Research Article

Morphing Wing Structural Optimization Using Opposite-Based Population-Based Incremental Learning and Multigrid Ground Elements

S. Sleesongsom[1] and S. Bureerat[2]

[1]Department of Mechanical Engineering, Faculty of Engineering, Chiangrai College, Chiangrai 57000, Thailand
[2]Sustainable and Infrastructure Research and Development Center, Department of Mechanical Engineering, Faculty of Engineering, Khon Kaen University, Khon Kaen 40002, Thailand

Correspondence should be addressed to S. Bureerat; sujbur@kku.ac.th

This paper has twin aims. Firstly, a multigrid design approach for optimization of an unconventional morphing wing is proposed. The structural design problem is assigned to optimize wing mass, lift effectiveness, and buckling factor subject to structural safety requirements. Design variables consist of partial topology, nodal positions, and component sizes of a wing internal structure. Such a design process can be accomplished by using multiple resolutions of ground elements, which is called a multigrid approach. Secondly, an opposite-based multiobjective population-based incremental learning (OMPBIL) is proposed for comparison with the original multiobjective population-based incremental learning (MPBIL). Multiobjective design problems with single-grid and multigrid design variables are then posed and tackled by OMPBIL and MPBIL. The results show that using OMPBIL in combination with a multigrid design approach is the best design strategy. OMPBIL is superior to MPBIL since the former provides better population diversity. Aeroelastic trim for an elastic morphing wing is also presented.

1. Introduction

Design/optimization of a morphing wing aircraft structure is a research field investigated throughout the world. It has been found that the shape and structural flexibility of a morphing aircraft greatly affect its flight performance [1]. This leads to continuous shape change and is called a morphing concept, which is an attempt to avoid using conventional hinged control surfaces. From the literature, external plane transformation of the wing is a recent popular morphing concept [1]. The idea is to change wing aerodynamic performance by deforming the initial wing plane, which can be categorized as wing camber variation [1–5], lateral wing bending [1], and wing twisting [5–9]. The variable camber of an aircraft wing is the most popular method [1], which can be implemented via a conventional hinged mechanism, a smart material, or a compliant mechanism. The compliant mechanism is the best method for such a morphing aircraft wing concept [4]. In

the lateral wing bending concept, the wing can bend up and down in a vertical direction so as to produce a wing span camber. Using this concept, wing aerodynamic performance can be varied and it is carried out by using an internal mechanism. The last concept, the wing twisting, can adjust a wing angle of attack by using structural flexibility and applying external torque. Later, there has been some work attempting to exploit adaptive internal structures [6]. The concept can be thought of as a combined lateral wing bending and variable camber concept. From the literature [1–4], it is revealed that the morphing wing aircraft structure is actuated directly by an external force to carry out such aircraft control. Furthermore, the morphing wing structures are elastic rather than perfectly rigid; thus, consideration of their aeroelastic behavior is essential in a design process. This leads to recent work in which the effects of the external force on aeroelastic characteristic of a morphing aircraft wing are studied [10]. The result shows that the actuating force has a significant

impact on the aeroelastic and mechanical characteristics and these effects should be taken into account during the design process of a morphing aircraft structure. Also, the aircraft internal structure including the structural layout and sizes significantly affects the aeroelastic characteristics.

In our recent work, it has been proposed to synthesize new internal structural layouts of an aircraft wing [8] and an aircraft morphing wing [9] using multiobjective optimization. The first study leads to an unconventional wing internal layout, which is aeroelastically superior to a traditional rib-spar layout. Design variables include partial topology and sizes of segments on a structure, while design objectives are wing weight, lift effectiveness, and buckling factor. The second study found the unconventional wing structures subject to external actuating torques which are practicable to apply for the morphing aircraft. Design variables include partial topology and sizes of segments on a structure like the first study, but the second study included nodal displacements. Design objectives are wing weight, percentage of change in lift effectiveness, and buckling factor. In both study, the combination of the two types of design variables is carried out by using a ground element approach. The method used to tackle the design problems is multiobjective population-based incremental learning (MPBIL). The optimizer used herein is one of the most powerful evolutionary methods for solving structural topology optimization [11]. The method is category as nongradient optimization methods, which does not require gradient information to converge to a solution. The design strategy proposed in these works can be used as a numerical tool for synthesizing an internal structural layout of a morphing wing, which usually requires high structural and aeroelastic performance for flight operation [8, 9].

The main motivation of this research work is to initiate an idea to synthesize unconventional aircraft wing structures that are expected to outperform their conventional counterparts. This paper is intended as an extension of the above literature. The work has two aims. Firstly, the use of several resolutions of ground elements at the same time is proposed instead of using one ground element resolution for a design problem and this is termed a multigrid ground element approach. Design variables are simultaneous topology, shape, and sizing optimization. The second part is performance enhancement of a multiobjective evolutionary optimizer MPBIL. The opposite-based evolutionary optimization is used to improve the search performance of the original MPBIL and it is named opposite-based multiobjective PBIL (abbreviated as OMPBIL). MPBIL and OMPBIL are employed to solve a morphing wing synthesizing design problem with the use of single-grid and multigrid ground elements. Optimum results reveal that OMPBIL is superior to MPBIL. The multigrid approach leads to better design results than the single-grid one, while the set of design variables that include structural layout, nodal positions, and sizes gives the best structural layout.

The rest of this paper is organized as follows. Section 2 details single- and multiground element design approaches for synthesizing a wing's internal structure. The opposite-based multiobjective population-based incremental learning is proposed in Section 3. A design problem and its conditions as well as a numerical experiment are given in Section 4, while the design results are in Section 5. The conclusions and discussion of the study are finally drawn in Section 6.

2. Partial Topology, Shape, and Sizing Optimization

Shape and sizing optimization [12] and simultaneous partial topology integrate three different types of structural design variables in the same design problem. This is somewhat complicated but found to be an efficient design strategy [8, 9, 13]. Such a design strategy can be achieved by using the ground finite element technique. The ground element technique is commonly used in a topology optimization process. The basic idea is to generate ground finite elements throughout a design domain being considered. Design variables determine pseudo-densities (such as element thickness and modulus) of those ground elements. Having obtained an optimum solution (e.g., from compliance minimization or dynamic stiffness maximization), ground elements with low pseudo-densities are assigned as holes on a structure whereas elements with high pseudo-density represent a structure. This concept has been proven effective for many design cases [11, 14, 15].

For simultaneous topology, shape, and sizing optimization of an aircraft wing, ground elements include all possible combinations of wing internal segments and wing skins. Figures 1(a) and 1(b) display the ground elements or ground segments for the wing internal structure used for design demonstration in this work. The details of this simple wing-box structure can be found in our previous work [8, 9]. Note that diagonal segments of a structure are used because it has been found that such ground segments result in aeroelastically superior structures compared to using only chordwise and spanwise segments as with a conventional wing structure [8]. The upper and lower wing skins shown in Figure 2 are also set as design variables. It has been found that the variable-thickness of element panels can enhance its aeroelastic performance [5, 16]. The combination of topology and sizing variables of the wing is carried out by assigning values to the thicknesses of those segments in meters (m). For example, let $x_i \in [0.0005, 0.001]$ m be design variables for segment thickness where the lower and upper bounds are based on manufacturing tolerance. The ith ground segment will be removed from the main structure if the value of x_i is close to its lower bound at the optimum point. Other thickness values higher than the lower bounds are used for structural sizing. It should be noted that the ideal lower bounds for topological design variables are zero segment thickness. However, in order to prevent singularity in a global stiffness matrix of a wing, we have to use a small positive value as lower bounds of the topological design variables. Figure 1 illustrates how to transform from particular design variables (Figure 1(a)) to be a wing structure (Figure 1(b)). Shape design variables, on the other hand, can be added to the partial topology and sizing optimization by assigning positions of internal nodes of those ground segments as design variables as illustrated in Figure 3. The shape design variables determine the positions of the external nodes in
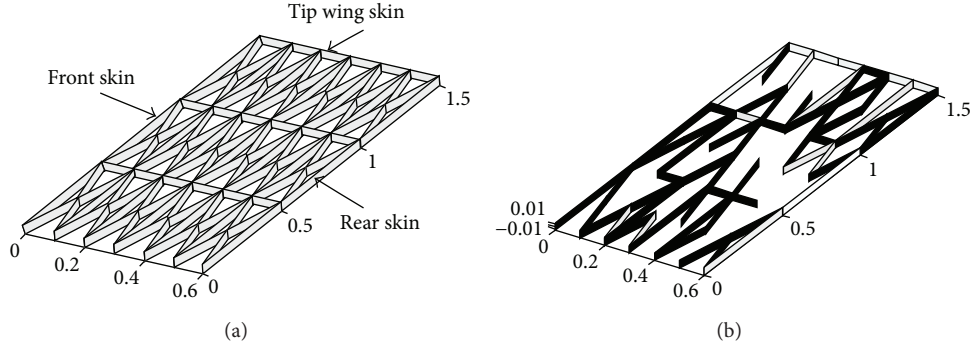
FIGURE 1: Wing-box (a) ground segments without nodal position and (b) an internal structural layout of a wing-box.
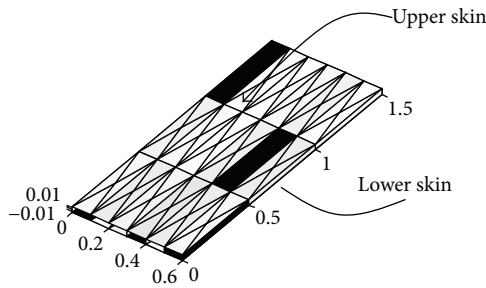

FIGURE 2: 6×3 ground segments for wing-box skins without nodal position.

x- and y-directions. Figure 4 shows the segments on lower and upper wing skins as the internal nodes positions are varied. The skin segments are also set as design variables.

