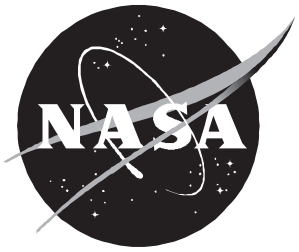
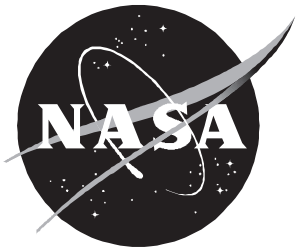
NASA Technical Paper 3329



Constrained Minimization of Smooth Functions Using a Genetic Algorithm

Daniel D. Moerder and Bandu N. Pamadi

November 1994



NASA Technical Paper 3329

Constrained Minimization of Smooth Functions Using a Genetic Algorithm

Daniel D. Moerder
Langley Research Center • Hampton, Virginia

Bandu N. Pamadi
ViGYAN, Inc. • Hampton, Virginia

National Aeronautics and Space Administration
Langley Research Center • Hampton, Virginia 23681-0001

November 1994

National Aeronautics and
Space Administration
Code JTT
Washington, D.C.
20546-0001

Official Business
Penalty for Private Use, $300

B U L K R A T E
POSTAGE & FEES PAID
NASA Permit No. G-27

Postmaster: If undeliverable (Section 158 Postal Manual) Do Not Return

Abstract

The use of genetic algorithms for minimization of differentiable functions that are subject to differentiable constraints is considered. A technique is demonstrated for converting the solution of the necessary conditions for a constrained minimum into an unconstrained function minimization. This technique is extended as a global constrained optimization algorithm. The theory is applied to calculating minimum-fuel ascent control settings for an energy state model of an aerospace plane.

Introduction

Genetic algorithms for optimization (refs. 1 to 4) are nonderivative, nondescent, random-search procedures for functional minimization, and their algorithmic structure is based on biological concepts. Familiar descent-type minimization algorithms construct a sequence of iterations, each of which modifies the independent variable vector from the previous iteration. Genetic algorithms, in contrast, construct a random sequence of generations in which a population of codings of bounded independent variable vectors is modified according to analogs of biological cross breeding and mutation. Rather than finalizing the value of the independent variable in an iteration by satisfying a descent condition, the genetic algorithm employs a "survival-of-the-fittest" heuristic that assigns a greater likelihood of appearing in the subsequent generation to the population elements that have lower objective function values than those that have higher objective function values.

A growing body of experimental evidence exists (refs. 5 to 8), supplemented by formal results (ref. 9), which indicates that genetic algorithms (GA's) are reliable methods for approximately determining the global minimum of a function. These algorithms lack a strict descent requirement, and their search operates on a population of iterates rather than on a single sequence of iterates. These features help prevent GA's from becoming "stuck" at local minima. On the other hand, GA's do not exploit derivative information in the search. This property, coupled with the fact that the algorithms operate on fairly coarse codings of the independent variable vector (rather than on floating point numbers), tends to limit the applicability of GA's to "rough-cut" analyses rather than highly accurate ones. When highly accurate solutions are required, GA's can be useful to generate initial guesses for gradient or Newton algorithms.

There have been a number of efforts in recent years to solve constrained optimization problems using GA's. The most straightforward approach is to convert the constrained problem into an unconstrained one by adding a penalty function on the constraint violation to the cost function. Difficulties exist, however, which are associated with both "light" and "heavy" penalty weightings, just as in the case of gradient-based optimization methods. When light penalties are employed, they generally fail to accurately enforce the constraint. When extremely heavy penalties are employed, that portion of the population which violates the constraints will have a vanishingly small probability of reproducing itself in subsequent generations. This "die-off" of illegal population elements results in an effectively smaller population (i.e., subsequent generations will have many replicates of the legal subset of the population and vanishingly few from the illegal subset). The resulting reduction in "genetic diversity" can adversely affect the performance of the algorithm.

This paper demonstrates a GA-based approach for solving nonlinearly constrained optimization problems. The method, which is simple to implement and generic to structure, is applicable to problems in which the cost function and the constraints are continuously differentiable. Moreover, this method can be adapted to calculate the global optimizer under nonrestrictive

assumptions. The algorithmic performance of the approach is explored in two numerical experiments. The first experiment compares the performance of the approach with a penalty function formulation for a simple test problem, and the second experiment extends the comparison to an aerospace performance optimization problem.

Symbols

B	heavy penalty weight in numerical examples
$\mathcal{B}$	user-defined volume
b	light penalty weight in numerical examples
C_D	drag coefficient
C_L	lift coefficient
C_M	aerodynamic moment coefficient
C_T	thrust coefficient
$\mathcal{C}^j$	set of j times continuously differentiable functions
c	cost function
$\bar{c}$	mean aerodynamic chord, ft
E	specific energy
$\mathcal{E}$	set of equality constraints
$\text{err}_{\mathbf{x}^*}$	Euclidean norm of distance between real value of best population element and known optimal solution point
f	constraint function
g	gravitational acceleration, ft/sec^2
h	altitude, ft
I_{SP}	specific impulse, sec
$\mathcal{I}$	set of inequality constraints
i, j, k, l	indices
K	penalty weight in global minimization formulation
$\mathcal{L}$	Lagrangian function
ℓ	line defined for search volume refinement
m	mass, slug
N_{pop}	number of population elements
N_{succ}	successful Newton-Raphson convergence
n	number of free parameters
P_{cross}	crossover probability
P_{mutate}	mutation probability
p	penalty function
Q_1, Q_2, Q_3	first through third quartiles

q	dynamic pressure, lbf/ft^2
$\mathcal{R}^j$	space of j-dimensional real vectors
r_{Earth}	equatorial Earth radius, ft
S	reference area, ft^2
T	thrust, lbf
vec( )	operator concatenating elements as vector
$\mathbf{x}$	vector of free parameters
α	angle of attack, deg
$\mathbf{\Gamma}$	system of equations for stationarity of Lagrangian function
δ_j	deflection of jth control effector, deg
ϵ	parameter defining transition from light to heavy penalty weight
η	fuel equivalence ratio
κ	number of new local minima identified in global minimization iteration
$\boldsymbol{\lambda}$	Lagrange multiplier
μ	number of inequality constraints
$\boldsymbol{\nu}$	function returning Lagrange multiplier vector at optimum
ξ	step-size scale factor in Newton-Raphson algorithm
ρ	atmospheric density, slug/ft^3
χ	user-specified search volume
Ψ	constrained minimization function

Subscripts:

CG	center of gravity
$(\ )_e$	pertaining to elevon deflection
$(\ )_{\text{pen}}$	pertaining to values returned by penalty function mimization
$(\ )_T$	pertaining to thrust
$(\ )_{\mathbf{x}}$	partial derivative with respect to $\mathbf{x}$

Acronyms and Abbreviations:

GA	generic algorithm
KT	Kuhn-Tucker conditions
max	maximum
min	minimum
NR	Newton-Raphson algorithm

Symbols with superscript stars $(\ )^*$, tildes $(\tilde{\ })$, plus signs $(\ )^+$, and zeros $(\ )^0$ indicate optimal value, active constraints, pseudo-inverse functions, and solutions returned from the global optimization algorithm interates, respectively. A bar above a symbol $(\bar{\ })$ indicates an admissible search value.

Problem Representation for Genetic Solution

We treat the problem of minimizing the $\mathcal{C}^1$ function $c(\mathbf{x})$ subject to $\mathcal{C}^1$ constraints; that is, $\mathbf{x}^* \in \mathcal{R}^n$ is sought such that

$$c(\mathbf{x}^*) \leq c(\mathbf{x}) \tag{1}$$

subject to

$$f_i(\mathbf{x}^*) = 0 \qquad (i \in \mathcal{E}) \tag{2}$$

$$f_j(\mathbf{x}^*) \geq 0 \qquad (j \in \mathcal{I}) \tag{3}$$

where $\mathcal{E}$ and $\mathcal{I}$ are the sets of indices of equality and inequality constraints, respectively. Identifying the active inequality constraint index set as

$$\widehat{\mathcal{I}} = \left\{ j : j \in \mathcal{I}, f_j(\mathbf{x}^*) = 0 \right\} \tag{4}$$

and defining

$$\hat{\mathbf{f}}(\mathbf{x}^*) = \text{vec}\left\{ f_k(\mathbf{x}^*) \ : \ k \in E \cup \widehat{\mathcal{I}} \right\} \tag{5}$$

assume that $\hat{\mathbf{f}} \in \mathcal{R}^\mu$, $\mu < n$, and

$$\text{rank} \left\{ \left. \frac{\partial \hat{\mathbf{f}}}{\partial \mathbf{x}} \right|_{\mathbf{x}^*} \right\} = \mu \tag{6}$$

If the above assumptions and equation (1) are true, then the Lagrange multipliers $\boldsymbol{\lambda}^*$ exist (ref. 9) such that equations (2) and (3) are satisfied:

$$\left. \frac{\partial \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda})}{\partial \mathbf{x}} \right|_{\mathbf{x}^*, \boldsymbol{\lambda}^*} = 0 \tag{7}$$

$$\lambda_j^* \geq 0 \qquad (j \in \mathcal{I}) \tag{8}$$

$$\lambda_k^* f_k(\mathbf{x}^*) = 0 \qquad (k \in \mathcal{E} \cup \mathcal{I}) \tag{9}$$

where

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = c(\mathbf{x}) - \sum_{k \in \mathcal{E} \cup \mathcal{I}} \lambda_k f_k(\mathbf{x}) \tag{10}$$

Equations (2) and (3) and (7) to (9), subject to equation (6), make up a typical statement of the first-order necessary conditions for a constrained local minimum, or Kuhn-Tucker (KT) conditions.

If $\boldsymbol{\lambda}^*$ were known, a GA could be used to satisfy the KT conditions by solving for the global minimum (zero) of

$$\Psi(\mathbf{x}, \boldsymbol{\lambda}^*) = \sum_{i=1}^{n} |\mathcal{L}_{\mathbf{x}_i}(\mathbf{x}, \boldsymbol{\lambda}^*)| + \sum_{j \in \mathcal{E}} |f_j(\mathbf{x})| + \sum_{k \in \mathcal{I}} |\min\{0, f_k(\mathbf{x})\}| \tag{11}$$

Note that, if one cast the first sum in equation (11) in the role of a cost function, then $\Psi(\mathbf{x}, \boldsymbol{\lambda}^*)$ has the structure of a typical penalty function formulation, but it has the penalty weights in the second two sums set to unity. The difference between the $\Psi(\mathbf{x}, \boldsymbol{\lambda}^*)$ and a penalty formulation of the constrained minimization problem is twofold. The first difference is that in the typical

case in which the solution is not known a priori, the optimal cost is unknown and may be nonzero. In this situation, it is well known that the minimum of the sum of cost and weighted penalty terms will vary with the selection of the penalty weighting parameters. In equation (11), however, all terms go to zero at $\mathbf{x}^*$; this situation results in the solution being invariant with respect to the nonunity scaling of the second two sums in equation (11). This property is advantageous because it eliminates ambiguity concerning the influence of penalty weightings on the solution. The second difference is that the "cost" in equation (11) is a measure of the constrained "stationarity" of the solution, rather than a direct measure of the performance.

