Probabilistic Modeling of Errors from Structural Optimization Based on Multiple Starting Points

Hongman Kim*, Raphael T. Haftka†, William H. Mason‡, Layne T. Watson§, and Bernard Grossman¶

Multidisciplinary Analysis and Design (MAD) Center for Advanced Vehicles
Virginia Polytechnic Institute and State University
Blacksburg, VA 24061-0203

ABSTRACT

With optimization increasingly used in engineering applications, a series of optimization runs may be required, and it may be too expensive to converge them to very high accuracy. A procedure for estimating average optimization convergence errors from a set of poorly converged optimization runs is developed. A probabilistic model is fitted to the errors in optimal objective function values of poorly converged runs. The Weibull distribution was identified as a reasonable error model both for the Rosenbrock function problem and the structural optimization of a high speed civil transport. Once a statistical model for the error is identified, it can be used to estimate average errors from a set of pairs of runs. In particular, by performing pairs of optimization runs from two starting points, accurate estimates of the mean and standard deviation of the convergence errors can be obtained.

1. Introduction

Optimization is an iterative procedure, which is rarely allowed to converge to high precision due to computational cost considerations. In design optimization of a complex system, sub-optimization problems are often solved within the system level optimization. Consequently, the optimization results are usually a noisy function of the parameters of the design problem [1]. When a single optimization is flawed, it may be difficult to tell. However, when many optimization results are available, such as when building a response surface model based on the sub-optimization results, statistical methods can be used to identify runs with very large errors [1] and estimate the average error of the multiple optimization runs [2]. It is certainly possible to estimate the convergence error by performing accurate optimization with tightened convergence criteria, but this can be very expensive. This paper shows that a probabilistic model can estimate average optimization error without performing very accurate optimization.

Numerical optimization errors are deterministic in that computer simulation gives the same output for the same input for repeated runs. However, when an optimization procedure experiencing convergence difficulties is very sensitive to small changes of input parameters, it is difficult to predict the magnitude of the error. Therefore, a probabilistic model can be useful to characterize the optimization error. Errors in computational simulations are an important source of design uncertainties and it is important to estimate the magnitude of the errors. Another advantage of probabilistic modeling is that the error models can be readily used in the framework of design under uncertainty.

In this paper the utility of a probabilistic model of optimization convergence errors is demonstrated for optimization used to obtain wing structural weight (W_s) for various configurations of a high-speed civil transport (HSCT) [3]. The structural optimization was a sub-optimization within a configuration design optimization of the HSCT, and was performed a priori on a carefully selected set of HSCT configurations to build a response surface model (*cf.* Ref. [4]) of W_s. The structural optimization had substantial errors because it was difficult to find good convergence criteria. The objective of the present work is to demonstrate that such convergence errors can be estimated by performing the optimization runs in pairs. In previous work [2], the authors successfully estimated average errors by using two sets of optimization runs with different convergence criteria. Change of convergence criteria may require expert knowledge of the optimization algorithm and depend on specifics of the optimization program. In contrast, changing the initial design point is straightforward and can be used for most optimization programs. Here it is shown that optimization results from two different sets of initial design points can serve to estimate the average optimization error.

2. Error from Structural Optimization

The application problem in this paper is a HSCT design model developed by the Multidisciplinary Analysis and Design (MAD) Center for Advanced Vehicles at Virginia Tech. A simplified version of the problem is used following Knill et al. [5] with five configuration design variables including wing root chord, wing tip chord, inboard leading edge sweep angle, airfoil thickness ratio, and fuel weight. Takeoff gross weight is minimized at the system level as a function of the five configuration variables. To improve wing weight equations based on historical data, GENESIS [6] structural optimization software based on finite element models is used. The finite element model has 1127 elements at 226 nodes with a total number of 1242 degrees of freedom. The structural optimization is a sub-optimization within a system level configuration optimization, and wing structural weight (W_s) is minimized in terms of 40 structural design variables, including 26 to control skin panel thickness, 12 to control spar cap areas, and two for the rib cap areas [3]. The structural optimization is performed a priori for many aircraft configurations and a response surface model of the optimum structural wing weight is constructed for use in the configuration optimization. For the response surface construction, the five design variables are coded so that each ranges between –1 and +1.

The structural optimization resulted in a noisy W_s in terms of the HSCT configuration variables [3], [1]. Figure 1 shows the W_s response for 21 HSCT configurations generated by a linear interpolation between two extreme designs. Design 1 corresponds to (-1, -1, -1, -1, -1) (all configuration variables at their lower bounds) and design 21 corresponds to (1, 1, 1, 1, 1) (all configuration variables at their upper bound) in a coded form of the HSCT configuration variables. Case 1 corresponds to the original results obtained by using the default convergence criteria of GENESIS. A conservative structural design from a previous study is used as an initial design point for all of the structural optimization runs of Case 1. Designs 13, 16, and 19 of Case 1 seem to have relatively large errors.

