MAXIMIZATION BY IMPROVED QUADRATIC HILL-CLIMBING AND OTHER METHODS

Stephen M. Goldfeld, Richard E. Quandt,
Hale F. Trotter

Econometric Research Program
Research Memorandum No. 95
April 1968

The research described in this paper was supported by the National Science Foundation Grant NSF GS 1840 and by Grant NSF GS 823.

Princeton University
Econometric Research Program
207 Dickinson Hall
Princeton, New Jersey

Maximization by Improved Quadratic Hill-Climbing and Other Methods

by

Stephen M. Goldfeld, Richard E. Quandt, Hale F. Trotter

1. Introduction

The frequent occurrence of many-dimensional maximization problems in the physical and social sciences has prompted the development of several efficient algorithms for numerical maximization of functions of several variables. Some of the most promising of these are the various types of conjugate gradient methods developed by Davidon [2], Powell [7], and others. The authors developed an algorithm, presented in [4], which is based on maximizing at each iteration a quadratic approximation to the function on a sphere the radius of which is determined by the goodness of the previous approximation.

Whenever a new algorithm is presented, it is customary to compare its performance to some leading or well-known competitors. Apart from the intrinsic difficulty of performing valid comparisons of this kind at all, most attempts at assessing the relative performance of various algorithms suffer from the disadvantage that they contrast one method with only one other.[1]

The purpose of this paper is twofold. First we shall introduce a generalization of our algorithm, based on an observation contained in [4].

---

1. For a notable exception, see [6].

Secondly, we shall compare the improved algorithm with the earlier version as well as with the Davidon and the Powell algorithms.[2] In general, we shall find that the predominant effect of the improvement is to enhance the reliability of the algorithm.

2. Quadratic Hill-Climbing

Consider the problem of maximizing the function $f(x)$, $x = (x_1, \ldots, x_n)$. The procedure employed by iterative schemes can usually be expressed as

$$x^{p+1} = x^p + D^p \tag{2-1}$$

where x^p and x^{p+1} represent the values of the variable vector at the pth and (p+1)th iterations, and D^p is a vector. Denoting by F_{x^p} the gradient evaluated at x^p and by S_{x^p} the matrix of second partial derivatives, Newton's method, for example, employs $D^p = -S_{x^p}^{-1} F_{x^p}$.

Let $\|x\|$ be the length of the vector x, defined to be $(x'x)^{\frac{1}{2}}$, and consider the quadratic function $Q(x)$. In [4] we proved the following theorems about a quadratic function $Q(x)$:

<u>Theorem 1</u>. Let x^p be any point and α any number such that $S_{x^p} - \alpha I$ is negative definite, and define

$$b_\alpha = x^p - (S_{x^p} - \alpha I)^{-1} F_{x^p} \tag{2-2}$$

$$r_\alpha = \| b_\alpha - x^p \| \tag{2-3}$$

then $Q(b_\alpha) \geqq Q(x)$ for all x such that $\|x - x^p\| = r_\alpha$.

---

2. For reasons of economy we have not included among the competing methods pattern search and random search algorithms. See [8] and [9].

Theorem 2. If $F_{x^p} \neq 0$ then the r_α defined by (2-2) and (2-3) is a strictly decreasing function of α on the interval (λ_1, ∞) where λ_1 is the maximum eigenvalue of S_{x^p}.

Theorem 3. Let α, b_α and r_α be as in Theorem 1, let B_α be the region consisting of all x such that $\| x - x^p \| \leqq r_\alpha$, and assume $F_{x^p} \neq 0$. Then the maximum value of $Q(x)$ on B_α is attained at b_α if $\alpha \geqq 0$ and is attained at b_0 if $\alpha < 0$. In this latter case b_0 is interior to the region B_α.

Theorem 4. Let u_1 be a unit eigenvector associated with λ_1. If $F_{x^p} = 0$, then the maximum value of $Q(x)$ on the region B_α consisting of all x with $\| x - x^p \| \leqq r_\alpha$ occurs at $x^p \pm r_\alpha u_1$ if λ_1 is positive and at x^p otherwise.

