72-26,447

LEVY, Alejandro Velasco, 1941-
MATHEMATICAL PROGRAMMING FOR CONSTRAINED MINIMAL PROBLEMS: MODIFICATIONS AND EXTENSIONS OF THE CONJUGATE GRADIENT-RESTORATION ALGORITHM.

Rice University, Ph.D., 1972
Mathematics

University Microfilms, A XEROX Company, Ann Arbor, Michigan

RICE UNIVERSITY

MATHEMATICAL PROGRAMMING FOR CONSTRAINED MINIMAL PROBLEMS:

MODIFICATIONS AND EXTENSIONS

OF THE CONJUGATE GRADIENT-RESTORATION ALGORITHM

by

ALEJANDRO VELASCO LEVY

A THESIS SUBMITTED
IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Thesis Director's signature

Angelo Miele

Houston, Texas

May, 1972

Abstract

Mathematical Programming for Constrained Minimal Problems:

Modifications and Extensions

of the Conjugate Gradient-Restoration Algorithm

by

ALEJANDRO VELASCO LEVY

In this thesis, the problem of minimizing a function f(x) subject to a constraint $\varphi(x) = 0$ is considered, where f is a scalar, x an n-vector, and φ a q-vector, with $q < n$. Several conjugate gradient-restoration algorithms are analyzed: these algorithms are composed of the alternate succession of conjugate gradient phases and restoration phases. In the conjugate gradient phase, one tries to improve the value of the function while avoiding excessive constraint violation. In the restoration phase, one tries to reduce the constraint error, while avoiding excessive change in the value of the function.

Concerning the conjugate gradient phase, two classes of algorithms are considered: for algorithms of Class I, the multiplier λ is determined so that the error in the optimum condition is minimized for given x; for algorithms of Class II, the multiplier λ is determined so that the constraint is satisfied to first order. Concerning the restoration phase, two topics are investigated: (a) restoration type, that is, complete restoration versus incomplete restoration and (b) restoration frequency, that is, frequent restoration versus infrequent restoration.

Depending on the combination of type and frequency of restoration, four algorithms are generated within Class I and within Class II, respectively:

Algorithm (α) is characterized by complete and frequent restoration; Algorithm (β) is characterized by incomplete and frequent restoration; Algorithm (γ) is characterized by complete and infrequent restoration; and Algorithm (δ) is characterized by incomplete and infrequent restoration.

If the function f(x) is quadratic and the constraint $\varphi(x)$ is linear, all of the previous algorithms are identical, that is, they produce the same sequence of points and converge to the solution in the same number of iterations. This number of iterations is at most $N_* = n-q$ if the starting point x_s is such that $\varphi(x_s) = 0$ and at most $N_* = 1+n-q$ if the starting point x_s is such that $\varphi(x_s) \neq 0$.

In order to illustrate the theory, five numerical examples are developed. The first example refers to a quadratic function and a linear constraint. The remaining examples refer to a nonquadratic function and a nonlinear constraint. For the linear-quadratic example, all the algorithms behave identically, as predicted by the theory. For the nonlinear-nonquadratic examples, Algorithm (II-δ), which is characterized by incomplete and infrequent restoration, exhibits superior convergence characteristics.

It is of interest to compare Algorithm (II-δ) with Algorithm (I-α), which is the sequential conjugate gradient-restoration algorithm of Ref. 1 and is characterized by complete and frequent restoration. For the nonlinear-nonquadratic examples, Algorithm (II-δ) converges to the solution in a number of iterations which is about one half to two thirds that of Algorithm (I-α).

ACKNOWLEDGMENT

The author is indebted to Dr. Angelo Miele for suggesting the topic and helpful discussions. He would also like to thank Mr. E. E. Cragg for checking the accuracy of the numerical results. This research was supported by the Office of Scientific Research, Office of Aerospace Research, United States Air Force, Grant No. AF-AFOSR-72-2185.

TABLE OF CONTENTS

1. Introduction

In Ref. 1, the problem of minimizing a function $f(x)$ subject to a constraint $\varphi(x) = 0$ was considered, where f is a scalar, x an n-vector, and φ a q-vector, with $q < n$. A sequential conjugate gradient-restoration algorithm was presented. The basic cycle of this algorithm involves a conjugate gradient phase and a restoration phase. In the conjugate gradient phase, one tries to improve the value of the function, while avoiding excessive constraint violation. In the restoration phase (Ref. 2), one tries to restore the constraint to a predetermined degree of accuracy, while avoiding excessive change in the value of the function.

In Ref. 3, a combined conjugate gradient-restoration algorithm was presented. It differs from the sequential conjugate gradient-restoration algorithm in that the displacement Δx leads toward the minimum point while simultaneously leading toward constraint satisfaction.

In this thesis, several modifications and extensions of the above algorithm are studied with the basic objective of improving the convergence characteristics. Concerning the conjugate gradient phase, two classes of algorithms are considered: for algorithms of Class I, the multiplier λ is determined as in Refs. 1 and 4, so that the error in the optimum condition is minimized for given x; for algorithms of Class II, the multiplier λ is determined as in Ref. 3, so that the constraint is satisfied to first order.

Concerning the restoration phase, the minimum distance algorithm of Refs. 1-2 is employed. Extensive computer experimentation has shown that this restoration algorithm is more stable and yields faster convergence than that of Ref. 3.

With the above ideas in mind, two topics are investigated: (a) restoration type and (b) restoration frequency. Regarding (a), we distinguish complete restoration from incomplete restoration: complete restoration involves several iterations until the constraint error becomes smaller than some preselected number; incomplete restoration involves a single iteration. Regarding (b), we distinguish frequent restoration from infrequent restoration. Frequent restoration means that a restoration phase precedes every conjugate gradient step; infrequent restoration means that a restoration phase precedes every n-q conjugate gradient steps.

