JOURNAL OF SPACECRAFT AND ROCKETS
Vol. 34, No. 1, January–February 1997

Three-Dimensional Aerodynamic Shape Optimization Using Genetic and Gradient Search Algorithms

Norman F. Foster*
Lockheed Martin Corporation, Schenectady, New York 12301
and
George S. Dulikravich†
Pennsylvania State University, University Park, Pennsylvania 16802

Two hybrid optimization methods used for preliminary aerodynamic design are introduced. The first is a gradient method based on Rosen's projection method and the method of feasible directions. The second technique is a genetic algorithm that uses elements of the Nelder–Mead simplex method to aid in search direction determination, as well as gradient methods to handle constrained problems. These methods are applied to three-dimensional shape optimization of ogive-shaped, star-shaped, spiked projectiles and lifting bodies in a hypersonic flow. Flowfield analyses are performed using Newtonian flow theory and, in one case, verified using a parabolized Navier–Stokes flow analysis algorithm. Three-dimensional geometrical rendering is achieved using a variety of techniques including beta splines from the computer graphics industry. In a comparison to the gradient-based method, the hybrid genetic algorithm is shown to be able to achieve impressive convergence on highly constrained problems while avoiding local minima.

Nomenclature

$f(\boldsymbol{x})$	= objective (or cost) function
fit	= cost function array for population
$g_j(\boldsymbol{x})$	= inequality constraint function
$h_k(\boldsymbol{x})$	= equality constraint function
$\boldsymbol{M}$	= population matrix
n	= number of design variables
n_{ag}	= number of active inequality constraints
n_{ah}	= number of active equality constraints
n_g	= number of inequality constraints
n_h	= number of equality constraints
n_{pop}	= number of population members (designs)
$\boldsymbol{s}$	= search direction vector
$\boldsymbol{x}$	= vector of design variables
$\boldsymbol{x}_L$	= vector of design variables' lower boundaries
$\boldsymbol{x}_U$	= vector of design variables' upper boundaries
α	= line search parameter

Introduction

AN integral part of a system's development very often includes the determination and design of the shape of some surface. The surface may be the exterior of some mechanical component, or it may be the interior of a duct system, for example. Geometric shape design may be performed through a trial-and-error process by an experienced designer and periodically tested experimentally to determine whether the design is adequate. Recently, computer assisted design and computer aided manufacturing hardware and software have provided a new approach to engineering design and development. Products that have been created using this new technology are generally superior in their performance and efficiency. The cost associated with experimental data acquisition has been brought to a minimum because most of the design's analyses can be performed computationally.

Shape design generally affords a special degree of difficulty. Determination of a surface's geometry so that some dependent parameter(s) is satisfactory involves a certain degree of intuition. The designer often needs some personal experience to predict whether the aesthetics of some shape will be functional. In an attempt to reduce the need for a designer's intuition, this study presents a tool to assist in the geometric shape design process. This tool is essentially an optimization procedure. The subjects of the optimization process are a set of design variables that uniquely define the three-dimensional surface of a body. The values for these variables need to be determined so that the whole geometry becomes better according to some criteria. This tool allows the designer to begin at some initial design, and then to achieve a more optimal design without further input.

The optimization processes presented in this study were chosen to be applied to the external geometries of bodies in a hypersonic flow. This flow regime was chosen because the flowfield solutions are not highly expensive; however, the optimization algorithms should be equally applicable to any fluid dynamics problem. The geometry of the hypersonic vehicle is optimized such that some chosen quantity is improved. This quantity generally is influenced by the aerodynamic forces on the vehicle and is thereby dependent on the geometry of the body.

Optimization of hypersonic bodies with axisymmetric cross sections has been performed in the past.[1–4] Optimization of arbitrary three-dimensional hypersonic vehicles was made feasible[5–7] by the use of an inexpensive Newtonian flow analysis method (a method that will also be used in this work) for determining aerodynamic forces. Generally, these studies used rank-two update optimization processes that could not operate well on highly constrained problems. Other works have concentrated on optimization of small regions within a larger configuration, such as nose shape,[8] airfoils,[9] wing size,[10] and scramjet engine integration.[11] This study presents two methodologies, both of which are hybrid techniques of existing methods. The second method uses a genetic algorithm in conjunction with other methods to handle constrained problems.

Optimization Problem Statement

In the most general sense, optimization is the process of achieving the best outcome of a given operation while satisfying a set of given constraints. The cost (or objective) function is the term applied to this outcome that needs to be improved (or optimized). In a computational sense, this cost function is expressed as a scalar value and it is mathematically dependent on a set of design variables. The best solution of an optimization problem would be the set of design variables such that the cost function reaches its global minimum value. The existence of constraints placed on design variables can

Received Feb. 3, 1996; revision received Aug. 9, 1996; accepted for publication Aug. 21, 1996.

*Computational Fluid Dynamics Engineer, P.O. Box 1072.
†Associate Professor, Department of Aerospace Engineering. Associate Fellow AIAA.

dramatically alter the nature, complexity, and solution method of an optimization problem. A set of design variables that does not violate any constraint is said to be feasible, whereas one that does violate a constraint is said to be infeasible. If a constraint is violated, or on the verge of violation, it is known as an active constraint.

The general constrained optimization problem can be mathematically stated as

$$\text{minimize:} \quad f(\boldsymbol{x}) \quad \boldsymbol{x} = \{x_1, x_2, \ldots, x_n\} \qquad \boldsymbol{x} : \boldsymbol{x} \in \Re^n \tag{1}$$

such that

$$\boldsymbol{x}_L \leq \boldsymbol{x} \leq \boldsymbol{x}_U \tag{2}$$

$$g_j(\boldsymbol{x}) \leq 0 \qquad j = 1, 2, \ldots, n_g \tag{3}$$

$$h_k(\boldsymbol{x}) = 0 \qquad k = 1, 2, \ldots, n_h \tag{4}$$

Hybrid Gradient Optimization

Standard gradient-based optimization techniques involve successively applying the following series of operations, which constitute one design cycle: 1) determine a search direction $\boldsymbol{s}$ in which to proceed from the current design, 2) perform a one-dimensional line search to determine a distance α along $\boldsymbol{s}$ that achieves an adequate reduction in $f(\boldsymbol{x})$, 3) update the current design by

$$\boldsymbol{x}^{\text{new}} = \boldsymbol{x}^{\text{old}} + \alpha \boldsymbol{s} \tag{5}$$

and 4) return to step 1. The presence of constraints on $\boldsymbol{x}$ can make determining $\boldsymbol{s}$ challenging especially when one or more constraints of mixed types (equality and inequality) are active. The hybrid gradient optimization scheme presented here is a simple conglomeration of two popular optimization techniques that are individually effective for certain constrained conditions. By using them together, the hybrid scheme is quite effective for handling both inequality and equality constraints.