The quadratic hill-climbing algorithm is based on maximizing at each iteration, say the p^{th}, the quadratic approximation to the function $f(x)$

$$f(x) \approx f(x^p) + (x - x^p)' F_{x^p} + \tfrac{1}{2}(x - x^p)' S_{x^p} (x - x^p)$$

on a spherical region centered at x^p. It is desirable to take this region as large as possible subject to the requirement that in the region the quadratic approximation remain reasonably good. If F_{x^p} is significantly different from zero, an iteration is accomplished by setting

$$\alpha = \lambda_1 + R \| F_{x^p} \|$$

where R is a positive number and choosing x^{p+1} by

$$x^{p+1} = x^p - (S_{x^p} - \alpha I)^{-1} F_{x^p}$$

or

$$x^{p+1} = x^p - S_{x^p}^{-1} F_{x^p}$$

depending on whether α is positive or not.[3] If F_{x^p} is sufficiently close to zero so that the length of the proposed step is less than some preset tolerance, and S_{x^p} is negative definite, the process terminates and x^p is accepted as the location of the maximum. If S_{x^p} is not negative definite, Theorem 4 is applied and a step is taken in the direction of the eigenvector corresponding to the largest root of S_{x^p}. The method described in this section will be referred to below as Quadratic Hill Climbing-1.

3. Improved Quadratic Hill-Climbing

Let A be an arbitrary positive definite matrix and define a metric $\| \ \|_A$ by $\|x\|_A = (x'Ax)^{\frac{1}{2}}$. It was observed in [4] that the theorems of the previous section continue to hold if $\| \ \|$ is replaced by $\| \ \|_A$ and $S - \alpha I$ is replaced by $S - \alpha A$. The region B_α on which the quadratic approximation is maximized becomes ellipsoidal rather than spherical as a result of this replacement. It is the purpose of the present section to exploit this generalization. The method will be referred to below as Quadratic Hill Climbing-2.

The overall size of the ellipsoid on which the approximation is maximized is still governed by a parameter whose value can be controlled by the same method as before. The new feature is that the shape and orientation of the ellipsoid are now also subject to change. The modified quadratic

---

3. It is easy to show that the radius of the region B_α, $\|(S_{x^p} - \alpha I)^{-1} F_{x^p}\|$, is bounded by R^{-1}. Therefore we automatically modify R at each iteration, increasing its value when the quadratic approximation is bad and decreasing it otherwise.

hill-climbing algorithm is based on the heuristically plausible assumption that the most useful direction of search at any point is close to the direction of the immediately preceding step. (If one is on a "ridge" of the function, one may expect further steps to follow along the ridge, rather than go transversely.)

Suppose we have completed p steps and that δ is the column vector expressing the last step (so $\delta = x^p - x^{p-1}$) in the coordinate system used for making the last step. For the moment, suppose that δ happens to be parallel to the first basis vector of the coordinate system (i.e., all its coordinates except the first are zero). We wish to use a new coordinate system for computing the next step which will "emphasize" the component in the direction of δ. Consider the coordinate change under which a vector with coordinates $(\xi_1, \xi_2, \ldots, \xi_n)$ in the old system is given coordinates $(\beta\xi_1, \xi_2, \ldots, \xi_n)$ in the new system, for some constant β. If $0 < \beta < 1$, a vector with (say) equal components in the new system will have a larger first component in the old system, and the desired emphasis will be achieved. In general, of course, δ will not be parallel to a coordinate vector. While one could first rotate axes to obtain a coordinate vector in the direction of δ and then proceed as above, the same effect is achieved with less computation as follows.

Working in the old coordinate system, any vector x may be expressed as the sum of

$$x_1 = \frac{\delta' x}{\delta' \delta} \delta ,$$

which is the projection of x along δ and

$$x_2 = x - x_1 ,$$

which is orthogonal to δ. The appropriate new coordinates for x may then be written as

$$\beta x_1 + x_2 = \beta \frac{\delta' x}{\delta'\delta}\delta + x - \frac{\delta' x}{\delta'\delta}\delta$$

$$= x + \frac{(\beta - 1)}{\delta'\delta}\delta(\delta' x)$$

$$= \left[I + \frac{\beta - 1}{\delta'\delta}\delta\delta'\right]x = Bx \qquad (3\text{-}1)$$

where the matrix B is defined by the last equation. (Here δ and x are both treated as 1-column matrices, with δ' denoting the transpose of δ. Thus $\delta'\delta$ is a scalar and $\delta\delta'$ an n-by-n matrix.)