For Case 2, an initial design point perturbed from that of Case 1 by multiplying each of the 40 structural design variables by random factors between 0.1 – 1.9, was used. It is seen that the results are still noisy, and that the noise error tends to be one-sided (W_s greater than the true minimum). That is because the noise error comes from incomplete convergence of minimization.

Efforts have been made to reduce the convergence error of the HSCT structural optimization. Papila and Haftka [7] repaired erroneous optimizations by changing optimization algorithms or trying different initial designs. After extensive experiments with convergence criteria, it was found that the most effective way to improve the optimization was to tighten one of the convergence criteria [2], [8]. However, it was not trivial to choose the right convergence tolerances, and the tightened convergence tolerances more than doubled the cost of the optimization.

Define optimization error

$$e = W_s - W_s^t, \tag{1}$$

where W_s is the calculated optimum and W_s^t is the true optimum, which is unknown for many practical engineering optimization problems. Note that we are mainly interested in the convergence error and W_s^t represents a true physical optimum of the computer model of the optimization problem. Another source of error can be inaccurate computational simulation models, which we do not address here. To estimate W_s^t, we need to perform fully converged optimization runs with properly tightened convergence criteria, which can be expensive. For the HSCT problem, computational cost was seen to be more than double for properly tightened convergence criteria compared to the default setting [2]. In practice, we estimated W_s^t by taking the best of repeated GENESIS runs: Case 1 and Case 2 with different initial designs as described above and additional six cases with different sets of convergence criteria [2]. To study the error in W_s from the structural optimization, we used a mixed experimental design of 126 HSCT configurations [8], intended to permit fitting a quadratic or cubic polynomial to create a W_s response surface approximation for the five-variable HSCT design problem. The optimization error, e, was calculated for each of the 126 HSCT configurations. Then, the mean and standard deviation of e were estimated for each case of different GENESIS parameters,

$$\hat{m}_{data} = \frac{\sum_{i=1}^{n} e_i}{n} = \bar{e}\,, \quad \hat{s}_{data} = \sqrt{\frac{\sum_{i=1}^{n} (e_i - \bar{e})^2}{n-1}}\,, \tag{2}$$

where n is the sample size. Table 1 shows that the average errors were not very different between Case 1 and Case 2, 5.51% and 5.34%, respectively. In terms of computational cost, Cases 1 and 2 took almost the same CPU time per GENESIS run since the only difference was the initial design point.

As seen in Figure 1, the results of Cases 1 and 2 did not match, because both of them are poorly converged optimizations. The magnitude of the difference in W_s between Cases 1 and 2 is an indication of the level of the convergence error; we expect the difference to be large when optimization runs have large errors. The average difference of W_s between Cases 1 and 2 was 5942 *lb*., which was 7.3% of the average of estimated true W_s and the largest magnitude of the difference was 58715 *lb*. (72.1%). In the next section, we will introduce a probabilistic modeling technique to use the differences to estimate the average optimization errors.

3. Probabilistic Modeling of Optimization Error

<u>Applying MLE to Establish a Probabilistic Model of Error</u>

With multiple optimization runs available, we can obtain a data driven model of the optimization error by fitting a probability distribution to the actual error obtained from Eq. 1. This approach is denoted an *error fit*. This approach requires fully converged high fidelity optimization runs to calculate error data, which are not always available. However, once the probabilistic distribution of the error is known, as we discuss later in this section, the model can be used to estimate the average errors even when fully converged results are not available. We use the maximum likelihood estimation [9] (MLE) method for the distribution fit. In MLE, we find a vector of distribution parameters b to maximize the likelihood function $l(\mathrm{b})$, which is a product of the probability density function *f* over the sample data x_i (i = 1, …, n),

$$l(\mathrm{b}) = \prod_{i=1}^{n} f(x_i;\mathrm{b}). \tag{3}$$

The quality of fit is checked via the c^2 goodness-of-fit test [9], which is essentially a comparison of histograms between the data and the fit. The test results will be given in terms of the *p*-value. A *p*-value near one implies a good fit and a small chance that the data is inconsistent with the distribution. Conversely, a small *p*-value implies a poor fit and a high chance that the data is inconsistent with the distribution.

Considering the one-sidedness of the optimization error, we selected the Weibull distribution [9], [10], which is defined by a shape parameter a and a scale parameter b. The probability density function (PDF) of the Weibull distribution is

$$f(x) = \begin{cases} a b^{-a} x^{a-1} \exp\left(-\left(\dfrac{x}{b} \right)^{a} \right) & if\ x \geq 0 \\ 0 & if\ \ x < 0 \end{cases}. \tag{4}$$

Once we obtain the distribution parameters a and b via MLE, estimates of the mean and standard deviation of e can be calculated from

$$\hat{m}_{fit} = \frac{b}{a}\Gamma\left(\frac{1}{a}\right), \quad \hat{s}_{fit} = \sqrt{\frac{b^2}{a}\left\{2\Gamma\left(\frac{2}{a}\right) - \frac{1}{a}\left[\Gamma\left(\frac{1}{a}\right)\right]^2\right\}}, \tag{5}$$

where Γ is the gamma function.