The scheme begins by initially setting $\boldsymbol{s} = -\nabla f$. If there are no active constraints, then a line search is performed in this steepest descent direction and the design cycle ends with updating the design by Eq. (5). In the unconstrained case, $\boldsymbol{s}$ can also be determined using a conjugate direction or a higher-order rank-two procedure such as the Davidon–Fletcher–Powell update method.[12] If $n_{ag} > 0$, then $\boldsymbol{s}$ is computed by utilizing the method of feasible directions[13] (MFD). The MFD is particularly effective for determining a search direction when inequality constraints are active. The resulting search direction will be a compromise between reducing the cost function and moving away from each active constraint boundary. However, if there are active equality constraints ($n_{ah} > 0$), then the appropriate search direction should not attempt to move away from the active equality constraint boundaries, but rather follow them. Therefore, after $\boldsymbol{s}$ has been determined (either by using the steepest descent direction or by the MFD), the search direction is projected onto a design subspace tangent to any active equality constraint boundaries. This projection operation is taken from Rosen's projection method[13] (RPM). The projection is performed by

$$\boldsymbol{s}^{\text{new}} = [\boldsymbol{P}]\boldsymbol{s}^{\text{old}} \tag{6}$$

where

$$\boldsymbol{P} = \boldsymbol{I} - \boldsymbol{N}(\boldsymbol{N}^T\boldsymbol{N})^{-1}\boldsymbol{N}^T \tag{7}$$

$$\boldsymbol{N} = [\nabla h_1 | \nabla h_2 | \cdots | \nabla h_i] \qquad i = 1, 2, \ldots, n_{ah} \tag{8}$$

Once $\boldsymbol{s}$ has been determined, the design cycle proceeds with a line search to obtain α, and finally with the design update given by Eq. (5). All gradients are computed using divided difference, and the line search is based on the golden section minimization procedure.[13] No design is accepted from the line search unless it is feasible.

Hybrid Genetic Optimization

Description of the Genetic Algorithms

Genetic algorithms (GA) are nongradient methods[13,14] that offer a promising answer to complex optimization problems. In general, a GA is broken into three major steps: evaluation, crossover, and mutation. An initial population of complete design variable sets is analyzed according to some cost function. Then this population is merged using a crossover and mutation methodology to create a new population. This process continues until a global minimum is found. Generally, the design variable set that corresponds to the current minimum point will be representative of the most successful features of previous generations of designs in the optimization process. The GA can be exceptional at avoiding local minima because it tests possible designs over a large domain in the design variable space.

To begin the genetic evolutionary process, an initial population (of designs) must be defined. Each population member (design) is in itself an entire complete set of design variables. Each member in this initial population can be specified meaningfully by performing inverse design of aerodynamic shapes.[15,16] However, commonly, only the first member is specified and the rest are generated randomly. The population matrix contains the n_{pop} design vectors augmented row-wise as

$$\boldsymbol{M} = \begin{bmatrix} \text{Pop. member}_1 \\ \text{Pop. member}_2 \\ \vdots \\ \text{Pop. member}_{n_{\text{pop}}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_{n_{\text{pop}}}^T \end{bmatrix} \tag{9}$$

Once the initial population has been constructed each member is evaluated for what is referred to as its fitness, which is defined as the negative of the cost function. Thus, a fitness vector is established as

$$\textbf{fit} = \begin{Bmatrix} -f(\boldsymbol{x}_1) \\ -f(\boldsymbol{x}_2) \\ \vdots \\ -f(\boldsymbol{x}_{n_{\text{pop}}}) \end{Bmatrix} \tag{10}$$

Next, population members are selected for crossover based on their fitness. That is to say, design sets corresponding to lower cost functions possess a better chance for being selected for the crossover process. To merge the two parent design sets, each design variable of both parents is usually coded into what is referred to as bit strings. These bit strings are binary (base_2) representations of the design variables' values as percentages of their allowed values defined by their upper and lower bound constraints.[17,18] When the crossover procedure is complete, a mutation procedure is conducted in which every bit of every design variable of every child design is subject to a chance for being toggled. At this point the fitness of each child design is evaluated followed again by crossover and mutation. This cycle is continued until the best fitness design reaches some acceptable value, or until it does not change after many iterations.

The best design is carried directly forward into the next design cycle (called elitism) without crossover or mutation. This best design can still be selected for crossover. Elitism ensures that the value for the best fitness does not worsen from one generation (design cycle) to the next. However elitism might lead to narrowing of the choices and convergence to a local minimum.

GA and Constraints

The classical GA[19] can handle bounds (boundary constraints) on the design variables, but it is inherently incapable of handling equality or inequality constraint functions. Previous implementations of the GA have involved problems posed in such a way as to eliminate constraint functions, or to penalize the cost function when a constraint is violated. These treatments of constraints reduce the chance of arriving at the global minimum.

The method used for satisfying constraints during the genetic optimization process in this study utilizes the restoration move given in RPM. To restore the design to a location $\boldsymbol{x}^{\text{new}}$ lying along the

constraint boundary, a restoration move is required from x^{old} so as to reduce the values of the violated constraints to zero. This restoration move is defined as

$$x^{new} = x^{old} - N(N^T N)^{-1} g_a(x^{old}) \tag{11}$$

where g_a is a vector containing the values of the violated constraints and N contains only gradients of the violated constraints. When the constraints are nonlinear, Eq. (11) must be applied iteratively. Also if the restoration direction given by $(x^{new} - x^{old})$ will immediately worsen any active constraint, then the restoration direction is projected tangent to them using Eq. (6). This restoration is performed until all violated constraints are simply active. Whereas implementing feasible searches is not a method of transforming the classical GA into a true constrained algorithm, it can effectively lead the GA to constrained solutions. This is because, before a new generation of designs is produced, all of its parent designs have been proven to be feasible. By doing so, the GA can be altered so that it can handle nonlinear constraint functions, which is not at all common for the classical GA.

