AUT Journal of Mechanical Engineering

AUT J. Mech. Eng., 4(4) (2020) 435-448
DOI: 10.22060/ajme.2020.16445.5819

Variable Population Models in a Neural Network- Augmented Genetic Algorithm for Shape Optimization

A. R. Davari*, S. M. Pourkiaei

Department of Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran

ABSTRACT: The optimization process for an airfoil using genetic algorithm has been an increasingly popular problem in recent years. In recent years, the role of the population model in genetic algorithm has been underlined. In many of the recently proposed models, the convergence time was adversely increased or the elitism or mutation operators failed to work properly due to the inherent oscillations in the oncoming generations. In this paper, the idea of continuous variable population size has been introduced to optimize the airfoil shape. This scheme has been shown to converge to higher performance airfoils and can decrease the convergence time, without any oscillatory behavior. Furthermore, to reduce the run time to evaluate the fitness value, a generalized regression neural network has been developed and trained by the numerical data to evaluate the lift to drag ratio for a vast range of NACA four digits airfoils. The values predicted by this neural network have been proved to be in good agreement with the other experimental and numerical data and were then used to calculate the lift-to-drag ratios as the fitness value for various airfoils generated during the optimization process. The idea can ever be more effective in similar problems with a huge amount of computational time to calculate the fitness values and converge to the most efficient airfoil in a reasonable time.

Review History:

Received: May, 28, 2019
Revised: Dec. 12, 2019
Accepted: Dec. 29, 2019
Available Online: Jan. 04, 2020

Keywords:

Genetic Algorithm
Population
Wind Turbine
NACA Airfoils
Optimization, GRNN

1- INTRODUCTION

Airfoil optimization has been a practical engineering problem in recent years. Though each airfoil has its own geometric parameters and produces particular performance characteristics, no realistic airfoil shape can necessarily be associated with a given performance. Therefore, the airfoil optimization problem today is a tradeoff between performance and physical constraints. For this reason, the classical optimization methods mainly adopt the trial and error approach, which strongly relies on the designer's future experience rather than the current needs. In such problems, a global optimization method based on evolutionary algorithms is likely to shorten and simplify the iterative optimization process and improve the final result.

Recently, Genetic Algorithms (GAs) have emerged as available tools in a wide range of application areas, such as medicine, image processing, laser technology, aeronautics, artificial neural networks, control, robotics, etc. The genetic algorithm belongs to a class of heuristic searching for a given optimization problem and attempts to imitate the mechanism of Darwin's survival theory [1].

The basic idea of a GA uses the probabilistic transition and non-deterministic rules to search the possible solution spaces, and eventually achieved a global optimum. It is realized in a computer by including the selection, crossover, and mutation operations in the algorithm. Bechert [2] provided an interesting summary of the early use of evolutionary strategies for experimental shape optimization in fluid dynamics. These approaches were very similar to the modern genetic algorithms used today.

*Corresponding author's email: ardavari@srbiau.ac.ir

Applications of GAs in the context of aerodynamic optimizations have been the focus of considerable attention for the past two decades. Chen [3] has reported a multi-objective genetic algorithm to optimize some classes of thick airfoils used in wind turbine blades. They have exploited a purely numerical approach to evaluate the airfoils with a classic genetic algorithm for optimization.

Gage and Kroo [4] applied GAs to the topological design of non-planar wings. Yamamoto et al. [5] proposed a new operator for GAs to determine the evolutionary direction, which is less time-consuming than the usual gradient-based optimizers. The introduction of evolutionary direction into GA, by this method, is more efficient than the traditional schemes in finding the global optima.

Anderson [6] used a penalty weight approach to consolidate several different subsonic wing design objectives into a single objective functional, but has noted that the solutions had an undesirably strong dependence on the weights. Doorly et al. [7] used a parallel GA in conjunction with a flow solver to determine optimal airfoil shapes approximated using a B-spline. The population size in the latter two studies was

80 and this number seems to be typical of research involving shape optimization using GAs.

More recently, various successful applications of the GA have been reported in multi-element airfoil design [8], helicopter blade shape optimization [9], drag reduction in hypersonic flow [10], and performance optimization of a vapor compression refrigeration system [11].

Once the high performance of the GA in optimization problems was acknowledged, attempts were made to improve this algorithm and its methodology to decrease the run time and enhance the algorithm efficiency and the convergence speed. The improved GAs have been usually shown to offer more optimized results than the classic ones.

Hacioglu [12] has augmented the genetic algorithm with a neural network to increase the speed of the aerodynamic calculations. He exploited the neural network to predict the airfoil shape corresponding to a given pressure distribution, though remarkable errors have been raised at the airfoil trailing edge and poor agreements were observed between the airfoil shape and the corresponding pressure distribution. The idea has been followed by Chen [13] to optimize the blunt trailing edge airfoils. They also employed neural networks to predict the aerodynamic coefficients using numerical data with a classic genetic algorithm in the optimization phase. In another survey, Chen [14] has reported a successful combination of the radial basis neural network and the response surface method, RSM, including the Kriging method to construct an aerodynamic model using numerical data to speed up the computations of the fitness value in a genetic algorithm.

For complex Computational Fluid Dynamics (CFD) applications, Duvigneau [15] proposed an inexact pre-evaluation using a neural network to reduce the number of evaluations through the flow solver at each generation and a hybrid method to perform the final local search. This method is shown to give acceleration to the algorithm. Liu [16] offered an intelligent cross-over process in GA to select good genes from the parents. Su et al [17] suggested an approximate model to evaluate the individual's fitness to reduce the computational costs in the GA.

Hacioglu [18] examined an improved vibrational mutation technique in a genetic algorithm as a stochastic search method to accelerate the algorithm for airfoil inverse design. He showed that a low population rate and a short generation cycle are the most important benefits of this method. He also combined GA with a neural network to search the design space and proposed an interactive process between a genetic algorithm and a neural network to improve the exploration power of the GA [19].

A genetic algorithm for an optimization problem starts with an initial solution as the parent or the first generation. The children or the next generations are created from the initially given solution by the cross over, mutation, and elite selection operations. In a classic genetic algorithm, the next generations have the same size as the previous ones. However, in the real world, the population size varies from one generation to the next. The number of individuals in a generation directly affects the convergence time and the final solution accuracy.

Some limited investigations started in the last two decades, have been devoted to this issue. Goldberg [20] studied the effect of stochasticity on the convergence behavior of the genetic algorithms. He derived a theoretical relation to calculate the optimized population size in a GA in each generation. Despite finding some improvements in the algorithm performance for a proper choice of the population size, the idea has been worked out through mathematical manipulations and no practical optimization problem has been examined to prove its applicability to real engineering problems.

Arabas et al. [21] proposed an adaptive method for maintaining variable population size. In this method, the lifetime and the age of each member have been taken into account. As time elapses, the old members are removed and are replaced by the new ones, creating a new population in the next generation. The final results would strongly depend on the choice of the optimum lifetime for the members and this strongly restricts the applicability of the proposed method.