Depending on the combination of type and frequency of restoration, four algorithms are generated within Class I and within Class II. They are designated by (α), (β), (γ), (δ) and are shown in Table 1. As an example, Algorithm (I-α) is the algorithm of Class I characterized by complete and frequent restoration; as another example, Algorithm (II-δ) is the algorithm of Class II, characterized by incomplete and infrequent restoration. Incidentally, Algorithm (I-α) is the sequential conjugate gradient-restoration algorithm of Ref. 1.

1.1 Outline. A statement of the problem is given in Section 2, together with the exact first-order conditions and a discussion of approximate methods. The conjugate gradient phase is discussed in Section 3, the restoration phase is discussed in Section 4, and a summary of conjugate-gradient restoration algorithms is given in Section 5. The experimental conditions and five numerical examples are described in Section 6. Next, the numerical results are given in Section 7, together with the conclusions. Finally, the details of the search technique employed in the conjugate gradient phase are given in the Appendix.

Table 1. Classification of Algorithms of Class I and Class II.

		Restoration type	
		Complete	Incomplete
Restoration frequency	Frequent	(α)	(β)
	Infrequent	(γ)	(δ)

2. Statement of the Problem

We consider the problem of minimizing the function

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

subject to the constraint

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

In the above equations, f is a scalar, x an n-vector, and φ a q-vector[1], where $q < n$. It is assumed that the first and second partial derivatives of the functions f and φ with respect to x exist and are continuous; it is also assumed that the constrained minimum exists.

2.1. Exact First-Order Conditions.

From theory of maxima and minima, it is known that the previous problem can be recast as that of minimizing the augmented function

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

subject to the constraint (2). Here, λ is a q-vector 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[2], the optimum solution x, λ must satisfy the simultaneous equations

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

---

[1] All vectors are column vectors.

[2] In Eq. (4), the gradients f_x and F_x denote n-vectors and the matrix φ_x is n x q.

2.2. Approximate Solutions. In general, the system (5) is nonlinear; consequently, approximate methods must be employed. These are of two kinds: first-order methods (such as the one discussed in subsequent sections of this thesis) and second-order methods. Here, we introduce the quantities

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

measuring the error in the constraint and the optimum condition, respectively. We observe that $P = 0$ and $Q = 0$ for the optimum solution, while $P > 0$ and $Q > 0$ for any approximation to the solution. When approximate methods are used, they must ultimately lead to values of x, λ such that

$$P(x) \le \varepsilon_1 \quad , \quad Q(x, \lambda) \le \varepsilon_2 \tag{7}$$

Alternatively, (7) can be replaced by

$$R(x, \lambda) \le \varepsilon_3 \tag{8}$$

where

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

denotes the cumulative error in the constraint and the optimum condition. Here, ε_1, ε_2, ε_3 are small, preselected numbers. Note that satisfaction of Ineq. (8) implies satisfaction of Ineq. (7), if one chooses $\varepsilon_1 = \varepsilon_2 = \varepsilon_3$.

3. Conjugate Gradient Phase

Let x denote the nominal point, $\tilde{x}$ the varied point, and Δx the displacement leading from the nominal point to the varied point. Let λ denote the Lagrange multiplier, p the present search direction, $\hat{p}$ the previous search direction, γ the directional coefficient, and α the gradient stepsize. Both p and $\hat{p}$ are n-vectors, while γ and α are scalars. With these definitions in mind, we consider the conjugate gradient algorithm represented by

$$\begin{aligned} F_x(x, \lambda) &= f_x(x) + \varphi_x(x)\lambda \\ p &= F_x(x, \lambda) + \gamma\hat{p} \\ \Delta x &= -\alpha p \\ \tilde{x} &= x + \Delta x \end{aligned} \tag{10}$$

whose form is suggested by the results of Ref. 1. For a given nominal point x, Eqs. (10) constitute a complete iteration leading to the varied point $\tilde{x}$, providing one specifies the Lagrange multiplier λ, the directional coefficient γ, and the gradient stepsize α.

3.1. Lagrange Multiplier. In accordance with the discussion of Section 1, two possible determinations of the multiplier are presented here.

Algorithms of Class I. In these algorithms, the multiplier is determined so that the error in the optimum condition (6-2) is minimized for given x (Refs. 1 and 4). Owing to the fact that

$$Q(x, \lambda) = [f_x(x) + \varphi_x(x)\lambda]^T [f_x(x) + \varphi_x(x)\lambda] \tag{11}$$

the multiplier is determined by the relation

$$Q_\lambda(x, \lambda) = 0 \tag{12}$$

which implies that

$$\varphi_x^T(x)\varphi_x(x)\lambda + \varphi_x^T(x)f_x(x) = 0 \tag{13}$$

This linear vector equation is equivalent to q linear scalar relations, in which the only unknown is the multiplier λ. The unique multiplier solving Eq. (13) is denoted by

$$\lambda = \lambda_o \tag{14}$$

Algorithms of Class II. In these algorithms, the multiplier is determined so that, at the end of any iteration, the constraint is satisfied to first order (Ref. 3). Let $\varphi(x) \neq 0$ and $\varphi(\tilde{x}) = 0$ to first order. If quasilinearization is employed, we obtain the relation

$$\varphi(x) + \varphi_x^T \Delta x = 0 \tag{15}$$

which, for convenience, is imbedded in the more general relation

$$\mu\varphi(x) + \varphi_x^T(x)\Delta x = 0 \tag{16}$$

where $\mu \geq 0$ denotes a scaling factor to be specified. If Eqs. (10-3) and (16) are combined, we see that

$$\mu\varphi(x) - \alpha\varphi_x^T(x)p = 0 \tag{17}$$

Let μ be proportional to α throughout the algorithm, that is, let

$$\mu = C\alpha \tag{18}$$

where C is a constant to be specified[3]. Then, Eq. (17) becomes

$$C\varphi(x) - \varphi_x^T(x)p = 0 \tag{19}$$

and, in the light of Eqs. (4) and (10-2), can be written as

$$\varphi_x^T(x)\varphi_x(x)\lambda + \varphi_x^T(x)f_x(x) + \gamma\varphi_x^T(x)\hat{p} - C\varphi(x) = 0 \tag{20}$$

