International Congress on Evolutionary Methods for Design,
Optimization and Control with Applications to Industrial Problems
EUROGEN 2003
G. Bugeda, J.A.- Désidéri, J. Periaux, M. Schoenauer and G. Winter (Eds)
©CIMNE, Barcelona, 2003

DESIGN OPTIMIZATION OF THE WING SHAPE FOR THE RLV BOOSTER STAGE USING EVOLUTIONARY ALGORITHMS AND NAVIER-STOKES COMPUTATIONS ON UNSTRUCTURED GRIDS

Kazuhisa Chiba[⋆], **Shigeru Obayashi**[⋆], **Kazuhiro Nakahashi**[†],
Alexios P. Giotis[‡], **and Kyriakos C. Giannakoglou**[‡]

[⋆]Institute of Fluid Science, Tohoku University
Katahira 2-1-1 Sendai 980-8577, JAPAN
e-mail: chiba@edge.ifs.tohoku.ac.jp, obayashi@ieee.org
web page: http://www.ifs.tohoku.ac.jp/edge/

[†]Department of Aeronautics and Space Engineering, Tohoku University
Aoba-yama 01, Sendai 980-8579, JAPAN
e-mail: naka@ad.mech.tohoku.ac.jp, web page: http://www.ad.mech.tohoku.ac.jp

[‡]Laboratory of Thermal Turbomachines, National Technical University of Athens
P.O.Box 64069, Athens 157 10, GREECE
e-mail: [agiotis, kgianna]@central.ntua.gr

Key words: CFD, MOEA, ANN, RLV, Aerodynamic Evolutionary Design.

Abstract. *The wing shape of the booster stage of a two-stage-to-orbit Reusable Launch Vehicle (RLV) is optimized by considering trajectory computations coupled with three 3-D Navier–Stokes flow analyses around the complete booster configuration at three instances of its flight, namely at supersonic, transonic and subsonic flow conditions. The four-objective optimization is carried out using the Evolutionary Algorithms System (EASY 1.3) software. EASY 1.3 makes use of evolutionary algorithms and Artificial Neural Networks (ANN). The role of the latter is to act as surrogate evaluation model for the preliminary approximate evaluation of the population members, in each generation. The role of the so-called Inexact Pre-Evaluation (IPE) phase is to pinpoint the most promising individuals to be then evaluated using the Navier-Stokes solver. Using the IPE technique, optimizations with "expensive" evaluation tools become affordable.*

1 INTRODUCTION

A key point concerning space utilization efforts is the reduction of the relevant costs. One of the ways to achieve this goal is through the use of Reusable Launch Vehicles (RLVs) to replace the expendable rocket systems currently in use.

In view of the above, this paper deals with the design of an optimal wing shape for such an RLV. The RLV mission is to transport a 10t payload and put it into low earth orbit. The two-stage-to-orbit (TSTO) RLV considered herein consists of the orbiter and the booster stages, with liquid propellant rocket engine, vertical take-off and horizontal landing. The booster should fly back to the launch site, so its aerodynamic performance is of primary importance. This work focuses only to the design of the booster; the three-dimensional wing shape will be optimized by considering a fixed fuselage geometry.

The complete analysis of the RLV flight requires the interaction of two kinds of analysis tools: the first concerns trajectory computations whereas the second is based on 3D flow analysis problems. The trajectory computation requires aerodynamic data computed from the numerical solution of the Navier-Stokes equations around the vehicle at a number of points along its path. On the other hand, the trajectory computing code provides the flow conditions for the Navier-Stokes analyses. However, in this paper, the (strong) coupling between trajectory computations and Navier-Stokes runs will be kept to the minimum, so as to avoid extremely time consuming computations. We recall that (regardless of the optimization method) the design process requires a great number of candidate solutions to be evaluated, so the cost of each evaluation should be kept as low as possible. Thus, only three 3D Navier-Stokes computations (using CFD tools developed in Tohoku University) per candidate solution were carried out, at three instances of the vehicle trajectory, at supersonic, transonic and subsonic flow conditions. The flow conditions for these runs were kept constant, without being affected by the trajectory computation.