The design problem is posed to minimize three design objectives simultaneously, while fulfilling safety constraints. A morphing wing in this study is based on the wing twisting concept. To achieve flight control, the wing is subject to a twisting actuating force on the wing internal structure as shown in Figure 5. This means that the wing generates lift variation through change of lift effectiveness due to actuation. Thus, the first objective function is the difference percentage between wing lift effectiveness with and wing lift effectiveness without actuating force. This function can be expressed as

$$\%\Delta\eta_L = \frac{(\eta_{L,\text{Fe}} - \eta_L)}{\eta_L} \times 100, \tag{1}$$

where $\eta_{L,\text{Fe}}$ is the lift effectiveness considering the effect of an external force. The second objective function is set to minimize structural mass, which is rather an unavoidable requirement for aircraft structural design. The third objective, buckling factor maximization, is set to increase the strength of the structure. The wing is considered as a thin-wall structure in which one of the most critical design criteria is buckling instability. The multiobjective design problem can be expressed as

$$\min \quad \begin{cases} f_1 = -\%\Delta\eta_L \\ f_2 = M \\ f_3 = -\lambda \end{cases}$$

$$\text{subject to} \quad V_F, V_D \geq 150\,\text{m/s}$$

$$\lambda \geq 1$$

$$N = \frac{\sigma_{yt}}{\sigma} \geq 1 \tag{2}$$

$$0.0005 \leq x_i \leq 0.001\,\text{m}$$

$$-0.075 \leq dc_j \leq 0.075\,\text{m}$$

$$-0.225 \leq dsp_j \leq 0.225\,\text{m},$$

where M is the structural mass, V_D is the divergence speed, V_F is the flutter speed of the wing, N is the safety factor, σ is the maximum equivalent stress on the wing due to aerodynamic and actuating forces, σ_{yt} is the yield stress, λ is the buckling factor, x_i is the thickness of a wing segment i, dc_j is the nodal position in the chordwise direction of node j, and dsp_j is the nodal position in the spanwise direction of node j. The last two constraints are assigned so as to prevent node positions and wing segments overlapping each other in cases of performing simultaneous partial topology, shape, and sizing design. For details of structural and aeroelastic models as well as other function evaluation procedures, see Sleesongsom and Bureerat [8, 9].

A question always arises at the preprocess stage [17], when using a ground element approach: what is the best ground element resolution for a design problem? As a result, using several sets of ground segments at the same time when performing optimization is investigated and it is termed a multigrid design approach. The multigrid approach is said to be an extension of the ground segment strategy, which has been proposed to solve a truss structure [18].
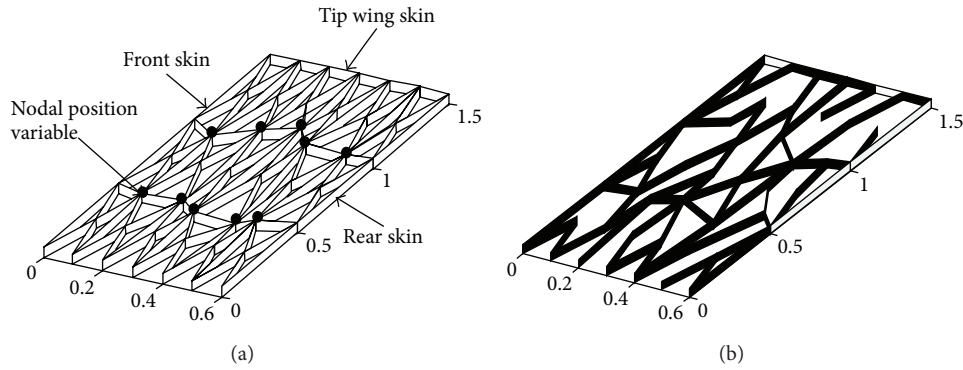
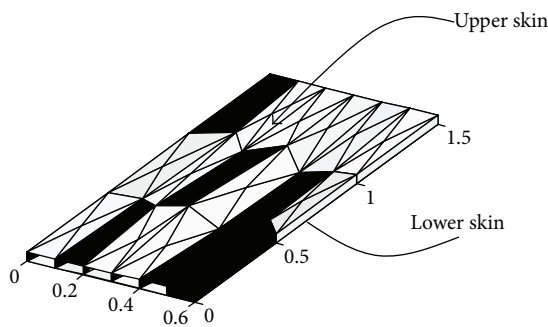FIGURE 3: Ground segments with nodal position variables (●) and an internal structural layout of a wing-box.

FIGURE 4: 6×3 ground segments for wing-box skins with nodal position variables.

Figure 6 shows three sets of ground segments with different grid resolutions used in this study. The first set of ground elements has 99 segments, which was used in the previous study [8]. The second set has 82 segments, while the third set has 53 segments. Using one set of ground elements is simple to perform since we can assign thickness values to all segments (and to nodal positions in cases of shape design variables). Nevertheless, when using many ground segment sets, a special encoding/decoding scheme for multiobjective evolutionary optimization is needed [8]. From Figure 6, let the first set of ground elements have N_{11} segments and N_{12} nodal positions. The second set has N_{21} segments and N_{22} nodal positions, while the third set has the numbers of ground segments and nodal positions as N_{31} and N_{32}, respectively. Given that $N_{11} \geq N_{21} \geq N_{31}$ and $N_{12} \geq N_{22} \geq N_{32}$, the variables encoding/decoding scheme for the multigrid design approach can be detailed as in Algorithm 1. As MPBIL and OMPBIL use binary design variables, the conversion of a binary string to become a real design vector $\mathbf{x}$ needs to be performed before entering Algorithm 1.

3. MPBIL and OMPBIL

This section details the two algorithms MPBIL and the new optimizer OMPBIL, which will be used to tackle the optimization presented in Section 2. Evolutionary algorithms (EAs) are alternative optimizers to classical gradient-based optimizers such as a steepest descent method and sequential linear programming. Unlike the gradient-based optimizers, EAs search for optima based upon evolutionary concepts in combination with randomization [11]. The methods, for example, genetic algorithms, are simple to use, robust, and capable of tackling global optimization. They can be used for almost any kinds of optimum design problems (such as discrete [11], continuous [14], and combinatorial optimization [19]) since they can search for optimum solutions without using function derivatives. In our design problem, there is no accuracy guarantee for function evaluation especially the flutter speed [10]. Apart from that, our new design variables are mixed integer/continuous where it is difficult to apply any gradient-based optimizer to solve the design problem. The successful use of EAs for complex engineering design has been reported, for example, in Lencus et al. [20] and G. Nowak and I. Nowak [21]. EAs for multi-objective optimization, traditionally termed multiobjective evolutionary algorithms (MOEAs), have one outstanding feature; that is, they can explore a nondominated front within one simulation run. Although having several advantages, two unavoidable drawbacks of EAs and MOEAs are the lack of convergence rate and consistency [18]. This makes EAs ineffective when solving topology optimization, which is said to be a large-scale design problem with a large number of design variables. Over the past two decades, many attempts to improve search performance of EAs for solving structural topology optimization have been made, for example, a morphological geometric representation approach [22], a chromosome repairing technique [23], a ground element filtering technique [11], and antioptimization [24]. The work in Kunakote and Bureerat [11] presents the comparative performance of MOEAs for solving structural topology optimization test problems based on the ground element filtering technique. The established MOEAs included MPBIL, Pareto archive evolution strategy (PAES) [25], nondominated sorting genetic algorithm (NSGA) [26], strength Pareto evolutionary algorithm (SPEA) [27], and multiobjective particle swarm optimization (MPSO) [28]. The results show that MPBIL is the unanimous best optimizer [29]. The simple concept of estimation of distribution algorithm embedded in MPBIL gives an advantage when dealing with multiobjective topology optimization. As a result, MPBIL is the only MOEA

<u>Initialization</u> Generate three sets of ground segments and proper finite element grid for each set.
<u>Input</u> $\mathbf{x}$ sized $(N_{11} + N_{12} + 1) \times 1$
<u>Output</u>: Thicknesses of wing segments and nodal positions
<u>Encoding</u>
$x_1 \in [1, 3]$ is used for selecting a set of ground elements
x_2 to $x_{N_{11}+1}$ are used for segment thicknesses
$x_{N_{11}+2}$ to $x_{N_{11}+N_{12}+1}$ are used for nodal position variables
<u>Decoding</u>
(1) Find $n = \text{round}(x_1)$ where round$(\cdot)$ is a round-off operator.
(2) If $n = 1$
  (2.1) x_2 to $x_{N_{11}+1}$ are set as N_{11} segment thicknesses
  (2.2) $x_{N_{11}+2}$ to $x_{N_{11}+N_{12}+1}$ are set as N_{12} nodal position variables
(3) If $n = 2$
  (3.1) x_2 to $x_{N_{21}+1}$ are set as N_{21} segment thicknesses
  (3.2) $x_{N_{11}+2}$ to $x_{N_{11}+N_{22}+1}$ are set as N_{22} nodal position variables
(4) If $n = 3$
  (3.1) x_2 to $x_{N_{31}+1}$ are set as N_{31} segment thicknesses
  (2.2) $x_{N_{11}+2}$ to $x_{N_{11}+N_{31}+1}$ are set as N_{32} nodal position variables

ALGORITHM 1: Design variables encoding/decoding scheme for a multigrid design approach.
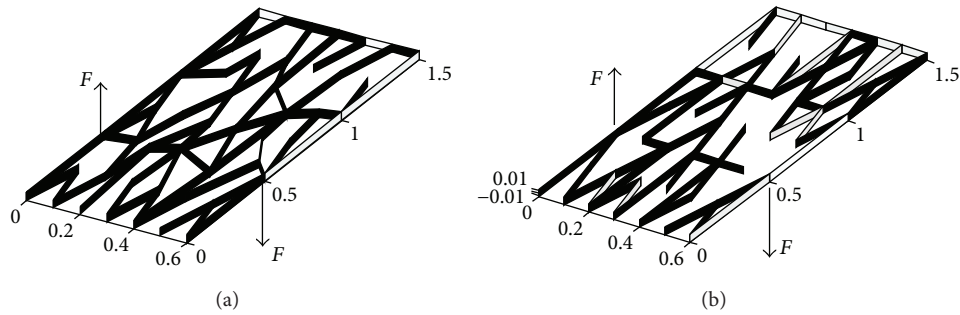

FIGURE 5: Internal wing models with actuator torque.

chosen to solve this design problem. This section briefly details the concept of MPBIL and then it is extended to the opposition-based multiobjective population-based incremental learning.

