ISME

Iranian Journal of Mechanical Engineering
The Iranian Society of Mechanical Engineers
Research Paper

DOI: 10.30506/JMEE.2022.556939.1295
DOR: 20.1001.1.16059727.2022.23.2.4.3

A New Modified Harmony Search Optimization Algorithm for Evaluating Airfoil Shape Parameterization Methods and Aerodynamic Optimization

F. Jalili[*]
M.Sc. Student

S. M. MalekJafarian[†]
Associated Professor

A. Safavinejad[‡]
Associate Professor

H. Masoumi[§]
M.Sc. Student

In this work, a modification has been made to increase the efficiency and convergence of the harmony search algorithm. Then, the capability of this amendment was investigated by applying it to the following aerodynamic problems for the first time. First, the methods of airfoil shape parametrization (Bezier curves, Parsec method, and NACA 4-digit airfoil) were investigated using an inverse optimization design by the present modified harmony search optimization algorithm. Then, inverse and direct optimization of an airfoil were carried out by the modified algorithm. Aerodynamic analysis of the problem was obtained using compressible Reynolds-Averaged Navier-Stokes (RANS) equations along with the Spalart-Allmaras turbulence model. Results showed that the Bezier curves and the Parsec method have higher flexibility than the NACA 4-digit airfoil. The Parsec method was introduced as the best approach, because of fewer control parameters. The inverse optimization results showed that the present airfoil shape optimization set can obtain the target shape with high accuracy. The Direct optimization with a maximum lift to drag ratio target function revealed that the shock waves significantly weaken at the optimum airfoil. Generally, the results obtained verify that using the modified harmony search algorithm together with the Parsec method provides a powerful tool for direct and inverse aerodynamic optimization.

Keywords: Inverse optimization design, Direct optimization, Meta-heuristic optimization.

1 Introduction

Generally, optimization of the geometry of aerodynamic shapes is carried out through three main tools: first, a tool for airfoil shape parameterization and changing shape through control

[*] M.Sc. Student, University of Birjand, Birjand, South Khorasan, 9718776391, Iran
[†] Corresponding Author, Associate Professor, Department of Mechanical Engineering, University of Birjand, Birjand, South Khorasan, 97175/615, Iran, mmjafarian@birjand.ac.ir
[‡] Associate Professor, Department of Mechanical Engineering, University of Birjand, Birjand, South Khorasan, 97175/615, Iran
[§] M.Sc. Student, University of Birjand, Birjand, South Khorasan, 9718776391, Iran

Receive: 2022/06/30 Accepted: 2022/08/07

variables, the second one is a tool for performing aerodynamic calculations, and a third tool is an optimization approach. Aerodynamic measures play the leading role in optimization. However, parameterizing the geometry and the optimization algorithm is very important too. Hence, using inappropriate parametrization methods and optimization algorithms will provide weak results and even may cause convergence to delay [1].
Method of airfoil shape parameterization (the first tool) is an approach through which an airfoil geometry is created. In creating of this airfoil geometry, some parameters must be involved so that airfoil geometry can be changed by changing these parameters. Different methods are used as these tools, including B-Spline functions [2, 3], Bezier Curves [4], Hicks-Henne Bump Functions [1], Parsec method [5, 6], NACA 4-digit airfoil, etc. [7].
In a general classification, optimization algorithms (the third tool), are divided into two major groups, gradient-based methods and non-gradient-based methods (Meta-heuristic Algorithms). Gradient-based methods usually deal with the first or second derivative of the objective function (and limiting functions). Despite the fast convergence of gradient optimization methods, these techniques sometimes encounter problems in solving some issues. These problematic cases include experiencing discontinuous functions, functions with sharp points, etc. Furthermore, these methods try to improve the solution in a neighborhood of the beginning of the solution. If an improper initial point is selected, sometimes the answer obtained will be a local one and will not cover the entire solution space. Due to these shortcomings, researchers have become interested in meta-heuristic methods and have conducted many types of research using them.
Meta-heuristic methods do not need gradient information, and if the value of an objective function (and limiting functions) is known at different points, it will be enough for optimization. These methods that were developed after 1970, are inspired by natural phenomena and, by combining regularity and randomness, try to find the optimum state [8]. They include simulated annealing algorithm, Tabu search, Evolutionary Algorithm, ant colony optimization, genetic algorithm, big bang–big crunch, etc.

1. 1 A review of previous works

Many types of research have been conducted for designing airfoils. Soemarwoto researched airfoils using inverse optimization design with the variational method [9]. Gardner and Selig investigated airfoils using the inverse optimization method with an evolutionary genetic algorithm [10]. Vitturi and Beux studied direct and inverse optimization design using a gradient-based approach [11]. Shahrokhi and Jahangirian used a genetic algorithm to optimize airfoil geometry through a direct optimization method [12]. In another research, Jahangirian and Shahrokhi dealt with airfoil optimization through inverse optimization. In this research, the genetic algorithm is used as an optimization algorithm [5]. Xu and Xia dealt with direct and inverse optimization of airfoils using the continuous adjoint method and Euler equations [13]. Alves et al. conducted airfoil shape optimization using Parsec and particle swarm optimization (PSO) methods [14]. Ebrahimi and jahangirian investigated airfoil geometry optimization using the genetic algorithm by doing some modifications and the Parsec method [15]. Yu et al. investigated the influence of optimization algorithm and initial design on a wing aerodynamic shape optimization [16].
In the research as mentioned earlier, gradient-based and non-gradient-based optimization methods have been investigated. One of the other non-gradient-based optimization methods is the harmony search algorithm introduced by Geem [8]. Fesanghary et al. used a harmony search algorithm in designing optimum tube and shell heat exchangers through sensitivity analysis [17]. Kaveh et al. used an improved harmony search for economic optimization of a composite floor system [18]. Yousefi et al. optimized the plate-fin heat exchanger using an improved harmony search algorithm [19]. Boryczka and Szwarc presented a modification of the Harmony

Search algorithm, adapted to the effective resolution of the asymmetric case of the Traveling Salesman Problem. The efficacy of the proposed approach was measured with benchmarking tests and in a comparative study based on the results obtained with the Nearest Neighbor Algorithm, Greedy Local Search, and Hill Climbing [20]. Yarmohamadi et al. presented an enhanced adaptive global-best harmony search (EAGHS) that hybridizes the concepts of swarm intelligence and conventional HS to solve global continuous optimization problems [21]. Abu Doush and Santos used the roulette wheel and tournament selection in memory consideration and enhanced the harmony search algorithm with a modified β-hill-climbing algorithm [22]. Zhang et al. proposed a reward population-based differential genetic harmony search algorithm to overcome the shortcomings of the harmony search algorithm, such as its slow convergence rate and poor global search ability [23]. Even though problems solved by this algorithm prove its high efficiency, it has never been used for optimization and design of aerodynamic problems.

1. 2 Presenting the current work

The optimization algorithm determines which airfoil geometry should be shaped to obtain an optimal airfoil (for a particular purpose). However, if the tool for airfoil shape parameterization cannot simulate that optimal airfoil geometry, naturally, a good result will not be obtained. Therefore, selecting a good tool for airfoil shape parameterization is necessary, and investigation of this tool is essential.

The other one, for airfoil shape optimization, is an optimization tool. In this work, harmony search Meta-heuristic algorithms have been used. Information from the authors shows that this algorithm is robust and has never been used for solving aerodynamic problems. So, some modifications are introduced to this algorithm to increase its accuracy and convergence. Then, the modified algorithm is used for airfoil shape optimization, and its ability is investigated.

Briefly, considering the above explanations, in the present work, a comprehensive modification is done to the harmony search optimization algorithm, and it is used for investigating the following three items:

1- Evaluation of typical airfoil shape parameterization methods.
2- Inverse optimization design and assessment of airfoil optimizing set.
3- Direct optimization design to increase lift-to-drag ratio.

2 Introducing Request Tools

2.1 Introducing typical airfoil shape parameterization method

2.1.1 Bezier curves

Consider the following set of points called control points or Bezier points.

$$P = \{P_0, P_1, P_2, \dots, P_n\};\ P_i \in IR^d;\ d = 2,3;\ i = 0, \dots, n$$

Then, the equation of the Bezier curve, which includes the above points, is introduced as follows:

$$C(t) = \sum_{i=0}^{n} P_i\, B_i^n(t) \qquad (1)$$

Where $B_i^n(t)$ is the Bernstein polynomials expressed as follows:

$$B_i^n(t) = \binom{n}{i} (1-t)^{n-i}\, t^i = \frac{n!}{i!\,(n-i)!} (1-t)^{n-i}\, t^i;\ i = 0, \dots, n \qquad (2)$$

Where n represents the order of polynomial so that n+1 control points form the polynomial of nth order [24].
To better understand the above equations, their two-dimensional form is reorganized as follows:

$$P_i = \begin{bmatrix} P_{x,i} \\ P_{y,i} \end{bmatrix};\ i = 0, \dots, n \tag{3}$$

$$C(t) = \begin{bmatrix} C_x(t) \\ C_y(t) \end{bmatrix} = \sum_{i=0}^{n} \begin{bmatrix} P_{x,i}\ B_i^n(t) \\ P_{y,i}\ B_i^n(t) \end{bmatrix} \tag{4}$$

Bezier curves can be used for airfoil shape parameterization in different ways. In the present work, 14 control points are used for airfoil shape parameterization, and 4 of them (beginning and end control points) are fixed. Furthermore, the length (x) of the other 10 control points is set, and only their width (y) is used as a controller. For airfoil shape parameterization, the upper and lower curves must be generated independently and then connected. Based on the above explanations, Figure 1) shows an airfoil made by Bezier curves.