For a given value of the constant C and for a specified value of the directional coefficient γ (see Section 3.2), this linear vector equation is equivalent to q linear scalar equations, in which the only unknown is the multiplier λ. The unique multiplier solving Eq. (20) is denoted by

$$\lambda = \lambda_* \tag{21}$$

3.2. Directional Coefficient. For both algorithms of Class I and Class II, the directional coefficient γ is determined by the relation (Refs. 1 and 3)

$$\gamma = 0 \tag{22}$$

or

$$\gamma = Q(x, \lambda_o) / Q(\hat{x}, \hat{\lambda}_o) \tag{23}$$

Equation (22) is to be employed for the first iteration of the conjugate gradient phase and means that the search direction p is identical with the gradient of the

3 If the function f, the constraint φ, and the vector x are scaled in such a way that the gradient stepsize α is 0(1), then the choice C = 1 is appropriate.

augmented function $F_x(x, \lambda)$. Equation (23) is to be employed for the remaining iterations of the conjugate gradient phase; since $\gamma \neq 0$, the search direction p is not identical with the gradient of the augmented function. In Eq. (23), x denotes the present point, $\hat{x}$ the previous point, and λ_o the solution of Eq. (13) at the present point, and $\hat{\lambda}_o$ the solution of Eq. (13) at the previous point.

3.3. Descent Properties. In the previous sections, we discussed the determination of the Lagrange multiplier λ and the directional coefficient γ for both algorithms of Class I and Class II. Prior to determining the gradient stepsize α for given values of γ and λ, we establish whether certain descent properties are satisfied. When the displacement (10-3) is employed, the first variations of the functions $F(x, \lambda)$ and $P(x)$ are given by[4]

$$\begin{aligned} \delta F(x, \lambda) &= F_x^T(x, \lambda)\Delta x = -\alpha F_x^T(x, \lambda)p \\ \delta P(x) &= 2\varphi^T(x)\varphi_x^T(x)\Delta x = -2\alpha\varphi^T(x)\varphi_x^T(x)p \end{aligned} \tag{24}$$

and, in the light of (10-2), can be rewritten as

$$\begin{aligned} \delta F(x, \lambda) &= -\alpha[Q(x, \lambda) + \gamma F_x^T(x, \lambda)\hat{p}] \\ \delta P(x) &= -2\alpha\varphi^T(x)\varphi_x^T(x)[F_x(x, \lambda) + \gamma\hat{p}] \end{aligned} \tag{25}$$

For algorithms of Class I and Class II, the second of these equations takes the particular form

$$\begin{aligned} &\underline{\text{Class I}} && \delta P(x) = -2\alpha\gamma\varphi^T(x)\varphi_x^T(x)\hat{p} \\ &\underline{\text{Class II}} && \delta P(x) = -2C\alpha P(x) \end{aligned} \tag{26}$$

[4] In the computation of the first-order change of F, the multiplier λ is held constant.

Augmented Function. For both algorithms of Class I and Class II, the first variation of the augmented function (25-1) is negative providing

$$Q(x, \lambda) + \gamma F_x^T(x, \lambda)\hat{p} > 0 \tag{27}$$

For the first iteration of the conjugate gradient phase ($\gamma = 0$), Ineq. (27) is satisfied, and the descent property $\delta F(x, \lambda) < 0$ holds. Therefore, for α sufficiently small, the decrease of the augmented function is guaranteed. For subsequent iteration ($\gamma \neq 0$), Ineq. (27) may or may not be satisfied, and the descent property $\delta F(x, \lambda) < 0$ may or may not hold. Whenever Ineq. (27) is violated [that is, whenever $\delta F(x, \lambda) > 0$], the conjugate gradient phase must be interrupted, and the restoration phase must be started.

Constraint Error. For algorithms of Class I, Eq. (26-1) applies. For the first iteration of the conjugate gradient phase ($\gamma = 0$), the first variation of the constraint error vanishes. For subsequent iterations ($\gamma \neq 0$), the first variation of the constraint error is negative only if

$$\varphi^T(x)\varphi_x^T(x)\hat{p} > 0 \tag{28}$$

Since Ineq. (28) may or may not be satisfied, the descent property $\delta P(x) < 0$ may or may not hold. For the above considerations, the restoration phase is indispensable to the stability of algorithms of Class I.

For algorithms of Class II, Eq. (26-2) applies and shows that $\delta P(x) < 0$, regardless of the value assigned to the directional coefficient γ. Therefore, for α sufficiently small, the decrease in the constraint error is guaranteed. For the above considerations, the restoration phase is not indispensable to the stability of the algorithms of Class II, but it is desirable in order to ensure quadratic terminal convergence.

3.4. Gradient Stepsize. The descent properties established in the previous section are instrumental in the determination of the optimum gradient stepsize for given values of the multiplier λ and the directional coefficient γ. If Eqs. (10-3) and (10-4) are combined, the position vector at the end of the conjugate gradient step becomes

$$\tilde{x} = x - \alpha p \tag{29}$$

where p is known through Eq. (10-2). This is a one-parameter family of varied points $\tilde{x}$, for which the augmented function and the constraint error are functions of the form

$$F(\tilde{x}, \lambda) = F(x - \alpha p, \lambda) = F(\alpha) \quad , \quad P(\tilde{x}) = P(x - \alpha p) = P(\alpha) \tag{30}$$

Along the straight line defined by Eq. (29), the augmented function admits the derivative

$$F_{\alpha}(\alpha) = -F_x^T(\tilde{x}, \lambda)p = -F_x^T(\tilde{x}, \lambda)\,[F_x(x, \lambda) + \gamma\hat{p}] \tag{31}$$

which, at $\alpha = 0$, becomes

$$F_{\alpha}(0) = -[Q(x, \lambda) + \gamma F_x^T(x, \lambda)\hat{p}] \tag{32}$$

If Ineq. (27) is satisfied, the descent property on $F(\alpha)$ holds, and the search for the optimum gradient stepsize can be initiated. If Ineq. (27) is violated, the descent property on $F(\alpha)$ does not hold; the search direction p must be discarded, the conjugate gradient phase must be interrupted, and the restoration phase must be started.

