Trans. Japan Soc. Aero. Space Sci.
Vol. 50, No. 167, pp. 41–47, 2007

Morphing-Based Shape Optimization in Computational Fluid Dynamics

By Yannick ROUSSEAU, Igor MEN'SHOV and Yoshiaki NAKAMURA

Department of Aerospace Engineering, Nagoya University, Nagoya, Japan

(Received March 23rd, 2006)

In this paper, a Morphing-based Shape Optimization (MbSO) technique is presented for solving Optimum-Shape Design (OSD) problems in Computational Fluid Dynamics (CFD). The proposed method couples Free-Form Deformation (FFD) and Evolutionary Computation, and, as its name suggests, relies on the morphing of shape and computational domain, rather than direct shape parameterization. Advantages of the FFD approach compared to traditional parameterization are first discussed. Then, examples of shape and grid deformations by FFD are presented. Finally, the MbSO approach is illustrated and applied through an example: the design of an airfoil for a future Mars exploration airplane.

Key Words: Genetic Algorithm, Free-Form Deformation, Shape Optimization, Aerodynamics

1. Introduction

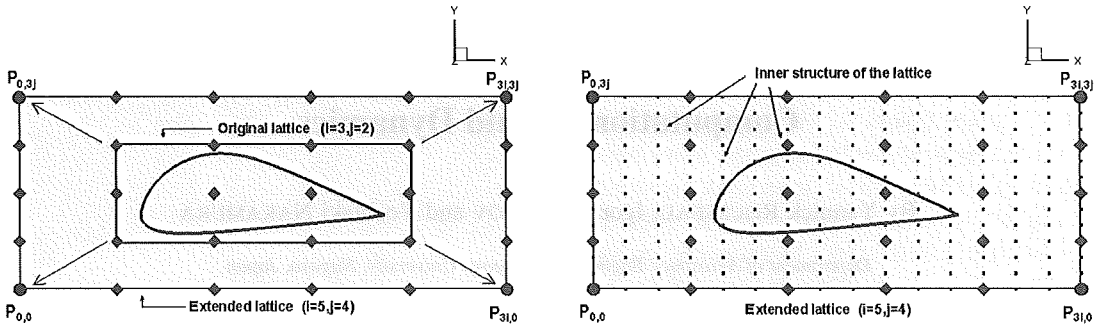
Due to recent progresses in Computational Fluid Dynamics (CFD) and grid generation software, a growing number of scientists and engineers are relying on numerical simulations to reduce the time of design process, and are consequently becoming interested in automatic shape optimization as a design tool. However, optimum-shape design in CFD is still a complex task requiring the integration of three components, namely optimizer, flow solver and mesh generation software. With computational power more easily available GAs and variants of Genetic Algorithms (GA) have gained wide acceptance and are being used as numerical optimizers in aerospace, automotive and other industries.[1–7] However, shape parameterization and integrability issues still impose serious limitations on automatic shape optimization.

Firstly, the parameterization must balance the demands for high geometric variability in shapes and the need to keep the number of design variables low. Indeed, the latter demand is important because a smaller variable set implies a faster convergence of the optimization process. In general, the population size in GA-based optimizer should be no smaller than 20 to 30 individuals regardless of the problem being tackled. For problems with high dimensions, larger populations on the order of hundreds are appropriate. As a larger population size often requires more generations for convergence, the total computational cost usually grows quadratically with the problem's dimension. As a consequence, the design space of the shape optimization problem should be defined by the smallest parametric description possible to guarantee a minimum number of design variables. This conflicts with a highly flexible representation by which a wide variety of geometries should be realized and their performances assessed.

Contemporary optimization practice is to directly parameterize a shape by using parametric curves or surfaces such as NURBS and Bézier patches. The optimization is carried out using control points of the parameterization as design variables. This method requires curve fitting of the original geometry into the parametric design space. To preserve fidelity, the number of control points unavoidably increases according to the complexity of the initial geometry. The total number of design variables may be some hundreds; in that context, optimization using GAs requires a large population size, and consequently thousands of function evaluations may be needed to obtain an improved design. Moreover, parameterization based on a parametric surface/curve requires careful investigation of the range of variation of design variables to ensure feasible shape and grid generation.

Secondly, there is an additional requirement for parameterization, which is specific to design optimization problems involving CFD calculations. In this case, a computational grid has to be generated for each individual. A complex geometry necessitates the use of mesh generation software. The remeshing must be performed. Integrating mesh generation software with optimization code and flow solver requires high-level skills in programming, and the time devoted to the development, debugging and implementation of this approach may seem difficult to justify for small research groups and laboratories who already have in-house flow solvers and numerical optimizers. For these reasons, a parameterization that also permits omission of the remeshing procedure is highly advantageous in the current design environment.

