Global Airfoil Shape Optimization with Swarm Intelligence

Chawla K. [a]
karangc2@illinois.edu

University of Illinois at Urbana-Champaign, 61820

A robust airfoil optimization system is designed using the Particle Swarm Optimization scheme, which consists of an efficient implementation of the optimization algorithm, high accuracy airfoil parametrization methods: Ferguson Spline and the PARSEC method and, a high fidelity interface for the aerodynamic analysis program, XFoil. Case studies representing the most common airfoil shape optimization problems are discussed and the resulting optimized airfoils are analyzed.

[a] Graduate Student, AE 416 - Applied Aerodynamics

Nomenclature

α = angle of Attack

C_l = coefficient of lift

C_d = coefficient of drag

Δ = difference operator

i = iteration number

LE = leading edge

max = maximum

min = minimum

PSO = Particle Swarm Optimization

Re = Reynolds number

TE = trailing edge

T_A = tangent magnitude at point A

T_B = tangent magnitude at point B

α_b = orientation of T_B^{lower}

α_c = orientation of T_B^{upper}

I. Introduction

Researchers have been working for decades to design algorithms that can augment an airfoil designer's knowledge of aerodynamics and expedite the process of airfoil optimization. Optimization plays a crucial role in airfoil design because of the existence of a plethora of competing design variables. One of the most important factor being the wing profile, which has perhaps the maximum influence on the aircraft in terms of it's performance, stability and overall aerodynamic characteristics. Wing profiles are chosen for a particular flight regime and operating conditions; a large jetliner such as Boeing 747 will have a different airfoil as compared to a fighter jet such as F-35B designed for rapid aerial maneuvers. Multidisciplinary optimization is used to size different components of modern aircraft but this is outside the scope of this project. In order to limit the scope of the project, only airfoil shape is considered for the optimization problem.

Great leaps of progress has been made in airfoil design with the advent of computational tools which allow the designer to specify aerodynamic performance goals. Most of the times, these goals are to improve a measure of efficiency such as maximizing $\frac{C_l}{C_d}$ to increase range, minimizing C_d for a certain cruise speed or a cruise C_l etc. Airfoil design is an amalgamation of art, designer's knowledge of aerodynamics and optimization carried out using mathematical modeling tools on rapid computers. A range of methods are employed in this process such as, inverse design to meet a certain pressure distribution over the airfoil or optimization algorithms driving the design to minimize a cost function. In this project a variety of single and multi-variable cost functions are considered to showcase the versatility of the optimization scheme developed.

The searching efficiency and precision of the optimization process is defined by the response time of the aerodynamic analysis program and the optimization algorithm. This project explores the application of Particle Swarm Optimization (PSO) in the domain of airfoil shape optimization. It is a global optimization scheme that was inspired by the flocking behavior in birds and is described in depth in section III. Due to its high searching efficiency and simplicity of application, it is applied to a range of problems in areas like Artificial Intelligence to train Neural Networks, parameter identification etc. PSO is a random algorithm that is extremely robust to local minima in global optimization problems due to the large number of particles that make up the swarm.

Section II describes the airfoil design and parametrization schemes that were implemented for the project. These include Ferguson Spline based parametrization and PARSEC method.

Section III describes the PSO algorithm in detail and its implementation to the airfoil shape optimization problem.

Section IV discusses the aerodynamic analysis software that was used in conjugation with the optimization scheme and gives an architectural overview of the complete optimization process that was implemented.

Section V discusses the case studies that were developed to showcase the robustness and the versatility of the optimization scheme. It also discusses the results that were thus obtained.

II. Airfoil Design

Airfoil Parametrization is one of the most widely studied topics since the advent of manned flight. The first step in optimizing an airfoil is selecting the design variables and determining the relationships between those variables and the upper and lower surfaces of the airfoil. These contours are then used at a later design stage to determine the aerodynamic behavior of the profile.

Over the past few decades researchers have come up with a range of airfoil parametrization approaches including the acclaimed NACA definitions, PARSEC polynomials, nth order polynomials etc. Kulfan and Bussoletti (Ref. [10]) came up with a list of eight desirable features an airfoil parametrization technique should have, and they are as follows:

1. Smooth and well behaved
2. Mathematically efficient
3. Parsimonious in terms of number of design variables
4. Allow for specification of key features like leading edge radius etc.
5. Easy control for editing
6. Intuitive geometric representation
7. Systematic and consistent
8. Robust

A. Ferguson's Spline Formulation

Cubic spline formulation was invented by James Ferguson in 1964 and since then has become quite popular especially in engineering design, computer graphics and statistics. Although Bezier and others have subsequently come up with slightly different and in some cases, superior formulations, the logic behind them remains the same.

Parametrizing airfoils using Ferguson's curves allows us to implement them in modern CAD software as argued by Sobester and Keane in Ref. [1]. Though this method makes a few compromises on the aforementioned eight principles, it improves the applicability of the technique in early design optimization, as will be shown in Chapter III.

