

Research Article

Shape Optimization of NREL S809 Airfoil for Wind Turbine Blades Using a Multiobjective Genetic Algorithm

Yilei He and Ramesh K. Agarwal

Department of Mechanical Engineering and Materials Science, Washington University in St. Louis, St. Louis, MO 63130, USA

Correspondence should be addressed to Ramesh K. Agarwal; rka@wustl.edu

Received 27 February 2014; Revised 9 July 2014; Accepted 23 July 2014; Published 9 September 2014

Academic Editor: R. Ganguli

Copyright © 2014 Y. He and R. K. Agarwal. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The goal of this paper is to employ a multiobjective genetic algorithm (MOGA) to optimize the shape of a well-known wind turbine airfoil S809 to improve its lift and drag characteristics, in particular to achieve two objectives, that is, to increase its lift and its lift to drag ratio. The commercially available software FLUENT is employed to calculate the flow field on an adaptive structured mesh using the Reynolds-Averaged Navier-Stokes (RANS) equations in conjunction with a two-equation $k-\omega$ SST turbulence model. The results show significant improvement in both lift coefficient and lift to drag ratio of the optimized airfoil compared to the original S809 airfoil. In addition, MOGA results are in close agreement with those obtained by the adjoint-based optimization technique.

1. Introduction

With recent emphasis on emission-free renewable energy, wind energy has taken a center stage in recent years with exponential growth in deployment of wind-turbines worldwide. Among wind-turbines, horizontal-axis-wind-turbines (HAWTs) are mostly deployed for power generation in Megawatt range. It is well established that the power generated by a HAWT is a function of the number of blades, the C_l/C_d of the blade airfoil section, and the tip speed ratio λ (= rotational speed of the blade at tip/wind speed in free stream). Thus, one of the goals of the design of a wind turbine blade is to maximize its C_l/C_d . As a result, there has been significant effort devoted in recent years to shape optimization of wind turbine blade to achieve high C_l/C_d . In last two decades, aerodynamic shape optimization has become an important tool in aircraft design [1–4]. The focus of this paper is on the aerodynamic shape optimization of airfoil sections used in wind turbine blades since they affect their aerodynamic performance which in turn influences the amount of power a wind turbine can generate [5]. In modern wind-turbines, thick airfoils such as NACA-63XXX and NACA-64XXX are frequently used; however, new airfoil families are increasingly being developed because of multiple requirements of aerodynamics performance at

rated power conditions and off-rated power conditions as well as strong structural properties [6]. National Renewable Energy Laboratory (NREL) has developed a family of airfoils for HAWT applications [7] since 1984.

The present paper focuses on the optimization of most well-known NREL airfoil, known as the S809 airfoil. This airfoil is 21% thick laminar flow airfoil whose design and experimental results are given in [8]. NREL Phase II, Phase III, and Phase VI HAWT blades are composed of S809 airfoil from root to tip [9]. Under class 3 to 4 wind conditions, S809 is subjected to low Mach number (almost incompressible) flow with Reynolds numbers in the range of one to two million. Laminar separation can occur on the suction surface for angles of attack ranging from zero to 5.13 degrees. Turbulent trailing edge separation occurs when angle of attack increases [10].

This paper presents shape optimization of S809 airfoil using a multiobjective genetic algorithm (MOGA). The commercially available software FLUENT is used for calculation of the flow field using the Reynolds-Averaged Navier-Stokes (RANS) equations in conjunction with a two-equation $k-\omega$ SST turbulence model. Using MOGA, globally optimal S809 airfoil shape is obtained which maximizes both C_l and C_l/C_d for a given wind speed, rotational speed, and pitch setting. The results show that the aerodynamics characteristics of

optimized S809 are significantly improved. For the purpose of validation, the results are compared with those obtained by Ritlop and Nadarajah [10] using the adjoint equation-based optimization method.

2. Brief Description of Genetic Algorithm and Airfoil Parameterization

2.1. Single Objective Genetic Algorithm (SOGA). Genetic algorithms are a class of stochastic optimization algorithms inspired by the biological evolution. In GA, a set or generation of input vectors, called *individuals*, is iterated over, successively combining traits (aspects) of the best individuals until a convergence in the objective values (namely C_l or C_l/C_d) is achieved. In the context of the present paper, the individuals are airfoils which are generated in each generation of GA. These airfoils in a given generation are randomly generated with some constraints so that their thickness, camber, and other geometric properties do not vary a great deal from the original S809 airfoil. The population of airfoils in a given generation is selected by the user; a population of 20 airfoils is selected in this work. Execution of GA employs the following steps [11, 12].

- (1) Initialization: randomly create N individuals (airfoils).
- (2) Evaluation: evaluate the fitness of each individual (airfoil) by calculating the fitness or objective function (e.g., lift coefficient or lift to drag ratio).
- (3) Natural selection: remove a subset of the individuals. Often the individuals that have the lowest fitness value are removed; although culling, the removing of those individuals with similar fitness is sometimes performed.
- (4) Reproduction: pick pairs of individuals to produce an offspring. This is often done by roulette wheel sampling; that is, the probability of selecting some individual h_i for reproduction is given by

$$P[h_i] = \frac{\text{fitness}(h_i)}{\sum_j \text{fitness}(h_j)}. \quad (1)$$

A crossover function is then performed to produce the offspring. Generally, crossover is implemented by choosing a crossover point on each individual and swapping *alleles* or vector elements at this point as illustrated in Figure 1.

- (5) Mutation: randomly alter some small percentage of the population.
- (6) Check for convergence: if the solution has converged, return the best individual observed. If the solution has not yet converged, label the new generation as the current generation and go to step 2. Convergence is achieved after a certain number of generations when the fitness value does not change for a number of consecutive generations (generally 3 to 5).

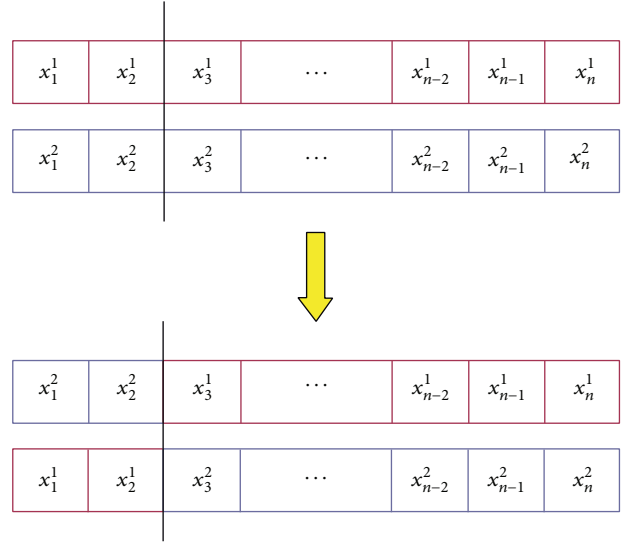


FIGURE 1: Illustration of the general crossover function in GA.

