BOUNDARY PROPERTIES OF PENALTY FUNCTIONS FOR CONSTRAINED MINIMIZATION

F. A. LOOTSMA

BOUNDARY PROPERTIES OF PENALTY FUNCTIONS FOR CONSTRAINED MINIMIZATION

PROEFSCHRIFT

TER VERKRIJGING VAN DE GRAAD VAN DOCTOR IN DE TECHNISCHE WETENSCHAPPEN AAN DE TECHNISCHE HOGESCHOOL EINDHOVEN OP GEZAG VAN DE RECTOR MAGNIFICUS PROF. DR. IR. A. A. TH. M. VAN TRIER, HOOGLERAAR IN DE AFDELING DER ELEKTROTECHNIEK, VOOR EEN COMMISSIE UIT DE SENAAT TE VERDEDIGEN OP VRIJDAG 15 MEI 1970 DES NAMIDDAGS TE 4 UUR

DOOR

FREERK AUKE LOOTSMA

GEBOREN TE MIDLUM

DIT PROEFSCHRIFT IS GOEDGEKEURD DOOR DE PROMOTOR
PROF. DR. J. F. BENDERS

Aan de nagedachtenis van mijn vader
Aan mijn moeder
Aan Riekje

De onderzoekingen in dit proefschrift beschreven zijn verricht in het Natuurkundig Laboratorium van de N.V. Philips' Gloeilampenfabrieken te Eindhoven. De directie van dit laboratorium ben ik zeer erkentelijk voor de mij geboden gelegenheid dit werk uit te voeren en in deze vorm te publiceren.

Abstract

This monograph is concerned with a number of penalty-function techniques for solving a constrained-minimization or nonlinear-programming problem. These techniques are designed to take into account the constraints of a minimization problem or, since almost none of the problems arising in practice have interior minima, to approach the boundary of the constraint set in a specifically controlled manner. The monograph starts therefore with a classification of penalty functions according to their behaviour in the neighbourhood of that boundary. Appropriate convexity and differentiability conditions are imposed on the problem under consideration. Furthermore, certain uniqueness conditions involving the Jacobian matrix of the Kuhn–Tucker relations are satisfied by assumption. This implies that the problem has a unique minimum $\bar{x}$ with a unique vector $\bar{u}$ of associated Lagrangian multipliers. Under these conditions the minimizing trajectory generated by a mixed penalty-function technique can be expanded in a Taylor series about $(\bar{x},\bar{u})$. This provides, as an important numerical application, a basis for extrapolation towards $(\bar{x},\bar{u})$. The series expansion is always one in terms of the controlling parameter independently of the behaviour of the mixed penalty function at the boundary of the constraint set. Next, there is the intriguing question of whether some penalty functions are easier or harder to minimize than other ones. Accordingly, the condition number of the principal Hessian matrix of a penalty function is studied. It comes out that the condition number varies with the inverse of the controlling parameter, independently of the behaviour of the mixed penalty function at the boundary of the constraint set. The parametric penalty-function techniques just named can be modified into methods which do not explicitly operate with a controlling parameter. They may be considered as penalty-function techniques adjusting the controlling parameter automatically. It is established how the rate of convergence of these methods depends on the vector $\bar{u}$ of Lagrangian multipliers associated with $\bar{x}$, on the boundary properties of a penalty function, on a weight factor p attached to the objective function and on a relaxation factor ϱ. The method of centres is a remarkable exception: its rate of convergence depends on the number of active constraints at $\bar{x}$, and on p and ϱ. The computational advantages and disadvantages of the penalty-function techniques treated in the monograph are discussed. There is an appendix presenting an ALGOL 60 procedure for constrained minimization via a mixed parametric first-order penalty function.

CONTENTS

1. INTRODUCTION

1.1. Constrained minimization via penalty functions

The constrained-minimization problem to be considered in this thesis is defined as

$$\left.\begin{array}{c}\text{minimize } f(x) \text{ subject to the constraints} \\ g_i(x) \geqslant 0;\ i = 1, \ldots, m,\end{array}\right\} \tag{1.1.1}$$

where f, $g_1, \ldots, g_m$ denote real-valued functions of a vector x in the n-dimensional vector space E_n. There is an extensive literature on this problem (alternatively referred to as nonlinear-programming problem) and a large number of methods for solving it have been proposed in the last two decades. We shall here be dealing with methods which reduce the computational process to *unconstrained minimization* of a *penalty function* combining in a particular way the objective function f, the constraint functions $g_1, \ldots, g_m$ and possibly one or more controlling parameters. Surveying the literature one can distinguish two classes of penalty-function techniques both of which have been referred to by expressive names. The *interior-point methods* operate in the interior R^o of the constraint set

$$R = \{x \mid g_i(x) \geqslant 0;\ i = 1, \ldots, m\}. \tag{1.1.2}$$

The *exterior-point methods*, on the other hand, present an approach to a minimum solution of (1.1.1) from outside the constraint set.

There are three interior-point methods that have attracted considerable theoretical and computational attention. First, there is the *logarithmic-programming method*, originally proposed by Frisch (1955). It was further developed by Parisot (1961) to solve linear-programming problems, and later on the present author (1967, 1968a) gave a detailed treatment of the method as a tool for solving nonlinear problems. Second, we find the *sequential unconstrained minimization technique* (SUMT). It was originally suggested by Carroll (1961) and further developed by Fiacco and McCormick (1964a, 1964b, 1966), Pomentale (1965), and Stong (1965). It is tending to be abandoned in favour of logarithmic programming, as appears from recent work of Fiacco and McCormick (1968). Last, there is an interior-point method described by Kowalik (1966), Box, Davies and Swann (1969), and Fletcher and McCann (1969).

The exterior-point methods have a somewhat longer history. The first suggestion here was given by Courant (1943). Further developments came from Ablow and Brigham (1955), Camp (1955), Butler and Martin (1962), Pietrzykowski (1962), Fiacco and McCormick (1967a), and Beltrami (1967, 1969a).