The optimization was carried out using a multi-objective optimization tool based on evolutionary algorithms (*EASY 1.3*, [1, 2], developed by NTUA). Among the several capabilities of this software, the use of low-cost surrogate evaluation models for the selection of the most promising individuals in each generation, which will then be exactly evaluated, should be mentioned. The use of the surrogate model (here, “locally” trained ANNs, in particular radial basis function netwoks) results to economy in the computational cost and is very useful in the current application where the evaluation cost per candidate solution is extremely high.

2 DEFINITION OF THE AERODYNAMIC OPTIMIZATION PROBLEM

As previously mentioned, the reference mission of the two-stage-to-orbit RLV is to transport a 10t payload into low-earth orbit. During a preliminary computation through software developed by the National Space Development Agency of Japan (NASDA), the booster sizing was obtained. Among the computed geometrical data, the minimum fuselage diameter and the fuselage length were utilized during the entire design process, for each and every candidate solution. This practically means that the fuselage geometry was fixed and only the wing shape was allowed to vary, according to the parameterization system discussed in the following paragraph. The main reason for keeping the fuselage geometry fixed is that, in liquid propellant rocket engines, the fuselage is filled with the liquid propellant, so its shape can hardly be modified.

2.1 Definition of the wing shape

The design variables are related to the planform, the airfoil shapes and the wing twist [4]. The wing planform is determined by five design variables as shown in Fig. 1. A kink is placed on the leading edge because the lift increases due to the leading-edge separation.

Airfoil shapes are defined at the wing root, the kink and the tip, respectively, using three thickness distributions and three camber lines. The thickness distributions were linearly interpolated in the spanwise direction and they were represented by Bézier curves, with eleven control points each. The camber line distributions, defined at the same location as the thickness distributions, were parameterized using Bézier curves with four control points each. The wing twist was modelled using B-splines with six control points. The entire wing shape was, thus, defined using 71 design variables.

Once the wing was defined, the junction line between wing and fuselage was computed and, by neglecting the part of the wing inside the fuselage, the final wing-fuselage geometry was derived, [5]. Figure 2 shows a sample geometry along with a close-up view of the unstructured 3D grid used for the Navier-Stokes computations.

2.2 Aerodynamic evaluation

The aerodynamic evaluation of the booster requires the generation of an unstructured mesh [6] around its geometry and the flow analysis which was performed using a Reynolds-averaged Navier-Stokes equations solver, using a finite-volume, cell-vertex scheme. The numerical fluxes were computed with the Harten-Lax-van Leer-Einfeldt-Wada Riemann solver[7]. The Lower-Upper-Symmetric Gauss-Seidel (LU-SGS) implicit method[8] was used for time integration. The modified Spalart-Allmaras (S-A) one-equation turbulence model by Dacles-Mariani *et.al.* [9] was implemented and unstructured hybrid meshes [10] were used to treat turbulent boundary layers in high Reynolds number viscous flows. The flow conditions for the Navier-Stokes solutions at the three flight instances at which the flow was to be analyzed are tabulated in Table 1.

2.3 The objective functions

Based on the mission mentioned above, the trajectory analysis [3] around a typical TSTO-RLV configuration is shown in Fig. 3. In the same figure, the Mach number at each point over the trajectory is also shown. The separation of the booster and orbiter takes place at about 30 km height, at Mach number 3. Then, the booster stage turns over, slows down, cruises at transonic and lands at subsonic speed. Note that the major part of its crossrange is in the transonic region.

In the present study, the following four objective functions were defined and used:

- **Obj.1:** Minimization of the shifting of the aerodynamic center between supersonic and transonic flights, $F_1 = |C_{Mp}^{supersonic} - C_{Mp}^{transonic}|$.

  A significant control problem related to the RLV flight is the large variation of the aerodynamic center between supersonic and transonic flight conditions. It is, then, desirable to design wing geometries which yield small changes in the aerodynamic center when shifting from supersonic to transonic flight so that the necessary control mechanisms be reduced along with the vehicle weight.