Koumousis [22] developed a genetic algorithm with variable population size and periodic partial re-initialization of the population in the form of a saw-tooth function. He studied the synergy of the combined effects of population size variation and re-initialization in performance enhancement in GA. However, a saw-tooth population model may cause an oscillatory behavior in the best fitness values between the generations and can prevent converging to the optimum solution or may considerably increase the convergence time.

In a very recent survey, the biologically-inspired concept of the hidden gene has been introduced [23]. The hidden gene in a chromosome does not affect the fitness value. It just carries the information to the next generation. With this concept, different lengths can be coded in a chromosome and can be employed in trajectory optimization problems.

Up to now, the impact of population size on the convergence speed and accuracy of the results in a GA is perfectly underscored. However, despite the valuable theoretical and basic information achieved on the matter, the authors found nearly no practical problem in the literature to be treated by the idea of variable population size.

The saw-tooth population model, the vibrational mutation technique, and the concept of lifetime for each member are the highlighted ideas in this regard. As stated earlier, assigning a lifetime for a member adds a new variable to the problem and there is no physical base to determine the proper value of this lifetime to achieve the desired optimum solution. The discontinuities in the saw-tooth model in some problems may induce oscillations during the evolution and inhibit convergence to the optimum point. The vibrational mutation technique has an inherent randomness behavior and can prevent the mutation of some elite members to the next generation.

Many optimizations using GA, on the other hand, are performed nowadays by MATLAB® in which, any change in the logic and the structure of the GA operators are not always possible and easy.

In this paper, several ascending and descending population continuous models were implemented in a genetic algorithm to optimize the aerodynamic shape of a series of thin airfoils which are widely used in such applications as the wind turbine blade section and general aviation. The GA performance with these models was compared to the classic algorithm of constant population size. The lift to drag ratio, *L/D*, was the fitness value to be optimized. Since calculating this ratio in a viscous flow is a time-consuming process and should be repeatedly performed, several 4-digits NACA airfoils have been analyzed using ANSYS Fluent and along with the existing experimental data, a database has been created to express the lift to drag ratio for various 4-digit NACA airfoil as a function of the airfoil thickness, camber and position of the maximum camber. A generalized regression neural network, GRNN, was then developed and trained by this database to calculate the fitness value for each airfoil in the optimization process, with high accuracy and in a minimum time.

2- OPTIMIZATION PROBLEM

The problem under consideration is the impact of the population size in a GA code developed for optimization of an airfoil in the family of NACA 4-digit series which are mostly used in certain wind turbine blade sections. Some of the wind turbines utilize NACA series airfoils such as NACA 230XX and NACA 44XX series with a thickness variation of %28 at root and %12 at tip sections [24].

These airfoils have a fairly sufficient lift for wind turbines. However, one of the most important disadvantages of them is their high sensitivity to surface containment. Despite, their good aerodynamic performance and easy-to-manufacture behavior make them still popular choices for wind turbines.

The initial airfoil for producing the next generations was NACA 4415. This airfoil has a camber-to-chord ratio, m, of 0.04, the maximum camber, p, is located at 0.4 chord and the thickness-to-chord ratio, t, is 0.15. The airfoil angle of attack was considered to be zero. According to the experimental results, the values of the lift coefficient and the lift to drag ratio for this airfoil at zero angle of attack and a Reynolds number of 6×10^5, are about 0.45 and 65 respectively [25].

The problem is to achieve a more efficient member in the family of 4-digits NACA airfoils with higher values of lift to drag ratio than the original NACA 4415, considering a variable population model in the optimization process. Several ascending and descending population models during the evolution process in the GA were examined and the performance in each case was compared to the constant population model to find the best remedy for the problem under consideration.

3- AERODYNAMIC MODEL

The lift to drag ratio, *L/D*, plays a vital role in airfoil aerodynamics and is known as the aerodynamic efficiency parameter for the airfoil. In airfoil optimization problems, it is a suitable choice for the fitness function or target to be maximized. The problem has cumbersome and time-consuming calculations to determine the drag force, whereas the sectional lift can quickly be determined using the well-known thin airfoil theory [26].

To facilitate the study of the algorithm performance for various population models, a quick method was needed to calculate the lift to drag ratio of this family of airfoils in a minimum time. This was conducted by a database containing the lift to drag ratios for several four-digit NACA airfoils. An aerodynamic model based on Generalized Regression Neural Network, GRNN, has been developed and trained by this database to calculate the lift to drag ratio for the airfoils under consideration in the optimization process. At the end of the training process, this network could predict *L/D* for the airfoils produced in each generation, knowing the values of *m*, *p,* and the thickness.

Once the proper and efficient population model for the GA was determined, this approach would work as well for more complex problems with massive calculations in various exact numerical schemes. During the optimization process, several airfoils have been produced and evolved through generations to converge to the best contour shape with a maximum *L/D*. For the airfoils with no available experimental data in the literature, a numerical simulation was performed using ANSYS Fluent.

In CFD simulations for this paper, C-type structured grids have been provided around the airfoils extending 21c downstream and 12.5c in other directions in the flowfield, where c is the airfoil chord length. Totally, 49000 grid points were generated for each airfoil using ANSYS GAMBIT 2.4.6. The clearance between the grids adjacent to the surface and the wall was chosen to be 10^{-4} with a growing rate of 1.11 for the next rows. Moreover, 15 nodes were provided inside the boundary layer. The k-ω SST turbulence model was used which is a usual choice for such 2-D problems. The solver was the pressure-based SIMPLE algorithm. Some results from these numerical simulations will be presented in the coming sections to support the results predicted by the GRNN. Fig. 1 shows the grid points on two typical airfoils, NACA0012 and NACA6409.

In 4-digit NACA airfoils, the mean camber line equation is expressed as [26]:

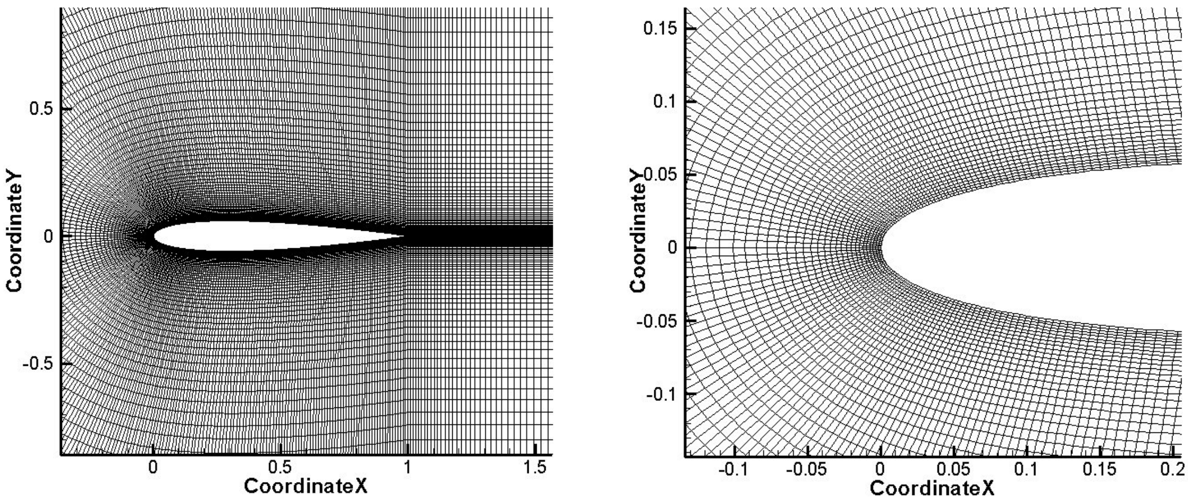
$$y_c = \frac{mc}{p^2}[2p\frac{x}{c} - (\frac{x}{c})^2] \tag{1}$$

