AIAA JOURNAL
Vol. 39, No. 12, December 2001

Detection and Repair of Poorly Converged Optimization Runs

Hongman Kim*
Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061-0203
Melih Papila†
University of Florida, Gainesville, Florida 32611-6250
William H. Mason‡
Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061-0203
Raphael T. Haftka§
University of Florida, Gainesville, Florida 32611-6250
and
Layne T. Watson¶ and Bernard Grossman**
Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061-0203

The use of iteratively reweighted least squares (IRLS) for detecting design points where optimizations converge poorly is demonstrated. Because optimization error tends to be one-sided with poor results producing overweight designs, a nonsymmetrical version of IRLS (NIRLS) that accounts for the asymmetry in optimization errors is also developed. A parameterized Rosenbrock function problem, where a programming error caused poor optimizations, is used to demonstrate the techniques. The identified outliers were repaired by more accurate optimizations to improve response surface approximations. Structural optimization of a high-speed civil transport (HSCT) produced inaccurate wing structural weight. Optimization studies with various sets of convergence criteria showed that the optimization errors were due to incomplete convergence. The IRLS technique could identify many of the points with very large optimization errors, but the NIRLS techniques were much more reliable in this task. However, results of HSCT configuration optimizations using wing weight response surface models show that it may be preferable to identify and repair only a subset of the poorly converged optima.

I. Introduction

OPTIMIZATION procedures often produce poor results due to algorithmic difficulties, software problems, or local optima. When a single optimization is flawed, it may be difficult to detect the problem, because the ill conditioning responsible for the problem may make it difficult to apply optimality criteria unambiguously. However, in many applications, a large number of optimizations are performed for a range of problem parameters. For example, our research group routinely performs thousands of structural optimizations of wing structures for the purpose of developing equations that predict wing structural weight as a function of the configuration of a high-speed civil transport (HSCT).[1] When such multiple optimizations are available, statistical analysis can detect poorly calculated optima. For example, Papila and Haftka[2,3] used iteratively reweighted least squares[4,5] (IRLS) to detect such points, called outliers. They repaired these outliers by reoptimization from slightly different starting points, by switching optimization algorithms, or by adjusting convergence criteria.

Poor optimization results may be due to local optima or premature convergence. For structural optimization of the HSCT, we have found that poor results are mostly caused by premature convergence due to the choice of the convergence parameters in the optimization procedure. An objective of this paper is to demonstrate the usefulness of outlier detection for discovering the poor choice of parameters and for estimating the magnitude of the resulting errors.

The outlier points (incorrectly computed optima) are consistently heavier than the true optimal designs, that is, the optimization error has positive bias. There are two implications of this bias in the optimization error. First, we expect that the incorrect optima will be lowered by repair. Second, the symmetric weighting function used in IRLS procedures can be improved by taking into account the error bias. One objective of this paper is to demonstrate this approach of using a biased weighting function in an IRLS procedure for detecting outliers. The procedure is first demonstrated on a parameterized five-dimensional Rosenbrock function problem and then applied to structural optimization of wing box elements for various aerodynamic configurations of an HSCT.

Section II introduces the IRLS technique along with various weighting functions including an asymmetric function leading to nonsymmetric IRLS (NIRLS). Section III demonstrates IRLS/NIRLS techniques on the Rosenbrock test function, where optimization error occurred due to a programming error. Section IV describes our application problem, the HSCT configuration design, and the associated structural optimization. Section V shows the effects of convergence parameters on the errors in optimal wing structural weight for the HSCT problem. Section VI compares the success of IRLS and NIRLS for identifying bad optimization results. Also, the effects of outlier repair on the accuracy of response surface approximations are discussed. Section VII shows the effects of outlier repair on the HSCT optimization results using wing weight response surface surrogate models. Section VIII offers concluding remarks.

Received 30 September 2000; revision received 3 June 2001; accepted for publication 6 June 2001.

*Graduate Research Assistant, Multidisciplinary Analysis and Design Center for Advanced Vehicles, Department of Aerospace and Ocean Engineering. Student Member AIAA.

†Graduate Research Assistant, Department of Aerospace Engineering, Mechanics and Engineering Science. Student Member AIAA.

‡Professor, Multidisciplinary Analysis and Design Center for Advanced Vehicles, Department of Aerospace and Ocean Engineering. Associate Fellow AIAA.

§Distinguished Professor, Department of Aerospace Engineering, Mechanics and Engineering Science. Fellow AIAA.

¶Professor, Multidisciplinary Analysis and Design Center for Advanced Vehicles, Departments of Computer Science and Mathematics.

**Professor and Department Head, Multidisciplinary Analysis and Design Center for Advanced Vehicles, Department of Aerospace and Ocean Engineering. Associate Fellow AIAA.

II. IRLS

The standard least-squares procedure for response surface estimation[6] is not robust with respect to bad data. Robust regression methods give reasonable estimates even if the data are contaminated with outliers (bad data). In addition, robust regression[7] can locate the outliers, allowing us to examine the outliers and repair them whenever possible.

To describe the robust regression method used here, known as M estimator, we first consider the standard linear regression model. With n data points and p parameters in the regression model, the linear regression model is

$$\boldsymbol{y} = X\boldsymbol{\beta} + \varepsilon \tag{1}$$

where X is an $n \times p$ Gramian matrix of the shape functions used in the model (typically monomials), $\boldsymbol{\beta}$ is a p vector of coefficients to be estimated, and ε is an n vector of uncorrelated random errors with constant variance σ^2.

For standard least squares, the expected value of $\boldsymbol{y}$ is estimated as

$$\hat{\boldsymbol{y}} = X\hat{\boldsymbol{\beta}} = X(X^T X)^{-1} X^T \boldsymbol{y} \tag{2}$$

and residual error $\boldsymbol{e}_r$ is the difference between the actual data and the estimate:

$$\boldsymbol{e}_r = \boldsymbol{y} - \hat{\boldsymbol{y}} \tag{3}$$

The root mean square error (RMSE) $\hat{\sigma}$ is the unbiased estimate of σ:

$$\hat{\sigma} = \sqrt{\boldsymbol{e}_r^T \boldsymbol{e}_r \big/ (n-p)} \tag{4}$$

For a more robust estimate of $\boldsymbol{\beta}$, we minimize a measure of the residuals r_i, where $r_i = (\boldsymbol{y} - X\hat{\boldsymbol{\beta}})_i / s$:

$$e(\hat{\boldsymbol{\beta}}) = \sum_{i=1}^{n} \rho(r_i) \tag{5}$$

Here, ρ is a function of the residual scaled by s, a known estimate of the standard deviation of ε. For example, in the case of the familiar least-squares method, $\rho(\xi) = \xi^2/2$.