If an entire set of newly generated designs has been analyzed without any improvement to the best current design, then a search direction is determined according to the Nelder–Mead method.[20] This search direction is projected tangent to any constraints that will be immediately violated. A line search is then performed in this search direction. The use of the nongradient Nelder–Mead search direction provides a reasonable direction in which to pursue when the GA has otherwise yielded no improvement in the design cycle.

Geometry Treatment

The external geometry of the hypersonic vehicle configuration constitutes the design variables. Therefore, the entire geometry must be represented by a single vector of scalar quantities. In this study, the method for describing the three-dimensional vehicle geometry was done in several ways depending on the type of problem considered. However, in all cases the surface nodes did not move axially along the length of the body. All points were grouped onto cross-sectional planes, and the total length of the vehicle was always preserved. Each surface point location was specified as to how it can vary on its cross section. Examples of different kinds of point motions include Cartesian and polar motion, and motion according to a chosen spline technique.[17]

There is a parametric piecewise curve representation method that uses what are called beta splines. These beta splines[21] have been used for geometric computer modeling and are closely related to the more common v splines and B splines or Bernstein polynomials. A beta spline curve is specified by a set of points called control vertices. The positions of these vertices completely define the shape of the dependent curve (beta spline), although they generally do not lie on it. The vertices are an ordered sequence of points that form what is referred to as the control polygon. The beta spline utilizes a piecewise representation, to achieve local control, by defining each segment as a function of only a few adjacent vertices. Specifically, each curve segment can be regarded as a weighted average of four local vertices. Let

$$G_i(u) = \begin{Bmatrix} x_i(u) \\ y_i(u) \end{Bmatrix} \tag{12}$$

denote the ith parametric two-dimensional beta curve segment, where u is a nondimensional curve following coordinate that varies from 0 at the beginning of the segment to unity at the end of the segment. In the case of a cubic beta spline, this segment is dependent on four neighboring vertices according to

$$G_i(u) = \sum_{r=-2}^{1} b_r(\beta_1, \beta_2; u) V_{i+r} \qquad (0 \le u < 1) \tag{13}$$

where V_{i+r} is the $(i+r)$th control vertex position vector. The scalar weighting factors in Eq. (13), $b_r(\beta_1, \beta_2; u)$, are called basis functions. These basis functions depend on the domain parameter u and two shape parameters β_1 and β_2. Each basis function is itself defined as a cubic polynomial,

$$b_r(\beta_1, \beta_2; u) = \sum_{j=0}^{3} c_{j,r}(\beta_1, \beta_2) u^j \tag{14}$$

In Eq. (14), there are 16 unknown constants, $c_{j,r}(\beta_1, \beta_2)$, where $0 \le u < 1$, $j = 0, 1, 2, 3$, and $r = -2, -1, 0$, and 1. These constants are fixed quantities (provided β_1 and β_2 are fixed), which can be found by imposing the following three connectivity boundary conditions on any two neighboring segments:

$$G_{i+1}(0) = G_i(1) \tag{15}$$

$$\frac{dG_{i+1}(0)}{du} = \beta_1 \frac{dG_i(1)}{du} \tag{16}$$

$$\frac{d^2 G_{i+1}(0)}{du^2} = \beta_1^2 \frac{d^2 G_i(1)}{du^2} + \beta_2 \frac{dG_i(1)}{du} \tag{17}$$

The first boundary condition [Eq. (15)] enforces simple connectivity. The second boundary condition [Eq. (16)] is derived from enforcing continuity of the unit tangent vector at the joint of two segments. The last boundary condition [Eq. (17)] comes from imposing continuity of the curvature vector (and thus curvature) at the connecting point of two adjacent segments.

The shape parameter β_1 is referred to as the bias parameter. For purposes of the present study, the bias parameter was set to unity. The other shape parameter, β_2, is called the tension parameter and should always be positive. For high values of β_2, the beta curve will be strongly attracted to the control vertices, and in the limit, will be identical to the control polygon. In this study, β_2 was also fixed at unity.

Implementing beta splines for the control of the geometry in this study was accomplished by specifying the number of control vertices on each of the x, y parallel planes. The locations of the control vertices are defined to be the design variables to be optimized.

Results

A single computer program was produced that performed the three-dimensional hypersonic vehicle shape optimization. Two optimization techniques were included: the hybrid gradient and the hybrid genetic techniques. Also, two analysis methods were included: modified Newtonian impact theory (MNIT) and a parabolized Navier–Stokes (PNS) solver. Either optimizer could use either flow solver at any point during execution.

In most of the cases that follow (except for verification case 1, which did not utilize the optimizers), two inequality constraints were imposed, in addition to any other constraints that are mentioned. It is convenient to explain them here. The first, which will be called the no twist constraint, simply prevents the optimizer from evaluating a design that has an overlap in the cross-sectional geometry. The value of the inequality constraint function is equal to the area that is twisted or pinched off, which should be less than or equal to zero. The second will be called the minimum thickness constraint. This constraint is defined as a tolerance of 3 mm, minus the shortest distance from any surface node to any other surface node or panel center. This constraint effectively keeps any part of the geometry cross section from becoming too jagged or spiked.

Verification Test 1: Half-Sphere-Cone Analysis

The accuracy of our MNIT code was compared to the results from a series of hypersonic wind-tunnel experiments[22] that were performed on three-dimensional models called half-sphere-cones. Aerodynamic measurements were taken for a wide range of angles of attack, for several different values of Mach number. This test case attempts to duplicate these experimental measurements using the MNIT flow solver.