In this paper, we propose a Morphing-based Shape Optimization (MbSO) technique. The method couples Free-Form Deformation (FFD) and Evolutionary Computation, and as its name suggests, relies on the morphing of shape and computational domain rather than direct shape



(a) Extended lattice (b) Internal structure

Fig. 1. Free-Form Deformation uniform control volume.

parameterization. Advantages of the FFD approach over traditional parameterization are first discussed. Then, examples of shape and grid deformations by FFD are presented. Finally, the MbSO approach is illustrated and applied through an example: the design of an airfoil for future Mars exploration airplane.

2. Free-Form Deformation as a Tool in Optimum Shape Design

2.1. Free-form deformation

FFD methods are new techniques in the field of Optimum Shape Design (OSD) in aerodynamics. They were initially developed in the field of computer graphics, allowing intuitive shape deformation by manipulating control points of a control volume (referred to as a *lattice* in computer graphics literature) that encloses the geometry, rather than the geometry itself. Sederberg and Parry[8] originally formulated the concept of FFD for parallelepiped-shaped lattice and Coquillart[9] extended this concept to any arbitrary shaped lattice.

As the topology of the lattice is independent of the complexity of the embedded geometry, complex geometry can be treated using a small number of control points. The design engineer can also group control points together to define macro deformations such as bending, twisting and stretching. Optimization can be carried out on control points and/or on the designer's pre-defined set of deformations.

2.2. Shape and grid morphing by FFD

The key point of the proposed method is to deform not only the shape, but also the computational grid inside the lattice. Therefore, the shape can be modified directly within the CFD model without the need to remesh each new geometry during calculations. This drastically simplifies the optimization process. Features of the MbSO approach are described in the following.

The shape that is modified during optimization is first embedded in a control volume defined by a set of control points. Then, the control volume is extended to embed both the shape and the computational domain. The coordinates of the resulting embedded geometries (shape and computational domain) in the lattice parameter space are calculated by a procedure referred to as "freezing".[8] For uniform parallelepiped control volume, "freezing" is rather simple.[9] For non-uniform parallelepiped control volume, "freezing" is more complex and involves inversion of the mapping between the lattice space coordinates and the original Cartesian coordinates.[9]

In this study the mapping inversion is realized by a Newphon-Raphson procedure.[10]

Finally, the convex hull property is used to decide whether or not a point is inside the deformable region. The freezing procedure has to be done only once at the beginning of the optimization process. To preserve the continuity between deformable and non-deformable regions, only the inner control points of the lattice are allowed to move. Then, by simply manipulating inner control points, one can deform geometries indefinitely.

In our implementation, Bézier bivariate tensor products are used, but other researchers may choose different deformation functions such as cubic B-splines. To illustrate our implementation of FFD, a simple parallelepiped lattice is employed. We initially embed a two-dimensional airfoil in a lattice of control points. Then, we extend the lattice to embed both airfoil shape and computational grid as shown in Fig. 1(a).

As a result, the internal structure of the control volume is defined by an array of $(3l+1) \times (3m+1)$ control points $P_{i,j}$ $(i = 0, \ldots, 3l, j = 0, \ldots, 3m)$, which can be seen from Fig. 1(b).

Finally, to ensure continuity inside the volume and at the boundary of the lattice only, the inner control points $(3i, 3j)$ $(i = 1, \ldots, l-1, j = 1, \ldots, m-1)$ are allowed to move in order to deform the geometries. Two macro deformations are implemented in this study to modify the camber and thickness of the airfoil by grouping control points that bend and stretch (or shrink) the airfoil shape. Figures 2(a) and 2(b) illustrate some examples of shapes and grid deformations obtained by these two transformations. As one can see, FFD techniques produce a smooth morphed mesh without any cell quality issues such as negative volume or highly distorted cells.