- **Obj.2:** Minimization of the pitching moment at the transonic flight conditions, $F_2 = |C_{Mp}^{transonic}|$.

  It is known that the arrow wing used in the present study ensures high aerodynamic performance but, unfortunately, with large pitching moment. Thus, at transonic flight conditions, the pitching moment should be minimized for flight stability purposes.

- **Obj.3:** Minimization of the drag at the transonic flight conditions, $F_3 = C_D^{transonic}$.

  The trajectory analysis shows that the range of RLV is mostly covered by the transonic flow conditions. Thus, the transonic drag should be minimized in order to increase the flight range.

- **Obj.4:** Maximization of the lift at the subsonic flight conditions, $F_4 = C_L^{subsonic}$.

  To reduce the required runway distance, the lift obtained at the subsonic flight conditions should be maximized.

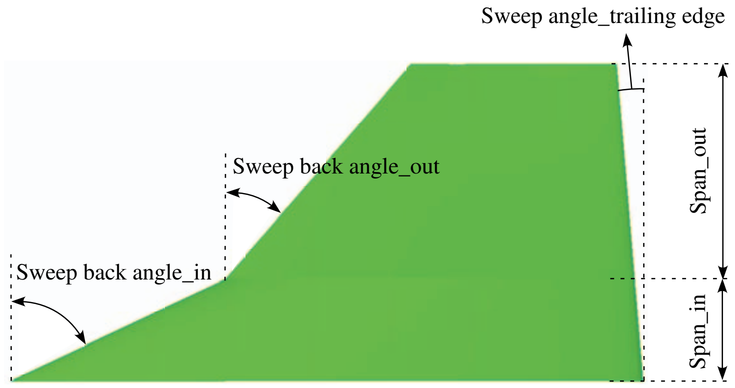


Figure 1: Planform shape definition along with some of the major design parameters.

Table 1: Flow conditions for the three Navier-Stokes computations.

Flying Condition	*Mach number*	*Angle of Attack*	*Reynolds number*
Supersonic flight	1.2	0.0	6×10^6
Transonic flight	0.8	8.0	6×10^6
Subsonic flight	0.3	13.0	6×10^7

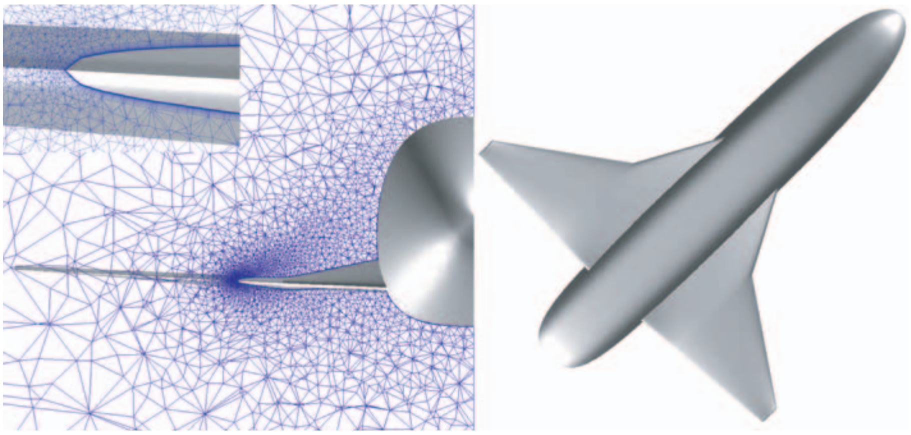


Figure 2: RLV geometry and close-up view of the volume grid near the leading edge at $x/L = 0.6$.