One particular set of experimental data was chosen for comparison. The geometry of the chosen half-sphere-cone is shown in Fig. 1. The radius of curvature of the hemispherical nose measured 0.365 in. The diameter of the semicircular rear plane measured 2.43 in. The

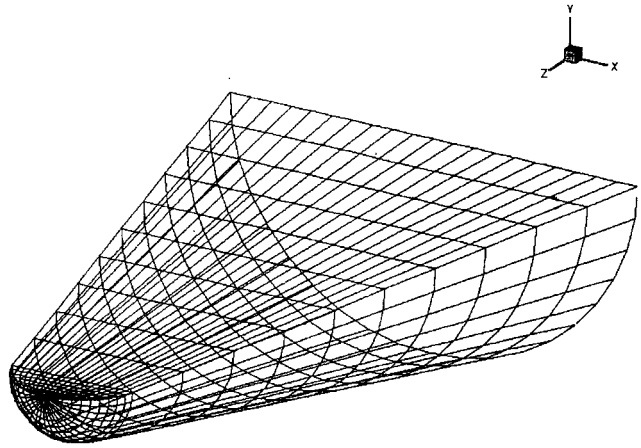

Fig. 1 Half-sphere-cone geometry.

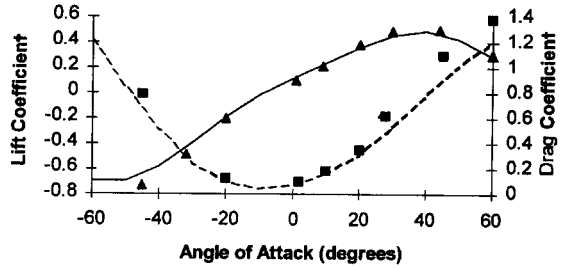

Fig. 2 Computed and experimental lift and drag coefficients: ——, computed lift coefficient; ▲, experimental lift coefficient; – – –, computed drag coefficient; and ■, experimental drag coefficient.

total length of the model measured 4.0 in., and the planform area was 6.025 in.^{2}. This model was subjected to a flow at a Mach number of 12.6; the Reynolds number (based on model length) was 4.9×10^5.

These test conditions were duplicated and analyzed using the MNIT flow solver. The MNIT analyses were performed for angles of attack ranging from −60 to 60 deg, stepping every 10 deg. Figure 2 displays MNIT computed and experimental values for lift and drag coefficients. These coefficients use planform area and a dynamic pressure of 2320.78 psf for normalization. The computed results shown in Fig. 2 show good agreement with the experimental data, although MNIT represents an oversimplified model of physics.

Verification Test 2: Wave Drag Minimization

This test case is intended to validate the operation of the optimization techniques. In this exercise, the geometry of axisymmetric body was optimized to reduce wave drag at zero angle of attack. Optimal bodies of revolution that minimize drag have previously been analytically determined. Two such solutions are known as the von Kármán and Sears–Haack bodies.[23] These two bodies yield the minimum wave drag under two different sets of assumptions. The von Kármán body assumes that the body terminates with a flat plane, that the base area in this plane is known, and that the total length of the body is specified. The Sears–Haack body assumes that the body is pointed at both ends (a spindle), that the total volume is known, and that the total length of the body is given. In our test case only the front-half of this body is of interest.

The hybrid gradient optimizer was used to determine computationally the body of revolution that minimizes wave drag at Mach = 10 and an altitude of 18 km. The MNIT was used for the flowfield analyses. Initially, the body was specified to be a 10-m long, 15-deg angle right circular cone. The design variables for this exercise were specified to be the radii of the body at 10 cross sections. Each design variable (the cross-sectional radii) was allowed to vary from 0 to 10 m. During the optimization process, the total volume of the body was constrained (with an equality constraint) not to change by more than 1.0 m^3 from its initial value. In addition, the no twist and minimum thickness inequality constraints were imposed. The optimization process converged to the bulged axisymmetric body

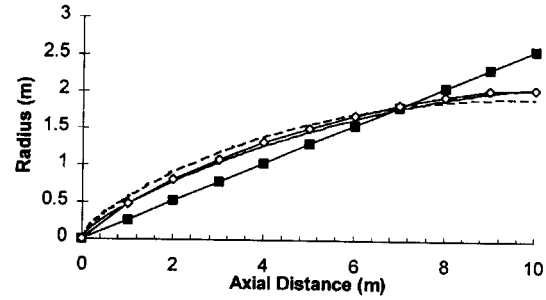

Fig. 3 Profile comparison of initial, optimized, and analytically optimal geometries: ■, initial cone; ◇, optimized body; ——, von Kármán ogive; and – – –, Sears–Haack spindle.

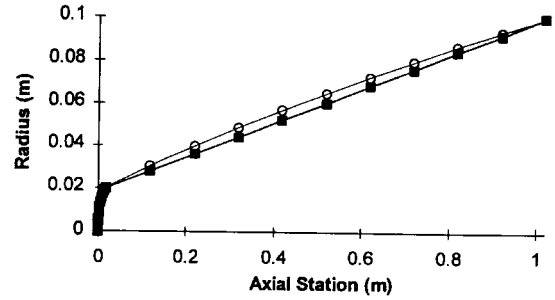

Fig. 4 Profile comparison of initial, optimized, and analytically optimal geometries: total length and center of aerodynamic pressure specified; spherical nose, length, and rear area were fixed: ■, initial; ○, optimized; and ——, analytic.

called an ogive that reduced the drag by 42%. Figure 3 displays plots of the profiles of the initial and final (optimized) bodies. The base area of the optimized body, and the total volume (fixed) were used to compute von Kármán and Sears–Haack bodies. The profiles of these analytically optimal bodies of revolution are also plotted in Fig. 3. Inspection of Fig. 3 shows the computed optimized body to be in excellent agreement with the analytic bodies.

Verification Test 3: Specified Center of Pressure

In this exercise, an initial body was optimized to minimize drag at zero angle of attack using the hybrid gradient optimizer and the MNIT flow solver. The initial body is a right circular cone with a spherical nose. The initial profile of the body was linear and tangent to the spherical nose.

The optimization problem was to find the optimal body that minimizes drag such that the total length of the body was fixed, the base area and radius were fixed, the radius of curvature of the nose was fixed, and the aerodynamic center of pressure x_{cp} was specified. The center of pressure was preserved using an equality constraint. The design variables were the radii of the nine cross sections between the spherical nose and the base. The spherical nose was described by 10 fixed cross sections.