Moreover, two control points $P_{3,6}$ and $P_{12,6}$ that are locat-

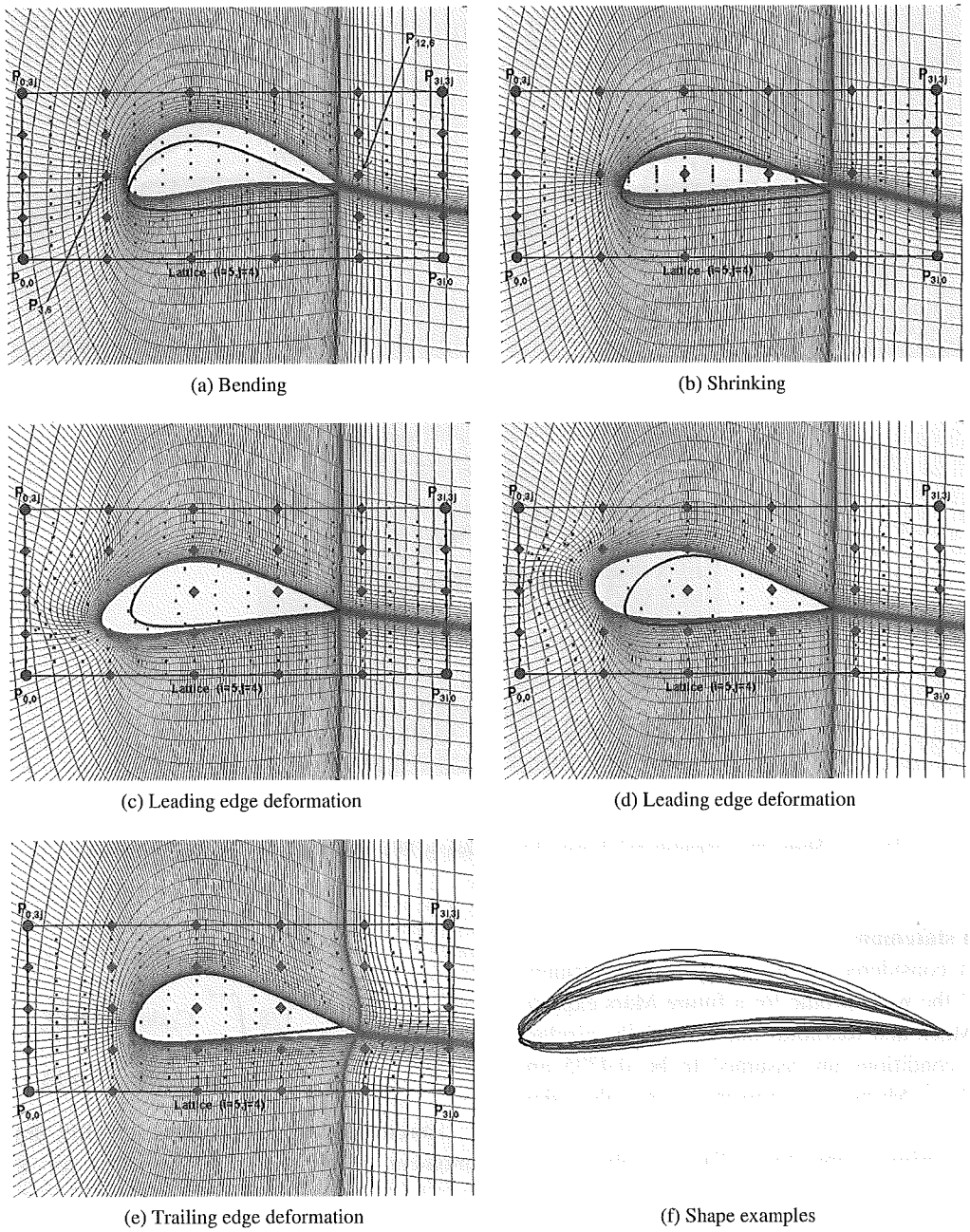


Fig. 2. Shape and computational domain deformations by FFD.