3.1. Multiobjective Population-Based Incremental Learning.
The principle of a PBIL search is governed by the learning rule, which is iteratively limiting the design space depending on current best design variables, and a random process [29]. The design space is iteratively narrowed until the optimum is approached. Unlike the popular genetic algorithm and other evolutionary algorithms (EAs), a binary population used in PBIL is represented by an estimated probability distribution for each binary bit, which is called a probability vector. Table 1 shows three examples of probability vectors and their corresponding binary populations given that a design solution is a row of a population. The ith element of a probability vector determines the probability of having a string "1" on the ith column of the population. It can be seen that one probability vector can produce a variety of binary populations.

Unlike single objective PBIL, MPBIL uses several probability vectors to represent a binary population in order

TABLE 1: Probability vectors and corresponding binary populations.

Population 1				Population 2				Population 3			
0	0	1	1,	1	1	1	0,	0	0	0	1
1	1	1	0,	0	1	0	0,	0	1	0	0
0	1	0	1,	1	0	1	1,	1	0	0	1
1	0	0	0,	0	0	0	1,	0	1	0	1
				Probability vectors							
[0.5, 0.5, 0.5, 0.5]				[0.5, 0.5, 0.5, 0.5]				[0.25, 0.5, 0, 0.75]			

to maintain population diversity. Thus, a set of probability vectors is called a probability matrix [29]. The search procedure of MPBIL starts with an initial Pareto archive for collecting nondominated solutions and a probability matrix whose elements are full of "0.5." Each row of the probability matrix is a probability vector that will be used to create a subpopulation. Let N_P be the number of design solutions in a population, l the number of probability vectors, and n_b the number of binary bits for one design vector $\mathbf{x}$. The probability matrix, therefore, has the size of $l \times n_b$ where each row of the matrix results in approximately N_P/l design solutions as
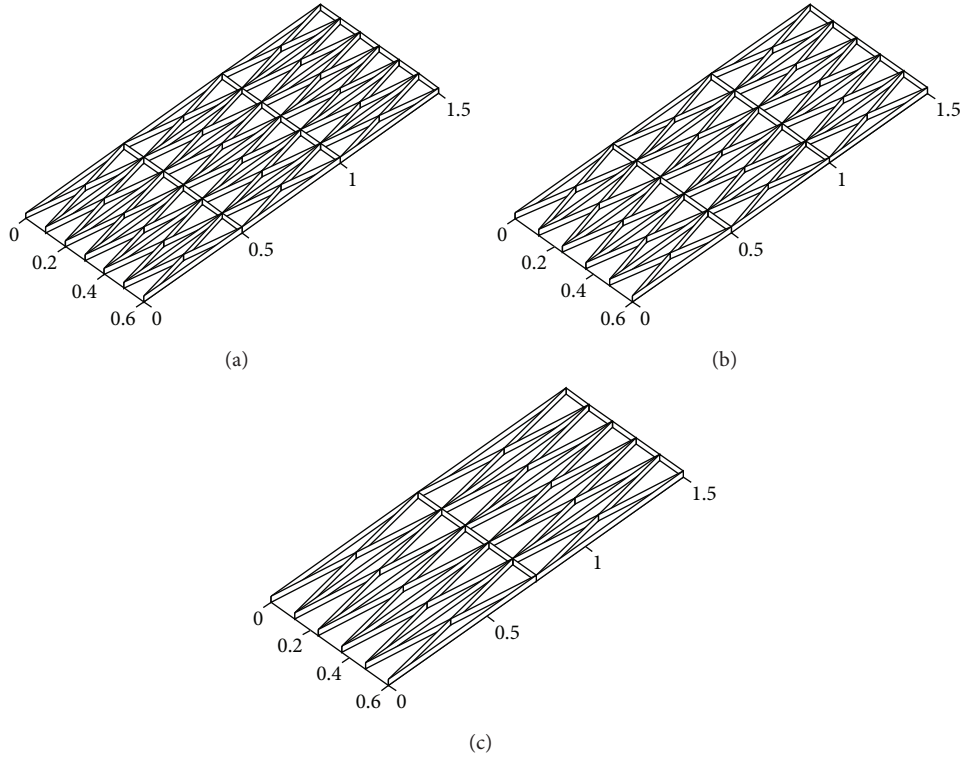
(a)

(b)

(c)

Figure 6: Three ground structure resolutions: (a) 99 internal segments, (b) 82 internal segments, and (c) 53 segments.
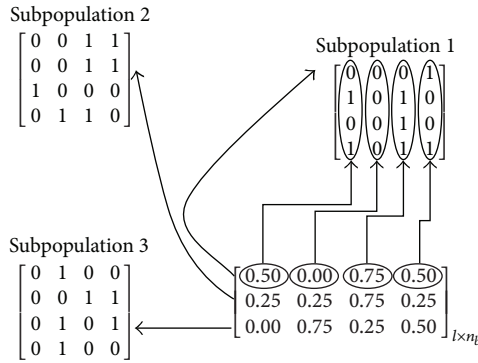

Figure 7: Probability matrix and subpopulations.

one subpopulation. Each design variable is represented by a binary string with n_b/n elements. Figure 7 displays a 3×4 probability matrix and its corresponding population ($N_P = 12$), which is the union set of all three subpopulations according to the three rows of the probability matrix [30].

Having generated the population and evaluated their corresponding objective values, the nondominated members sorted from the combination of members in the current population and the old external Pareto archive are taken as a new external Pareto set. If the external Pareto set is full, some solutions are removed from the external Pareto set using an archiving technique [29]. In MOEAs, the archiving technique is proposed to filter some nondominated solutions

out of the archive matrix to prevent excessive memory usage of a computer because MOEAs can normally explore a large number of Pareto solutions within one run. The probability matrix and the nondominated solutions are improved iteratively until the termination criterion is met.

For the updating scheme for the probability matrix on each generation, the matrix is updated in such a way that $n_0 < N_A$ binary solutions are randomly selected from the current Pareto archive to modify the ith row of the probability matrix. The ith row of the probability matrix is updated using the relation

$$P_{ij}^{\text{new}} = P_{ij}^{\text{old}} \left(1 - L_R\right) + b_j L_R, \tag{3}$$

where $L_R \in (0,1)$ is called the learning rate to be defined and b_j is the average value of the jth bit position of those n_0 randomly selected nondominated solutions. In our work, the learning rate is set as

$$L_R = 0.5 + \text{rand} \left(+0.1 \text{ or} -0.1\right), \tag{4}$$

where rand $\in [0,1]$ is a uniform random number. It is also useful to apply mutation to the probability matrix at some predefined probability as

$$P_{ij}^{\text{new}} = P_{ij}^{\text{old}} \left(1 - \text{ms}\right) + \text{rand} \left(0 \text{ or } 1\right) \cdot \text{ms}, \tag{5}$$

where ms (default = 0.2) is the amount of shift used in the mutation.

The procedure for MPBIL is detailed in Algorithm 2. For more details of MPBIL, see Bureerat and Sriworamas [29] and Bureerat and Srisomporn [30].

<u>Initialization</u> probability matrix $\mathbf{P} = [\mathbf{0.5}]_{l \times n_b}$, external Pareto archive **Pareto** = {}.
(1) Generate a binary population **B** from **P** (see Figure 8).
(2) Decode the binary population to be $\mathbf{x}_{n \times N_p}$ and find the objective values $\mathbf{f}_{m \times N_p}$.
(3) Update **Pareto** by replacing it with the non-dominated solutions of a union set **Pareto** ∪ **x**.
(4) If the number of members in **Pareto** exceeds the predefined archive size N_A, remove some of them by using an archiving
    technique.
(5) If the termination criterion is fulfilled, stop the procedure. Otherwise, go to step 6:
(6) Update **P**.
(6.1) For $i = 1$ to l.
    (6.1.1) Select n_0 binary solutions from **Pareto** randomly.
    (6.1.2) Generate L_R using (4).
    (6.1.3) Update the ith row of **P** by using (3).
    (6.1.4) Generate rand ∈ [0, 1] a uniform random number.
    (6.1.5) If rand < the predefined mutation probability, update the ith row of **P** using (5).
(6.2) Next i.
(7) Go to step 1.

ALGORITHM 2: MPBIL computational steps.

3.2. Opposite-Based Multiobjective Population-Based Incremental Learning.

MPBIL is governed by the probability update rule (3), which is a variation of Hebb's rule one of the first neural network learning laws [31]. From (3), if L_R approaches one, the probability update rule quickly forgets P_{ij}^{old} and remembers only the most recent b_i. Otherwise, the probability update rule remembers only P_{ij}^{old} and quickly forgets b_i. A small value of learning rate is usually recommended for the conventional PBIL [32]. This L_R setting is not always effective for MPBIL as showed in Bureerat and Srisomporn [30], which had good results from randomly defining L_R value in the range of [0.4, 0.6]. The question arises: how can we choose a proper value of L_R for a general problem, which can accelerate a convergence rate or increase search performance? A new idea to deal with such a problem is motivated from Rahnamayan et al. [33] and Wang et al. [34] as they proposed to enhance EAs using opposition-based learning (OBL). Such a concept could be used for improving population initialization [33] or a hybrid with EA main operators [34, 35]. Both methods can enhance EA search performance. Furthermore, it has been found that this idea was used to maintain diversity in PBIL and increase its performance [36] for single objective design cases.