2.1.2 Parsec method

The Parsec method is one of the methods of airfoil shape parameterization. In this approach, the airfoil is generated by 11 control parameters and can be controlled by them according to the following equation:

$$Z_{parsec} = \sum_{n=1}^{6} a_n . X^{n-\frac{1}{2}} \tag{5}$$

These 11 control parameters are properly exhibited in Figure 2). They include a radius of the fro nt edge (r_{le}), and the maximum position of lower and upper surfaces (Z_{lo} , X_{lo} , Z_{up} , X_{up}).

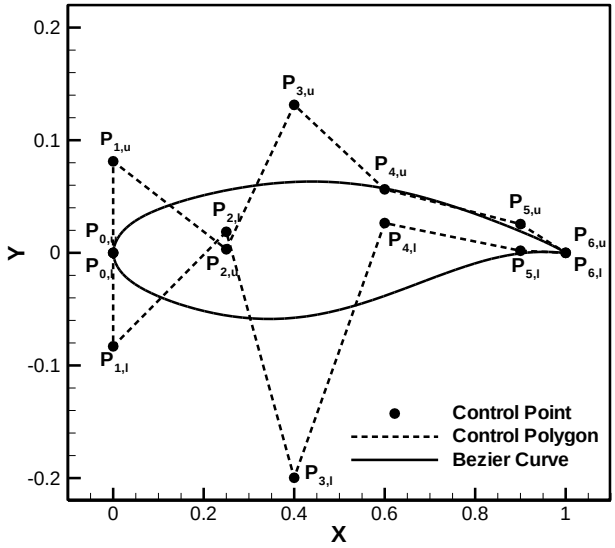

Figure 1 airfoil shape parametrization using Bezier curves

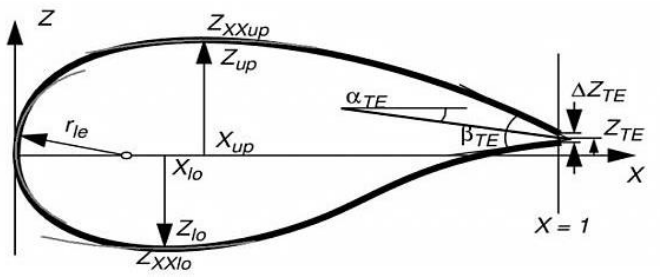

Figure 2 airfoil shape parametrization using the Parsec method

The second derivative at the maximum position of lower and upper surfaces (Z_{xxup} ,Z_{xxlo}), the position of the final edge and its thickness (ΔZ_{TE} ,Z_{TE}), and finally, the direction and width of the final edge. Equation (5) must be used separately once for the upper surface of the airfoil and once for its lower surface. Coefficients are determined by considering the known parameters of airfoil geometry [12]. In some cases, the total number of control parameters can be decreased by considering parameters ΔZ_{TE} and Z_{TE} equal to zero. This can be explained as follows:

1- In most cases, it is interested in investigating optimization under the fixed angle of attack conditions. On the other hand, the existence of Z_{TE} causes the angle of attack to change.

2- Parameter ΔZ_{TE} creates a small hole between the upper and lower surfaces. This hole creates suspicious solutions in some cases [25]. On the other hand, it is shown by experience that acceptable results can be achieved by considering this parameter equal to zero.

Thus, in the present work, the Parsec method with 9 control parameters is used for airfoil shape parameterization.

2.1.3 NACA 4-digit airfoil

One of the other well-known methods for airfoil shape parameterization is to use NACA 4-digit airfoils. In this approach, airfoil geometry is created and controlled only by three control parameters. These three parameters are the maximum camber (m), the maximum camber location (p), and finally, the maximum thickness (τ) [7]. Figure 3) displays these three parameters. Equations describing the upper and lower surfaces of an airfoil in this method are as follows:

$$X_{up} = x - y_{th} \sin\theta \quad (6)$$

$$Y_{up} = y_c + y_{th} \cos\theta \quad (7)$$

$$X_{lo} = x + y_{th} \sin\theta \quad (8)$$

$$Y_{lo} = y_c - y_{th} \cos\theta \quad (9)$$

Where y_{th} is the thickness distribution and y_c is the middle curve line introduced by the following equations:

$$y_{th} = 5\tau \times c(0.2969\sqrt{\frac{x}{c}} - 0.126\frac{x}{c} - 0.3537\left(\frac{x}{c}\right)^2 + 0.2843\left(\frac{x}{c}\right)^3 - 0.1015\left(\frac{x}{c}\right)^4) \quad (10)$$

$$y_c = \begin{cases} \frac{m}{p^2}\left(2p\frac{x}{c} - \left(\frac{x}{c}\right)^2\right) & \text{For } \frac{x}{c} \le p \\ \frac{m}{(1-p)^2}\left(1 - 2p + 2p\frac{x}{c} - \left(\frac{x}{c}\right)^2\right) & \text{For } \frac{x}{c} \ge p \end{cases} \quad (11)$$

In the equation above, θ depends on the slope of the middle curve at each point (Figure 4) and can be obtained by taking the derivative of equation (11):

$$\theta(x) = \tan^{-1}\frac{dy_c(x)}{dx} \quad (12)$$

2.2 Aerodynamic calculations

To obtain aerodynamic efficiency for an airfoil, equations of the flow around it must be solved. This is done through a numerical solver. This tool must have enough accuracy to obtain the pressure and velocity field around the airfoil. The flow around an airfoil can be simulated in different ways. In the present work, the flow is considered viscous, turbulent, and subsonic. Time-dependent compressible Reynolds-Averaged Navier-Stokes (RANS) equations together with turbulence Spalart-Allmaras model [26] are solved using an explicit Runge-Kutta scheme. Conservative and dimensionless forms of these equations in a two-dimensional coordination system are as follows:

$$\frac{\partial W}{\partial t} + \frac{\partial F_i}{\partial x} + \frac{\partial G_i}{\partial y} = \frac{1}{Re_\infty}\left(\frac{\partial F_v}{\partial x} + \frac{\partial G_v}{\partial y}\right) + S \quad (13)$$

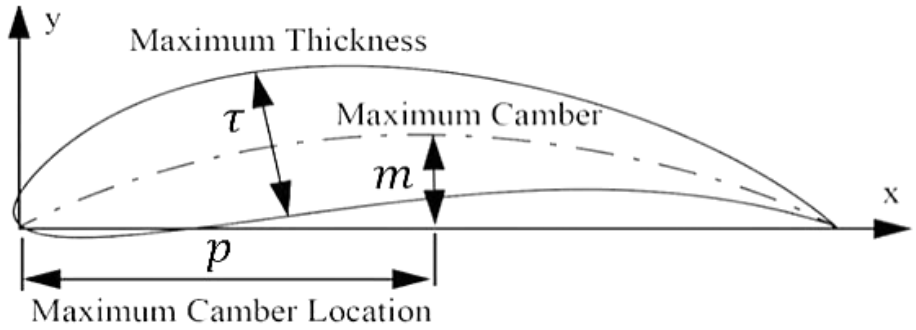

Figure 3 airfoil shape parametrization using NACA 4-digit airfoil

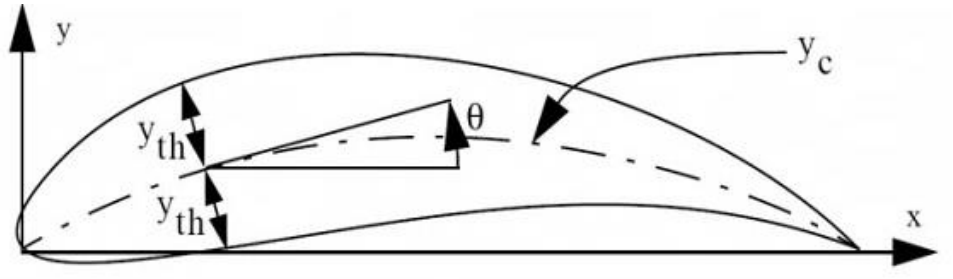

Figure 4 Presentation of the slope of the middle curve

Where W is the vector of the quantities and is defined as follows:

$$W = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ E \\ \rho \hat{\nu} \end{bmatrix} \quad (14)$$

Where F_i and G_i are inviscid flux vectors, F_v and G_v are viscous flux vectors, and S is the source term defined as follows:

$$F_i = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ (E + p)u \\ \rho \hat{\nu} u \end{bmatrix} \quad (15)$$

$$G_i = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ (E + p)v \\ \rho \hat{\nu} v \end{bmatrix} \quad (16)$$

$$F_v = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{yx} \\ \tau_{xx} u + \tau_{xy} v + q_x \\ \frac{\rho}{\sigma}(\nu + \hat{\nu}) \frac{\partial \hat{\nu}}{\partial x} \end{bmatrix} \quad (17)$$

$$G_v = \begin{bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ \tau_{yy} v + \tau_{yx} u + q_y \\ \frac{\rho}{\sigma}(\nu + \hat{\nu}) \frac{\partial \hat{\nu}}{\partial y} \end{bmatrix} \quad (18)$$

$$S = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ D \end{bmatrix} \quad (19)$$

u, v, ρ, p, $\hat{\nu}$, and E represent velocity components in x and y directions, density, static pressure, intermediate chaotic kinematics viscosity variable, and the sum of internal and kinetic energies,

respectively. Furthermore, τ and q represent tension and thermal flux. D is the source term related to the turbulent model. Spalart-Allmaras model is a conventional turbulence model that has shown its capability and efficiency in modeling turbulent flow in aerodynamics. Therefore, in this work, this model is used to simulate turbulent flow.
Equation (13) is solved by a numerical code. In this numerical code, discretization of the equation is done using the finite volume approach (central differential scheme). By converting the governing equations into entirely solvable differential equations and considering a computational cell with volume A (with unit depth) and integrating the equations, we have:

$$\frac{d}{dt}(WA) + Q(w) = SA \tag{20}$$

Where

$$Q(w) = \iint F_i dy + G_i dx - F_v dy - G_v dx \tag{21}$$

Moreover, explicit time integration is performed using a 4-stage Runge-Kutta time-stepping scheme. Principles of the numerical solution were first introduced by Jameson and Schmidt for solving compressible flow problems [27].