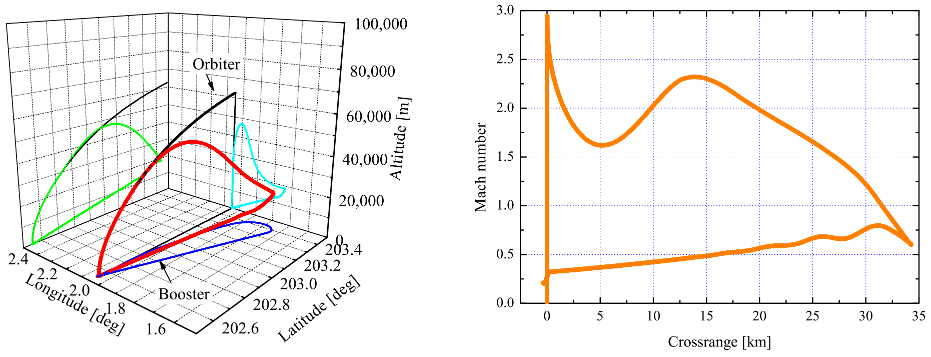


Figure 3: A typical flight trajectory (left) and its crossrange data (right) of the glideback TSTO using the X-34 aerodynamic characteristics [3].

3 THE OPTIMIZATION METHOD

The *EASY 1.3* optimization software should be considered a generalization of the most frequently used EA variants (Genetic Algorithms [11], [12] and Evolution Strategies [13]) for single- and multi-objective problems. In addition *EASY 1.3* offers the possibility of using ANNs as built-in surrogate evaluation models, in order to reduce the number of exact (CFD-based) evaluations required for the same solution quality [1], [2].

The notation symbols used for the description of the multi-objective optimization method are the standard ones used in Evolution Strategies. So, (μ, κ, λ) denotes an EA with μ parents and λ offspring, where the maximum allowed life span for parent individuals is equal to κ generations. Let $S^{g,\mu}$ and $S^{g,\lambda}$ denote the set of parents and offspring in the g-th generation. An additional archival set, denoted as $S^{g,a}$, is continuously updated and used during the evolution. In multi-objective optimization, $S^{g,a}$ contains the nondominated solutions (Pareto front) computed thus far and helps preserving elitism during the evolution.

At first, let us make clear that the evolution starts with a small number of generations (usually two or three) during which the exact (CFD) evaluation model is exclusively used. Data and results of exact evaluations are stored in a database which will later be used to build local surrogate models (ANNs).

In the subsequent generations, the first step is the so-called IPE phase. For each one of the λ individuals, a local ANN is built and trained using the closest database entries. The so-computed approximate objective function values for the population members yield temporary cost values ϕ^*, using domination criteria. The ϕ^* value of any individual is set equal to the number of the $S^{g,a}$ members which dominate it minus the number of $S^{g,a}$ members it dominates. Since $S^{g,\alpha}$ contains only the nondominated solutions, individuals with $\phi^* \leq 0$ are absolutely nondominated by the current archival front. Recall that this is a comparison between approximate and exact cost values. Upon completion of the IPE phase, the $S^{g,\lambda}$ members are rank sorted using the so-computed ϕ^* values and only a small percentage of them is selected for exact evaluations using the CFD tool.

In the next step, $S^{g,\alpha}$ is updated to $S^{g+1,\alpha}$ by taking into account only the exactly evaluated subset of $S^{g,\lambda}$. However, if $S^{g+1,\alpha}$ is overcrowded, (according to a user-defined maximum allowed population) an iterative thinning process, that eliminates one of its members at a time and finally reduces the population size to α, is employed, [14]. The inexactly evaluated individuals of $S^{g,\lambda}$ are not allowed to enter $S^{g+1,\alpha}$, not even to eliminate any individual from $S^{g,\alpha}$.