2.2. Multiobjective Genetic Algorithm (MOGA). For many design problems, it is desirable to achieve, if possible, simultaneous optimization of multiple objectives [13]. These objectives, however, are usually conflicting, preventing simultaneous optimization of each objective [14]. Therefore, instead of searching for a single optimal solution, a multiobjective genetic algorithm (MOGA) is necessary to find a set of optimal solutions (generally known as the set of Pareto-optimal solutions). A solution belongs to the Pareto set if there is no other solution that can improve at least one of the objectives without degrading any other objective. For Pareto-optimal solutions, any individual inside the Pareto set dominates any individual outside the set while any individual in the set is not dominated by another individual in the solution set. An individual dominating another individual in the Pareto sense means that this individual is either better than or the same as the other individual for all objectives and at least there is one objective or fitness function for which this individual is strictly better than the other individual. A solution is said to be Pareto optimal if and only if there does not exist another solution that dominates it. The set of all Pareto-optimal solutions is called the Pareto-optimal set. The MOGA used to find the Pareto-optimal solutions to the airfoil optimization problem in this study is widely known as NSGA-II. It has the following three features: (1) it uses an elitist principle, (2) it uses an explicit diversity preserving mechanism, and (3) it emphasizes nondominated solutions in a population [15]. The basic philosophy behind the NSGA-II algorithm is as follows.

First, the population is initialized based on the constraints. The population is then sorted based on nondomination into various fronts, the first front being completely the nondominant set in the current population and the second front being dominated by the individuals in the first front only and so on for the other fronts. Individuals (airfoils) in each front are assigned rank (fitness) values based on the front to which they belong. Individuals in first front are assigned

a given fitness value and individuals in the second front are assigned another fitness value and so on. In addition to the fitness value, a new parameter called *crowding distance* is calculated for each individual. The crowding distance is a measure of how close an individual is to its neighbors. Large average crowding distance usually results in better diversity in the population. Parents are selected from the population by using binary tournament selection based on the rank and the crowding distance. An individual selected in the rank is lesser than the other if the crowding distance is greater than the other. The selected population generates offsprings from crossover and mutation operators. The population with the current offsprings is sorted again based on nondomination and only the best N individuals are selected, where N is the population size. The selection is based on rank and on the crowding distance on the last front. The implementation procedure of NSGA-II is briefly as follows [16].

- (1) At 0th generation: a random parent population of airfoils P_0 of size N is created based on constraints; it is sorted based on the nondomination into various fronts, the first front being the completely nondominant set in the population and the second front being dominated by the individuals in the first front only and so on for the other fronts. Once the nondominated sorting is complete, the crowding distance is assigned. Since the individuals are selected based on the rank and crowding distance, all the individuals in the population are assigned a crowding distance value. Crowding distance is assigned frontwise. Once the individuals are sorted based on nondomination and with crowding distance assigned, the selection is carried out using a crowded-comparison-operator. P_0 is then sent to the selection, recombination, and mutation operators to create offspring population Q_0 of size N .
- (2) At t th generation: a combined population $R_t = P_t \cup Q_t$ of size $2N$ is formed and is sorted according to nondomination. Then again the individuals in R_t are divided into the best nondominated set F_1 , the next-best nondominated set F_2 , and so on. If the size of F_1 is smaller than N , all members of F_1 go to P_{t+1} , with the remaining members chosen from F_2, F_3, \dots until the size of P_{t+1} is N . It is accomplished as described for the 0th generation. Then the new population P_{t+1} is sent to selection, crossover, and mutation operators to create a new population Q_{t+1} of size N .
- (3) Termination: the procedure is terminated when convergence criterion is met. The convergence is considered achieved when the two objective values do not change from one generation to the other; generally, the algorithm termination condition is applied when no improvements are observed after a number of consecutive generations.

The java code package utilized in this study is called jMetal. It is a Java-based framework for multiobjective optimization using metaheuristics. It is easy to use and is flexible and extensible [17].

2.3. Airfoil Parameterization. The airfoil shapes are parameterized using Bezier curves. Bezier construction is a curve fitting method for constructing free-form smooth parametric curves which are widely used in CAE design data structure modelling and computer graphics application [18]. A Bezier curve is defined by a set of Bezier control points. Each curve can be expressed as polynomial equations containing the information of Bezier control points. The number of control points required to parameterize a curve depends on the shape of the curve.

Each airfoil is divided into top and bottom boundary curves by the airfoil chord joining its leading edge and trailing edge. Considering the complexity of S809 airfoil shape, 12 control points are used for parameterization. For an airfoil curve, two points are fixed since they represent the leading and trailing edge of the airfoil. The intermediate points are allowed to move within the specified boundaries. A maximum thickness constraint of 19%–22% of chord is imposed on the airfoil. The constraints applied to the Bezier control points are shown in Table 1.

3. The Shape Optimization of NREL S809 Airfoil

This section presents the optimization process for S809 airfoil using the multiobjective genetic algorithm (MOGA). An optimization procedure is established by coupling the MOGA code coupled with the mesh generation code “ICEM” and the CFD solver “FLUENT” as shown in Figure 2.

The individuals in each generation of MOGA are represented by a set of control points, which generate the airfoil shape through the Bezier curve. The mesh around the airfoil shape is generated using the grid generation software ICEM, which is used to create a two-dimensional structured or unstructured mesh as an input to CFD solver FLUENT. FLUENT is used to calculate the flow field for given flow conditions. Using the flow field data, FLUENT calculates the lift coefficient C_l and the drag coefficient C_d which are used to calculate the objective values for a given airfoil shape. Using the information about the objective values for all the airfoils in a given generation, MOGA is applied to create a next generation of airfoils and the process is repeated to obtain the Pareto front following the MOGA procedure outlined above. From the Pareto front approximation, Pareto front-based solution for the best objective values is obtained. The airfoil shape that corresponds to the best objective values is the final shape of the optimized airfoil.

3.1. Implementation of MOGA. NSGA-II [16] and the jMetal [17] multiobjective GA software packages are employed. We choose 20 individuals (airfoils) for each generation. The crossover rate of 0.9 is considered. The mutation rate is determined to be 1/24. jMetal MOGA framework offers multiple operators; here, we employ the simulated binary crossover (SBX) operator and the polynomial mutation operator for crossovers and mutations, respectively. The selection employs the binary tournament operator.

TABLE 1: Coordinates of the control points used in airfoil parameterization.

		Upper limit	Lower limit			Upper limit	Lower limit
Top boundary	x_1	0.020	0.000	Bottom boundary	m_1	0.020	0.000
	x_2	0.065	0.025		m_2	0.100	0.060
	x_3	0.340	0.300		m_3	0.300	0.260
	x_4	0.400	0.360		m_4	0.400	0.360
	x_5	0.500	0.460		m_5	0.480	0.440
	x_6	0.850	0.810		m_6	0.770	0.730
	y_1	0.030	0.010		n_1	-0.013	-0.025
	y_2	0.090	0.070		n_2	-0.053	-0.065
	y_3	0.128	0.108		n_3	-0.125	-0.145
	y_4	0.128	0.108		n_4	-0.125	-0.145
	y_5	0.128	0.108		n_5	-0.125	-0.145
	y_6	0.045	0.025		n_6	0.020	0.008

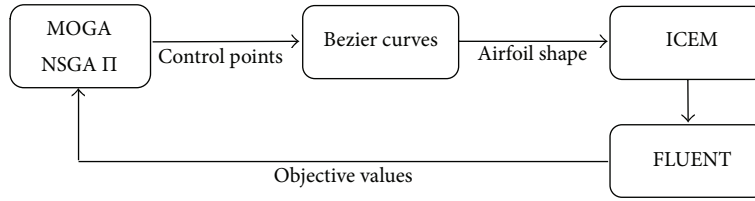


FIGURE 2: Schematic of information flow in the optimization process.