The aim of the formulation is to design a curve $r(u)$ with $u \in [0,1]$, which connects two points $r(0) = A$ and $r(1) = B$. Each end point is constrained with a particular magnitude and direction of the tangent at that point: $\frac{dr}{du}|_{u=0} = T_A$ and $\frac{dr}{du}|_{u=1} = T_B$ as shown in Fig. 1. The curve is defined as:

$$r(u) = \sum_{i=0}^{3} a_i u^i, u \in [0,1] \tag{1}$$

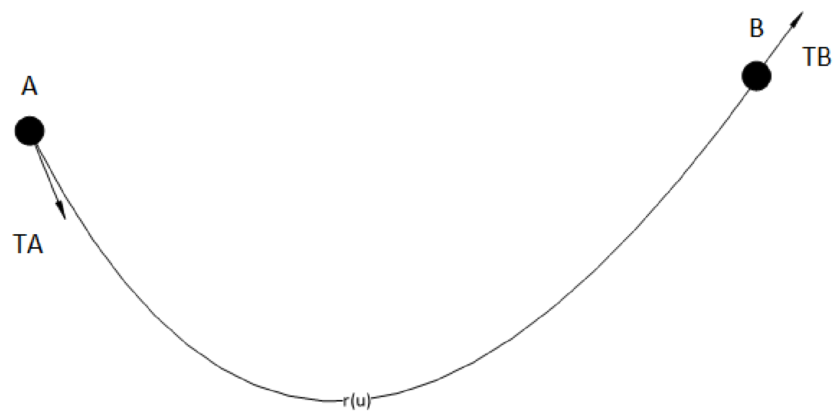

Fig. 1 Ferguson spline and its boundary conditions

After setting the aforementioned boundary constraints, we obtain the following equation:

$$r(u) = A(1 - 3u^2 + 2u^3) + B(3u^2 - 2u^3) + T_A(u - 2u^2 + u^3) + T_B(-u^2 + u^3) \tag{2}$$

Eqn. 2 is essentially a Hermitian interpolant of the bracketed terms in the equation, known as the basis functions. Eqn. 2 can be written in a more readable form as follows:

$$r(u) = \begin{bmatrix} 1 & u & u^2 & u^3 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -3 & 3 & -2 & -1 \\ 2 & -2 & 1 & 1 \end{bmatrix} \begin{bmatrix} A \\ B \\ T_A \\ T_B \end{bmatrix} \tag{3}$$

B. Airfoil Parametrization

Researchers have been describing airfoils using splines since decades, this approach is not new. However, most of these methods involve the definition of points, which are then interpolated by a spline. This process though widely used for airfoil definition is not ideal for airfoil optimization; as each of the individual points add a degree of freedom to the optimization problem. This, thus increases the computational requirements and for a single-variable optimization increases the complexity by k^n where k is the sampling locations in an n dimensional space.

With the use of Ferguson splines, the number of design variables can be reduced to six. Each airfoil can be defined by two splines, one for the upper surface and one for the lower surface. T_A^{upper} and T_A^{lower} represent the magnitude of tangents at the leading edge of the airfoil, for the upper and lower spline respectively. Similarly, T_B^{upper} and T_B^{lower} define the magnitude of tangent vectors at the trailing edge. Since, the direction of T_A^{upper} and T_A^{lower} is always $\pi/2$, thus controlling the "bluntness" of the leading edge. α_c determines the orientation of T_B^{lower} and, α_b determines the orientation of T_B^{upper} with respect to T_B^{lower}.

This method offers enough control to design realistic airfoils for the optimization process and reduces the computational requirements for the optimization process. The only drawback of this method is that it diminishes designer's control on the local airfoil shape, as this method influences the overall shape of the airfoil (Ref. [1]).

C. PARSEC Method for Airfoil Parametrization

PARSEC is one of the most common techniques used for airfoil parametrization in modern airfoil design and optimization. It uses eleven basic parameters to define the airfoil shape as opposed to six, used in Ferguson's spline based parametrization. Though, this increases the computational complexity of the optimization process, it offers better control over the airfoil geometry to the designer.

The various parameters are LE radius(r_{LE}), upper crest location(X_{UP}, Z_{UP}), lower crest location (X_{LO}, Z_{LO}), upper and lower curvature(Z_{xxUP}, Z_{xxLO}), trailing edge coordinate(Z_{TE}) and direction(ATE), trailing edge wedge angle(β_{TE}) and thickness(ΔZ_{TE}). **??** In this method, the airfoil shape is described using a linear combination of shape functions. Similar to NACA 4 digit series, a polynomial is used for the airfoil shape description, though of 6^{th} order.

$$Z_k = \sum_{n=1}^{6} a_{n,k} X_k^{\frac{n-1}{2}} \tag{4}$$

Coefficients, a_n are determined with the help of geometric parameters and the subscript k takes values from 1 and 2 for upper and lower surface respectively.

In order to make significant contributions to airfoil design/optimization process, a parametrization method must be able to represent a wide array of airfoils. A meta-optimizer is used to optimize the parameters for the parametrization process, such that the shape obtained matches the original airfoil shape. Choosing only this as the sole metric of a parametrization technique, PARSEC performs better than Ferguson's spline technique as it is able to represent a larger number of airfoils as compared to the former method.

III. Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a swarm based stochastic optimization technique developed by Dr. Eberhart and Dr. Kennedy inspired by the evolution of a group or swarm traveling the search space and exchanging information (schools of fish, flocks of birds, etc.)