At each iteration a new B-transformation takes place. Let them be denoted by B_0, B_1, ... and for any vector let x_k stand for its coordinates in the system used to make the k^{th} step. Then

$$x_k = B_{k-1}B_{k-2} \cdots B_1 B_0 x_0. \qquad (3\text{-}2)$$

(We take $B_0 = I$, since there is no previous step and δ is undefined.) For the $(k+1)^{st}$ iteration we maximize over a "sphere"

$$x'_{k+1}x_{k+1} = (B_k x_k)'(B_k x_k) = \text{constant} \qquad (3\text{-}3)$$

which is in general an ellipsoid in the original coordinate system.[4] The new step computed at this point may be expressed easily in terms of the

---

4. The choice of β at each iteration is based on a comparison of the actual improvement in the function with the apparent improvement due to the quadratic approximation. Defining z = (actual improvement)/(apparent improvement) and $c = (z - 1)^2 - \varepsilon$, the actual modification of β is given by

$\beta = .9$ if $z \leqq 0$ or $z \geqq 2$, otherwise

$\beta = \beta + (.9 - \beta)c$ if $c \geqq 0$ and

$\beta = \beta - (.1 - \beta)c$ if $c < 0$.

In practice we employed for ε the value .5.

original coordinate system prevailing at the beginning of the process. Setting $B_{k-1} \cdots B_1 B_0 = B$, it is easy to verify that the gradient evaluated at x^k and expressed in the original coordinates is $(B^{-1})' F_{x^k}$ and the matrix of second partial derivatives is $(B^{-1})' S_{x^k} B^{-1}$. The step in the new coordinate system is clearly $-\left((B^{-1})' S_{x^k} B^{-1} - \alpha I\right)^{-1} (B^{-1})' F_{x^k}$ and premultiplication by B^{-1} yields the step in the old coordinates and it equals $-(S_{x^k} - \alpha B'B)^{-1} F_{x^k}$. Thus $B'B$ is the positive definite matrix A mentioned at the beginning of this section.[5]

4. Comparison of Algorithms: Cum Grano Salis

The efficiency of an algorithm is a composite of at least three types of factors: (1) its reliability which might be measured by the probability that convergence to a true maximum ultimately does take place; (2) its cost in terms of human effort required to make the algorithm "work" on

---

5. It is obvious that the present modification is computationally more expensive per iteration than the simpler variant of Section 2. Nevertheless, the increase in the amount of computation is not quite as great as may appear at first. Although the matrix B^{-1} has to be computed, this inverse can be obtained directly. In general, given a matrix

$$A = \begin{bmatrix} 1 + k\delta_1^2 & k\delta_1\delta_2 & \cdots & k\delta_1\delta_n \\ k\delta_1\delta_2 & 1 + k\delta_2^2 & \cdots & k\delta_2\delta_n \\ \cdot & \cdot & \cdot & \cdot \\ k\delta_1\delta_n & k\delta_2\delta_n & \cdots & 1 + k\delta_n^2 \end{bmatrix}$$

we have

$$A^{-1} = \frac{1}{1 + k \sum_{i=1}^{n} \delta_i^2} \begin{bmatrix} 1 + k \sum_{i \neq 1} \delta_i^2 & -k\delta_1\delta_2 & \cdots & -k\delta_1\delta_n \\ -k\delta_1\delta_2 & 1 + k \sum_{i \neq 2} \delta_i^2 & \cdots & -k\delta_2\delta_n \\ \cdot & \cdot & \cdot & \cdot \\ -k\delta_1\delta_n & -k\delta_2\delta_n & \cdots & 1 + k \sum_{i \neq n} \delta_i^2 \end{bmatrix}$$

a given problem; and (3) its cost in terms of the computer time required to achieve a solution. In the first category we place the possibility that some algorithm may mistake a saddlepoint for a true maximum. From this point of view the quadratic hill-climbing algorithms may be superior to the others which are being compared with it. In the second category we place considerations of the effort that may be required to calculate derivatives of the function to be maximized. From this point of view Davidon's method (requiring only first derivatives) is superior to the quadratic hill-climbing methods and Powell's method (requiring no derivatives) is superior to Davidon's. The relative effort required for these several methods may be equalized by evaluating derivatives numerically, thus dispensing with the effort of obtaining and programming for the computer the formulas for first (and second) partial derivatives.[6]