The multiobjective optimization algorithm is performed with two objective functions. The first objective is to minimize $10/C_l$ and the second objective is to minimize $100 * C_d/C_l$. The goal is to find the Pareto front for these two objective functions. When the value of both objective functions does not change for a number of consecutive generations (normally 3 to 5), the solution is considered converged to a Pareto-optimal approximation of C_l and C_l/C_d . The airfoil shape that corresponds to the optimal objective values is the final shape of the optimized airfoil.

3.2. Mesh Generation. The commercially available software “ICEM” is used to generate a structured C-mesh around the NREL S809 airfoil. A reply file is scripted to automatically generate mesh around different airfoils in a given generation. The reply file is edited to be able to generate mesh based on different airfoil shapes. Figure 3 shows a typical C-mesh around an airfoil. In this mesh, there are 58460 quadrilateral cells. Far field boundary is set at 20 chord lengths. Adaptive meshing is employed. The mesh is transported to the CFD solver FLUENT for calculation of the flow field.

3.3. Flow Field Computations. The commercial software FLUENT is employed to calculate the lift coefficient and the drag coefficient of an individual airfoil in a generation. Because of low Mach number $M = 0.1$, the convergence is very slow. It is due to the well-known fact that the time-marching flow solvers designed for compressible flows do not perform well at very low subsonic Mach numbers due to large condition number of the associated eigenvalue matrix [19]. To avoid

TABLE 2: Computed fully turbulent lift coefficients and the experimental lift coefficients.

AoA [deg.]	Comp. C_l	Exp. C_l
0.0	0.12529	0.07
2.1	0.35428	0.3
4.1	0.55472	0.55
6.1	0.75412	0.79
8.2	0.94169	0.9
10.1	1.0678	0.94
11.2	1.1046	0.93

the slow convergence, we multiply the free stream velocity by four which increases the Mach number but still keeps it within the low subsonic flow regime. To keep the Reynolds number unchanged, chord length of the airfoil is divided by four. First we validate our calculations by comparing them with the numerical results of Ritlop and Nadarajah [10] and the experimental data of Ramsay et al. [20]. The simulations are performed at Reynolds number of 1 million and Mach number of 0.044. Nine different angles of incidence are selected for validation, which are 0.0° , 2.1° , 4.1° , 6.1° , 8.2° , 10.1° , and 11.2° . The turbulence model employed is a two-equation k- ω SST turbulence model. Tables 2 and 3 and Figures 4 and 5 show that our computed results for lift and drag coefficient agree well with the experimental values [20] as well as the computations of Ritlop and Nadarajah [10]. These comparisons validate the numerical methodology used in our calculations.

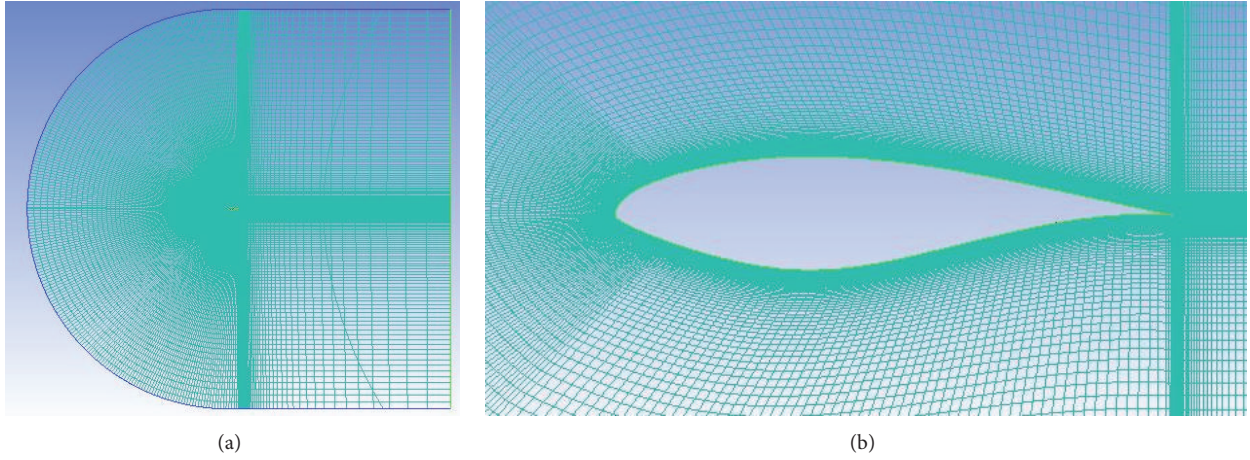


FIGURE 3: Structured C-grid around NREL S809 airfoil: (a) entire mesh in the computational domain and (b) zoomed-in view of the mesh near the airfoil.

TABLE 3: Computed fully turbulent drag coefficients and experimental drag coefficients.

AoA [deg.]	Comp. C_d	Exp. C_d
0.0	0.012006	0.0022
2.1	0.012824	0.0037
4.1	0.015248	0.005
6.1	0.017615	0.0063
8.2	0.021507	0.0096
10.1	0.027757	0.0231
11.2	0.03413	0.0236

It can be seen from Figure 4 that while the two sets of computational results for lift coefficient agree quite well with each other for angle of attack up to 11 deg., the experimental results agree with the computations only up to an angle of attack of 8 deg. There is separation observed in the experiment for AOA > 8 deg. which is not found in the calculations by two different investigators. The separation near the aft region of the airfoil in the experiment for AOA > 8 deg. is the reason for discrepancy between the computed and experimental values of the lift coefficient. In Figure 5, the computed and experimental values of the drag coefficient are compared. The drag is a very difficult quantity to compute as well as to measure experimentally. Nevertheless, the two sets of computations are in acceptable agreement. The discrepancy between the computed and experimental values of the drag coefficient is again due to the reasons explained above.

A journal file is written for autorunning of FLUENT in the optimization process. Temperature and static pressure are defined at standard sea level condition and are taken as 288.16 K and 101325 Pa, respectively. Both values are quite reasonable for a wind turbine whose maximum altitude does not exceed a few hundred meters. Density is taken as $\rho = 1.225 \text{ kg/m}^3$ and laminar viscosity is taken as $\mu = 1.7894 \times 10^{-5} \text{ kg/m}\cdot\text{s}$.

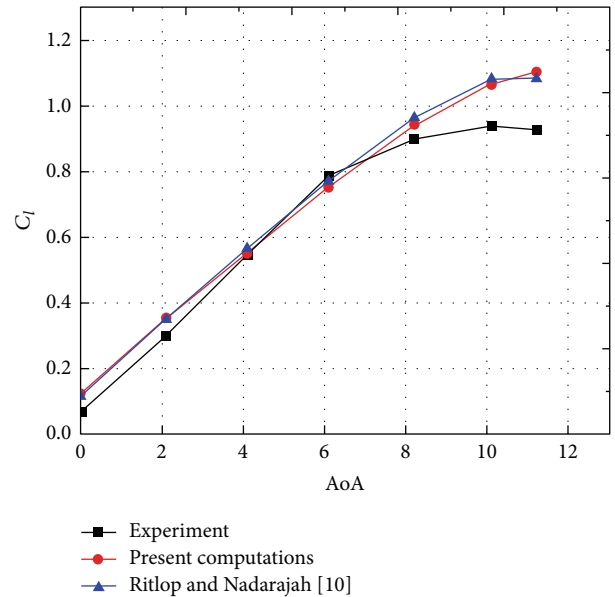


FIGURE 4: Variation of lift coefficient with angle of attack.