The analytic solution to this problem has been previously determined[1] using Newton impact theory assumptions. The optimization process converged to the body shown in Fig. 4, which also shows profiles of the initial, the optimized, and the analytically optimal body. This figure shows an excellent agreement between the computed and the analytically optimum bodies.

Exploratory Case 1

In case 1, an initial body was optimized to minimize wave drag at zero angle of attack using the hybrid gradient optimizer and the MNIT flow solver. The initial shape was a right circular cone described earlier. The shape of the body was optimized holding its total length fixed. Total volume was constrained to remain constant (using an equality constraint), and the no twist and the minimum thickness inequality constraints were imposed. The geometry was described by 6 cross-sectional planes with 40 nodes on each plane. All of the surface nodes on the first cross section moved together radially and were controlled by one design variable. On the other

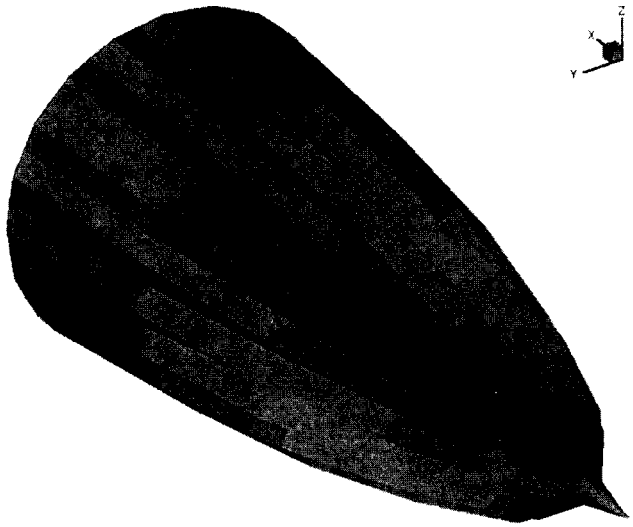

Fig. 5 Optimized star-shaped body: drag minimized using hybrid gradient technique; initial geometry was a 15-deg circular cone, 391 design variables, Mach = 10, and altitude = 18 km.

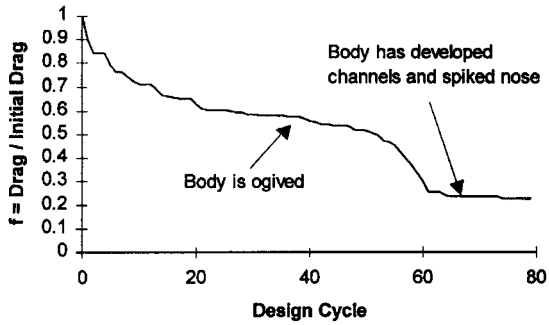

Fig. 6 Convergence history for the nonlifting body.

five planes, all of the surface nodes had two degrees of freedom except for the seam points that were allowed to move only vertically (in the y direction) in the symmetry plane. The total number of design variables was 391. Figure 5 shows the final optimized body. The final configuration reduced drag by 77% over 75 design cycles and called the flow solver 60,001 times. The execution took 4282 s on a Cray C-90 computer with a single processor. Figure 6 displays the hybrid gradient optimizer convergence history for this case.

The initial geometry and the final optimized geometry were then analyzed using the PNS flow solver.[24] The size of the computational grid was 240 circumferential cells $\times$ 30 radial cells $\times$ 200 longitudinal cells. The freestream temperature, density, and Mach number were specified to be 218 K, 0.1206 kg/m^3, and 10, respectively. A nonequilibrium gas model and Wilkes mixing rule were used in the PNS flow analysis. The total drag of the optimized body, as computed by the PNS flowfield solver, was found to be 53% lower than the drag of the initial conical geometry.

The predominant characteristic of the optimized geometry is the deep channels that formed along the length of the body. Star-shaped hypersonic projectile shapes have been studied experimentally in the past.[25,26] Russian researchers[27] determined that star-shaped bodies are optimal (have lowest drag) at high Mach numbers and at altitudes below 90 km because of their shock capturing characteristics. Another characteristic of the optimal body shown in Fig. 5 is the needled nose. The conical shape of the nose was enforced because geometry of the first cross section was specified to be dependent on one design variable: radial distance from the axis. To reduce drag, the optimizer could only reduce the inclination angle of these panels by reducing the radius of the first cross section and adding the resulting lost volume by increasing the size of the spikes on the aft body. Aerodynamic bodies that have needled noses have also been studied.[28,29] The fact that the nose was driven by the optimizer to a slender cone rather than to a wide cone is interesting in light of these previous studies.

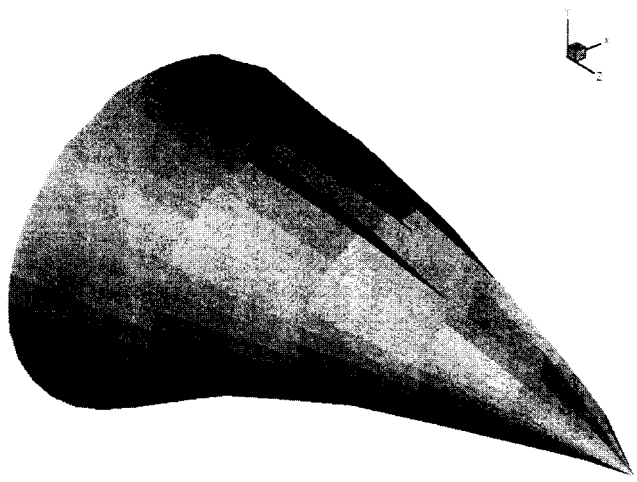

Fig. 7 Optimized lifting body: drag minimized using hybrid gradient technique; initial geometry was a 15-deg circular cone, 240 design variables, Mach = 10, and altitude = 18 km.

Exploratory Case 2

An initial body was optimized to maximize the lift to drag ratio at zero angle of attack using the MNIT flowfield solver. For this case, the initial shape was specified to be a 15-deg right circular cone, identical to the one in exploratory case 1. The shape of the body was optimized holding its total length and volume (equality constraint) fixed. The no twist and minimum thickness inequality constraints were also enforced. There were 40 points on each of 6 cross-sectional planes. Every surface node was specified to vary radially on its cross-sectional plane. The hybrid gradient optimizer was executed for 40 design cycles that spent 1458 CPU seconds on a Cray C-90 and required 19,961 objective function analyses. Figure 7 shows the final optimized body having $L/D = 1.29$. This geometry is cambered and has ridges that have formed on its upper surface. The optimizer cambered the body so that a greater surface area on the underside faced the freestream so as to increase the lift, and formed ridges on top of the body so that downward pressure was minimized. Yet the body still has an ogive appearance, which helps to reduce overall drag.