Because $\boldsymbol{\lambda}^*$ is not generally known a priori, consider estimating $\boldsymbol{\lambda}^*$ in equation (11) during execution of the GA. Define

$$\widehat{\mathcal{I}}(\mathbf{x}) = \left\{ j : j \in \mathcal{I}, f_j(\mathbf{x}) \leq 0 \right\} \tag{12}$$

as an index set of constraints which are active or violated at a given $\mathbf{x}$, and

$$\tilde{\mathbf{f}}(\mathbf{x}) = \text{vec} \left\{ f_k(\mathbf{x}) : k \in \mathcal{E} \cup \widetilde{\mathcal{I}}(\mathbf{x}) \right\} \tag{13}$$

Now, estimate $\boldsymbol{\lambda}^*$ by $\boldsymbol{\nu}(\mathbf{x})$, where

$$\boldsymbol{\nu}_i(\mathbf{x}) = \begin{cases} \tilde{\boldsymbol{\nu}}_i(\mathbf{x}) & (i \in \mathcal{E}) \\ |\tilde{\boldsymbol{\nu}}_i(\mathbf{x}) & (i \in \widetilde{\mathcal{I}}(\mathbf{x})) \\ 0 & (i \in \mathcal{I}(\mathbf{x}) - \widetilde{\mathcal{I}}(\mathbf{x})) \end{cases} \tag{14}$$

and

$$\tilde{\boldsymbol{\nu}}(\mathbf{x}) = [\tilde{\mathbf{f}}_{\mathbf{x}}^T(\mathbf{x})]^+ c_{\mathbf{x}}(\mathbf{x}) \tag{15}$$

where $(\ )^+$ denotes the pseudoinverse operator. Note that from equations (6) to (10),

$$\boldsymbol{\nu}(\boldsymbol{x}^*) = \boldsymbol{\lambda}^* \tag{16}$$

The use of absolute values in equation (14) is an algorithmic measure to reject constrained stationary points that fail to satisfy equation (8). The KT conditions are satisfied by solving

$$\Psi[\mathbf{x}, \boldsymbol{\nu}(\mathbf{x})] = 0 \tag{17}$$

The nonsmooth equation (17) is solved in the rest of the paper by using a GA to solve the nonsmooth unconstrained minimization problem

$$\mathbf{x}^* = \arg \min_{\mathbf{x} \in \chi} \Psi[\mathbf{x}, \boldsymbol{\nu}(\mathbf{x})] \tag{18}$$

where χ is the user-specified bounded volume over which the genetic search takes place:

$$\chi = \{ \mathbf{x} : (\mathbf{x}_i)_{\min} \leq \mathbf{x}_i \leq (\mathbf{x}_i)_{\max} \qquad (i = 1, ..., n) \} \tag{19}$$

In this context, the GA provides a robust means to identify candidate local minima, in the sense of finding points that satisfy the KT conditions. Furthermore, because the minimum value of Ψ is known a priori, the GA can be stopped when $|\Psi|$ is "sufficiently small."

This latter characteristic can be useful in practice. Identification of the appropriate generation at which to terminate a GA minimization is an open research topic for general applications. For this reason, the length of the GA runs in practical studies is often set by the patience of the analyst or the availability of the computer resources. An objective threshold for termination can significantly shorten run times without loss of confidence in the solution.

Global Minimization Algorithm

The use of GA's to obtain solutions to the KT conditions, as expressed by equation (17), supplies the basis for a global optimization algorithm that is subject to the following two assumptions. Assumption 1 (i.e., smoothness) states that

$$\left.\begin{matrix} c(\mathbf{x}) \\ f_i(\mathbf{x}) \end{matrix}\right\} \in C^1 \qquad \left(i \in \mathcal{E} \cup \widehat{\mathcal{I}};\ \mathbf{x} \in \boldsymbol{\chi}\right)$$

Assumption 2 states that solutions $(\mathbf{x}^0, \boldsymbol{\lambda}^0)$ of

$$\boldsymbol{\Gamma}(\mathbf{x}^0, \boldsymbol{\lambda}^0) = \begin{bmatrix} \mathcal{L}_{\mathbf{x}}(\mathbf{x}^0, \boldsymbol{\lambda}^0) \\ \hat{\mathbf{f}}(\mathbf{x}^0) \end{bmatrix} = 0 \tag{20}$$

are regular at all $\mathbf{x}^0 \in \chi$. These assumptions lead to the following assertion, which is proved in appendix A: *If assumptions 1 and 2 are true, then there are a finite number of points* $\mathbf{x}^0 \in \chi$ *at which equation (20) is satisfied.*

The global constrained minimization procedure essentially consists of identifying all the local constrained inflection points $\{\mathbf{x}^0\}$ of $\mathcal{L}$ and then accepting that point which returns the lowest value of $c(\mathbf{x}^0)$ as the globally minimizing solution. By this assertion, the survey will be completed after a finite number of identifications.

The set χ can be surveyed for the global minimizer by a penalty function-based extension of the approach for solving local necessary conditions. At the conclusion of a successful GA execution of equation (18) for a given χ, there will be $\kappa \geq 1$ roots returned, $\{\mathbf{x}^0\} = \{\mathbf{x}^0_l, l = 1, ..., \kappa\}$, thus corresponding to local solutions of the necessary conditions. The problem of solving equation (18) can then be reposed to ensure that the roots $\{\mathbf{x}^0\}$ are excluded from the solution by replacing $\Psi[\mathbf{x}, \boldsymbol{\nu}(\mathbf{x})]$ with

$$\Psi'(\mathbf{x}, \{\mathbf{x}^0\}) = \Psi[\mathbf{x}, \boldsymbol{\nu}(\mathbf{x})] + K\sigma(\mathbf{x}, \{\mathbf{x}^0\}) \tag{21}$$

where $K > 0$ is a user-chosen penalty weighting term and σ is a function that becomes positive when $\mathbf{x}$ is "close" to any point in the set $\{\mathbf{x}^0\}$ but is otherwise zero. For example,

$$\sigma = \begin{cases} 1 & \left(\mathbf{x} \in \{\bigcup_{l=1}^{\kappa} \mathcal{B}(\mathbf{x}^0_l)\}\right) \\ 0 & \text{(Otherwise)} \end{cases} \tag{22}$$

where each $\mathcal{B}(\mathbf{x}^0_l)$ is a user-defined volume surrounding the corresponding $\mathbf{x}^0_l$. This approach can be generalized for a global survey of χ by using the GA to minimize the sequence of functions

$$\Psi_j(\mathbf{x}) = \Psi[\mathbf{x}, \boldsymbol{\nu}(\mathbf{x})] + K\sigma_{j-1} \qquad (j = 1, 2, ...) \tag{23}$$

$$\sigma_i(\mathbf{x}) = \begin{cases} 0 & \text{(i = 0)} \\ \sigma(\mathbf{x}, \bigcup_{k=0}^{i-1}\{\mathbf{x}^0\}_k) & \text{(i > 0)} \end{cases} \tag{24}$$

where $\{\mathbf{x}^0\}_k$ is the set of local solutions identified when minimizing $\Psi_{k-1}(\mathbf{x})$. The resulting algorithmic structure is as follows:

1. Set $j = 1$. Set $\text{cost}_0 \approx \infty$.
2. Generate a random population distributed over χ.

3. Execute a GA to minimize $\Psi_j(\mathbf{x})$ from equation (23). If the GA is unsuccessful in finding $\mathbf{x}$ such that $\Psi_j(\mathbf{x}) \approx 0$, go to step 8.

4. Collect $\{\mathbf{x}^0\}_j$ such that $\Psi_j(\mathbf{x})|_{\mathbf{x}\in\{\mathbf{x}^0\}_j} \approx 0$.

5. Set $\text{cost}_j = \min\left\{\text{cost}_{j-1}, \min_{\mathbf{x}\in\{\mathbf{x}^0\}_j} c(\mathbf{x})\right\}$.

6. If $\text{cost}_j < \text{cost}_{j-1}$, then set $\mathbf{x}_*^0 = \arg\ \text{cost}_j$.

7. Set $j = j + 1$. Go to step 2.

8. If $j > 1$, accept $\mathbf{x}_*^0$ as global minimizer; if $j \not> 1$, abort.

The algorithm terminates when it is no longer able to identify values of $\mathbf{x} \in \chi$ for which $\Psi_j \approx 0$. It is presumed that this occurs when all the local solutions have been identified and included in the penalty term of Ψ_j's.

Note that the proposed algorithm would still terminate in a finite number of iterations even if local nonunique roots of $\mathbf{\Gamma}$ exist, thus invalidating the assertion. The termination occurs because the penalties K that exclude a growing union of finite subvolumes from χ. On the other hand, the resolution of the algorithm to separately identify the closely spaced solution points depends on the selection of the volumes $\mathcal{B}$. Recall, however, that the motivation for selecting a GA over a more accurate method is to robustly obtain a "rough-cut" answer. Therefore, it is not felt that this latter concern prevents the algorithm from having practical utility.

The concept of using penalty functions to exclude known local minima from future iterations of a global optimization algorithm has been established for descent-based methods (refs. 10 to 12). The penalty function employed in this work is similar to the "tunneling" technique developed in references 10 and 11. In these references, the penalty varies smoothly toward a huge value as $\mathbf{x} \to \mathbf{x}_l$. Penalty functions of this type can lead to numerical difficulties in algorithms that use gradient information for defining a search direction. These difficulties are avoided in the present approach because of the nondescent nature of the GA.

Numerical Experiments

This section describes the numerical experiments that explore the performance of the GA-based constrained minimization procedure developed in this paper. Because the procedure involves more complexity than formulations in which constraints are enforced via penalty functions, the GA-based solutions of equation (18) are compared with the GA-based solutions of a "generic" penalty function problem formulation of the form

$$\mathbf{x}_{\text{pen}}^* = \arg\min_{\mathbf{x}\in\chi}\left\{c(\mathbf{x}) + \sum_{k\in\mathcal{E}\cup\mathcal{I}} p[\mathbf{x}, f_k(\mathbf{x})]\right\} \tag{25}$$

where

$$p[\mathbf{x}, f_k(\mathbf{x})] = \begin{cases} B\cdot|f_k| & (|f_k| > \epsilon) \\ b\cdot|f_k| & (|f_k| \le \epsilon) \end{cases} \tag{26a}$$

when $k \in \mathcal{E}$ and

$$p[\mathbf{x}, f_k(\mathbf{x})] = \begin{cases} B\cdot|f_k| & (-f_k > \epsilon) \\ b\cdot\max\{0, -f_k\} & (-f_k \le \epsilon) \end{cases} \tag{26b}$$

when $k \in \mathcal{I}$. The parameters $\epsilon > 0$ and $B \ge b > 0$ are to be chosen by the user. This formulation allows heavy penalties that strongly violate constraints and light penalties that

"nearly" comply with constraints to be applied to $\mathbf{x}$. Heavy penalties could be appropriate to reject the "artifact" local minima in equation (25) which would not approximately correspond to the solutions of the KT conditions for the underlying problem. The provision for lighter penalties within the ϵ-defined region of light constraint penalty is intended to provide a subset of χ in which the variation of $c(\mathbf{x})$ is not dominated by the penalty terms.