Wind-turbines generate power due to rotation of the blades. Blade Element Momentum (BEM) theory is used to determine the generated power [21]. The BEM theory is based on Glauert's propeller theory [22] which has been modified for application to wind-turbines [23]. In the present paper, we are simply interested in determining the relative velocity faced by an airfoil of the wind turbine as shown in Figure 6 from [23]. The mathematical expressions for axial velocity and blade velocity at radius r are given by the axial induction factor a and the radial or rotational induction factor a' , respectively. The relative velocity can be expressed as [23]

$$\begin{aligned} V_{\text{rel}} \sin \theta &= V_{\text{axial}} = V_0 (1 - a), \\ V_{\text{rel}} \cos \theta &= V_{\text{blade}} = \omega r (1 + a'), \end{aligned} \quad (2)$$

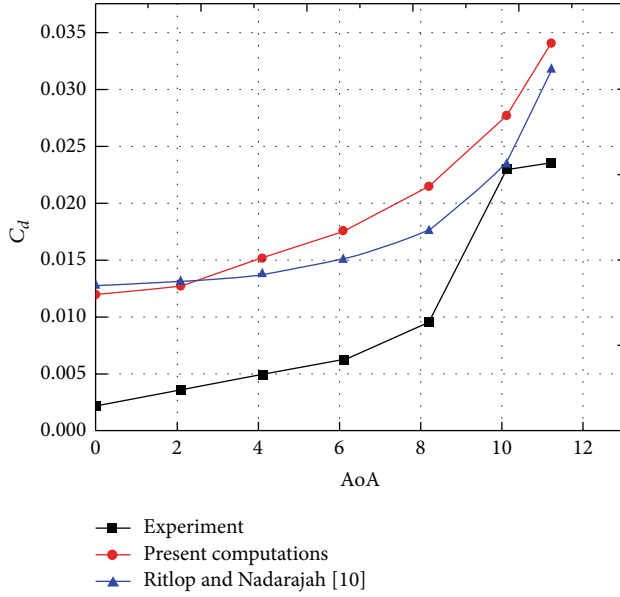


FIGURE 5: Variation of drag coefficient with angle of attack.

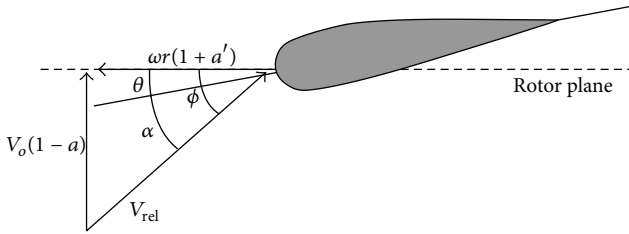


FIGURE 6: Velocities in the rotor plan of the wind turbine [22].

where ω is the angular velocity of wind turbine blade, r is the radial position of airfoil section, and V_0 is the free stream velocity.

Using the published data in [9], the effective angles of attack α_{eff} and the relative velocity V_{rel} for free stream velocity $V_0 = 6.7$ m/s, rotational speed $\omega = 72$ rpm, and pitch setting of 5 deg. are obtained as shown in Table 4.

4. Results and Discussion

As mentioned before, in the application of multiobjective genetic algorithm (MOGA), we optimize the S809 airfoil at three locations of the blade, 23%, 58%, and 98% locations from the center of the rotor which correspond to the root, mid, and tip section of the blade, respectively. We consider the free stream wind velocity of 6.7 m/s, rotational speed of 72 rpm, and pitch setting of 5 deg. We set two objectives: minimize $10/C_l$ and $100 * C_d/C_l$. The airfoil shape that results in the best possible lowest values of both objectives from the Pareto front approximation is considered the shape of the optimized airfoil. Figures 7, 8, and 9 show the evolution process of the airfoil shape and variation in lift to drag ratio at various generations of MOGA at 23%, 58%, and 98% span location, respectively. Figure 10 shows the dominant solutions in the Pareto front for the airfoil sections at three

TABLE 4: Effective angle of attack and relative velocity for $V_0 = 6.7$ m/s.

AoA	α_{eff}	V_{rel}
0.23	4.871484	10.95754347
0.28	5.33559	12.49505605
0.33	5.507533	14.12325328
0.38	5.695772	15.81257285
0.43	5.65495	17.54835289
0.48	5.481622	19.31840765
0.53	5.285769	21.11286633
0.58	5.113385	22.9245775
0.63	4.943813	24.74883296
0.68	4.799144	26.58353782
0.73	4.692116	28.4262618
0.78	4.619902	30.27473603
0.83	4.540881	32.12794747
0.88	4.399174	33.98431527
0.93	4.133288	35.84205712
0.98	3.397045	37.68558576

span locations. Figures 11, 12, and 13 show the comparison of airfoil shapes and pressure coefficients between the MOGA optimized airfoils and the original airfoils at three different span locations.

A comparison between the present optimized airfoil shapes using MOGA and those obtained by Ritlop and Nadarajah [10] using the adjoint method is shown in Figure 14. Tables 5 and 6 show the comparison of C_l and C_l/C_d for the present MOGA optimized airfoils with those optimized by Ritlop and Nadarajah [10] using the adjoint method. For the lift coefficient, the present results compare quite well with those in [10]. For the ratio C_l/C_d , the present results give a slightly higher value than those given in [10]. These results show that MOGA can give optimized airfoils that meet the two objectives of minimizing $10/C_l$ and $100 * C_d/C_l$; the results are close to those obtained by the adjoint method.

5. Conclusions

In this paper, a multiobjective genetic algorithm (MOGA) has been employed to optimize the shape of a well-known wind turbine airfoil S809 to improve its lift and drag characteristics, in particular to achieve two objectives, that is, to increase its lift and its lift to drag ratio. The commercially available software FLUENT is employed to calculate the flow field on an adaptive structured mesh using the Reynolds-Averaged Navier-Stokes (RANS) equations in conjunction with a two-equation $k - \omega$ SST turbulence model. The results show significant improvement in both the lift coefficient and lift to drag ratio of the optimized airfoil compared to the original S809 airfoil. In addition, the present results are in close agreement with those obtained by the adjoint-based optimization technique [10].