$$\begin{aligned} W^0 &= W^n \\ W^1 &= W^0 - \alpha^1 \frac{\Delta t}{A}[Q(W^0) - D(W^0)] \\ W^2 &= W^0 - \alpha^2 \frac{\Delta t}{A}[Q(W^1) - D(W^0)] \\ W^3 &= W^0 - \alpha^3 \frac{\Delta t}{A}[Q(W^2) - D(W^0)] \\ W^4 &= W^0 - \alpha^4 \frac{\Delta t}{A}[Q(W^3) - D(W^0)] \\ W^{n+1} &= W^4 \end{aligned} \tag{22}$$

W^0 is the value of W at the beginning of a time step, and W^m is the value of Wafter passing the m stage of the calculation at that time step.
The scalar dissipation scheme has been used in the numerical code to remove oscillations in the vicinity of shock waves. Since the steady-state solution is needed, convergence accelerating approaches such as local time-stepping and implicit residual averaging is also used in the solution [28]. The numerical solution used in this paper has also been used by many researchers in the recent three decades for solving Navier-Stokes and Euler's equations for a wide range of aerodynamic applications, and its accuracy is proved.

2.3 Optimization algorithm

The algorithm used in the current work is the harmony search (HS) algorithm. This algorithm

is inspired by the musician's thoughts that finding the optimum state in problems is equivalent to finding a desirable harmony in music [8]. In the present work, a modification is done to this algorithm to increase its efficiency. Therefore, first, the harmony search algorithm is explained and then the modified harmony search algorithm is introduced.

2.3.1 Harmony search algorithm

In the HS algorithm, each solution is called harmony and is introduced as an n-dimensional vector. First, an initial population is randomly created and saved in HM*. Then a new Harmony Vector is made based on taking the memory into account, pitch adjustment, and random selection. Finally, the created Harmony Vector is compared with the worst Harmony Vector in the memory $\vec{X}_W$†. If it is better, it replaces by the worst Harmony Vector; by, doing so, HM is updated. This process continues until the stopping criterion is met. Considering the presented explanations, the HS algorithm has three main steps; initializing the HS memory, improvising a new solution, and updating HM. These steps are explained below.

2.3.1.1 Introduction of the problem and algorithm parameters

Generally, any optimization problem can be expressed as a minimization problem as follows:

$$\text{Min } F(\vec{X});\ \vec{X} = (x(1), \dots, x(n));\ x(j) \in [LB(j), UB(j)] \quad (23)$$

In the above equation, $F(\vec{X})$ is the objective function, $\vec{X}$ is the vector of design variables, n is the number of design variables, and UB(j) and LB(j) are the upper and lower limits of the variable, respectively. Parameters of the HS algorithm are harmony memory size (HMS) or the number of existing solution members in the harmonic memory, harmonic memory considering rate (HMCR), pitch adjustment rate (PAR), bandwidth (BW), and severe iterations.

2.3.1.2 Initializing the harmony memory

HM includes HMS solution vectors. If it is assumed that $\vec{X}_i = (x_i(1),\dots,x_i(n))$ is the ith solution vector of HM, then the solution vectors of HM are randomly created using the following equation:

$$X_i(j) = LB(j) + \big(UB(j) - LB(j)\big) \times r : \ j = 1, \dots, n;\ i = 1, \dots, HMS \quad (24)$$

Where r is a random number between 0 and 1. Thus, the HM matrix is filled with HMS solution vectors and is represented as follows:

$$HMS = \begin{bmatrix} x_1(1) & x_1(2) & \cdots & x_1(n-1) & x_1(n) \\ x_2(1) & x_2(2) & \cdots & x_2(n-1) & x_2(n) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{HMS}(1) & x_{HMS}(2) & \cdots & x_{HMS}(n-1) & x_{HMS}(n) \end{bmatrix} \quad (25)$$

---

* Harmony Memory
† Worst Harmony Vector in the HM

2.3.1.3 Improvising a new harmony

A new solution vector, X_{New}, is improved using three rules; considering memory, pitch adjustment, and random selection. To do this, first, a random number (r_1) is selected between 0 and 1. If r_1 is smaller than HMCR, $X_{New}(j)$ is selected from HM (26). Otherwise, $X_{New}(j)$ is obtained randomly (i.e., similar to equation (24) selected between the lower and the upper limits). Choosing from HM is according to equation (26), where a chosen randomly from the set {1,…, HMS}.

$$X_{New}(j) = X_a(j) \ , a \in \{1, \dots, HMS\} \quad (26)$$

If $X_{New}(j)$ is selected from HM, parameter PAR will be introduced. Accordingly, another random number (r_2) is chosen between 0 and 1. If r_2 is smaller than PAR, $X_{New}(j)$ is changed according to the following equation, where r is a random number between 0 and 1.

$$X_{New}(J) = X_{New}(J) + Bw(J) \times r \quad (27)$$

2.3.1.4 Updating the harmonic memory

After creating a new Harmony Vector $\vec{X}_{new}$, HM must be updated. This is done by comparing the objective function of $\vec{X}_{new}$ with the worst member in the memory ($\vec{X}_W$). If the objective function of $\vec{X}_{new}$ is better than that of $\vec{X}_W$, then $\vec{X}_{new}$ replaces by $\vec{X}_W$. So, $\vec{X}_{new}$ enters as a new member. Finally, steps 3 and 4 are repeated until the stopping criterion is met, so the optimum solution is obtained [29, 30].

2.3.2 The modified harmony search algorithm in the present work

Determining the main parameters of the HS algorithm (BW, PAR, and HMCR) has a significant effect on the performance of the algorithm. Hence, in the present work, an effort is made to present an approach for determining these parameters to increase the accuracy and convergence of the algorithm. In this regard, parameter HMCR will have a fixed value, parameter PAR will be dynamically updated, and parameter BW will be introduced as self-modifying. Furthermore, a general modification is presented in how the new solution vector is selected.

2.3.2.1 Modification of the parameter HMCR

In the present modification, only in the first 0.1 of iterations, a random solution is proposed, and the value of parameter HMCR is considered to be 0.95. In the remaining 0.9 iterations, this value is considered to be 1. This means that in the remaining 0.9 iterations, no variable is selected randomly, and only the existing solution vectors in the memory are modified.

2.3.2.2 Modification of parameter PAR

To modify the parameter PAR, maximum and minimum values are considered for it, and the solution is started with its maximum value. Then, whenever one-tenth of the total number of iterations is reached, PAR is updated according to the following equation:

$$\mathrm{PAR} = \mathrm{PAR} - \frac{\mathrm{PAR_{MAX}} - \mathrm{PAR_{MIN}}}{9} \quad (28)$$

PAR_{MAX} is often considered between 0.9 and 1, but PAR_{MIN} will depend on the number of variables in the solution vector. Based on our experience, for two variables problems, 0.9 is a suitable value for PAR_{MIN}, and for eight variables problems, 0.45 is an appropriate value for PAR_{MIN}. The smallest value that PAR_{MIN} can have is equal to 0.1, which is used in problems with a large number of variables.

2.3.2.3 Modification of parameter BW

In the present work, the parameter PAR is introduced as self-modifying. This parameter begins to work with an initial average value (BW_M, which is usually one-tenth of the allowed variations range of each variable). When this parameter is needed, the average value random decreases or increases by one-tenth of its value.
If a new Harmony Vector enters the memory, the amount of improvement in the objective function compared to the worst member is calculated and is saved for the corresponding BW (i.e., increase or decrease in the average value). After AP* iterations, the average value of this parameter is updated by the following equation:

$$\mathrm{BW_M} = \begin{cases} \mathrm{BW_M} + \dfrac{\mathrm{BW_M}}{30} & \text{if } (\mathrm{Improvement_{up}} > \mathrm{Improvement_{low}}) \\ \mathrm{BW_M} - \dfrac{\mathrm{BW_M}}{30} & \text{if } (\mathrm{Improvement_{up}} \leq \mathrm{Improvement_{low}}) \end{cases} \quad (29)$$

In the above equation, $Improvement_{up}$ represents the sum of improvements in the objective function during AP iterations for BWs their values are more significant than BW_M by BW_M /10. Similarly, $Improvement_{low}$ represents the sum of progress in the objective function during AP iterations for BWs that their values are less than BW_M by BW_M /10.

2.3.2.4 Change in selecting a new solution vector

In this modification, it is proposed to ultimately select a solution vector from memory in a random manner. Next, parameter HMCR determines the probability of random selection of each variable in the solution vector. Moreover, parameter PAR will determine whether pitch adjustment needs to be performed on a variable or not.

3. Results

3.1 Evaluation of typical airfoil shape parameterization methods

A suitable method of airfoil shape parameterization is expected to satisfy the following:
1- It needs to have high flexibility.
2- The number of its control parameters must be as low as possible.
3- Independent control parameters must be used.
4- It must present smooth and realistic curves.
5- It needs to be easy to formulate.

---

*Adjusting Pace

Among the abovementioned items, 3, 4, and 5 are satisfied by all the approaches as mentioned above (Bezier curves, Parsec method, and NACA 4-digit airfoil). Accordingly, requirements 1 and 2 play the crucial role in selecting a proper strategy.