The ψ function is defined as the derivative of ρ:

$$\psi(\xi) = \frac{\mathrm{d}\rho(\xi)}{\mathrm{d}\xi} \tag{6}$$

Then, a necessary condition for a minimum $\nabla e(\hat{\boldsymbol{\beta}}) = 0$, where $\boldsymbol{\psi} = (\psi_1, \psi_2, \ldots, \psi_n)^T = [\psi(r_1), \psi(r_2), \ldots, \psi(r_n)]^T$, becomes

$$X^T \boldsymbol{\psi} = 0 \tag{7}$$

To write Eq. (7) in a form of weighted least squares, we define the weighting function as

$$w(\xi) = \psi(\xi)/\xi \tag{8}$$

Now, with $\boldsymbol{r} = (r_1, \ldots, r_n)^T$, Eq. (7) becomes

$$X^T W(\boldsymbol{r})\boldsymbol{r} = 0 \tag{9}$$

where $W(\boldsymbol{r}) = \mathrm{diag}[w(r_1), w(r_2), \ldots, w(r_n)]$. Note that for ordinary least squares $\psi(\xi) = \xi$ and $w(\xi) = 1$.

For ordinary least squares, Eq. (9) is a linear equation for the coefficient vector $\hat{\boldsymbol{\beta}}$ through the scaled residual $\boldsymbol{r}$. However, in general, Eq. (9) is a system of nonlinear equations, and iterative methods are required to obtain the solution. The most popular approach is iteratively reweighted least squares (IRLS), which is attributed to Beaton and Tukey.[4] When the definition of residual is used, the necessary condition of Eq. (9) is written as

$$X^T W(\boldsymbol{r}) X\hat{\boldsymbol{\beta}} = X^T W(\boldsymbol{r})\boldsymbol{y} \tag{10}$$

which can be expressed as an iterative formula:

$$\hat{\boldsymbol{\beta}}^{(i+1)} = \hat{\boldsymbol{\beta}}^{(i)} + \{X^T W[(\boldsymbol{y} - X\hat{\boldsymbol{\beta}}^{(i)})/s]X\}^{-1} X^T W[(\boldsymbol{y} - X\hat{\boldsymbol{\beta}}^{(i)})/s] \times (\boldsymbol{y} - X\hat{\boldsymbol{\beta}}^{(i)}) \tag{11}$$

Table 1 Weighting functions for *M* estimator

$W(r)$	Range	Tuning constant
Huber's minimax[8]		
1	$\lvert r\rvert \le H$	$H = 1.0$
$H\lvert r\rvert^{-1}$	$\lvert r\rvert > H$	
Beaton and Tukey's biweight[4]		
$[1-(r/B)^2]^2$	$\lvert r\rvert \le B$	$B = 1.0$–3.0
0	$\lvert r\rvert > B$	
NIRLS		
$H\lvert r\rvert^{-1}$	$r \le -H$	$H = 1.0$
1	$-H < r \le 0$	$B = 1.0$–3.0
$[1-(r/B)^2]^2$	$0 < r \le B$	
0	$r > B$	

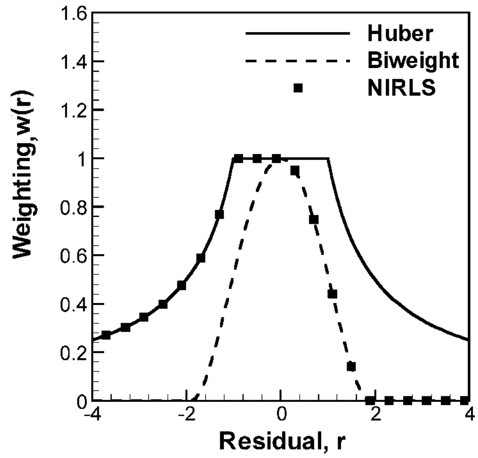

Fig. 1 Various weighting functions of *M* estimator.

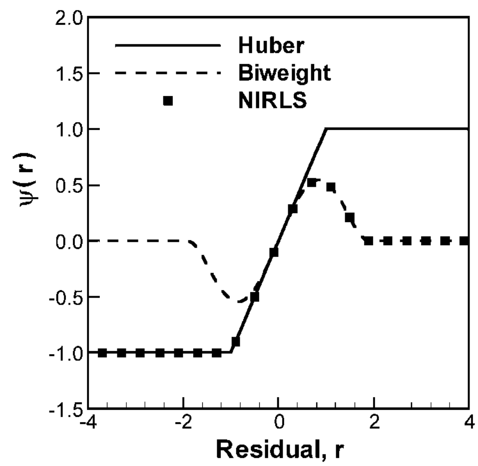

Fig. 2 Various ψ functions of *M* estimator.

Several possible weighting functions summarized in Table 1 and Fig. 1 were considered here. We preferred Beaton and Tukey's biweight function[4] (Fig. 1) to Huber's minimax (see Ref. 8) (Fig. 1) because it gives zero weighting to the outliers, and thus, the outliers are distinctly identified. The other function in Table 1, labeled NIRLS, will be discussed later in this section. An estimate of s was calculated as 1.5 $\mathrm{median}_i\, \lvert(\boldsymbol{y} - X\hat{\boldsymbol{\beta}})_i\rvert$ as recommended by Myers.[9] B is a tuning constant depending on the characteristics of the error distribution. Myers[9] suggested limiting the tuning constant to a range of 1–3. The shape of the weighting functions in Fig. 1 clearly shows that they penalize outliers with zero or low weighting while giving a weighting of one or near one to inliers (data points following the main trend). Equation (11) is not guaranteed to converge to the global minimum of $e(\hat{\boldsymbol{\beta}})$. Because the IRLS results depend on the initial guess for the regressor coefficients $\hat{\boldsymbol{\beta}}$, we need a good initial guess of $\hat{\boldsymbol{\beta}}$. With a nonredescending ψ function (Fig. 2) like Huber's