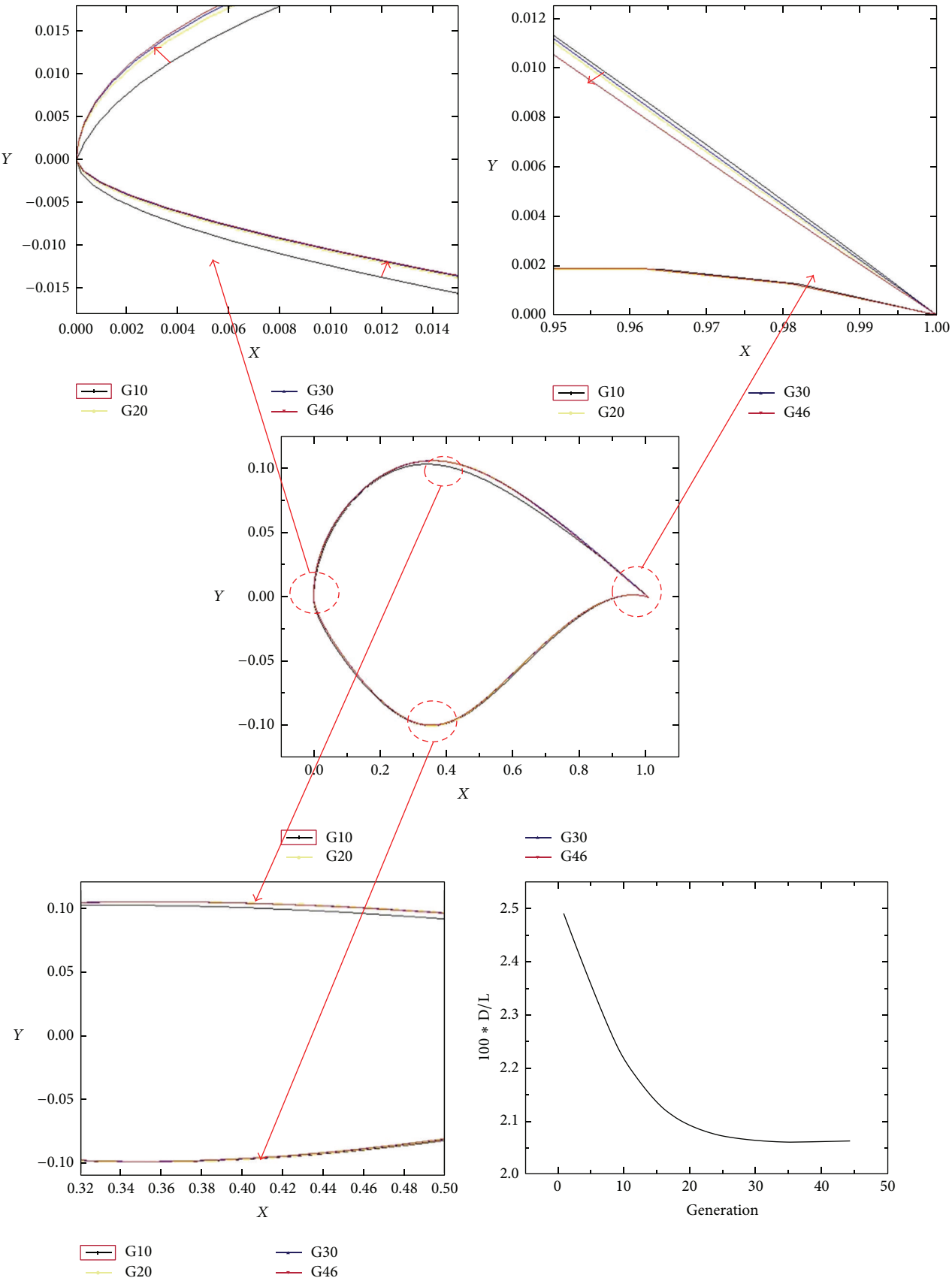


FIGURE 7: Shape evolution of airfoil and variation in drag to lift ratio at 23% span location in various generations of MOGA.

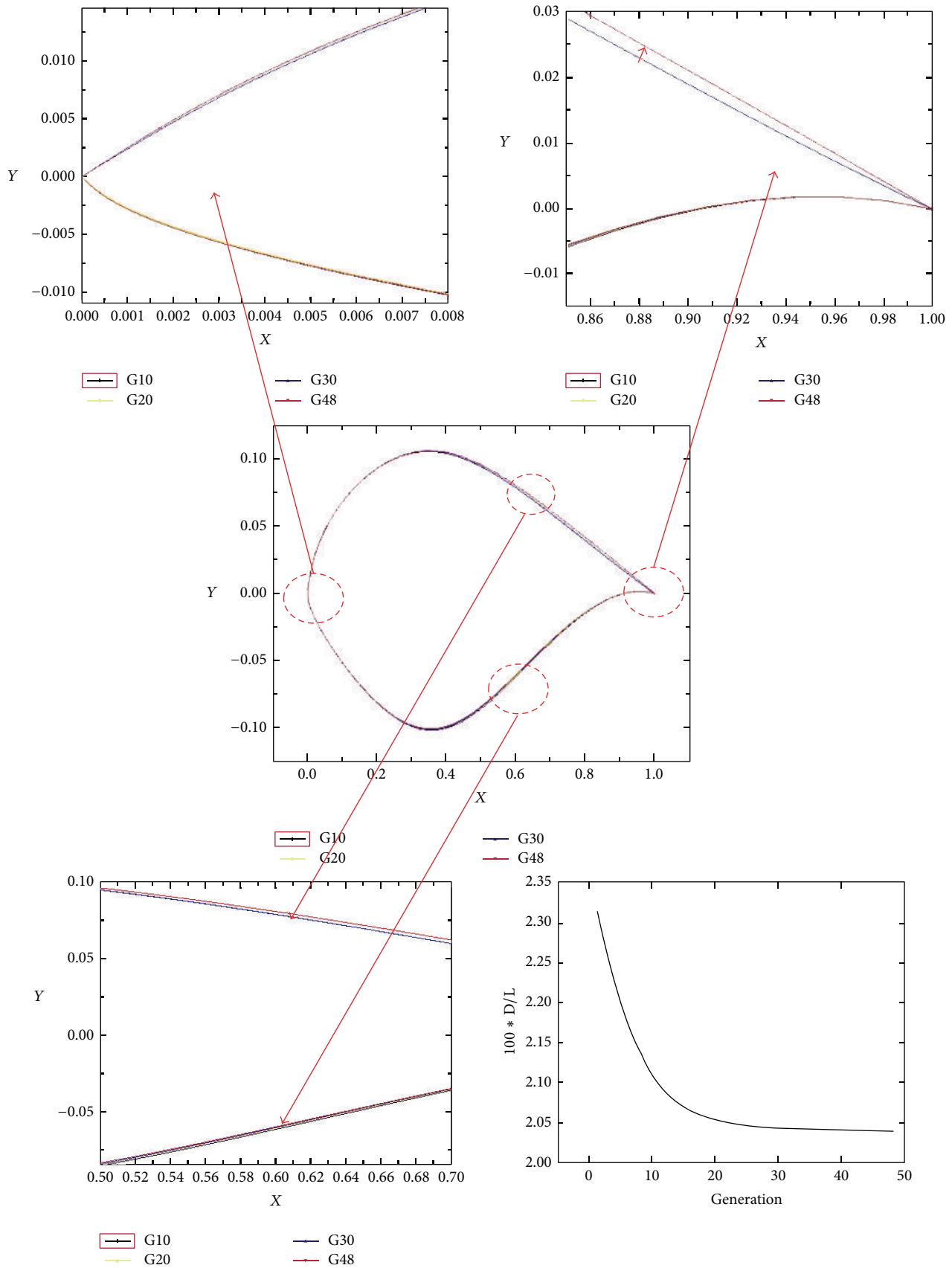


FIGURE 8: Shape evolution of airfoil and variation in drag to lift ratio at 58% span location in various generations of MOGA.