for $x/c \le p$ and

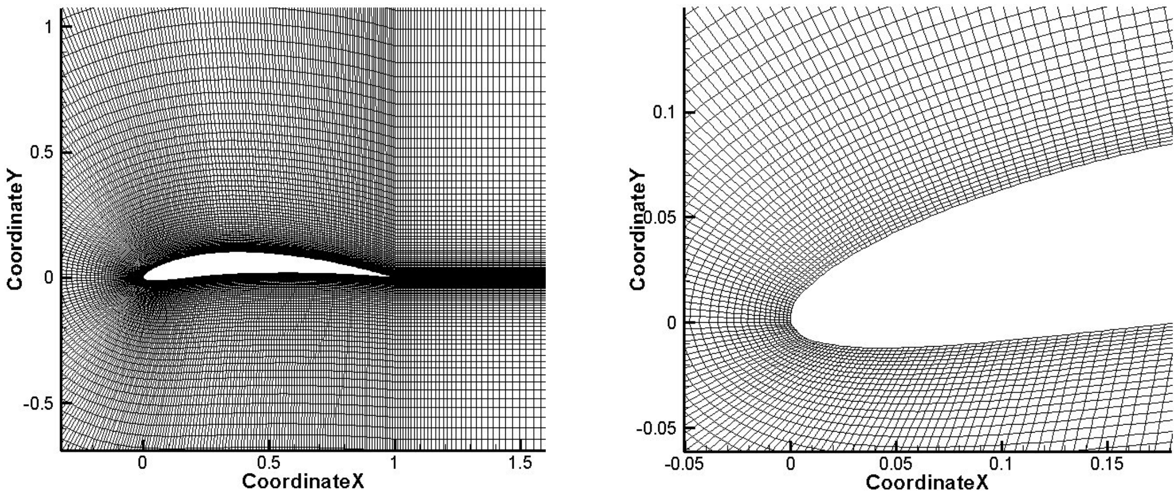
$$y_c = \frac{mc}{(1-p)^2}[(1-2p) + 2p\frac{x}{c} - (\frac{x}{c})^2] \tag{2}$$

for $x/c > p$.

As stated earlier, *m* is the maximum value of y_c expressed as a fraction of the chord *c*, and *p* is the value of *x/c* corresponding to this maximum. The thickness distribution for the NACA 4-digit sections is given by the following



(a) NACA 0012



(b) NACA 6409

Fig. 1. Grid generation for CFD calculations around two typical airfoils

equation [26]:

$$y_t = \pm 5ct[0.2969\sqrt{\xi} - 0.1260\xi - 0.3516\xi^2 + 0.2843\xi^3 - 0.1015\xi^4] \quad (3)$$

where t is the maximum thickness expressed as a fraction of the chord and ζ= x/c. Thus, using the independent variables, m, p, and t, the airfoil shape can easily be determined. To calculate the lift to drag ratio for a given profile, the values of m, p and t have been entered into the trained GRNN and the output was L/D.

Let X_i denotes the input array for the i^{th} airfoil in the training process.

$$X_i = [m_i, p_i, t_i]$$

And the output Y_i is the associated lift to drag ratio, *L/D*. The probability density function of the training inputs and outputs used in GRNN is the normal Gaussian type distribution and each input X_i is the mean for this normal distribution.

The independent array *X* contains the geometric information of the newly born airfoil in a generation and the dependent variable *Y* is the associated *L/D* value for this airfoil. In this paper, the range of the inputs, X_i, was from [m,p,t]=[0,0,0.12] to [0.06, 0.4, 0.09]. This covers nearly the full range of NACA 4 digit series used for wind turbine blade sections.

Using the training sets of X_i and Y_i to correlate the new airfoil in the evaluation phase to the existing airfoils from the

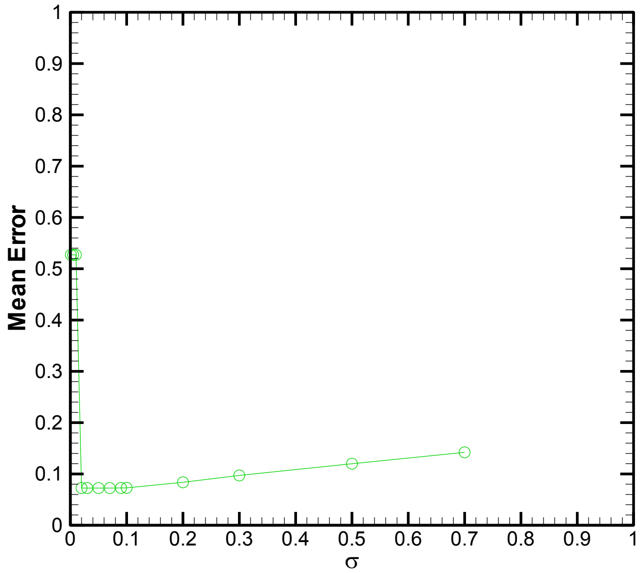


Fig. 2. Variations of the mean error with the correlation parameter, σ at zero angle of attack

training phase, the estimation Y(X), i.e. the lift to drag ratio for the new airfoil would be in the following form [27]:

$$Y(X)=\frac{\sum_{1}^{n} Y_i \exp[-\frac{D_i^2}{2\sigma^2}]}{\sum_{1}^{n} \exp[-\frac{D_i^2}{2\sigma^2}]} \tag{4}$$

where n is the number of the training data sets and D_i is the distance between each training sample and the point of prediction which is defined as:

$$D_i^2=(X-X_i)^T(X-X_i)$$

This distance parameter in the present work indicates how similar is an airfoil born in a generation, to either of the NACA 4-Digits airfoils used in the network training phase. Each airfoil in a generation was produced by a random selection of the parameters *m*, *p,* and *t*. Knowing the input array *X*, the output *Y*(*X*), which is the fitness value for the airfoil, is predicted by Eq. (4).

The geometric parameters and the corresponding lift to drag ratios of each training airfoils are in arrays X_i and Y_i, respectively. The parameter σ is the probability width for each training data set. It is the standard deviation of the training data and is usually called the smoothness parameter. As observed in Eq. (4), the prediction accuracy directly depends on the value of σ. In this paper, the holdout method, suggested by Specht [27], was used to expediently choose σ. In this method, one sample from the entire set is removed and for a fixed σ, GRNN is used again to predict this sample with the remainder set of training samples. The squared difference between the predicted value of the removed training sample and the training sample itself is then calculated and stored.

The process is repeated for several values of σ and for all training samples. The σ for which, the sum of the mean squared difference is a minimum for all mean squared differences, is the proper choice for σ and should be used for the predictions using this set of the training samples.