Error Statistics from Differences of Poorly Converged Optimization Runs

When fully converged results are available, estimating errors in the poorly converged results is of use in that it can provide information for the more common case where converged results are not available. In particular, the statistical model identified from the error fit using fully converged data can be used to estimate error statistics for poorly converged optimization runs. Indeed, using the knowledge that the errors can be fit well by a Weibull distribution, we estimated the distribution parameters from the differences of optimal values from two different convergence settings [4]. Here, we propose to use different initial points (*e.g.*, Cases 1 and 2 of the HSCT structural optimization) instead of convergence criteria. Changing convergence settings may require expert level knowledge depending on the optimization software, whereas it is much simpler to change initial designs to generate other sets of optimization results.

For the pair of optimization results, W_s^1 with optimization parameter setting #1 and W_s^2 with optimization parameters setting #2, model the optimization errors as random variables s and t,

$$s = W_s^1 - W_s^t, \qquad t = W_s^2 - W_s^t. \tag{6}$$

Note that the true optimum W_s^t is not random, although it may be unknown for many practical engineering optimization problems. Random properties of the errors are due to noisy W_s^1 and W_s^2. Since we want to avoid expensive calculation of W_s^t, the difference of s and t is defined as the *optimization difference*,

$$x = s - t = (W_s^1 - W_s^t) - (W_s^2 - W_s^t) = W_s^1 - W_s^2. \tag{7}$$

If s and t are independent, the probability density function (PDF) of x can be obtained from the following integration of the PDF functions $g(s; \mathrm{b}_1)$ and $h(t; \mathrm{b}_2)$,

$$f(x; \mathrm{b}_1, \mathrm{b}_2) = \int_{-\infty}^{\infty} g(s; \mathrm{b}_1) h(s - x; \mathrm{b}_2) ds. \tag{8}$$

Note that the optimization difference x is easily calculated from W_s^1 and W_s^2 that are readily available. Then, we can fit Eq. 8 to the optimization differences via MLE. This *difference fit* does not require estimation of true optima, and the error distributions of the two cases involved are obtained simultaneously.

4. Test Problem Study of Probabilistic Modeling of Optimization Error

Before addressing the HSCT problem, we demonstrate the estimation of the optimization error statistics for a simple optimization problem, the generalized Rosenbrock function [11] in five dimensions:

$$f(\boldsymbol{x}) = \sum_{k=1}^{n-1}\left[100(x_{k+1} - x_k^{\ 2})^2 + (1 - x_k)^2\right], \quad (n = 5). \tag{9}$$

The unconstrained minimization problem has a unique optimum $\boldsymbol{x}^*$ = (1, 1, 1, 1, 1) at which $f(\boldsymbol{x}^*) = 0$. Optimization was performed from randomly selected initial design points using DOT [12], MATLAB [13], and PORT [14], all with finite difference gradients. DOT and MATLAB produced incorrect results in 7 and 27 runs, respectively, out of 500 runs. All failures occurred at essentially the same point $\boldsymbol{x}$ = (-0.962, 0.936, 0.881, 0.778, 0.605). The condition number of the Hessian matrix at the point was about 2400, which is an indication of ill-conditioning of the problem. The routine DMNF of the PORT mathematical library produced incorrect results for most of the 500 runs, converging to a different design point for each run. This unexpected failure was traced to a programming error on our part: the name of the Fortran function calculating $f(\boldsymbol{x})$ was not declared as double precision, while the double precision PORT optimization routine was used. When the programming error was corrected, PORT found the true optimum for all the 500 runs.

The programming error caused loss of significant figures in the objective function values passed to DMNF and the optimization terminated prematurely for many runs. In this particular problem, the programming error caused convergence errors. Since user's programming errors are not an uncommon source of optimization error, we selected this problem for initial demonstration of the low-fidelity fit.

In order to generalize the problem to produce a set of runs, we added parameters $\boldsymbol{b}$ to Eq. 9 [1] so that the Rosensbrock function becomes

$$f(\mathbf{x};\mathbf{b}) = \sum_{k=1}^{4}\left[100(x_{k+1} - b_k x_k^{\ 2})^2 + (1 - x_k)^2\right], \quad \text{where } \mathbf{b} = (b_1, b_2, b_3, b_4). \tag{10}$$

We define

$$f_o(\boldsymbol{b}) \equiv \min_{\boldsymbol{x}} f(\boldsymbol{x};\boldsymbol{b}), \tag{11}$$

and 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. $\boldsymbol{b}$ corresponds to the configuration design variables of the HSCT in the system level and $\boldsymbol{x}$ corresponds to the design variables of the structural optimization, the sub-optimization. In this section, we will estimate optimization errors when the erroneous PORT was applied.

