AFOSR - TR - 72 - 1299

AD 745958

AERO-ASTRONAUTICS REPORT NO. 94

# COMPARISON OF SEVERAL GRADIENT ALGORITHMS FOR MATHEMATICAL PROGRAMMING PROBLEMS

by

A. MIELE, J.L. TIETZE, and A.V. LEVY



RICE UNIVERSITY

1972

Security Classification

# DOCUMENT CONTROL DATA - R & D

*(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)*

| | |
|---|---|
| 1. ORIGINATING ACTIVITY (Corporate author)<br>Department of Mechanical and Aerospace Engineering and Materials Science<br>Rice University, Houston, Texas 77001 | 2a. REPORT SECURITY CLASSIFICATION<br>UNCLASSIFIED<br>2b. GROUP |
| 3. REPORT TITLE<br>COMPARISON OF SEVERAL GRADIENT ALGORITHMS FOR MATHEMATICAL PROGRAMMING PROBLEMS | |
| 4. DESCRIPTIVE NOTES (Type of report and inclusive dates)<br>Scientific Interim | |
| 5. AUTHOR(S) (First name, middle initial, last name)<br>A. Miele, J.L. Tietze, and A.V. Levy | |
| 6. REPORT DATE<br>27 June 1972 | 7a. TOTAL NO. OF PAGES: 29 — 7b. NO. OF REFS: 3 |
| 8a. CONTRACT OR GRANT NO.<br>AFOSR-72-2185<br>b. PROJECT NO.<br>9749<br>c. 61102F<br>d. 681304 | 9a. ORIGINATOR'S REPORT NUMBER(S)<br>9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)<br>AFOSR-TR-72-[illegible] |
| 10. DISTRIBUTION STATEMENT<br>A. Approved for public release; distribution unlimited. | |
| 11. SUPPLEMENTARY NOTES<br>TECH, OTHER | 12. SPONSORING MILITARY ACTIVITY<br>Air Force Office of Scientific Research (NM)<br>1400 Wilson Blvd.<br>Arlington, Virginia 22209 |

13. ABSTRACT


In this paper, the numerical solution of the basic problem of mathematical programming is considered. This is the problem of minimizing a function $f(x)$ subject to a constraint $\varphi(x) = 0$. Here, $f$ is a scalar, $x$ an n-vector, and $\varphi$ a q-vector, with $q<n$.

Six variations of the sequential gradient-restoration algorithm and the combined gradient-restoration algorithm are considered, and their relative efficiency (in terms of number of iterations for convergence) is evaluated. The variations being considered are as follows: (i)SGRA-CR, sequential gradient-restoration algorithm, complete restoration, (ii)SGRA-IR, sequential gradient-restoration algorithm, incomplete restoration, (iii)SGRA-OR, sequential gradient-restoration algorithm, optional restoration, (iv)CGRA-NR, combined gradient-restoration algorithm, no restoration, (v) CGRA-AR, combined gradient-restoration algorithm, alternate restoration, (vi)CGRA-OR, combined gradient-restoration algorithm, optional restoration

Evaluation of these algorithms is accomplished through eight numerical examples. The first two examples pertain to quadratic functions subject to linear constraints. The remaining examples pertain to nonquadratic functions subject to nonlinear constraints. The results indicate that (a) the inclusion of a restoration phase is necessary for rapid convergence and (b) the algorithms with alternate restoration or optional restoration are the most efficient among those considered here.


DD FORM 1473

# Comparison of Several Gradient Algorithms For Mathematical Programming Problems[1]

by

A. MIELE[2], J.L. TIETZE[3], and A.V. LEVY[4]

Abstract. In this paper, the numerical solution of the basic problem of mathematical programming is considered. This is the problem of minimizing a function $f(x)$ subject to a constraint $\varphi(x) = 0$. Here, $f$ is a scalar, $x$ an $n$-vector, and $\varphi$ a $q$-vector, with $q < n$.