The comparison of the algorithms from the third point of view, i.e., in terms of the computing time required for satisfactory convergence to be achieved, is a difficult matter. Computing requirements will generally depend not only upon the intrinsic characteristics of the algorithm but also upon the efficiency with which the necessary computer programs have been prepared. Although it would be desirable to program the various algorithms with "equal efficiency," this term is not well defined and must remain no more than approximate in meaning. Thus, even though the time required for computation is the most relevant measure of efficiency, the reliability of the measure is not complete.

---

6. The use of numerical instead of analytic derivatives may, however, increase the time required for computation, particularly by causing the algorithms to perform extra iterations in the neighborhood of the maximum where numerically evaluated derivatives are sometimes quite inaccurate.

As a supplementary measure of efficiency one may employ the number of times the function is evaluated in the course of computations. Since much of the computational work is, indeed, in evaluating the function, this will shed new light on the efficiency question. In particular, it may be that certain methods perform faster only because we are dealing with simple functions. That is, the rank ordering of methods with respect to efficiency will not in general be invariant to the nature of the functions. But it should be emphasized that by itself the number of function evaluations is not a completely suitable measure. If first and second derivatives are evaluated analytically, it is not clear how the work performed in these evaluations should be counted relative to the work involved in function evaluations. Only some measure of time can provide a common denominator. In the present paper the derivatives are evaluated numerically so that this problem did not arise.[7]

For these reasons, we shall employ both the time required and the number of function evaluations involved in achieving given accuracy. A final measure of performance is, of course, the relative frequency of cases in which an algorithm eventually converges to the true extremum, i.e., does not terminate at some other point.

The functions employed are partly well-known test functions that had already been employed by other investigators and partly new functions we devised for test purposes. They are as follows:

1. Rosenbrock's function

$$z = 100(y - x^2)^2 + (1 - x)^2$$

---

7. The relative performance of algorithms in terms of function evaluations depends upon the slightly extraneous circumstance of whether the numerically evaluated derivatives are computed from function values placed symmetrically about the point at which the derivatives are required or not.

which has a single minimum at (1,1) and resembles a U-shaped valley with steep walls.[8]

2. The function

$$z = (x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4$$

with a single minimum at (0,0,0,0).[9]

3. The function

$$z = \exp(\exp(\exp(\exp(\exp(\exp(\exp(-x^2 - 3y^2)/10)/10)/10)/10)))$$

with a single maximum at (0,0).

4. Beale's function, given by

$$z = (1.5 - x(1 - y))^2 + (2.25 - x(1 - y^2))^2 + (2.625 - x(1 - y^3))^2$$

which is a narrow curving valley with a minimum at (3,.5).[10]

5. A three dimensional Rosenbrock function

$$z = 100(x_3 - x_1^2)^2 + 100(x_2 - x_1^2)^2 + (1 - x_1)^2$$

with a minimum at (1,1,1).

6. The function

$$z = \frac{1}{Q(1 + x^k)} \exp\left\{-\frac{(y - x^2)^2}{(Q(1 + x^k))^2}\right\} \qquad \begin{array}{l} Q > 0 \\ k > 0 \text{ and even} \end{array}$$

which is a parabolic Gaussian ridge, achieving its maximum at (0,0). It has the particular advantage that the computational characteristics of this function can be radically altered by varying the quantities Q and k.

Each of these functions was maximized or minimized by each of the

---

8. Investigated previously by [3], [4], [6], [7].

9. Investigated by [7].

10. Investigated by [9].

four algorithms and we employed several different starting points. Starting points were generated randomly and Tables 4-1 to 4-8 report the summary results for the experiments in which derivatives[11] were computed by numerical evaluation. These tables report for each method (a) the mean time and the standard deviation of time, in seconds; (b) the mean and the standard deviation of the number of function evaluations; (c) the mean value of the function at the point of convergence; (d) the mean distance of the alleged location of the extremum from the true location and (e) the failure rate of the algorithm.[12] All programs were written in FORTRAN IV and computations performed on an IBM 7094.