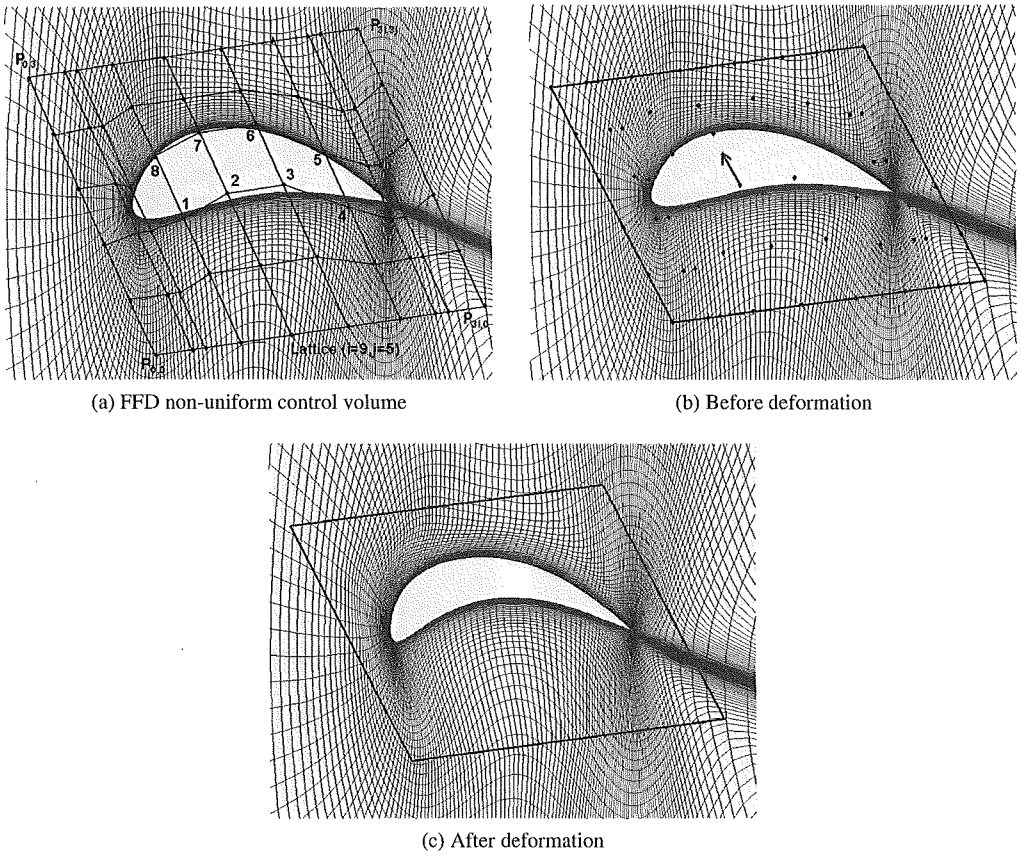
ed near the leading and trailing edges of the airfoil are allowed to move freely in the x and y directions. These control points not only have strong influence on the shape, but also allow implicit changes in the angle of attack as seen from Figs. 2(c) to 2(e). Thus, the airfoil is parameterized with only six design variables: two macro deformations and (x, y) of the two control points.

Figure 2(f) illustrates examples of shapes obtained with these six design variables. Thus, FFD techniques allow a wide variety of shape deformation with a small number of design variables, and also avoid the problem of re-meshing. In the next section, we will show our results of MbSO by coupling FFD and Evolutionary Computation, and discuss the quality of the morphed meshes.

3. Morphing-Based Shape Optimization by Coupling FFD and Evolutionary Computation

3.1. Flow solver

The Reynolds-Averaged Navier-Stokes equations are solved using a finite-volume cell-centered method for space discretization and a 5-multistage Runge-Kutta scheme for time integration.[11] Implicit residual averaging, combination of second and fourth order artificial dissipations, local-time stepping and enthalpy damping are used along with the Baldwin-Lomax turbulence model.[12] The GA is parallelized using message-passing interface (MPI) libraries. Calculations were performed on a supercomputer system, Fujitsu PRIMEPOWER HPC2500, at the Information Technology Center of Nagoya University.



(a) FFD non-uniform control volume

(b) Before deformation

(c) After deformation

Fig. 3. Shape and computational domain deformation by FFD with a non-uniform control volume.

3.2. Problem statement

The problem considered in this study is the optimum-shape design of the wing profile for a future Mars exploration airplane. Mach and Reynolds numbers of the airplane under cruising conditions are assumed to be 0.4735 and 10^5, respectively.[13] Martian air properties[13] are also taken into account.

There are two primary objectives. The first objective is to achieve a high maximum lift coefficient, C_{Lmax}, which will result in high maneuverability. The second objective is to obtain good characteristics of lift over drag ratio, L/D, at more moderate lift coefficients. In addition, the airfoil should exhibit docile stall characteristics. One major constraint is placed on the design of this airfoil; that is, the zero-lift pitching-moment coefficient, $C_{\mathrm{M}(0)}$, also referred to hereafter as C_{M}, must be no more negative than -0.15.

The design target is to achieve $C_{\mathrm{Lmax}} \sim 1.50$, $L/D \sim 40$ at moderate C_{L} and $C_{\mathrm{M}} \geq -0.15$. The objective of good L/D characteristics is transformed into two additional constraints; the lift coefficient at zero angle of attack must equal 0.70, and the average drag coefficient over $\alpha \in [-4°, 4°]$ must be no greater than 0.020.