Requirement 1, which seems the most important one, is about having high flexibility. This means that the desired method must be able to simulate different geometries. This enables us to search for the optimum geometry in a more extensive search space. If the method of airfoil shape parametrization does not have such a capability, we will not be able to reach a global solution. This is because the low flexibility of the method causes some geometries to be missed out, and there may be a more appropriate geometry among these missed geometries. Therefore, it seems reasonable to pay more attention to this issue because even if the other elements used have the best performance, we will not be able to reach a global solution due to the inability of the method of airfoil shape parameterization.

In this section, the goal is to investigate the flexibility of the three airfoil shape parameterization methods by using an inverse optimization design. Since taking aerodynamic calculations into account increases the time consumed for running the program; therefore, a non-aerodynamic objective function will be used to investigate the flexibility. To perform this, an airfoil is considered as the target airfoil, and the objective function is the difference between the geometries of the airfoil under investigation and the target one, which is intended to be minimized by the algorithm. Those above non-aerodynamic objective function is introduced as follows:

$$\text{Cost function} = \sum_{i=1}^{n} \left| y_i - y_{i,\text{Target}} \right| \qquad (30)$$

Where n is the number of points that form the airfoil, y_i is the lowing y of ith x forming the airfoil under investigation, and $y_{i,TARGET}$ is the y of ith x forming the target airfoil. Therefore, the difference between the width of the airfoil under investigation and that of the target one is calculated and added. Then it is introduced as the objective function. To better understand, Figure (5) is presented the objective function, which is plotted for n=21.

To perform this investigation, 10 different airfoils are considered target airfoils. Therefore, each approach is evaluated 10 times. The approach which has the slightest error will undoubtedly have high flexibility. Among these 10 target airfoils, three are created by NACA 4-digit airfoil, three are randomly generated by the Parsec method, and three are randomly created by Bezier curves. The tenth one is the well-known RAE2822 airfoil. Naturally, each approach is able comprehensively to properly simulate the airfoil that is generated by itself because it has done this before. If the optimization tool is accurate, it will be able to simulate it again. But the capability of the other approaches will depend on their ability.

The modified harmony search algorithm is used to perform this evaluation. The algorithm memory is considered to be 10 in all cases, and the algorithm has started from a completely random airfoil and moves towards reaching the target airfoil. The value of n (related to equation (30)) equals 121. Each experiment is carried out several times, and finally, the best result is reported as the final result.

Table 1) shows the results of this evaluation. The value of the objective function (which corresponds to the error of each approach) is presented for each experiment, separately. Table 2 reports the sum of all errors in 10 experiments for three approaches. Figure 6) to Figure 15) also disclose the results of each of the experiments presented in Table 1). These diagrams exhibit the flexibility of the three methods in reaching the target airfoil. In the present work, the airfoils have become dimensionless by their length. But for the sake of being brief, coordination related to the airfoil is represented by X and Y. Investigating the abovementioned tables and figures demonstrates the NACA 4-digit airfoil does not provide high flexibility. Therefore,

despite having a smaller number of control parameters, this approach is put away.
But the other two approaches (the Parsec and Bezier curves) appear competent in this experiment. Despite the superiority of Bezier curves over the Parsec method, the other condition of the problem is discussed. Because this superiority is not a remarkable one, and both of them have almost the same sum of errors. The other essential requirement is having a small number of control parameters. The Parsec method has 9 control parameters, and the Bezier curves used in this experiment have 10 control parameters. Since the control parameter has a direct effect on convergence, therefore, choosing the Parsec method, proper flexibility can also be achieved as well as a smaller number of control parameters. Correspondingly, the Parsec method has been introduced for airfoil shape parameterization in the present work.
It is worth noting that if, in specific applications, flexibility is of greater importance, the Bezier curves (with a more significant number of control parameters) can be used. This will result in an approach with high flexibility, but convergence will be delayed because of a more significant number of control parameters that must coordinate with each other.
The accuracy and flexibility of these three airfoil shape definition methods clearly show their importance in optimizing aerodynamic problems. Because if the geometry of the airfoil is not obtained with good accuracy; as a result, the flow around the airfoil is not solved correctly by the numerical solver. Therefore, the aerodynamic parameters such as lift and drag coefficients will not be accurately estimated.

Table 1 The results of the evaluation of the flexibility of airfoil shape parameterization methods

Experiment No.	Method		
	NACA 4-digit airfoil	Parsec Method	Bezier Curves
Experiment 1 (target airfoil NACA0012)	1.18E-6	5.46E-3	3.29E-2
Experiment 2 (target airfoil NACA4812)	5.18E-3	1.2E-1	1.09E-1
Experiment 3 (target airfoil NACA2211)	4.72E-3	1.79E-1	5.15E-2
Experiment 4 (target airfoil randomly created by the Parsec method)	6.54E-1	1.88E-4	9.55E-2
Experiment 5 (target airfoil randomly created by the Parsec method)	1.15	3.35E-3	1.11E-1
Experiment 6 (target airfoil randomly created by the Parsec method)	4.62E-1	6.24E-4	9.83E-2
Experiment 7 (target airfoil randomly created by the Bezier curves)	5.44E-1	3.3E-2	3.3E-3
Experiment 8 (target airfoil randomly created by the Bezier curves)	3.3E-1	2.07E-1	8.69E-3
Experiment 9 (target airfoil randomly created by the Bezier curves)	6.55E-1	1.22E-1	3.62E-4
Experiment 10 (target airfoil RAE 2822)	5.22E-1	2.27E-2	3.5E-2

Table 2 The Sum of All Errors in 10 Experiments Related to the Evaluation of the Flexibility of Airfoil Shape Parameterization Methods

	NACA 4-digit airfoil	Parsec Method	Bezier Curves
Sum of errors	4.02	6.94E-1	5.46E-1

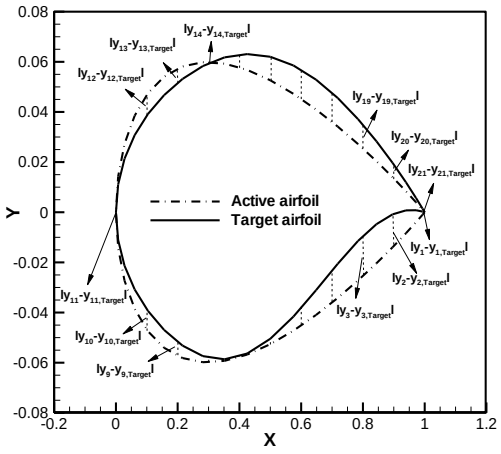

Figure 5 Objective function calculation to investigate the flexibility

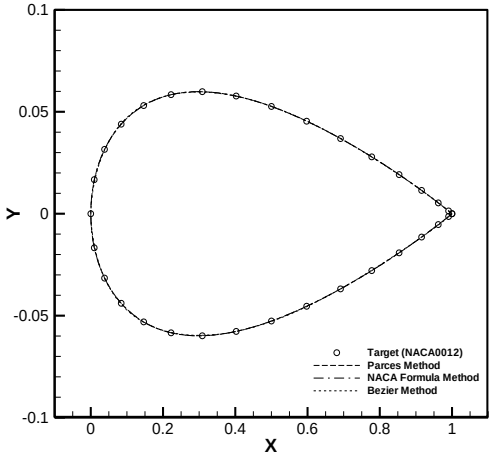

Figure 6 Experiment 1 (target airfoil NACA0012)

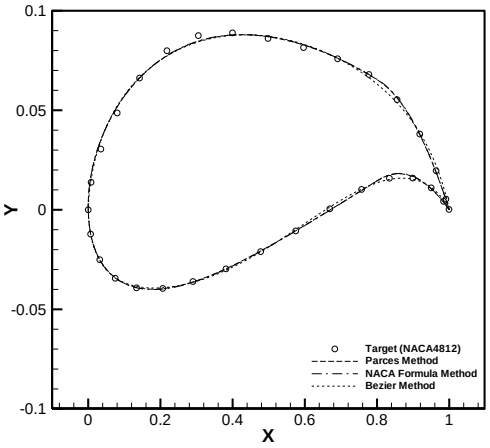

Figure 7 Experiment 2 (target airfoil NACA4812)

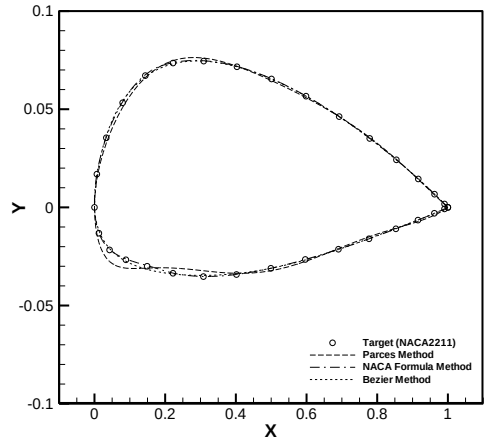

Figure 8 Experiment 3 (target airfoil NACA2211)

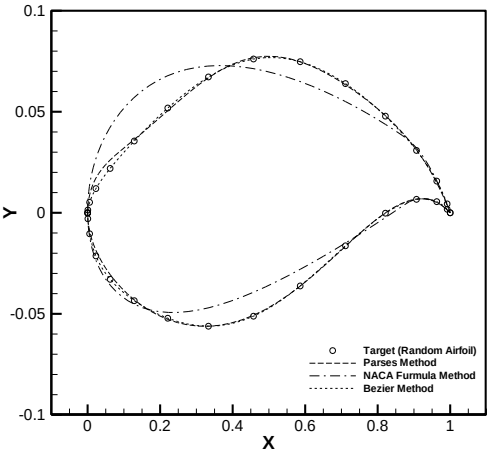

Figure 9 Experiment 4 (target airfoil randomly created by the Parsec method)