PSO has a lot of similarities as compared to other evolutionary optimization techniques such as Genetic Algorithm, gradient decent search etc. However, unlike Genetic Algorithm (GA) PSO doesn't have evolution operators such as mutation, crossover etc. Compared to GA, PSO is easier to implement and has fewer parameters to tune (Ref. [8]).

In the PSO scheme, the system is initialized with random solutions and the algorithm searches for the optimum solution over a number of iterations, by using social dynamics of the flock, known as particles.

In general, the PSO algorithm consists of the following three main steps(Ref. [9]):

1. Generating the positions and velocities of particles,
2. Updating the velocities, and
3. Changing the positions.

Each particle consists of:

1. Data representing a possible solution in the n dimensional solution space.

2. A velocity value, synonymous to the magnitude with which the data can be altered.

3. A personal best (*pBest*) value indicating the closest the particle's Data has ever come to the Optima.

The Data stored in each particle can be anything and depends on the optimization problem. For the case of airfoil shape optimization, the data is the airfoil shape expressed as airfoil shape parameters(derived from parametrization). Each point on the airfoil is manipulated (under certain constraints) such that the Cost Function is optimized.

The velocity value is calculated from how far the data is from the global best (*gBest*), which is the closest a particle has ever come to the optimum. The further the particle's data is from *gBest*, the larger the velocity value. Particles' velocity in each dimension is clamped to maximum and minimum velocity. If the velocity assigned to the particle after each iteration is greater than maximum possible velocity, then the particle velocity is limited to V_{max}. Fig. 2 shows how *gBest* and *pBest* affect the position of a particle, in an iteration.

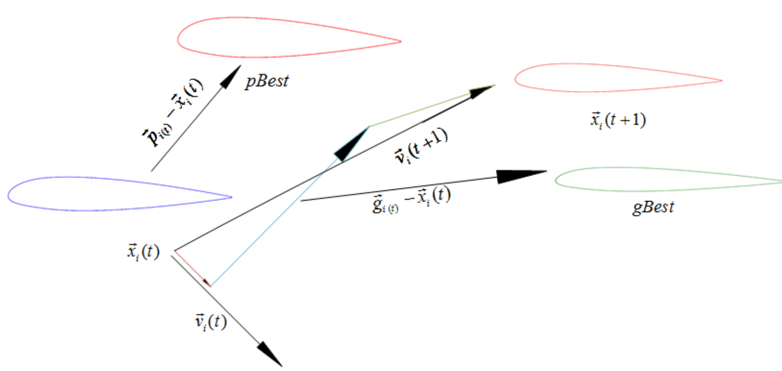

Fig. 2 Resultant vector for position change of particle

The coefficients$w$, c_1 and c_2 indicate the effect of the current motion, the particle's best (memory) and the global best (swarm intelligence), respectively. According to the explanation, the velocity update formula takes the following form:

$$v^i(t+1) = w \times v^i(t) + c_1 \times rand() \times [P^i - x_t^i] + c_2 \times rand() \times [p_t^g - x_t^i] \tag{5}$$

Where w is the inertia factor in the range of 0.5 to 1.5, $v_i(t)$ is the velocity of the particle in the current motion, $c_1$is the self-confidence factor in the range of 1.5 to 2.0, c_2 is the swarm confidence factor in the range of 2.0 to 2.5, $x_i(t)$ is the position of the i^{th} particle in the current iteration, P_i is the personal best of i^{th}particle and p_t^g is the global best of the swarm. Using the updated velocity vector as calculated in the equation, the position of each particle is updated according to the following equation:

$$x^i(t+1) = x^i(t) + v^i(t+1) \tag{6}$$

The $gBest$ value is only altered when any particle's $pBest$ is better than the $gBest$ of the swarm. Over the iterations, the $gBest$ gradually moves closer to the optimum solution until one of the particle reaches the optimum.

This algorithm is repeated until a stop criterion is reached as shown in Fig. 3. The stop criterion may be a maximum number of epochs or a predefined tolerance value for the best global position.

IV. Airfoil Analysis with XFoil

For the optimization scheme to work effectively, a reliable solver must be employed to analyze the airfoil profiles. XFoil is widely renowned for its accuracy and ease of use in incompressible airfoil design and analysis (Ref. [7]). XFoil is a freely distributed multi-variable aerodynamic analyses package that uses airfoil geometry input directly using a set of points via a text file. It can be also be controlled via Python using a simple script. The aforementioned functionality make it ideal for our goals.