Opposition is concerned with the relationship between entities, objects, or their abstractions of the same nature, which are completely different in some manner [37]. For example, "remember" and "forget" are descriptions of the brain, but completely different. The opposition-based thinking is a basic element of human thinking; it has been used in many fields. In statistics, the Bernoulli distribution says if the probability of an event is p, then the probability of its contrary is $1 - p$. Also, the pair $(x, -x)$ represent opposite real numbers. It had been proved mathematically and experimentally that utilizing opposition in learning yields a more efficient algorithm than using only pure randomness [38]. For this reason, EAs were enhanced by using OBL. When evaluating a solution **x** for a given problem, usually computing its opposite solution is performed together. This

process will provide another chance for finding a candidate solution, which is closer to the global optimum.

Definition 1 (opposite points, see [34]). Let x be a real number defined on the interval $[a, b]$. The opposite number $\check{x}$ is defined as follows:

$$\check{x} = a + b - x. \tag{6}$$

For $a = 0$ and $b = 1$, we have

$$\check{x} = 1 - x. \tag{7}$$

Similarly, the opposite number in a multidimensional search space can be defined.

Let $\mathbf{x} = \{x_1, \ldots, x_n\}^T$ be a point in an n-dimensional coordinate system with $x_1, \ldots, x_n \in \mathfrak{R}$ and $x_i \in [a_i, b_i]$. The opposite point $\check{\mathbf{x}}$ is defined by its coordinates $\check{x}_1, \ldots, \check{x}_n$ where

$$\check{x}_i = a_i + b_i - x_i; \quad i = 1, \ldots, n. \tag{8}$$

The above opposition-based concept can be embedded into MPBIL, which is governed by a probability update rule. In MPBIL, a probability matrix represents a binary population where L_R is a user-defined value in the range of $[0, 1]$. When evaluating the value of L_R for a probability matrix **P**, it is possible to simultaneously compute its opposite L_R (or OL_R) for the opposite probability matrix (**OP**), which will provide better population variety in a PBIL search process. For OMPBIL, an initial probability matrix **P** and its opposition **OP** are sized $l/2 \times n_b$. A binary population is generated using both matrices. When updating the ith row of the matrix by using (3), the ith row of the opposite probability matrix is also updated using OL_R and the ith row of **P** as

$$OP_{ij}^{\text{new}} = P_{ij}^{\text{old}} \left(1 - OL_R\right) + b_i OL_R, \tag{9}$$

where $OL_R = 1 - L_R$. Two binary subpopulations with N_P/l members based upon to the ith row of the probability matrix and its opposite values obtained from (9) are then created.

<u>Initialization</u> Probability matrix $\mathbf{P} = [\mathbf{0.5}]_{l/2 \times n_b}$, opposite probability matrix $\mathbf{OP} = [\mathbf{0.5}]_{l/2 \times n_b}$, external Pareto archive **Pareto** = {}.
(1) Generate a binary population $\mathbf{B}$ from $\mathbf{P}$.
(2) Decode the binary population to be $\mathbf{x}_{n \times N_p}$ and find the objective values $\mathbf{f}_{m \times N_p}$.
(3) Update **Pareto** by replacing it with the non-dominated solutions of a union set **Pareto** $\cup$ **x**.
(4) If the number of members in **Pareto** exceeds the predefined archive size N_A, remove some of them by using an archiving
    technique.
(5) If the termination criterion is fulfilled, stop the procedure. Otherwise, go to step 6:
(6) Update $\mathbf{P}$ and create a binary population
(6.1) Set a binary population $\mathbf{B} = \{\}$.
(6.2) For $i = 1$ to $l/2$.
      (6.2.1) Select n_0 binary solutions from **Pareto** randomly.
      (6.2.2) Find L_R using (4).
      (6.2.3) Update the ith row of $\mathbf{P}$ by using (3).
      (6.2.4) Compute $OL_R = 1 - L_R$ and generate the ith row of the opposite probability matrix $\mathbf{OP}$ using (9).
      (6.2.5) Generate rand $\in [0, 1]$ a uniform random number.
      (6.2.6) If rand < the predefined mutation probability, update the ith row of $\mathbf{P}$ and $\mathbf{OP}$ using (5).
      (6.2.7) Generate binary subpopulations $\mathbf{SB}$ and $\mathbf{SBO}$ from the ith row of $\mathbf{P}$ and $\mathbf{OP}$ respectively.
      (6.2.8) Set $\mathbf{B} = \mathbf{B} \cup \mathbf{SB} \cup \mathbf{SBO}$
(6.3) Next i.
(7) Go to step 2.

ALGORITHM 3: Procedure for OMPBIL.

TABLE 2: Four design cases.

Test case number	Number of resolutions	Resolution ID	Number of variables			Total
			Internal components	Skins	Nodal positions	
F1	1: Figure 6(a)	0	48	99	—	**147**
F2	1: Figure 6(a)	0	48	99	20	**167**
F3	3: Figures 6(a)–6(c)	1	48	99		**148**
F4	3: Figures 6(a)–6(c)	1	48	99	20	**168**

Other computational steps are similar to MPBIL. The outline of the OMPBIL algorithm, which includes the opposition-based concept, is shown in Algorithm 3.

4. Numerical Experiment

The purpose of this research is to find a new internal structure for the morphing wings, which can adjust the aeroelastic characteristics by applying external torque actuation. This can be classified as a twisting morphing wing concept. Firstly, we start with finding the internal structural layout of a morphing aircraft wing three times a problem by employing an optimization process to synthesize the unconventional internal layout of the morphing aircraft wing. Then, we investigate aeroelastic response due to an actuating torque at a particular wing angle of attack to maintain or change the altitude of flight.

4.1. Synthesizing the Internal Structure of a Morphing Aircraft Wing. For the first purpose of this study, an initial design domain was a simple rectangular unswept and untapered wing-box. In order to laterally verify the proposed design approach, four design problems are posed, while OMPBIL and MPBIL are used to solve the design problems three optimization runs for all test problems. The four sets of

design variables or test cases are presented in Table 2. The first problem (F1) uses the partial topology and component sizing variables, while the second problem (F2) uses the same set of design variables of F1 with nodal positions design variables being added. The third (F3) uses multigrid or multiple sets of ground segments with topology and sizing design variables, while the fourth (F4) design problem is similar to F3 with shape design variables being added. The design variables sets for F3 and F4 have one additional variable for indicating a set of ground elements. According to Algorithm 1, the numbers of segments and nodal positions for the three sets of ground elements are set as $N_{11} = 147$, $N_{12} = 20$, $N_{21} = 123$, $N_{22} = 16$, $N_{31} = 82$, and $N_{32} = 8$. All design problems have design objectives and constraints as presented in (2). The root attachments of all wing models have a constant thickness of 0.005 m. The structures are modeled by using four-node quadrilateral shell elements.

The aeroelastic constraints require that the flutter and divergence speeds must be greater or equal to 150 m/s to avoid the aircraft structural failure at a desired flight speed range (100 m/s). Buckling and stress constraints are also considered. The void threshold is set to be 0.0005 m.

The parameters used for all wing-box models are given in Table 3. The wing is made of aluminum with Young modulus $E = 70$ GPa, Poisson's ratio $\nu = 0.34$, density $\rho = 2700$ kg/m^3,

(1) Start with the initial interval of the root angle of attack $[\alpha_1, \alpha_3]$. It should be noted that the lift of this interval covers the given weight of an aircraft; otherwise, find a new interval.
(2) Calculate lift, L_{α_1} and L_{α_3} at α_1 and α_3 respectively.
(3) Find $\alpha_2 = (\alpha_1 + \alpha_3)/2$, then calculate L_{α_2}.
(4) If $L_{\alpha_2} - W$ is sufficiently small for accuracy requirement, then the procedure is stopped. Otherwise, the new interval of angle of attack is determined.
(5) If $L_{\alpha_2} < W$, the new interval is $\alpha_1 = \alpha_2$. Set $L_{\alpha_1} = L_{\alpha_2}$ and go to step 3.
(6) Otherwise, the new interval is $\alpha_3 = \alpha_2$. Set $L_{\alpha_3} = L_{\alpha_2}$ and go to step 3.

ALGORITHM 4: Aircraft trimming.

TABLE 3: Wing-box parameters.

Number	Parameter	Value
1	Semispan length, L (m)	1.5
2	Root chord length, RC (m)	0.6
3	Tip chord length, TC (m)	0.6
4	Sweep angle, Λ (degree)	0°

and yield stress $\sigma_{yt} = 100$ MPa. Aerodynamic analysis for static and dynamic aeroelasticity is carried out by using the vortex ring panel method [7–10]. The wing lifting surface has 8×5 panels, while its wake has 8×30 panels.

Flutter analysis is computed through the discrete-time aeroelastic model using a reduced-order aerodynamic model with static correction [39]. Buckling factor and stress are computed considering both aerodynamic force and actuator torque. The first load (lift force) is computed at a wing angle of attack 5°, while the free stream velocity is 150 m/s. The actuator torques applied to twist a wing to increase or decrease wing lift is ±60 N-m (positive for clockwise torque). The forces are applied at the first rib as shown in Figure 5. All design constraints are computed where the prestressed effects due to external actuation are taken into account. For more details of each objective and constraint, see our previous studies [8–10].

The multidisciplinary optimization procedure is the combination of a multiobjective evolutionary algorithm, aerodynamic analysis, aeroelastic analysis, and finite element analysis. The vertex ring method for aerodynamic analysis is coded in MATLAB, while ANSYS software is used for FEA. Both optimization algorithms (MPBIL and OMPBIL) are also coded using the MATLAB language. Structural analysis, aerodynamic analysis, and an optimization algorithm are connected using MATLAB.

For performance assessment of the propose method, the conventional MPBIL and the new OMPBIL are employed to solve all of the design problems where the population size is set to be 50, the total number of generations is set to be 400, the external Pareto archive size is 50, and the number of binary bits for each design variable is 4. The learning rate (L_R) is generated randomly in the interval [0.4, 0.6]. The mutation probability and mutation shift are 0.1 and 0.2, respectively. These parameters settings were successfully used with MPBIL in Bureerat and Sriworamas [29], Bureerat and Srisomporn [30], Kunakote and Bureerat [11], and Sleesongsom and

Bureerat [8, 9]. Design constraints are handled by means of the nondominated sorting scheme [40]. The procedure is terminated when reaching the maximum generation number.