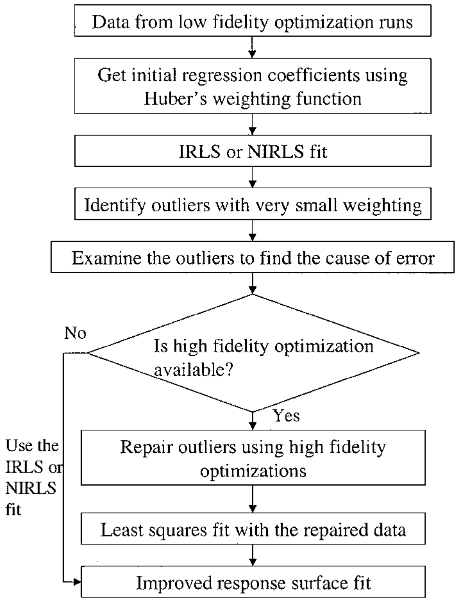

Fig. 3 Flowchart of IRLS/NIRLS procedure for low-fidelity optimization data.

minimax, Birch[10] proved that Eq. (11) is globally convergent to a unique solution, the global minimum of $e(\hat{\beta})$. Therefore, to get the initial estimates of the regressor coefficients $\hat{\beta}$, we adopted Huber's minimax function. Then the IRLS procedure using the biweight function was continued using these initial coefficients. The IRLS procedure with Huber's minimax function converged faster; it took about 20 iterations for convergence, whereas using the biweight required about 100 iterations for the data in this work.

Now we consider NIRLS in Table 1. The usual IRLS procedure assumes no bias of the residual sign. However, for our structural optimization data, the residual r_i is mostly positive for outliers because they are mostly caused by failure of optimization seeking a minimum. To account for the bias in the outliers, we devised a biased weighting function by combining the biweight and Huber's weighting function, and labeled it as the NIRLS weighting function. For the NIRLS weighting function shown in Fig. 1, data points with negative residuals are downweighted according to Huber's function, whereas points with positive residuals are downweighted by the biweight function. In this way, NIRLS downweights points with positive residual error more severely. NIRLS has been successfully applied to noisy data of structural optimization of HSCT[11] and to low-fidelity optimization data of cracked composite panels.[12]

IRLS/NIRLS techniques effectively remove outliers from the fit by downweighting them. However, when only scarce data points are used in response surface fit due to high cost of the data, outlier rejection may be undesirable because it may cause poor prediction where the outliers are located. Therefore, a better strategy will be to repair the detected outliers by performing higher fidelity simulation runs if they are available.[3] If the high-fidelity runs are much more expensive than the lower fidelity runs, repairing only the outliers detected by IRLS/NIRLS will have computational advantage over performing high-fidelity runs for all of the runs. The general IRLS/NIRLS procedure described in this section is summarized in Fig. 3.

III. Test Problem Study of Optimization Failure and IRLS/NIRLS Techniques: Rosenbrock Function in Five Dimensions

Before addressing the HSCT problem, it is worthwhile to illustrate the convergence difficulties experienced by numerical optimization for a simple problem. Here we demonstrate the failure of some optimization algorithms for a simple unconstrained minimization problem, the generalized Rosenbrock function (see Ref. 13) in five dimensions:

$$f(\boldsymbol{x}) = \sum_{k=1}^{n-1} 100\left(x_{k+1} - x_k^2\right)^2 + (1 - x_k)^2, \qquad n = 5 \qquad (12)$$

Table 2 Failure of various optimization programs for the five-dimensional Rosenbrock function

Software	Algorithm	Option	Number of failures out of 500 runs
MATLAB (*fminu*)	BFGS	——	7
DOT	BFGS	——	27
PORT (*DMNF*)	Trust region	With programming error	488
PORT (*DMNF*)	Trust region	With programming error corrected	0

The unconstrained minimization problem has a unique optimum $\boldsymbol{x}^* = (1, 1, 1, 1, 1)$ at which $f^* = f(x^*) = 0$. We performed 500 runs from different initial points, which were randomly generated in the range of [0, 2] for each of the five variables, to check for any optimization failures, defined as $f^* > 0.0001$. For the purpose of this demonstration we used DOT,[14] MATLAB®,[15] and trust region routines from the PORT[16] library, all with finite difference gradients. The results are summarized in Table 2. An unconstrained minimization routine, *fminu*, of MATLAB failed to find the true optimum in 7 out of 500 runs. DOT failed in 27 out of 500 runs. All failures occurred at essentially the same point (−0.962, 0.936, 0.881, 0.778, 0.605). The condition number of the Hessian matrix at the point was about 2400. The trust region algorithm[17] in the PORT library is known to have a robust convergence criterion, but the trust region routine *DMNF* with finite difference gradients failed in 488 out of the 500 cases, converging to different points for each failure case. This unexpected failure was traced to a programming error on our part: the function calculating $f(\boldsymbol{x})$ was not declared as double precision, while the double precision PORT library was used. When the double precision PORT library tries to use very small steps for finite difference gradient calculations, the objective function, mistakenly in single precision, cannot resolve the difference in $f(\boldsymbol{x})$. Consequently the optimization terminates prematurely. When the programming error was corrected, PORT had no failures.

The Rosenbrock function example shows that the optimization may produce poor optimum due to algorithmic difficulties (DOT and MATLAB) or user's programming error (PORT), which is a not uncommon source of optimization error. In both cases, incomplete convergence produced one-sided optimization error. The statistical approaches used in this paper are useful to identify the poor optimizations regardless of what caused the error.

We can generalize the five-dimensional Rosenbrock function by adding some artificial parameters to Eq. (12). For example, where $\boldsymbol{b} = (b_1, b_2, b_3, b_4)$,

$$f(\boldsymbol{x}; \boldsymbol{b}) = \sum_{k=1}^{4} 100\left(x_{k+1} - b_k x_k^2\right)^2 + (1 - x_k)^2 \qquad (13)$$

Denote $\min_x f(\boldsymbol{x}; \boldsymbol{b})$ by $f_o(\boldsymbol{b})$. Assume that we want to find $\boldsymbol{b}$ that minimizes $f_o(\boldsymbol{b})$. Thus there are two levels of optimization: $\boldsymbol{b}^*$ is sought to minimize $f_o(\boldsymbol{b})$ in the upper level, and $\boldsymbol{x}^*$ is sought to minimize f for a given $\boldsymbol{b}$ to find $f_o(\boldsymbol{b})$ in the lower level. Now the task here is to build a response surface approximation of $f_o(\boldsymbol{b})$ with respect to $\boldsymbol{b}$. This mimics the HSCT problem described in the next section.