Aerodynamic analysis of the given airfoil is carried out in viscous mode using a modified panel method to determine the aerodynamic coefficients of lift, drag, moments at user-specified test conditions of angle of attack (α) and Reynolds number(Re). Xfoil consists of a range of menu-driven routines controlled via DOS like command line input. Outputs of the aerodynamic coefficients may be written to a text file for storage or external manipulation. Figure 4 shows the architec-

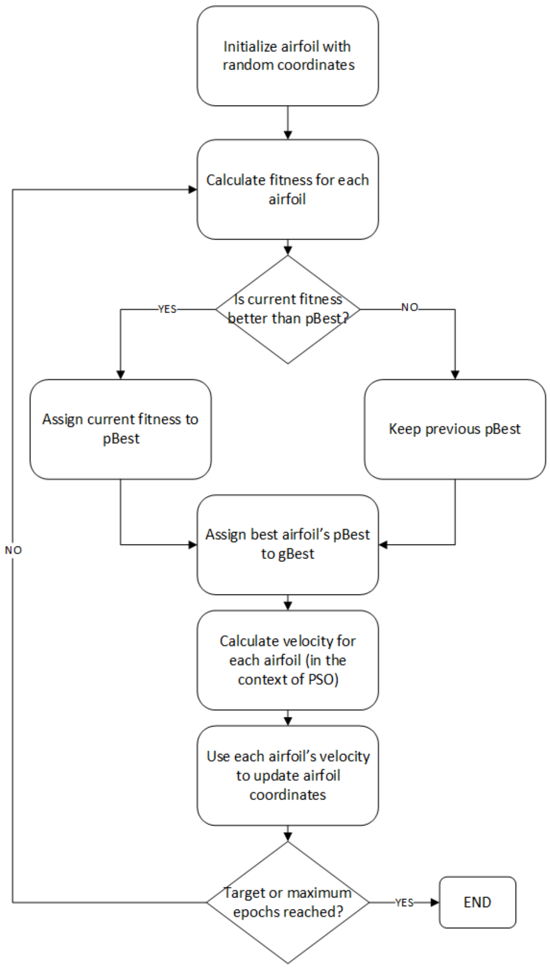

Fig. 3 Flowchart showing the working of PSO

tural overview for the airfoil optimization scheme that was implemented using Python and XFoil in conjugation.

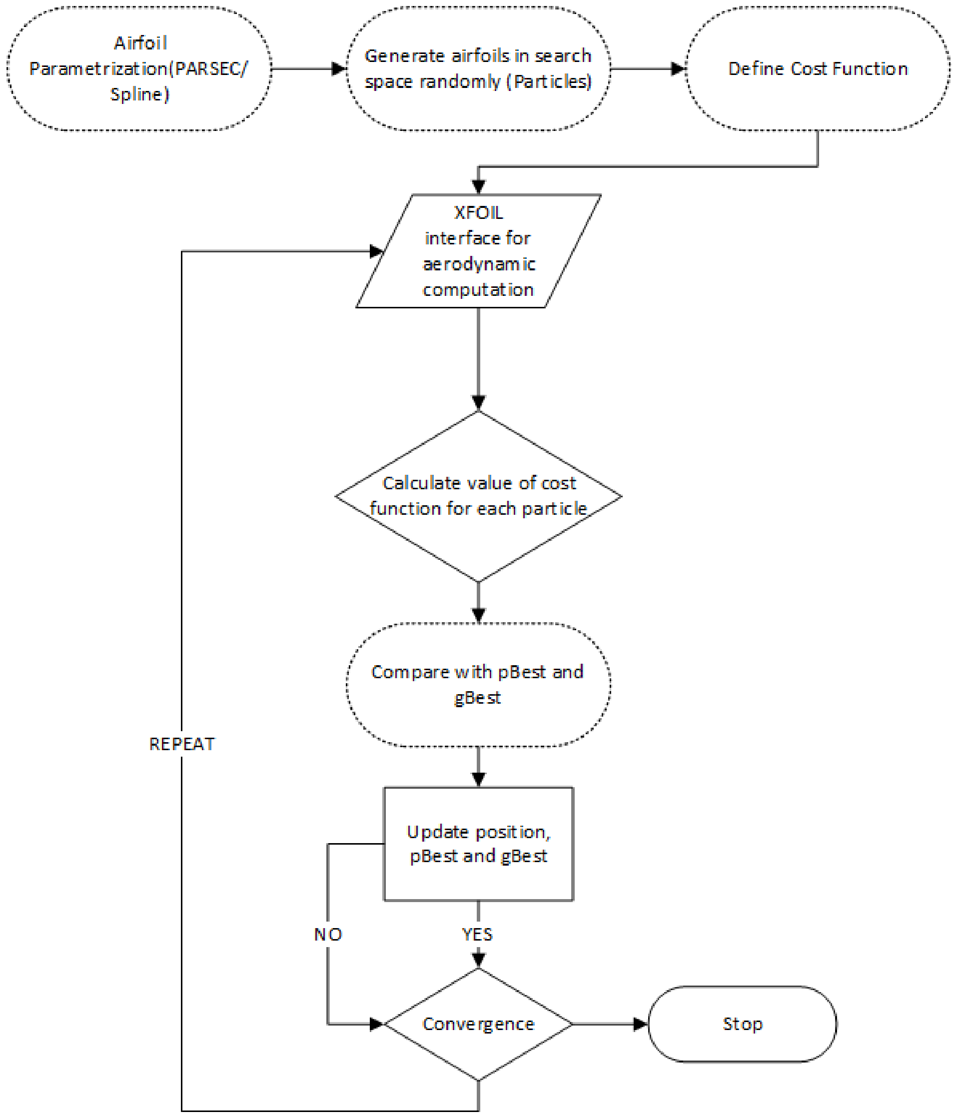

Fig. 4 Airfoil Optimization Process

V. Airfoil Optimization: Case Study

This section discusses the implementation of the optimization scheme on a variety of optimization problems. It considers different cost functions, commonly used during airfoil design and discusses the result thus obtained for each case.