4.2. Aircraft Trimming. For this study, the aeroelastic characteristics and mechanical parameters need to be studied in steady level flight to ensure that the new internal structure can be used as a morphing aircraft wing. Normally, motion of an aircraft depends upon forces acting on an aircraft such as lift (L), drag (D), thrust (T), and weight (W). Both lift and drag can be found by aerodynamic analysis in which lift is perpendicular, while drag is parallel to the free stream direction. Lift is created by the wing to carry the weight of the aircraft whilst drag is a resistance of the aircraft motion, which is against the thrust force produced by an engine or propeller. At a steady level flight, with thrust vector aligned with the direction of flight, the static equilibrium can be expressed as follows:

$$L = W. \tag{10}$$

Equation (10) shows lift is balanced by weight in steady level flight. This equation has been used to study how much the angle of attack for an aircraft is trimmed to maintain a steady level flight if an aircraft is disturbed by changes in the structural stiffness with moving of rotating spars [9]. For our work, (10) is used to find the angle of attack to maintain the aircraft at steady level flight. At this angle of attack, if an aircraft is disturbed by input torque (control input) to change the aircraft to other steady level flight, this implies that the new internal structure can be used as the morphing aircraft wing. To find an angle of attack at a flight speed and weight of aircraft, the bisection method is employed in Algorithm 4.

The searching procedure for a required angle of attack to maintain the aircraft in steady level flight is terminated when $L_{\alpha 2}$ is equal to the weight of aircraft (Equation (10)) at the root of the angle of attack α_2.

5. Design Results

5.1. Unconventional Structures of a Morphing Aircraft Wing. Nondominated fronts and the search history (front hypervolume versus generation numbers) of all design problems obtained from the best runs of both MPBIL and OMPBIL are shown in Figures 8 and 9, respectively. In Figure 8, approximate surfaces are used to represent the Pareto fronts as it is difficult to display a 3D front. The search history shows

TABLE 4: Performance comparison based on hypervolume (HV).

MOEAS	Number of runs	Test problems			
		F1	F2	F3	F4
MPBIL	1	13.0720	15.4056	17.5563	20.4751
	2	11.8899	12.4865	12.7354	15.8351
	3	13.5215	10.8927	8.7126	6.6081
	Average	**12.8278**	**12.9283**	**13.0014**	**14.3061**
OMPBIL	1	13.8675	16.3227	18.9464	24.2921
	2	14.6692	13.7767	17.6144	18.2879
	3	14.8824	13.3155	22.8222	26.7169
	Average	**14.4730**	**14.4716**	**19.7943**	**23.0990**
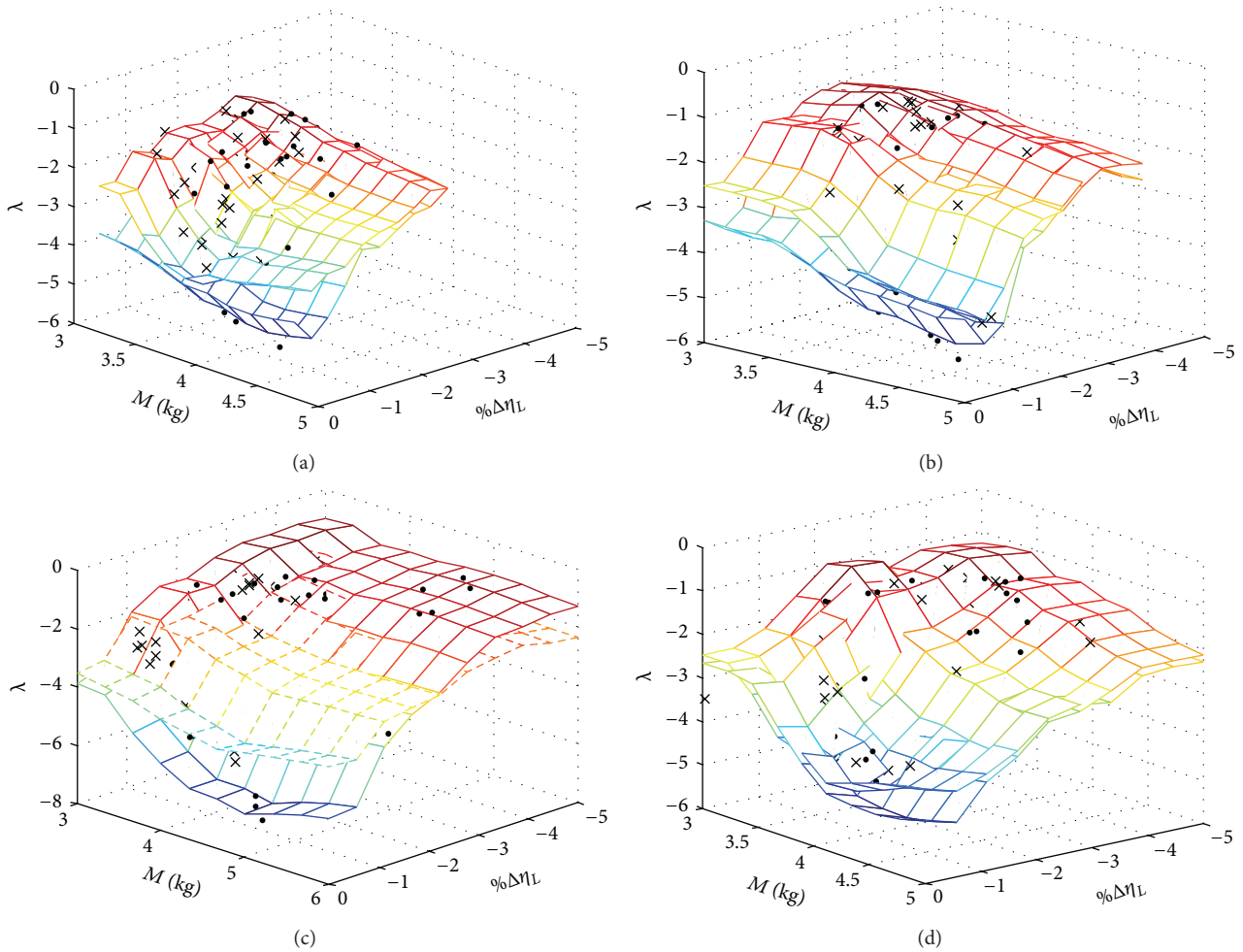
FIGURE 8: Nondominated front of MPBIL (×) and OMPBIL (•) best runs for (a) F1, (b) F2, (c) F3, and (d) F4.

that OMPBIL outperforms MPBIL before the halfway point of searching except for the case of F2 where the two methods interchangeably outperform each other and OMPBIL won at the end. The fronts obtained using OMPBIL are slightly changed for the last 10 iterations. Therefore, we assume that the optimizer converge to the global optimum. The comparative Pareto fronts are somewhat difficult to observe. As a result, a quantitative study, the use of a hypervolume indicator, is conducted to compare the quality of the Pareto fronts. This indicator determines an area (for two objectives) or volume (for more than two objectives) covered by a particular Pareto front with respect to a predefined reference point [41]. The hypervolumes of the fronts of all test problems for all optimization runs are given in Table 4 where the reference point for computing the hypervolumes is set to be $\{0, 6.0 \text{ kg}, -1\}^T$.

This table reveals that the fronts for all case studies obtained from using OMPBIL are better (higher average
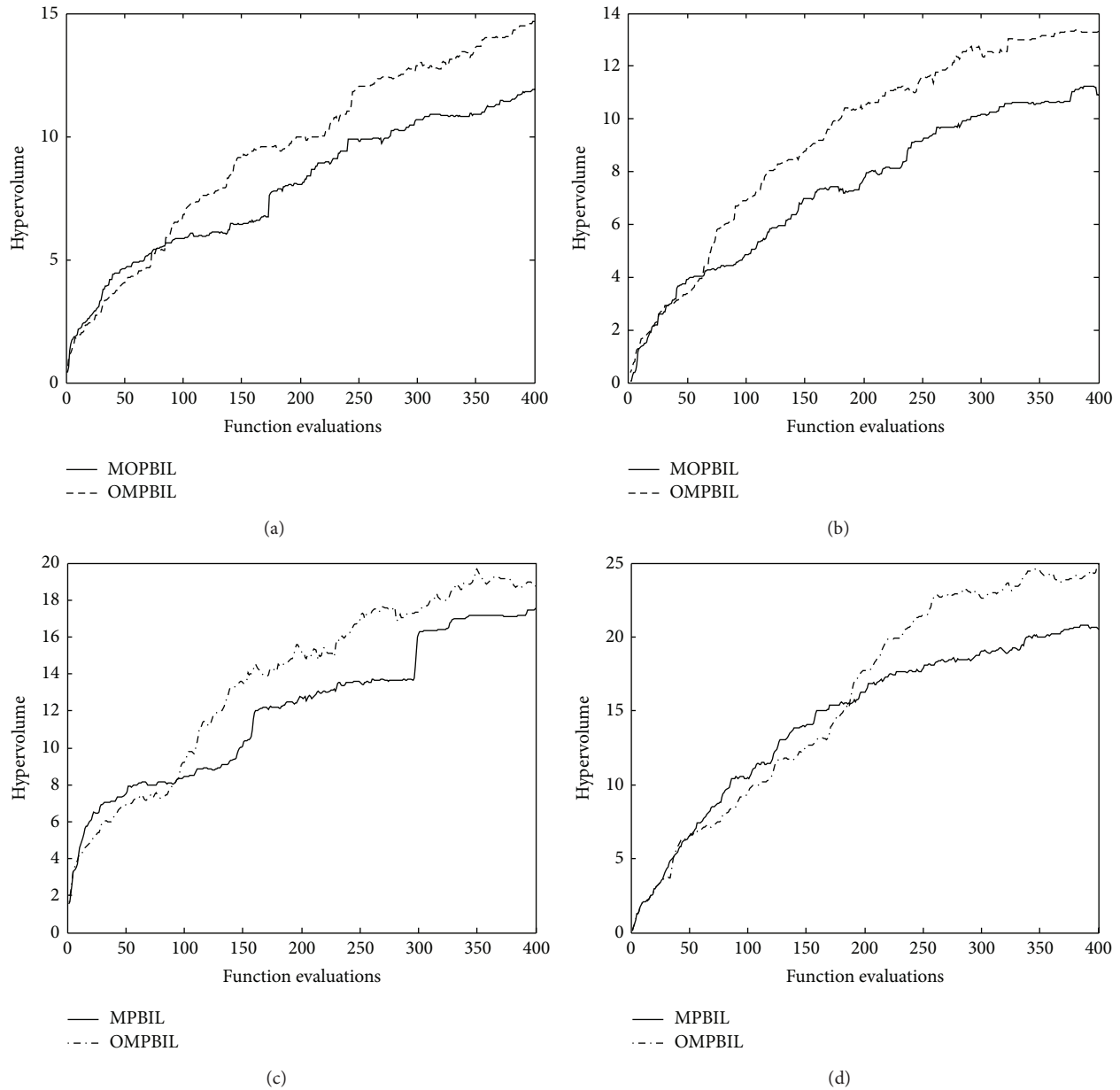
(a)