They were mainly concerned with a penalty function which is here referred to as the *quadratic loss function.* A more general treatment of the exterior-point methods was presented by Zangwill (1967) and Roode (1968).

Interior- and exterior-point methods have particular advantages and suffer from particular disadvantages that will be explained later on. Accordingly, combinations of these methods have been designed. The first ideas came from Fiacco and McCormick (1966) who proposed a penalty function for incorporating the inequalities as well as the equality constraints of a problem. *Mixed* penalty functions have independently been studied by Fiacco (1967), by the author (1968b), and by Fiacco and McCormick (1968).

The appearance of controlling parameters in a penalty function poses the numerical question of how to choose appropriate values for them and how to use the information gathered during the computational process. One has to compromise between the desire for rapid convergence and the necessity to avoid minimization of extremely steep-valleyed penalty functions, which may cause all kinds of numerical difficulties. Acceleration of the convergence has been obtained by extrapolation, which is generally a powerful tool for approximating the limit of an infinitesimal process; we may, for instance, refer to Laurent (1963), Bulirsch (1964), Bulirsch and Stoer (1964, 1966), and Veltkamp (1969). In the field of penalty-function techniques a basis for extrapolation (the Taylor series expansion of the minimizing trajectory about a minimum solution) was first derived by Fiacco and McCormick (1966) for SUMT, later on by the author (1968a, 1968b) for logarithmic programming and the mixed penalty-function techniques.

Murray (1967) introduced the question of conditioning of a penalty function in order to compare some interior- and exterior-point penalty functions. This idea has recently been generalized by the author (1969) to study how rapidly, for various methods, a certain condition number varies with the controlling parameter.

An interesting development was initiated by Rosenbrock (1960) and continued by Huard (1964) who proposed the *method of centres.* It has been explored, theoretically and computationally, by Faure and Huard (1965, 1966), Bui Trong Lieu and Huard (1966), Huard (1967, 1968) and Tremolières (1968). The method of centres generates a sequence of points converging to a minimum solution of the problem. Each of these points (centres) is obtained by unconstrained maximization of a distance function: a particular combination of the objective function and the constraint functions. However, some distance functions may also be regarded as penalty functions *without controlling parameters.* Starting from this point of view, Fiacco and McCormick (1967b) presented a parameter-free version of SUMT, and Fiacco (1967) demonstrated that similar versions can be obtained for a large class of interior-point as well as exterior-point methods. Slightly earlier, a parameter-free exterior-point method was suggested by

Kowalik (1966). Computational experience, however, prompted the author (1968c) to undertake a theoretical study of the *rate of convergence* of these methods as compared with the above-mentioned, parametric techniques.

The above survey does not include all the penalty functions that have been proposed in the last few years. We have restricted ourselves to methods which operate with penalty functions possessing at least continuous first-order partial derivatives in their definition area. Then, the gradient of a penalty function vanishes at a minimizing point. This appears to be a particularly useful relation for theoretical investigations. Fiacco and McCormick (1964a) discovered that SUMT provides primal-feasible as well as dual-feasible solutions of the problem. In so doing, they made a connection between penalty-function techniques and the duality theory for nonlinear programming developed in the years before by Dorn (1960a,b), Wolfe (1961), Huard (1962, 1963) and Mangasarian (1962). The vanishing of the gradient of a penalty function at a minimizing point is also the basis for investigation of the minimizing trajectory and its Taylor series expansion about a minimum solution of problem (1.1.1).

Differentiability has even more implications, however. Computational successes with penalty-function methods depend critically on the efficiency of unconstrained-minimization techniques. Among these, some of the gradient techniques, using first-order and possibly second-order partial derivatives of the function to be minimized have proved to be very successful. The method of *steepest descent* (Curry (1944), Goldstein (1962)) is generally insufficient for minimizing penalty functions. More effective are the *conjugate-gradient methods* (Hestenes and Stiefel (1952), Fletcher and Reeves (1964), Shah, Buehler and Kempthorne (1964), Daniel (1967a, 1967b), Polak and Ribière (1969)). A very powerful technique is *Newton's method* (Crockett and Chernoff (1955), Goldstein and Price (1967), Fiacco and McCormick (1968)), but it has the serious disadvantage that explicit evaluation of the second-order partial derivatives is required. Therefore, one finds an abundant literature on the *quasi-Newton* or *variable-metric methods* requiring first-order derivatives only, but presenting a sophisticated combination of conjugate-gradient techniques and Newton's method (Davidon (1959), Fletcher and Powell (1963), Broyden (1965), Rosen (1966), Broyden (1967), Stewart (1967), Bard (1968), Davidon (1968), Fiacco and McCormick (1968), Myers (1968), Zeleznik (1968), Pearson (1969), Fletcher (1969a, 1969c), Goldfarb (1969), Powell (1969)). There are also several methods for *minimization without calculating derivatives* (Nelder and Mead (1964), Powell (1964), Zangwill (1967c)), but at least to our knowledge, only Powell's method has been used in conjunction with penalty-function techniques. It is doubtful whether this method will be successful if the penalty function is not differentiable at its minimizing point.

Survey papers with some comparison of a number of methods have been presented by Spang (1962), Fletcher (1965), Box (1966), Greenstadt (1967), Topkis and Veinott (1967), Box, Davies and Swann (1969), and Beltrami (1969b).

Our interest in efficient methods of constrained minimization was aroused, first, by the problems arising in the design of the Philips Stirling engine (see Meijer (1969)). Shortly thereafter, our attention was asked for the problem of economic dispatching, a description of which may be found in Carpentier (1962) and Sasson (1969). The research which is reported in the present thesis was carried out since that time, mainly on the grounds of the idea that penalty-function techniques may be useful in solving technological problems.

1.2. Behaviour of penalty functions at the boundary of the constraint set

In view of the abundance of penalty-function techniques just sketched we have been searching for a significant classification. Basically, penalty-function techniques are designed to take into account the constraints of a minimization problem or, since almost none of the practical problems have interior minima, to approach the boundary in a specifically controlled manner. It is therefore natural to classify penalty functions according to their behaviour in the neighbourhood of that boundary. This is the point of departure for the present thesis.