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 minimum of zero at (b_1, b_2, b_3) = (1, 1, 1). The ranges of b_k's are chosen to be between 0.9 and 1.1. For a given set of b_k's, the parameterized Rosenbrock function is minimized from an initial design point $\boldsymbol{x}$ = (1.1, 0.9. 1.1, 0.9. 1.1). Figure 2 is a design-line plot of f_o showing the noisy response of $f_o(b_1, b_2, b_3)$ when PORT with the programming error was used. It is a one-dimensional cut of the f_o response on eleven data points linearly interpolated between $\boldsymbol{b}$ = (0.9, 0.9, 0.9) and $\boldsymbol{b}$ = (1.1, 1.1, 1.1). The true response corresponds to results of PORT without the programming error. We can see that PORT with the programming error gave satisfactory results for only two out of the eleven runs. Following Eq. 1, we define optimization error as

$$e = f_o^{\,e} - f_o^{\,t}, \tag{12}$$

where f_o^e is the result of the erroneous PORT and f_o^t is the result of the correct PORT. It is apparent that poor optimizations result in heavier designs and the optimization error is one-sided.

To measure the average error of PORT on the parameterized Rosenbrock function, we need a sample set of optimization runs. We used 125 (= 5×5×5) data points from a full factorial experimental design of five levels in $\boldsymbol{b}$. PORT with the programming error was used to calculate f_o for each of the 125 variants of the parameterized Rosenbrock function. Two sets of 125 data points were generated by using two different initial $\boldsymbol{x}$'s: Case 1 using $\boldsymbol{x_0}$ = (1.1, 0.9, 1.1, 0.9, 1.1) and Case 2 using $\boldsymbol{x_0}$ = (0.9, 1.1, 0.9, 1.1, 0.9). Also, PORT without the programming error (denoted Case 0) was used to calculate the error of Case 1 and Case 2 according to Eq. 12. Table 2 summarizes the average errors. The errors were large compared to the true f_o, whose average is 0.00399; the average errors of Case 1 and Case 2 were 0.00658 (165%) and 0.00505 (127%), respectively. The average difference of f_o between Cases 1 and 2 was 0.00797 (200%) and the largest difference was as high as 1965%.

<u>Check of the Weibull Model</u>

We first check that the convergence errors for the Rosenbrock problems can be modeled by the Weibull distribution. A Weibull model was fit to the distribution of the optimization errors of Case 1 and Case 2 using the *weibfit* routine of MATLAB, and the results are summarized in Table 3. According to the p-values of the c^2 goodness-of-fit test, the fit to Case 1 was marginally rejected at the 0.05 significance level, while the fit to Case 2 was reasonable. The overall characteristics of the error can be described by $\hat{m}_{fit}$ and $\hat{s}_{fit}$ from the MLE fit. On the other hand, the mean and standard deviation can be estimated directly from e by Eq. 2. In Table 3, $\hat{m}_{fit}$ and $\hat{s}_{fit}$ were compared with $\hat{m}_{data}$ and $\hat{s}_{data}$. The agreement is good except for the standard deviation of Case 1, with 35.6% discrepancy.

The histograms in Figure 3 compare the shape of the error distribution to the Weibull fit results. The optimization error is nonnegative, and the probability for large error decreases rapidly to zero. Although the fit to Case 1 was marginally rejected by the c^2 test, the frequencies predicted by the error fit (solid line) show a reasonable match with the error data in Figure 3(a). The error fit to Case 2 in Figure 3(b) describes the error well. We conclude, therefore, that the Weibull model may be good enough to estimate the errors from pairs of poorly-converged runs, which we test in the next section.

Difference Fit for the PORT Optimization Error

Given the Weibull model for the error, we can use the pair of Cases 1 and 2 to estimate the error from only poorly-converged optimization data. Note that the initial $\boldsymbol{x}$ of Case 2 was selected such that it is a vertex point diagonally located with respect to the initial $\boldsymbol{x}$ of Case 1, to reduce possible dependence of the errors between Case 1 and Case 2. The correlation between e_1 (error of Case 1) and e_2 (error of Case 2) was low, -0.0371.

The results of the poorly converged data fit using the Weibull model are shown in Table 4. Note that the distribution parameters a and b of Case 1 and Case 2 are simultaneously estimated. Because there is no closed form for the probability density of the difference for the Weibull model, Eq. 8 was numerically integrated using Gaussian quadrature. According to the c^2 test, the difference fit was reasonable with a p-value of 0.4357.

From results of the difference fit, we can estimate the mean and standard deviation of the optimization error of the two cases involved (Table 4), which can be compared to the estimates from data using Eq. 2. The estimates for Case 1 were reasonable, with 2.7% discrepancy for the mean (m) and 6.1% discrepancy of the standard deviation (s). The estimates for Case 2 were in closer agreement with a 2.0% discrepancy for m and a 0.9% discrepancy for s. The results demonstrate the usefulness of the probabilistic model for the optimization error. By incorporating the difference data in the probability model, we were able to estimate the average error and standard deviation without obtaining accurate optimization results. Figure 3 shows that the error distributions predicted by the difference fit (dashed lines) describe well the error distribution and the difference fits are comparable to the error fits.

5. Estimation of Errors from the HSCT Structural Optimization

In the previous section, probabilistic modeling was applied to the optimization error due to programming error. An interesting observation for the Rosenbrock example was that the programming error happened to cause convergence error. We found that the Weibull distribution successfully modeled the convergence error. For the HSCT structural optimization, we have a more typical situation where inadequate convergence criteria resulted in convergence error. Unlike the Rosenbrock example, we do not know the true optimum for this practical engineering problem.