Exploratory Case 3

In this case, everything was the same as in exploratory case 2 (including the constraints), except that the geometry for this case was entirely described using beta splines. Therefore, the design variables for this case were specified to be the locations of the beta spline control vertices, and the actual geometry surface was generated by computing the beta splines, which depend on the control vertices. Only one-half of the geometry was optimized, and the other half was mirrored across the vertical plane of symmetry where the slope of the geometry was specified to be perpendicular to the plane of symmetry. Initially, the geometry was modeled using only one beta spline on each half cross section. This modeling required $6 \times 4 = 24$ design variables for the entire geometry. The hybrid gradient optimizer was used until convergence was reached, at which point the converged geometry was redefined using two beta splines on each half cross section, and the optimization process was restarted. This redefined geometrical model required $6 \times 8 = 48$ design variables for the entire geometry. Once again, when convergence was reached, more degrees of freedom were added to the problem. This entire cycle was repeated twice more: first with $6 \times 24 = 144$ design variables and finally with $6 \times 40 = 240$ design variables for the entire geometry. Figure 8 shows the convergence history for the entire process. Figure 9 shows the converged geometry for the final stage of the procedure. The final lift to drag ratio was $L/D = 1.77$, and the entire optimization process required a total of 5288 objective function analyses and a total of 141 CPU seconds on a Cray C-90. The results from this case demonstrate that the use of beta splines in geometry

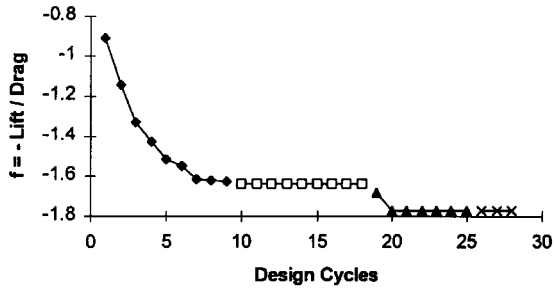

Fig. 8 Convergence history of hybrid gradient technique: number of beta splines progressively increased; ♦, number of design variables (NDV) = 24; □, NDV = 48; ▲, NDV = 144; and ×, NDV = 240.

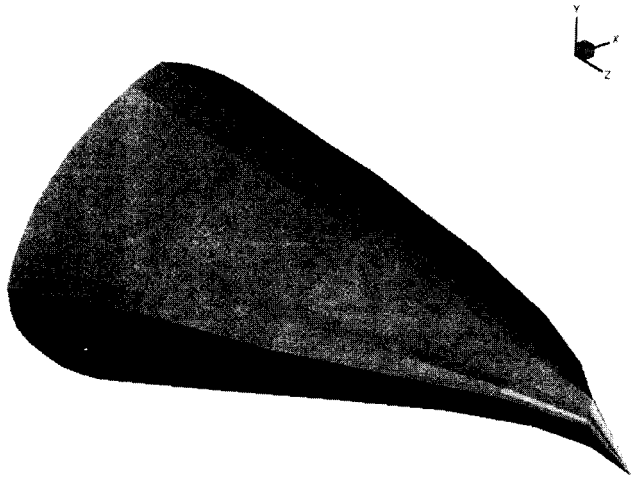

Fig. 9 Optimized geometry (lift/drag maximized) using 240 design variables and the hybrid gradient technique.

definition can lead to an improved design costing fewer objective function analyses. Because fewer design variables were used at first, the optimizer was able to quickly determine the general features of the optimal design. After progressively adding degrees of freedom to the geometry, the final solution had a comparatively high L/D and was smooth and realistic looking.

Exploratory Case 4

This case is exactly the same as exploratory case 3 (including the constraints) except the hybrid genetic optimizer was used for the entire execution and the geometry was only redefined once. The initial shape was modeled by one beta spline on each half cross section ($6 \times 4 = 24$ design variables). This case was executed for 600 design cycles requiring 15,287 objective function analyses and 2660 CPU seconds on a Cray C-90. After convergence was reached, the geometry (Fig. 10) was redefined using two beta splines on each half cross section ($6 \times 8 = 48$ design variables) and the optimization process was continued for an additional 129 design cycles requiring 3122 objective function analyses and 2355 CPU seconds. The final optimized geometry is shown in Fig. 11, depicting a shoveled or spatulate nose typical of a low-drag geometry on hypersonic waveriders. The fact that the geometry did not completely smooth out to such a shape probably is due in large part to only having six cross-sectional planes to vary. An even more shoveled nose may have been produced if more planes were allowed to vary clustered near the front of the geometry.

The final value for the lift to drag ratio was $L/D = 2.4527$. This value for L/D is higher than that of any other case in this study. Interestingly, this case is modeled with the fewest number of geometric design variables (48 for the entire body, using beta splines). This case shows the ability of the hybrid genetic optimizer to effectively find optimal designs. Note that the point at which more degrees of freedom were added to the problem and the point at which the entire execution was stopped were chosen by inspection of the convergence history. Unlike gradient methods, the hybrid (or any) genetic optimization technique does not generally exhibit

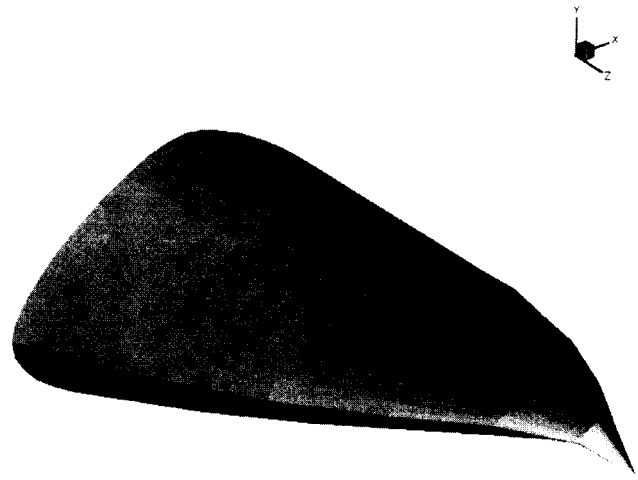

Fig. 10 Optimized geometry (lift/drag maximized) using 24 design variables and the hybrid genetic technique.