To be specific, let us start with the parametric interior-point methods. For this class of methods we have been concerned with penalty functions of the form

$$f(x) - r \sum_{i=1}^{m} \varphi[g_i(x)]. \tag{1.2.1}$$

Here, r denotes a positive controlling parameter. The function φ is a function of one variable η, defined and continuously differentiable for positive values of η, and such that $\varphi(0+) = -\infty$. Hence, the function (1.2.1) is defined in the interior R^0 of R, but it has a positive singularity at every boundary point of R. Under mild conditions a point $x(r) \in R^0$ exists minimizing (1.2.1) over R^0 for any $r > 0$. This is due to the second term in (1.2.1) which presents itself as a barrier in order to prevent violation of the constraints. Following Murray (1967) we shall therefore briefly refer to interior-point penalty functions as *barrier functions*. Let $\{r_k\}$ denote a monotonic, decreasing null sequence as $k \to \infty$. Then any limit point of $\{x(r_k)\}$ is a minimum solution of (1.1.1).

Formula (1.2.1) shows that there are no differences in the treatment of the constraints: they are all subject to the same transformation φ, in our opinion a reasonable approach as long as one does not make any special assumption on some of the constraint functions.

The classification that we have introduced is based on a property of the derivative φ' of φ: a barrier function is said to be of order λ if the function φ' is analytic and if it has a *pole of order* λ at $\eta = 0$. The choice of the derivative instead of the function itself is not surprising; in the preceding section we have seen that the first-order partial derivatives of penalty functions are of great importance.

Illustrative examples are given by the cases where

$$\varphi'(\eta) = \eta^{-\lambda}, \tag{1.2.2}$$

with a positive λ. For $\lambda = 1$ we obtain the *logarithmic barrier function* on which the logarithmic-programming method is based. For $\lambda = 2$ the function (1.2.1) reduces to the *inverse barrier function* for the sequential unconstrained-minimization technique. Finally, the *inverse quadratic barrier function* named by Kowalik (1966), Box, Davies and Swann (1969), and Fletcher and McCann (1969) is obtained for $\lambda = 3$.

Parametric exterior-point methods can be classified in a similar manner. Here we have been concerned with penalty functions of the form

$$f(x) - s^{-1} \sum_{i=1}^{m} \psi[g_i(x)], \tag{1.2.3}$$

where s is a positive controlling parameter, and ψ a continuously differentiable function of one variable η such that

$$\begin{aligned} \psi(\eta) &= 0 \quad \text{for} \quad \eta \geqslant 0, \\ \psi(\eta) &< 0 \quad \text{for} \quad \eta < 0. \end{aligned} \tag{1.2.4}$$

The second term in (1.2.3) gives a (positive) contribution if, and only if, $x \notin R$. Constraint violation is progressively weighted as s decreases to 0. Under certain conditions a point $x(s)$ exists minimizing (1.2.3) over E_n for sufficiently small, positive values of s. Any limit point of the sequence $\{x(s_k)\}$, where $\{s_k\}$ is a monotonic, decreasing null sequence, is a minimum solution of problem (1.1.1). Following Fiacco and McCormick (1968) we shall refer to penalty functions of the type (1.2.3) as *loss functions.*

For classification purposes we have introduced a function ω such that

$$\omega(\eta) = \psi(\eta) \quad \text{for} \quad \eta \leqslant 0.$$

Now a loss function is said to be of order μ if the derivative ω' of ω is analytic and if it has a *zero of order* μ at $\eta = 0$.

Simple examples of loss functions are obtained by using

$$\omega'(\eta) = (-\eta)^{\mu} \tag{1.2.5}$$

with positive μ. For $\mu = 1$ we find the *quadratic loss function* which has been referred to in the previous section.

We have thus far confined ourselves to penalty functions which contain a controlling parameter. The above classification can, however, readily be extended to a class of methods which may be considered as a generalization of the method of centres. These methods are based on penalty functions *without* *controlling parameter*. A detailed treatment, however, is postponed until that subject is reached in chapter 4.

In the present thesis, parametric barrier functions will be represented by

$$B_r(x) = f(x) - r^\lambda \sum_{i=1}^{m} \varphi[g_i(x)], \tag{1.2.6}$$

where λ denotes the order of the pole of φ' at $\eta = 0$. Raising r to the power λ yields certain advantages when we are dealing with the Taylor series expansion of the minimizing function or "minimizing trajectory" associated with the barrier function in question. Similarly, a parametric loss function is given by

$$L_s(x) = f(x) - s^{-\mu} \sum_{i=1}^{m} \psi[g_i(x)], \tag{1.2.7}$$

where μ stands for the order of the loss function (the order of the zero of ω' at $\eta = 0$).

1.3. Scope of the thesis

In chapter 2 we present material which is needed in the rest of the thesis: necessary conditions (sec. 2.1) and sufficient conditions (sec. 2.2) for constrained minima, a characterization of the boundary and the interior of the constraint set (sec. 2.3), the definition and some properties of convex sets and convex functions (sec. 2.4), and lastly the concept of a convex-programming problem and some duality theorems (sec. 2.5).

In chapter 3 the parametric penalty functions are studied. Mixed penalty functions are introduced in sec. 3.1. In so doing we avoid a separate treatment of barrier-function and loss-function methods. Primal and dual convergence of mixed penalty-function methods are established in secs 3.2 and 3.3 respectively. In sec. 3.4 the behaviour of the minimizing trajectory in a neighbourhood of the constrained minimum is investigated. The analysis is carried out under the so-called Jacobian uniqueness conditions. Lastly, sec. 3.5 deals with the Hessian matrix of mixed penalty functions evaluated at a minimizing point, and with the behaviour of its eigenvalues as r decreases to 0.

In chapter 4 generalizations of the method of centres are presented. A rough sketch of the basic idea (moving truncations of the constraint set) is contained in sec. 4.1. The convergence of the moving-truncations barrier-function techniques and their relationship with parametric barrier-function techniques are established in sec. 4.2. In sec. 4.3 the rate of convergence of these methods is studied. A similar analysis of the moving-truncations loss-function techniques is presented in sec. 4.4.