Six variations of the sequential gradient-restoration algorithm and the combined gradient-restoration algorithm are considered, and their relative efficiency (in terms of number of iterations for convergence) is evaluated. The variations being considered are as follows:

(i) SGRA-CR, sequential gradient-restoration algorithm, complete restoration,

(ii) SGRA-IR, sequential gradient-restoration algorithm, incomplete restoration,

(iii) SGRA-OR, sequential gradient-restoration algorithm, optional restoration,

[1] This research was supported by the Office of Scientific Research, Office of Aerospace Research, United States Air Force, Grant No. [redacted]AFOSR-72-2185.

[2] Professor of Astronautics and Mathematical Sciences, Rice University, Houston, Texas.

[3] Graduate Student in Electrical Engineering, Rice University, Houston, Texas.

[4] Post-Doctoral Fellow in Aero-Astronautics, Rice University, Houston, Texas.

(iv) CGRA-NR, combined gradient-restoration algorithm, no restoration,

(v) CGRA-AR, combined gradient-restoration algorithm, alternate restoration,

(vi) CGRA-OR, combined gradient-restoration algorithm, optional restoration.

Evaluation of these algorithms is accomplished through eight numerical examples. The first two examples pertain to quadratic functions subject to linear constraints. The remaining examples pertain to nonquadratic functions subject to nonlinear constraints. The results indicate that (a) the inclusion of a restoration phase is necessary for rapid convergence and (b) the algorithms with alternate restoration or optional restoration are the most efficient among those considered here.

## 1. Introduction

In previous papers (Ref. 1-3), two basic algorithms for the minimization of constrained functions were developed: the sequential gradient-restoration algorithm (SGRA) and the combined gradient-restoration algorithm (CGRA). The former is an iterative algorithm which consists of the alternate succession of gradient phases and restoration phases; the latter is an iterative algorithm in which the gradient phase and the restoration phase are combined in a single phase.

In the gradient phase of SGRA, one generates a displacement $\Delta x$ lowering the value of the function, while avoiding excessive constraint violation; in the restoration phase of SGRA, one generates a displacement $\Delta x$ restoring the constraint to a predetermined accuracy, while avoiding excessive change in the value of the function. On the other hand, in the gradient-restoration phase of CGRA, one generates a displacement $\Delta x$ lowering the value of the augmented function, while simultaneously reducing the constraint violation.

In this paper, six variations of the sequential gradient-restoration algorithm and the combined gradient-restoration algorithm are considered, and their relative efficiency (in terms of number of iterations for convergence) is evaluated through eight numerical examples. The variations being considered are indicated below:

(i) SGRA-CR, sequential gradient-restoration algorithm, complete restoration,

(ii) SGRA-IR, sequential gradient-restoration algorithm, incomplete restoration,

(iii) SGRA-OR, sequential gradient-restoration algorithm, optional restoration,

(iv) CGRA-NR, combined gradient-restoration algorithm, no restoration,

(v) CGRA-AR, combined gradient-restoration algorithm, alternate restoration,

(vi) CGRA-OR, combined gradient-restoration algorithm, optional restoration.

## 2. Statement of the Problem

We consider the problem of minimizing the function

$$f = f(x) \tag{1}$$

subject to the constraint

$$\varphi(x) = 0 \tag{2}$$

where f is a scalar, x an n-vector, and $\varphi$ a q-vector, with $q < n$. Here, all vectors are column vectors. It is assumed that the first and second partial derivatives of the function f(x) and $\varphi(x)$ exist and are continuous and that the constrained minimum exists.

### 2.1. First-Order Conditions.

From theory of maxima and minima, it is known that the above problem is equivalent to that of minimizing the augmented function

$$F(x, \lambda) = f(x) + \lambda^T \varphi(x) \tag{3}$$

subject to the constraint (2). Here, the q-vector $\lambda$ is the Lagrange multiplier and the superscript T denotes the transpose of a matrix. If

$$F_x(x, \lambda) = f_x(x) + \varphi_x(x)\lambda \tag{4}$$

denotes the gradient of the augmented function, the optimum solution for x and $\lambda$ must satisfy the relations

$$\varphi(x) = 0 \quad , \quad F_x(x, \lambda) = 0 \tag{5}$$

which are a system of $n + q$ equations in the $n + q$ components of $x$ and $\lambda$. In Eqs. (4)-(5), the gradients $f_x$ and $F_x$ denotes n-vectors and the matrix $\varphi_x$ is $n \times q$.