Now, we assume that $F_\alpha(0) < 0$ and that a minimum $F(\alpha)$ exists. Then we employ some one-dimensional search scheme (for instance, quadratic interpolation, cubic interpolation, or quasilinearization) to determine the value of α for which

$$F_\alpha(\alpha) = 0 \tag{33}$$

Ideally, this procedure should be used iteratively until the modulus of the slope satisfied any of the following inequalities:

$$|F_\alpha(\alpha)| \leq \epsilon_4 \quad \text{or} \quad |F_\alpha(\alpha)| \leq \epsilon_5 \; |F_\alpha(0)| \tag{34}$$

where ϵ_4 and ϵ_5 are small, preselected numbers. Of couse, the value of α satisfying Ineq. (34) must be such that

$$F(\alpha) < F(0) \tag{35}$$

In practice, the solution of (33) must be subordinated not only to (35) but also to certain additional inequalities, designed to counteract the effects arising from these undesirable situations: (a) the minimum of the function $F(\alpha)$ occurs for such a large gradient stepsize that the constraint error $P(\alpha)$ at the end of the iteration is excessive; and (b) the function $F(\alpha)$ decreases nonotonically versus the gradient stepsize. With these ideas in mind, Ineq. (35) must be supplemented by

$$P(\alpha) < kP(0) \qquad \text{if} \qquad P(0) \geq P_* \tag{36}$$

$$P(\alpha) < P_* \qquad \text{if} \qquad P(0) < P_* \tag{37}$$

where P_* and k are prescribed constants. Specifically, $k > 1$ for algorithms of Class I and $k = 1$ for algorithms of Class II.

In closing, we note that Ineq. (36) applies only to algorithms with incomplete restoration, namely, Algorithms (β) and (δ). On the other hand, Ineq. (37) applies to all of the previous algorithms, regardless of whether complete restoration or incomplete restoration is used.

3.5. Convergence Properties. If the function f(x) is quadratic, if the constraint $\varphi(x)$ is linear, and if the starting point x_s is such that $\varphi(x_s) = 0$, then algorithms of Class I and algorithms of Class II become identical. They produce the same sequence of points and converge to the solution in at most $N_* = n-q$ iterations. If any of the above conditions is violated, the quadratic convergence property does not hold. However, quadratic *terminal* convergence can be achieved if a suitable restoration phase is inserted in the algorithm (see Section 4).

4. Restoration Phase

Let x denote the nominal point, $\tilde{x}$ the varied point, and Δx the displacement leading from the nominal point to the varied point. Let σ denote the Lagrange multiplier, p the search direction, and μ the restoration stepsize. Here, σ is a q-vector, p an n-vector, and μ a scalar. With these definitions in mind, we consider the restoration algorithm represented by

$$\begin{aligned} &\varphi_x^T(x)\varphi_x(x)\sigma - \varphi(x) = 0 \\ &p = \varphi_x(x)\sigma \\ &\Delta x = -\mu p \\ &\tilde{x} = x + \Delta x \end{aligned} \tag{38}$$

whose form is suggested by Refs. 1-2. For a given nominal point x, Eqs. (38) represent a complete iteration leading to the varied point $\tilde{x}$, providing one specifies the restoration stepsize μ.

4.1. Descent Property. Prior to determining the restoration stepsize μ, we establish a basic descent property. When the displacement (38-3) is employed, the first variation of the function $P(x)$ is given by

$$\delta P(x) = 2\varphi^T(x)\varphi_x^T(x)\Delta x = -2\mu\varphi^T(x)\varphi_x^T(x)p \tag{39}$$

and, in the light of (6-1), (38-1), (38-2), can be written as

$$\delta P(x) = -2\mu P(x) \tag{40}$$

Since μ is positive and $P(x)$ is positive, Eq. (40) shows that $\delta P(x) < 0$. Therefore, for μ sufficiently small, the decrease of the constraint error is guaranteed.

4.2. Restoration Stepsize. The descent property established in the previous section is instrumental in determining the optimum restoration stepsize. If Eqs. (38-3) and (38-4) are combined, the position vector at the end of a restoration step becomes

$$\tilde{x} = x - \mu p \tag{41}$$

where p is known through Eq. (38-2) . This is a one-parameter family of varied points $\tilde{x}$, for which the constraint error is a function of the form

$$P(\tilde{x}) = P(x - \mu p) = P(\mu) \tag{42}$$

Along the straight line defined by Eq. (41), the constraint error admits the derivative

$$P_{\mu}(\mu) = -2\varphi^{T}(\tilde{x})\varphi_{x}^{T}(\tilde{x})p \tag{43}$$

which, at $\mu = 0$, becomes

$$P_{\mu}(0) = -2P(x) \tag{44}$$

a results consistent with (40). Since $P_{\mu}(0) < 0$, the search for the optimum restoration stepsize can be initiated.