In chapter 5 the results of the preceding chapters are used in order to motivate the choice of a mixed parametric first-order penalty function for computational purposes.

Finally, there is an appendix presenting an ALGOL 60 procedure for constrained minimization via the last-named penalty function.

2. MATHEMATICAL PRELIMINARIES

2.1. Necessary conditions for constrained minima

We begin by introducing the following terminology.

Definition. Any point $x \in E_n$ satisfying the constraints of problem (1.1.1) is a *feasible solution* of (1.1.1).

Definition. The set of all feasible solutions

$$R = \{x \,|g_i(x) \geqslant 0; \quad i = 1, \ldots, m\} \tag{2.1.1}$$

is the *constraint set* of (1.1.1).

Definition. A feasible solution $\bar{x}$ is a *local minimum solution,* or briefly a *local minimum,* of (1.1.1) if there is an ε-neighbourhood

$$N(\bar{x},\varepsilon) = \{x \,|x \in E_n; \quad ||x - \bar{x}|| < \varepsilon\}$$

of $\bar{x}$ such that $f(\bar{x}) \leqslant f(x)$ for all $x \in R \cap N(\bar{x},\varepsilon)$.

Definition. A feasible solution $\bar{x}$ is a *global minimum solution,* or briefly a *global minimum,* of (1.1.1) if $f(\bar{x}) \leqslant f(x)$ for all $x \in R$.

Definition. A local (or global) minimum $\bar{x}$ of problem (1.1.1) is a local (or global) *unconstrained minimum* of f if an ε-neighbourhood $N(\bar{x},\varepsilon)$ of $\bar{x}$ can be found such that $f(\bar{x}) \leqslant f(x)$ for all $x \in N(\bar{x},\varepsilon)$.

We shall be assuming that the problem functions f, $g_1, \ldots, g_m$ have continuous first-order partial derivatives in E_n. The gradients of f and g_i will be denoted by ∇f and ∇g_i respectively.

It will be convenient to distinguish the constraints which are *active* at a feasible solution x. Therefore we introduce:

$$A(x) = \{i \,|g_i(x) = 0; \quad 1 \leqslant i \leqslant m\}. \tag{2.1.2}$$

We shall now move on to necessary conditions for local minima of (1.1.1) which have been formulated by John (1948) and Kuhn and Tucker (1951). The concepts to be used in deriving them are largely due to the work of Arrow, Hurwicz and Uzawa (1961).

Definition. A vector $y \in E_n$ is a *feasible direction* at $x \in R$ if there exists a positive number μ_0 such that $x + \mu\, y$ is a feasible solution of (1.1.1) for all $0 \leqslant \mu < \mu_0$.

Lemma 2.1.1. If the constraint functions $g_1, \ldots, g_m$ have continuous first-order partial derivatives in E_n, then any $y \in E_n$ satisfying the strict inequalities

$$\nabla g_i(x)^T y > 0; \quad i \in A(x)$$

at a feasible solution x is a feasible direction at x.

Proof. Let us start with an arbitrary $i \in A(x)$ and let us define $h_i(\mu) = g_i(x + \mu y)$. Then $h_i(0) = 0$ and $h_i'(0) = \nabla g_i(x)^T y > 0$. Hence we have, by the continuity of h_i', that a positive μ_i exists such that $h_i(\mu) > 0$ for any μ, $0 \leqslant \mu < \mu_i$.

If $i \notin A(x)$, then $g_i(x) > 0$ and consequently $g_i(x + \mu y) > 0$ for all nonnegative μ smaller than a certain positive μ_i.

Lastly, we choose $\mu_0 = \min(\mu_1, \ldots, \mu_m)$, which completes the proof.

An attempt to find necessary conditions for local minima of inequality-constrained problems was made by John (1948). His result is based on the theory of linear inequalities. A detailed treatment of this theory falls beyond the scope of the present thesis. We shall use some theorems, the proof of which can be found for instance in Zoutendijk (1960), sec. 2.2. The result of John's study is expressed in:

Theorem 2.1.1. If the functions f, $g_1, \ldots, g_m$ have continuous first-order partial derivatives in E_n, and if $\bar{x}$ is a local minimum of (1.1.1), then there exist nonnegative multipliers $\bar{u}_0$, $\bar{u}_1, \ldots, \bar{u}_m$, at least one of which is positive, such that

$$\left.\begin{aligned} &\bar{u}_0 \nabla f(\bar{x}) - \sum_{i=1}^{m} \bar{u}_i \nabla g_i(\bar{x}) = 0, \\ &\bar{u}_i\, g_i(\bar{x}) = 0; \quad i = 1, \ldots, m. \end{aligned}\right\} \tag{2.1.3}$$

Proof. It must be true that either the system

$$\nabla g_i(\bar{x})^T y > 0; \quad i \in A(\bar{x}) \tag{2.1.4}$$

is inconsistent or that, by lemma 2.1.1,

$$\nabla f(\bar{x})^T y \geqslant 0$$

for all $y \in E_n$ satisfying (2.1.4). Anyhow, the system

$$\left.\begin{aligned} -\nabla f(\bar{x})^T y &> 0 \\ \nabla g_i(\bar{x})^T y &> 0; \quad i \in A(\bar{x}) \end{aligned}\right\}$$

is inconsistent. We can then invoke the following theorem (Zoutendijk (1960), p. 9): Let B denote an n-column matrix and y an n vector. The system $By > 0$ (the inequality sign expresses a vector inequality such that any component of By is positive) is inconsistent if, and only if, the transposed system $B^T u = 0$ has a nontrivial, nonnegative solution. Thus, the theorem states that the system $By > 0$ is inconsistent if, and only if, one of the rows of B is a nonpositive linear combination of the remaining rows. Applying this we find that nonnegative multipliers $\bar{u}_0$ and $\bar{u}_i$, $i \in A(\bar{x})$, exist, at least one of which is positive, such that

$$\bar{u}_0 \nabla f(\bar{x}) - \sum_{i \in A(\bar{x})} \bar{u}_i \nabla g_i(\bar{x}) = 0.$$

Finally, we define $\bar{u}_i = 0$ for all $i \notin A(\bar{x})$, and this completes the proof.