Table 1 defines the constraints under which random airfoils (particles) for the PSO optimizer were generated. This is thus, the design space for the first case study that uses Ferguson spline based parametrization alongside PSO.

Table 1 Constraints for spline based parametrization

T_A^{upper}	T_A^{lower}	T_B^{upper}	T_B^{lower}	α_b(deg)	α_c(deg)
0.1 - 0.4	0.1 - 0.4	0.1 -2.0	0.1 - 2.0	-15 - 15	1 - 30

A. Case Study 1

Problem Definition

1. Cost Function: $min(C_d)$

2. Reynolds Number: $Re = 1 \times 10^6$

3. Angle of Attack: $\alpha = 0$ deg

The PSO optimizer searches for the airfoil with the least drag coefficient at the aforementioned conditions. Here the optimizer is used to search for an airfoil with required aerodynamic parameters instead of optimizing a particular airfoil. The PSO scheme is initialized with a random starting point and the optimizer tries to reduce the cost function i.e C_d at each iteration. Since tuning a PSO optimizer is a difficult task, the method described in Ref. [6] was used to tune the parameter values. Table 2 shows the values that were chosen after tuning the parameters. These parameters will be used for all subsequent case studies.

Table 2 PSO tuned Parameters

Iterations	Swarm Size	ω	θ_g	θ_p
12	12	-0.2	2.8	0

Figure 5 shows the airfoil obtained after the PSO optimization was carried out in the search space mentioned above. Global best can be seen reducing as the number of iterations progress. As can be observed in the figure, the airfoil shape though optimized, has an extremely thin trailing edge, which may not be realistic. But, since the design space isn't constrained, the optimization scheme aims to simply minimize the cost function. In conclusion, specifying a design space and very few or no constraints may not always produce a realistic airfoil.

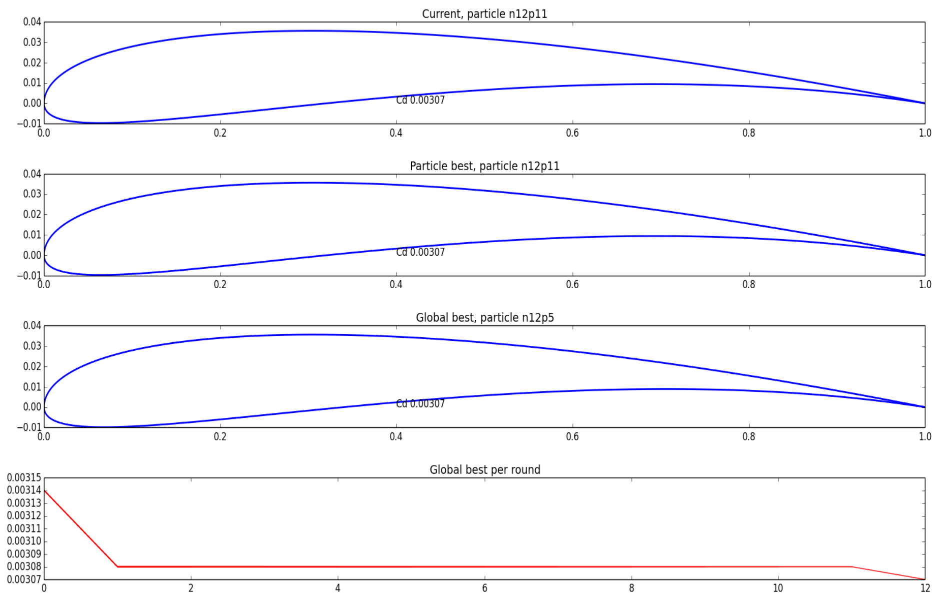

Fig. 5 Variation of airfoil shape with iterations

B. Case Study 2

Problem Definition

1. Cost Function: $min(C_d)$

2. Reynolds Number: $Re = 1 \times 10^6$

3. Angle of Attack: $C_l = 0.4$

Here the aim is to search for an airfoil that produces minimum C_d while generating a C_l of 0.4. It can be observed in Fig. 6, that the airfoil shape thus obtained is different from the one in Case Study 1. As discussed in Case Study 1, these results are only preliminary and may be improved by adding external constraints on the search space such as maximum camber, TE thickness etc. In general, the external constraints are highly problem specific and are usually derived from system level requirements.