The GA used in this study was a simple GA, which was similar to that used in reference 8; however, it included a modification that was suggested in reference 13. In this modification, the best-valued population element from each generation was guaranteed survival into the next generation. The real-valued independent variables were coded as 8-bit binary strings such that

$$\mathbf{x}(\text{string}) = x_{\min} + (x_{\max} - x_{\min})\frac{\text{base}_{10}(\text{string})}{2^8 - 1} \tag{27}$$

thus yielding a resolution of roughly 0.4 percent over the range of χ for each problem. In both experiments, the string representations for the vector-valued independent variables were formed by concatenating the 8-bit substrings for each scalar. Key parameters affecting the performance of a GA (ref. 3) are population size N_{pop}, crossover probability P_{cross}, and mutation probability P_{mutate}. All runs in this study were made with $N_{\text{pop}} = 30$ and $P_{\text{cross}} = 0.95$, as per guidelines from reference 3. This simple GA with the modification from reference 13, however, appears to benefit from a more aggressive mutation rate than the $P_{\text{mutate}} = 0.01$ that was recommended in reference 3. Some adjustment of this parameter was done in the experiments below.

The first experiment (denoted example 1 in the table titles of tables 1 to 5) compares the performance of the generic GA penalty function approach with that of the GA solution of reference 11, henceforth referred to as a KT solution, for the problem

$$c(x_1, x_2) = x_1^2 + x_2^2$$

$$f(x_1, x_2) = x_1 - x_2 - 2 = 0$$

with the search volume

$$\chi = \{-5 \le x_1 \le 5\,; -5 \le x_2 \le 5\}$$

Monte Carlo experiments of 100 runs, each consisting of 100 generations, were performed for the KT formulation (ref. 11), and for the penalty formulation for a number of combinations of $\{B, b, \epsilon, P_{\text{mutate}}\}$. The performance results are given in tables 1 to 5. Table 1 displays the KT performance for $P_{\text{mutate}} = 0.01, 0.03, 0.05, 0.07$, and tables 2 to 5, in turn, display the GA penalty function results for each value of P_{mutate}. The performance of the GA in these experiments was characterized by the number of runs that successfully satisfied error thresholds of the form $\text{err}_{\mathbf{x}^*} \le k$, where $\text{err}_{\mathbf{x}^*}$ is the Euclidean norm of the distance between the real value of the best population element and the known optimal solution point $\mathbf{x}^* = (1, -1)$. Tables 2 to 5 also display the median number of generations necessary for the successful runs to cross the thresholds.

The comparison of table 1 with tables 2 to 5 immediately reveals that the KT formulation was generally more successful in finding the optimum for this problem and, when successful, tended to find it more quickly, except in the case of $P_{\text{mutate}} = 0.01$. Note that, both for the KT and penalty approaches, $P_{\text{mutate}} = 0.05$ and 0.07 returned significantly better performance than the lower values. The case $P_{\text{mutate}} = 0.07$ did not show any significant advantage over $P_{\text{mutate}} = 0.05$ in the penalty function runs, and it actually resulted in a small performance degradation in the KT experiment. If attention is restricted to the penalty results, then a closer examination of tables 4 and 5 suggests that all the combinations of small b ($b = 10$) and relatively large ϵ

($\epsilon = 0.1$ and 0.2) tend to outperform other parameter combinations, particularly for success in the $\text{err}_{\mathbf{x}^*} \leq 0.1$ criterion.

The success of these parameter combinations, which are cases 8, 9, 17, 18, 23, and 24 of tables 4 and 5, can be loosely interpreted in terms of the processes operating in the GA. Although the comparatively heavy B penalties result in a population being "killed off" outside the region $|f| < \epsilon$, the high mutation rates tend to introduce enough "new genetic information" to prevent the population elements $\mathbf{x}$ from stagnating; that is, they accumulate away from the actual minimizing value of $\mathbf{x}$ simply because other population elements fall outside $|f| < \epsilon$ and were eliminated by the B penalty. Similarly, the combination of larger values of ϵ and low b was advantageous because it assigned more volume in the parameter space to the population elements that could be expected to survive through enough generations to exchange meaningful amounts of information through crossover operations.

The second experiment (denoted Example 2 in the table title of table 6) extends the comparison of KT and penalty function algorithm performance to a more challenging optimization problem: selecting optimizing altitude and control settings for an energy-state approximation (ref. 14) of the minimum-fuel ascent to orbit for the "Langley Accelerator" (ref. 15) aerospace plane concept. In this experiment, a thrust-vectoring capability is added to the model. The energy-state approximate solution for this problem is calculated by performing algebraic minimizations for altitude and controls along a locus of specific energies E leading to orbital injection. This experiment considers the algebraic minimization at a single value of E. The search variable ranges are

$$\bar{\mathbf{x}} \in \chi = \begin{cases} -1 \leq \alpha \leq 12 \\ 20\,000 \leq h \leq 30\,000 \\ -20 \leq \delta_e \leq 20 \\ -20 \leq \delta_T \leq 20 \\ 0.5 \leq \eta \leq 1.5 \end{cases}$$

and the cost function is $-dE/dm$, where m is mass. At a given value of E, the nondimensional cost is expressed as

$$c(\bar{\mathbf{x}}) = V(E,h) I_{SP}(\eta) \left[\cos(\delta_T + \alpha) - \frac{C_D(\alpha, \delta_e)}{C_T(\eta)} \right] \frac{1}{E} \tag{28}$$

where $V = \sqrt{2g(E-h)}$ and g is the gravitational acceleration, which is assumed to be constant. Expressions for the coefficients I_{SP} and $C_{(\ )}$ and for all constants in this problem are given in appendix B. Two equality constraints appear. The first constraint is a vertical acceleration balance

$$\frac{1}{mg} \left(q(E,h) S \left[C_L(\alpha, \delta_e) + C_T(\eta) \sin(\alpha + \delta_T) \right] + m \left\{ \frac{[V(E,h)]^2}{r_{\text{Earth}}} - g \right\} \right) = 0 \tag{29}$$

where S is the reference area. The second constraint is a pitch moment balance

$$\frac{1}{C_T(\eta) x_T} \{ C_M(\alpha, \delta_e) \bar{c} + x_{CG} [C_D(\alpha, \delta_e) \sin\alpha + C_L(\alpha, \delta_e) \cos\alpha] \} - \sin\delta_T = 0 \tag{30}$$

where $q = \rho V^2/2$ is the dynamic pressure and x_{CG} and x_T are the moment arms. There is also an inequality constraint on dynamic pressure, which is

$$1 - \frac{q}{q_{\max}} \geq 0 \tag{31}$$

where $q_{\max} = 2000$ lbf/ft^2. The specific energy ($E = 10^5$ ft) considered here approximately corresponds to flight at Mach 2.

In this second example, KT and penalty function versions of the problem were again compared in the Monte Carlo experiments. In this case, each experiment consisted of 100 trial GA runs, with each run having 600 generations. Two evaluation criteria were employed to compare the quality of the KT and the penalty results. The first criterion was the distribution of values of $\Psi_1(\bar{\mathbf{x}})$ from equation (23); these values were returned by the populations of final-generation $\bar{\mathbf{x}}$'s from the KT and the penalty run sets. The second criterion informally quantified the usefulness of the GA results as initial guesses for high-accuracy Newton-type methods. This quantification was done by using the final-generation $\bar{\mathbf{x}}$'s from the GA's as initial guesses for a restricted-step Newton-Raphson (NR) algorithm that solved a system of equations equivalent to the KT conditions for this problem:

$$\widehat{\Psi}_i(\bar{\mathbf{x}}^*) = \{[I - \hat{f}_{\mathbf{x}}(\hat{f}_{\mathbf{x}}^T)^+]c_{\mathbf{x}}\}_i^2 + \hat{f}_i^2 = 0 \qquad (i = 1, ..., 5) \tag{32}$$

where $f(\bar{\mathbf{x}})$ is the concatenation of equations (29) to (31), with equation (31) treated as an equality and $\hat{f}^T(\bar{\mathbf{x}}) = [f^T(\bar{x})\ 00]$. At the optimal solution,

$$(\bar{\mathbf{x}}^*)^T = [1.0211,\ 20\,385,\ 0.1789,\ 0.7182,\ 0.7688]$$

where the treatment of equation (31) as an equality constraint is justified because the value of its Lagrange multiplier is positive at $\bar{\mathbf{x}}^*$, thus taking on the value of $\boldsymbol{\lambda}_{q_{\max}} = 6.1866$.

The NR iteration took the form

$$\bar{x}_{k-1} = \bar{x}_k + \xi_k s_k \qquad s_k = -[\hat{g}_x(x_k)]^{-1}\hat{g}(x_k) \tag{33}$$

where $\hat{g}_x(x_k)$ was approximated by a first-order forward difference formula, and the line search parameter ξ_k was chosen by the logic, iterated over $j = 0, 1, ...,$ as

$$\begin{aligned} (\xi_k)_0 &= \min\{2\xi_{k-1}, 1\} \\ (\xi_k)_j &= \left\{ \begin{array}{ll} (\xi_k)_j/2 & (\hat{g}[\bar{x}_k + (\xi_k)_j s_k] \geq \hat{g}(\bar{x}_{k-1})) \\ \text{Abort} & ((\xi_k)_j \leq \xi_{\min}) \end{array} \right\} \end{aligned} \tag{34}$$

where $\xi_{\min}$ was chosen as 10^{-7}. The algorithm was considered to have converged if the criterion $\sum_1^5 \hat{g}_i(\bar{\mathbf{x}}) < 10^{-4}$ was satisfied in 100 iterations or fewer.

First, GA minimization of the KT formulation was considered. A Monte Carlo set of 100 GA runs minimizing $\Psi_1(\mathbf{x})$ over χ was calculated. Figure 1 displays the resulting distribution of Ψ_1 values, referred to as KT errors. The median Ψ_1 for these runs is $\Psi_1 = 0.3821$, which corresponds to

$$(\bar{x}_0^T)_{\text{median}} = [0.8353,\ 24\,980,\ -1.9608,\ 8.8627,\ 0.7745]$$

An examination of the final population elements from the set of trials revealed that, rather than clustering around a point, the δ_e and δ_T components of the $\bar{\mathbf{x}}$'s were distributed along a line. This distribution is not surprising, given the correlation in the effects of δ_e and δ_T on pitching moment. There was also a significant linear trend of the deflections with α. Figure 2 displays relationships.

Figure 3 gives the corresponding distribution of α and η as functions of h. The trends in these variables are not as strong as those seen in the relations among α, δ_e, and δ_T. Part of the reason for this is that the $q_{\max}$ constraint (eq. (31)) only weakly affects the performance for this

model, even though q at the maximum altitude is little more than one-half of $q_{\max}$. Note that the trends in figure 3 are explained by the reduction of q with the increase in altitude. A larger α is called for because of the dimunition in lift, and a slightly larger η is called for to balance the reduction in $\cos(\delta_T + \alpha)$ and the increase in $C_D(\alpha, \delta_e)$ in equation (28).