Assuming that a minimum of $P(\mu)$ exists, we employ some one-dimensional search scheme (for instance, quadratic interpolation, cubic interpolation, or quasilinearization) to determine the value of μ for which

$$P_\mu(\mu) = 0 \tag{45}$$

Ideally, this procedure should be used iteratively until the modulus of the slope satisfies any of the following inequalities:

$$|P_\mu(\mu)| \le \epsilon_6 \quad \text{or} \quad |P_\mu(\mu)| \le \epsilon_7 \; |P_\mu(0)| \tag{46}$$

where ϵ_6 and ϵ_7 are small, preselected numbers. Of course, the value of μ satisfying Ineq. (46) must be such that

$$P(\mu) < P(0) \tag{47}$$

Since a rigorous search might take excessive computer time, we propose here an alternate procedure. We observe that, for a linear constraint, Eq. (45) is solved by $\mu = 1$. This result and the descent property of the previous section suggest replacing the rigorous search by a bisection process on μ starting from $\mu = 1$. Specifically, we first assign the value $\mu = 1$ to the restoration stepsize and verify Ineq. (47). If Ineq. (47) is satisfied, the iteration is complete. If Ineq. (47) is violated, μ is bisected several times until satisfaction of Ineq. (47) occurs. This is guaranteed by the descent property of the previous section.

Remark 4.1. The restoration phase is important for two reasons: (i) it gives stability to algorithms of Class I: for these algorithms, the descent property on the function $P(\alpha)$ is guaranteed during the conjugate gradient phase; and (ii) it accelerates the convergence of both algorithms of Class I and Class II; if the function $f(x)$ is quadratic, if the constraint $\varphi(x)$ is linear, and if the starting point x_s is such that $\varphi(x_s) \neq 0$, then convergence to the solution in at most $N_* = 1+n-q$

iterations is possible if the restoration phase precedes the conjugate gradient phase.

Remark 4.2. Algorithms (α) and (γ) are characterized by complete restoration: the restoration phase involves several iterations, the number of which is determined through satisfaction of Ineq. (7-1). In turn, Algorithms (β) and (δ) are characterized by incomplete restoration: the restoration phase involves one iteration.

5. Summary of Conjugate Gradient-Restoration Algorithms

The conjugate gradient-restoration algorithms discussed here involve the alternate succession of conjugate gradient phases and restoration phases. A summary of these phases is given below.

5.1. Conjugate Gradient Phase. For algorithms of Class I, the conjugate gradient phase involves n-q iterations, each of which is represented by the following equations:

$$\varphi_x^T(x)\varphi_x(x)\lambda_o + \varphi_x^T(x)f_x(x) = 0$$

$$F_x(x, \lambda_o) = f_x(x) + \varphi_x(x)\lambda_o$$

$$Q(x, \lambda_o) = F_x^T(x, \lambda_o)F_x(x, \lambda_o)$$

$$\gamma = Q(x, \lambda_o) / Q(\hat{x}, \hat{\lambda}_o) \tag{48}$$

$$p = F_x(x, \lambda_o) + \gamma\hat{p}$$

$$\Delta x = -\alpha p$$

$$\tilde{x} = x + \Delta x$$

For the first iteration, Eq. (48-4) is bypassed and is replaced by $\gamma = 0$.

For algorithms of Class II, the conjugate gradient phase involves n-q iterations, each of which is represented by the following equations:

$$\varphi_x^T(x)\varphi_x(x)\lambda_o + \varphi_x^T(x)f_x(x) = 0$$

$$F_x(x, \lambda_o) = f_x(x) + \varphi_x(x)\lambda_o$$

$$Q(x, \lambda_o) = F_x^T(x, \lambda_o)F_x(x, \lambda_o)$$

$$\gamma = Q(x, \lambda_o)/Q(\hat{x}, \hat{\lambda}_o)$$

$$\varphi_x^T(x)\varphi_x(x)\lambda_* + \varphi_x^T(x)f_x(x) + \gamma\varphi_x^T(x)\hat{p} - C\varphi(x) = 0 \qquad (49)$$

$$F_x(x, \lambda_*) = f_x(x) + \varphi_x(x)\lambda_*$$

$$p = F_x(x, \lambda_*) + \gamma\hat{p}$$

$$\Delta x = -\alpha p$$

$$\tilde{x} = x + \Delta x$$

For the first iteration, Eq. (49-1) through (49-4) are bypassed and replaced by $\gamma = 0$.

Search Technique. The search for the optimum gradient stepsize is made on the augmented function $F(\tilde{x}, \lambda) = F(\alpha)$, where $\lambda = \lambda_o$ for algorithms of Class I and $\lambda = \lambda_*$ for algorithms of Class II. First, one checks the sign of the derivative

$$F_\alpha(0) = -[Q(x, \lambda) + \gamma F_x^T(x, \lambda)\hat{p}] \qquad (50)$$

If $F_\alpha(0) < 0$, the search for the optimum gradient stepsize is initiated. If $F_\alpha(0) \geq 0$, the conjugate gradient phase is interrupted, and the restoration phase is started.

Whenever $F_\alpha(0) < 0$, the search is terminated when any of the following inequalities is satisfied:

$$|F_\alpha(\alpha)| \le \epsilon_4 \quad \text{or} \quad |F_\alpha(\alpha)| \le \epsilon_5\, |F_\alpha(0)| \tag{51}$$

and must be subordinated to the further inequalities

$$F(\alpha) < F(0) \tag{52}$$

and

$$P(\alpha) < kP(0) \quad \text{if} \quad P(0) \ge P_* \tag{53}$$

$$P(\alpha) < P_* \quad \text{if} \quad P(0) < P_* \tag{54}$$

where P_* and k are prescribed constants. Specifically, $k > 1$ for algorithms of Class I and $k = 1$ for algorithms of Class II.

5.2. Restoration Phase. Depending on the restoration type (incomplete or complete), the restoration phase involves one or several iterations, each of which is represented by the following equations:

$$\begin{aligned} &\varphi_x^T(x)\varphi_x(x)\sigma - \varphi(x) = 0 \\ &p = \varphi_x(x)\sigma \\ &\Delta x = -\mu p \\ &\tilde{x} = x + \Delta x \end{aligned} \tag{55}$$

For every iteration, the search for the restoration stepsize is made on the constraint error $P(\tilde{x}) = P(\mu)$. Specifically, one employs a bisection process on μ (starting from $\mu = 1$) until satisfaction of the following inequality occurs:

$$P(\mu) < P(0) \tag{56}$$

5.3. Special Conditions. In this section, special conditions relevant to the computer implementation of conjugate gradient-restoration algorithms are presented.