Let us discuss this result in more details. Suppose that (2.1.4) happens to be consistent. Then the system

$$\sum_{i \in A(\bar{x})} \bar{u}_i \nabla g_i(\bar{x}) = 0$$

has the trivial solution only, and it must accordingly be true that $\bar{u}_0 \neq 0$. Similarly, $\bar{u}_0$ cannot vanish if the gradients $\nabla g_i(\bar{x})$, $i \in A(\bar{x})$, are linearly independent. Dividing then the first relation in (2.1.3) by $\bar{u}_0$ we find that $\nabla f(\bar{x})$ is a nonnegative linear combination of the gradients $\nabla g_i(\bar{x})$, $i \in A(\bar{x})$. This is precisely a result we need in the subsequent analysis. We shall therefore concern ourselves with a regularity condition (in this field frequently referred to as a constraint qualification) implying $\bar{u}_0 \neq 0$ if it is imposed on problem (1.1.1).

The basic idea underlying the proof of John's theorem was that a decrease of the objective function cannot be found if one performs a small step from $\bar{x}$ into the constraint set. In proving the theorem, however, one only considers the effect of small steps along those feasible directions which satisfy the *strict* inequalities (2.1.4). A natural extension could probably be obtained by treating directions y satisfying

$$\nabla g_i(\bar{x})^T y \geqslant 0; \quad i \in A(\bar{x}). \tag{2.1.5}$$

However, a simple example is sufficient to show that every y which satisfies (2.1.5) is not necessarily a feasible direction at $\bar{x}$. Let us therefore first consider sets of directions which allow us to perform small steps from $\bar{x}$ into R *along curves.*

Definition. A vector $y \in E_n$ is an *attainable direction* at $x \in R$ if there exists an n-vector valued function θ of a real variable η which has the following properties.

1. A positive $\bar{\eta}$ exists such that $\theta(\eta)$ is defined for $0 \leqslant \eta < \bar{\eta}$ and contained in R.
2. $\theta(0) = x$.
3. The function θ has a right-hand-side derivative $\theta'(0)$ at $\eta = 0$, and $\theta'(0) = y$.

The function θ is said to define a *contained path* with *origin* x and *original direction* y.

The paper by Arrow, Hurwicz, and Uzawa (1961) contains an example which shows that the set of attainable directions at $x \in R$ is not necessarily closed. Therefore we introduce the following:

Definition. Any element of the closure of the set of attainable directions at $x \in R$ is a *semi-attainable* direction at x.

Lemma 2.1.2. If the constraint functions $g_1, \ldots, g_m$ have continuous first-order partial derivatives in E_n, then any semi-attainable direction y at $x \in R$ satisfies the inequalities

$$\nabla g_i(x)^T y \geqslant 0; \quad i \in A(x).$$

Proof. It is sufficient to prove the validity of these inequalities for an attainable direction y at x. Let $\theta(\eta)$ define a contained path with origin x and original direction y. For any $i \in A(x)$ we have $g_i[\theta(0)] = 0$ and $g_i[\theta(\eta)] \geqslant 0, 0 \leqslant \eta < \bar{\eta}$, whence

$$\lim_{\eta \downarrow 0} \frac{g_i[\theta(\eta)] - g_i[\theta(0)]}{\eta} = \nabla g_i(x)^T y \geqslant 0.$$

Lemma 2.1.3. If the functions $f, g_1, \ldots, g_m$ have continuous first-order partial derivatives in E_n, and if $\bar{x}$ is a local minimum of (1.1.1), then

$$\nabla f(\bar{x})^T y \geqslant 0,$$

for any semi-attainable direction y at $\bar{x}$.
Proof. This lemma can be proved in a similar way as the preceding one.

Definition. A vector $y \in E_n$ is a *locally constrained direction* at $x \in R$ if

$$\nabla g_i(x)^T y \geqslant 0; \quad i \in A(x).$$

We may now summarize the above results as follows. Any feasible direction at $x \in R$ is attainable; any attainable direction at x is semi-attainable; any semi-attainable direction at x is locally constrained at x. An example which demonstrates that a locally constrained direction at $x \in R$ is not necessarily semi-attainable may be found in the paper by Kuhn and Tucker (1951). For this reason we introduce the following *qualification.*

Definition. A feasible solution x of (1.1.1) is *qualified* if any locally constrained direction at x is semi-attainable at x.

A discussion of the above qualification will be presented later on. We are now in a position to show that the relations (2.1.3) must hold with nonzero $\bar{u}_0$ at a *qualified* minimum solution. This is expressed by the well-known theorem of Kuhn and Tucker:

Theorem 2.1.2. If the functions $f, g_1, \ldots, g_m$ have continuous first-order partial derivatives in E_n, and if $\bar{x}$ is a qualified feasible solution of (1.1.1), then a necessary condition for $\bar{x}$ to be a local minimum of (1.1.1) is that nonnegative multipliers $\bar{u}_1, \ldots, \bar{u}_m$ can be found such that

$$\left.\begin{aligned} \nabla f(\bar{x}) - \sum_{i=1}^{m} \bar{u}_i \nabla g_i(\bar{x}) = 0, \\ \bar{u}_i\, g_i(\bar{x}) = 0; \quad i = 1, \ldots, m. \end{aligned}\right\} \tag{2.1.6}$$

Proof. Using lemma 2.1.2 and 2.1.3 we find that

$$\nabla f(\bar{x})^T y \geqslant 0$$

for any vector $y \in E_n$ such that

$$\nabla g_i(\bar{x})^T y \geqslant 0; \quad i \in A(\bar{x}).$$

We may now restate the well-known theorem of Farkas: Let c and x denote vectors in E_n and let B be an n-column matrix. Then $c^T x \geqslant 0$ for any x satisfying $Bx \geqslant 0$ if, and only if, c^T is a nonnegative linear combination of the rows of B. For a proof of Farkas' theorem the reader is referred to Zoutendijk (1960), p. 8. Applying Farkas' theorem we find that nonnegative multipliers $\bar{u}_i$, $i \in A(\bar{x})$, exist such that

$$\nabla f(\bar{x}) - \sum_{i \in A(\bar{x})} \bar{u}_i \nabla g_i(\bar{x}) = 0.$$

Defining $\bar{u}_i = 0$, $i \in A(\bar{x})$, we can readily complete the proof.