In 100 trials, the lowest value of Ψ_1 achieved was no better than 0.0979; this occurence can be explained by the properties of Ψ_1 for this problem and by the characteristics of the crossbreeding and reproduction operations in GA's. Near its minimum, Ψ_1 is least sensitive along the locus whose projection into $(\alpha \times \delta_e \times \delta_T)$-space is depicted in figure 2. In the reproduction operation of the GA's, population elements away from this locus have significantly higher Ψ_1 values and, therefore, are assigned a significantly lower probability of surviving into the next generation. The GA "crossbreeding" operation modifies pairs of population elements by swapping substrings from binary codings of both elements. When "near-optimum" values of Ψ_1 are distributed along a surface, rather than a point, the swapping generally results in moving the modified elements away from the surface. These elements, in turn, lose reproduction probability and disappear from the population.

To address this difficulty, δ_e and δ_T were transformed to tailor the search volume χ to the behavior of Ψ_1 such that

$$\delta_e = \ell_e(\alpha) + \tilde{\delta}_e \qquad \delta_T = \ell_T(\alpha) + \tilde{\delta}_T \tag{35}$$

where the coefficients in

$$\left.\begin{aligned} \ell_e(\alpha) &= (c_1)_e \alpha + (c_2)_e \\ \ell_T(\alpha) &= (c_1)_T \alpha + (C_2)_T \end{aligned}\right\} \tag{36}$$

were calculated by least-squares fits over the data from the Monte Carlo experiment. The values for the coefficients and for the residuals measure

$$s_{(\ )} = \sqrt{\sum_{i=1}^{100} \left[\ell_{(\ )}(\alpha_i) - (\delta_{(\ )})_i \right]^2} \tag{37}$$

were (in degrees)

$$\begin{aligned} (c_1)_e &= 288.2072 & (c_1)_T &= -291.4726 \\ (c_2)_e &= -4.2551 & (c_2)_T &= 8.6639 \\ s_e &= 3.6505 & s_T &= 7.9568 \end{aligned}$$

The new search vector was chosen as $\tilde{\mathbf{x}}^T = [\alpha, h, \tilde{\delta}_e, \tilde{\delta}_T, \eta]$, and the search volume was redefined as

$$\tilde{x} \in \tilde{\chi} = \begin{cases} 0.1127 \leq \alpha \leq 2.7832 \\ 20\,227 \leq h \leq 31\,356 \\ -3.6505 \leq \tilde{\delta}_e \leq 3.6505 \\ -7.9568 \leq \tilde{\delta}_T \leq 7.9568 \\ 0.7381 \leq \eta \leq 0.8194 \end{cases}$$

The bounds in $\tilde{\chi}$ for α, h, and η were chosen as the sample mean values ± 1.5 times the sample standard deviation. The bounds on $\tilde{\delta}_e$ and $\tilde{\delta}_T$ were $\pm s_e$ and $\pm s_T$, respectively.

Again, a Monte Carlo set of 100 runs was performed. Figure 4 displays the distribution of Ψ_1 for this experiment. The median value of Ψ_1 for this set of runs was $(\Psi_1)_{\text{median}} = 0.0627$, and the lowest value was $(\Psi_1)_{\text{best}} = 0.0074$, with the corresponding control vectors

$$(\bar{\mathbf{x}}_1^T)_{\text{median}} = [1.0867,\ 23\,064,\ 1.9988,\ 1.2944,\ 0.7699]$$

$$(\bar{\mathbf{x}}_1^T)_{\text{best}} = [1.2543,\ 25\,770,\ 2.1832,\ 1.8149,\ 0.7722]$$

expressed in the original $\bar{\mathbf{x}}$ coordinates. Figure 5 gives the distribution of α, δ_e, and δ_T, along with the boundaries of $\widetilde{\chi}$, shown as straight lines.

The GA-generated KT solutions were next used as initial guesses for the NR algorithm. Eighty-two of the NR runs converged from the first set of 100 GA solutions, based on optimization over the full χ. Ninety-eight of the NR runs converged of the second set, based on optimization over the restricted $\widetilde{\chi}$. For comparative purposes, the NR algorithm was executed for 100 initial guesses that were chosen from a uniform distribution over χ. Twenty-three of the NR runs converged in this case. We infer that the use of a genetic algorithm to minimize Ψ_1 provided an effective means of generating an initial guess for the NR algorithm for this example.

Once the performance of the KT approach for this problem was established, the penalty scheme (eq. (25)) was applied to the problem of minimizing equation (28) subject to equations (29) to (31). Sequences of 100 Monte Carlo runs were performed with $P_{\text{mutate}} = 0.05$ for the various penalty parameter combinations from the first example. Each of the sequences of runs was begun from the same random number seed. Because cases 1 to 3 and 10 to 12 (all of which used the penalty weighting $b = 10^4$) performed poorly in the first example, they were eliminated from consideration in this experiment.

Table 6 summarizes the algorithm performance for these cases. This table displays the minimum, first through third quartile, which is displayed as Q_1, Q_2, Q_3, and the maximum values of the KT cost function Ψ_1 which is evaluated at the penalty function solution points. In addition, table 6 displays the number of successful NR convergences, denoted by N_{succ}, achieved using the solutions as initial guesses. As was observed in the first example, cases 8, 9, 17, 18, 23, and 24 tended to provide better results than those of other parameter combinations, in the sense that Ψ_1 errors tended to be smaller and N_{succ} tended to be larger than with other combinations.

Cases 23 and 24 were the most successful pair in producing good initial guesses for the NR runs. The slight advantage seen in these latter two cases may be attributed to their employment of the smallest B penalty weight in the study. Because of the small B weight, the nonconstraint-compliant population elements are granted a somewhat higher likelihood of reproduction and are, thus, more likely to enhance the"genetic diversity" of the population. Nonetheless, the most striking characteristic of the data in table 6 is that even the best results from the penalty runs compare poorly with the results of the KT experiments. The former solutions are less "close" to the optimum than those of the latter, in the sense of vanishing Ψ_1, and they did not provide particularly reliable initial guesses for subsequent NR solutions.

Figure 6 displays the detailed Ψ_1 cost distribution for case 23. Figure 7 gives the distribution of α, δ_e, and δ_T from these runs, along with the optimal solution and the case 23 solution that returns the lowest value of the penalty-based cost function from equation (25). Note that this distribution is markedly different from those in figures 2 and 4. Although the best penalty-based solution,

$$(\bar{\mathbf{x}}_{\text{pen}}^T)_{\text{best}} = [1.1926,\ 233\,017,\ 0.2354,\ 0.8630,\ 0.7591]$$

is fairly close to $\bar{\mathbf{x}}^*$, the overall trends of the penalty solutions are significantly different from the KT solutions. Comparing figures 2 and 4 with figure 7, note that there is a strongly linear trend between δ_T and δ_e in all sets of solutions, but the slope of the penalty solution trends is opposite in sign to the KT solution trends and much different in slope magnitude. Also, the variation of α in the penalty function solutions is much smaller than that in the KT solutions.

The solution $\bar{\mathbf{x}}^*$ was tentatively verified as a global optimizer for this example by performing 100 of the 600-generation GA runs that minimize Ψ_2 from equation (23), using the $\tilde{\mathbf{x}}$ coordinates from equations (35) and (36), and

$$\mathcal{B}(\tilde{\mathbf{x}}^*) = \begin{cases} \alpha = \alpha^* \\ h = h^* \\ \tilde{\delta}_e^* - s_e \le \tilde{\delta}_e \le \tilde{\delta}_e^* + s_e \\ \tilde{\delta}_T^* - s_T \le \tilde{\delta}_T \le \tilde{\delta}_T^* + s_T \\ \eta = \eta^* \end{cases}$$

This form of $\Psi_2(\tilde{\mathbf{x}})$ was intended to deny the locus of the $(\delta_e\,, \delta_T)$ pairs from the Ψ_1 search to the GA. The best value of Ψ_2 from this set of runs was 1.2715. Because this number is considerably higher than the worst value of Ψ_1, we infer that no other local minima were identified and, hence, $\bar{\mathbf{x}}^*$ is the global minimizer.

Summary of Results

This paper has examined the use of a simple genetic algorithm to solve minimization problems for differentiable functions that are subject to differentiable equality and inequality constraints. The first-order necessary conditions for a constrained minimum have been adapted to convert a given constrained minimization problem into an unconstrained minimization of a nonsmooth function whose minimum value is zero and whose minimization is equivalent to satisfying the first-order necessary conditions for the original problem. The unconstrained nonsmooth minimization is carried out using the genetic algorithm.

This solution approach was exercised and compared with a penalty function formulation for two constrained minimization problems. In the first problem, the approach significantly outperformed the penalty function technique over a range of penalty function tuning parameters. In the second problem, the approach provided significantly more accurate solutions than the penalty function technique, despite numerically challenging features, such as correlated control variables.

NASA Langley Research Center
Hampton, VA 23681-0001
March 14, 1994

Appendix A

Proof of Assertion

Assume that an infinite number of solutions $\hat{\mathbf{x}}_k$ exist in χ with corresponding values of $\hat{\boldsymbol{\lambda}}_k = \boldsymbol{\nu}(\hat{\mathbf{x}}_k)$ from equation (15). This assumption and the fact that χ is closed and bounded implies that $\{\hat{\mathbf{x}}_k\}$ will contain an accumulation point $\bar{\mathbf{x}} \in \chi$. Construct a sequence of points $\hat{\mathbf{x}}_k \to \bar{\mathbf{x}}$, and consider variation of $\|\boldsymbol{\Gamma}[\tilde{\mathbf{x}}, \boldsymbol{\nu}(\tilde{\mathbf{x}})]\|$ along the lines

$$\tilde{\mathbf{x}}_k(\alpha) = (1-\alpha)\bar{\mathbf{x}}_k + \alpha\hat{\mathbf{x}} \qquad (0 \le \alpha \le 1) \tag{A1}$$

By the extreme value theorem, for each k, there exist α_k that satisfy

$$\kappa_k = \max_{\alpha} \|\boldsymbol{\Gamma}\{\tilde{\mathbf{x}}(\alpha), \boldsymbol{\nu}[\tilde{\mathbf{x}}(\alpha)]\}\| \tag{A2}$$

As $k \to \infty$,

$$\frac{\kappa_k}{\alpha_k \|\hat{\mathbf{x}}_k - \bar{\mathbf{x}}\|} \to \left\langle \left. \frac{\partial \|\boldsymbol{\Gamma}(\mathbf{x})\|}{\partial \mathbf{x}} \right|_{\bar{\mathbf{x}}}, \frac{\hat{\mathbf{x}}_k - \bar{\mathbf{x}}}{\|\hat{\mathbf{x}}_k - \bar{\mathbf{x}}\|} \right\rangle \tag{A3}$$

If $\kappa_k \neq 0$ as $k \to \infty$, assumption 1 of the assertion is violated. Because of assumption 1

$$\kappa_k \le M \|\hat{\mathbf{x}}_k - \bar{\mathbf{x}}\| \tag{A4}$$

for some constant $M > 0$. This implies that $\kappa_k \to 0$ as $k \to \infty$, so that

$$\operatorname{rank} \left[\frac{\partial \boldsymbol{\Gamma}(\mathbf{x})}{\partial \mathbf{x}} \right]_{\mathbf{x}=\bar{\mathbf{x}}} < n \tag{A5}$$

from equation (A3). Equation (A5), however, implies that there exists $\theta^T = [\delta\mathbf{x}^T, \delta\boldsymbol{\lambda}^T] \neq 0$ such that

$$\nabla \boldsymbol{\Gamma}(\mathbf{x}, \boldsymbol{\lambda})|_{\bar{\mathbf{x}}, \boldsymbol{\lambda}=\boldsymbol{\nu}(\bar{\mathbf{x}})} \theta = 0 \tag{A6}$$

which violates assumption 2. This contradiction proves the assertion.