The authors [2] applied the probabilistic models to the errors of the HSCT structural optimization and showed that the Weibull distribution successfully modeled the error for several cases of different GENESIS convergence criteria. In addition, the difference fit approach was applied to sets of optimization results with different convergence criteria and gave reasonable estimates of the average error. Here we will show that the difference fit can also be applied to optimization runs with two different initial designs. Changing the initial design of the optimization is straightforward and has a computational advantage over tightening the convergence criteria.

Check of the Weibull Model

The Weibull model was fitted to Case 1 and Case 2, where the only difference is the choice of initial design point, and the results are summarized in Table 5. *p*-values of the c^2 test indicated a poor fit for Case 1, while the fit was acceptable for Case 2 with a 5% confidence level. Figure 4 compares histograms of the optimization error *e* with the predicted frequencies from the fitted Weibull models. It is seen that the error distribution has a mode near zero and decreases rapidly for large error. The Weibull fits give reasonable descriptions of the error distributions for both Case 1 and Case 2, although the c^2 test implied an unsatisfactory fit for Case 1.

The average errors $\hat{m}_{fit}$ estimated from the fit were in reasonable agreements with $\hat{m}_{data}$: -5.63% and –8.54% discrepancies for Case 1 and Case 2, respectively. The estimates of standard deviation $\hat{s}_{fit}$ from the fits were less accurate, particularly for Case 2, with a discrepancy of –14.6% and –23.4% for Case 1 and Case 2, respectively. Figure 4, comparing histograms of the error *e* data (bars) and the error fit (solid line), indicates that the Weibull model is suited for the optimization errors for both Case 1 and Case 2

Difference Fit of the Weibull Model

The difference fit was performed using the Weibull distribution on the pair of Cases 1 and 2. Recall that relatively large perturbations (multiplication factors between 0.1 – 1.9) were applied to the initial design point. The large perturbation was intended to reduce dependence of errors between Case 1 and Case 2, and the correlation coefficient was estimated to be 0.0565. The c^2 test on the optimization difference indicated a reasonable fit with a *p*-value of 0.5494. From the difference fit, we estimate the mean and standard deviation of the optimization error of each of the two cases involved. Table 6 shows that the estimates of mean error $\hat{m}_{fit}$ by the difference fit have reasonable agreements with $\hat{m}_{data}$: -14.7% and –19.4% discrepancies for Case 1 and Case 2, respectively. The estimates of standard deviation $\hat{s}_{fit}$ are also in a reasonable match with $\hat{s}_{data}$: 12.0% and 0.704% discrepancies for Case 1 and Case 2, respectively. Figure 4 shows that the error distributions predicted by the difference fit (dashed lines) are in reasonable agreement with the data, and the difference fits are comparable to the error fits.

For both the Rosenbrock function results and the HSCT structural optimization results, the Weibull model was useful in estimating convergence errors causing noisy optimization results. This information about the error distribution family helped us to use the difference fit effectively to estimate errors of low fidelity optimizations. In practice, some preliminary knowledge about the error distribution family may be sufficient for the difference fit, because parametric families of distributions like the Weibull and gamma offer considerable flexibility for representing data. Furthermore, a posteriori tests like c^2 test can be used to validate a distribution choice.

6. Concluding Remarks

This paper showed that a probabilistic model of optimization convergence errors can be used to obtain statistics of these errors from two sets of optimization runs with different starting points. Both the previous results based on varying convergence criteria and the examples presented in this paper indicated that the Weibull distribution may model optimization convergence errors sufficiently well for this purpose. One test problem was the parameterized Rosenbrock function, where a programming error caused convergence error. A second problem was structural optimization of the HSCT wing, where the optimal wing structural weight was inaccurate due to inadequate convergence criteria.

For both problems the Weibull model allowed estimation of the average error in poorly converged optimizations without requiring any fully converged runs. The approach of fitting the differences of pairs of poorly converged runs successfully estimated the averages and standard deviations of the errors. The difference fit can be applied by changing the convergence criteria, or by changing the initial design points. Since initial design points are simple and straightforward to change, one may easily apply the difference fit to estimate error statistics of various optimization problems.

Acknowledgements: This work has been supported by NSF grant DMI-9979711, NSF grant EIA-9974956, and AFOSR grant F49620-02-1-0090.

Footnotes

[*] Graduate Research Assistant, Department of Aerospace and Ocean Engineering, Virginia Polytechnic Institute and State University, currently at Phoenix Integration, Inc., Blacksburg, VA.

[†] Distinguished Professor, Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, FL.

[‡] Professor, Department of Aerospace and Ocean Engineering, Virginia Polytechnic Institute and State University.

[§] Professor, Departments of Computer Science and Mathematics, Virginia Polytechnic Institute and State University.

[¶] Professor, Department of Aerospace and Ocean Engineering, Virginia Polytechnic Institute and State University.