Shown in Fig. 2 is the mean prediction error for various amounts of σ for NACA 4415 at zero angle of attack as the initial airfoil to start the optimization process in this paper. Here, the mean estimation error is defined as the difference between the value of *L/D* for the airfoil predicted by the GRNN for each choice of σ and the corresponding value obtained from the present numerical solution.

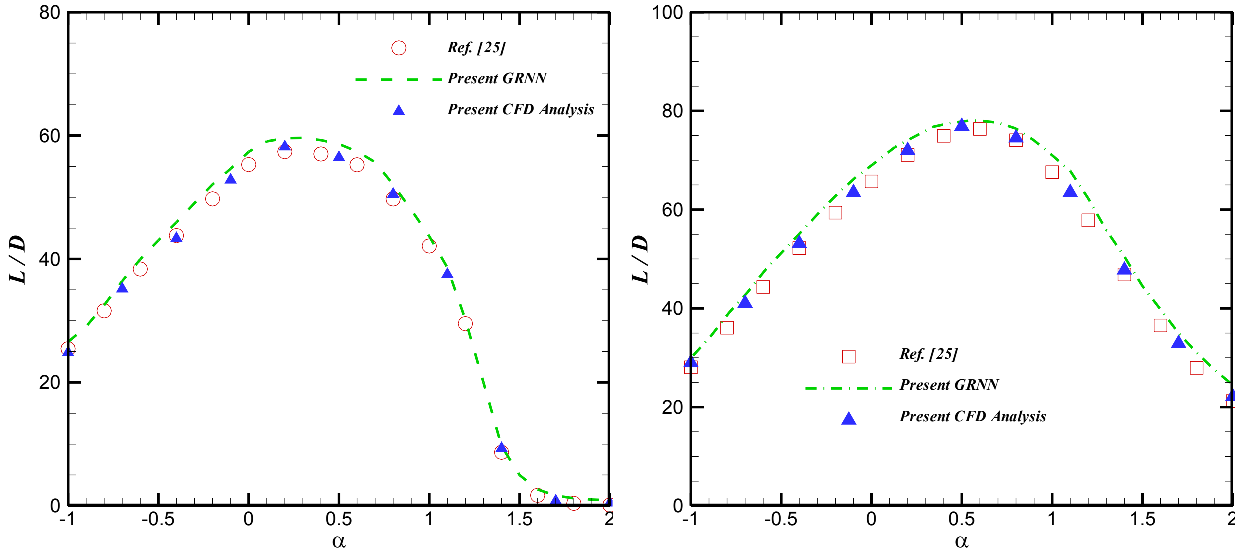
For the range of σ between 0.01 to 0.1, the mean estimation error is minimum. Beyond this range, a dramatic increase is observed in prediction error. The value σ=0.05 has thus been selected for present predictions throughout the optimization process.

4- MODEL VALIDATION

The performance of the GRNN with this choice of σ to predict the *L/D* will be first shown to give reasonable results for various NACA 4-digits series. Fig. 3 shows the values of *L/D* predicted by the present GRNN for NACA 4415 and NACA 2424 airfoils as typical. Within the small angles of attack range considered in this paper, the predicted values of *L/D* are in good agreement with both the experimental and numerical data, especially at zero angle of attack. However, the values predicted by the GRNN primarily show a slight over-estimation comparing to those obtained by CFD and experiment. This can make sure that the aerodynamic model constructed to calculate the fitness value, i.e. the *L/D*, in the optimization process works satisfactorily.

5- THE GA OPERATORS

In this paper, a classic approach, i.e. constant population



(a) NACA 2424 (b) NACA 4415

Fig. 3. The lift to drag ratio, predicted by the present GRNN

GA was first developed as a baseline and then several variable population models were implemented as well and the associated performances were compared. Totally 100 generations were considered in this survey and in the classic code, the population size in each generation was chosen to be constantly 100 members. A permutation encoding system was used to identify the chromosomes and they have been designated by natural numbers.

For airfoil shapes as the members in a GA-based optimization, this system has been found to be more efficient. The initial population of the independent variables, i.e. m, p, and t in this study were randomly generated by a FORTRAN code. The Random generation was according to the CPU clock to prevent a duplicated random number in each generation. This inhibits generating identical airfoils in a generation and remarkably increases the convergence speed in comparison to the conventional random generation mechanism in which, the random number is picked up from a pre-produced seed of numbers. Once all random numbers within the seed are used, the same numbers will be repeated over and over again.

The fitness function, as stated before, was the airfoil lift to drag ratio and has been calculated by the GRNN trained to predict L/D for NACA 4-digits series as a surrogate to time-consuming CFD runs for about10^4 airfoils. The top %5 of members in each generation with higher values of L/D as predicted by the aforementioned trained GRNN have been directly transferred to the next generation by an elite selection operator.

Each new generation receives 65% of the top members of the previous generation by a "*cut and splice*" approach cross-over operator and the remainder 30% were constructed by the mutation operator from the previous generation. The mutation probability in this study was considered to be 0.01% of each chromosome fitness value, i.e. its L/D value. With this operation, the good chromosomes which have been vanished during elite selection or cross-over operations could be retrieved. Using a natural number designation for the airfoils in this algorithm, this mutation operator would have a remarkable effect in converging the algorithm to the best result.

Two constrainers were provided in the algorithm during evolution. The first is the number of generations that terminates the process at N_{Gen}=100, even if the convergency has not been achieved, i.e. in the case of an unsuccessful optimization. The second constraint is the airfoil shape produced in each generation. The airfoils having the camber values larger than m=0.0475 or the maximum camber locations larger than p=0.475, are somehow, unrealistic and do not worth considering. Once such airfoil is born, it will be replaced by another choice in the desired ranges of m and p.

To sum up, the procedure followed in this paper for airfoil optimization has been elucidated schematically in Fig. 4. A database has been provided containing the geometric parameters of a vast range of NACA 4 digit airfoils, including m, p, and t as inputs and the lift-to-drag ratio as output for each airfoil. This database encompassed both the available airfoils data in the literature and those analyzed by the numerical simulations performed by the authors.

A GRNN subroutine in FORTRAN was then developed and trained by this database to get the values of *m*, *p* and *t* as input and deliver the associated *L/D* for any 4-digit NACA airfoil. Since the exact numerical simulation of various airfoils is a time-consuming process, this is an essential and innovative step in optimization. This subroutine has been repeatedly called in the main genetic algorithm program during the optimization process to evaluate the fitness value, i.e. *L/D* for each airfoil. In the main GA program, several ascending and descending population models have been included and