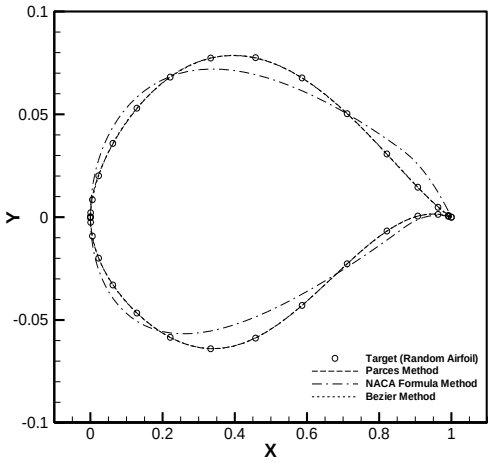

Figure 10 Experiment 5 (target airfoil randomly created by the Parsec method)

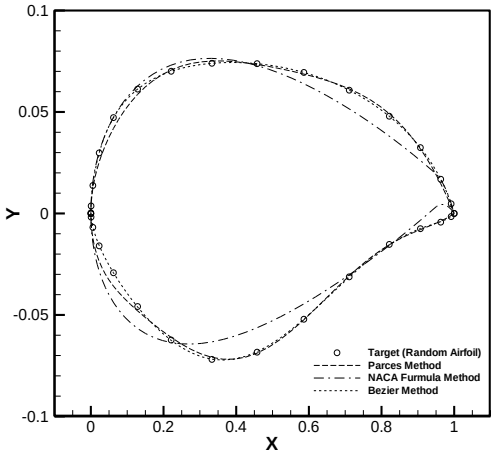

Figure 11 Experiment 6 (target airfoil randomly created by the Parsec method)

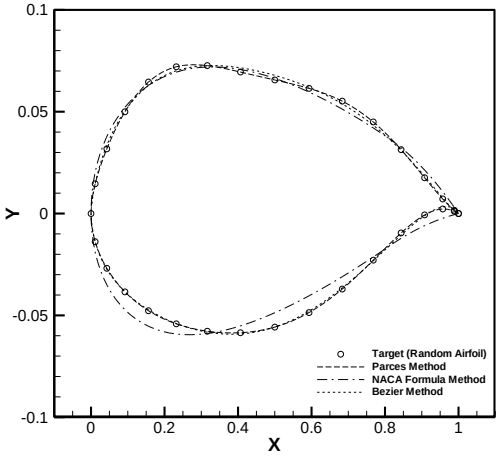

Figure 12 Experiment 7 (target airfoil randomly created by the Bezier curves)

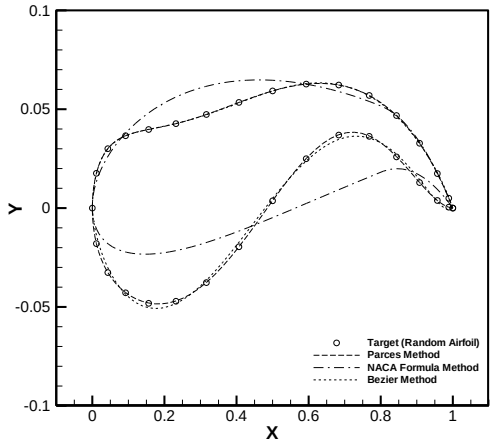

Figure 13 Experiment 8 (target airfoil randomly created by the Bezier curves)

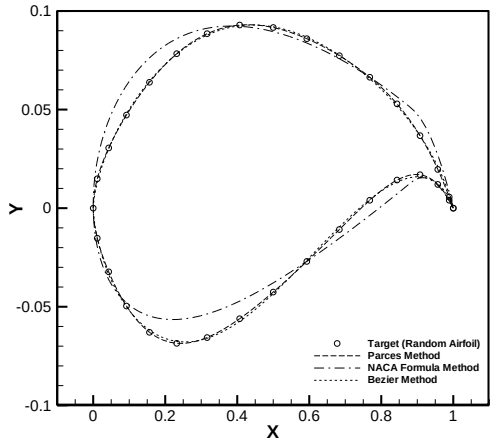

Figure 14 Experiment 9 (target airfoil randomly created by the Bezier curves)

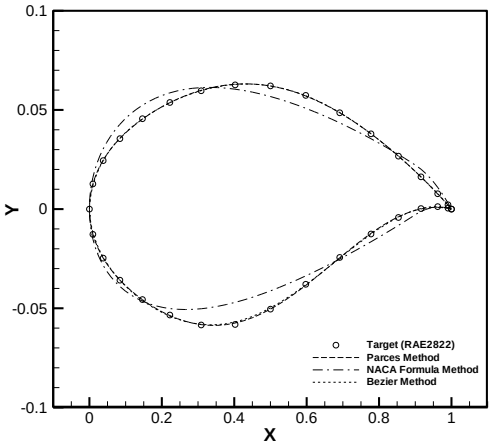

Figure 15 Experiment 10 (target airfoil RAE 2822)

3.2 Aerodynamic inverse optimization design

In this section, an aerodynamic inverse optimization design is performed.
Inverse optimization is of interest in the present work because it can be used to validate the current improved harmony search algorithm.
In inverse design, the target airfoil shape is rebuilt by making the pressure distribution coefficient close to the defined airfoil pressure distribution. This type of design can test the airfoil optimizing set.
By the results of the previous section, the Parsec method will be used. In this section, the memory size of the HS algorithm is considered to be 10. Also, the upper and lower limits of this allowable range are shown in Table (3). To start optimization, one airfoil will be considered to be a NACA0012 airfoil (in the second evaluation RAE2822), and the remaining 9 airfoils will be random, which are generated by applying some changes on NACA0012 airfoil (in the second evaluation of RAE2822). First, the NACA0012 and RAE2822 airfoils are considered the initial and target airfoils, respectively. The goal is to reach the target airfoil by minimizing the surface pressure difference. This is done under the following conditions:

$$\alpha = 2^{\circ}, \mathrm{M}_{\infty} = 0.73, \mathrm{Re}_{\infty} = 6.5\mathrm{E}6$$

An example of mesh around the airfoil (type O) is shown in Fig. (16). Because the viscous flow is considered, the mesh near the body has more density. This makes it possible to simulate the turbulent boundary layer well. Also, the network on the leading edge and trailing edge of the airfoil, which have more gradients than the middle parts of the airfoil, is non-uniform. This work makes a smoother surface with fewer points, and finally, a better mesh is obtained. After the mesh study, the number of mesh 131 x 81 was selected for all the simulations.
Figure (17) and Figure (18) show the result of this experiment as the geometry and pressure coefficient distribution of the initial and final airfoils. As can be seen, this set has been able to start from an initial airfoil and lead it towards the target one with good accuracy. The objective function versus several iterations presented, to investigate the accuracy and convergence of the optimization (Figure 19). The convergence history of the best member of the memory is shown along with an average of the whole members of the memory. According to this figure, it can be seen that the set has been able to decrease the value of the objective function to 2.5E-6. Furthermore, an incomplete convergence is observed. This means that by increasing the number of iterations, it is also likely that the value of the objective function decreases to a value less than the obtained value. However, since the obtained value of the objective function shows the accuracy of the optimization set, this value is considered to be enough (for several iterations).
To demonstrate the independence of the set from the initial and target airfoils, the previous experiment is performed in the inverse direction. This means that the investigation is carried out under the same conditions, but starts from the RAE2822 airfoil and reaches NACA0012 one (target airfoil) by minimizing the objective function. The geometry and surface pressure coefficient distribution of the initial and final airfoils are presented in Figure (20) and Figure (21). The convergence history is shown in Figure (22).
These results emphasize that the initial and target airfoils do not affect the airfoil optimizing set significantly. The second result is the verification of the accuracy of the airfoil optimizing set. This is because the set has been able to decrease the value of the objective function to 4.3E-6 (which is a good accuracy for this problem).
The efficiency of the optimizing airfoil set is evaluated under different flow conditions. To do this, the NACA0012 airfoil is considered the initial airfoil, and the RAE2822 airfoil is considered as target one under the following flow conditions:

$$\alpha = 2.79^{\circ}, \mathrm{M}_{\infty} = 0.734, \mathrm{Re}_{\infty} = 6.5\mathrm{E}6.$$

Figure (23) and Figure (24) exhibit the geometry and surface pressure coefficient distribution of the initial and final airfoils. Figure (25) shows the convergence history of this evaluation. The results prove that the optimization set performs well under different flow conditions because the algorithm has been able to decrease the objective function to 4E-5.