Starting Condition. The algorithms can be started from any nominal point x_s, regardless of whether $\varphi(x_s) = 0$ or $\varphi(x_s) \neq 0$.

Initial Phase. The algorithms are started with a restoration phase if $P(x_s) > \epsilon_1$ and a conjugate gradient phase if $P(x_s) \leq \epsilon_1$.

Restoration Phase: Stopping Condition. For Algorithms (α) and (γ), the restoration phase is stopped when Ineq. (7-1) is satisfied. For Algorithms (β) and (δ), the restoration phase is stopped after a single iteration.

Restoration Phase: Bypassing Condition. Usually, a complete cycle includes a restoration phase and a conjugate gradient phase. However, if at the beginning of the restoration phase, Ineq. (7-1) is met, the restoration phase is bypassed, and the conjugate gradient phase is started directly.

Conjugate Gradient Phase: Stopping Condition. The conjugate gradient phase must be stopped under the following conditions: (a) every n-q iterations, (b) if $F_\alpha(0) \geq 0$, where $F_\alpha(0)$ is given by Eq. (50), and (c) if the gradient stepsize minimizing the augmented funtion $F(\alpha)$ cannot be employed due to violation of Ineq. (53)-(54).

Conjugate Gradient-Restoration Algorithm: Stopping Condition. A conjugate gradient-restoration algorithm is stopped when Ineq. (7) is satisfied or Ineq. (8) is satisfied.

5.4. Convergence Properties. If the function $f(x)$ is quadratic and if the constraint $\varphi(x)$ is linear, then all of the previous algorithms become identical, regardless of whether they are of Class I or Class II, and regardless of whether they are of type (α), (β), (γ), (δ). They produce the same sequence of points and converge to the solution in the same number of iterations. This number of iterations is at most $N_* = n-q$ if the starting point x_s is such that $\varphi(x_s) = 0$ and at most $N_* = 1+n-q$ if the starting point x_s is such that $\varphi(x_s) \neq 0$.

6. Experimental Conditions and Numerical Examples

In order to illustrate the theory, five numerical examples were developed using a Burroughs B-5500 computer and double-precision arithmetic. The algorithms were programmed in FORTRAN IV. The constant C was specified to be C = 1; the constant P_* was given the value $P_* = 10$; and the constant k was selected to be k = 10 for algorithms of Class I and k = 1 for algorithms of Class II.

Concerning the conjugate gradient phase, the one-dimensional search on the function $F(\alpha)$ was done in accordance with Section 5.1; a modification of quasilinearization was employed (see Appendix); the stopping condition for the one-dimensional search was

$$F_\alpha^2(\alpha) \le F_\alpha^2(0) \times 10^{-6} \tag{57}$$

and the gradient stepsize α was subordinated to satisfaction of Ineq. (52)-(54). Concerning the restoration phase, the one-dimensional search on the function $P(\mu)$ was done in accordance with Section 5.2.

Convergence was defined as follows:

$$R(x, \lambda) \le 10^{-12} \tag{58}$$

and the number of iterations for convergence N_* was recorded[5]. Incidentally, satisfaction of Ineq. (58) implies that[6]

$$P(x) \le 10^{-12} \quad , \quad Q(x, \lambda) \le 10^{-12} \tag{59}$$

[5] The number N_* includes both the iterations of the conjugate gradient phase and the iterations of the restoration phase.

[6] Inequality (59-1) constitutes the bypassing condition for the restoration phase.

Conversely, nonconvergence was defined by means of the inequalities

$$\text{(a)} \qquad N \geq 1000 \tag{60}$$

or

$$\text{(b)} \qquad N_s \geq 20 \tag{61}$$

Here, N is the iteration number and N_s is the number of bisections of the stepsize required to satisfy Ineqs. (52)-(54) or Ineq. (56).

In the numerical experiments, the effect of the length of the conjugate gradient phase ΔN on the convergence characteristics of the algorithms was studied, where ΔN denotes the number of iterations of the conjugate gradient phase. Three values of ΔN were considered, namely, $\Delta N = 1$, $\Delta N = n-q$, and $\Delta N = n$. Note that, for $\Delta N = 1$, the present algorithm reduces to the ordinary gradient-restoration algorithm.

Example 6.1. Consider the problem of minimizing the function[7]

$$f = (x - y)^2 + (y + z - 2)^2 + (u - 1)^2 + (w - 1)^2 \tag{62}$$

subject to the constraints

$$x + 3y = 0 \;, \quad z + u - 2w = 0 \;, \quad y - w = 0 \tag{63}$$

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

$$x = -0.7674 \;, \quad y = 0.2558 \;, \quad z = 0.6279 \;, \quad u = -0.1162 \;, \quad w = 0.2558 \tag{64}$$

and

$$\lambda_1 = 2.0465 \;, \quad \lambda_2 = 2.2325 \;, \quad \lambda_3 = -5.9534 \tag{65}$$

[7] For simplicity, the symbols employed in the examples denote scalar quantities.

The nominal point chosen to start the algorithm is the point of coordinates

$$x = y = z = u = w = 2 \tag{66}$$

not consistent with (63).

Example 6.2. Consider the problem of minimizing the function

$$f = (x - y)^2 + (y - z)^4 \tag{67}$$

subject to the constraint

$$x(1 + y^2) + z^4 - 3 = 0 \tag{68}$$

This function admits the relative minimum $f = 0$ at the point defined by

$$x = y = z = 1 \tag{69}$$

and

$$\lambda_1 = 0 \tag{70}$$

The nominal point chosen to start the algorithm is the point of coordinates

$$x = y = z = 2 \tag{71}$$

not consistent with (68).