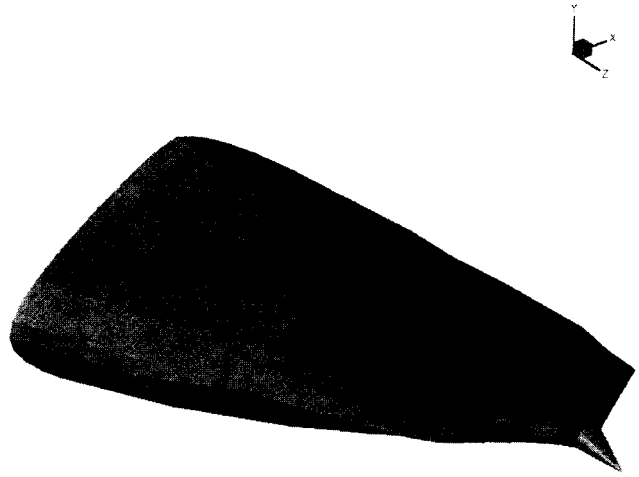

Fig. 11 Optimized geometry (lift/drag maximized) using 48 design variables and the hybrid genetic technique.

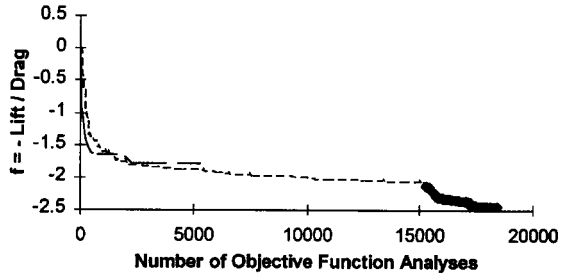

Fig. 12 Convergence comparison of both hybrid techniques: — —, hybrid gradient [number of design variables (NDV) = 24, 48, 144, 240]; – – –, hybrid genetic (NDV = 24); and ◇, hybrid genetic (NDV = 48).

smooth convergence characteristics. Rather, they may show slow (or no) improvement on the design over many design cycles; then suddenly dramatic improvement may occur. The implementation of a Nelder–Mead search direction calculation, as was done in the current hybrid genetic method, can significantly improve convergence regularity. Even so, there is no way to know whether convergence has actually been reached or whether a large design improvement is only a few design cycles away, when using most nongradient methods.

Figure 12 displays the decrease of the cost functions for this case and exploratory case 3 plotted together against the number of objective function analyses. This figure shows that the hybrid gradient optimizer initially reduces the cost function with fewer cost function analyses; however, the hybrid genetic optimizer quickly surpasses the gradient method and obtains designs that have a higher

L/D than any design that the gradient method ever achieves. This indicates that the hybrid GA avoided a local minimum and that the gradient method could not.

Conclusions

Results from this study have indicated that hybrid nongradient methods can be effectively applied to difficult engineering problems. A hybrid GA that has been shown to achieve convergence rates comparable with more standard gradient techniques was developed. Furthermore, the hybrid GA was able to explore design configurations that gradient methods would not. The hybrid GA presented was also able to simultaneously adhere to inequality and equality constraints.

When the hybrid GA is applied to shape optimization in which the objective function depends on aerodynamic properties computed from a flow analysis, it is advantageous to describe the variable geometry using local control spline techniques. The use of beta splines was shown to effectively lead to optimized geometry configurations when either technique was used.

Both the MNIT and the PNS flowfield analysis methods can be used with either of the optimization techniques. It was found, however, that the PNS flow solver could be used in the optimization cycle only in restricted cases. During the course of the optimization procedure, intermediate design geometries that may cause areas of recirculation in the flowfield need to be analyzed. The PNS flow solver will become unstable and fail during the flow analysis for such geometries. Therefore, additional geometrical constraints might be required to keep intermediate designs adequately smooth so that flow analyses will maintain stability. Alternatively, the PNS flowfield solver can be used to verify the final results that are obtained using the MNIT flow analysis method, as was done in this study.

The MNIT flow solver was used extensively in this study. However, because of its simplicity the MNIT does not account for some of the important physical phenomena that actually occur in the flowfield. As a result, the optimizer may arrive at a design that would be nonsensical if all of the physical phenomena were taken into consideration. The optimizer is not dependent on the flow solver that is utilized, but will search for an optimal design regardless of the accuracy of the procedures used to compute the objective function. For this reason it would be desirable to use a flow solver that accounts for the true physical phenomena in the flowfield, such as a full Navier–Stokes flowfield solver, with an optimizer that requires the fewest number of objective function analyses. Nevertheless, as a preliminary design tool, the hybrid techniques presented show great promise in finding optimized solutions to constrained problems.

From the results in this study, it can be seen that the internal design update logic of the hybrid genetic optimization method is much more expensive in CPU time than that in the hybrid gradient method. However, the hybrid GA evaluates the objective function considerably fewer times. Therefore, if a very expensive flow solver, such as a full Navier–Stokes flow solver, were utilized then the hybrid genetic method might be superior as far as CPU expense is concerned. It has been seen that the hybrid genetic method can also arrive at designs that the gradient method cannot, because the GA is very effective at avoiding local minima. For these reasons, hybrid genetic optimization techniques, such as the one presented in this study, are becoming widely used to solve difficult multidisciplinary engineering problems.

Acknowledgments

The authors would like to express their appreciation for the advice and help provided by Isaiah Blankson of NASA headquarters and Gordon Blom and Gregory Molvik of Boeing Company, and to NASA Ames Research Center for the computing time on the National Aerodynamics Simulation Cray C-90 computer.

References

[1]Strand, T., "Design of Missile Bodies for Minimum Drag at Very High Speeds—Thickness Ratio, Lift, and Center of Pressure Given," *Journal of the Aerospace Sciences*, Vol. 29, No. 9, 1959, pp. 568–570.