Appendix B

Smoothed Aerospace Plane Model

This appendix describes the smooth analytical adaptation of the piecewise linear tabular "Langley Accelerator" vehicle model described in reference 15. The following aerodynamic and propulsion coefficient expressions are intended for flight conditions between Mach 2 and 2.5:

$$C_L(\alpha, \delta_e) = -0.0062 + 0.0242\alpha - 0.00067\delta_e + (0.896 \times 10^{-7})\alpha\delta_e$$

$$C_D(\alpha, \delta_e) = 0.0261 - 0.000206\alpha + 0.000526\alpha^2 - 0.000025\alpha\delta_e + 0.0000124\delta_e^2$$

$$C_M(\alpha, \delta_e) = 0.00102 - 0.0032\alpha - 0.0002\alpha^2 + 0.0005\delta_e$$

$$I_{SP}(\eta) = 3713 + 1208\eta - 1740\eta^2$$

$$C_T(\eta) = 0.0062 + 0.1316\eta - 0.0182\eta^2$$

where all angles are expressed in degrees. The variation of these quantities with Mach number has been ignored for simplicity. Figures 8 to 12 display the errors between the above analytical expressions and the linearly interpolated values from reference (22); the values are normalized by the latter and expressed as percentages. The vehicle-related constants appearing in the text are as follows:

$$S = 3603 \text{ ft}^2$$

$$\bar{c} = 80 \text{ ft}$$

$$\mathbf{x}_{CG} = 14.01 \text{ ft}$$

$$\mathbf{x}_T = 61.99 \text{ ft}$$

$$m = 4800 \text{ slugs}$$

Figure 13 displays the geometry of the vehicle. Finally, the atmospheric density model for this study is

$$\rho(h) = \rho_0 \exp(-\beta h)$$

where $\rho_0 = 0.002378$ slug/ft^3 and $\beta = 0.0000547$ 1/ft.

References

1. Holland, John H.: *Adaptation in Natural and Artificial Systems: An Introductory Analysis With Applications to Biology, Control, and Artificial Intelligence.* Univ. of Michigan Press, 1975.

2. De Jong, Kenneth Alan: *An Analysis of the Behavior of a Class of Genetic Adaptive Systems.* Ph.D. Diss., Univ. of Michigan, 1975.

3. Grefenstette, John J.: Optimization of Control Parameters for Genetic Algorithms. *IEEE Trans. Syst., Man, & Cybern.*, vol. SMC-16, no. 1, Jan./Feb. 1986, pp. 122–128.

4. Goldberg, David E.: *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley Publ. Co., 1989.

5. Grefenstette, John J., ed.: *Proceedings of the First International Conference on Genetic Algorithms and Their Applications.* Lawrence Erlbaum Assoc., 1985.

6. Grefenstette, John J., ed.: *Proceedings of the Second International Conference on Genetic Algorithms.* Lawrence Erlbaum Assoc., 1987.

7. Schaffer, J. David, ed.: *Proceedings of the Third International Conference on Genetic Algorithms.* M. Kaufmann Publ., 1989.

8. Krishnakumar, K.; and Goldberg, David E.: Control System Optimization Using Genetic Algorithms. *J. Guid., Control, & Dyn.*, vol. 15, no. 3, May–June 1992, pp. 735-740.

9. Fletcher, R.: *Practical Methods of Optimization. Volume 2—Constrained Optimization.* John Wiley & Sons, Inc., 1981.

10. Vilkov, A. V.; Zidkov, N. P.; and Shchedrin, B. M.: A Method of Finding the Global Minimum of Function of One Variable. *J. Comput. Math. & Math. Phys.*, vol. 15, no. 4, 1975, pp. 1040–1047.

11. Levy, A. V.; and Montalvo, A.: The Tunneling Algorithm for the Global Minimization of Functions. *SIAM J. Sci. & Stat. Comput.*, vol. 6, no. 1, Jan. 1985, pp. 15–29.

12. Ge, Ren Pu: The Theory of Filled Function Method for Finding Global Minimizers on Nonlinearly Constrained Minimization Problems. *J. Comput. Math.*, vol. 5, no. 1, 1987, pp. 1–9.

13. Deshpande, Samir M.; Kumar, Renjith R.; Seywald, Hans; and Siemers, Paul M., III: Air Data System Optimization Using a Generic Algorithm. *A Collection of Technical Papers, Part 2—AIAA Guidance, Navigation and Control Conference*, Aug. 1992, pp. 762–770.

14. Calise, Anthony J.; and Moerder, Daniel D.: *Singular Perturbation Techniques for Real Time Aircraft Trajectory Optimization and Control.* NASA CR-3597, 1982.

15. Shaughnessy, John D.; Pinckney, S. Zane; McMinn, John D.; Cruz, Christopher I.; and Kelley, Marie-Louise: *Hypersonic Vehicle Simulation Model: Winged-Cone Configuration.* NASA TM-102610, 1990.

Table 1. Effect of Mutation Probability
on KT Performance in Example 1

P_{mutate}	Successes (median generations)		
	$\text{err}_{\mathbf{x}^*} \leq 0.10$	$\text{err}_{\mathbf{x}^*} \leq 0.05$	$\text{err}_{\mathbf{x}^*} \leq 0.01$
0.01	70(12)	49(18)	38(28)
.03	91(16)	75(23)	72(41)
.05	98(14)	83(31)	64(39)
.07	98(15)	78(25)	64(42)

Table 2. Penalty Function GA Performance for $P_{\text{mutate}} = 0.01$ in Example 1

Case	B	b	ϵ	Successes (median generations)		
				$\text{err}_{\mathbf{x}^*} \leq 0.10$	$\text{err}_{\mathbf{x}^*} \leq 0.05$	$\text{err}_{\mathbf{x}^*} \leq 0.01$
1	10^{20}	10^4	0.05	2(5)	1(20)	1(20)
2	10^{20}	10^4	.10	3(29)	1(29)	1(29)
3	10^{20}	10^4	.20	3(9)	2(64)	2(64)
4	10^{20}	10^2	.05	11(8)	8(19)	6(38)
5	10^{20}	10^2	.10	15(8)	13(25)	11(37)
6	10^{20}	10^2	.20	12(18)	9(41)	9(41)
7	10^{20}	10^1	.05	9(10)	8(30)	7(62)
8	10^{20}	10^1	.10	10(12)	7(7)	6(42)
9	10^{20}	10^1	.20	12(14)	7(59)	6(66)
10	10^6	10^4	.05	11(9)	6(34)	6(34)
11	10^6	10^4	.10	6(4)	3(9)	3(11)
12	10^6	10^4	.20	9(5)	5(9)	5(57)
13	10^6	10^2	.05	10(12)	7(19)	7(43)
14	10^6	10^2	.10	10(8)	6(24)	6(28)
15	10^6	10^2	.20	13(10)	7(27)	6(44)
16	10^6	10^1	.05	10(24)	8(28)	7(29)
17	10^6	10^1	.10	6(8)	6(22)	5(34)
18	10^6	10^1	.20	14(9)	9(33)	9(45)
19	10^4	10^2	.05	3(2)	1(60)	1(60)
20	10^4	10^2	.10	6(16)	4(43)	4(46)
21	10^4	10^2	.20	3(2)	1(23)	1(23)
22	10^4	10^1	.05	9(5)	6(29)	6(34)
23	10^4	10^1	.10	10(6)	5(25)	5(25)
24	10^4	10^1	.20	17(11)	10(22)	10(30)

Table 3. Penalty Function GA Performance for $P_{\text{mutate}} = 0.03$ in Example 1

Case	B	b	ϵ	Successes (median generations)		
				$\text{err}_{\mathbf{x}^*} \leq 0.10$	$\text{err}_{\mathbf{x}^*} \leq 0.05$	$\text{err}_{\mathbf{x}^*} \leq 0.01$
1	10^{20}	10^4	0.05	15(51)	8(52)	8(52)
2	10^{20}	10^4	.10	15(40)	10(62)	10(62)
3	10^{20}	10^4	.20	18(52)	13(73)	13(73)
4	10^{20}	10^2	.05	21(38)	14(40)	13(59)
5	10^{20}	10^2	.10	27(39)	18(46)	17(56)
6	10^{20}	10^2	.20	34(28)	23(46)	22(49)
7	10^{20}	10^1	.05	29(34)	23(62)	23(62)
8	10^{20}	10^1	.10	29(51)	20(55)	19(58)
9	10^{20}	10^1	.20	35(35)	27(57)	27(57)
10	10^6	10^4	.05	26(39)	20(54)	19(56)
11	10^6	10^4	.10	27(31)	22(53)	21(56)
12	10^6	10^4	.20	28(28)	19(56)	18(62)
13	10^6	10^2	.05	27(59)	19(66)	19(66)
14	10^6	10^2	.10	39(36)	28(56)	28(56)
15	10^6	10^2	.20	37(49)	27(62)	27(62)
16	10^6	10^1	.05	29(37)	22(52)	22(54)
17	10^6	10^1	.10	34(54)	27(63)	27(67)
18	10^6	10^1	.20	41(37)	28(64)	26(64)
19	10^4	10^2	.05	26(51)	18(68)	18(68)
20	10^4	10^2	.10	28(44)	20(62)	20(62)
21	10^4	10^2	.20	33(39)	23(60)	23(60)
22	10^4	10^1	.05	33(56)	28(66)	28(66)
23	10^4	10^1	.10	37(38)	26(66)	26(68)
24	10^4	10^1	.20	48(60)	30(74)	28(76)

Table 4. Penalty Function GA Performance for $P_{\text{mutate}} = 0.05$ in Example 1