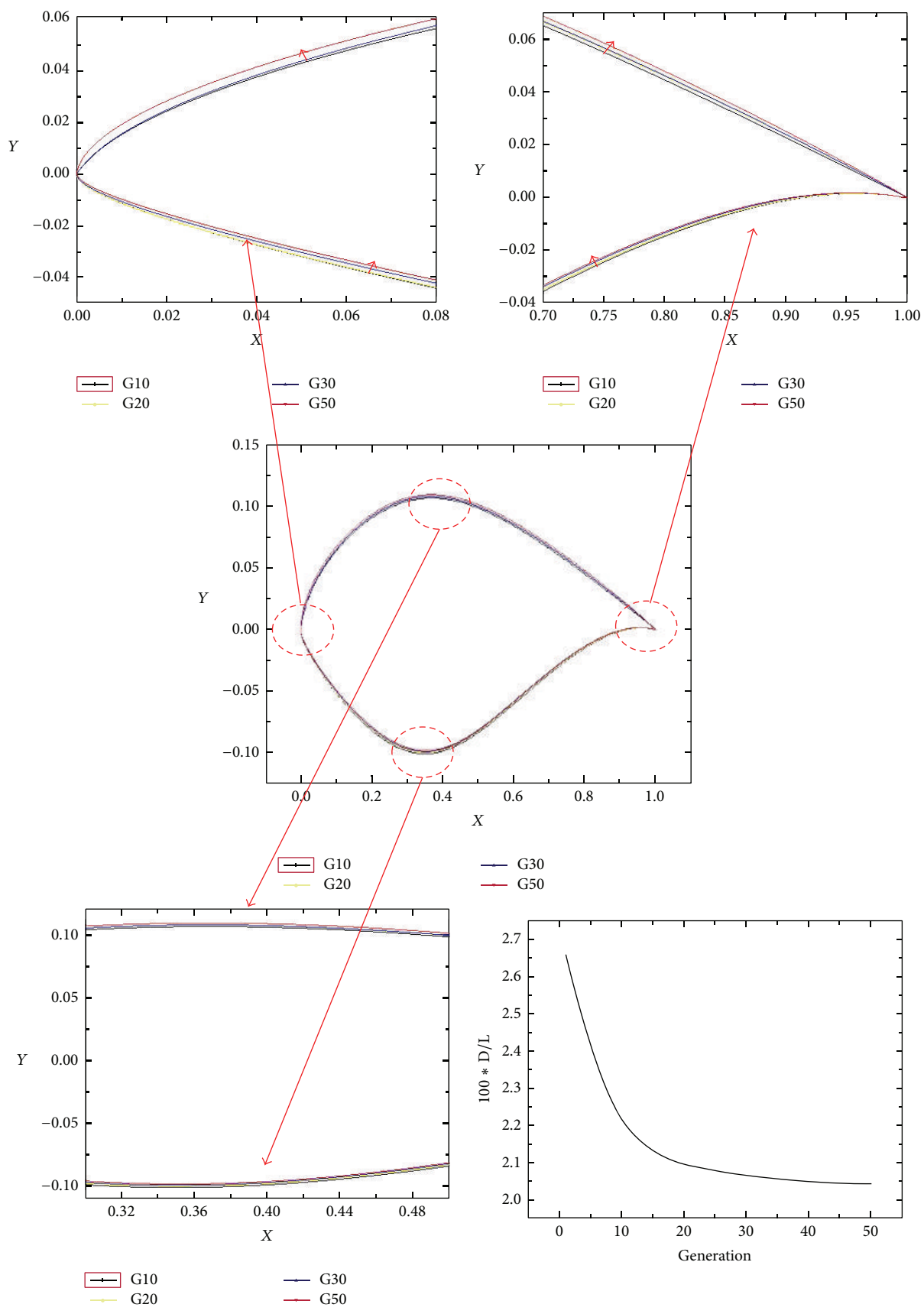


FIGURE 9: Shape evolution of airfoil and variation in drag to lift ratio at 98% span location in various generations of MOGA.

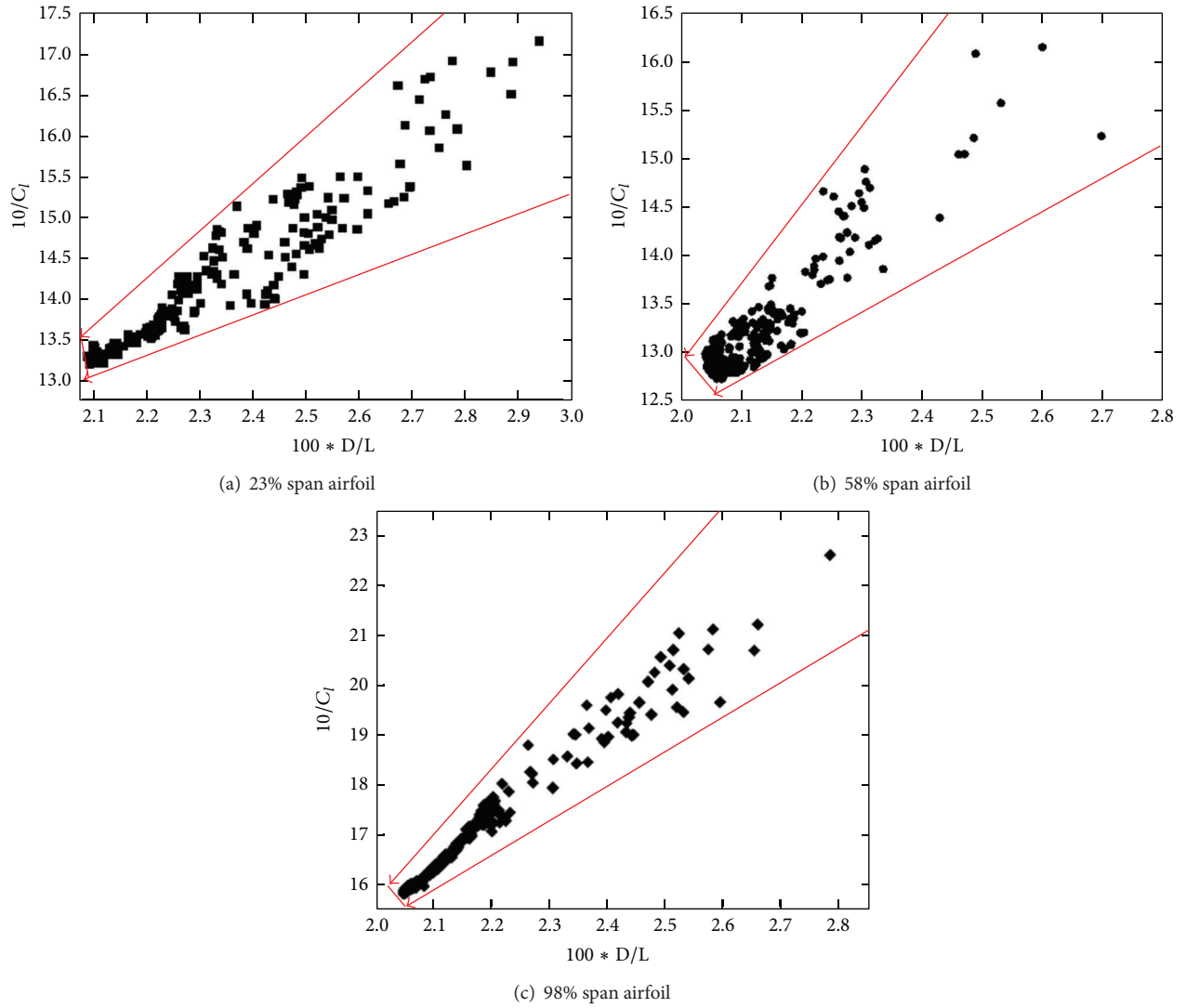


FIGURE 10: Dominated solutions in Pareto front approximation for airfoils at 23%, 58%, and 98% span locations.

