, 2006.
- [7] J. T. Olkkonen and H. Olkkonen, "Least squares matrix algorithm for state-space modelling of dynamic systems," *Journal of Signal and Information Processing*, vol. 2, no. 4, pp. 287–291, 2011.
- [8] C. Kamali, A. A. Pashilkar, and J. R. Raol, "Evaluation of recursive least squares algorithm for parameter estimation in aircraft real time applications," *Aerospace Science and Technology*, vol. 15, no. 3, pp. 165–174, 2011.
- [9] R. V. Jategaonkar and E. Plaetschke, "Algorithms for aircraft parameter estimation accounting for process and measurement noise," *Journal of Aircraft*, vol. 26, no. 4, pp. 360–372, 1989.
- [10] L. C. S. Góes, E. M. Hemerly, B. C. De Oliveira Maciel, W. R. Neto, C. B. Mendonca, and J. Hoff, "Aircraft parameter estimation using output-error methods," *Inverse Problems in Science and Engineering*, vol. 14, no. 6, pp. 651–664, 2006.

- [11] J. A. Grauer and M. J. Boucher, "Identification of aeroelastic models for the X-56a longitudinal dynamics using multisine inputs and output error in the frequency domain," *Aerospace*, vol. 6, no. 2, 2019.
- [12] M. Majeed and J. Singh, "Frequency and time domain recursive parameter estimation for a flexible aircraft," *IFAC*, 2013.
- [13] M. E. Campbell and S. Brunke, "Nonlinear estimation of aircraft models for on-line control customization," *IEEE Aerospace Conference Proceedings*, vol. 2, pp. 2621–2628, 2001.
- [14] A. G. Kallapur and S. G. Anavatti, "UAV linear and nonlinear estimation using extended kalman filter," in *Proceedings of the International Conference on Computational Intelligence for Modelling, Control and Automation, Jointly with IAWTIC 2006: International Conference on Intelligent Agents Web Technologies CIMCA 2006*, Sydney, Australia, December 2006.
- [15] G. Chowdhary and R. Jategaonkar, "Aerodynamic parameter estimation from flight data applying extended and unscented kalman filter," *Aerospace Science and Technology*, vol. 14, no. 2, pp. 106–117, 2010.
- [16] S. Singh and A. K. Ghosh, "Estimation of lateral-directional parameters using neural networks based modified delta method," *Aeronautical Journal*, vol. 111, no. 1124, pp. 659–667, 1968.
- [17] S. C. Raisinghani and A. K. Ghosh, "Parameter estimation of an aeroelastic aircraft using neural networks," *Sadhana - Academy Proceedings in Engineering Sciences*, vol. 25, no. 2, pp. 181–191, 2000.
- [18] N. K. Peyada and A. K. Ghosh, "Aerodynamic parameter estimation using new filtering technique based on neural Network and gauss-Newton method," *Arms*, vol. 3273, pp. 243–252, 2008.
- [19] X. Li and M. Yin, "Parameter estimation for chaotic systems by hybrid differential evolution algorithm and artificial bee Colony algorithm," *Nonlinear Dynamics*, vol. 77, no. 1–2, pp. 61–71, 2014.
- [20] A. Ghosh Roy and N. K. Peyada, "Aircraft parameter estimation using hybrid neuro Fuzzy and artificial bee Colony optimization (HNFABC) algorithm," *Aerospace Science and Technology*, vol. 71, pp. 772–782, 2017.
- [21] I. Khoja, T. Ladhari, F. M'sahli, and A. Sakly, "Cuckoo search approach for parameter identification of an activated sludge process," *Computational Intelligence and Neuroscience*, vol. 2018, Article ID 3476851, 9 pages, 2018.
- [22] S. Slesongsom and S. Bureerat, "Optimal synthesis of four-bar linkage path generation through evolutionary computation with a novel constraint handling technique," *Computational Intelligence and Neuroscience*, vol. 2018, Article ID 5462563, 16 pages, 2018.
- [23] R. Soto, B. Crawford, A. Aste Toledo et al., "Solving the manufacturing cell design problem through binary cat Swarm optimization with dynamic mixture ratios," *Computational Intelligence and Neuroscience*, vol. 2019, Article ID 4787856, 16 pages, 2019.
- [24] A. Espinal, H. Rostro-Gonzalez, M. Carpio et al., "Quadrupedal robot locomotion: a biologically inspired approach and its hardware implementation," *Computational Intelligence and Neuroscience*, vol. 2016, Article ID 5615618, 16 pages, 2016.
- [25] S. Bureerat and N. Pholdee, "Inverse problem based differential evolution for efficient structural health monitoring of trusses," *Applied Soft Computing*, vol. 66, pp. 462–472, 2018.
- [26] S. Bureerat and N. Pholdee, "Adaptive sine cosine algorithm integrated with differential evolution for structural damage detection," *Lecture Notes in Computer Science*, 2017.