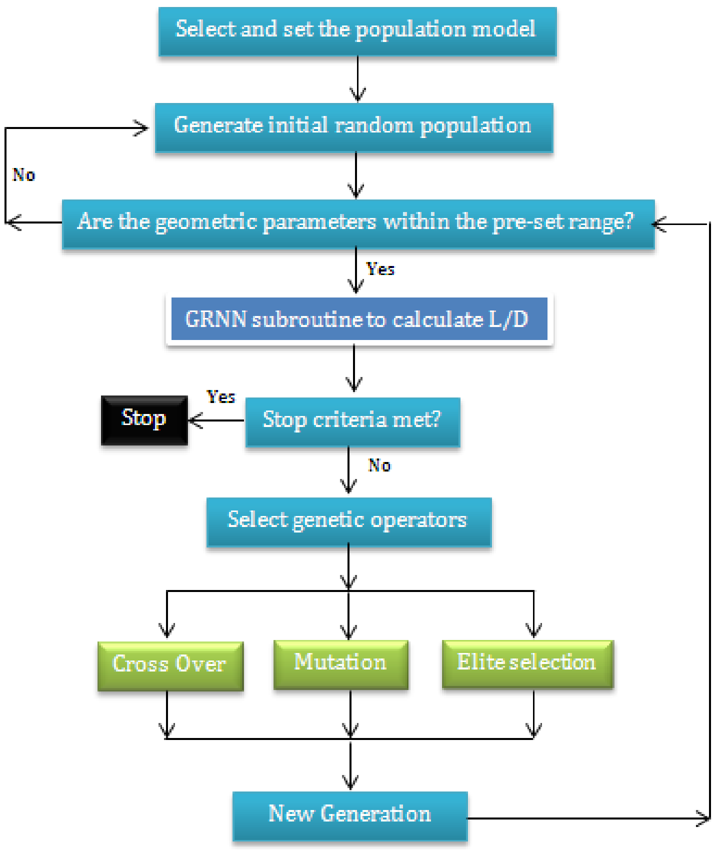


Fig. 4. The optimization flow chart in the present paper

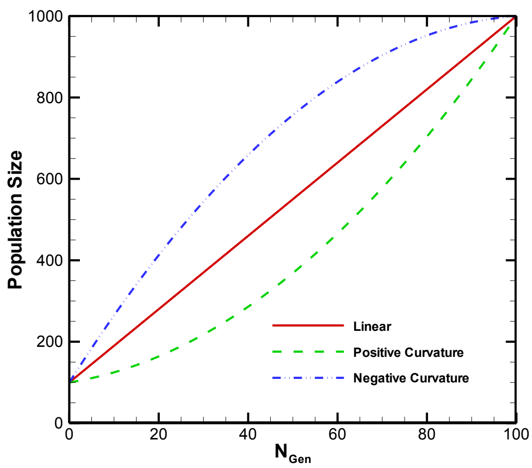


Fig. 5. The ascending population models

the user can select the desired one. The optimization process starts with selecting the population model. For each model, the optimization process has been individually carried out and the associated performance parameters were then compared.

6- POPULATION MODELS

A robust code based on the genetic algorithm was developed and different population models were implemented to compare the convergence behavior as well as the final results. A constant 100 members in each generation was first examined as a baseline. Three ascending population models including linear increasing, positive curvature increasing, and negative curvature increasing models were considered in the second step. The two later models were third-order polynomials. For these ascending models, the population increases from 100 initial members in the first generation to 1000 members in the 100th generation. Figs. 5 and 6 show the increasing and decreasing models, respectively.

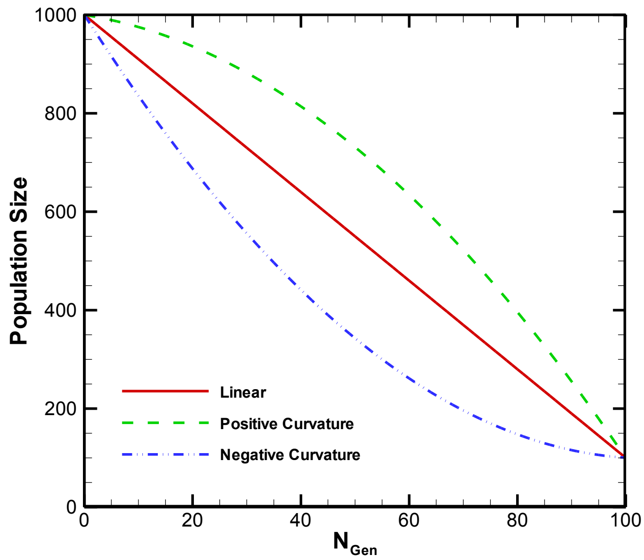


Fig. 6. The descending population models

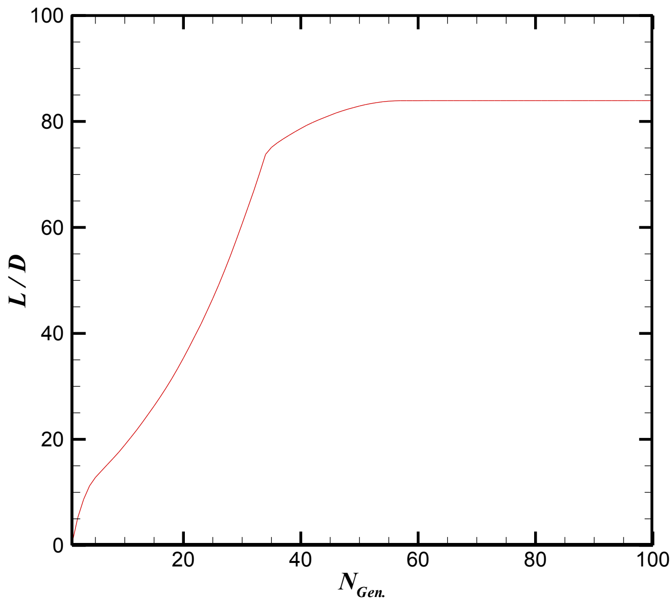


Fig. 7. The evolution of the averaged L/D for different generations in a constant population model

For the descending population schemes, shown in Fig. 5, the corresponding linear, positive curvature, and negative curvature models were used to decrease the initial 1000 members in the first generation to 100 members in the last one.

7- RESULTS AND DISCUSSION

Fig. 7 shows the variations of the averaged *L/D* in each generation as the algorithm evolves with a constant population of 100. The convergence occurs at about N_{Gen}=55 and the optimized airfoil has a lift to drag ratio of 84, which is higher than that for the original NACA 4415 airfoil. This behavior would be the baseline to be compared with the variable population models.

Shown in Fig. 8 are the best airfoils, i.e. those having the maximum *L/D* in each generation for some typical successive generations up to the convergence point. For the last generations, both the optimized airfoil camber and the associated location of the maximum camber increase. At about N_{Gen}=40 the shape variations decrease and it remains more or less constant. From N_{Gen}=55 on, the algorithm converges to the best result.

For the ascending population models, the evolution of lift to drag ratio through the generations is shown in Fig. 9. It shows that there is a subtle increase in the optimized lift coefficient compared to the constant population scheme.