Consequently, the problem is formulated as a single objective constrained optimization problem:

Maximize C_{Lmax}

subject to the following three constraints:

$$C_{\mathrm{L}}(\alpha = 0°) \geq 0.7$$
$$1/8 \int_{-4°}^{+4°} C_{\mathrm{D}} \mathrm{d}\alpha \leq 0.020$$
$$C_{\mathrm{M}}(\alpha = 0°) \geq -0.15 \qquad (1)$$

Aerodynamic characteristics (i.e., lift, drag and pitching moment coefficients) are evaluated for angle of attack $\alpha \in [-4°, 14°]$ with 1° steps. The GA population size is 30 individuals.

3.3. Application

The commercial software GRIDGEN is used to generate C-type meshes. The initial airfoil geometry is a low Reynolds number airfoil on Earth, an *Eppler 205* taken from the UIUC Airfoil Coordinates Database. First, a C-type mesh is generated, which has a total of 351 × 54 cells, with many of them clustered near the airfoil surface (the first grid spacing of 10^{-4}) and the trailing edge. The shape and computation domain are embedded in the previously mentioned uniform parallelepiped lattice resulting in a parameterization of only six design variables, shown in Fig. 1.

Calculations are stopped after 1000 CFD calls, and a new computational domain is generated using the best shape solution of the optimization run. Since control points that are close to the shape have maximum influence on the shape deformation, a non-uniform parallelepiped lattice is employed to refine the parameterization. Figure 3 shows the non-uniform control volume for which eight control points

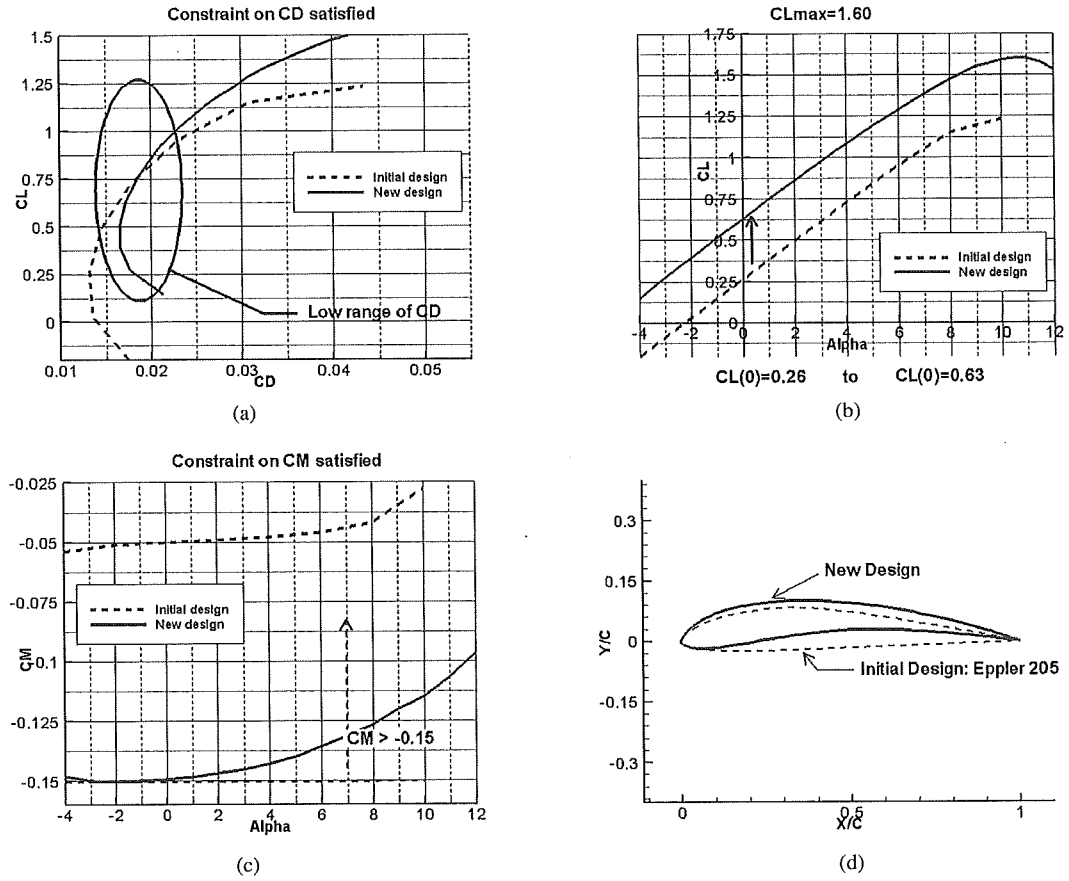


Fig. 4. Performance comparison of the new design with the initial *Eppler 205*.