- [27] K. Wichapong, S. Bureerat, and N. Pholdee, "Solving inverse kinematics of robot manipulators by means of meta-heuristic optimisation," *IOP Conference Series: Materials Science and Engineering*, vol. 370, no. 1, 2018.
- [28] S. Bureerat, N. Pholdee, T. Radpukdee, and P. Jaroenapibal, "Self-adaptive MRPBIL-DE for 6D robot multiobjective trajectory planning," *Expert Systems with Applications*, vol. 136, pp. 133–144, 2019.
- [29] S. Bureerat and S. Slesongsom, "Constraint handling technique for four-bar linkage path generation using self-adaptive teaching-learning-based optimization with a diversity archive," *Engineering Optimization*, pp. 1–18, 2020.
- [30] K. Yu, X. Chen, X. Wang, and Z. Wang, "Parameters identification of photovoltaic models using self-adaptive teaching-learning-based optimization," *Energy Conversion and Management*, vol. 145, pp. 233–246, 2017.
- [31] S. Li, W. Gong, X. Yan et al., "Parameter extraction of photovoltaic models using an improved teaching-learning-based optimization," *Energy Conversion and Management*, vol. 186, pp. 293–305, 2019.
- [32] A. Srivastava, A. Kumar, and A. Kanti Ghosh, "Estimation of longitudinal aerodynamic derivatives using genetic algorithm optimized method," *American Journal of Engineering and Technology Management*, vol. 4, no. 2, p. 34, 2019.
- [33] S. Slesongsom and S. Bureerat, "Four-bar linkage path generation through self-adaptive population size teaching-learning based optimization," *Knowledge-Based Systems*, vol. 135, pp. 180–191, 2017.
- [34] S. Mirjalili, "The Ant lion optimizer," *Advances in Engineering Software*, vol. 83, pp. 80–98, 2015.
- [35] S. Mirjalili and A. Lewis, "The Whale optimization algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, 2016.
- [36] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, "Salp Swarm algorithm: a bio-inspired optimizer for engineering design problems," *Advances in Engineering Software*, vol. 114, pp. 163–191, 2017.
- [37] S. Mirjalili, "Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm," *Knowledge-Based Systems*, vol. 89, pp. 228–249, 2015.
- [38] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.
- [39] S. Saremi, S. Mirjalili, and A. Lewis, "Grasshopper optimization algorithm: theory and application," *Advances in Engineering Software*, vol. 105, pp. 30–47, 2017.
- [40] S. Mirjalili, "Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems," *Neural Computing & Applications*, vol. 27, no. 4, pp. 1053–1073, 2016.
- [41] H. Eskandar, A. Sadollah, A. Bahreininejad, and M. Hamdi, "Water Cycle algorithm - a novel metaheuristic optimization method for solving constrained engineering optimization problems," *Computers & Structures*, vol. 110–111, pp. 151–166, 2012.
- [42] S. Mirjalili, S. M. Mirjalili, and A. Hatamlou, "Multi-verse optimizer: a nature-inspired algorithm for global optimization," *Neural Computing & Applications*, vol. 27, no. 2, pp. 495–513, 2016.
- [43] S. Mirjalili, "SCA: a Sine Cosine Algorithm for solving optimization problems," *Knowledge-Based Systems*, vol. 96, pp. 120–133, 2016.

- [44] G. G. Wang, S. Deb, and Z. Cui, "Monarch butterfly optimization," *Neural Computing & Applications*, vol. 31, no. 7, pp. 1995–2014, 2019.
- [45] S. Li, H. Chen, M. Wang, A. A. Heidari, and S. Mirjalili, "Slime mould algorithm: a new method for stochastic optimization," *Future Generation Computer Systems*, vol. 111, pp. 300–323, 2020.
- [46] G.-G. Wang, S. Deb, and L. Coelho, "Elephant herding optimization," in *Proceedings of the 2015 3rd International Symposium on Computational and Business Intelligence (ISCBI)*, pp. 1–5, Bali, Indonesia, December 2015.
- [47] B. Basturk, "An artificial bee Colony (ABC) algorithm for numeric function optimization," in *Proceedings of the IEEE Swarm Intelligence Symposium*, pp. 181–184, Indianapolis, IN, USA, May 2006.
- [48] A. K. Qin and P. N. Suganthan, "Self-adaptive differential evolution algorithm for numerical optimization," in *Proceedings of the 2005 IEEE Congress on Evolutionary Computation*, vol. 2, pp. 1785–1791, Edinburgh, Scotland, UK, September 2005.
- [49] R. V. Rao, V. J. Savsani, and D. P. Vakharia, "Teaching-learning-based optimization: a novel method for constrained mechanical design optimization problems," *CAD Computer Aided Design*, vol. 43, no. 3, pp. 303–315, 2011.
- [50] E. Morelli, *System Identification Programs for Aircraft (SID-PAC)*, 2002.