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

BULK RATE
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