Case	B	b	ϵ	Successes (median generations)		
				$\text{err}_{\mathbf{x}^*} \leq 0.10$	$\text{err}_{\mathbf{x}^*} \leq 0.05$	$\text{err}_{\mathbf{x}^*} \leq 0.01$
1	10^{20}	10^4	0.05	28(52)	15(43)	13(56)
2	10^{20}	10^4	.10	34(51)	23(59)	21(71)
3	10^{20}	10^4	.20	39(48)	24(64)	23(59)
4	10^{20}	10^2	.05	36(39)	26(56)	23(56)
5	10^{20}	10^2	.10	47(60)	33(60)	29(60)
6	10^{20}	10^2	.20	53(62)	34(76)	30(76)
7	10^{20}	10^1	.05	52(46)	39(48)	38(54)
8	10^{20}	10^1	.10	55(44)	37(69)	32(72)
9	10^{20}	10^1	.20	66(50)	49(71)	43(67)
10	10^6	10^4	.05	42(57)	24(59)	21(58)
11	10^6	10^4	.10	33(52)	24(65)	21(67)
12	10^6	10^4	.20	33(40)	27(50)	24(54)
13	10^6	10^2	.05	41(48)	27(62)	25(48)
14	10^6	10^2	.10	56(56)	39(72)	36(72)
15	10^6	10^2	.20	47(59)	35(75)	30(68)
16	10^6	10^1	.05	38(58)	26(70)	24(66)
17	10^6	10^1	.10	50(46)	34(64)	26(64)
18	10^6	10^1	.20	66(44)	40(56)	37(63)
19	10^4	10^2	.05	48(45)	38(56)	35(59)
20	10^4	10^2	.10	40(44)	27(64)	25(64)
21	10^4	10^2	.20	49(45)	37(59)	35(59)
22	10^4	10^1	.05	45(51)	29(51)	27(54)
23	10^4	10^1	.10	62(47)	42(50)	38(55)
24	10^4	10^1	.20	60(56)	38(70)	34(70)

Table 5. Penalty Function GA Performance for $P_{\text{mutate}} = 0.07$ in Example 1

Case	B	b	ϵ	Successes (median generations)		
				$\text{err}_{\mathbf{x}^*} \leq 0.10$	$\text{err}_{\mathbf{x}^*} \leq 0.05$	$\text{err}_{\mathbf{x}^*} \leq 0.01$
1	10^{20}	10^4	0.05	35(52)	18(53)	15(60)
2	10^{20}	10^4	.10	39(58)	22(71)	20(76)
3	10^{20}	10^4	.20	38(43)	20(52)	20(52)
4	10^{20}	10^2	.05	45(50)	29(65)	28(65)
5	10^{20}	10^2	.10	40(46)	27(69)	27(69)
6	10^{20}	10^2	.20	45(51)	26(64)	26(66)
7	10^{20}	10^1	.05	47(49)	33(46)	29(50)
8	10^{20}	10^1	.10	62(54)	37(66)	32(73)
9	10^{20}	10^1	.20	57(38)	38(43)	38(48)
10	10^6	10^4	.05	32(36)	17(62)	16(63)
11	10^6	10^4	.10	41(62)	25(75)	25(75)
12	10^6	10^4	.20	40(58)	22(66)	22(66)
13	10^6	10^2	.05	56(48)	28(46)	27(49)
14	10^6	10^2	.10	44(64)	23(74)	21(74)
15	10^6	10^2	.20	49(50)	34(66)	33(74)
16	10^6	10^1	.05	48(48)	35(57)	27(63)
17	10^6	10^1	.10	55(38)	24(36)	22(40)
18	10^6	10^1	.20	61(46)	39(59)	33(61)
19	10^4	10^2	.05	39(57)	18(67)	16(69)
20	10^4	10^2	.10	34(55)	20(57)	18(62)
21	10^4	10^2	.20	48(52)	27(62)	25(65)
22	10^4	10^1	.05	44(38)	31(53)	29(61)
23	10^4	10^1	.10	57(40)	38(58)	36(58)
24	10^4	10^1	.20	67(37)	44(60)	42(62)

Table 6. Summary of Penalty Function GA Performance for Example 2

Case	B	b	ϵ	KT performance distribution					N_{succ}
				min	Q_1	Q_2	Q_3	max	
4	10^{20}	10^2	0.05	3.782	22.810	34.852	44.911	107.101	27
5	10^{20}	10^2	.10	8.205	25.862	35.294	53.612	101.633	29
6	10^{20}	10^2	.20	4.402	20.192	34.863	47.886	93.153	36
7	10^{20}	10^1	.05	4.949	13.055	20.162	28.005	56.846	45
8	10^{20}	10^1	.10	1.807	12.340	18.999	28.049	59.982	48
9	10^{20}	10^1	.20	2.919	11.700	21.318	26.641	46.687	34
13	10^6	10^2	.05	3.782	22.810	34.320	43.926	96.346	28
14	10^6	10^2	.10	8.205	25.862	35.294	53.612	101.633	28
15	10^6	10^2	.20	4.402	20.889	35.167	48.078	93.153	34
16	10^6	10^1	.05	4.949	13.496	20.162	28.005	56.846	43
17	10^6	10^1	.10	1.807	12.340	18.999	28.049	59.982	49
18	10^6	10^1	.20	2.919	11.700	21.318	26.895	46.687	35
19	10^4	10^2	.05	3.782	23.666	34.756	44.386	91.330	38
20	10^4	10^2	.10	4.198	29.236	37.646	51.673	97.060	33
21	10^4	10^2	.20	2.801	24.732	37.204	46.266	95.789	36
22	10^4	10^1	.05	4.751	13.671	20.700	29.621	57.700	43
23	10^4	10^1	.10	1.854	12.705	20.948	27.805	59.982	53
24	10^4	10^1	.20	3.387	11.652	20.313	27.335	43.526	38

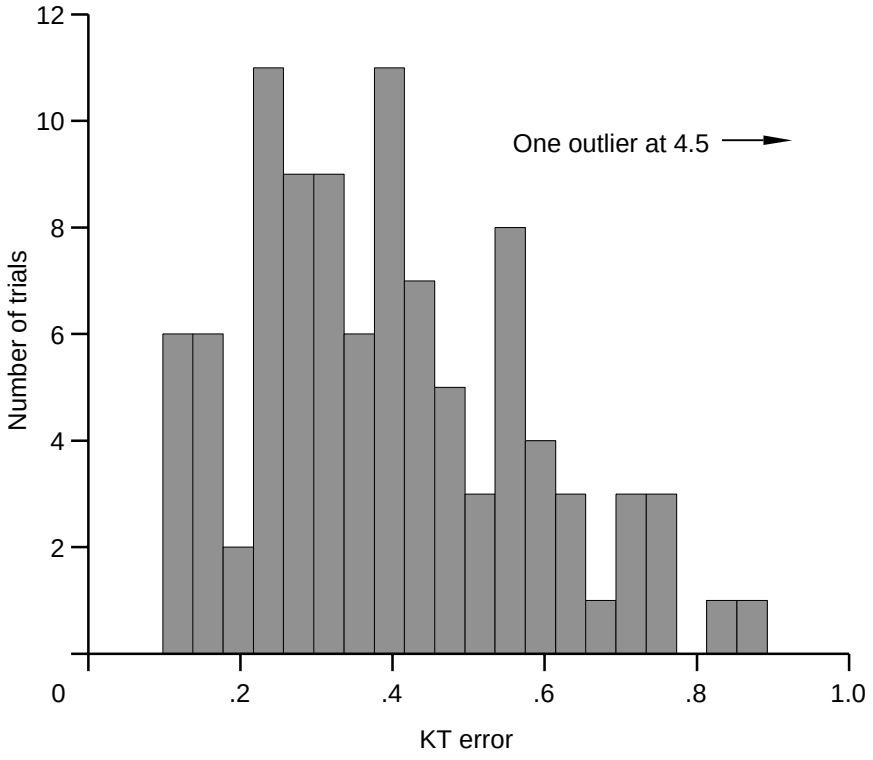


Figure 1. Distribution of KT error for first aerospace plane Monte Carlo experiment.

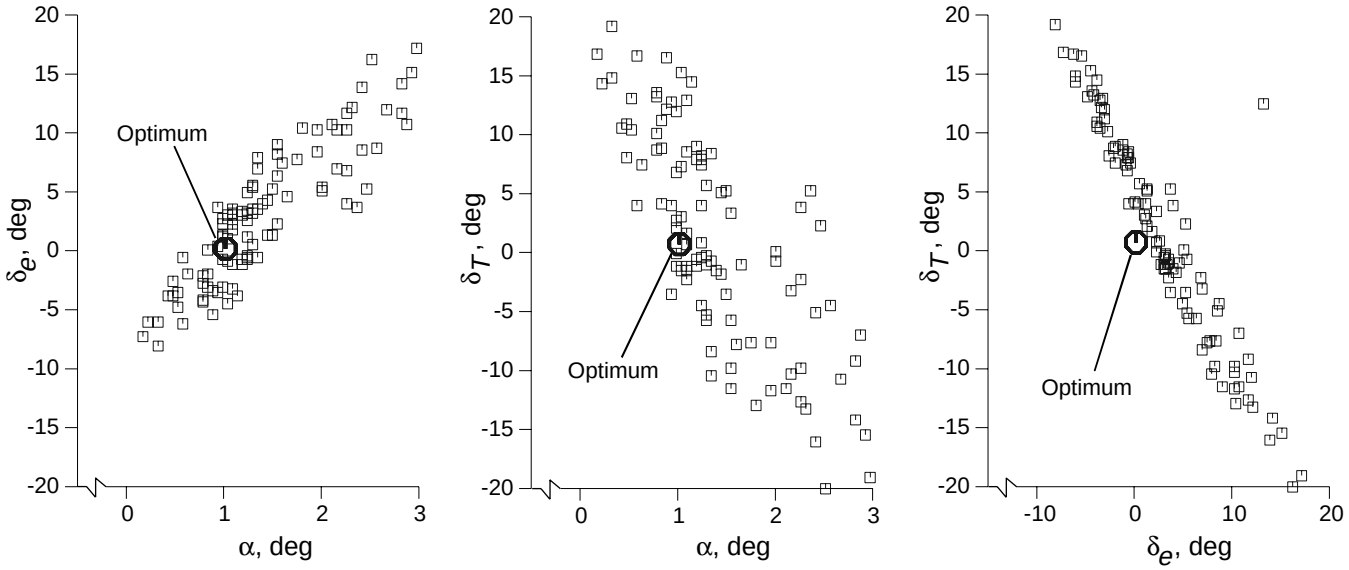


Figure 2. Distribution of α, δ_e, and δ_T for first aerospace plane Monte Carlo experiment.