Using the values of the objective function, a unique cost value $\phi^{(i)}$, $i \in S^{g,\mu} \cup S^{g,\lambda} \cup S^{g+1,\alpha}$ per individual is computed. Through this assignment, standard single-objective evolution operators can be used. There is a large literature on the assignment of cost (or fitness) values ([14, 15]), based on domination criteria and the concept of the Pareto front. In the present analysis, a front ranking based method was used being one the numerous algorithmic variants offered to the users of *EASY*. The $S^{g,\mu} \cup S^{g,\lambda} \cup S^{g+1,\alpha}$ members are ranked in fronts using a repetitive procedure. Note that the first front (front 0) with the absolutely nondominated solutions is already known ($S^{g+1,\alpha}$). Its members are initially given the same lowest ϕ value. Then, in order to promote diversity, these values are penalized using sharing functions (niching). The cost assignment algorithm ensures that the ϕ value of any individual over the j-th front is greater than the highest ϕ value over the $(j-1)$-th front. This method is conceptually similar to the one proposed in [15], with certain modifications.

Aiming at preserving elite solutions in the active population, a small random fraction of $S^{g+1,\alpha}$ is copied to $S^{g,\lambda}$, by replacing an equal number of the worst individuals. The new $S^{g+1,\mu}$ set is created from the $S^{g,\lambda} \cup S^{g,\mu}$ individuals. First, individuals that reached the maximum allowed life span are eliminated from $S^{g,\mu}$. Then, the members of $S^{g,\lambda} \cup S^{g,\mu}$ are rank sorted in terms of their ϕ values and the μ top individuals form $S^{g+1,\mu}$. The new offspring set $S^{g+1,\lambda}$ is created by applying the parent selection operators to $S^{g+1,\mu} \cup S^{g+1,\alpha}$. Parent individuals are randomly selected from $S^{g+1,\mu}$ (with probability p_{ps}) or $S^{g+1,\alpha}$ (with probability $1-p_{ps}$). Whenever additional selective pressure needs to be exerted, the possibility of selecting parents with lower cost values increases using schemes such as the probabilistic tournament selection. The number of candidates participating in the tournament and the probability of selecting the candidate with the smaller cost value are user–defined parameters. Once two (or more) parents have been selected, recombination and mutation operators are applied to create a new offspring to be inserted into $S^{g+1,\lambda}$.

4 RESULTS AND CONCLUSIONS

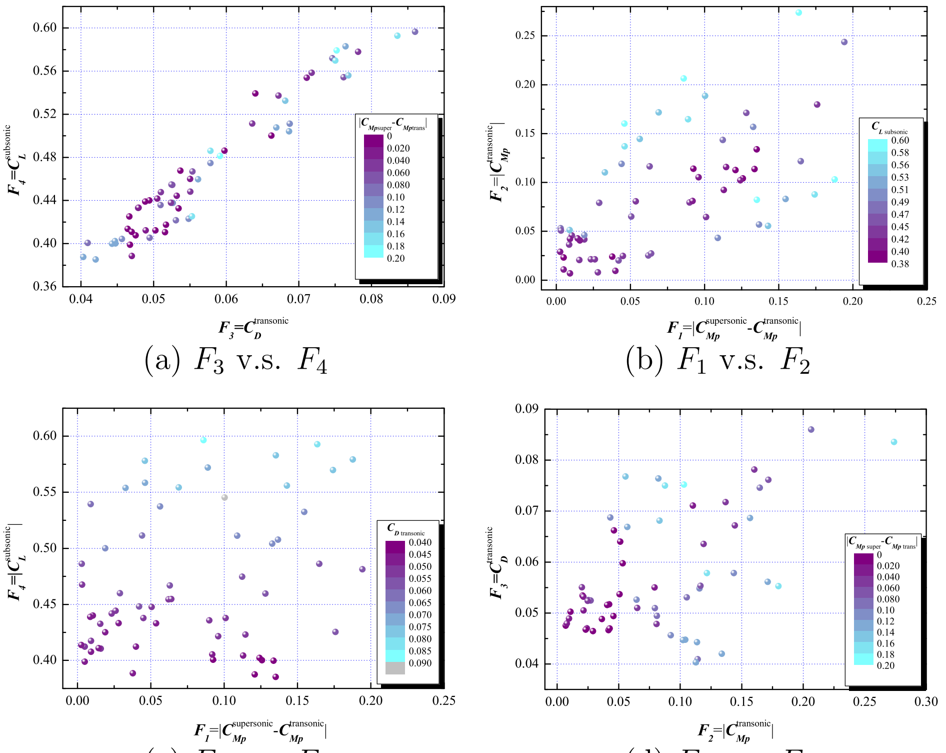
An EA with $\mu = 8$, $\lambda = 25$, $\kappa = 2$ and real coding was used in the present study. Each offspring was created by applying the discrete recombination operator with probability 90% on three parent individuals selected from $S^{g,\mu}$ through binary tournament with low probability (65%). Standard deviations were adapted using a generalized intermediate recombination operator. Since evaluations were extremely time consusing, a total of only 70 exact evaluation was allowed. The IPE phase was activated at the end of the second generation (after the first 50 exact evaluations) thus, allowing only two more generations in which ANNs and CFD were both used for the evaluation.