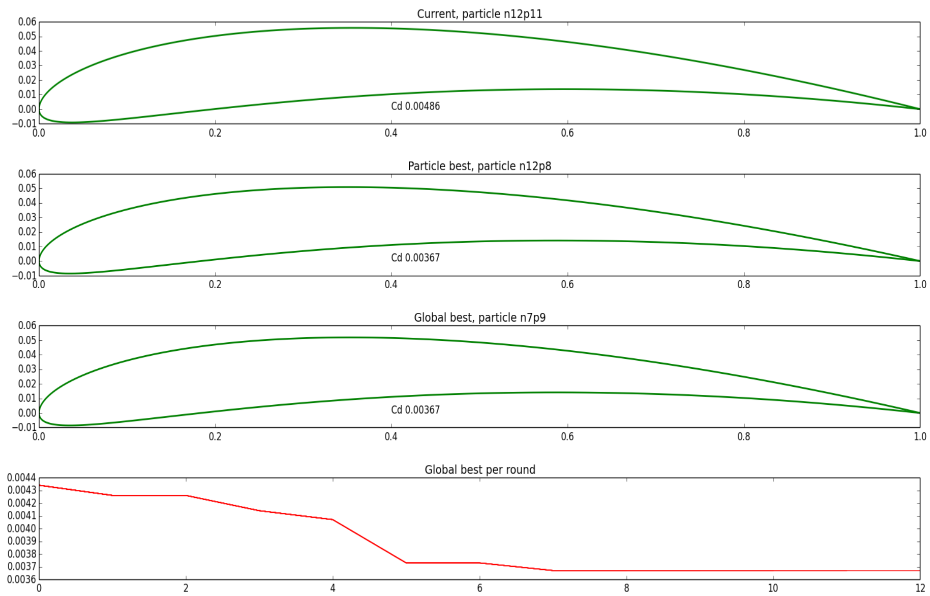

Fig. 6 Variation of airfoil shape with iterations

C. Case Study 3

Problem Definition

1. Cost Function: $min(C_d/C_l) \times 10^3$
2. Reynolds Number: $Re = 1 \times 10^6$
3. Angle of Attack: $\alpha = 4$ deg

The aim here is to generate the most efficient airfoil, in the search space defined in Case Study 1. Efficiency of an airfoil is synonymous to the value of $\frac{L}{D} = \frac{C_l}{C_d}$, which is the amount of lift the airfoil is producing for a unit drag. The PSO optimizer was applied with a cost function of $min((C_d/C_l) \times 10^3)$ such that the minimum of $\frac{C_d}{C_l}$ is obtained which translates to a maximum of $\frac{C_l}{C_d}$.

D. Case Study 4

1. Cost Function: $min(C_d/C_l) \times 10^3$
2. Reynolds Number: $Re = 1 \times 10^6$

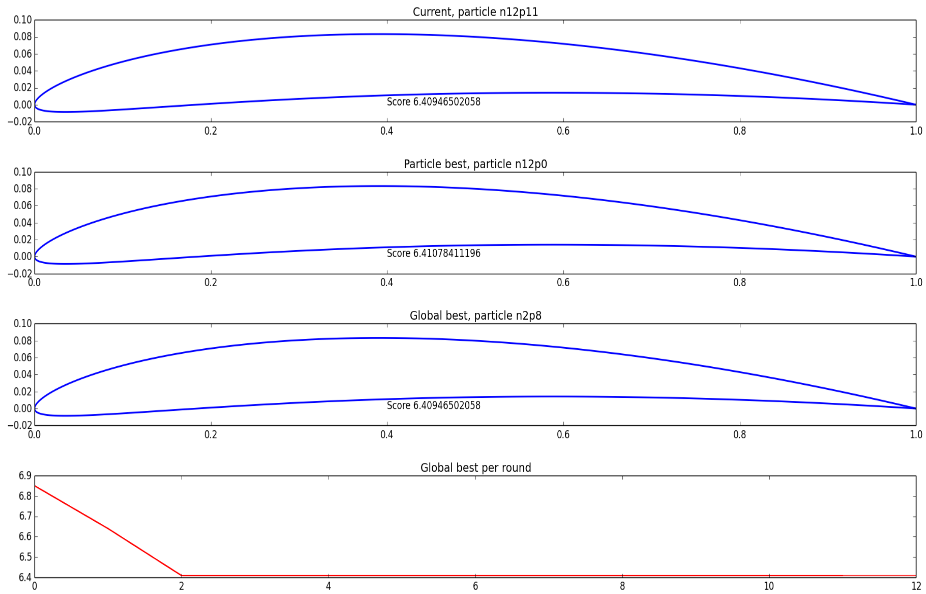

Fig. 7 Variation of airfoil shape with iterations

3. Angle of Attack: $\alpha = 4$ deg

Quite often the airfoil optimization process starts with a particular airfoil, since the designer wants to emulate the behavior of that airfoil and optimize it for some condition. Keeping in mind such a scenario, the aim of this case study was to start with an airfoil and then optimize it for the cost function mentioned above. NACA21012 was chosen as the base profile and was parametrized using the Ferguson Spine formulation and the only variables allowed to change during the optimization process were: T_A^{upper}, T_A^{lower}, T_B^{upper} and, α_c such that the optimized airfoil doesn't deviate a lot from the original profile.

Figure 8 shows the comparison between the original and the optimized airfoil shapes. Figure 9 and, Fig. 10 compare the C_d and C_l coefficients of the two profiles respectively. It can be observed from Fig. 11 that the $\frac{C_l}{C_d}$ value of the optimized airfoil is greater than the original airfoil between 0-12 deg α , even though the aim of the optimization process was to maximize $\frac{C_l}{C_d}$ at 4 deg α.