We elected to change only b_1, b_2, and b_3, while keeping $b_4 = 1$, to make $f_o(\boldsymbol{b})$ have a unique global optimum of zero at $(b_1, b_2, b_3) = (1, 1, 1)$. The ranges of b_k are chosen to be between 0.9 and 1.1. For a given set of b_k, the parameterized Rosenbrock function is minimized from an initial design point $\boldsymbol{x} = (1.1, 0.9.1.1, 0.9.1.1)$. Figure 4 is a design line plot of f_o between the lowest $\boldsymbol{b}$ and the highest $\boldsymbol{b}$, showing the noisy response of $f_o(b_1, b_2, b_3)$. We can see that PORT with the programming error gave satisfactory results for only 2 out of the 11 runs. It is apparent that poor optimizations result in higher responses and the optimization error tends to be one-sided. The true response was obtained by using PORT without the programming error. The IRLS/NIRLS techniques are demonstrated using SAS[18] statistical software on this problem to identify erroneous optimizations of PORT with the programming error. To construct a

Table 3 Results of outlier repair for the parameterized five-dimensional Rosenbrock function

RS fit	B	Number of outliers a/b/c[a]	Mean of repair on outliers[b]	Mean of repair on inliers[c]	Ratio of mean repair on outliers to inliers	RMSE (% to the mean f_o)	Mean f_o	R^2
Before repair	NA	NA	NA	NA	NA	0.00748 (54.3%)	0.0138	0.6800
IRLS repair	1.0	9/7/20	0.0124	0.0065	1.91	0.00439 (45.5%)	0.0096	0.8342
NIRLS repair	1.0	8/8/20	0.0213	0.0030	7.02	0.00222 (29.7%)	0.0075	0.8658
Full repair[d]	NA	NA	NA	NA	NA	0.00023 (4.3%)	0.0053	0.9970

[a]Number of detected outliers a, number of big outliers detected (estimated error is greater than 10%) b, and total number of big outliers out of the 27 data points (estimated error is greater than 10%) c.
[b](Sum of f_o repair on outliers)/a.
[c](Sum of f_o repair on data points other than outliers)/(total number of points $-a$).
[d]All 27 points repaired.

Table 4 Results of optimization of $f_o(b_1, b_2, b_3)$ from the parameterized Rosenbrock function using RS approximations before and after outlier repair

Parameter	Without repair (optimum O)	IRLS repair (optimum I)	NIRLS repair (optimum N)	Full repair (optimum F)	Exact
b_1^*	0.9583	0.9922	1.0014	1.0057	1.0
b_2^*	1.1000	1.0591	0.9877	1.0072	1.0
b_3^*	0.9000	0.9965	1.0378	1.0060	1.0
$f_o(b_1^*, b_2^*, b_3^*)$ Predicted by RS	0.00066	−0.00175	−0.00018	−0.00007	——
$f_o(b_1^*, b_2^*, b_3^*)$ True	0.00303	0.00082	0.00028	0.00005	0.0

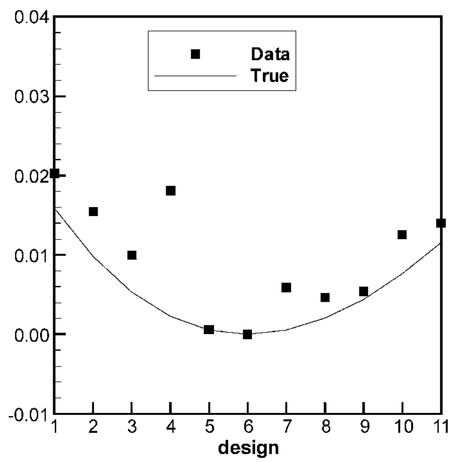

Fig. 4 Design line plot of the optimal values (f_o) for the parameterized Rosenbrock function; design 1 corresponds to b variables at their lower limits, and design 11 corresponds to b variables at their upper limits. Noisy data are the results of PORT with programming error, and the solid line is for the true optima.

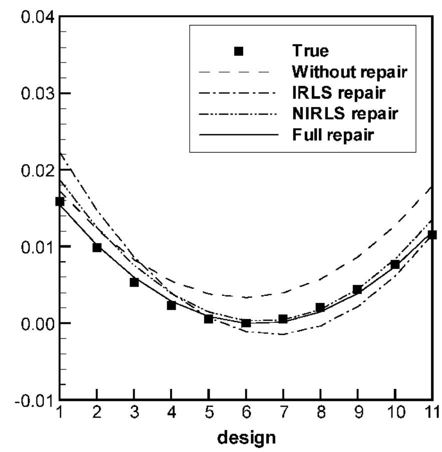

Fig. 5 Design line plot of optimal values (f_o) for the parameterized Rosenbrock function predicted by quadratic response surface approximations before and after outlier repair. The true minimum of f_o is located at design 6.

quadratic response surface approximation of $f_o(b_1, b_2, b_3)$, 27 variations of the Rosenbrock function are generated by using a three-level full factorial design in b_k, $k = 1, 2, 3$. The identified outliers having IRLS/NIRLS weighting less than 0.01 are repaired using the correct PORT optimization.

Table 3 summarizes the results of outlier repair. Before outlier repair, the RMSE of the quadratic response surface is 54.3% and the R^2 value is only 0.68 because many data points suffered from poor optimization. An aggressive outlier search was performed using a tuning constant $B = 1.0$. We defined big outliers as those showing 10% or greater error, and 20 out of 27 data points satisfied this criterion. IRLS declared 9 points as outliers, but only 7 of them were big outliers, whereas NIRLS detected 8 outliers and all of them were big outliers. Note that the mean repair on the outliers by NIRLS is 0.0213, whereas the mean repair on the outliers by IRLS is only 0.0124. The data points not declared as outliers are called inliers. The ratio of average repairs between outliers and inliers can be considered as a measure of success of outlier detection. For NIRLS the ratio was 7.02 compared to 1.91 of IRLS. The response surface fit is improved via IRLS repair, and there is still substantial improvement by NIRLS repair over IRLS repair; the RMSE is 45.5% for IRLS repair and 29.7% for NIRLS repair. When all 27 points were repaired, the RMSE was only 4.3% and the R^2 was 0.9970.