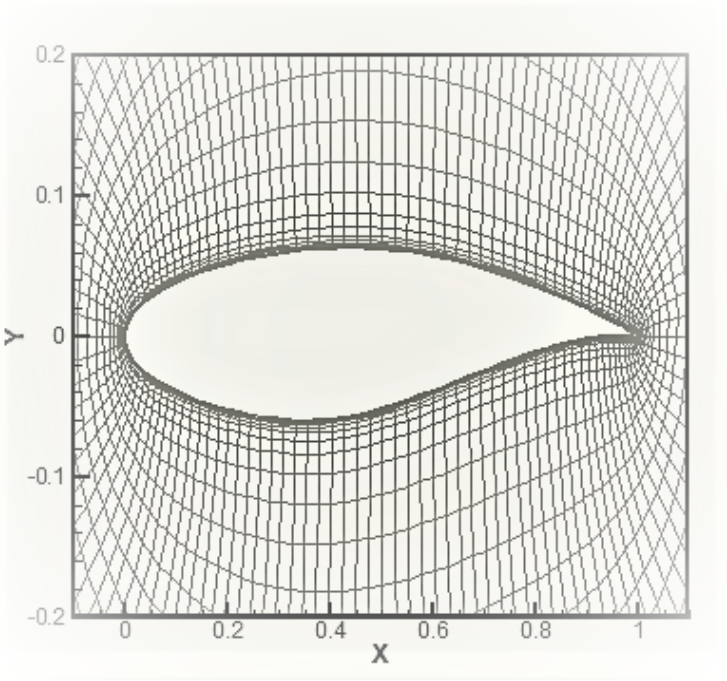

Figure 16 mesh around the RAE2822 airfoil from a close-up view (131 x 81)

Table 3 The allowed range for controlling parameters

	β_{TE}	α_{TE}	Z_{xxlo}	Z_{xxup}	Z_{lo}	X_{lo}	Z_{up}	X_{up}	r_{le}
Upper limit	0.5	0.01	0.3	-0.01	0.06	0.7	0.1	0.8	0.08
Lower limit	0	-0.8	-1	-1/2	-0.04	0.2	0.02	0.1	0.005

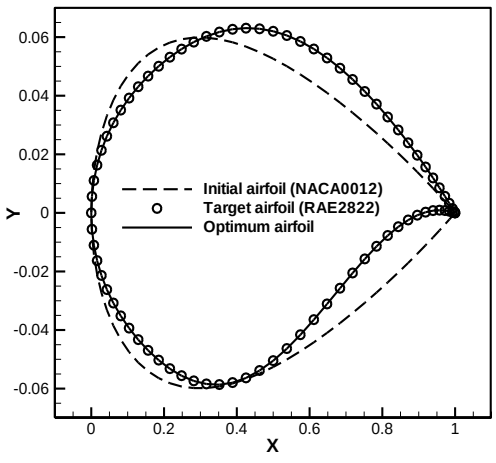

Figure 17 Inverse constructed shape Starting from NACA0012 to RAE2822 ($\alpha = 2^{\circ}, M_{\infty} = 0.73, Re_{\infty} = 6.5E6$)

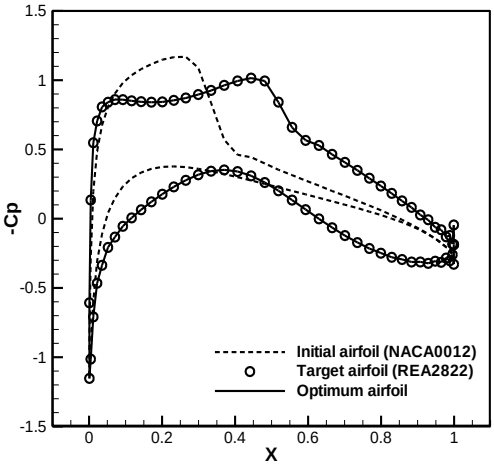

Figure 18 pressure coefficient distribution in inverse optimization Starting from NACA0012 to RAE2822 ($\alpha = 2^{\circ}, M_{\infty} = 0.73, Re_{\infty} = 6.5E6$)

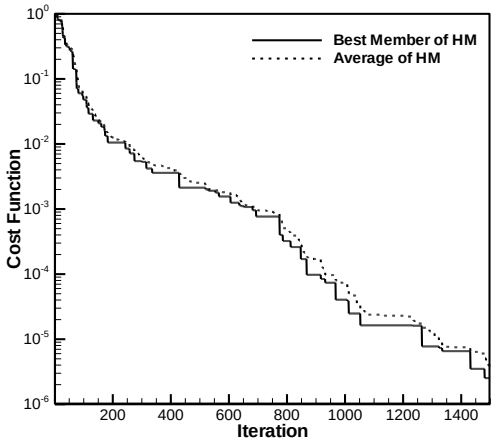

Figure 19 convergence history in inverse optimization Starting from NACA0012 to RAE2822 ($\alpha = 2^{\circ}, M_{\infty} = 0.73, Re_{\infty} = 6.5E6$)

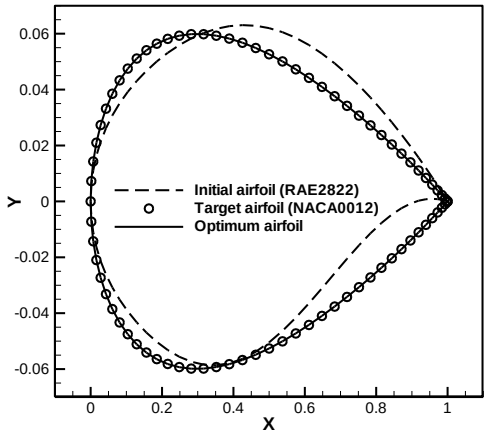

Figure 20 Inverse constructed shape Starting from Rea2822 to NACA0012 ($\alpha = 2^{\circ}, M_{\infty} = 0.73, Re_{\infty} = 6.5E6$)

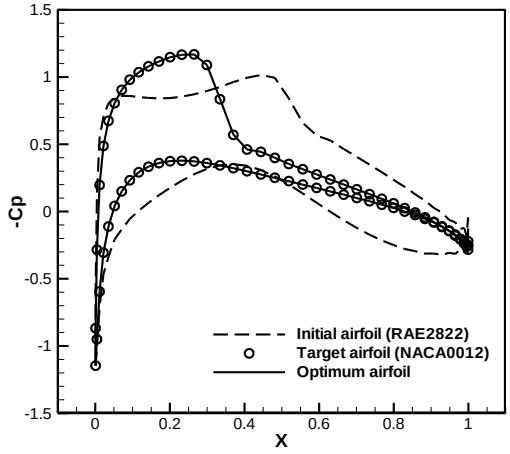

Figure 21 Pressure coefficient distribution in inverse optimization Starting from RAE2822 to NACA0012 ($\alpha = 2^{\circ}, M_{\infty} = 0.73, Re_{\infty} = 6.5E6$)

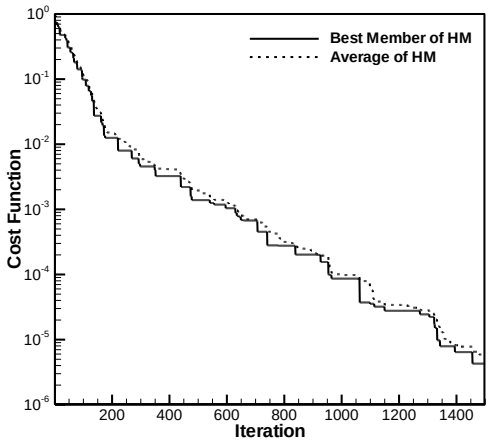

Figure 22 Convergence history in inverse optimization Starting from Rea2822 to NACA0012 ($\alpha = 2^{\circ}, M_{\infty} = 0.73, Re_{\infty} = 6.5E6$)

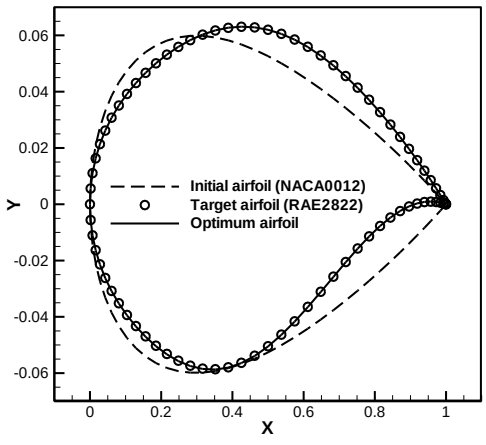

Figure 23 Inverse constructed shape Starting from NACA0012 to RAE2822 ($\alpha = 2.79^{\circ}, M_{\infty} = 0.734, Re_{\infty} = 6.5E6$)

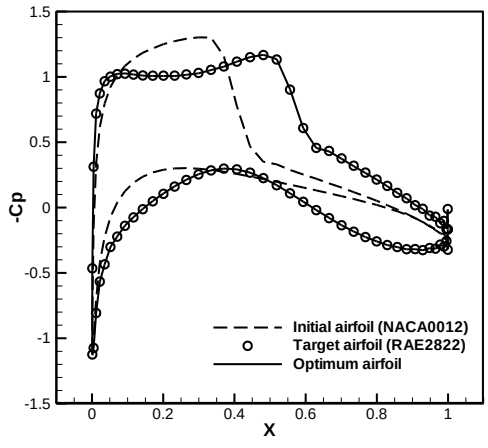

Figure 24 Pressure coefficient distribution in inverse optimization Starting from NACA0012 to Rea2822 ($\alpha = 2.79^{\circ}, M_{\infty} = 0.734, Re_{\infty} = 6.5E6$)

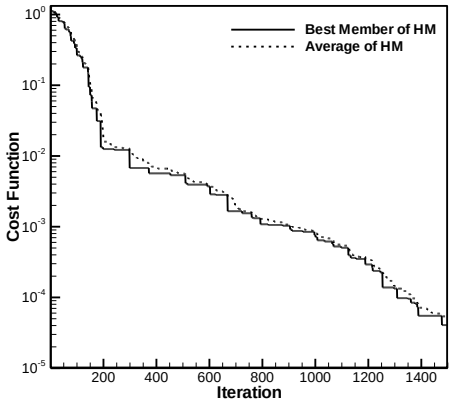

Figure 25 Convergence history in inverse optimization Starting from NACA0012 to RAE2822 ($\alpha = 2.79^{\circ}, \mathrm{M}_{\infty} = 0.734, \mathrm{Re}_{\infty} = 6.5\mathrm{E}6$)

3.3 Direct aerodynamic optimization design aimed at obtaining the maximum lift to drag ratio

By the results of the previous section, the performance of the optimization setting is confirmed. So, in this section, airfoil shape optimization is performed to obtain the maximum lift-to-drag ratio. The viscous and turbulent flow has been considered under the following conditions:

$$\alpha = 2.79^{\circ}, \mathrm{M}_{\infty} = 0.734, \mathrm{Re}_{\infty} = 6.5\mathrm{E}6$$

The Parsec method describes the airfoil geometry, and the memory size of the algorithm is equal to 10. Considering the NACA0012 airfoil as the initial airfoil, the initial memory of the algorithm is filled with one NACA0012 and another 9 random airfoils.