2.2. Approximate Solutions. Since the system (5) is generally nonlinear, approximate methods must be employed. In this connection, we introduce here the scalar performance indexes

$$P(x) = \varphi^T(x)\varphi(x) \quad , \quad Q(x, \lambda) = F_x^T(x, \lambda)F_x(x, \lambda) \tag{6}$$

which measure the errors in the constraint and the optimum condition, respectively. Then, we observe that $P = 0$ and $Q = 0$ for the optimum solution, while $P > 0$ and/or $Q > 0$ for any approximation to the solution. When approximate methods are used, they must ultimately lead to values of $x$ and $\lambda$ such that

$$P(x) \le \epsilon_1 \quad , \quad Q(x, \lambda) \le \epsilon_2 \tag{7}$$

Alternatively, (7) can be replaced by

$$R(x, \lambda) \le \epsilon_3 \tag{8}$$

where

$$R(x, \lambda) = P(x) + Q(x, \lambda) \tag{9}$$

denotes the cumulative error in the constraint and the optimum condition. In (7)-(8), $\epsilon_1$, $\epsilon_2$, $\epsilon_3$ are small, preselected numbers. Note that, if one chooses $\epsilon_1 = \epsilon_2 = \epsilon_3$, satisfaction of Ineq. (8) implies satisfaction of Ineqs. (7).

## 3. Description of the Algorithms

In this section, the algorithms being investigated are described.

(i) SGRA-CR: Sequential gradient-restoration algorithm, complete restoration. This algorithm consists of the alternate succession of gradient phases and restoration phases.

The gradient phase is started providing

$$P(x) \leq \epsilon_1 \tag{10}$$

It involves a single iteration, in which the augmented function is reduced subject to an upper limit for the constraint error, that is,[5]

$$F(\tilde{x}, \lambda) < F(x, \lambda) \quad , \quad P(\tilde{x}) \leq \epsilon_4 \tag{11}$$

The restoration phase is started providing

$$P(x) > \epsilon_1 \tag{12}$$

It involves several iterations, in each of which the constraint error is reduced, that is,

$$P(\tilde{x}) < P(x) \tag{13}$$

The restoration phases is terminated whenever Ineq. (10) is satisfied.

---

[5] The symbol x denotes the nominal point, $\tilde{x}$ the varied point, and $\lambda$ the Lagrange multiplier.

Remark. The algorithm is started with a gradient phase if Ineq. (10) is satisfied or a restoration phase if Ineq. (10) is violated. Normally, a gradient phase is followed by a restoration phase. Occasionally, the gradient phase is followed by another gradient phase, that is, the restoration phase is bypassed: this is precisely the case whenever Ineq. (10) is satisfied.

(ii) SGRA-IR: Sequential gradient-restoration algorithm, incomplete restoration. This algorithm consists of the alternate succession of gradient phases and restoration phases.

The gradient phase is started regardless of whether Ineq. (10) is satisfied. It involves a single iteration, in which the augmented function is reduced subject to an upper limit on the constraint error, that is,

$$F(\tilde{x}, \lambda) < F(x, \lambda) \quad , \quad P(\tilde{x}) \leq P(x) + \varepsilon_4 \tag{14}$$

The restoration phase is started only if Ineq. (12) is satisfied. It involves a single iteration, in which the constraint error is reduced in accordance with Ineq. (13).

The starting condition and the bypassing condition for SGRA-IR are identical with those of SGRA-CR (see Remark).

(iii) SGRA-OR: Sequential gradient-restoration algorithm, optional restoration. This algorithm consists of the alternate succession of gradient phases and restoration phases.