Certain general conclusions emerge easily. (1) In terms of mean time Powell performs best in almost all cases, its advantage being a factor of 1.5 to 3.5. (2) Davidon does well in some cases and poorly in others. (3) Quadratic Hill-Climbing-1 always beats Quadratic Hill-Climbing-2 and both tend to take more time than the other two methods.[13] (4) In terms of

---

11. Relevant for the two quadratic hill-climbing algorithms and for Davidon's method.

12. This statistic is based on samples of different sizes for the various algorithms. The reason for this is as follows. It was desired to have a matched sample in the sense that a random starting point was generated and <u>all</u> <u>four</u> algorithms applied with that starting point. The algorithms were applied in the following order: Quadratic Hill-Climbing-1, Quadratic Hill-Climbing-2, Davidon, Powell. If any algorithm failed (mistakenly converged to a "wrong" point or gave some other error termination), the successful solutions for that starting point by the preceding algorithms were discarded, a new starting point was generated and the cycle of four methods restarted. Thus, if in the attempt to have a sample of 10 tries each algorithm failed exactly once, the corresponding failure rates would be 1/14, 1/13, 1/12 and 1/11 respectively.

13. Some limited experimentation with analytically rather than numerically evaluated derivatives indicates some modest improvement in the time required for computation by the Quadratic Hill-Climbing and Davidon algorithms.

the number of function evaluations Davidon tends to perform worst, the two Quadratic Hill-Climbing methods best and Powell in between. (5) In terms of the failure rate, Quadratic Hill-Climbing-2 is superior to Quadratic Hill-Climbing-1 which is, in turn, superior to the other two methods. This factor becomes particularly noticeable for various forms of the Gaussian ridge, as shown in Tables 4-6, 4-7 and 4-8. For $Q = 1.0$ Davidon and Powell do quite well, with failure rates of .133 and .240 respectively. For $Q = .5$ the failure rate of Powell rises to .706 and for $Q = .1$ both Davidon and Powell have 100 per cent failure rates. For such a low value of Q Quadratic Hill-Climbing-1 also fails nearly half the time although Quadratic Hill-Climbing-2 still has a zero failure rate. Finally, for $Q = .01$ all four of the methods have a 100 per cent failure rate. One may conclude, parenthetically, that the parabolic Gaussian ridge is a severe test for apparently all algorithms.

As a final test we employed Quadratic Hill-Climbing-2 and Powell in maximizing the likelihood function associated with a small econometric model of the U.S. economy (Klein's Model I). After eliminating identities, the equations of this model are[14]

$$\begin{aligned}
u_{1i} &= -y_{1i}+\beta_{12}y_{2i}+\beta_{13}y_{3i}+\beta_{12}x_{1i}+\gamma_{12}x_{2i}-\beta_{13}x_{3i} - \beta_{13}x_{5i}+\beta_{13}x_{6i} \\
u_{2i} &= \beta_{21}y_{1i} - y_{2i} + \gamma_{24}x_{4i}+\beta_{21}x_{5i} + \gamma_{27}x_{7i} \qquad (4\text{-}1) \\
u_{3i} &= \beta_{31}y_{1i} - y_{3i} + \gamma_{32}x_{2i}+\gamma_{33}x_{3i}
\end{aligned}$$

where (i) $x_{1i}, x_{2i}, \ldots, x_{7i}$ are assumed to be nonstochastic; (ii) u_{1i}, u_{2i} and u_{3i} are (unobservable) stochastic terms assumed to be jointly normally distributed with mean vector $\mu = 0$ and covariance matrix $\sum$; (iii) observa-

---

14. See [1] and [5].

tions are available[15] on all $y_{1i}, y_{2i}, y_{3i}, x_{1i}, \ldots, x_{7i}$, $i = 1, \ldots, 21$; and (iv) the β's, γ's and the elements of $\sum$ are to be estimated. The system given by (4-1) can be written more compactly as

$$U = YB + XA \tag{4-2}$$

where Y is the (21 x 3) matrix of observations on the jointly dependent variables, X is the (21 x 7) matrix of observations on the predetermined variables, B and A are the corresponding coefficient matrices and U is the (21 x 3) matrix of residuals.