Since, PSO is a randomized optimization algorithm, we can only expect that the optimized airfoil performs better than the original, but the performance of the optimized will vary.

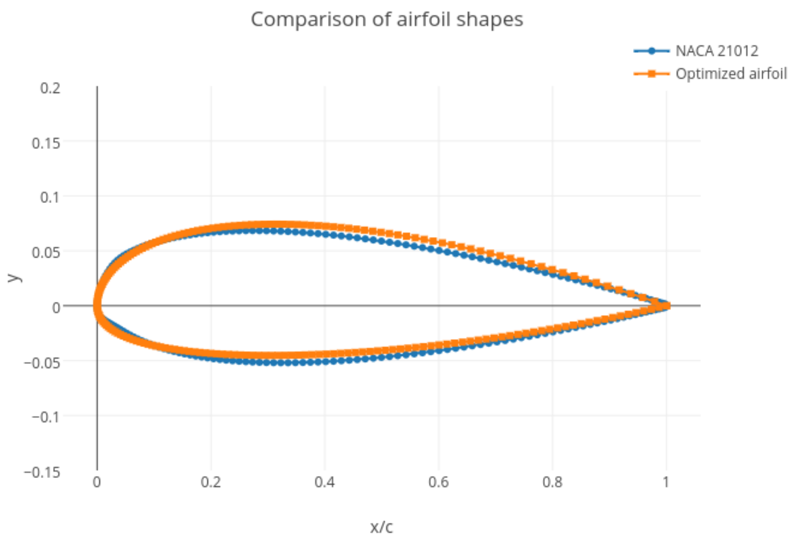

Fig. 8 Profile comparison of NACA21012 and the optimized airfoil

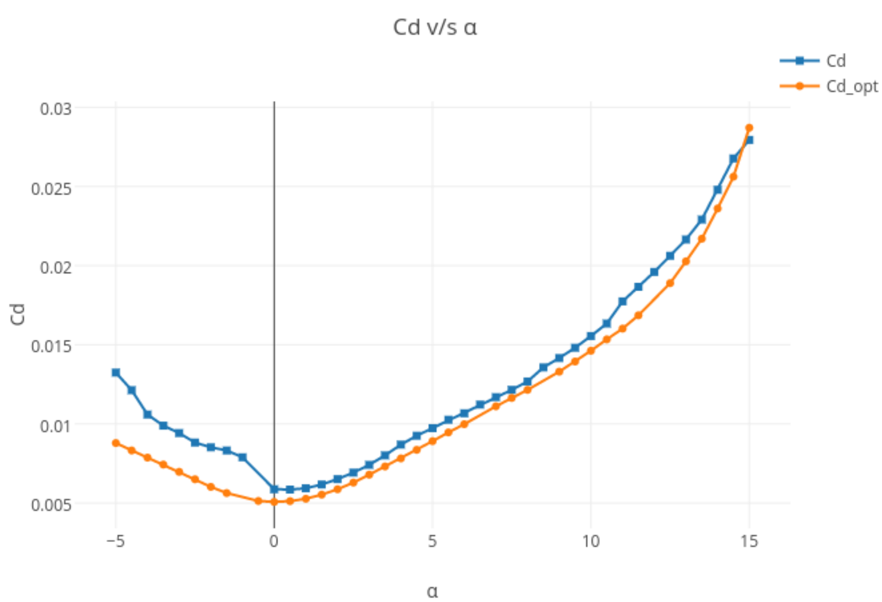

Fig. 9 Comparison of C_d values for original and optimized airfoil

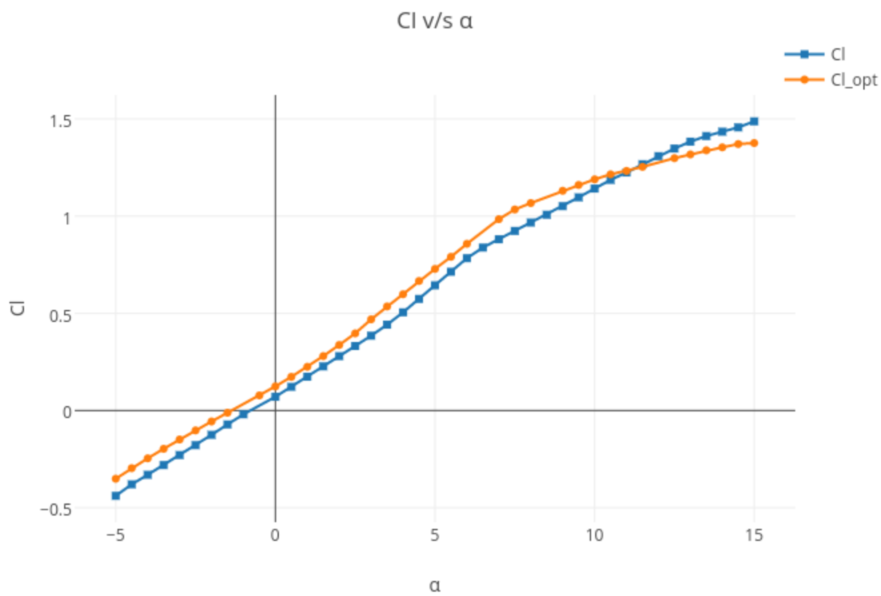

Fig. 10 Comparison of C_l values for original and optimized airfoil

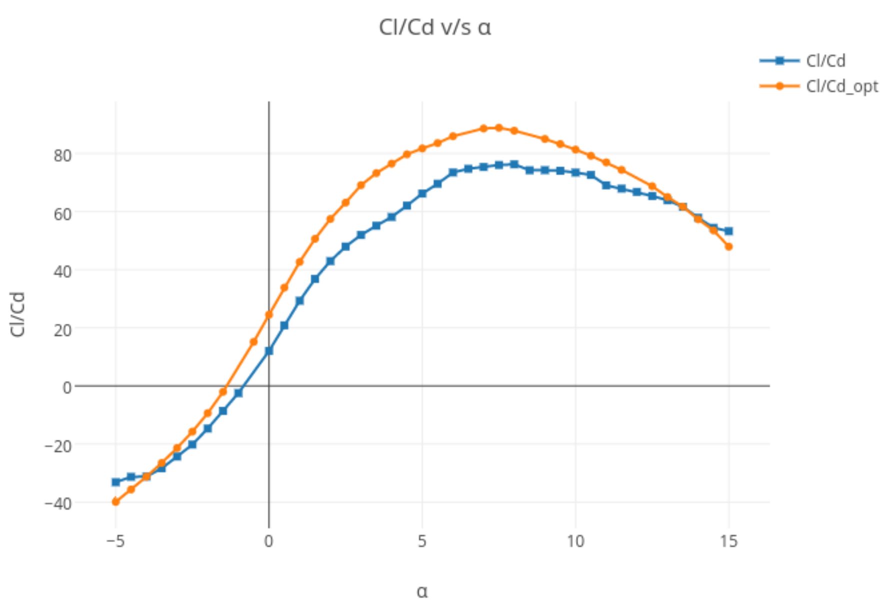

Fig. 11 Comparison of $\frac{C_l}{C_d}$ values for original and optimized airfoil

VI. Conclusion and Future Work

Aircraft optimization is a balancing act between the optimization of individual components such as airfoil shape and the multidisciplinary design optimization of the aircraft itself. This project showed the global optimization of airfoil with a very few design constraints but in the actual application external constraints derived from the systems design need to be incorporated in the optimization process. This was not incorporated as it is outside the scope of this project. Invariably a compromise and balance must be reached between the aerodynamic, structural, control, manufacturing and other aspects of aircraft's design optimization.

Airfoil design parametrization using Ferguson Spline and PARSEC method was implemented. PSO as an optimization scheme in conjugation with the aforementioned airfoil parametrization was successfully demonstrated using a Python-XFoil interface. The optimization could be extended to include different optimization schemes such as Ant Colony Optimization, Simulated Annealing, etc. For greater usefulness and control over airfoil shape, design variables could be included in the spline parametrization such as ϵ for a finite TE thickness. Since this was not implemented in the current spline based formulation, all the airfoil shapes obtained as a result have a zero TE thickness. It is recommended that the LE radius and maximum thickness location be included as design variables for spline formulation in future work.