According to the theorem of Kuhn and Tucker it is necessary for $\bar{x}$ to be a local minimum that $\nabla f(\bar{x})$ is a nonnegative linear combination of the gradients $\nabla g_i(\bar{x})$ of the active constraints at $\bar{x}$. This is expressed by the *Kuhn–Tucker relations* (2.1.6). However, in proving (2.1.6) we have imposed an additional condition on $\bar{x}$ in order to guarantee that $\nabla f(\bar{x})^T y \geqslant 0$ for any locally constrained direction at $\bar{x}$. Conditions of this kind have become quite familiar in nonlinear programming under the name of *constraint qualification.* Kuhn and Tucker (1951), for instance, required any locally constrained direction to be *attainable* at any feasible solution.

Several authors have been dealing with the question of how to find simple conditions implying a constraint qualification. An extensive treatment of these attempts will not be given here. In the next theorem we only recall a number of results which are due to Arrow, Hurwicz and Uzawa (1961), Mangasarian and Fromowitz (1967), and Fiacco and McCormick (1968).

Theorem 2.1.3. Let x be a feasible solution of problem (1.1.1). If (a) the functions $g_1, \ldots, g_m$ have continuous first-order partial derivatives in E_n, and (b) for some locally constrained direction y_0 at x a partitioning of $A(x)$ into two disjunct subsets $A_1(x)$ and $A_2(x)$ can be found with the following two properties:

(i) $\nabla g_i(x)^T y_0 > 0; \quad i \in A_1(x)$,

(ii) the gradients $\nabla g_i(x)$, $i \in A_2(x)$, linearly independent,

then x is a qualified feasible solution of problem (1.1.1).

Proof. Let y be a nonzero locally constrained direction at x. It is sufficient to demonstrate that the direction $z = y + \varepsilon y_0$ is attainable at x for any positive ε. Then y is semi-attainable at x. We can easily obtain

$$\nabla g_i(x)^T z > 0; \quad i \in A_1(x),$$
$$\nabla g_i(x)^T z \geqslant 0; \quad i \in A_2(x).$$

Let
$$\tilde{A}_2(x,z) = \{i \mid i \in A_2(x); \quad \nabla g_i(x)^T z = 0\}.$$

We try to construct a contained path $\theta(\eta)$ with origin x and original direction z. Let $G(\eta)$ denote the matrix with columns $\nabla g_i[\theta(\eta)]$, $i \in \tilde{A}_2(x,z)$. We define $\theta(0) = x$ and $\theta'(\eta)$ as the projection of z on the linear subspace of E_n which is orthogonal to the columns of $G(\eta)$. Then

$$\theta'(\eta) = [I_n - G(\eta)\, \{G(\eta)^T\, G(\eta)\}^{-1}\, G(\eta)^T]\, z.$$

The choice is possible since the columns of $G(0)$ are, by assumption, linearly independent. This implies that the inverse of $G(0)^T\, G(0)$ exists. Similarly, the inverse of $G(\eta)^T\, G(\eta)$ exists by the continuity of the gradients, implying that the columns of $G(\eta)$ are linearly independent for sufficiently small, positive η. Obviously, $\theta'(0) = z$.

It remains to show that we have constructed a path which is *contained* in R. For any $i \in A_1(x)$ and any $i \in A_2(x) - \tilde{A}_2(x,z)$ we have $g_i[\theta(0)] = 0$ and

$$\nabla g_i[\theta(0)]\, \theta'(0) = \nabla g_i(x)^T z > 0,$$

so that $g_i[\theta(\eta)] > 0$ for sufficiently small, positive η. Let us finally consider an $i \in \tilde{A}_2(x,z)$. The mean-value theorem leads to

$$g_i[\theta(\eta)] = g_i[\theta(0)] + \eta\, \nabla g_i[\theta(\zeta)]\, \theta'(\zeta),$$

with $0 \leqslant \zeta \leqslant \eta$. The right-hand side vanishes since, by construction, $\theta'(\zeta)$ is orthogonal to $\nabla g_i[\theta(\zeta)]$ for any $i \in \tilde{A}_2(x,z)$.

Combining the results we find that z is an attainable direction at x, and consequently y is semi-attainable at x. This proves the theorem.

It is worthwhile to note that either $A_1(x)$ or $A_2(x)$ may be empty. Hence, the above theorem provides two sufficient conditions for a feasible solution to be qualified, namely existence of a direction y_0 such that

$$\nabla g_i(x)^T y_0 > 0, \quad i \in A(x),$$

or linear independence of the gradients $\nabla g_i(x)$, $i \in A(x)$. These conditions have also been discussed at the end of theorem 2.1.1.

Every feasible solution of a *linearly constrained* problem is qualified: then, namely, any locally constrained direction at a feasible solution x is a feasible direction at x. The Kuhn–Tucker relations (2.1.6) are thus satisfied at any local minimum, without additional conditions.

If none of the constraints is active at a local minimum $\bar{x}$, theorem 2.1.2 states merely that $\nabla f(\bar{x}) = 0$, a well-known result from classical analysis.

We conclude this section by introducing some terminology.