The response surface approximations are compared in Fig. 5 along the same design line used for Fig. 4. Before outlier repair, the response surface approximation overpredicts the true response because of the optimization error. With IRLS repair, the response surface prediction is improved, but the trend of the response is not accurate. The response surface fit with NIRLS repair follows the true response closely. The response surface models are to be used to find the minimum of $f_o(b_1, b_2, b_3)$. It is clear from Fig. 5 that the response surface fit with NIRLS repair will find a more accurate optimum f_o^* than the response surface fit with IRLS repair. Table 4 summarizes the results of minimization of f_o according to the response surface model used. When using the response surface fit without repair, the optimum of f_o is located at the boundary of the design space of $\boldsymbol{b}$, which have ranges between 0.9 and 1.1. When it is considered that the true optimum is located at the center of the design box, the original response surface approximation completely failed to capture the trend of the true response. It is seen that f_o^* gets closer to the true optimum as more outliers are repaired. When the accuracy is compared at optima f_o^*, the response surface model with IRLS repair underpredicts the response by $0.00257 = 0.00082 - (-0.00175)$, whereas the error of NIRLS is only $0.00046 = 0.00028 - (-0.00018)$.

IV. Description of the HSCT Design Problem

The second problem studied in this paper is a 250-passenger HSCT design with a 10,186-km (5500-nmile) range and cruise speed of Mach 2.4. A general HSCT model[1,19] developed by the Multidisciplinary Analysis and Design Center for Advanced Vehicles at Virginia Polytechnic Institute and State University includes 29 configuration design variables. The configuration optimization problem is to minimize the takeoff gross weight W_{TOGW} of the HSCT subject to 68 constraints. Following Balabanov et al.,[1] our task is to fit a response surface model to optimal wing bending material weight W_b from structural optimization as a function of configuration design variables. This response surface model for W_b is later used in the configuration optimization. There are four load cases for the HSCT design study that represent different points on the flight envelope.

Following Knill et al.,[20] we considered a simplified version of the HSCT design problem with only five configuration variables: the root chord c_{root}, tip chord c_{tip}, inboard leading-edge sweep angle Λ_{ILE}, thickness to chord ratio for the airfoil, t/c, and the fuel weight W_{fuel} (Table 5). Figure 6 illustrates the design variables. Other variables, such as fuselage, vertical tail, mission, and thrust-related parameters, are kept unchanged at the baseline values.

We applied a structural optimization procedure based on finite element (FE) analysis using GENESIS.[21] The finite element model, developed by Balabanov et al.,[1] uses 40 design variables, including 26 variables to control skin panel thickness, 12 variables to control spar cap areas, and 2 variables to control the rib cap areas. The HSCT codes[19,20] calculate aerodynamic loads for each of the four load cases, and a mesh generator by Balabanov et al.[1] calculates the FE mesh, converts the aerodynamic loads to applied loads at the structural nodes, and creates the input for GENESIS. Structural optimization is performed for each aircraft configuration using GENESIS, where the objective function is the total wing structural weight W_s and the W_b accounts for only the bending related elements. In previous papers,[2,11] we calculated W_b by considering the skin elements that are not at minimum gauge. However, this procedure added noise error besides the error due to incomplete optimization,[3] which is our main concern in this paper. Therefore in the IRLS procedure we used W_s to detect outliers of the structural optimization and W_b is assumed to be 70% of the W_s.

Table 5 Configuration design variables of HSCT with corresponding value ranges

Design variable	Range
Root chord $c_{\mathrm{root}}(v_1)$	45.720–57.912 m (150–190 ft)
Tip chord $c_{\mathrm{tip}}(v_2)$	2.134–3.962 m (7–13 ft)
Inboard leading-edge sweep $\Lambda_{\mathrm{ILE}}(v_3)$	67–76 deg
Thickness to chord ratio at root $(t/c)_{\mathrm{root}}(v_4)$	0.015–0.027
Fuel weight $W_{\mathrm{fuel}}(v_5)$	158,752–204,109 kg (350,000–450,000 lb)

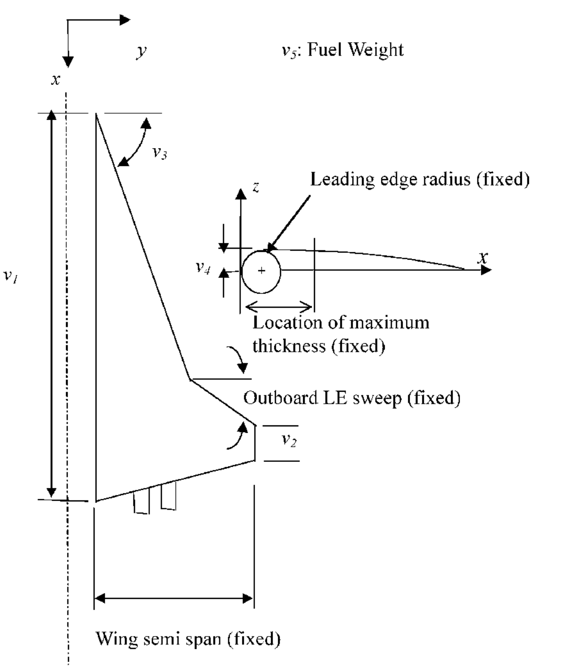

Fig. 6 Simplified HSCT design of five variables.

V. Effects of Convergence Parameters on Optimization Error

The optimization error depends on many convergence parameters, called control parameters in GENESIS. There are two loops in GENESIS.[22] In the outer loop, an approximation for the optimization problem is generated, and this approximate problem is passed to the inner loop of a gradient-based optimization, where we used the modified methods of feasible direction. After convergence of the approximate problem, a new approximation is constructed at the optimum of the approximate problem. The approximation and optimization are continued until no further change of design variables, called soft convergence, or no further change of the objective function, called hard convergence, occurs.

For the W_b response surface fit, we used a mixed experimental design[2,3] of 126 data points that included face centered central composite design and an orthogonal array, to which the full quadratic polynomial model is fit. GENESIS structural optimizations using four different sets of parameters (Table 6) were performed at each of the 126 points. Case A2 employs the default parameters provided by GENESIS. Case A5 is the same as case A2 except that the parameter ITRMOP was increased to 5. ITRMOP controls the convergence of the inner optimization. For the approximate optimization to converge, the inner-loop convergence criterion for the objective function change must be satisfied ITRMOP consecutive times. The default value of ITRMOP is 2, and by increasing it, the inner loop is forced to iterate further. In turn, this may force the outer loop to continue, because the soft convergence criteria, changes in design variables, are not met, which may have been satisfied if ITRMOP had been 2. This may result in significant improvement in the final optimum. After extensive experimentation with the optimization parameters and help from the developers of GENESIS, we found that ITRMOP was the most important parameter for improving the accuracy of the optimization for our problem. Case B2 employs tighter move limits and convergence criteria than case A2. It reflects our first attempt to improve the optimization results. Case B5 is the same as case B2 except that ITRMOP is 5.