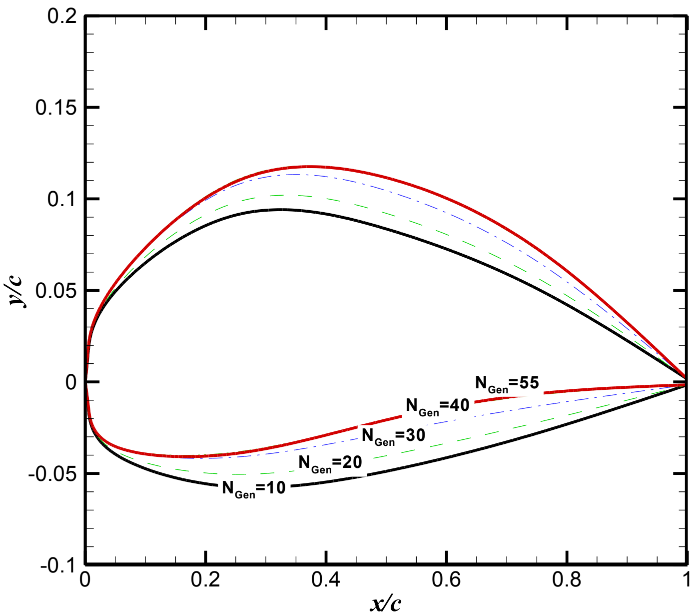


Fig. 8. The evolution of the best airfoil in each generation for typical generations up to convergence point for constant population model

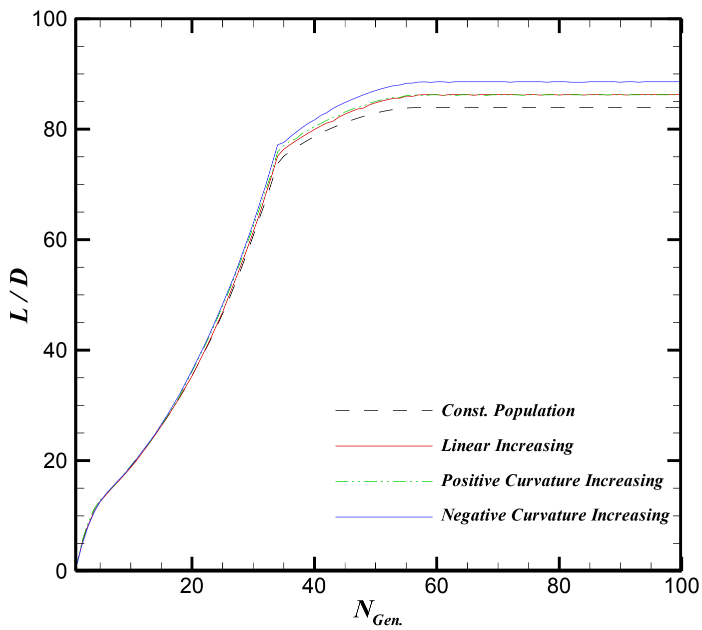


Fig. 9. The evolution of the averaged L/D for different generations in various ascending population models

For the linear ascending model, the optimized value of *L/D* has been increased to 86.2, which means that the algorithm has been converged to a relatively better and more efficient airfoil. However, the convergence point has been postponed to higher generations, namely N_{Gen}=57.

In ascending population with positive curvature, the optimized *L/D* and the convergence generation are 86.2 and N_{Gen}=57 respectively, which are the same as the linear ascending model. For the negative curvature ascending model, the convergence generation remains the same as that of the two later ones, however, the optimized *L/D* has been converged to a slightly higher value of 88.5. This shows that the ascending population models have improved the solution accuracy comparing to the constant population while deteriorated the convergence speed by a small value.

Figs. 10 to 12 show the evolution of the best airfoil in each generation for some typical generations up to convergence point for the three ascending population models mentioned above. A comparison between the airfoil evolution trend with that shown in Fig. 8 approves that the shape optimization mechanism towards the best airfoil with maximum *L/D* is essentially the same in both constant and ascending models,

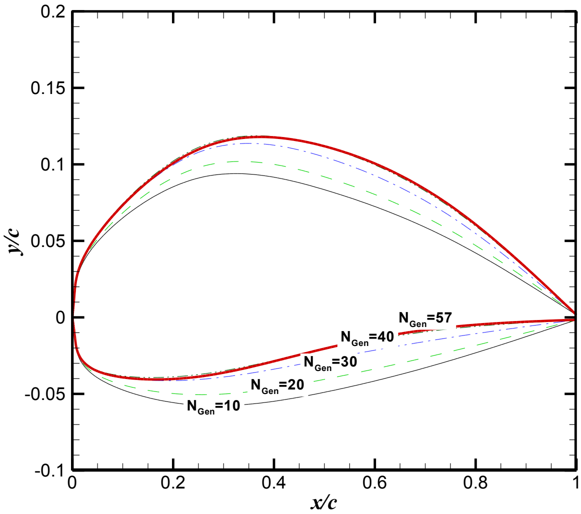


Fig. 10. The evolution of the best airfoil in each generation for typical generations up to convergence point for linear ascending population model

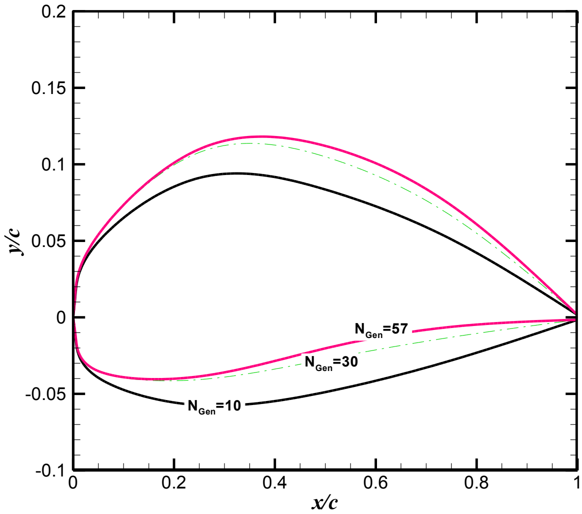


Fig. 11. The evolution of the best airfoil in each generation for typical generations up to convergence point for positive curvature ascending population model

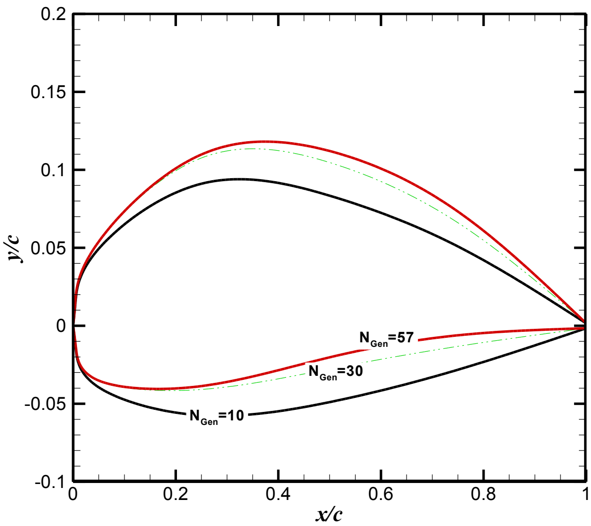


Fig. 12. The evolution of the best airfoil in each generation for typical generations up to convergence point for negative curvature ascending population model