Definition. A pair $(\bar{x},\bar{u}) \in E_n \times E_m$ is a *Kuhn–Tucker point* of problem (1.1.1) if the requirements of (2.1.7) to (2.1.10) are satisfied:

$$g_i(\bar{x}) \geqslant 0; \quad i = 1, \ldots, m, \tag{2.1.7}$$

$$\bar{u}_i \geqslant 0; \quad i = 1, \ldots, m, \tag{2.1.8}$$

$$\bar{u}_i\, g_i(\bar{x}) = 0; \quad i = 1, \ldots, m, \tag{2.1.9}$$

$$\nabla f(\bar{x}) - \sum_{i=1}^{m} \bar{u}_i \nabla g_i(\bar{x}) = 0. \tag{2.1.10}$$

The constraints and the multipliers $\bar{u}_i$ are, as it is shown by (2.1.9), *complementary*: the multiplier $\bar{u}_i$ can only be positive if the ith constraint is active. We shall say that the ith constraint is *strongly active* if $\bar{u}_i > 0$, and *weakly active* if it is active but if $\bar{u}_i = 0$. A Kuhn–Tucker point $(\bar{x},\bar{u})$ is *strict complementary* if $\bar{u}_i > 0$ for any $i \in A(\bar{x})$.

The results of theorem 2.1.2 may now be summarized as follows: if $\bar{x}$ is a qualified, local minimum of problem (1.1.1), then a vector $\bar{u} \in E_m$ can be found such that $(\bar{x},\bar{u})$ is a Kuhn–Tucker point of (1.1.1).

2.2. Sufficient conditions for constrained minima

A sufficient condition for a point $\bar{x}$ to be a local *unconstrained* minimum of a function f can, it is known, be formulated with the help of the second-order derivatives of f at $\bar{x}$. This idea can readily be extended to the case of *constrained* minima. We shall henceforth assume that the problem functions $f, g_1, \ldots, g_m$ have continuous second-order partial derivatives in E_n, and we introduce the following notation. The matrix of second-order derivatives of f evaluated at x, usually referred to as the *Hessian matrix* of f at x, will be represented by $\nabla^2 f(x)$. A similar notation will be employed for the Hessian matrices of $g_1, \ldots, g_m$. Lastly, we introduce

$$D(x,u) = \nabla^2 f(x) - \sum_{i=1}^{m} u_i \nabla^2 g_i(x). \tag{2.2.1}$$

Theorem 2.2.1. If (a) the functions $f, g_1, \ldots, g_m$ have continuous second-order partial derivatives in E_n, (b) a Kuhn–Tucker point $(\bar{x},\bar{u})$ of problem (1.1.1) exists, and (c) an ε-neighbourhood $N(\bar{x},\varepsilon)$ of $\bar{x}$ can be found such that $D(x,\bar{u})$ is positive semi-definite for any $x \in R \cap N(\bar{x},\varepsilon)$, then $\bar{x}$ is a local minimum of (1.1.1).

Proof. Let us assume the contrary, that $\bar{x}$ is not a local minimum. Then a sequence $\{x_k\}$ of feasible solutions can be found, converging to $\bar{x}$ and such that $f(x_k) < f(\bar{x})$. Writing $x_k = \bar{x} + y_k$ and using a Taylor series expansion about $\bar{x}$ we find that

$$f(\bar{x} + y_k) - \sum_{i=1}^{m} \bar{u}_i \, g_i(\bar{x} + y_k) =$$

$$= f(\bar{x}) - \sum_{i=1}^{m} \bar{u}_i \, g_i(\bar{x}) + [\nabla f(\bar{x}) - \sum_{i=1}^{m} \bar{u}_i \, \nabla g_i(\bar{x})]^T \, y_k + \tfrac{1}{2} \, y_k{}^T \, D(\xi_k, \bar{u}) \, y_k,$$

where $\xi_k = \bar{x} + \lambda_k \, y_k$ for some $0 \leqslant \lambda_k \leqslant 1$. Using (2.1.7) to (2.1.10) we obtain

$$f(\bar{x} + y_k) - f(\bar{x}) \geqslant \tfrac{1}{2} \, y_k{}^T \, D(\xi_k, \bar{u}) \, y_k,$$

whence

$$\tfrac{1}{2} \, y_k{}^T \, D(\xi_k, \bar{u}) \, y_k < 0.$$

For k sufficiently large, however, it must be true that $\xi_k \in R \cap N(\bar{x}, \varepsilon)$. This leads to a contradiction and proves the theorem.

Definition. A local minimum $\bar{x}$ of problem (1.1.1) is *isolated*, or *locally unique*, if an ε-neighbourhood $N(\bar{x}, \varepsilon)$ of $\bar{x}$ exists such that $f(\bar{x}) < f(x)$ for any $x \in R \cap N(\bar{x}, \varepsilon)$.

One may expect that $\bar{x}$ will be an isolated local minimum of (1.1.1) if $D(\bar{x}, \bar{u})$ is *positive definite*. The next theorem shows that we can find a weaker condition implying local uniqueness of $\bar{x}$. We only have to require that $D(\bar{x}, \bar{u})$ be positive definite with respect to some locally constrained directions at $\bar{x}$.

Theorem 2.2.2. If (a) the functions f, $g_1, \ldots, g_m$ have continuous second-order derivatives in E_n, (b) a Kuhn–Tucker point $(\bar{x}, \bar{u})$ of problem (1.1.1) exists, and (c) it is true that

$$y^T \, D(\bar{x}, \bar{u}) \, y > 0$$

for any $y \in E_n$, $y \neq 0$, satisfying

$$\begin{array}{llll} \nabla g_i(\bar{x})^T \, y \geqslant 0 & \text{for any} & i \in A(\bar{x}) & \text{such that} \quad \bar{u}_i = 0, \\ \nabla g_i(\bar{x})^T \, y = 0 & \text{for any} & i \in A(\bar{x}) & \text{such that} \quad \bar{u}_i > 0, \end{array}$$

then $\bar{x}$ is an isolated local minimum of (1.1.1).