Key Words: structural optimization, convergence error, Weibull distribution, probabilistic error model

References

1. Kim, H., Papila, M., Mason, W. H., Haftka, R. T., Watson, L. T, and Grossman, B., "Detection and Repair of Poorly Converged Optimization Runs," *AIAA Journal*, Vol. 39, No. 12, pp. 2242-2249, 2001.
2. Kim, H., Haftka, R. T., Mason, W. H., Watson, L. T, and Grossman, B., "A Study of the Statistical Description of Errors from Structural Optimization," AIAA 2000-4840, 8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Long Beach, CA, September, 2000.
3. 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.
4. Myers, R. H. and Montgomery, D. C., *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*, John Wiley and Sons, New York, N. Y., 1995.
5. 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.
6. VMA Engineering, *GENESIS User Manual*, Version 5.0, Colorado Springs, CO, 1998.
7. Papila, M. and Haftka, R. T., "Uncertainty and Wing Structural Weight Approximations," AIAA Paper 99-1312, 40th AIAA/ASME/ASCE/ASC Structures, Structural Dynamics, and Material Conference, St. Louis, MO, April, 1999.
8. Papila, M. and Haftka, R. T., "Response Surface Approximations: Noise, Error Repair, and Modeling Errors," *AIAA Journal*, Vol. 38, No. 12, pp. 2336-2343, 2000.
9. Law, A. M. and Kelton, W. D., *Simulation Modeling and Analysis*, McGraw Hill, New York, 1982.
10. Hahn, G. J. and Shapiro, S. S., *Statistical Models in Engineering*, John Wiley and Sons, Inc., New York, 1994.
11. 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.
12. Vanderplaats Research & Development, Inc., *DOT Design Optimization Tools Users Manual*, Colorado Springs, CO, 1995.
13. Math Works, Inc., *Matlab Optimization Toolbox Users Guide*, Version 5, Natick, MA, 1997.
14. Gay, D. M., "Subroutines for Unconstrained Minimization Using a Model/Thrust-Region Approach," *ACM Transactions on Mathematical Software*, Vol. 9, No. 4, December, 1983, pp.503-512, see also http://www.bell-labs.com/projects /PORT.

Table 1: Average error of the HSCT structural optimization runs on 126 HSCT configurations for two cases of different initial design points.

Set of optimization runs	Case 1	Case 2
Description	Using the original initial point	Using a perturbed initial point from the original
Average W_s	85340 *lb.*	85202 *lb.*
Average error (% compared to the average of true W_s)	4458 *lb.* (5.51%)	4321 *lb.* (5.34%)
Average CPU time on a SGI Origin	75.5 sec.	76.5 sec.
Average of absolute differences of W_s between Cases 1 and 2 (% compared to the average of true W_s)	5942 *lb.* (7.3%)	
Maximum of absolute differences of W_s between Cases 1 and 2 (% compared to the average of true W_s)	58715 *lb.* (72.1%)	

Table 2: Summary of PORT runs for 125 variants of the parameterized Rosenbrock function.

Set of optimization runs	Case 1	Case 2
Description	With programming error, $\boldsymbol{x_0}$ = (1.1, 0.9, 1.1, 0.9, 1.1)	With programming error, $\boldsymbol{x_0}$ = (0.9, 1.1, 0.9, 1.1, 0.9)
Average error of f_o (% compared to the average of true f_o)	0.00658 (165%)	0.00505 (127%)
Average of absolute differences of f_o between Cases 1 and 2 (% compared to the average of true f_o)	0.00797 (200%)	
Maximum of absolute differences of f_o between Cases 1 and 2 (% compared to the average of true f_o)	0.0784 (1965%)	

Table3: Check of the Weibull model to the PORT optimization error of the parameterized Rosenbrock function.

Set of optimization runs	Case 1 $\boldsymbol{x_0}$ = (1.1, 0.9, 1.1, 0.9, 1.1)	Case 2 $\boldsymbol{x_0}$ = (0.9, 1.1, 0.9, 1.1, 0.9)
$\hat{m}_{data}$	0.00658 (165%)*	0.00505 (127%)
$\hat{m}_{fit}$	0.00692 (173%)	0.00496 (124%)
$(\hat{m}_{fit} - \hat{m}_{data})/\hat{m}_{data}$	5.2%	-1.8%
$\hat{s}_{data}$	0.00752 (189%)	0.0110 (276%)
$\hat{s}_{fit}$	0.0102 (256%)	0.0102 (256%)
$(\hat{s}_{fit} - \hat{s}_{data})/\hat{s}_{data}$	35.6%	-7.3%
a (shape parameter)	0.6941	0.5297
b (scale parameter)	0.005421	0.002744
p-value of c^2 test	0.0392	0.6989

* Percentage with respect to the average of true f_o

Table 4: Difference fit to optimization error of the parameterized Rosenbrock function.