It is easy to show that the logarithm of likelihood function for y_1, y_2 and y_3 can be condensed to the following form[16]

$$L = -\frac{1}{2}\log\left(\frac{1}{21}\det(U'U)\right) + \log\left(\det(B)\right) \tag{4-3}$$

The function (4-3) was maximized employing two starting points: Start 1 is the point (0,0,0,0,0,0,0,0,0) and Start 2 is the point (.20410,.10250, .22967,.72465,.23273,.28341,.23116,.54600,.85400).[17,18] The results are displayed in Tables 9 and 10. Inspection of these tables reveals that Quadratic Hill-Climbing-2 performs substantially better than Powell in terms of time which is not surprising since it is generally true that Powell requires more function evaluations. Thus, as indicated earlier, when the function

---

15. See [5].

16. See [1].

17. This point represents the parameter estimates by the method of limited information maximum likelihood and are in effect maximum likelihood estimates taking the equations of the model one at a time and disregarding the a priori restrictions on the remaining equations.

18. Since the logarithm of a negative number is not defined, both algorithms were slightly modified in order to avoid generating steps that would take one into a forbidden region in the parameter space.

evaluation is relatively expensive, as in the present case, Quadratic Hill-Climbing has the advantage. In terms of accuracy the two methods are quite similar.

5. Conclusion

There is obviously no single best method for maximizing functions of many variables. Powell is very fast except when the evaluation of the function is very time consuming. Quadratic Hill-Climbing-2 is very robust and yields the true extremum with high probability. Davidon seems, on the whole, inferior to these two algorithms. The difficulty of choosing between Quadratic Hill-Climbing-2 and Powell is further evidenced by examining the question of accuracy. Thus, for the given set of the various computational tolerances, Powell sometimes gives better accuracy than the two Hill-Climbing methods (Table 4-1), but gives worse results in others (Table 4-2).[19] However, numerous questions remain unanswered. Some of these are: (1) Is it possible to categorize or arrive at a typology of functions which would allow one to judge reliably a priori which algorithm is going to operate better on a given function? (2) How do other methods (random search, pattern search, etc.) compare with the present methods? (3) How can methods that employ derivatives evaluate these more efficiently? These and other questions we hope to answer in some future work.

---

19. The attempt to improve the accuracy of Powell by altering these tolerances resulted in substantial increase in computing time.

Table 4-1

The Function $z = 100(y - x^2)^2 + (1 - x)^2$

Numerical Derivatives. 25 Random Starting Points in the Cube $-2 \leqq x, y \leqq 2$

		Quadratic Hill Climbing-1	Quadratic Hill Climbing-2	Davidon	Powell
Time (seconds)	Mean Standard Dev.	.84 .24	.92 .18	.79 .16	.49 .11
Number of Function Evaluations	Mean Standard Dev.	254 89	246 56	519 126	395 109
Function Value	Mean	$.20 \times 10^{-8}$	$.15 \times 10^{-10}$	$.54 \times 10^{-8}$	$.23 \times 10^{-9}$
Distance from True Extremum	Mean	$.84 \times 10^{-5}$	$.70 \times 10^{-5}$	$.36 \times 10^{-4}$	$.11 \times 10^{-5}$
Failure Rate		0.0	0.0	0.0	.038

Table 4-2

The Function $z = (x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4$

Numerical Derivatives. 25 Random Starting Points in the Hypercube $-3 \leqq x_i \leqq 3$

		Quadratic Hill Climbing-1	Quadratic Hill Climbing-2	Davidon	Powell
Time (seconds)	Mean Standard Dev.	1.77 .25	2.13 .40	1.15 .23	1.34 .24
Number of Function Evaluations	Mean Standard Dev.	624 87	632 93	562 125	992 192
Function Value	Mean	$.18 \times 10^{-15}$	$.32 \times 10^{-14}$	$.58 \times 10^{-9}$	$.38 \times 10^{-7}$
Distance from True Extremum	Mean	$.47 \times 10^{-4}$	$.71 \times 10^{-4}$	$.28 \times 10^{-2}$	$.34 \times 10^{-2}$
Failure Rate		0.0	0.0	0.0	.038

Table 4-3

The Function $z = \exp(\exp(\exp(\exp(\exp(\exp(\exp(-x^2 - 3y^2)/10)/10)/10)/10)))$