The wall clock time for each evaluation is about 8 hours. During the first 1.5h preprocessing including grid generation takes place sequencially on an SGI ORIGIN2000 scalar machine. Then, three flow analyses, through the numerical solution of the Navier-Stokes equations at predefined flow conditions are performed simultanesouly on different processors of a NEC SX-5 vector machine.

Though, the use of the ANN significantly reduces the number of exact evaluations required, the total number of 70 evaluations is considered small. Nevertheless, the results show an excellent trend towards optimal solution.

Since we are dealing with a four-objective optimization problem, the visualization of the obtained solutions is cumbersome. We summarized everything in two figures, Fig. 4 and 5. Fig. 4 presents four two-objective plots in order to have a better understanding of the correlation between the objectives. For instance, Fig. 4(a) shows that there is an almost linear relation between F_3 and F_4. However, this does not mean that these two objectives are non-conflicting because F_3 should be minimized and F_4 should be maximized. Fig. 5 presents 3D plots using F_1, F_2 and F_3 along the three vertical axes. We purposely eliminated F_4, first for visualization purposes and, second, since the F_4 values could be estimated using the correlation depicted in Fig. 4(a) On the left part of Fig. 5, all the evaluated solutions are illustrated. On the right part of the same figure, the non-dominated Pareto front is shown, by still taking into account only the F_1, F_2 and F_3 objective function values. It consists of eight solutions which split into two groups. The left-most part of the front has low F_1 and F_2 values. The right-most part of the front yields low F_3 values.

To summarize, an extremely time-consuming, the solution of a four-objective optimization problem, which involves three 3D Navier-Stokes computations per candidate solution, was made possible using evolutionary algorithms enhanced with artificial neural networks. This is a very good example to show that EA, enhanced through surrogate evaluation models, is a promising optimization tool.



(a) F_3 v.s. F_4

(b) F_1 v.s. F_2

(c) F_1 v.s. F_4

(d) F_2 v.s. F_3

Figure 4: Solutions plotted in the objective function space. All but F_4 should be minimized, F_4 should be maximized.