To visualize the behavior of the errors, Fig. 7 shows W_s for 21 HSCT designs along a line connecting two diagonal vertices,

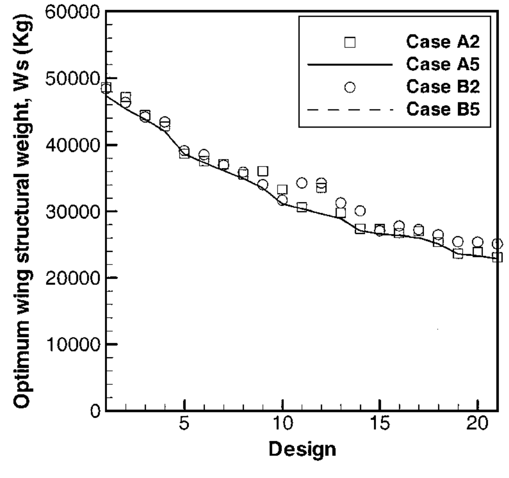

Fig. 7 Design line plots of W_s for different convergence criteria settings.

Table 6 Performance of structural optimization of HSCT for various GENESIS convergence parameter settings

Case	Move limit parameter	Outer-loop convergence parameter	Inner-loop convergence parameter (ITRMOP)	Number of points for which the best W_s was achieved out of 126 points	Mean of estimated error, kg (percent of mean W_s)	CPU time per GENESIS run, s (SGI Origin 2000)
A2	Default	Default	Default(2)	0	1780 (4.85)	78.1
A5	Default	Default	Tight(5)	65	70.67 (0.193)	156.7
B2	Tight	Tight	Default(2)	0	2062 (5.62)	61.4
B5	Tight	Tight	Tight(5)	61	12.15 (0.033)	143.3

Table 7 Results of outlier repair for case B2 of structural optimization of HSCT

RS fit	B	Number of outliers a/b/c[a]	Mean of repair on outliers, kg[b]	Mean of repair on inliers, kg[c]	Ratio of mean repair on outliers to inliers	RMSE, kg (% to the mean W_s)	Mean W_s	R^2
Before repair	NA	NA	NA	NA	NA	3,064 (8.7)	35,243	0.9297
IRLS repair	1.9	27/11/22	3,943	1,503	2.62	1,600 (4.7)	34,333	0.9769
NIRLS repair	1.9	29/19/22	5,161	1,046	4.93	1,380 (4.1)	33,964	0.9824
IRLS repair	1.0	49/16/22	3,247	1,216	2.67	1,384 (4.1)	33,884	0.9828
NIRLS repair	1.0	52/22/22	3,845	643.6	5.97	1,059 (3.2)	33,535	0.9902
Full repair[d]	NA	NA	NA	NA	NA	1,169 (3.5)	33,177	0.9879

[a]Number of detected outliers a, number of big outliers detected (estimated error is greater than 10%) b, and total number of big outliers out of the 117 data points (estimated error is greater than 10%) c.
[b](Sum of W_s repair on outliers)/a.
[c](Sum of W_s repair on data points other than outliers)/(total number of points $-a$).
[d]All 117 points corrected.

$(1, -1, 1, -1, 1)$ and $(-1, 1, -1, 1, -1)$ in a coded form, of the HSCT configuration design space. It is clear from Fig. 7 that cases A2 and B2 have substantial errors. When ITRMOP is increased to 5, the noise in W_s was greatly reduced. Poor optimizations apparently produce heavier results.

To calculate the error, we need to know the true W_s, which, strictly speaking, cannot be known due to the iterative procedure inherent in the optimization. Here we estimate the true W_s by taking the best of the four GENESIS runs we already did using different parameters. Table 6 shows the performance of each set of GENESIS parameters. The optimization error was calculated by comparing W_s to the best of the four W_s available.

Table 6 shows how many times each case produced the lowest, the best among the four W_s for 126 points. Cases A2 and B2 never found the best results. However, with ITRMOP = 5, cases A5 and B5 achieved the best W_s 65 and 61 times, respectively, about half of the data each. For the default GENESIS parameter (case A2), the mean error was 4.85% of the mean W_s. It is seen that tightening the move limits and the outer-loop convergence criteria of case A2 keeping ITRMOP = 2 actually had a detrimental effect because the mean error increased to 5.62% for case B2. With cases A5 and B5, the mean error was very small, 0.193% and 0.033%, respectively.

The estimate of the true optimum by taking the best of four GENESIS runs of difference parameters (cases A2, A5, B2, and B5) turned out to be very accurate because additional runs using case B5 with different initial designs achieved very little improvement of W_s. Therefore, we will call the best of four W_s as estimated true W_s. Our concern about the optimization error will be mainly for the low-fidelity (ITRMOP = 2) cases. For the high-fidelity cases (ITRMOP = 5), the error appears to be negligible. In terms of computational cost, the high-fidelity optimization using ITRMOP = 5 required more than twice the CPU time of the low-fidelity optimization, ITRMOP = 2.

VI. Detection and Repair of Poor Optimization Runs by IRLS and NIRLS

The IRLS procedure to identify poor optimizations was applied to case B2, the most noisy optimization data. A full quadratic response surface model was used, and the IRLS/NIRLS outlier detection was applied. Two different values of B, 1.9 and 1.0, were used for IRLS or NIRLS to check the effects of the tuning constant, which acts like a threshold for outlier detection (Table 1). A smaller B leads to a more aggressive outlier search; the allowable region around the response surface fit for good points becomes narrower, and more points will be declared outliers. For NIRLS, the tuning constant H controlling the weighting function for negative residuals was kept at 1.0. Detected outlier points that had a weighting of less than 0.01 were corrected by the estimated true W_s. A full quadratic response surface model was fit to the data before and after repair, using least squares, to measure the noise level remaining in the data.

Table 7 presents a summary of the outlier detection and repair with IRLS and NIRLS. In the response surface fit, we removed 9 points of extraordinarily heavy designs [W_s greater than 68,036 kg (150,000 lb), which is about triple the usual optimum designs] from the 126 data points and used only 117 points, because we did not want to consider unreasonable designs. This approach of excluding design points with unreasonable results is known as the reasonable design space approach.[1] To assess the success in outlier detection, all 117 points were also repaired, and the results are shown in the last row in Table 7.