Numerical Derivatives. 10 Random Starting Points in the Cube $-2 \leqq x, y \leqq 2$

		Quadratic Hill Climbing-1	Quadratic Hill Climbing-2	Davidon	Powell
Time (seconds)	Mean Standard Dev.	.49 .09	.58 .15	.92 .45	.26 .03
Number of Function Evaluations	Mean Standard Dev.	89 23	95 33	276 154	61 12
Function Value	Mean	21.3205	21.3205	21.3205	21.3205
Distance from True Extremum	Mean	$.17 \times 10^{-2}$	$.27 \times 10^{-2}$	$.36 \times 10^{-2}$	$.46 \times 10^{-2}$
Failure Rate		.138	0.0	.240	.474

Table 4-4

The Function $z = (1.5 - x(1 - y))^2 + (2.25 - x(1 - y^2))^2 + (2.625 - x(1 - y^3))^2$

Numerical Derivatives. 10 Random Starting Points in the Cube $-2 \leqq x, y \leqq 2$

		Quadratic Hill Climbing-1	Quadratic Hill Climbing-2	Davidon	Powell
Time (seconds)	Mean Standard Dev.	.46 .06	.54 .10	.40 .09	.33 .08
Number of Function Evaluations	Mean Standard Dev.	138 24	144 29	241 74	240 76
Function Value	Mean	$.99 \times 10^{-11}$	$.18 \times 10^{-11}$	$.63 \times 10^{-11}$	$.85 \times 10^{-11}$
Distance from True Extremum	Mean	$.12 \times 10^{-5}$	$.18 \times 10^{-5}$	$.22 \times 10^{-5}$	$.12 \times 10^{-5}$
Failure Rate		.300	0.0	.214	.091

Table 4-5

The Function $z = 100(x_3 - x_1^2)^2 + 100(x_2 - x_1^2)^2 + (1 - x_1)^2$

Numerical Derivatives. 10 Random Starting Points in the Hypercube $-2 \leqq x_1, x_2, x_3 \leqq 2$

		Quadratic Hill Climbing-1	Quadratic Hill Climbing-2	Davidon	Powell
Time (seconds)	Mean	1.78	1.90	1.21	.65
	Standard Dev.	.27	.67	.16	.20
Number of Function Evaluations	Mean	640	584	846	563
	Standard Dev.	100	205	123	193
Function Value	Mean	$.39 \times 10^{-10}$	$.46 \times 10^{-10}$	$.32 \times 10^{-8}$	$.50 \times 10^{-8}$
Distance from True Extremum	Mean	$.12 \times 10^{-4}$	$.17 \times 10^{-4}$	$.56 \times 10^{-4}$	$.25 \times 10^{-4}$
Failure Rate		0.0	0.0	.167	0.0

Table 4-6

The Function $z = \left[1/(Q(1 + x^2))\right] \exp\left\{-(y - x^2)^2/(Q(1 + x^2))^2\right\}$

for Q = 1. Numerical Derivatives. 10 Random Starting Points in the Cube $-5 \leqq x, y \leqq 5$

		Quadratic Hill Climbing-1	Quadratic Hill Climbing-2	Davidon	Powell
Time (seconds)	Mean	.43	.57	.76	.32
	Standard Dev.	.06	.12	.38	.11
Number of Function Evaluations	Mean	105	123	393	158
	Standard Dev.	18	32	230	77
Function Value	Mean	1.00000	1.00000	.99991	1.00000
Distance from True Extremum	Mean	$.31 \times 10^{-4}$	$.35 \times 10^{-4}$	$.32 \times 10^{-2}$	$.30 \times 10^{-4}$
Failure Rate		0.0	0.0	.133	.240

Table 4-7

The Function $z = \left[1/\left(Q(1 + x^2)\right)\right] \exp\left\{-(y - x^2)^2/\left(Q(1 + x^2)\right)^2\right\}$

$Q = .5$. Numerical Derivatives. 10 Random Starting Points in the Cube $-5 \leqq x,y \leqq 5$