[2]Shipilin, A. V., "Optimal Shapes of Bodies with Attached Shock Waves," *Mekhanika Zhidkosti i Gaza*, Vol. 1, No. 5, 1966, pp. 9–13.

[3]Brown, L. B., "Axisymmetric Bodies of Minimum Drag in Hypersonic Flow," Rept. U-Tex-EMRL-TR-1016, Univ. of Texas, Austin, TX, July 1967.

[4]Dulikravich, G. S., Buss, R. N., Strang, E. J., and Lee, S., "Aerodynamic Shape Optimization of Hypersonic Missiles," AIAA Paper 90-3073, Aug. 1990.

[5]Dulikravich, G. S., and Sheffer, S. G., "Aerodynamic Shape Optimization of Arbitrary Hypersonic Vehicles," *Proceedings of the 3rd International Conference on Inverse Design Concepts and Optimization in Engineering Sciences*, ICIDES-III, edited by G. S. Dulikravich, Washington, DC, 1991, pp. 347–358.

[6]Dulikravich, G. S., and Sheffer, S. G., "Aerodynamic Shape Optimization of Hypersonic Configurations Including Viscous Effects," AIAA Paper 92-2635, June 1992.

[7]Sheffer, S. G., and Dulikravich, G. S., "Constrained Optimization of Three-Dimensional Hypersonic Vehicle Configurations," AIAA Paper 93-0039, Jan. 1993.

[8]Large, E., "Nose Shape for Minimum Drag in Hypersonic Flow," *Journal of the Aerospace Sciences*, Vol. 39, No. 1, 1962, pp. 98, 99.

[9]Thompson, R. A., and Hull, D. G., "Hypersonic Airfoils of Maximum Lift-to-Drag Ratio," Rept. U-Tex-AMRL-TR-1009, Univ. of Texas, Austin, TX, Sept. 1969.

[10]Wilhite, A. W., "Optimum Wing Sizing of a Single-Stage-to-Orbit Vehicle," *Journal of Spacecraft and Rockets*, Vol. 20, No. 2, 1983, pp. 115–121.

[11]O'Neill, M. K., and Lewis, M. J., "Optimized Scramjet Integration on a Waverider," *Journal of Aircraft*, Vol. 29, No. 6, 1992, pp. 1114–1121.

[12]Fletcher, R., and Powell, M. J. D., "A Rapidly Convergent Descent Method for Minimization," *Computer Journal*, Vol. 6, 1963, pp. 163–168.

[13]Haftka, R. T., and Gürdal, Z., *Elements of Structural Optimization*, 3rd ed., Kluwer Academic, Boston, MA, 1992.

[14]Goldberg, D. E., *Genetic Algorithms in Search, Optimization and Machine Learning*, 1st ed., Addison–Wesley, Reading, MA, 1989.

[15]Dulikravich, G. S., "Aerodynamic Shape Design and Optimization: Status and Trends," *Journal of Aircraft*, Vol. 29, No. 5, 1992, pp. 1020–1026.

[16]Dulikravich, G. S., "Shape Inverse Design and Optimization for Three-Dimensional Aerodynamics," AIAA Paper 95-0695, Jan. 1995.

[17]Foster, N. F., "Shape Optimization Using Constrained Genetic Evolution and Gradient Search Algorithms," M.S. Thesis, Dept. of Aerospace Engineering, Pennsylvania State Univ., University Park, PA, Aug. 1995.

[18]Foster, N. F., Dulikravich, G. S., and Bowles, J., "Three-Dimensional Shape Optimization Using Genetic Evolution and Gradient Search Algorithms," AIAA Paper 96-0555, Jan. 1996.

[19]Crispin, Y., "Aircraft Conceptual Optimization Using Simulated Evolution," AIAA Paper 94-0092, Jan. 1994.

[20]Nelder, J. A., and Mead, R., "A Simplex Method for Function Minimization," *Computer Journal*, Vol. 7, No. 1, 1965, pp. 308–313.

[21]Barsky, B. A., *Computer Graphics and Geometric Modeling Using Beta-Splines*, 1st ed., Springer–Verlag, Berlin, 1988.

[22]Geiger, R. E., "Experimental Lift and Drag of a Series of Glide Configurations at Mach Numbers 12.6 and 17.5," *Journal of the Aerospace Sciences*, Vol. 39, No. 4, 1962, pp. 410–419.

[23]Ashley, H., and Landahl, M., *Aerodynamics of Wings and Bodies*, 1st ed., Addison–Wesley, Reading, MA, 1965.

[24]Molvik, G. A., "A Computational Model for the Prediction of Hypersonic, Reacting Flows," Ph.D. Thesis, Mechanical Engineering, Pennsylvania State Univ., University Park, PA, Dec. 1989.

[25]Vedernikov, Y. A., Gonor, A. L., Zubin, M. A., and Ostapenko, N. A., "Aerodynamic Characteristics of Star-Shaped Bodies at $M = 3$–5," *Izvestiya Akademii Nauk SSSR, Mekh. Zhidk. Gaza*, No. 4, 1981, p. 88.

[26]Gusarov, A. A., Dvoretskii, V. M., Ivanov, M. Y., Levin, V. A., and Chernyl, G. G., "Theoretical and Experimental Investigation of the Aerodynamic Characteristics of Three-Dimensional Bodies," *Izvestiya Akademii Nauk SSSR, Mekh. Zhidk. Gaza*, No. 3, 1979, p. 97.

[27]Bunimovich, A. I., and Kuz'menko, V. I., "Aerodynamic and Thermodynamic Characteristics of Three-Dimensional Bodies in a Rarefied Gas," *Izvestiya Akademii Nauk SSSR, Mekh. Zhidk. Gaza*, No. 4, 1982, pp. 181–183.

[28]Holden, M., "Separated Flow Studies at Hypersonic Speeds. Part I—Separated Flows over Axisymmetric Spiked Bodies," Cornell Aeronautical Lab. Inc., Rept. AF-1285-A-13 (I), 1964.

[29]Bogdonoff, S. M., and Vas, I. E., "Preliminary Investigations of Spiked Bodies at Hypersonic Speeds," *Journal of the Aerospace Sciences*, Vol. 29, No. 2, 1959, pp. 65–74.

T. C. Lin
Associate Editor