Next, the algorithm searches for the optimum airfoil by changing the design parameters and creating new airfoils. To develop reasonable geometries, design parameters are determined in an allowable range. The upper and lower limits of the range are shown in Table (3).

Thin airfoils intrinsically have a smaller drag value. Whereas, thin airfoils are not a very desirable feature in practice. Thus, a thickness limitation is applied to the algorithm to avoid investigating thinner airfoils. If this thickness limitation is not used, the geometries will be pulled towards thin airfoils. The allowed range is considered 11 to 12.2 percent of the airfoil length, at the present work.

Figure (26) shows the result of the present aerodynamic optimization process as the initial and optimized shapes. This figure clearly shows that to achieve the maximum value of the objective function, how airfoil geometry has changed. Initial and optimized airfoil thickness is 12 and 11.05 percent of the airfoil chord, respectively. Considering the allowed range determined for the airfoil thickness, this could be expected. The reason is its drag being intrinsically low.

The present optimization algorithm acts repeatedly and gradually approaches the optimum shape. Figure (27) makes this process more tangible and shows the steps of shaping progress. Five iterations as a representation of whole iterations show the optimization process. Furthermore, Figure (28) shows all the airfoils created by the algorithm. This figure illustrates that to obtain the optimum airfoil, the algorithm has created and investigated different shapes and has led the airfoil towards the optimum one. Generally, delaying or reducing shock waves results in increased lift and decreasing drag. Figure 29) shows the surface pressure coefficient distribution of the initial and optimized shapes. It is seen that the shock wave has significantly decreased.

To better understand how pressure is distributed around the initial and final airfoils, pressure

contours are reported separately. Figure (30) shows pressure contours around the initial airfoil (NACA0012). The existence of a relatively strong shock wave is seen in this figure. Figure (31) shows pressure contours around the optimum airfoil. The weakened shock wave that has decreased and diffused along the optimized airfoil, compared to the initial airfoil, is observed in this figure.

Figure (32) shows the best and average value of the objective function existing in the memory of the optimization algorithm versus the number of iterations. As can be seen in this figure, after about 700 iterations, the results converge to the desired value, so no significant change is seen in the value of the objective function. Therefore, these results can be introduced as converged results.

In this optimization process, the lift coefficient increased from 0.4316 to 0.517, and the drag coefficient decreased from 0.01868 to 0.01104. This proves a relative increment of lift coefficient by 19.79 percent and a relative decrement of drag coefficient by 69.13 percent. The lift-to-drag ratio (objective function) has increased from 27.95 to 46.28 (relative increment of about 67.51 percent), and this phenomenon is of designer's interest.

Precise direct optimization of an aerodynamic body will lead to the production of a wing and a body that has a low drag coefficient and a high lift coefficient. As a result, the aerodynamic device will have higher maneuverability and lower fuel consumption. The results of this research show that the modified harmony search algorithm can direct optimization with the desired accuracy. Therefore, this algorithm can be used in this matter.

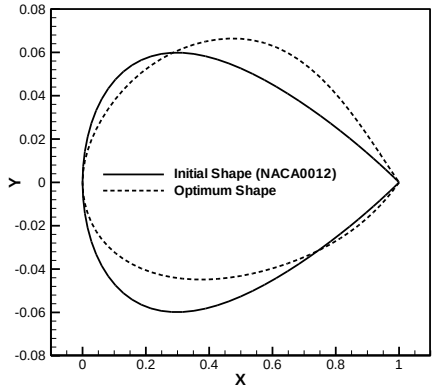

Figure 26 Directly constructed shape with maximum lift/drag target function, initial and optimum shapes (α=2.79°, M∞=0.734, Re∞=6.5E6)

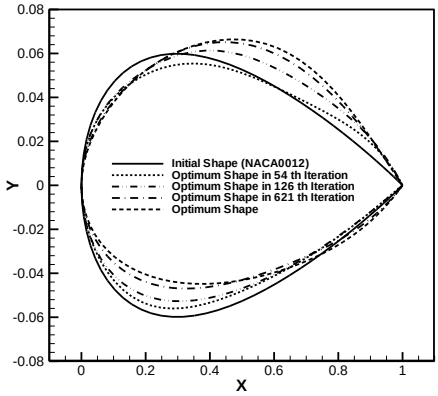

Figure 27 Shape convergence progress along with iterations in direct optimization (α=2.79°, M∞=0.734, Re∞=6.5E6)

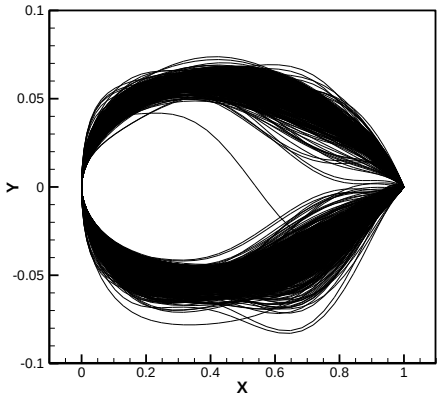

Figure 28 All obtained shapes along with direct optimization with maximum lift/drag target function (α=2.79°, M∞=0.734, Re∞=6.5E6)

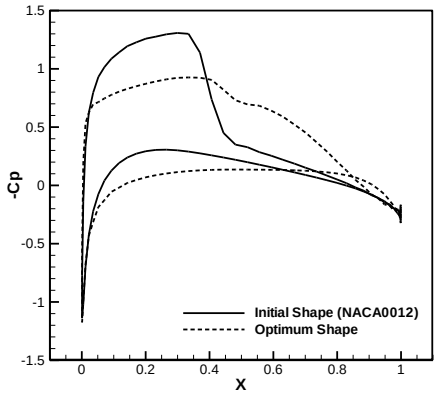

Figure 29 Surface pressure coefficient distribution of the initial and optimized shapes in direct optimization (α=2.79°, M∞=0.734, Re∞=6.5E6)

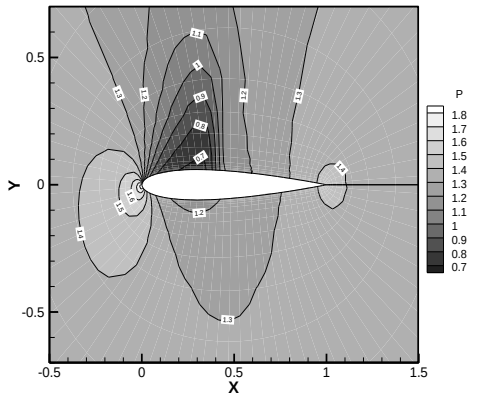

Figure 30 pressure contours of an initial airfoil in direct optimization (α=2.79°, M∞=0.734, Re∞=6.5E6)

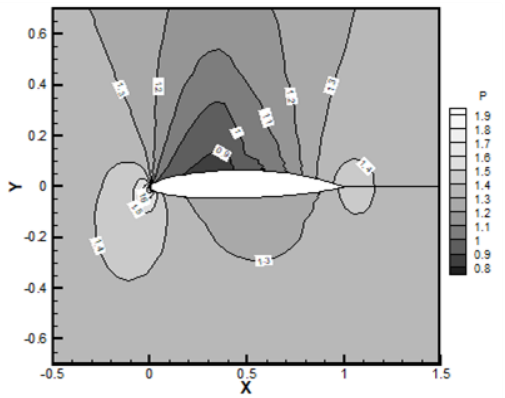

Figure 31 pressure contours of an initial airfoil in direct optimization (α=2.79°, M∞=0.734, Re∞=6.5E6)

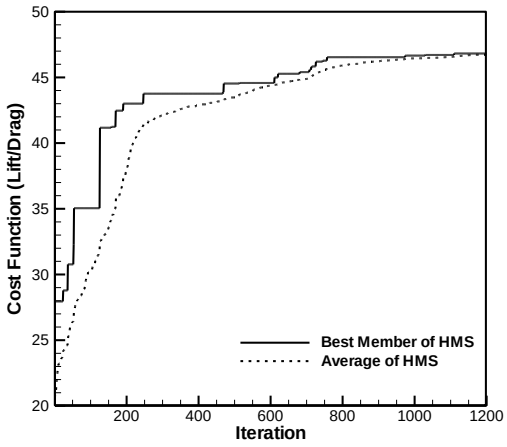

Figure 32 convergence history in direct optimization (α=2.79°, M∞=0.734, Re∞=6.5E6)

4 Conclusion

A modified harmony search optimization algorithm was introduced and was used to investigate aerodynamic problems. First, the capability of methods of airfoil shape parametrization (Bezier curves, Parsec method, and NACA 4-digit airfoil) was investigated by a modified harmony search optimization algorithm. Then, inverse and direct optimization were carried out by a modified harmony search optimization algorithm.

In inverse optimization, the objective function is the difference in pressure coefficient distributions of the airfoil being investigated and the target airfoil. In direct optimization, the objective function is lift-to-drag ratio, and the modified harmony search optimization algorithm wants to maximize this ratio.

Aerodynamic analysis of the problem was obtained using compressible Reynolds-Averaged Navier-Stokes (RANS) equations along with the Spalart-Allmaras turbulence model. General results obtained from these studies are as follows:

- Bezier curves and the Parsec method have higher flexibility than NACA 4-digit airfoil. The Parsec method was introduced as the best approach, because of less number of control parameters.
- In cases where flexibility is highly important, using Bezier curves with more control

parameters will provide better results.