(b)

(c)

(d)

Figure 9: Search histories of MPBIL and OMPBIL best run for (a) F1, (b) F2, (c) F, and (d) F4.

hypervolume) than those obtained from using a conventional MPBIL. The simultaneous partial topology, shape, and sizing design approach gives better results than the approach without nodal position design variables as the average hypervolume of F4 is higher than F3 and that of F2 is higher than F1. In addition, the results from using a multigrid approach are far superior to those from using single resolution of ground segments (F1 versus F3 and F2 versus F4).

Some unconventional internal structures for the third and fourth design problems (first run) are selected using an even Pareto filter technique [42]. The selected structures of F3 with their aerodynamic (a.a.) and elastic axes (e.a.) are shown in Figure 10 whereas their lower and upper skin

thickness distributions are given in Figure 11. The elastic axis can be thought of as a beam representing a wing structure. When the wing is subject to torsion, it will rotate with respect to this axis. For all wings in Figure 10, the elastic axis is backward behind the aerodynamic axis, which means the wing under aerodynamic loads will have an increased angle of attack and consequently higher value of lift effectiveness. Moreover, some of the wings have forward swept elastic axes, which means they behave like a forward swept wing (usually have high lift effectiveness) although their wing shape are unswept. The structures have various internal layouts and skin thickness distributions. The optimum structures are mostly from the ground segments with medium (Figure 6(b)) and low (Figure 6(c)) resolutions. The use of high ground
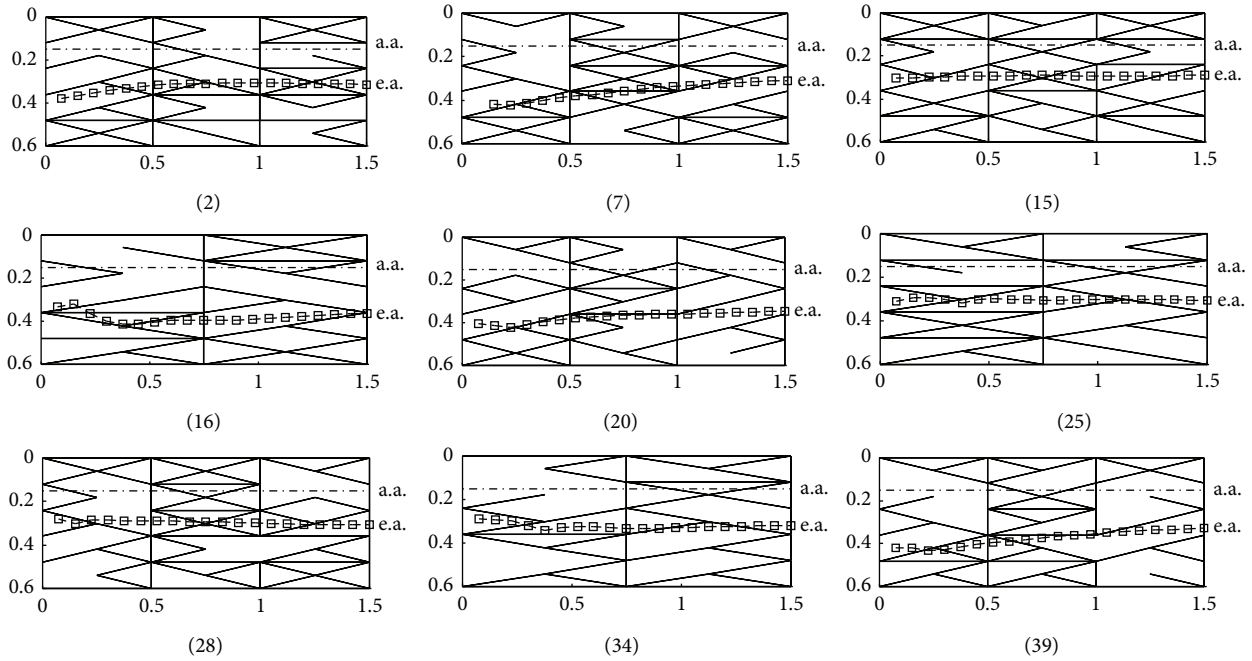
FIGURE 10: Internal structural layouts of the selected wing-boxes in Figure 8(c).
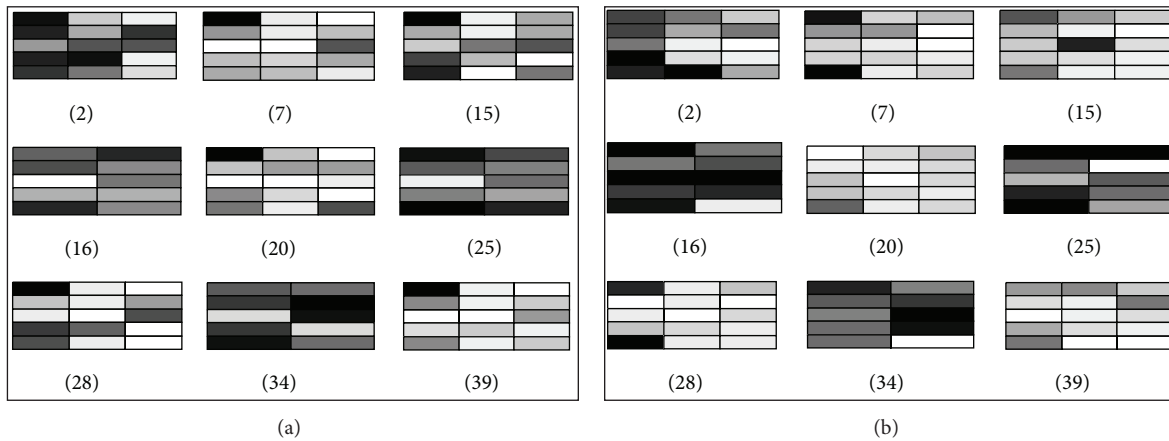

FIGURE 11: (a) Upper and (b) lower skin thickness distribution of the selected wing-boxes in Figure 10 (skin thickness: □ = 0.0001 m, ■ = 0.001 m).

element resolution (Figure 6(a)) is not a good choice for F1 and F2. However, in practice, a designer will never know, which resolution is most suitable; therefore, employing a multigrid approach is advantageous.

Figure 12 shows the selected wing structures of F4 where their skin thickness distributions are illustrated in Figure 13. Similarly to the case of F3 in Figure 10, all the wings in Figure 12 have their elastic axes backward behind their aerodynamic axes, while some of them behave like a forward swept wing. It is shown that all the structures are from the low ground element resolution. This shows that when adding nodal positions to the design problems the proper ground elements resolution is different from the F3 case. The use of a multigrid approach automatically helps finding the suitable resolution during the optimization process.