The gradient phase is started providing

$$Z(x, \lambda) \leq 1 \tag{15}$$

where the parameter Z is defined by

$$Z = \epsilon P(x)/Q(x, \lambda) \qquad (16)$$

with

$$\epsilon = \epsilon_2/\epsilon_1 \qquad (17)$$

It involves a single iteration, in which the augmented function is reduced in accordance with Ineqs. (14).

The restoration phase is started providing

$$Z(x, \lambda) > 1 \qquad (18)$$

It involves several iterations, in each of which the constraint error is reduced in accordance with Ineq. (13). The restoration phase is terminated whenever Ineq. (15) is satisfied.

The bypassing condition for SGRA-OR is identical with that of SGRA-CR (see Remark).

(iv) CGRA-NR: Combined gradient-restoration algorithm, no restoration. In this algorithm, the gradient phase and the restoration phase are combined together in a single phase. It involves a single iteration, in which the augmented function is reduced in accordance with Ineqs. (14).

(v) CGRA-AR: Combined gradient-restoration algorithm, alternate restoration. This algorithm consists of the alternate succession of combined gradient-restoration phases and restoration phases.

The combined gradient-restoration phase is started regardless of whether Ineq. (10) is satisfied. It involves a single iteration, in which the augmented function is reduced in accordance with Ineqs. (14).

The restoration phase is started only if Ineq. (12) is satisfied. It involves a single iteration, in which the constraint error is reduced in accordance with Ineq. (13).

The starting condition and the bypassing condition for CGRA-AR are identical with those of SGRA-CR (see Remark).

(vi) CGRA-OR: Combined gradient-restoration algorithm, optional restoration. This algorithm consists of the alternate succession of combined gradient-restoration phases and restoration phases.

The combined gradient-restoration phase is started providing Ineq. (15) is satisfied. It involves a single iteration,in which the augmented function is reduced in accordance with Ineqs. (14).

The restoration phase is started providing Ineq. (18) is satisfied. It involves several iterations, in each of which the constraint error is reduced in accordance with Ineq. (13). The restoration phase is terminated whenever Ineq. (15) is satisfied.

The bypassing condition for CGRA-OR is identical with that of SGRA-CR (see Remark).

Remark. For the algorithms with optional restoration, the multiplier $\lambda$ appearing in (15)-(18) is computed as follows. For SGRA-OR, Eq. (19-1) must be solved with $C_1 = 1$ and $C_2 = 0$. For CGRA-OR, Eq. (19-1) must be solved with $C_1 = 1$ and $C_2 = 1$.

## 4. Generalized Algorithm

Let x denote the nominal point, $\tilde{x}$ the varied point, $\Delta x$ the displacement leading from the nominal point to the varied point, and $\alpha$ the stepsize. With this understanding, the previous algorithms can be represented in the following generalized form:

$$\varphi_x^T(x)\varphi_x(x)\lambda + C_1\varphi_x^T(x)f_x(x) - C_2\varphi(x) = 0 \quad (19\text{-}1)$$

$$p = C_1 f_x(x) + \varphi_x(x)\lambda \quad (19\text{-}2)$$

$$\Delta x = -\alpha p \quad (19\text{-}3)$$

$$\tilde{x} = x + \Delta x \quad (19\text{-}4)$$

For given nominal point x and constants $C_1$ and $C_2$, Eqs. (19) constitute a complete iteration leading to the varied point $\tilde{x}$ providing one specifies the stepsize $\alpha$. The constants $C_1$ and $C_2$ depend on the particular algorithm and take the values given in Table 1. The detailed derivation of Eqs. (19) is presented in Refs. 1-3 and, hence, is not repeated here.