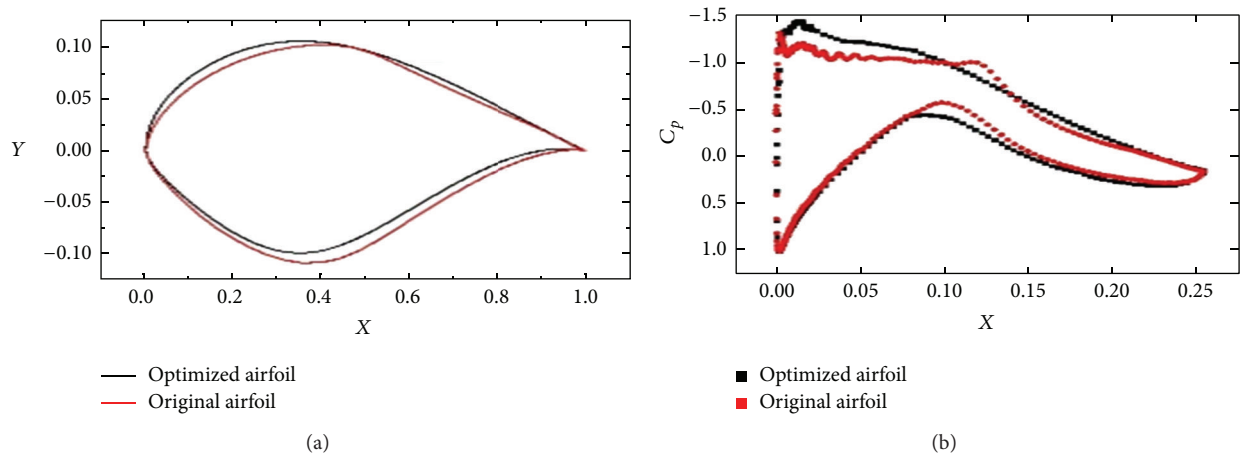


FIGURE 11: Comparison of original airfoil and optimized airfoil (23% span): (a) shape and (b) pressure distribution.

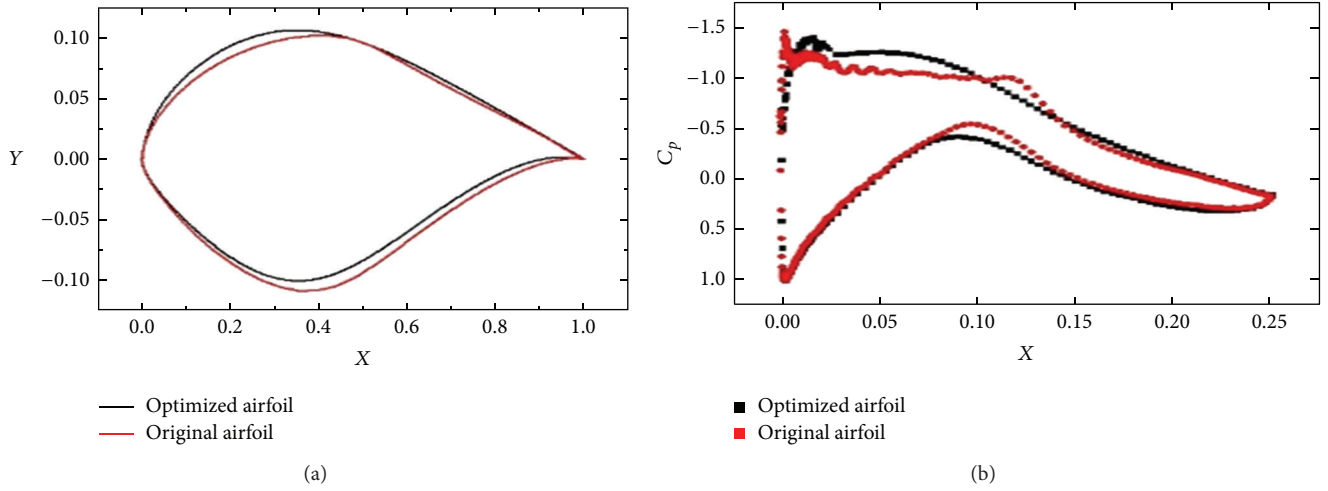


FIGURE 12: Comparison of original airfoil and optimized airfoil (58% span): (a) shape and (b) pressure distribution.

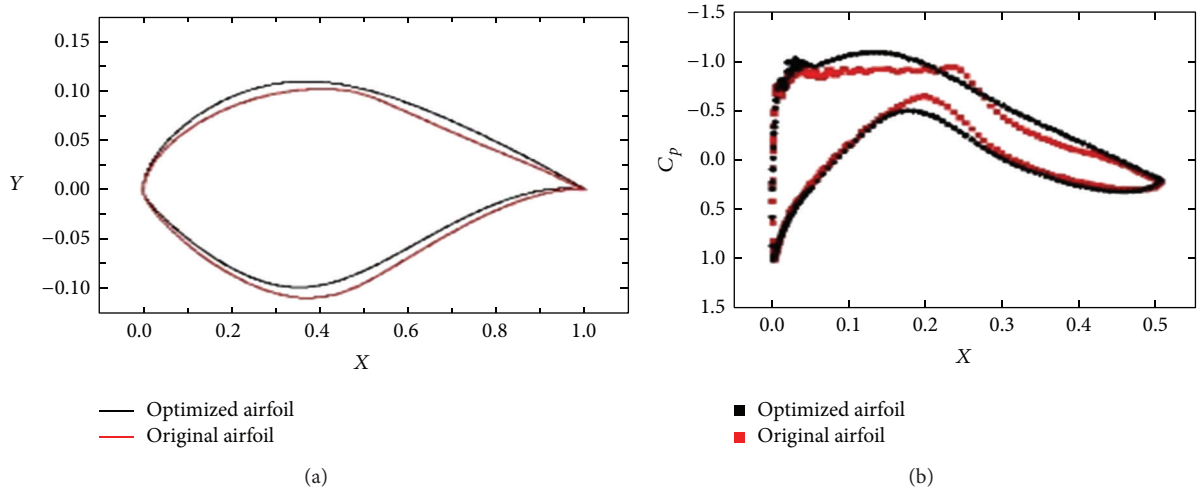


FIGURE 13: Comparison of original airfoil and optimized airfoil (98% span): (a) shape and (b) pressure distribution.

TABLE 5: Comparison of C_l of optimized airfoils: present results and those of Ritlop and Nadarajah [10].

	Present results at 23% span	Ritlop and Nadarajah [10] results at 23% span	Present results at 58% span	Ritlop and Nadarajah [10] results at 58% span	Present results at 98% span	Ritlop and Nadarajah [10] results at 98% span
Original C_l	0.64528	0.64	0.68023	0.675	0.492	0.49
Optimal C_l	0.7461	0.76	0.78274	0.78	0.63	0.68
C_l improved by	15.62%	18.75%	15.06%	15.55%	28.04%	38.78%

TABLE 6: Comparison of C_l/C_d of optimized airfoils: present results and those of Ritlop and Nadarajah [10].