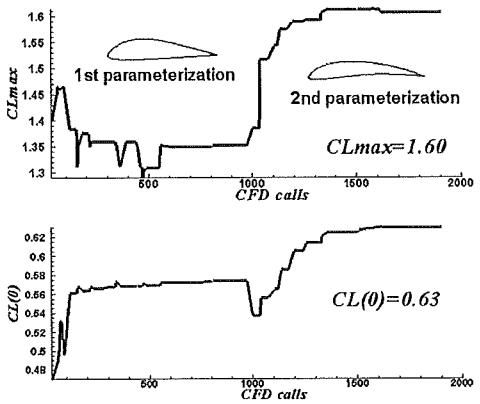


Fig. 5. Cumulative convergence histories of the shape optimization problem: $C_{L\max}$ and $C_L(\alpha = 0°)$.

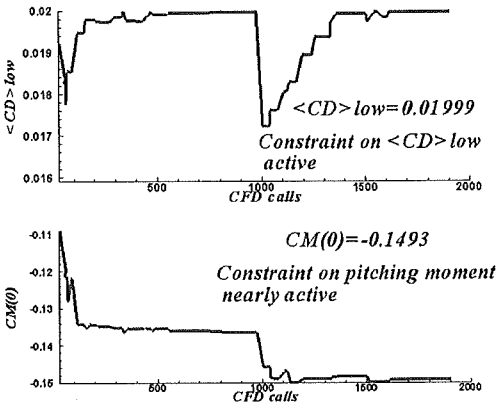


Fig. 6. Cumulative convergence histories of the shape optimization problem: $\int C_D$ and $C_M(\alpha = 0°)$.

(numerated by 1 to 8 in Fig. 3(a)) are allowed to move in the direction illustrated by the arrow in Fig. 3(b). In addition, three macro deformations are implemented to modify the camber, thickness and leading edge radius of the airfoil. As a consequence, 11 design variables are used for the second parameterization. Figures 3 (b) and (c) illustrate an example of deformation obtained by moving only one control point. As can be seen, by positioning control points of a non-uniform lattice near the shape, FFD can make very surprising, large shape and computational domain deformations.

Figures 5 and 6 show the cumulative convergence histories of objective and constraint functions of the two successive optimizations.

Characteristics of the newly designed airfoil obtained as the morphed mesh solution are shown in Fig. 4. The maximum lift coefficient for a design Reynolds number of 10^5 and a Mach number of 0.4735 is 1.60, which meets the design objective. The lift coefficient at zero angle of attack is predicted to be 0.63, which is below the design constraint of 0.70. During the design process, it was determined that violating the constraint was necessary to meet

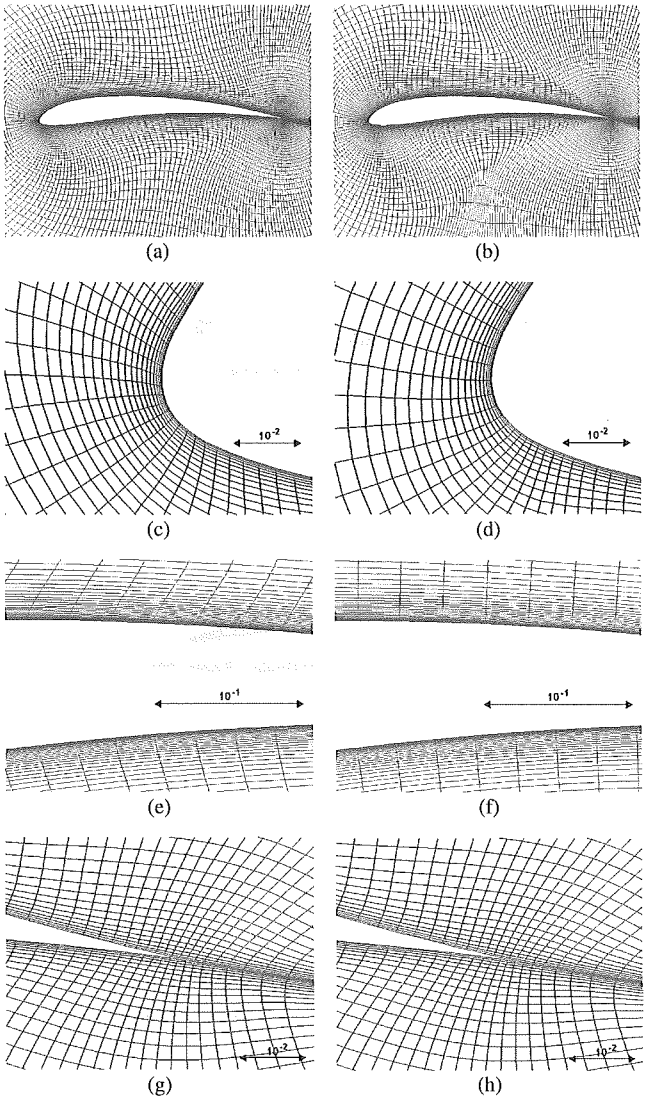


Fig. 7. Comparison of mesh patterns among the morphed mesh solution (a, c, e, g) and a high-quality mesh (b, d, f, h) generated using GRIDGEN.