Table 1. Characteristic constants.

| Algorithm | Phase | $C_1$ | $C_2$ |
|---|---|---|---|
| SGRA | Gradient | 1 | 0 |
| | Restoration | 0 | 1 |
| CGRA | Gradient-restoration | 1 | 1 |
| | Restoration | 0 | 1 |

5. Stepsize Determination

For all of the previous algorithms, the position vector at the end of any step can be written as

$$\tilde{x} = x - \alpha p \tag{20}$$

where p denotes the search direction, which is given by (19-2). This is a one-parameter family of varied points $\tilde{x}$, for which the augmented function (3), the constraint error (6-1), and the error in the optimum condition (6-2) take the form

$$F(\tilde{x}, \lambda) = F(x - \alpha p, \lambda) = \tilde{F}(\alpha) \tag{21}$$

$$P(\tilde{x}) = P(x - \alpha p) = \tilde{P}(\alpha) \tag{22}$$

$$Q(\tilde{x}, \lambda) = Q(x - \alpha p, \lambda) = \tilde{Q}(\alpha) \tag{23}$$

For the gradient phase of a SGRA-algorithm or the combined gradient-restoration phase of a CGRA-algorithm, Ineqs. (11) and (14) can be written in the general form

$$\tilde{F}(\alpha) < \tilde{F}(0) \quad , \quad \tilde{P}(\alpha) \le \tilde{P}(0) + \epsilon_4 \tag{24}$$

Their satisfaction can be ensured by employing a bisection process, starting from a suitably chosen reference stepsize

$$\alpha = \alpha_o \tag{25}$$

For the determination of the reference stepsize, see Section 6.

For the restoration phase of a SGRA-algorithm or a CGRA-algorithm, Ineq. (13) can be written as

$$\tilde{P}(\alpha) < \tilde{P}(0) \tag{26}$$

Its satisfaction can be ensured by employing a bisection process, starting from the reference stepsize

$$\alpha = 1 \tag{27}$$

This value reduces the constraint error $P(x)$ to zero, if the constraint function $\varphi(x)$ is linear in $x$.

## 6. Reference Stepsize

The search technique outlined in Section 5 for the gradient stepsize employs a bisection process, starting from the reference stepsize (25), until satisfaction of Ineqs. (24) occurs. A procedure useful to determine this reference stepsize is outlined here and is based on a quadratic representation of the augmented function associated with the one-parameter family of solutions (20).

Let the function $\tilde{F}(\alpha)$ be represented in the quadratic form

$$\tilde{F}(\alpha) = k_0 + k_1\alpha + k_2\alpha^2 \tag{28}$$

and let the coefficients of the quadratic be determined so as to match the values of the ordinate and the slope at $\alpha = 0$ and the value of the ordinate at $\alpha = 1$. This yields the relations

$$\tilde{F}(0) = k_0 \quad , \quad \tilde{F}_\alpha(0) = k_1 \quad , \quad \tilde{F}(1) = k_0 + k_1 + k_2 \tag{29}$$

which imply that

$$k_0 = \tilde{F}(0) \quad , \quad k_1 = -\tilde{Q}(0) \quad , \quad k_2 = \tilde{F}(1) - \tilde{F}(0) + \tilde{Q}(0) \tag{30}$$

With the coefficients known, the following possibilities arise:

$$\text{(i)} \;\; k_2 > 0 \quad \text{or} \quad \text{(ii)} \;\; k_2 \le 0 \tag{31}$$

In Case (i), the quadratic function (28) has a minimum for the following value oı the gradient stepsize:

$$\alpha = - k_1/2k_2 \tag{32}$$

In Case (ii), the quadratic function (28) decreases monotonically with $\alpha$.

This suggests the use of the following reference values for the gradient stepsize:

$$\alpha_o = -k_1/2k_2 \quad \text{if} \quad k_2 > 0$$
$$\alpha_o = 1 \quad \text{if} \quad k_2 \leq 0 \qquad (33)$$

7. Experimental Conditions

In order to evaluate the previous algorithms, eight numerical examples were considered. The first two examples pertain to quadratic functions subject to linear constraints. The remaining examples pertain to nonquadratic functions subject to nonlinear constraints. Each example was solved with the three versions of SGRA and the three versions of CGRA outlined in Section 3. All of the algorithms were programmed in FORTRAN IV, and the numerical results were obtained using a Burroughs B-5500 computer and double-precision arithmetic.