Set of optimization runs	Case 1	Case 2
$\hat{m}_{data}$	0.00658 (165%)*	0.00505 (127%)
$\hat{m}_{fit}$	0.00676 (169%)	0.00515 (129%)
$(\hat{m}_{fit} - \hat{m}_{data})/\hat{m}_{data}$	2.7%	2.0%
$\hat{s}_{data}$	0.00752 (189%)	0.0110 (276%)
$\hat{s}_{fit}$	0.00798 (200%)	0.0111 (278%)
$(\hat{s}_{fit} - \hat{s}_{data})/\hat{s}_{data}$	6.1%	0.9%
a	0.8507	0.5134
b	0.006214	0.002698
p-value of c^2 test	0.4357	

* Percentage with respect to the average of true f_o

Table 5: Check of the Weibull model to the errors of the HSCT structural optimization.

Set of optimization runs	Case 1	Case 2
$\hat{m}_{data}$	4458 *lb*. (5.51%)*	4321 *lb*. (5.34%)
$\hat{m}_{fit}$	4207 *lb*. (5.20%)	3952 *lb*. (4.88%)
$(\hat{m}_{fit} - \hat{m}_{data})/\hat{m}_{data}$	-5.63%	-8.54%
$\hat{s}_{data}$	8383 *lb*. (10.4%)	9799 *lb*. (12.1%)
$\hat{s}_{fit}$	7157 *lb*. (8.85%)	7505 *lb*. (9.28%)
$(\hat{s}_{fit} - \hat{s}_{data})/\hat{s}_{data}$	-14.6%	-23.4%
a	0.6161	0.5646
b	2891	2415
p-value of c^2 test	0.0005	0.0925

* Percentage with respect to the average of true W_s

Table 6: Difference fit to the errors of the HSCT structural optimization.

Set of optimization runs	Case 1	Case 2
$\hat{m}_{data}$	4458 *lb*. (5.51%)*	4321 *lb*. (5.34%)
$\hat{m}_{fit}$	3804 *lb*. (4.70%)	3481 *lb*. (4.30%)
$(\hat{m}_{fit} - \hat{m}_{data})/\hat{m}_{data}$	-14.7%	-19.4%
$\hat{s}_{data}$	8383 *lb*. (10.4%)	9799 *lb*. (12.1%)
$\hat{s}_{fit}$	9393 *lb*. (11.6%)	9868 *lb*. (12.2%)
$(\hat{s}_{fit} - \hat{s}_{data})/\hat{s}_{data}$	12.0%	0.704%
a	0.4666	0.4262
b	1659	1236
p-value of c^2 test	0.5494	

* Percentage with respect to the average of true W_s

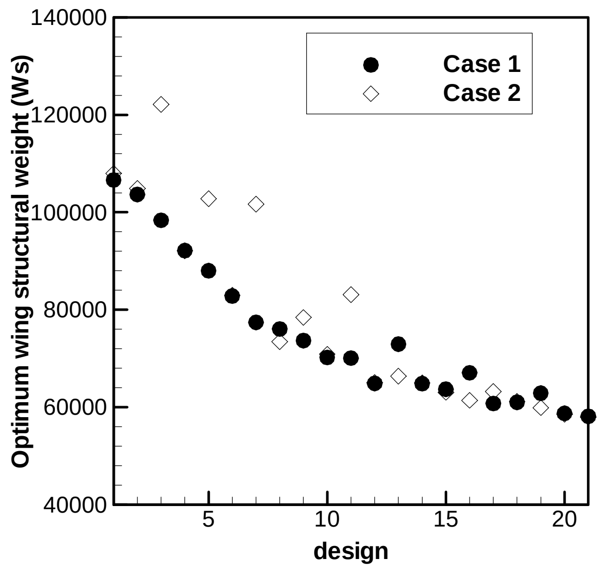


Figure 1: Noisy W_s response from structural optimization along a line connecting two extreme configurations. Case 1 and Case 2 used different initial design points.