other, more important goals such as constraints on pitching moment coefficient and low range of drag coefficients. The average drag coefficient over the range of angle of attack $[-4°, 4°]$ is expected to be 0.019 (the corresponding constraint is satisfied), which results in a range of lift over drag ratio L/D superior to 40 for C_L range of $[0.7, 1.3]$. The pitching moment coefficient is predicted to be -0.149, which satisfies the design constraint. Moreover, the airfoil is expected to exhibit docile stall characteristics (trailing edge stall). The design requirement is therefore fulfilled. The airfoil shape (see Fig. 4(d)) is very similar to the ND 01 wing profile used as the midspan wing section of the Mars airplane conceptual design reported in other literature.[13,14]

3.4. Mesh quality

Because low-quality meshes may produce erroneous results and compromise the final optimum shape solution, the quality of the deformed grids must be guaranteed during calculations. It is first verified by visual inspection.

The final mesh resulting from the calculations using the 2nd parameterization are presented in Figs. 7(a)(c)(e) and (g). Figures 7(b)(d)(f) and (h) show the mesh patterns of the high-quality mesh generated using GRIDGEN based on the same shape solution. As one can see in Fig. 7(e), the camber movement, which is often encountered in a design optimization process, results in a large computational domain deformation when employing the proposed approach. The orthogonality of the near-wall mesh, although less othogonal than the mesh generated by the mesh generation software, is preserved with good quality and local mesh metrics, which are important for resolving the boundary layer. To further validate the proposed approach, we solve the flow around the shape on the high-quality grid generated with the mesh generation software and compare the results with the deformed mesh solution. Figure 8 illustrates this comparison. Except for a small difference at high angles of attack, the maximum lift coefficient with the high-quality mesh is 1.55, and the two solutions fall on the same line. The quality of the mesh deformed by FFD is therefore satisfactory.

4. Conclusion

In this paper, we present a MbSO technique for solving OSD problems in CFD by coupling FFD and Evolutionary Computation. The key point of the method is to embed inside the FFD control volume not only the shape, but also part of the computational domain in order to allow shape and grid deformations within the CFD model. The advantages of shape and grid morphing by FFD within evolutionary computation are twofold. Firstly, since FFD allows to decouple geometry complexity from the parameterization, optimization can be carried out with a small number of design variables, which is important when using a GA-based numerical optimizer. Secondly, the proposed method permits the omission of the re-meshing process, which drastically simplifies the optimization process. MbSO is illustrated and successfully applied through an example, the design of an airfoil for a future Mars exploration airplane, for which only two grids were generated in the design process. FFD produces a smooth morphed mesh without any cell quality issues, and with an acceptable degree of accuracy. Since FFD techniques are independent of grid topology, the proposed approach is not limited to structured grids and can be applied to unstructured, structured multi-blocks and hybrid grids. This shows the effectiveness of MbSO, and future work will focus on the application of MbSO to three-dimensional OSD problems as well as investigating volume-constraint and other constraint deformation techniques.

References

1) Holland, J.: *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Michingan, 1975.
2) Goldberg, D. E.: *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Massachusetts, 1989.