With a tuning constant $B = 1.9$, IRLS and NIRLS identified 27 and 29 candidate outliers, respectively. The error in the outliers is calculated by comparing W_s (case B2) to the repaired value. For comparison we also repaired points not flagged as outliers. For IRLS, the average repair was 3943 kg (8694 lb) for outliers compared to 1503 kg (3314 lb) for inliers. NIRLS did a better job in finding outliers; the average repair was 5161 kg (11,379 lb) for outliers and 1046 kg (2307 lb) for inliers. The ratio between the average repairs of W_s for outliers and inliers was 4.93 for NIRLS compared to 2.62 of IRLS. Out of 117 points, 22 were big outliers showing error greater than 10%, and IRLS and NIRLS detected 11 and 19 of them, respectively (Table 7).

When $B = 1.0$, NIRLS successfully detected all of the 22 big outliers, whereas IRLS missed 6 of the big outliers. The ratio of repair of W_s for outliers and inliers is 5.97 for NIRLS compared to 2.67 for IRLS, indicating that NIRLS is more successful in homing

Table 8 Results of five-variable HSCT configuration optimizations using W_b response surface approximations before and after outlier repair (based on data of case B2)

		Optima using W_b RS			
Parameter	Initial design	Without repair (optimum O)	IRLS repair (optimum I)	NIRLS repair (optimum N)	Full repair (optimum F)
W_{TOGW}, kg	356,023	357, 489	353,568	351,944	350,022
W_b by RS_O, kg	24,716	18,103	18,116	18,085	18,098
	(12.0)[a]	(9.1)	(9.5)	(9.6)	(9.6)
W_b by RS_I, kg	23,687	15,781	16,601	16,529	15,737
	(7.4)	(−4.9)	(0.4)	(0.1)	(−4.7)
W_b by RS_N, kg	23,188	15,462	15,454	15,982	15,373
	(5.1)	(−6.8)	(−6.6)	(−3.2)	(−7.0)
W_b by RS_F, kg	22,452	15,374	15,358	15,276	15,279
	(1.8)	(−7.3)	(−7.2)	(−7.5)	(−7.5)
W_b by best of four GENESIS runs, kg	22,059	16,589	16,543	16,509	16,517
v_1, m	51.82	55.47	55.36	55.29	55.19
v_2, m	3.048	3.107	3.107	3.106	3.106
v_3, deg	71.50	67.49	67.56	67.48	67.52
v_4	0.02100	0.02123	0.02125	0.02126	0.02127
v_5, kg	195,038	187,789	185,777	184,873	183,889

[a]Error, %.

in on points with large errors. The aggressive outlier search with $B = 1.0$ does find more outliers. However, it declares more points as outliers, and the computational advantage of repairing only outliers may be reduced. For example, the number of outliers declared by NIRLS increased from 29 with $B = 1.9$ to 52 with $B = 1.0$. Taking the mean CPU hour of case B2 as a single unit, our efforts to repair case B2, requiring GENESIS runs of cases A2, A5, and B5, costs about extra 6 units of CPU hour. Compared to obtaining accurate W_s for all data points using cases A2, A5, B2, and B5, NIRLS repair costs 36 and 52% CPU hours for $B = 1.9$ and 1.0, respectively.

Quadratic response surface approximations using least squares were fit to the original and repaired data of case B2. Before repair, the RMSE was 8.7% of the mean W_s and was reduced to 4.7% by IRLS repair with $B = 1.9$ and reduced further to 4.1% by NIRLS repair. When more outliers were repaired with $B = 1.0$, the RMSE was 4.1% with IRLS repair and 3.2% with NIRLS repair. Taken together, the results indicate that IRLS procedures are useful for detecting points with large optimization errors and that the NIRLS procedures are more reliable.

VII. HSCT Configuration Optimization Using W_b Response Surface Approximations

We performed HSCT configuration optimizations using the quadratic W_b response surface fits obtained in the preceding section. The effects of improvements of response surface approximations by outlier repair on optimum HSCT designs were investigated. Note that W_b response surfaces were obtained by multiplying the W_s response surfaces by a factor of 0.7. Table 8 contains the initial and optimal designs according to the response surface fits used. The initial design point was the center of the design box except that the fuel weight was increased to 195,038 kg (430,000 lb) to satisfy the range constraint at the beginning of the configuration optimization. The response surface (RS) fit based on case B2 data without outlier repair (RS_O) was first used to find the optimal HSCT configuration, called optimum O. Then, configuration optimizations starting from optimum O are performed using RS fits based on repaired data: IRLS repair with $B = 1.9$ (RS_I), NIRLS repair with $B = 1.9$ (RS_N), and full repair (RS_F). The corresponding optimal HSCT designs were labeled as optimum I, N, and F, respectively. At the initial design, the estimated true W_b was 22,059 kg (48,633 lb) but the RS fit without outlier repair, RS_O, overpredicted W_b with 12.0% error. By repairing outliers, the error was reduced to 7.4, 5.1, and 1.8% for RS_I, RS_N, and RS_F, respectively.

Comparing the RS prediction for a given optimum design (column), we observe that the predicted W_b decreases along RS_O, RS_I, RS_N, and RS_F. It is seen that the original RS approximation, RS_O, has positive errors whereas RS fits with repair tend to have negative errors, which increase in the absolute sense along RS_I, RS_N, and RS_F. For example, at optimum O, RS_O has a positive error of 9.1%, but the errors became more negative along RS_I, RS_N, and RS_F with −4.9, −6.8, and −7.3% errors, respectively. The same trend is observed for optima I, N, and F. This may be attributed to overoptimism of the optimizer; optimizers tend to exploit regions where the RS fits are erroneously optimistic and converge to designs for which the RS approximations underpredict the response. By intensive outlier repair such as NIRLS or full repair, W_b estimates by the RS fits are lowered, and then any exploitation of weakness of the RS approximations by the optimizer is fully experienced at the optimum.

Therefore, a moderate outlier search such as IRLS may be conservative in the framework of the configuration optimization. It has a protective margin against overoptimism of the optimizer because the moderately corrected RS fit would slightly overpredict the structural weight at much of the design space. The errors of the RS approximations at optima support this conjecture. For optimum O, RS_O overpredicted W_b with 9.1% error. For optimum I of IRLS repair, the error of RS_I was only 0.4%, although RS_I cannot be expected to show such accuracy in general because, for other optima, RS_I may have error around −5%. For optimum N of NIRLS repair, RS_N underpredicted the W_b with −3.2% error. Furthermore, for optimum F of full repair, RS_F has −7.5% error. The takeoff gross weight W_{TOGW} of the optima decreases as W_b predicted by RS approximations decreases with more outliers repaired. Table 8 contains also the configuration variables at the optima. Compared to optimum O, optimum I is seen to have a slightly reduced wing root chord v_1 and fuel weight v_5. For all of the optima, the range constraint was active and the fuel weight decreases further for optima N and F, as W_b prediction decreases.