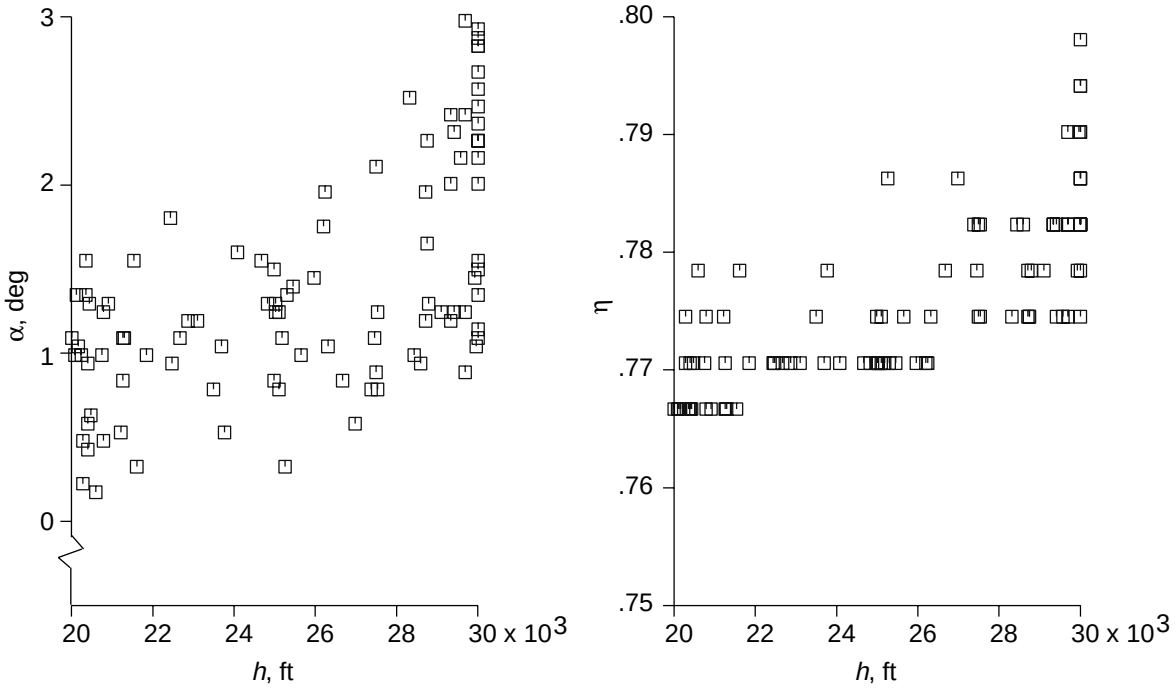


Figure 3. Distribution of α, h, and η for first aerospace plane Monte Carlo experiment.

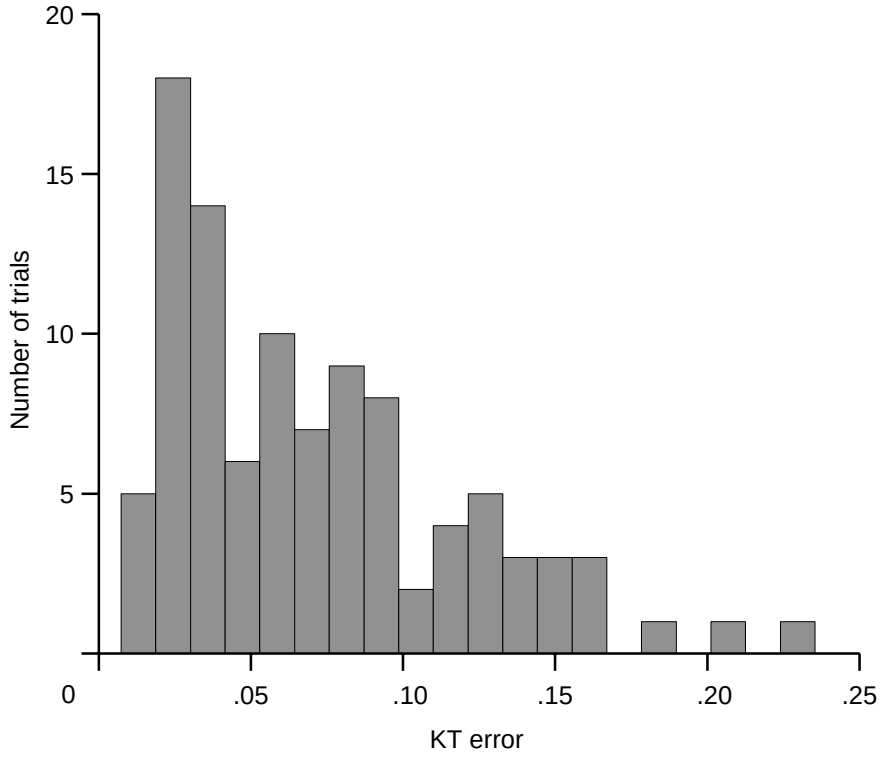


Figure 4. Distribution of KT error for refined aerospace plane Monte Carlo experiment.

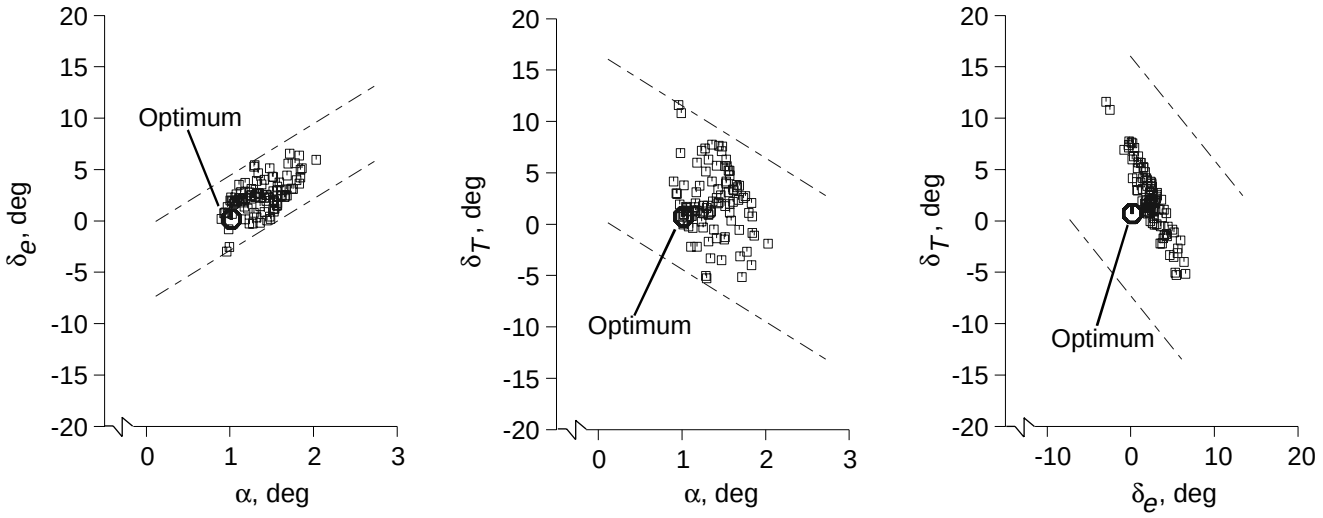


Figure 5. Distribution of α, δ_e, and δ_T for refined aerospace plane Monte Carlo experiment.

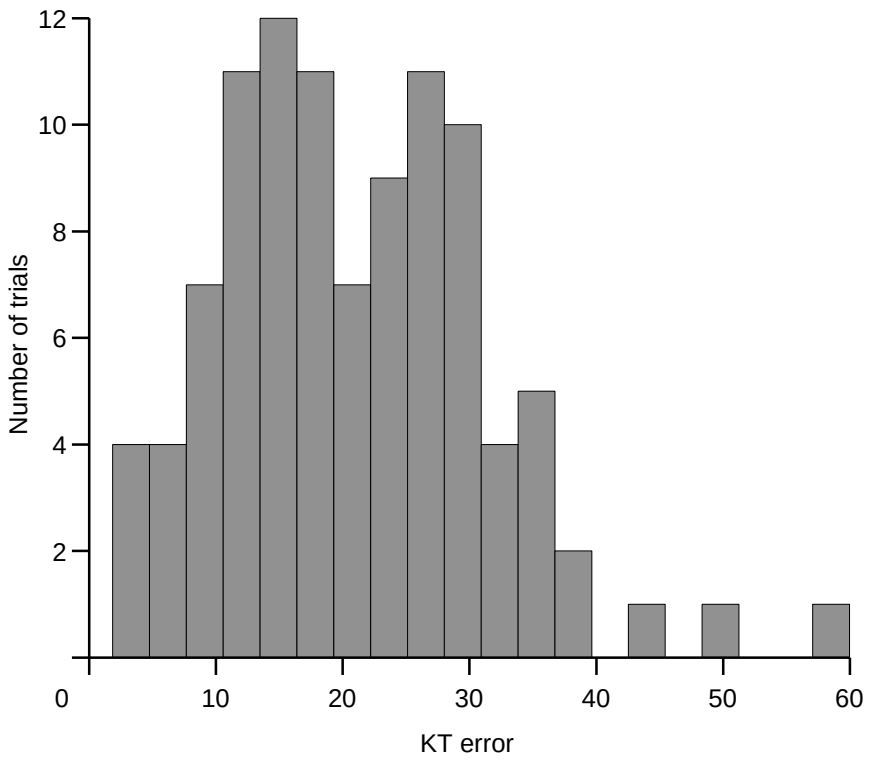


Figure 6. Distribution of KT error for penalty formulation aerospace plane Monte Carlo experiment.

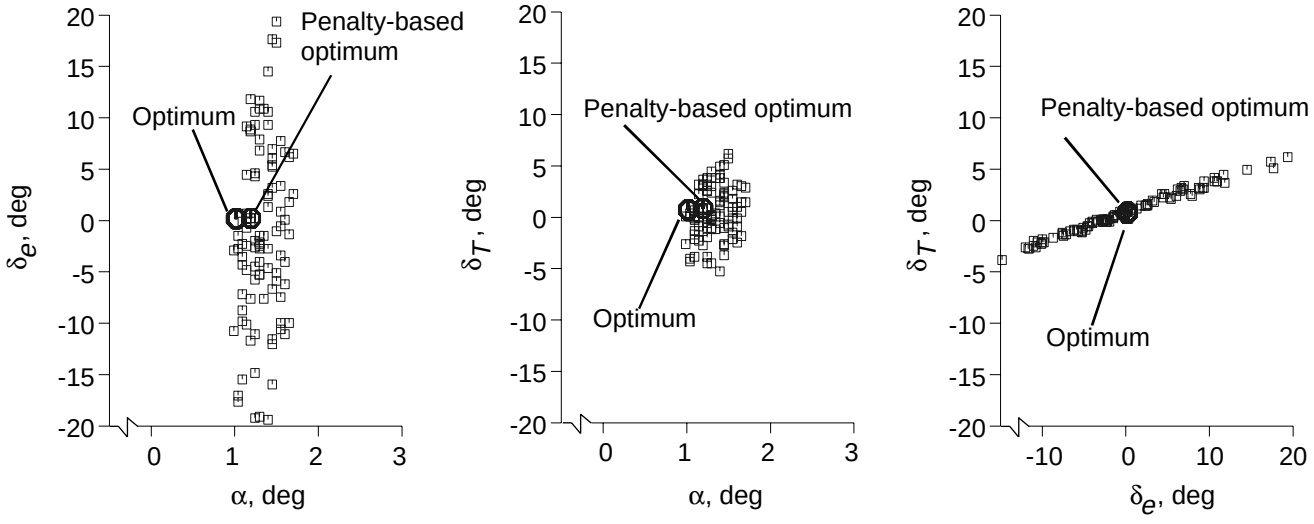


Figure 7. Distribution of α, δ_e, and δ_T for penalty function aerospace plane Monte Carlo experiment.