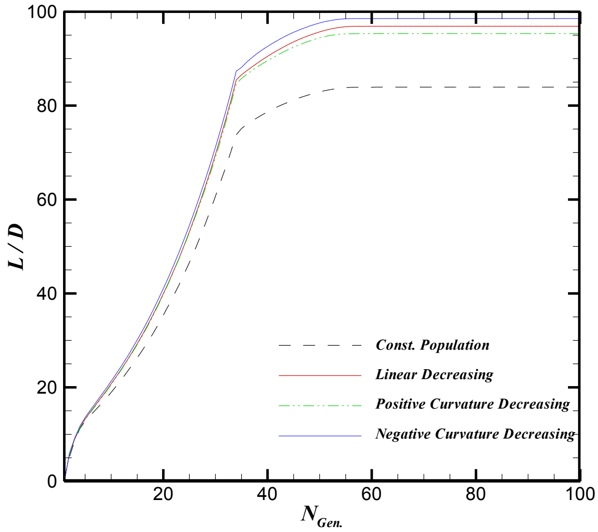


Fig. 13. The averaged L/D growth during evolution in various descending population models

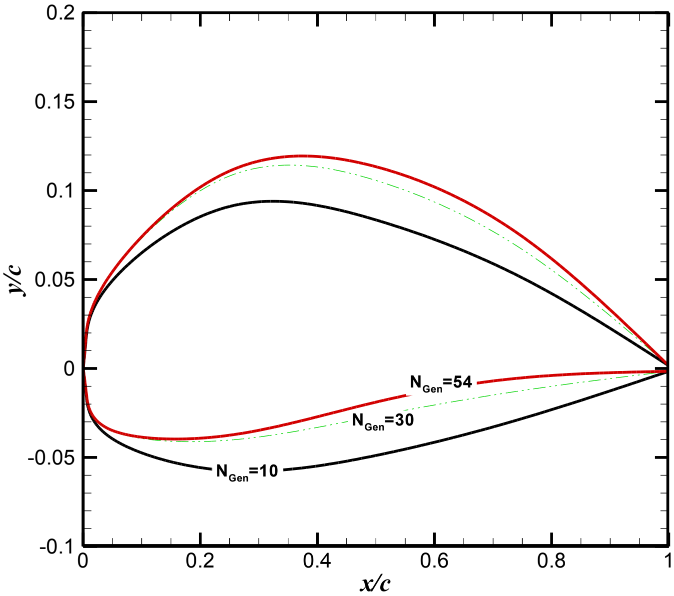


Fig. 14. The best airfoil in each generation for typical generations up to convergence for positive curvature descending population model

especially for the initial generations.

Shown in Fig. 13 is the evolution of the lift coefficient for the descending population model. At a glance, the higher performance of this model compared to the fixed and descending population ones can be observed. In linear descending, the optimized lift to drag ratio is 96.8 and the algorithm converged at N_{Gen}=56.

For the positive curvature descending scheme, the optimized *L/D* decreased to 95.2 and the convergence was observed to occur at the 54th generation which means a little lower optimized value at a faster optimization process than the linear descending case. For the last model, i.e. negative curvature descending, the optimized *L/D* has been reduced to 98.5, converging at 56th generation. This model is less favorable than the two former descending ones.

The evolution of the best airfoil in each generation for some typical generations up to convergence for positive curvature descending population model, which observed to be the best model, is shown in Fig. 14. The differences between the airfoil shape at the 30th generation and that at the end of the process at N_{Gen}=54 starts to decrease. This approves faster convergence comparing to the other models.

A comparison has been made between the final airfoil optimized by the negative curvature descending population model, as an instance, and the initial NACA4415. Fig. 15 shows the surface mold lines for these two airfoils.

To facilitate performance comparison, these results have been summarized in Table 1. The number of the function calls, NFC, has been determined by defining a counter in the code when calling the operators. Note that NFC in this case

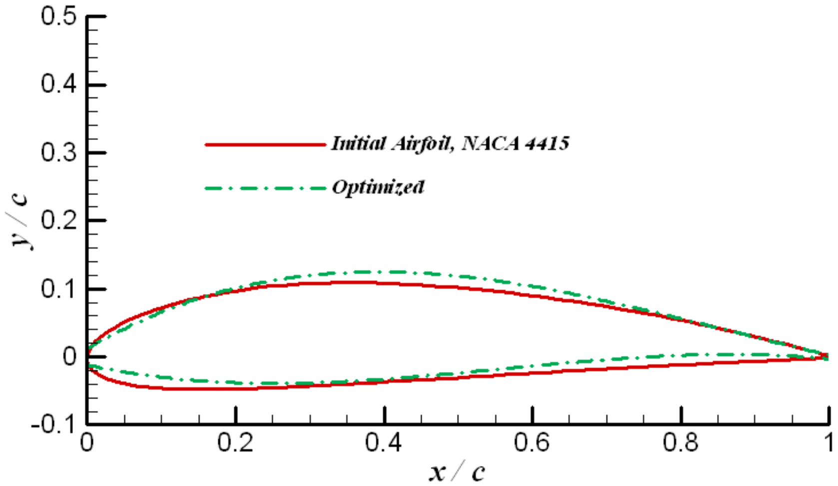


Fig. 15. The final airfoil optimized by the negative curvature descending population model in GA

Table 1. Summary of the performances for each model

Model	*L/D*	NFC	% Increase in *L/D* comparing to the constant population	The convergence generation
Cont. population (Baseline)	84	5600	-	55
Linear ascending	86.2	20667	2.7	57
+ curvature ascending	86.2	13284	2.7	57
- curvature ascending	88.5	29115	5.4	57
Linear descending	96.8	42636	15	56
+ curvature descending g	95.2	48492	13	54
- curvature descending	98.5	34401	17	56

can be a measure of the operation time as well.

According to this table, despite all of the variable population schemes were more successful in the optimization process than the constant population model, the descending population models have overally shown to have a better performance in either convergence speed or the maximized lift to drag ratio than the others. Among various population models examined in this paper, the higher performance of the airfoil obtained by the negative curvature descending model compromises its NFC comparing to other schemes and makes it a suitable choice for such optimization problems.

Excluding the constant population classic model, the increasing models have fewer function calls than the decreasing ones. The reason is that the increasing models are started with smaller function calls at the initial generations and NFC gradually increases leading to the stop generation which is less than 100 in this paper. The increasing models never get the chance to engage with the highest-member generation, i.e. 1000 members at N_{Gen}=100, since before N_{Gen}=100 the stop criteria have been met. However, the decreasing models start with the 1000-member generation at N_{Gen}=1. That is why the NFC for the decreasing models is larger than the increasing ones.

Since the NFC in this paper, stands for the computational time, it can evidently be deduced that the decreasing population models are more time-consuming than the increasing schemes. However, this additional time, as indicated in Table 1 above, is actually the time required for the algorithm to generate more eligible members among the crowded initial generations, while in the increasing models, the members for the next generation should be selected and created from a sparse community in initial generations. As a result, the fitness values for the individuals in decreasing population models are much higher than the increasing models.

In the negative curvature decreasing model, the rate of decrease gradually fades off as N_{Gen} approaches 100. This is to ensure that more elite members are coming into play when 50% of the total generation number has been passed and the algorithm could not find the optimum solution. For the problems in which the algorithm has been converged to the final solution in the first half of the total generation number, the rate of decreasing the members is rather steep and the non-elite members are rapidly removed. For this reason, among the decreasing models, with a slightly higher NFC, has been converged to the highest fitness value member in the last generation.