Starting Point. For all of the examples, the nominal point chosen to start an algorithm was defined by

$$x_1 = x_2 = \dots = x_n = 2 \tag{34}$$

where n denotes the dimension of the vector x.

Search Technique. The determination of the gradient stepsize and the restoration stepsize was performed in accordance with Sections 5 and 6. For the gradient phase, the stepsize $\alpha$ was subject to the inequalities

$$\tilde{F}(\alpha) < \tilde{F}(0) \quad , \quad \tilde{P}(\alpha) \leq \tilde{P}(0) + 1 \tag{35}$$

For the restoration phase, the stepsize was subject to the inequality

$$\tilde{P}(\alpha) < \tilde{P}(0) \tag{36}$$

Convergence. Convergence of an algorithm was defined through the inequalities

$$P(x) \leq 10^{-8} \quad , \quad Q(x, \lambda) \leq 10^{-4} \tag{37}$$

Nonconvergence. Conversely, nonconvergence of an algorithm was defined by means of the inequalities

(a) $$N > 100 \tag{38-1}$$

or

(b) $$N_s > 20 \tag{38-2}$$

or

(c) $$M > 0.4 \times 10^{69} \tag{38-3}$$

Here, N is the iteration number, $N_s$ is the number of bisections of the stepsize $\alpha$ required to satisfy Ineq. (35) or (36), and M is the modulus of any of the quantities employed in the algorithm. Satisfaction of Ineq. (38-1) indicates divergence or extreme slowness of convergence; satisfaction of Ineq. (38-2) indicates extreme smallness of the displacement $\Delta x$; and satisfaction of Ineq. (38-3) indicates exponential overflow. Each of these situations is undesirable.

## 8. Numerical Examples

In this section, eight numerical examples are described. The first two examples pertain to quadratic functions subject to linear constraints. The remaining examples pertain to nonquadratic functions subject to nonlinear constraints.

Example 8.1. Consider the problem of minimizing the function

$$f = (x_1 - x_2)^2 + (x_2 + x_3 - 2)^2 + (x_4 - 1)^2 + (x_5 - 1)^2 \tag{39}$$

subject to the constraints

$$x_1 + 3x_2 = 0 \quad , \quad x_3 + x_4 - 2x_5 = 0 \quad , \quad x_2 - x_5 = 0 \tag{40}$$

This function admits the relative minimum $f = 4.0930$ at the point defined by

$$x_1 = -0.7674 \; , \; x_2 = 0.2558 \; , \; x_3 = 0.6279 \; , \; x_4 = -0.1162 \; , \; x_5 = 0.2558 \tag{41}$$

and

$$\lambda_1 = 2.0465 \; , \; \lambda_2 = 2.2325 \; , \; \lambda_3 = -5.9534 \tag{42}$$

Example 8.2. Consider the problem of minimizing the function

$$f = (4x_1 - x_2)^2 + (x_2 + x_3 - 2)^2 + (x_4 - 1)^2 + (x_5 - 1)^2 \tag{43}$$

subject to the constraints

$$x_1 + 3x_2 = 0 \quad , \quad x_3 + x_4 - 2x_5 = 0 \quad , \quad x_2 - x_5 = 0 \tag{44}$$

This function admits the relative minimum $f = 5.3266$ at the point defined by

$$x_1 = -0.9455 \times 10^{-1},\ x_2 = 0.3151 \times 10^{-1},\ x_3 = 0.5157,\ x_4 = -0.4527,\ x_5 = 0.3151 \times 10^{-1} \quad (45)$$

and

$$\lambda_1 = 3.2779 \quad , \quad \lambda_2 = 2.9054 \quad , \quad \lambda_3 = -7.7478 \quad (46)$$