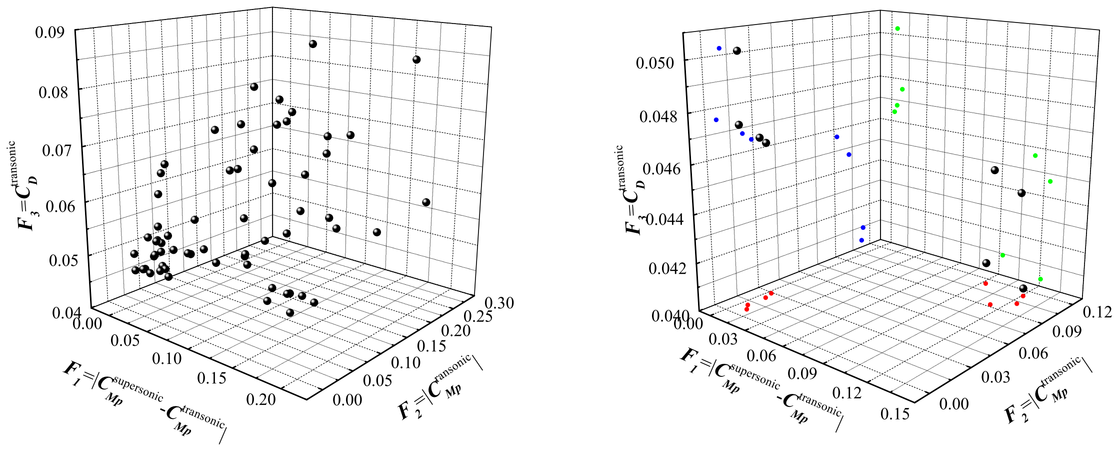


Figure 5: All solutions (left) and nondominated solutions (right) plotted in the (F_1, F_2, F_3) objective functions space.

REFERENCES

[1] K. C. Giannakoglou, A. P. Giotis and M. K. Karakasis. “Low–Cost Genetic Optimization Based on Inexact Pre–Evaluations and the Sensitivity Analysis of Design Parameters,” *Journal of Inverse Problems in Engineering*, Vol. **9**, 389–412, 2001.

[2] K. C. Giannakoglou. “Design of Optimal Aerodynamic Shapes using Stochastic Optimization Methods and Computational Intelligence,” *Progress in Aerospace Sciences*, Vol. **38**, 43–76, 2002.

[3] T. Iwata, K. Sawada and K. Kamijo. “Conceptual Study of Rocket Powered TSTO with Fly-back Booster,” AIAA paper 2003-4813, 2003.

[4] D. Sasaki, S. Obayashi and K. Nakahashi. “Navier-Stokes Optimization of Supersonic Wings with Four Objectives Using Evolutionary Algorithm,” *Journal of Aircraft*, Vol. **39**, No. 4, 621–629, 2002.

[5] D. Sasaki, G. Yang and S. Obayashi. “Automated Aerodynamic Optimization System for SST Wing-Body Configuration,” AIAA paper 2002-5549, 2002.

[6] Y. Ito and K. Nakahashi. “Direct Surface Triangulation Using Stereolithography Data,” *AIAA Journal*, Vol. **40**, No. 2, 490–496, 2002.

[7] S. Obayashi and G. P. Guruswamy. “Convergence Acceleration of an Aeroelastic Navier-Stokes Solver,” *AIAA Journal*, Vol. **33**, No. 6, 1134–1141, 1994.

[8] D. Sharov and K. Nakahashi. “Reordering of Hybrid Unstructured Grids for Lower-Upper Symmetric Gauss-Seidel Computations,” *AIAA Journal*, Vol. **36**, No. 3, 484–486, 1998.

[9] J. Dacles-Mariani. G. G. Zilliac, J. S. Chow and P. Bradshow. “Numerical/Experimental Study of a Wingtip Vortex in the Near Field,” *AIAA Journal*, Vol. **33**, No. 9, 1561–1568, 1995.

[10] Y. Ito and K. Nakahashi. “Unstructured Mesh Generation for Viscous Flow Computations,” *Proceedings of the 11th International Meshing Roundtable*, Ithaca, NY, 367–377, 2002.

[11] D.E. Goldberg.“Genetic Algorithms in search, optimization & machine learning,”, Addison-Wesley, 1989.

[12] Z. Michalewicz. “Genetic Algorithms + Data Structures = Evolution Programs,” 2nd edition, Springer-Verlag, 1994.

[13] Th. Bäck. “Evolutionary Algorithms in Theory and Practice. Evolution Strategies, Evolutionary Programming, Genetic Algorithms,”Oxford University Press, 1996.

[14] E. Zitzler, M. Laumanns, and L. Thiele. “SPEA2 Improving the Stength Pareto Evolutionary Algorithm, ”, TIK-Report No. 103, Computer Engineering and Communication Networks Lab, ETH Zurich, 2001.

[15] N. Srinivas and K. Deb. “Multiobjective optimization using nondominated sorting in genetic algorithms,” *Evolutionary Computation*, 221–248, 1994.