References

[1] Sobester, A. and, Keane, A. J., "Airfoil Design via Cubic Splines - Ferguson's Curves Revisited," *AIAA Infotech@Aerospace*, May, 2007, pp. 1–15. doi: 10.2514/6.2007-2881

[2] Sobieczky, H., "Parametric Airfoils and Wings," *Notes on Numerical Fluid Mechanics*, Vol. 68, pp. 71–88.

[3] Wang, Y., Bin-qian Z. and Chen, Y., "Robust Airfoil Optimization based on Improved Particle Swarm Optimization Method" *Applied Mathematics and Mechanics*, Vol. 32, No. 10, 2011, pp. 1245. doi: 10.1007/s10483-011-1497-x

[4] Derksen, R. W. and, Rogalsky T., "Bezier-PARSEC: An optimized aerofoil parameterization for design," *Advances in Structural Optimization*, Vol.41, July, 2010, pp. 923–930. doi: 10.1016/j.advengsoft.2010.05.002

[5] Terzopoulos, D. and, Qin H., "Dynamic NURBS with Geometric Constraints for Interactive Sculpting," *ACM Transactions on Graphics*, Vol.13(2), April, 1994, pp. 103–136.

[6] Pedersen, M. E. H., "Good Parameters for Particle Swarm Optimization," *Hvass Laboratories Technical Report no. HL1001*, 2010.

[7] Drela, M., *XFoil Subsonic Airfoil Development System*, https://web.mit.edu/drela/Public/web/xfoil/ [retrieved 15th Oct, 2016]

[8] Kennedy, J. and Eberhart, R. C., "Particle swarm optimization" *Proc. IEEE int'l conf. on neural networks*, Vol. IV, pp. 1942-1948. IEEE service center, Piscataway, NJ, 1995

[9] McCullock., J., *http://mnemstudio.org/particle-swarm-introduction.htm* [retrieved 23rd Nov, 2016]

[10] Kulfan, B. M. and Bussoletti, J. E., "Fundamental Parametric Geometry Representations for Aircraft Component Shapes," AIAA 2006-6948, 2006.