		Quadratic Hill Climbing-1	Quadratic Hill Climbing-2	Davidon	Powell
Time (seconds)	Mean Standard Dev.	.54 .13	.66 .14	.73 .29	.47 .17
Number of Function Eval-uations	Mean Standard Dev.	154 51	165 42	420 193	308 131
Function Value	Mean	2.00000	2.00000	1.99231	1.99627
Distance from True Extremum	Mean	$.14 \times 10^{-4}$	$.21 \times 10^{-4}$	$.21 \times 10^{-1}$	$.12 \times 10^{-1}$
Failure Rate		0.0	0.0	.091	.706

Table 4-8

The Function $z = \left[1/\left(Q(1 + x^2)\right)\right] \exp\left\{-(y - x^2)^2/\left(Q(1 + x^2)\right)^2\right\}$

$Q = .1$. Numerical Derivatives. 10 Random Starting Points in the Cube $-5 \leqq x, y \leqq 5$

		Quadratic Hill Climbing-1	Quadratic Hill Climbing-2	Davidon	Powell
Time (seconds)	Mean Standard Dev.	.77 .11	1.08 .18		
Number of Function Eval-uations	Mean Standard Dev.	228 37	278 50		
Function Value	Mean	10.0000	10.0000		
Distance from True Extremum	Mean	$.27 \times 10^{-4}$	$.36 \times 10^{-4}$		
Failure Rate		.474	0.0	1.000	1.000

Table 4-9

The Function $L = -\frac{1}{2}\log\left(\frac{1}{21}\det(U'U)\right) + \log\left(\det(B)\right)$

		Quadratic Hill Climbing-2	Powell
Time (seconds)	Start 1	41.55	64.48
	Start 2	23.52	63.78
Number of Function Evaluations	Start 1	3289	6324
	Start 2	1830	6233
Function Value	Start 1	-2.75562	-2.75797
	Start 2	-2.75581	-2.75625

Table 4-10

Location of Maximum

for the Function $L = -\frac{1}{2}\log\left(\frac{1}{21}\det(U'U)\right) + \log\left(\det(B)\right)$

Parameter	Quadratic Hill Climbing-2		Powell		Chow's Estimate[20]
	Start 1	Start 2	Start 1	Start 2	
β_{12}	-.16426	-.16957	-.15926	-.15295	-.16079
β_{13}	.82053	.85037	.81734	.77604	.81144
γ_{12}	.31561	.30402	.29741	.31927	.31295
β_{21}	.31122	.31977	.28905	.29574	.30569
γ_{24}	.30674	.30357	.31245	.31191	.30662
γ_{27}	.37202	.36948	.36366	.36714	.37170
β_{31}	-.77638	-.72073	-.79786	-.89165	-.80099
γ_{32}	1.05149	1.02007	.99084	1.07300	1.0519
γ_{33}	.85120	.84839	.85150	.85300	.85190

20. From [1].

Bibliography

[1] Chow, G. C., "Two Methods of Computing Full Information Maximum Likelihood Estimates in Simultaneous Stochastic Equations," International Economic Review, 9 (1968), 100-12.

[2] Davidon, W. C., "Variable Metric Method for Minimization," A.E.C. Research and Development Report, ANL-5990, 1959.

[3] Fletcher, R. and M. J. D. Powell, "A Rapidly Convergent Descent Method for Minimization," Computer Journal, 6 (1963), 163-68.

[4] Goldfeld, S. M., R. E. Quandt and H. F. Trotter, "Maximization by Quadratic Hill-Climbing," Econometrica, 34 (1966), 541-51.

[5] Klein, L. R., Economic Fluctuations in the United States, Cowles Commission Monograph No. 11, Wiley, 1950.

[6] Leon, A., "A Comparison Among Eight Known Optimizing Procedures," Space Sciences Laboratory, Working Paper No. 20, University of California, Berkeley, August 1964.

[7] Powell, M. J. D., "An Efficient Method for Finding the Minimum of A Function of Several Variables without Calculating Derivatives," Computer Journal, 7 (1964), 155-62.

[8] Schumer, M. A. and K. Steiglitz, "Adaptive Step Size Random Search," Paper presented at the Fifth Annual Allerton Conference on Circuit and System Theory, Urbana, October 1967.

[9] Spang, H. A., "A Review of Minimization Techniques for Nonlinear Functions," S.I.A.M. Review, 4 (1962),