Example 8.3. Consider the problem of minimizing the function

$$f = (x_1 - 1)^2 + (x_1 - x_2)^2 + (x_2 - x_3)^4 \quad (47)$$

subject to the constraint

$$x_1(1 + x_2^2) + x_3^4 - 4 - 3\sqrt{2} = 0 \quad (48)$$

This function admits the relative minimum $f = 0.3256 \times 10^{-1}$ at the point defined by

$$x_1 = 1.1048 \quad , \quad x_2 = 1.1966 \quad , \quad x_3 = 1.5352 \quad (49)$$

and

$$\lambda_1 = -0.1072 \times 10^{-1} \quad (50)$$

Example 8.4. Consider the problem of minimizing the function

$$f = (x_1 - 1)^2 + (x_1 - x_2)^2 + (x_3 - 1)^2 + (x_4 - 1)^4 + (x_5 - 1)^6 \quad (51)$$

subject to the constraints

$$x_1^2 x_4 + \sin(x_4 - x_5) - 2\sqrt{2} = 0 \quad , \quad x_2 + x_3^4 x_4^2 - 8 - \sqrt{2} = 0 \quad (52)$$

This function admits the relative minimum $f = 0.2415$ at the point defined by

$$x_1 = 1.1661 \;,\; x_2 = 1.1821 \;,\; x_3 = 1.3802 \;,\; x_4 = 1.5060 \;,\; x_5 = 0.6109 \qquad (53)$$

and

$$\lambda_1 = -0.8553 \times 10^{-1} \;,\; \lambda_2 = -0.3187 \times 10^{-1} \qquad (54)$$

Example 8.5. Consider the problem of minimizing the function

$$f = (x_1 - 1)^2 + (x_1 - x_2)^2 + (x_2 - x_3)^2 + (x_3 - x_4)^4 + (x_4 - x_5)^4 \qquad (55)$$

subject to the constraints

$$x_1 + x_2^2 + x_3^3 - 2 - 3\sqrt{2} = 0 \;,\; x_2 - x_3^2 + x_4 + 2 - 2\sqrt{2} = 0 \;,\; x_1 x_5 - 2 = 0 \qquad (56)$$

This function admits the relative minimum $f = 0.7877 \times 10^{-1}$ at the point defined by

$$x_1 = 1.1911 \;,\; x_2 = 1.3626 \;,\; x_3 = 1.4728 \;,\; x_4 = 1.6350 \;,\; x_5 = 1.6790 \qquad (57)$$

and

$$\lambda_1 = -0.3882 \times 10^{-1} \;,\; \lambda_2 = -0.1672 \times 10^{-1} \;,\; \lambda_3 = -0.2879 \times 10^{-3} \qquad (58)$$

Example 8.6. Consider the problem of minimizing the function

$$f = 0.01(x_1 - 1)^2 + (x_2 - x_1^2)^2 \qquad (59)$$

subject to the inequality constraint

$$x_1 \leq -1 \qquad (60)$$

Introduce the auxiliary variable $x_3$ defined by

$$x_1 + x_3^2 + 1 = 0 \tag{61}$$

Then, the previous problem can be recast as that of minimizing the function (59) subject to the equality constraint (61). The function (59) admits the relative minimum $f = 0.04$ at the point defined by

$$x_1 = -1 \quad , \quad x_2 = 1 \quad , \quad x_3 = 0 \tag{62}$$

and

$$\lambda_1 = 0.04 \tag{63}$$

Example 8.7. Consider the problem of minimizing the function

$$f = -x_1 \tag{64}$$

subject to the inequality constraints

$$x_2 \geq x_1^3 \quad , \quad x_2 \leq x_1^2 \tag{65}$$

Introduce the auxiliary variables $x_3$ and $x_4$ defined by

$$x_2 - x_1^3 - x_3^2 = 0 \quad , \quad x_1^2 - x_2 - x_4^2 = 0 \tag{66}$$

Then, the previous problem can be recast as that of minimizing the function (64) subject to the equality constraints (66). The function (64) admits the relative minimum $f = -1$ at the point defined by

$$x_1 = 1 \quad , \quad x_2 = 1 \quad , \quad x_3 = 0 \quad , \quad x_4 = 0 \tag{67}$$