5.2. Results of Aircraft Trimming. To study the performance of the new morphing aircraft wing, a steady level flight at a flight speed of 100 m/s is defined and the mass of the whole aircraft is set to weigh 25 kg for simplicity. The wing-box (4) illustrated in Figure 12 is selected for this study. From the iterative procedure, it is found that an appropriate angle of attack is 5°. Next, the aeroelastic characteristics and mechanical parameters need to be examined at this angle of attack when an input torque is applied. An input torque in the range of [−60, 60] N-m is applied to the wing at the first wing rib by means of couple as shown in Figure 5. The aeroelastic characteristics and mechanical parameters of the unconventional wing due to the various input torque values are given in Table 5. The relationship between the input torques and the aeroelastic parameters of the wing is illustrated in
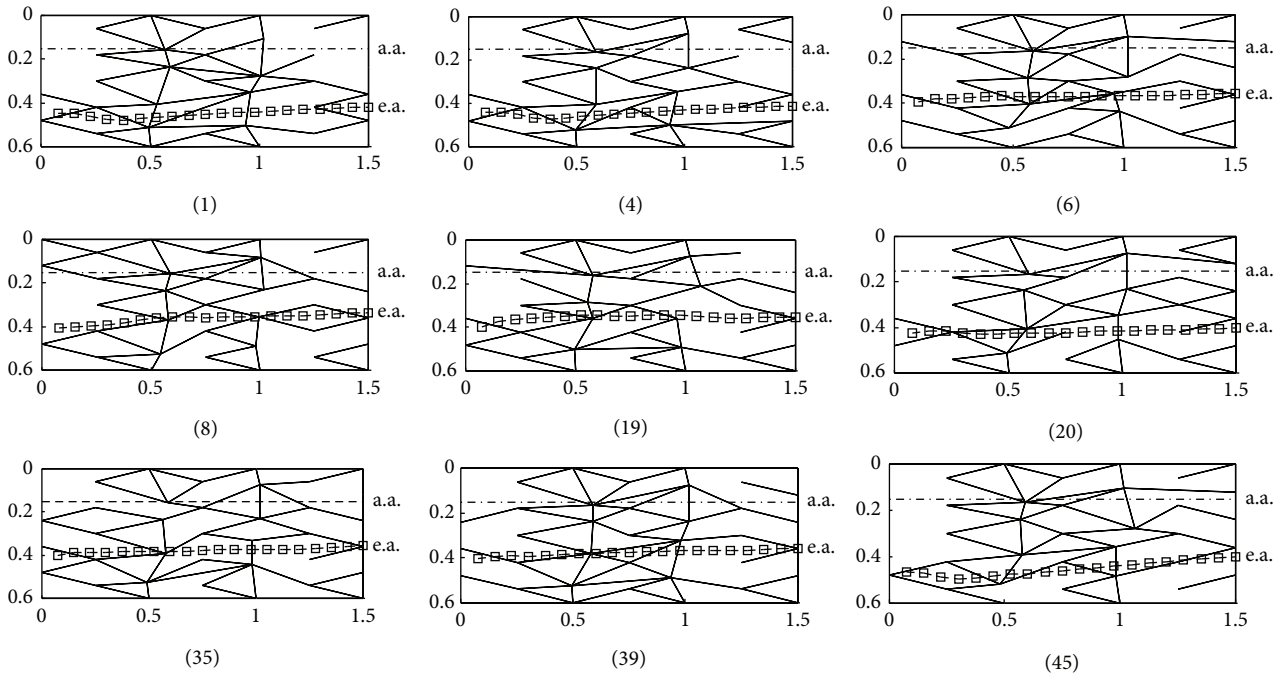
Figure 12: Internal structural layouts of the selected wing-boxes in Figure 8(d).
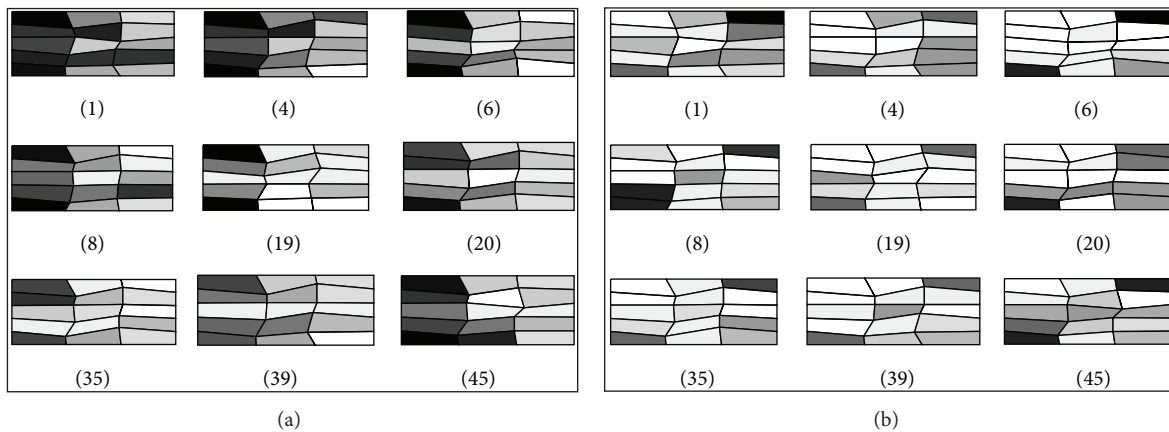

Figure 13: (a) Upper and (b) lower skin thickness distribution of the selected wing-boxes in Figure 12 (skin thickness: □ = 0.0001 m, ■ = 0.001 m).

Figure 14, while the buckling factor and the maximum von Misses stress on the wings are displayed in Figure 15. The lift effectiveness of the unconventional wing ranges between 0.967 and 1.045. This shows that the new unconventional wing can change the lift effectiveness over a wider range compared to the conventional two-spar wing in Sleesongsom et al. [9]. The lift effectiveness of the unconventional wings, in cases of unloaded conditions, is 1.006. When increasing the input torque in the positive direction, the lift effectiveness of the wing is increased whereas it is decreased with negative torques. The divergence speed of the unconventional wing has maximum values when no external forces are applied and it is slightly reduced as the torque magnitudes become larger. The flutter speed of the wing has a similar trend to the

divergence speed as its value decreases with increasing input torque. Buckling and stress on the wing are varied in such a way that the higher external force magnitude will cause more damage to the structure. The buckling factor is reduced, while the maximum stress is increased.

6. Conclusions and Discussion

Two significant contributions of this paper are the improved version of multiobjective PBIL based on an opposition-based approach and the use of a multigrid ground element strategy to synthesize a wing structure. Firstly, the multigrid design approach for optimization of an unconventional morphing wing is proposed. The new design strategy is the

TABLE 5: Aeroelastic characteristics and mechanical parameters due to various input torques of wing-box (4) in **Figure 14** at trimming angle of attack = 5°.

Torque (N-m)	Lift eff.	Flutter speed (m/s)	Divergence speed (m/s)	Buckling factor	Max stress (σ_y = 100 MPa)	Mass (kg)
60	1.045	245.549	151.797	2.297	96.868	
45	1.036	259.348	155.042	2.924	74.624	
30	1.026	269.558	157.150	4.014	52.380	
15	1.016	273.997	158.471	6.347	30.155	
0	1.006	279.534	159.053	13.464	7.998	4.252
−15	0.996	278.303	158.987	10.176	14.389	
−30	0.987	272.400	158.280	4.739	36.598	
−45	0.977	266.160	156.775	3.070	58.842	
−60	0.967	257.898	153.326	2.267	81.085	

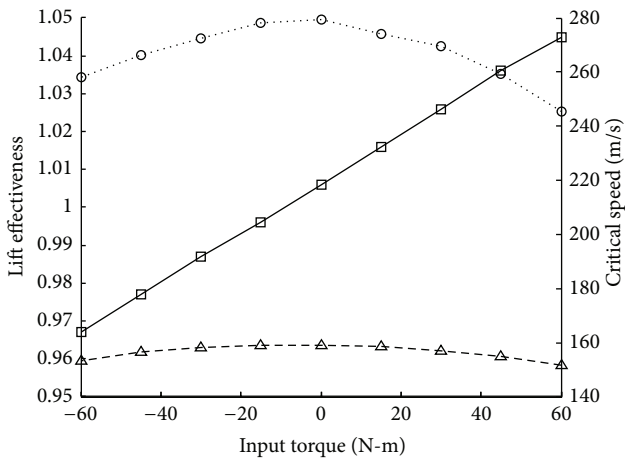FIGURE 14: Input torque versus aeroelastic parameters: □ = lift effectiveness, △ = divergence, and ○ = flutter from Table 3.
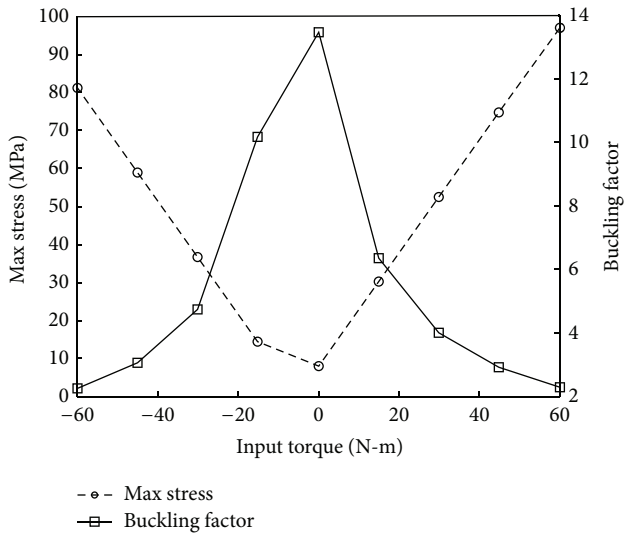

FIGURE 15: Input torque versus stress and buckling factor from Table 5.

procedure for simultaneous partial topology, sizing, and/or shape optimization, which uses multiple ground segment resolutions. This multigrid approach is more efficient than using single-resolution ground elements in the sense that the suitable grid resolution is automatically detected and used in one optimization run. The simultaneous topology, shape, and sizing design gives better results than design without shape design variables. The approximate Pareto front gives a variety of morphing wing internal layouts and skin thickness distributions. Rather than guessing a resolution of ground elements as in the case of using one resolution, an optimizer will automatically find the proper resolutions for design problems with the multigrid approach. The proposed design approach can be extended for practical aircraft design.

Secondly, opposite-based multiobjective population-based incremental learning is proposed to be compared with the original multiobjective population-based incremental learning. The multiobjective design problems of morphing wing structure with single-grid and multigrid design variables are then posed and tackled by both OMPBIL and MPBIL. The optimum results show that the proposed optimizer is superior to the traditional MPBIL. In addition, the results show that using OMPBIL in combination with a multigrid design approach is the best design strategy. The proposed concept is said to be acceptable as an alternative aircraft morphing wing design.

The proposed ideas can be extended real aircraft wing and then realized and used in some type of aerial vehicles. For a small unmanned aerial vehicle, the obtained complicated wing structures can be manufactured by means of three-dimensional printing such as the Southampton University laser sintered aircraft [43, 44]. Our future work is to apply this design concept to a real wing, which is made of high-grade aluminum or carbon fiber composite. That means higher lift effectiveness and structural strength can be expected. Furthermore, rather than generating simple ground elements as used in this paper, using chaos generation [45, 46] or complex networks [47, 48] for generating unconventional wing internal topologies can be a potential means to enhance the wing performance. Moreover, it is also possible to use chaos for improving EA search performance.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

References

[1] A. Y. N. Sofla, S. A. Meguid, K. T. Tan, and W. K. Yeo, "Shape morphing of aircraft wing: status and challenges," *Materials and Design*, vol. 31, no. 3, pp. 1284–1292, 2010.

[2] L. Saggere and S. Kota, "Static shape control of smart structures using compliant mechanisms," *AIAA journal*, vol. 37, no. 5, pp. 572–578, 1999.