8- CONCLUSION

Several population change patterns were examined in a genetic algorithm and compared with the classic fixed population model to study the effects of population change and pattern on the optimization result and convergence speed in a wind turbine section optimization process. The lift-to-drag ratio has been chosen as the fitness value for optimization. To avoid the time-consuming numerical simulations to determine the fitness value, a generalized regression neural network has been trained using a database from several numerical and experimental results provided either by the authors or those available in the literature. The network has then been employed to predict the lift-to-drag ratio for every 4-digit NACA airfoils. The results of the optimization operations using various population models in GA show that there is a small difference in convergence time between the fixed and variable population models and for a real problem with exact and heavy calculation of the fitness value, an increase in this time difference is anticipated. Amongst the various decreasing population patterns, the negative curvature decreasing one offers a good improvement in the optimized fitness value while increasing negligibly the convergence time. For the problems of this type, with remarkable convergence time, when a GA is chosen, a descending population model with positive curvature, similar to that occurs in a real-life, exhibits the highest performance and is suggested to be implemented in this algorithm.

NOMENCLATURE

c	Airfoil chord length
x	Airfoil longitudinal coordinate, measured from the apex
y	Airfoil lateral coordinate, measured from the apex
y_c	Mean camber line from x Axis
y_t	Thickness distribution from x axis
t	Airfoil thickness, expressed as a fraction of the chord
m	The Max. camber, expressed as a fraction of the chord
p	The longitudinal position of the Max camber, expressed as a fraction of the chord
a	Angle of attack
Re	Reynolds number, based on chord length
L/D	Airfoil Lift to Drag Ratio, the fitness value
N_{Gen}	The counter for the generation number

REFERENCES

[1] D.E. Goldberg, J.H. Holland, Genetic algorithms and machine learning, (1988).

[2] D.W. Bechert, Some Considerations on Measurements in Fluid Dynamics, Near-Wall Turbulent Flows, Elsevier Science B.V., New York, 1993.

[3] X. Chen, R. Agarwal, Optimization of FX, DU and NACA airfoils for wind turbine blades using a multi-objective genetic algorithm, in: 51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, 2013, pp. 782.

[4] P. Gage, I. Kroo, A role for genetic algorithms in a preliminary design environment, in: Aircraft Design, Systems, and Operations Meeting, 1993, pp. 3933.

[5] K. Yamamoto, O. Inoue, New evolutionary direction operator for genetic algorithms, AIAA journal, 33(10) (1995) 1990-1993.

[6] M. Anderson, The potential of genetic algorithms for subsonic wing design, in: Aircraft Engineering, Technology, and Operations Congress, 1995, pp. 3925.

[7] D. Doorly, J. Peiro, J.-P. Oesterle, Optimisation of aerodynamic and coupled aerodynamic-structural design using parallel genetic algorithms, in: 6th Symposium on Multidisciplinary Analysis and Optimization, 1996, pp. 4027.

[8] B. Bai, X. Li, H. Chen, Aerodynamic and Aeroacoustics Optimization Design of Multi-Element Airfoil by a Genetic Algorithm, in: 25th AIAA/CEAS Aeroacoustics Conference, 2019, pp. 2762.

[9] S. Darwish, M. Abdelrahman, A.M. Elmekawy, K. Elsayed, Aerodynamic Shape Optimization of Helicopter Rotor Blades in Hover using Genetic Algorithm and Adjoint Method, in: 2018 AIAA Aerospace Sciences Meeting, 2018, pp. 0044.

[10] C. Seager, R.K. Agarwal, Hypersonic blunt-body shape optimization for reducing drag and heat transfer, Journal of Thermophysics and Heat Transfer, 31(1) (2015) 48-55.

[11] K.R. Anderson, C. McNamara, T.J. Gross, A. Gatti, Multi-objective Genetic Algorithm Optimization of a Four-Stage Cascade Vapor Compression Refrigeration System, in: 2018 Joint Thermophysics and Heat Transfer Conference, 2018, pp. 3909.

[12] A. HACIOGLU, AUGMENTED GENETIC ALGORITHM WITH NEURAL NETWORK AND IMPLEMENTATION TO INVERSE AIRFOIL DESIGN, JOURNAL OF AERONAUTICS AND SPACE TECHNOLOGIES, 1(3) (2004) 1-7.

[13] X. Chen, R. Agarwal, Optimization of flatback airfoils for wind-turbine blades using a genetic algorithm, Journal of Aircraft, 49(2) (2012) 622-629.

[14] Y. Chen, K. Hord, R. Prater, Y. Lian, L. Bai, Design optimization of a vertical axis wind turbine using a genetic algorithm and surrogate models, in: 12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference and 14th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, 2012, pp. 5434.

[15] R. Duvigneau, M. Visonneau, Hybrid genetic algorithms and neural networks for fast CFD-based design, in: 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, 2002, pp. 5465.

[16] J.-L. Liu, Intelligent genetic algorithm and its application to aerodynamic optimization of airplanes, AIAA journal, 43(3) (2005) 530-538.

[17] W. Su, Y. Zuo, Z. Gao, Preliminary aerodynamic shape optimization using genetic algorithm and neural network, in: 11th AIAA/ISSMO multidisciplinary analysis and optimization conference, 2006, pp. 7106.

[18] Y.V. Pehlivanoglu, A. Hacıoglu, Inverse design of 2-D airfoil via vibrational genetic algorithm, Journal of Aeronautics and Space Technologies, 2(4) (2006) 7-14.

[19] A. Hacioglu, Fast evolutionary algorithm for airfoil design via neural network, AIAA journal, 45(9) (2007) 2196-2203.

[20] D.E. Goldberg, K. Deb, J.H. Clark, Genetic algorithms, noise, and the sizing of populations, COMPLEX SYSTEMS-CHAMPAIGN-, 6 (1992) 333-333.

[21] J. Arabas, Z. Michalewicz, J. Mulawka, GAVaPS-a genetic algorithm with varying population size, in: Proceedings of the First IEEE Conference on Evolutionary Computation. IEEE World Congress on Computational Intelligence, IEEE, 1994, pp. 73-78.

[22] V.K. Koumousis, C.P. Katsaras, A saw-tooth genetic algorithm combining the effects of variable population size and reinitialization to enhance performance, IEEE Transactions on Evolutionary Computation, 10(1) (2006) 19-28.

[23] S.A. Darani, O. Abdelkhalik, Convergence analysis of hidden genes genetic algorithms in space trajectory optimization, Journal of Aerospace Information Systems, 15(4) (2018) 228-238.
[24] J. McCosker, Design and optimization of a small wind turbine, Rensselaer Polytechnic Institute, Troy, NY, (2012).
[25] I.H. Abbott, E. Albert, von Doenhoff. Theory of Wing Sections, in, Dover Publications, Inc., New York, 1959.
[26] E.L. Houghton, P.W. Carpenter, Aerodynamics for engineering students, Elsevier, 2003.
[27] D.F. Specht, A general regression neural network, IEEE transactions on neural networks, 2(6) (1991) 568-576.

HOW TO CITE THIS ARTICLE

A. R. Davari, S. M. Pourkiaei. Variable Population Models in a Neural Network-Augmented Genetic Algorithm for Shape Optimization. AUT J. Mech Eng., 4(4) (2020) 435-448.

DOI: 10.22060/ajme.2020.16445.5819