	Present results at 23% span	Ritlop and Nadarajah [10] results at 23% span	Present results at 58% span	Ritlop and Nadarajah [10] results at 58% span	Present results at 98% span	Ritlop and Nadarajah [10] results at 98% span
Original C_l/C_d	40.17289	45.5	43.22983	47	37.77	36.2
Optimal C_l/C_d	48.48394	50.8	49.01435	51.8	48.89	47
C_l/C_d improved by	20.688%	13%	13.38%	10%	29.44%	29.83%

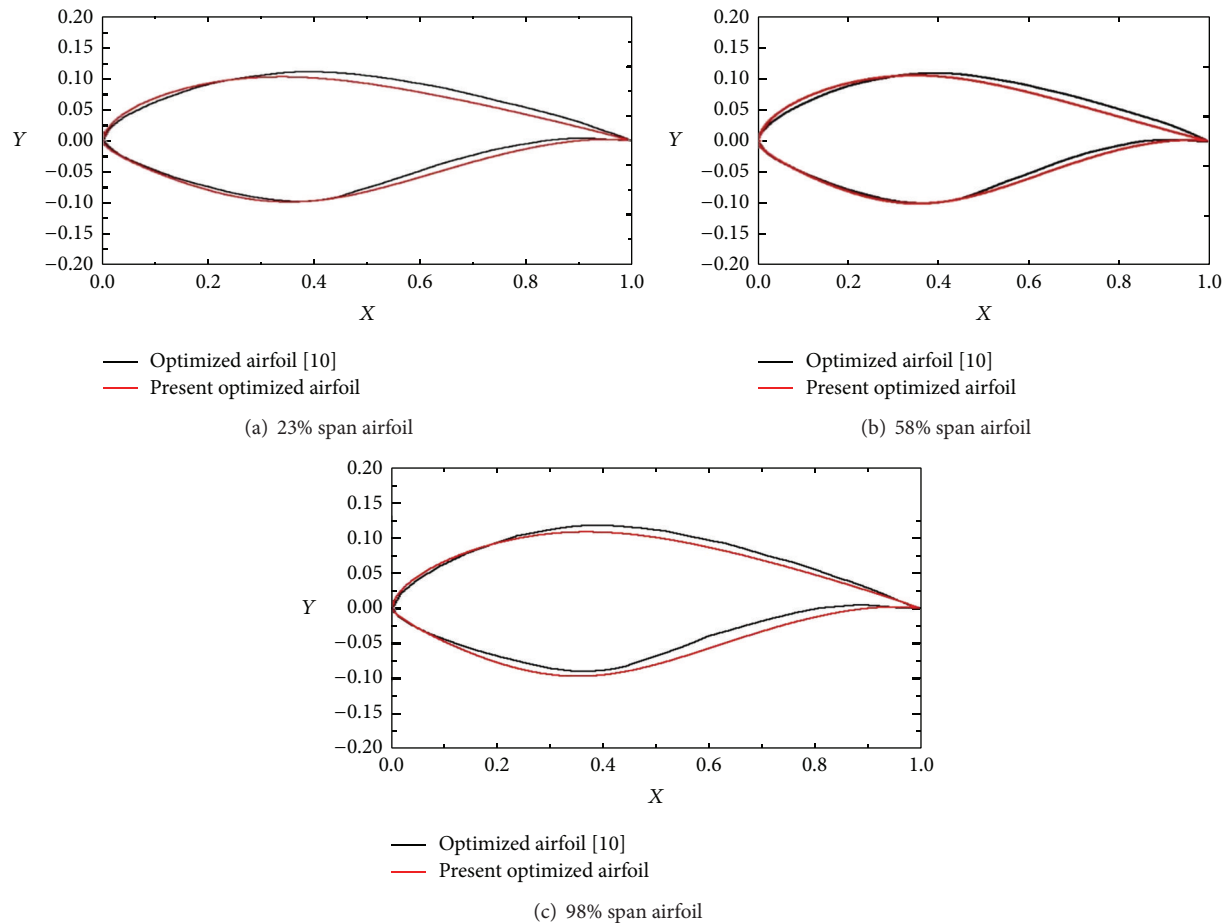


FIGURE 14: Comparison of present optimized airfoil shapes with those of Ritlop and Nadrajah [10].

Conflict of Interests

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

References

- [1] E. A. Leylek, J. E. Manzo, and E. Garcia, "Bat-inspired wing aerodynamics and optimization," *Journal of Aircraft*, vol. 47, no. 1, pp. 323–328, 2010.
- [2] D. Nelson, "Numerical optimization of airfoils in low Reynolds number flows," *Journal of Aircraft*, vol. 46, no. 1, pp. 331–337, 2009.
- [3] D. J. Pate, M. D. Patterson, and B. J. German, "Optimizing families of reconfigurable aircraft for multiple missions," *Journal of Aircraft*, vol. 49, no. 6, pp. 1988–2000, 2012.
- [4] R. Hu, A. Jameson, and Q. Wang, "Adjoint-based aerodynamic optimization of supersonic biplane airfoils," *Journal of Aircraft*, vol. 49, no. 3, pp. 802–814, 2012.
- [5] H. A. M. Perez-Blanco, "Aerodynamic performance of preferred wind turbine airfoils," in *Proceeding of ASME Turbo Expo: Power for Land, Sea and Air*, 2012.
- [6] F. Grasso, "Usage of numerical optimization in wind turbine airfoil design," *Journal of Aircraft*, vol. 48, no. 1, pp. 248–255, 2011.
- [7] J. L. Tangler and D. M. Somers, "NREL airfoil families for HAWTs," in *Wind Power*, pp. 117–123, Washington, DC, USA, 1995.
- [8] D. M. Somers, "Design and Experimental Results for the S809 Airfoil," NREL/SR-440-6918, January 1997.
- [9] P. Giguère and M. S. Selig, "Design of a rapered and rwisted blade for the NREL combined experiment rotor," Subcontractor Report NREL/SR-500-26173, 1999.
- [10] R. Ritlop and S. K. Nadarajah, "Toward the aerodynamic shape optimization of wind turbine profiles," in *Proceedings of the 17th Annual Conference of the CFD Society of Canada*, Ottawa, Canada, 2009.
- [11] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, 1989.
- [12] B. Morgan, "Gairfoils: finding high-lift Joukowski airfoils with a genetic algorithm," Tech. Rep., Department of Mechanical Engineering, Washington University in St. Louis, 2007.
- [13] N. Srinivas and K. Deb, "Muultiobjective optimization using nondominated sorting in genetic algorithms," *Evolutionary Computation*, vol. 2, no. 3, pp. 221–248, 1994.

- [14] A. Konak, D. W. Coit, and A. E. Smith, "Multi-objective optimization using genetic algorithms: a tutorial," *Reliability Engineering and System Safety*, vol. 91, no. 9, pp. 992–1007, 2006.
- [15] K. Deb, "Single and multi-objective optimization using evolutionary computation," in *Proceedings of the 6th International Conference on Hydro-Informatics*, pp. 14–35, Singapore, 2004.
- [16] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [17] J. J. Durillo, A. J. Nebro, and E. Alba, "The jMetal framework for multi-objective optimization: design and architecture," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '10)*, pp. 4138–4325, July 2010.
- [18] M. A. Zaman and S. Chowdhury, "Modified Bézier curves with shape-preserving characteristics using differential evolution optimization algorithm," *Advances in Numerical Analysis*, vol. 2013, Article ID 858279, 8 pages, 2013.
- [19] J. M. Weiss and W. A. Smith, "Preconditioning applied to variable and constant density flows," *AIAA Journal*, vol. 33, no. 11, pp. 2050–2057, 1995.
- [20] R. R. Ramsay, M. J. Hoffmann, and G. M. Gregorek, "Effects of Grit Roughness and Pitch Oscillations on the S809 Airfoil," NREL/TP-442-7817, 1995.
- [21] R. Lanzafame and M. Messina, "Fluid dynamics wind turbine design: critical analysis, optimization and application of BEM theory," *Renewable Energy*, vol. 32, no. 14, pp. 2291–2305, 2007.
- [22] H. Glauert, *The Elements of Airfoil and Airscrew Theory*, Cambridge University Press, Cambridge, UK, 1926.
- [23] J. F. Manwell and J. G. McGowan, *Wind Energy Explained: Theory, Design and Application*, Wiley, New York, NY, USA, 2010.