[3] K. Maute and G. W. Reich, "Integrated multidisciplinary topology optimization approach to adaptive wing design," *Journal of Aircraft*, vol. 43, no. 1, pp. 253–263, 2006.

[4] S. Liu, W. Ge, and S. Li, "Optimal design of compliant trailing edge for shape-changing," *Chinese Journal of Aeronautics*, vol. 21, no. 2, pp. 187–192, 2008.

[5] B. Stanford and P. Ifju, "Aeroelastic topology optimization of membrane structures for micro air vehicles," *Structural and Multidisciplinary Optimization*, vol. 38, no. 3, pp. 301–316, 2009.

[6] R. Kittipichai and J. E. Cooper, "Optimization of UAVs with adaptive internal structures," in *Proceedings of the 5th ASMO Conference*, Stratford-upon-Avon, UK, 2004.

[7] U. Intachub and S. Bureerat, "Aeroelastic analysis of an aircraft wing structure—the use of a discrete-time model," in *Proceedings of the 20th Conference of Mechanical Engineering Network of Thailand*, Nakhon Ratchasima, Thailand, 2006.

[8] S. Sleesongsom and S. Bureerat, "New conceptual design of aeroelastic wing structures by multi-objective optimization," *Engineering Optimization*, vol. 45, no. 1, pp. 107–122, 2013.

[9] S. Sleesongsom, S. Bureerat, and K. Tai, "Aircraft morphing wing design by using partial topology optimization," *Structural and Multidisciplinary Optimization*, vol. 48, no. 6, pp. 1109–1128, 2013.

[10] S. Sleesongsom and S. Bureerat, "Effect of actuating forces on aeroelastic characteristics of a morphing aircraft wing," *Applied Mechanics and Materials*, vol. 52-54, pp. 308–317, 2011.

[11] T. Kunakote and S. Bureerat, "Multi-objective topology optimization using evolutionary algorithms," *Engineering Optimization*, vol. 43, no. 5, pp. 541–557, 2011.

[12] T. Kunakote and S. Bureerat, "Surrogate-assisted multiobjective evolutionary algorithms for structural shape and sizing optimisation," *Mathematical Problems in Engineering*, vol. 2013, Article ID 695172, 13 pages, 2013.

[13] N. Noilublao and S. Bureerat, "Simultaneous topology, shape, and sizing optimisation of plane trusses with adaptive ground finite elements using MOEAs," *Mathematical Problems in Engineering*, vol. 2013, Article ID 838102, 9 pages, 2013.

[14] S. Bureerat and J. Limtragool, "Performance enhancement of evolutionary search for structural topology optimisation," *Finite Elements in Analysis and Design*, vol. 42, no. 6, pp. 547–566, 2006.

[15] S. Bureerat and J. Limtragool, "Structural topology optimisation using simulated annealing with multiresolution design variables," *Finite Elements in Analysis and Design*, vol. 44, no. 12-13, pp. 738–747, 2008.

[16] B. Stanford and P. Beran, "Minimum-mass panels under probabilistic aeroelastic flutter constraints," *Finite Elements in Analysis and Design*, vol. 70-71, pp. 15–26, 2013.

[17] K. Maute and E. Ramm, "Adaptive topology optimization," *Structural Optimization*, vol. 10, no. 2, pp. 100–112, 1995.

[18] C. Noilublao and S. Bureerat, "Topology and sizing optimization of trusses with adaptive ground finite elements using multiobjective PBIL," *Advanced Materials Research*, vol. 308–310, pp. 1116–1121, 2011.

[19] R. Roberti and P. Toth, "Models and algorithms for the asymmetric traveling salesman problem: an experimental comparison," *EURO Journal on Transportation and Logistics*, vol. 1, no. 1-2, pp. 113–133, 2012.

[20] A. Lencus, O. M. Querin, G. P. Steven, and Y. M. Xie, "Aircraft wing design automation with ESO and GESO," *International Journal of Vehicle Design*, vol. 28, no. 4, pp. 356–369, 2002.

[21] G. Nowak and I. Nowak, "Shape design of internal cooling passages within a turbine blade," *Engineering Optimization*, vol. 44, no. 4, pp. 449–466, 2012.

[22] K. Tai and S. Akhtar, "Structural topology optimization using a genetic algorithm with a morphological geometric representation scheme," *Structural and Multidisciplinary Optimization*, vol. 30, no. 2, pp. 113–127, 2005.

[23] J. A. Madeira, H. C. Rodrigues, and H. Pina, "Multiobjective topology optimization of structures using genetic algorithms with chromosome repairing," *Structural and Multidisciplinary Optimization*, vol. 32, no. 1, pp. 31–39, 2006.

[24] P. Tontragunrat and S. Bureerat, "Antioptimisation of trusses using two-level population-based incremental learning," *Journal of Applied Mathematics*, vol. 2013, Article ID 434636, 12 pages, 2013.

[25] J. D. Knowles and D. W. Corne, "Approximating the nondominated front using the Pareto archive evolution strategy," *Evolutionary Computation*, vol. 8, no. 2, pp. 149–172, 2000.

[26] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[27] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: improving the strength Pareto evolutionary algorithm for multi-objective optimization," in *Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems: Proceedings of the EUROGEN2001 Conference, Athens, Greece, September 19–21, 2001*, International Center for Numerical Methods in Engineering, 2002.

[28] M. Reyes-Sierra and C. A. Coello Coello, "Multi-objective particle swarm optimizers: a survey of the state-of-the-art," *International Journal of Computational Intelligence Research*, vol. 2, no. 3, pp. 287–308, 2006.

[29] S. Bureerat and K. Sriworamas, "Population-based incremental learning for multiobjective optimisation," *Advances in Soft Computing*, vol. 39, pp. 223–232, 2007.

[30] S. Bureerat and S. Srisomporn, "Optimum plate-fin heat sinks by using a multi-objective evolutionary algorithm," *Engineering Optimization*, vol. 42, no. 4, pp. 305–323, 2010.

[31] M. T. Hagen, H. B. Demuth, and M. Beale, *Neural Network Design*, PWS Publishing, Boston, Mass, USA, 1996.

[32] B. Yuan and M. Gallagher, "Playing in continuous spaces: some analysis and extension of population-based incremental learning," in *Proceedings of the Congress on Evolutionary Computation (CEC '03)*, pp. 443–450, Piscataway, NJ, USA, December 2003.

[33] S. Rahnamayan, H. R. Tizhoosh, and M. M. A. Salama, "A novel population initialization method for accelerating evolutionary algorithms," *Computers and Mathematics with Applications*, vol. 53, no. 10, pp. 1605–1614, 2007.

[34] H. Wang, Z. Wu, S. Rahnamayan, Y. Liu, and M. Ventresca, "Enhancing particle swarm optimization using generalized opposition-based learning," *Information Sciences*, vol. 181, no. 20, pp. 4699–4714, 2011.

[35] X. Z. Gao, X. Wang, S. J. Ovaska, and K. Zenger, "A hybrid optimization method of harmony search and opposition-based learning," *Engineering Optimization*, vol. 44, no. 8, pp. 895–914, 2012.

[36] M. Ventresca and H. R. Tizhoosh, "A diversity maintaining population-based incremental learning algorithm," *Information Sciences*, vol. 178, no. 21, pp. 4038–4056, 2008.

[37] F. S. Al-Qunaieer, H. R. Tizhoosh, and S. Rahnamayan, "Opposition based computing—a survey," in *Proceedings of the International joint conference on neural networks (IJCNN '10)*, Barcelona, Spain, July 2010.

[38] S. Rahnamayan, H. R. Tizhoosh, and M. M. A. Salama, "Opposition versus randomness in soft computing techniques," *Applied Soft Computing Journal*, vol. 8, no. 2, pp. 906–918, 2008.

[39] K. C. Hall, "Eigenanalysis of unsteady flows about airfoils, cascades, and wings," *AIAA Journal*, vol. 32, no. 12, pp. 2426–2432, 1994.

[40] K. Deb, A. Pratap, and T. Meyarivan, "Constrained test problems for multi-objective evolutionary optimization," in *Evolutionary Multi-Criterion Optimization*, vol. 1993 of *Lecture Notes in Computer Science*, pp. 284–298, Springer, Berlin, 2001.

[41] T. Friedrich, C. Horoba, and F. Neumann, "Multiplicative approximations and the hypervolume indicator," in *Proceedings of the 11th Annual Genetic and Evolutionary Computation Conference (GECCO '09)*, pp. 571–578, ACM, July 2009.

[42] S. Sleesongsom, "Multiobjective optimization with even pareto filter," in *Proceedings of the 4th International Conference on Natural Computation (ICNC '08)*, pp. 92–96, Jinan, China, October 2008.

[43] Southampton University Laser Sintered Aircraft (SULSA), December 2014, http://www.southampton.ac.uk/~decode/index_files/Page804.htm.

[44] B. Stark, B. Stevenson, K. Stow-Parker, and Y. Chen, "Embedded sensors for the health monitoring of 3D printed unmanned aerial systems," in *Proceedings of the International Conference on Unmanned Aircraft Systems (ICUAS '14)*, pp. 175–180, 2014.

[45] J. Lü and G. Chen, "A new chaotic attractor coined," *International Journal of Bifurcation and Chaos in Applied Sciences and Engineering*, vol. 12, no. 3, pp. 659–661, 2002.

[46] C. Shen, S. Yu, J. Lu, and G. Chen, "Designing hyperchaotic systems with any desired number of positive lyapunov exponents via a simple model," *IEEE Transactions on Circuits and Systems*, vol. 61, no. 8, pp. 2380–2389, 2014.

[47] J. Lü and G. Chen, "A time-varying complex dynamical network model and its controlled synchronization criteria," *IEEE Transactions on Automatic Control*, vol. 50, no. 6, pp. 841–846, 2005.

[48] Y. Chen, J. Lü, and Z. Lin, "Consensus of discrete-time multi-agent systems with transmission nonlinearity," *Automatica*, vol. 49, no. 6, pp. 1768–1775, 2013.