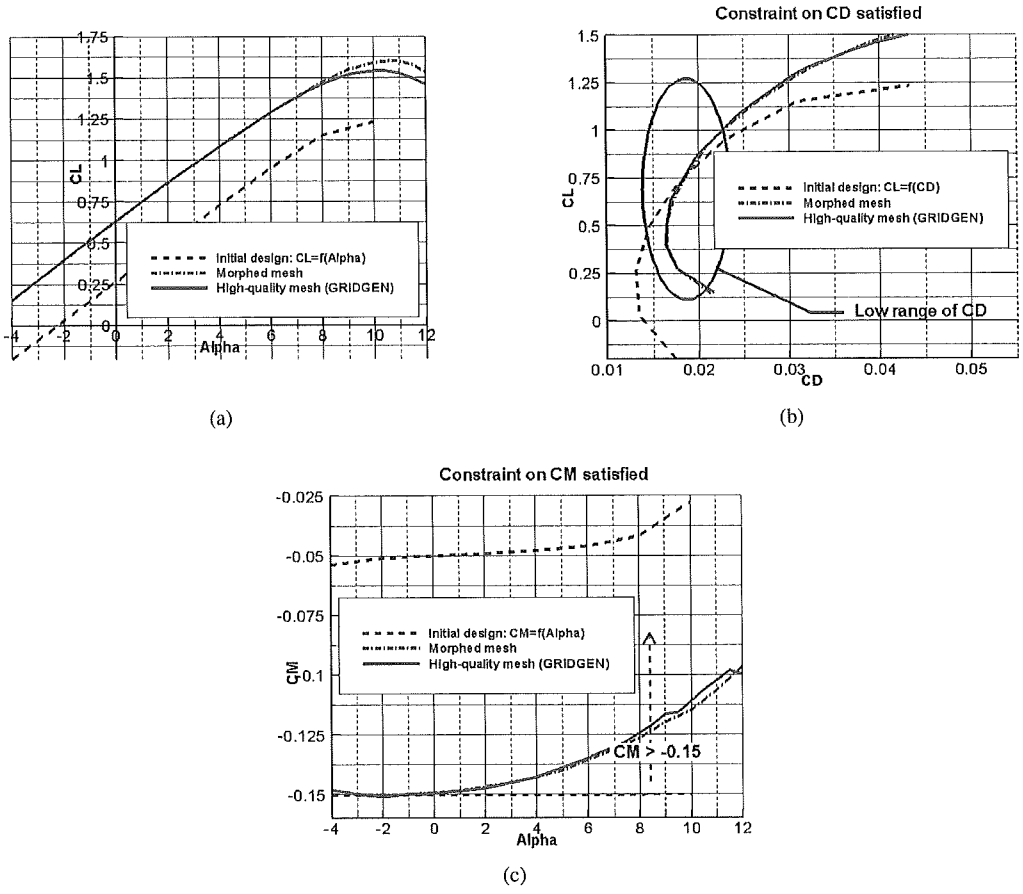


Fig. 8. Performance comparison of the new design solved over the morphed mesh solution and a high-quality mesh generated with GRIDGEN.

3) Schwefel, H. P.: *Evolution and Optimum Seeking*, John Wiley and Sons, Inc., New York, 1995.
4) Baker, J. E.: Reducing Bias and Inefficiency in the Selection Algorithm, Proceedings of the Second International Conference on Genetic Algorithms and their Application, 1987, pp. 14–21.
5) Syswerda, G.: Uniform Crossover in Genetic Algorithms, Proceedings of the Third International Conference on Genetic Algorithms, 1989, pp. 2–9.
6) Gottlieb, J.: Evolutionary Algorithms for Constrained Optimization Problems, Ph.D. Dissertation, Technical University of Clausthal, Aachen, Germany, 2000.
7) Coello Coello, C. A.: Constraint-Handling Using an Evolutionary Multiobjective Optimization Technique, *Civil Engineering and Environmental Systems*, vol. 17, 2000, pp. 319–346.
8) Sederberg, T. W. and Parry, S. R.: Free-Form Deformation of Solid Geometric Models, *Computer Graphics*, **20** (1986), pp. 151–160.
9) Coquillart, S.: Extended Free-Form Deformation: A Sculpturing Tool for 3D Geometric Modeling, *Computer Graphics*, **24** (1990), pp. 187–196.
10) Ipbuker, C. and Bildirici, I. O.: A General Algorithm for the Inverse Transformation of Map Projections Using Jacobian Matrices, Proceedings of the Third International Conference on Mathematical and Computational Applications, Konya, Turkey, 2002, pp. 175–182.
11) Jameson, A., Schmidt, W. and Turkel, E.: Numerical Solution of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time Stepping Schemes, AIAA Paper 81-1259, 1981.
12) Baldwin, B. S. and Lomax, H.: Thin Layer Approximation and Algebraic Model for Separated Turbulent Flows, AIAA Paper 78-257, presented at the 16th Aerospace Sciences Meeting, Huntsville, Alabama, 1978.
13) Hall, D. W., Parks, R. W. and Morris, S.: Airplane for Mars Exploration, available: http://www.redpeace.org.
14) Guynn, M. D., Croom, M. A., Smith, S. C., Parks, R. W. and Gelhausen, P. A.: Evolution of a Mars Airplane Concept for the ARES Mars Scout Mission, AIAA Paper 2003-6578, presented at the 2nd AIAA Unmanned Unlimited Systems, Technologies, and Operations, 2003.