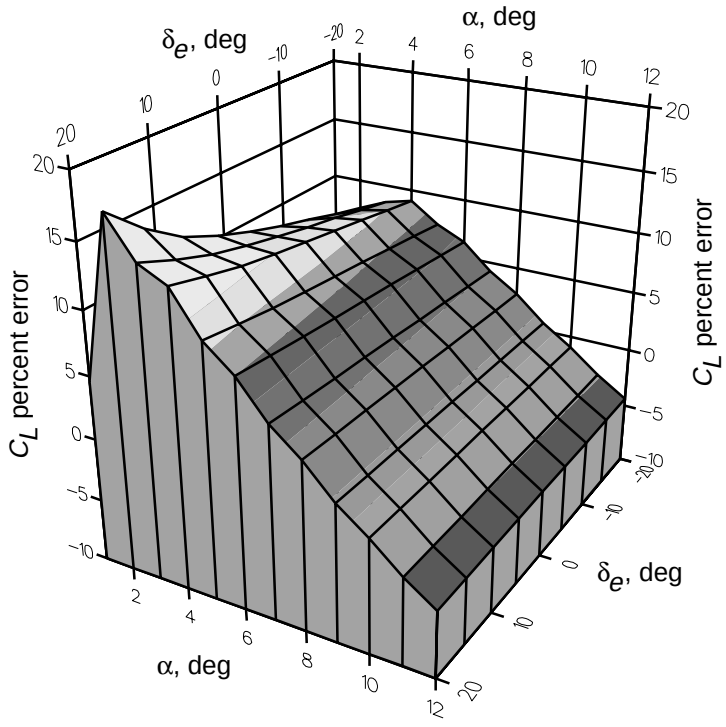


Figure 8. C_L percent error in aerospace plane model.

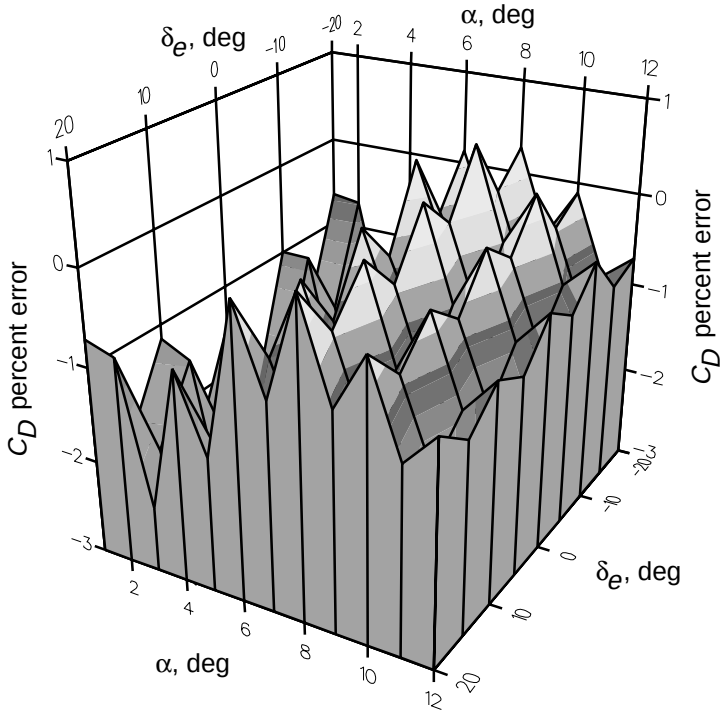


Figure 9. C_D percent error in aerospace plane model.

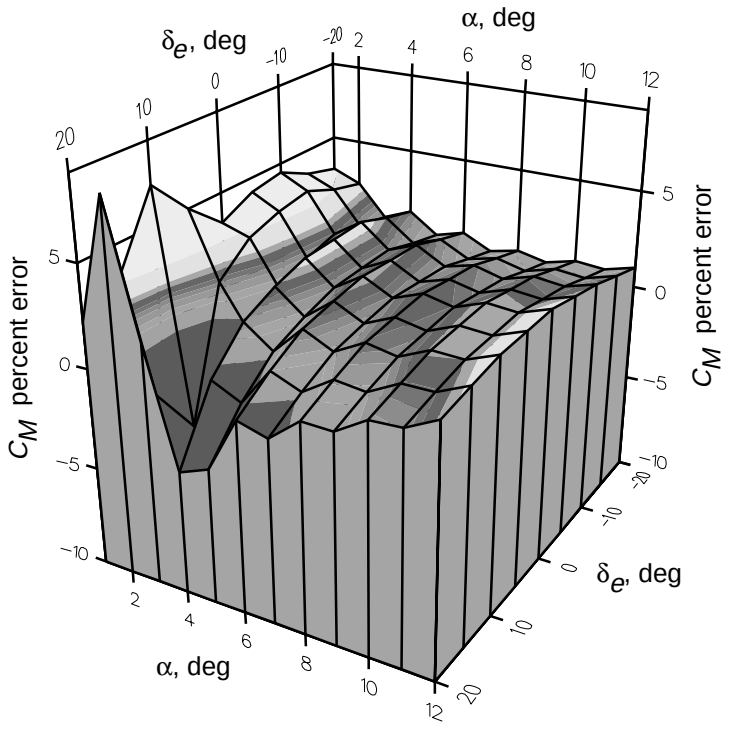


Figure 10. C_M percent error in aerospace plane model.

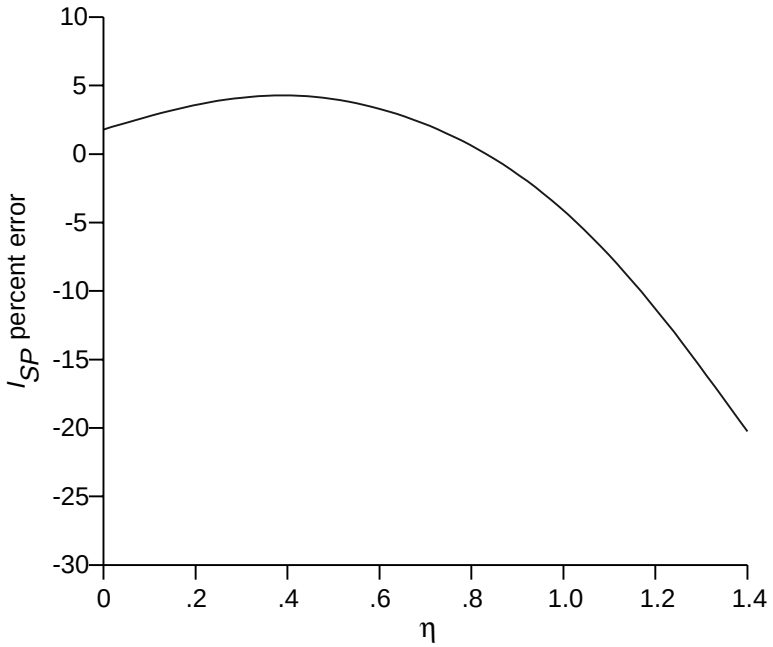


Figure 11. I_{SP} percent error in aerospace plane model.

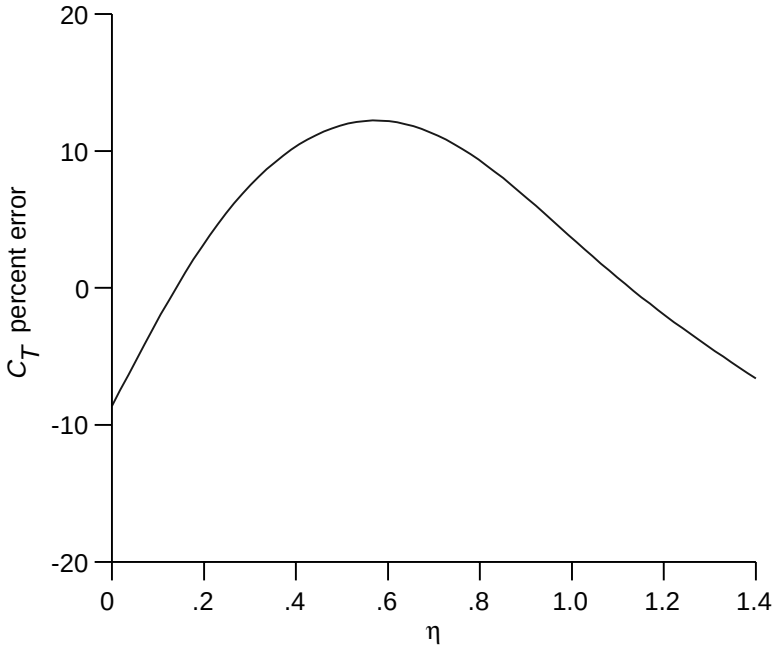


Figure 12. C_T percent error in aerospace plane model.

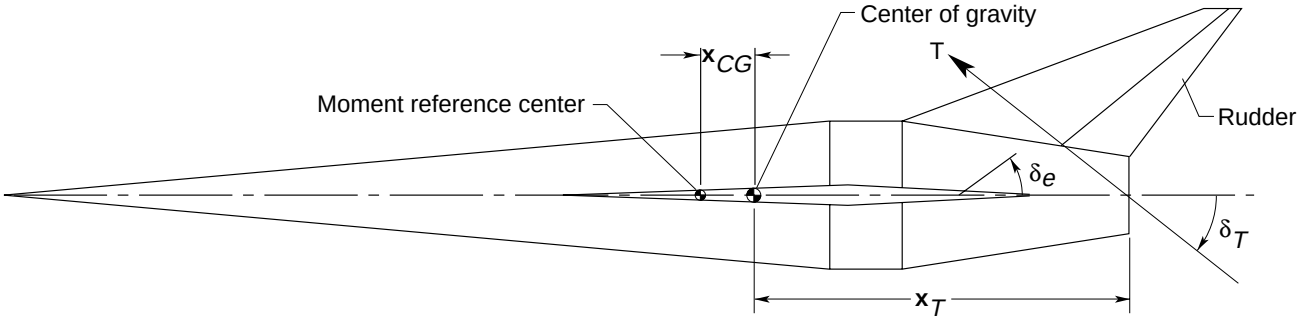


Figure 13. Aerospace plane geometry. Subscript CG indicates center of gravity.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	November 1994	Technical Paper

4. TITLE AND SUBTITLE
Constrained Minimization of Smooth Functions Using a Genetic Algorithm

5. FUNDING NUMBERS
WU 506-59-61-01

6. AUTHOR(S)
Daniel D. Moerder and Bandu N. Pamadi

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
NASA Langley Research Center
Hampton, VA 23681-0001

8. PERFORMING ORGANIZATION REPORT NUMBER
L-17221

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
National Aeronautics and Space Administration
Washington, DC 20546-0001

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
NASA TP-3329

11. SUPPLEMENTARY NOTES
Moerder: Langley Research Center, Hampton, VA; Pamadi: ViGYAN, Inc. Hampton, VA.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Unclassified–Unlimited

Subject Category 18

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*
The use of genetic algorithms for minimization of differentiable functions that are subject to differentiable constraints is considered. A technique is demonstrated for converting the solution of the necessary conditions for a constrained minimum into an unconstrained function minimization. This technique is extended as a global constrained optimization algorithm. The theory is applied to calculating minimum-fuel ascent control settings for an energy state model of an aerospace plane.

14. SUBJECT TERMS
Genetic algorithms; Constrained optimization; Minimization; Energy-state methods

15. NUMBER OF PAGES
29

16. PRICE CODE
A03

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	