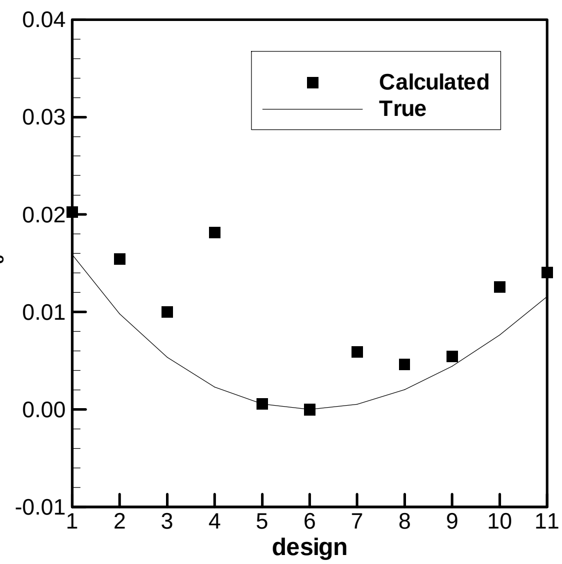
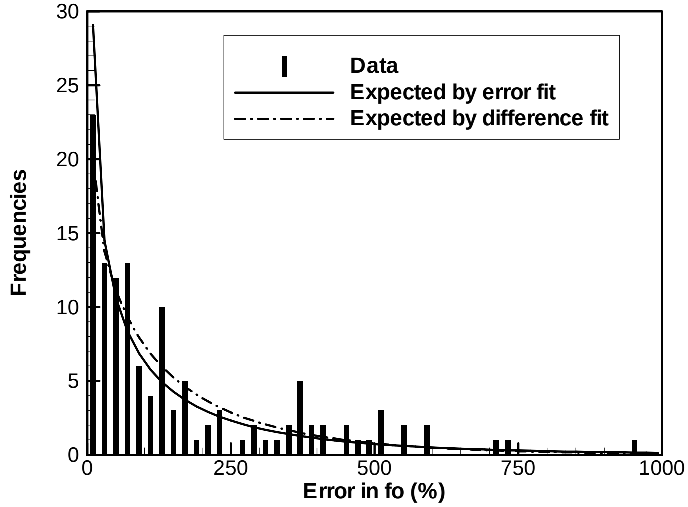
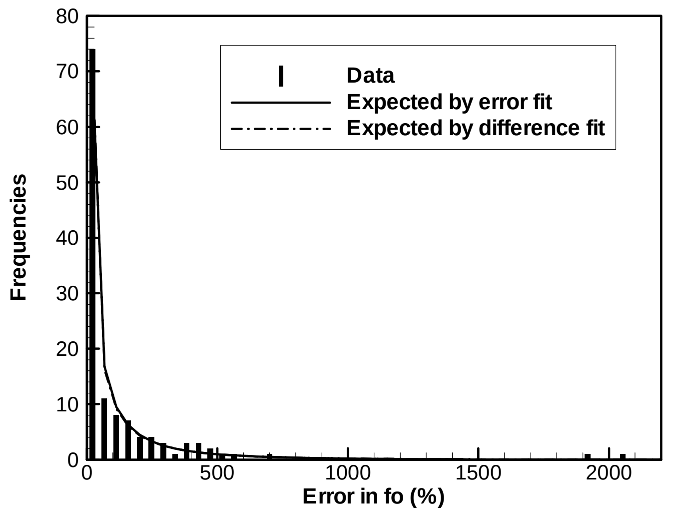


Figure 2: Design line plot of the optimal values (f_o) for the parameterized Rosenbrock function. The line is between $\boldsymbol{b}$ = (0.9, 0.9, 0.9) and $\boldsymbol{b}$ = (1.1, 1.1, 1.1).

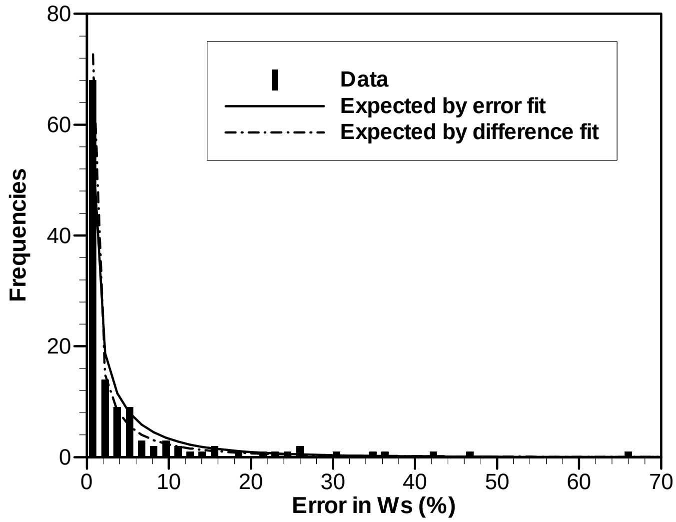


(a) For Case 1

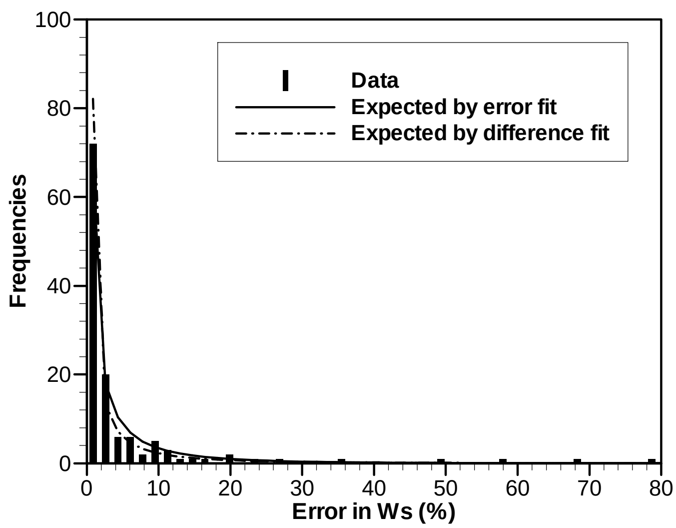


(b) For Case 2

Figure 3: Comparison of histograms between the optimization error data and the Weibull fits for the parameterized Rosenbrock function problem.



(a) For Case 1



(b) For Case 2

Figure 4: Comparison of histograms between the optimization error data and the Weibull fits for the HSCT structural optimization problem.