VIII. Conclusions

RS techniques provide statistical tools to identify outliers in the data that do not fit the underlying model. We have demonstrated the use of one of these tools, IRLS, for detecting design points where optimization gave poor designs. Because optimization error is one-sided, we have also proposed a nonsymmetrical version of IRLS (NIRLS) that takes into account the asymmetry in optimization errors.

Optimization of the five-dimensional Rosenbrock test function experienced convergence difficulties due to a programming error, which became the basis for applying IRLS/NIRLS techniques to minimization of a parameterized Rosenbrock function. NIRLS was more effective than IRLS in identifying outliers, which were repaired by accurate optimizations. The outlier repair improved the accuracy of the RS approximations, and the RS fit based on NIRLS repair was more successful than IRLS in locating the optimal solution.

Structural optimization of an HSCT produced poor optimizations due to incomplete convergence. We focused mostly on big outliers, where the error in optimal weight was more than 10%. NIRLS procedures were more reliable in detecting big outliers than IRLS. The accuracy of the RS fit was substantially improved by IRLS repair, and relatively small further improvements were obtained by NIRLS or full repair.

HSCT configuration optimizations were performed to check the effects of the improved RS approximation of optimal wing bending material weight. As more outliers were repaired, the W_{TOGW} of the optimal designs was lowered. Also, it was noted that moderate outlier repair was a conservative choice for configuration optimization than aggressive repair because the optimizer underestimated the structural weight by exploiting weaknesses of the response surface approximations.

Acknowledgments

The work of M. Papila at the University of Florida was supported by NASA Grant NAG1-2177, and the work of R. T. Haftka and authors at Virginia Polytechnic Institute and State University was partially supported by National Science Foundation Grant DMI-9979711.

References

[1]Balabanov, V., Giunta, A. A., Golovidov, O., Grossman, B., Mason, W. H., Watson, L. T., and Haftka, R. T., "Reasonable Design Space Approach to Response Surface Approximation," *Journal of Aircraft*, Vol. 36, No. 1, 1999, pp. 308–315.

[2]Papila, M., and Haftka, R. T., "Uncertainty and Wing Structural Weight Approximations," AIAA Paper 99-1312, April 1999.

[3]Papila, M., and Haftka, R. T., "Response Surface Approximations: Noise, Error Repair, and Modeling Errors," *AIAA Journal*, Vol. 38, No. 12, 2000, pp. 2336–2343.

[4]Beaton, A. E., and Tukey, J. W., "The Fitting of Power Series, Meaning Polynomials, Illustrated on Band-Spectroscopic Data," *Technometrics*, Vol. 16, No. 2, 1974, pp. 147–185.

[5]Holland, P. H., and Welsch, R. E., "Robust Regression Using Iteratively Reweighted Least Squares," *Communications in Statistics: Theory and Methods*, Vol. 6, No. 9, 1977, pp. 813–827.

[6]Myers, R. H., and Montgomery, D. C., *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*, Wiley, New York, 1995, pp. 16–78.

[7]Rousseuw, P. J., and Leroy, A. M., *Robust Regression and Outlier Detection*, Wiley, New York, 1987, pp. 1–9, 216–247.

[8]Huber, P. J., "Robust Estimation of a Location Parameter," *Annals of Mathematical Statistics*, Vol. 35, No. 1, 1964, pp. 73–101.

[9]Myers, R. H., *Classical and Modern Regression with Applications*, PWS-Kent, Boston, 1990, pp. 351–356.

[10]Birch, J. B., "Some Convergence Properties of Iteratively Reweighted Least Squares in the Location Model," *Communications in Statistics: Simulation and Computation*, Vol. 9, No. 4, 1980, pp. 359–369.

[11]Kim, H., Papila, M., Mason, W. H., Haftka, R. T., Watson, L. T., and Grossman, B., "Detection and Correction of Poorly Converged Optimizations by Iteratively Reweighted Least Squares," AIAA Paper 2000-1525, April 2000.

[12]Papila, M., and Haftka, R. T., "Response Surfaces for Optimal Weight of Cracked Composite Panels: Noise and Accuracy," AIAA Paper 2000-4755, Sept. 2000.

[13]Schittkowski, K., *More Test Problems for Nonlinear Programming Codes, Lecture Notes in Economics and Mathematical Systems*, Vol. 282, Springer-Verlag, Berlin, 1987, pp. 118–123.

[14]*DOT Design Optimization Tools Users Manual*, Vanderplaats Research and Development, Colorado Springs, CO, 1995.

[15]*MATLAB Optimization Toolbox Users Guide*, Ver. 5, MathWorks, Natick, MA, 1997.

[16]Gay, D. M., "Subroutines for Unconstrained Minimization Using a Model/Thrust-Region Approach," *ACM Transactions on Mathematical Software*, Vol. 9, No. 4, 1983, pp. 503–512; see also URL //http://www.bell-labs.com/projects /PORT.

[17]Dennis, J. E., Jr., and Schnabel, R. B., *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice–Hall, Englewood Cliffs, NJ, 1983, pp. 129–147.

[18]*SAS Users Guide*, Ver. 6, 4th ed., SAS Inst., Cary, NC, 1990.

[19]Giunta, A. A., Balabanov, V., Burgee, S., Grossman, B., Haftka, R. T., Mason, W. H., and Watson, L. T., "Multidisciplinary Optimisation of a Supersonic Transport Using Design of Experiments Theory and Response Surface Modelling," *Aeronautical Journal*, Vol. 101, No. 1008, 1997, pp. 347–356.

[20]Knill, D. L., Giunta, A. A., Baker, C. A., Grossman, B., Mason, W. H., Haftka, R. T., and Watson, L. T., "Response Surface Models Combining Linear and Euler Aerodynamics for Supersonic Transport Design," *Journal of Aircraft*, Vol. 36, No. 1, 1999, pp. 75–86.

[21]*GENESIS User Manual*, Ver. 5.0, VMA Engineering, Colorado Springs, CO, 1998.

[22]Vanderplaats, G. N., and Miura, H., "GENESIS-Structural Synthesis Software Using Advanced Approximation Techniques," AIAA Paper 92-4839, Sept. 1992.

E. Livne
Associate Editor