and

$$\lambda_1 = -1 \quad , \quad \lambda_2 = -1 \tag{68}$$

Example 8.8. Consider the problem of minimizing the function

$$f = \log x_3 - x_2 \tag{69}$$

subject to the equality constraint

$$x_2^2 + x_3^2 - 4 = 0 \tag{70}$$

and the inequality constraint

$$x_3 \geq 1 \tag{71}$$

Introduce the auxiliary variable $x_1$ defined by

$$x_3 = 1 + x_1^2 \tag{72}$$

Then, the previous problem can be recast as that of minimizing the function

$$f = \log(1 + x_1^2) - x_2 \tag{73}$$

subject to the equality constraint

$$(1 + x_1^2)^2 + x_2^2 - 4 = 0 \tag{74}$$

Note that $x_3$ has been eliminated from the problem and can be computed a posteriori with (72). The function (73) admits the relative minimum $f = -\sqrt{3}$ at the point defined by

$$x_1 = 0 \; , \quad x_2 = \sqrt{3} \; , \quad x_3 = 1 \qquad (75)$$

and

$$\lambda_1 = 1/2\sqrt{3} \qquad (76)$$

## 9. Results and Conclusions

The examples described in Section 8 were solved with the three versions of SGRA and the three versions of CGRA described in Section 3. The numerical results are presented in Tables 2-3, where the number of iterations for convergence $N_*$ is shown. For the eight examples considered, Table 4 shows the cumulative number of iterations for convergences $\Sigma N_*$. From the tables, the following conclusions arise: (a) a restoration of some form is necessary for rapid convergence; and (b) while SGRA-CR is the most stable among the algorithms considered here, rapidity of convergence can be increased somewhat if one employs algorithms with alternate restoration or optional restoration.

Table 2. Number of iterations for convergence $N_*$.

| Example | SGRA-CR | SGRA-IR | SGRA-OR |
|---|---|---|---|
| 8.1 | 5 | 5 | 5 |
| 8.2 | 8 | 8 | 8 |
| 8.3 | 18 | 14 | 16 |
| 8.4 | 56 | 51 | 42 |
| 8.5 | 8 | 7 | 7 |
| 8.6 | 15 | 12 | 16 |
| 8.7 | 9 | 15 | 9 |
| 8.8 | 11 | 11 | 10 |

Table 3. Number of iterations for convergence $N_*$.

| Example | CGRA-NR | CGRA-AR | CGRA-OR |
|---|---|---|---|
| 8.1 | 17 | 5 | 5 |
| 8.2 | 65 | 8 | 8 |
| 8.3 | 22 | 16 | 16 |
| 8.4 | 36 | 54 | 43 |
| 8.5 | 7 | 7 | 7 |
| 8.6 | >100 | 19 | 13 |
| 8.7 | 13 | 7 | 9 |
| 8.8 | 15 | 8 | 10 |

## References

1. MIELE, A., HUANG, H.Y., and HEIDEMAN, J.C., Sequential Gradient-Restoration Algorithm for the Minimization of Constrained Functions, Ordinary and Conjugate Gradient Versions, Journal of Optimization Theory and Applications, Vol. 4, No. 4, 1969.

2. MIELE, A., and HEIDEMAN, J.C., Mathematical Programming for Constrained Minimal Problems, Part 1, Sequential Gradient-Restoration Algorithm, Rice University, Aero-Astronautics Report No. 59, 1969.

3. MIELE, A., HEIDEMAN, J.C., and LEVY, A.V., Mathematical Programming for Constrained Minimal Problems, Part 3, Combined Gradient-Restoration Algorithm, Rice University, Aero-Astronautics Report No. 68, 1970.

Table 4. Cumulative number of iterations for convergence $\Sigma N_*$.

| Algorithm | $\Sigma N_*$ |
|---|---|
| SGRA-CR | 130 |
| SGRA-IR | 123 |
| SGRA-OR | 113 |
| CGRA-NR | >275 |
| CGRA-AR | 124 |
| CGRA-OR | 111 |