- The results expressed that using a non-aerodynamic objective function, as inverse optimization design is a valuable approach for evaluating the flexibility of parameterization methods.
- The inverse optimization results showed that the present airfoil shape optimization set can obtain the target shape with high accuracy.
- The Direct optimization with a maximum lift to drag ratio target function revealed that the shock waves significantly weaken at the optimum airfoil. Furthermore, the position of shock waves moves towards the end of the body. To achieve the desirable airfoil, the pressure distribution must be smoother and more monotonic.
- Generally, the results obtained verify that using the modified harmony search algorithm together with the Parsec method provides a powerful tool for direct and inverse aerodynamic optimization.

References

[1] Avinash, G.S., and Anil Lal, S., "Inverse Design of Airfoil using Vortex Element Method", International Journal of Fluid Machinery and Systems, Vol. 11(2), pp. 163-170, (2018).

[2] Sobester, A., and Thomas, B.T., "The Quest for a Truly Parsimonious Airfoil Parameterization Scheme", ICAS Congress Including the 8th AIAA ATIO Conference, Sep. 14-19, Anchorage, Alaska, pp. 1-9, (2008).

[3] Shahrokhi, A., and Jahangirian, A., "The Effects of Shape Parameterization on the Efficiency of Evolutionary Design Optimization for Viscous Transonic Airfoils", Journal of Aerospace Science and Technology (JAST), Vol. 5, pp. 35-43, (2008).

[4] Boehm, W., "Bezier Presentation of Airfoils", Computer Aided Geometric Design, Vol. 4, pp. 17-22, (1987).

[5] Jahangirian, A., and Shahrokhi, A., "Inverse Design of Transonic Airfoils using Genetic Algorithm and a New Parametric Shape Method", Inverse Problems in Science and Engineering, Vol. 17, pp. 681-699, (2009).

[6] H´ajek, J., "Aerodynamic Optimization of Airfoils and Wings using Fast Solvers", Ph.D. Thesis, Charles University, Prague, Czech, (2009).

[7] Wauquiez, C., "Shape Optimization of Low-speed Airfoils using MATLAB and Automatic Differentiation", Licentiate's Thesis, Department of Numerical Analysis and Computing Science University of Royal Institute of Technology, Stockholm, Sweden (2000).

[8] Geem, Z.W., "Music-inspired Harmony Search Algorithm", Studies in Computational Intelligence, Vol. 191, pp. 1-14, (2009).

[9] Soemarwoto, B.I., "The Variational Method for Aerodynamic Optimization using the Navier-stokes Equations", Icase Report, pp. 71-97, (1997).

[10] Gardner, B.A., and Selig, M.S., "Airfoil Design using a Genetic Algorithm and an Inverse Method", 41st Aerospace Sciences Meeting and Exhibit, 6-9 January, Reno, Nevada, (2003).

[11] De' Michieli Vitturi, M., and Beux, F., "A Discrete Gradient-based Approach for Aerodynamic Shape Optimization in Turbulent Viscous Flow", Finite Elements in Analysis and Design, Vol. 43, pp. 68–80, (2006).

[12] Shahrokhi, A., and Jahangirian, A., "Airfoil Shape Parameterization for Optimum Navier-stokes Design with Genetic Algorithm", Aerospace Science and Technology, Vol. 11, pp. 443-450, (2007).

[13] Xu, Z., and Xia, J., "Aerodynamic Optimization Based on Continuous Adjoint Method for a Flexible Wing", International Journal of Aerospace Engineering, Article ID 4706925, (2016).

[14] Alves, R., Goncalves, L., and Aguiar, J., "Parsec Parameterization Methodology for Enhancing Airfoils Geometry using PSO Algorithm", CILAMCSE 2016, Proceedings of the XXXVII Iberian Latin-American Congress on Computational Methods in Engineering, Braslia, DF, Brazil, November 6-9, (2016).

[15] Ebrahimi, M., and Jahangirian, A.R., "Airfoil Shape Optimization with Adaptive Mutation Genetic Algorithm", Journal of Aerospace Science and Technology (JAST), Vol. 11(1), pp. 47-59, (2017).

[16] Yu, Y., Lyu, Z., Xu, Z., and Martins, J., "On the Influence of Optimization Algorithm and Initial Design on Wing Aerodynamic Shape Optimization", Aerospace Science and Technology, Vol. 75, pp. 183-199, (2018).

[17] Fesanghary, M., Damangir, E., and Soleimani, I., "Design Optimization of Shell and Tube Heat Exchangers using Global Sensitivity Analysis and Harmony Search Algorithm", Applied Thermal Engineering, Vol. 29(5–6), pp. 1026-1031, (2009).

[18] Kaveh, A., Shakouri, A., and Abadi, M., "Cost Optimization of a Composite Floor System using an Improved Harmony Search Algorithm", Journal of Constructional Steel Research, Vol. 66(5), pp. 664-669, (2010).

[19] Yousefi, M., Enayatifar, R., Darus, A.N., and Abdullah, A.H., "Optimization of Plate-fin Heat Exchangers by an Improved Harmony Search Algorithm", Applied Thermal Engineering, Vol. 50(1), pp. 877-885, (2013).

[20] Boryczka, U., and Szwarc, K., "The Adaptation of the Harmony Search Algorithm to the ATSP with the Evaluation of the Influence of the Pitch Adjustment Place on the Quality of Results", Journal of Information and Telecommunication, Vol. 3(1), pp. 2-18, (2019).

[21] Yarmohamadi, H., Zhang, Q., Jiao, P., and Alavi, A.H., "An Enhanced Adaptive Global-best Harmony Search Algorithm for Continuous Optimization Problems", Engineering Reports Journal, Vol. 2(11), pp. 1-19, (2020).

[22] Abu Doush, I., and Santos, E., "Best Polynomial Harmony Search with Best β-hill Climbing Algorithm", Journal of Intelligent Systems, Vol. 30(1), pp. 1-17, (2020).

[23] Zhang, Y., Li, J., and Li, L., "A Reward Population-based Differential Genetic Harmony Search Algorithm", Algorithms, Vol. 15(23), pp. 1-19, (2022).

[24] Iglesias, A., “Bezier Curves and Surfaces", Department of Applied Mathematics and Computational Sciences University of Cantabria UC—CAGD Group”, https://ocw.unican.es/pluginfile.php/1949/course/section/2243/01-Bezier.pdf, (2001).

[25] Endo, M., "Wind Turbine Airfoil Optimization by Particle Swarm Method", M.Sc. Thesis, Department of Mechanical and Aerospace Engineering University of Case Western Reserve, Cleveland, Ohio, USA, (2011).

[26] Spalart, P.R., and Allmaras, S.R., "A One-equation Turbulence Model for Aerodynamic Flows", La Recherche Aerospatiale, Vol. 1, pp. 5-21, (1994).

[27] Jameson, A., and Schmidt, W., "Numerical Solution of the Euler Equations by Finite Volume Methods using Runge-Kutta Time-stepping Schemes", American Institute of Aeronautics and Astronautics (AIAA Journal), Vol. 81, pp. 1259, (1981).

[28] Jameson, A., "The Present Status, Challenges, and Future Developments in Computational Fluid Dynamics", Twelfth Australasian Fluid Mechanics Conference, Dec. 7-10, The University of Sydney, Sydney, Australia, pp. 423-468, (1995).

[29] Lee, K.S., and Geem, Z.W., "A New Meta-heuristic Algorithm for Continuous Engineering Optimization: Harmony Search Theory and Practice", Computer Methods in Applied Mechanics and Engineering, Vol. 194, pp. 3902–3933, (2005).

[30] Pan, Q.K., Suganthan, P.N., Tasgetiren, M.F., and Liang, J.J., "A Self-adaptive Global Best Harmony Search Algorithm for Continuous Optimization Problems", Applied Mathematics and Computation, Vol. 216, pp. 830–848, (2010).

Nomenclature

a_n	Parsec coefficients
A	cell volume
B_i	Bernstein polynomials
BW	Bandwidth
c	chord
C	vector
C_x, C_y	components of vector C
D	source term related to the turbulent model
E	sum of internal and kinetic energies
F	objective function
F_i,G_i	inviscid flux vectors
F_v,G_v	viscous flux vectors
HMS	harmony memory size
HMCR	harmonic memory considering the rate
M_∞	Mach number

m	maximum camber
n	order of a polynomial
p	maximum camber location
P_i	Bezier points
p	static pressure
q_x, q_y	heat fluxes
PAR	pitch adjustment rate
Q	summation of inviscid and viscous flux vectors
Re	Reynolds number
S	source term
t	time
u,v	velocity components
UB, LB	upper and lower limits of the variable
W	vector of the quantities
X	Parsec control parameters
x,y	longitudinal and transverse coordinates
$\vec{X}$	vector of design variables
X_{up}, Y_{up}	coordinates of upper surfaces of NACA 4-digit airfoils
X_{lo}, Y_{lo}	coordinates of lower surfaces of NACA 4-digit airfoils
y_{th}	thickness distribution of airfoils
y_c	middle curve line of the airfoil
z_{Parsec}	Parsec equation
Z_{lo}, Z_{up}	the maximum position of lower and upper surfaces
Z_{xxup},Z_{xxlo}	the second derivative at the maximum position of lower and upper surfaces
Z_{TE}	the thickness of the final edge
α	angle of attack
α^0	coefficients of Runge-Kutta scheme
θ	slope of the middle curve of an airfoil
ρ	density
σ	constant of Spalart-Allmaras turbulence model
τ	maximum thickness
τ_{xx}	Stress tensor components
$\hat{\nu}$	intermediate chaotic kinematics viscosity
ν	laminar kinematics viscosity
ΔZ_{TE}	the position of the final edge
Δt	local time step