Example 6.3. Consider the problem of minimizing the function

$$f = (x - 1)^2 + (x - y)^2 + (y - z)^4 \tag{72}$$

subject to the constraint

$$x(1 + y^2) + z^4 - 4 - 3\sqrt{2} = 0 \tag{73}$$

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

$$x = 1.1048 \quad , \quad y = 1.1966 \quad , \quad z = 1.5352 \tag{74}$$

and

$$\lambda_1 = -0.1072 \times 10^{-1} \tag{75}$$

The nominal point chosen to start the algorithm is the point of coordinates

$$x = y = z = 2 \tag{76}$$

not consistent with (73).

<u>Example 6.4</u>. Consider the problem of minimizing the function

$$f = (x - 1)^2 + (x - y)^2 + (z - 1)^2 + (u - 1)^4 + (w - 1)^6 \tag{77}$$

subject to the constraints

$$ux^2 + \sin(u - w) - 2\sqrt{2} = 0 \quad , \quad y + z^4 u^2 - 8 - \sqrt{2} = 0 \tag{78}$$

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

$$x = 1.1661 \quad , \quad y = 1.1821 \quad , \quad z = 1.3802 \quad , \quad u = 1.5060 \quad , \quad w = 0.6109 \tag{79}$$

and

$$\lambda_1 = -0.8553 \times 10^{-1} \quad , \quad \lambda_2 = -0.3187 \times 10^{-1} \tag{80}$$

The nominal point chosen to start the algorithm is the point of coordinates

$$x = y = z = u = w = 2 \tag{81}$$

not consistent with (78).

Example 6.5. Consider the problem of minimizing the function

$$f = (x - 1)^2 + (x - y)^2 + (y - z)^2 + (z - u)^4 + (u - w)^4 \tag{82}$$

subject to the constraints

$$x + y^2 + z^3 - 2 - 3\sqrt{2} = 0 \;,\; y - z^2 + u + 2 - 2\sqrt{2} = 0 \;,\; xw - 2 = 0 \tag{83}$$

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

$$x = 1.1911 \;,\; y = 1.3626 \;,\; z = 1.4728 \;,\; u = 1.6350 \;,\; w = 1.6790 \tag{84}$$

and

$$\lambda_1 = -0.3882 \times 10^{-1} \;,\; \lambda_2 = -.1672 \times 10^{-1} \;,\; \lambda_3 = -0.2879 \times 10^{-3} \tag{85}$$

The nominal point chosen to start the algorithm is the point of coordinates

$$x = y = z = u = w = 2 \tag{86}$$

not consistent with (83).

7. Numerical Results and Conclusions

For the previous examples and experimental conditions, the conjugate gradient-restoration algorithms of Class I and Class II were tested in versions (α), (β), (γ), (δ), all of which include a restoration phase. For completeness, the algorithm of Class II was also tested in version (ϵ), which does not include a restoration phase. The numerical results are given in Tables 2-11, where the number of iterations for convergence N_* is shown versus the parameter ΔN of the conjugate gradient phase. From the tables, the following conclusions arise:

(a) For the linear-quadratic Example 6.1, Algorithms (I-α) through (I-δ) and Algorithms (II-α) through (II-δ) behave identically, as predicted by the theory. For these algorithms, the quadratic convergence is verified, while this is not the case with Algorithm (II-ϵ). This clearly establishes the importance of the restoration phase.

(b) For the nonlinear-nonquadratic Examples 6.2 through 6.5, the algorithms do not behave identically. A detailed analysis is given below.

(c) Concerning the conjugate gradient phase, algorithms characterized by $\Delta N = n-q$ generally require a smaller number of iterations than algorithms characterized by $\Delta N = 1$ or $\Delta N = n$. Also, algorithms of Class II generally require a smaller number of iterations than algorithms of Class I.

(d) Concerning the restoration phase, Algorithms (β) and (δ) generally require a smaller number of iterations than Algorithms (α) and (γ), respectively. Also, Algorithms (γ) and (δ) generally require a smaller number of iterations than Algorithms (α) and (β), repsectively. Therefore, for fast convergence, incomplete restoration is to be preferred to complete restoration and infrequent restoration is to be preferred to frequent restoration.

(e) Among the algorithms tested, Algorithm (II-δ) with $\Delta N = n-q$ is the best. This algorithm is characterized by the multiplier being determined so as to satisfy the constraint to first order. It is also characterized by incomplete and infrequent restoration, that is, a single restoration step prior to every $\Delta N = n-q$ gradient steps.

(f) It is of interest to compare Algorithm (II-δ) with Algorithm (I-α), which is the sequential conjugate gradient-restoration algorithm of Ref. 1 and is characterized by complete and frequent restoration. For Examples 6.2, 6.3, 6.4, Algorithm (II-δ) converges in a number of iterations which is about one half that of Algorithm (I-α). For Example 6.5, Algorithm (II-δ) converges in a number of iterations which is about two thirds that of Algorithm (I-α).

Table 2. Number of iterations at convergence N_* for Example 6.1, algorithms of Class I.

ΔN	(α)	(β)	(γ)	(δ)
1	11	11	11	11
n-q=2	3	3	3	3
n = 5	3	3	3	3

Table 3. Number of iterations at convergence N_* for Example 6.1, algorithms of Class II.

ΔN	(α)	(β)	(γ)	(δ)	(ϵ)
1	11	11	11	11	24
n-q=2	3	3	3	3	27
n = 5	3	3	3	3	34

Table 4. Number of iterations at convergence N_* for Example 6.2, algorithms of Class I.

ΔN	(α)	(β)	(γ)	(δ)
1	>1000	>1000	>1000	>1000
n-q = 2	32	25	25	18
n = 3	46	17	28	23

Table 5. Number of iterations at convergence N_* for Example 6.2, algorithms of Class II.

ΔN	(α)	(β)	(γ)	(δ)	(ϵ)
1	>1000	599	>1000	599	>1000
n-q = 2	27	22	27	16	>1000
n = 3	25	25	27	20	>1000

Table 6. Number of iterations at convergence N_* for Example 6.3, algorithms of Class I.

ΔN	(α)	(β)	(γ)	(δ)
1	40	34	40	34
$n-q = 2$	21	15	18	15
$n = 3$	23	14	18	19

Table 7. Number of iterations at convergence N_* for Example 6.3, algorithms of Class II.

ΔN	(α)	(β)	(γ)	(δ)	(ϵ)
1	40	34	40	34	35
$n-q = 2$	21	14	17	12	36
$n = 3$	23	13	17	15	26

Table 8. Number of iterations at convergence N_* for Example 6.4, algorithms of Class I.

ΔN	(α)	(β)	(γ)	(δ)
1	56	49	56	49
n-q = 3	27	16	19	21
n = 5	30	18	23	31

Table 9. Number of iterations at convergence N_* for Example 6.4, algorithms of Class II.

ΔN	(α)	(β)	(γ)	(δ)	(ϵ)
1	56	41	56	41	39
n-q = 3	26	14	19	13	29
n = 5	31	14	22	16	22

Table 10. Number of iterations at convergence N_* for Example 6.5, algorithms of Class I.

ΔN	(α)	(β)	(γ)	(δ)
1	16	11	16	11
n-q = 2	15	11	13	11
n = 5	16	13	19	21

Table 11. Number of iterations at convergence N_* for Example 6.5, algorithms of Class II.

ΔN	(α)	(β)	(γ)	(δ)	(ϵ)
1	16	21	16	21	17
n-q = 2	14	11	13	10	17
n = 5	16	11	12	13	31

8. Appendix: Search for the Optimum Gradient Stepsize

In this section, we describe the one-dimensional search employed in order to determine the optimum gradient stepsize. The search is carried out on the augmented function

$$F(\tilde{x}, \lambda) = F(\alpha) \tag{87}$$

along the straight line defined by

$$\tilde{x} = x - \alpha p \tag{88}$$

Here, $\lambda = \lambda_o$ for algorithms of Class I and $\lambda = \lambda_*$ for algorithms of Class II. Before the search is started, one must check whether the decrease of the augmented function is guaranteed. To this effect, one computes the derivative

$$F_\alpha(0) = -[Q(x, \lambda) + \gamma F_x^T(x, \lambda)\hat{p}] \tag{89}$$

and verifies whether $F_\alpha(0) < 0$. If this is the case, one starts the search; otherwise, if $F_\alpha(0) \geq 0$, one abandons the search and starts the restoration phase.

Owing to the analytical nature of the examples considered here, modified quasilinearization is employed; however, in a more realistic situation, one would use cubic interpolation or equivalent first-order technique. When quasilinearization is employed, evaluation of the derivatives

$$F_\alpha(\alpha) = -F_x^T(\tilde{x}, \lambda)p \quad , \quad F_{\alpha\alpha}(\alpha) = p^T F_{xx}(\tilde{x}, \lambda)p \tag{90}$$

is in order.

Let α_o denote the nominal stepsize, α the varied stepsize, and $\Delta\alpha_o$ the increment leading from α_o to α, so that

$$\alpha = \alpha_o + \Delta\alpha_o \tag{91}$$

If modified quasilinearization is employed, one has

$$\Delta\alpha_o = -\rho F_\alpha(\alpha_o)/|F_{\alpha\alpha}(\alpha_o)| \tag{92}$$

where the scale factor ρ is such that

$$0 \le \rho \le 1 \tag{93}$$

With the above considerations in mind, the sequence of computation is as follows:[8]

(a) Select a nominal stepsize α_o; compute the derivatives $F_\alpha(\alpha_o)$ and $F_{\alpha\alpha}(\alpha_o)$ with (90).

(b) For any given ρ, compute $\Delta\alpha_o$ with (92) and α with (91); select ρ so that the inequality

$$F(\alpha) < F(\alpha_o) \tag{94}$$

is satisfied; to this effect, employ a bisection process on ρ starting from $\rho = 1$.

(c) Once ρ and α are known, check the inequality

$$\begin{aligned} &P(\alpha) < kP(0) \quad \text{if} \quad P(0) \ge P_* \\ &P(\alpha) < P_* \qquad \text{if} \quad P(0) < P_* \end{aligned} \tag{95}$$

where k = 10 for algorithms of Class I and k = 1 for algorithms of Class II.

---

[8] In order to start the search, one sets $\alpha_o = 0$.

(d) If Ineq. (95) is satisfied, return to (a) and continue the search until the stopping condition

$$F_{\alpha}^{2}(\alpha) \leq F_{\alpha}^{2}(0) \times 10^{-6} \tag{96}$$

is satisfied; with α known, the search is completed, the state x is updated with (88), and the next iteration of the conjugate gradient phase is started.

(e) If Ineq. (95) is violated, a further bisection of the scale factor ρ is performed until Ineq. (95) is satisfied; with α known, the search is completed disregarding satisfaction of Ineq. (96), and the state x is updated with (88); then, one interrupts the conjugate gradient phase and starts the restoration phase.

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., HEIDEMAN, J.C., and DAMOULAKIS, J.N., The Restoration of Constraints in Holonomic and Nonholonomic Problems, Journal of Optimization Theory and Applications, Vol. 3, No. 5, 1969.

3. MIELE, A., HEIDEMAN, J.C., and LEVY, A.V., Combined Conjugate Gradient-Restoration Algorithm for Mathematical Programming Problems, Ricerche Di Automatica, Vol. 2, No. 2, 1971.

4. MIELE, A., and LEVY, A.V., Modified Quasilinearization and Optimal Initial Choice of the Multipliers, Part 1, Mathematical Programming Problems, Journal of Optimization Theory and Applications, Vol. 6, No. 5, 1970.