Proof. Let us assume the contrary. Then a sequence $\{x_k\}$ of feasible solutions can be found, converging to $\bar{x}$ and such that $f(x_k) \leqslant f(\bar{x})$. We can write $x_k = \bar{x} + \delta_k \, y_k$ with $||y_k|| = 1$ and $\delta_k > 0$. Then a limit point $(0, \bar{y})$ of the sequence $\{(\delta_k, y_k)\}$ exists, and $||\bar{y}|| = 1$. We can now obtain

$$\lim_{k \to \infty} \frac{f(\bar{x} + \delta_k \, y_k) - f(\bar{x})}{\delta_k} = \nabla f(\bar{x})^T \, \bar{y} \leqslant 0,$$

and for any $i \in A(\bar{x})$ we have

$$\lim_{k\to\infty} \frac{g_i(\bar{x} + \delta_k\, y_k) - g_i(\bar{x})}{\delta_k} = \nabla g_i(\bar{x})^T\, \bar{y} \geqslant 0.$$

Application of the Kuhn–Tucker relations leads to

$$0 \geqslant \nabla f(\bar{x})^T\, \bar{y} = \sum_{i=1}^{m} \bar{u}_i\, \nabla g_i(\bar{x})^T\, \bar{y} \geqslant 0.$$

Hence

$$\bar{u}_i\, \nabla g_i(\bar{x})^T\, \bar{y} = 0, \quad i \in A(\bar{x}),$$

and we can write, according to condition (c) of the theorem,

$$\bar{y}^T\, D(\bar{x},\bar{u})\, \bar{y} > 0. \tag{2.2.2}$$

Using a Taylor series expansion about $\bar{x}$ we obtain

$$f(\bar{x} + \delta_k\, y_k) - \sum_{i=1}^{m} \bar{u}_i\, g_i(\bar{x} + \delta_k\, y_k) =$$

$$= f(\bar{x}) - \sum_{i=1}^{m} \bar{u}_i\, g_i(\bar{x}) + \delta_k\, [\nabla f(\bar{x}) - \sum_{i=1}^{m} \bar{u}_i\, \nabla g_i(\bar{x})]^T\, y_k +$$

$$+ \tfrac{1}{2}\, \delta_k{}^2\, y_k{}^T\, [\nabla^2 f(\xi_k) - \sum_{i=1}^{m} \bar{u}_i\, \nabla^2 g_i(\xi_k)]\, y_k,$$

which can, by (2.1.7) to (2.1.10), be reduced to the inequality

$$y_k{}^T\, D(\xi_k,\bar{u})\, y_k \leqslant 0.$$

Here, ξ_k represents a point on the line segment connecting $\bar{x}$ and $\bar{x} + \delta_k\, y_k$. Taking the limit as $k \to \infty$ we find that

$$\bar{y}^T\, D(\bar{x},\bar{u})\, \bar{y} \leqslant 0,$$

which contradicts (2.2.2) so that the proof of the theorem is completed.

If there are no active constraints at $\bar{x}$ the above theorem reduces to the following well-known result: if $\nabla f(\bar{x}) = 0$ and $\nabla^2 f(\bar{x})$ is positive definite, then $\bar{x}$ is an isolated local *unconstrained minimum* of f.

In the next chapter we shall frequently make an appeal to a theorem which supplies a set of conditions implying, amongst other things, local uniqueness of a Kuhn–Tucker point of (1.1.1). The theorem is based on the idea that a

Kuhn–Tucker point $(\bar{x},\bar{u})$ solves the system

$$\left.\begin{aligned} &\nabla f(x) - \sum_{i=1}^{m} u_i \nabla g_i(x) = 0, \\ &u_i g_i(x) = 0; \quad i = 1, \ldots, m, \end{aligned}\right\} \tag{2.2.3}$$

consisting of $m + n$ nonlinear equations and involving $m + n$ variables. Let $\bar{J}$ denote the Jacobian matrix of (2.2.3), evaluated at $(\bar{x},\bar{u})$. If $\bar{J}$ is nonsingular, it must be true by the inverse-function theorem (De la Vallée Poussin (1946)) that a neighbourhood of $(\bar{x},\bar{u})$ exists where $(\bar{x},\bar{u})$ is the unique solution of (2.2.3).

Definition. A Kuhn–Tucker point $(\bar{x},\bar{u})$ of problem (1.1.1) satisfies the *Jacobian uniqueness conditions*, if the following three conditions are simultaneously satisfied.

Condition 2.1. The multipliers $\bar{u}_i$, $i \in A(\bar{x})$, are positive.

Condition 2.2. The gradients $\nabla g_i(\bar{x})$, $i \in A(\bar{x})$, are linearly independent.

Condition 2.3. For any $y \in E_n$, $y \neq 0$, such that

$$\nabla g_i(\bar{x})^T y = 0, \quad i \in A(\bar{x}),$$

it must be true that

$$y^T D(\bar{x},\bar{u})\, y > 0.$$

Theorem 2.2.3. If (a) the functions f, $g_1, \ldots, g_m$ have continuous second-order partial derivatives in E_n, and (b) a Kuhn–Tucker point $(\bar{x},\bar{u})$ of problem (1.1.1) exists which satisfies the Jacobian uniqueness conditions 2.1 to 2.3, then the Jacobian matrix $\bar{J}$ of the Kuhn–Tucker relations (2.2.3) evaluated at $(\bar{x},\bar{u})$ is nonsingular. This implies that the point $\bar{x}$ is an isolated local minimum of (1.1.1) and that the vector $\bar{u}$ of associated multipliers is uniquely determined.

Proof. To start with we introduce some additional notation. We think of the constraints as arranged in such a way that

$$g_i(\bar{x}) = 0, \quad \bar{u}_i > 0; \quad i = 1, \ldots, \alpha,$$

$$g_i(\bar{x}) > 0, \quad \bar{u}_i = 0; \quad i = \alpha + 1, \ldots, m,$$

and we employ $\bar{U}$ to denote a diagonal matrix of order α with the *positive* diagonal elements $\bar{u}_i$, $i = 1, \ldots, \alpha$. The matrix G will represent a diagonal matrix of order $m - \alpha$ with the *positive* diagonal elements $g_i(\bar{x})$, $i = \alpha + 1, \ldots, m$. Let H_1 denote the matrix with the *linearly independent* columns $\nabla g_i(\bar{x})$, $i = 1, \ldots, \alpha$, and H_2 the matrix with the columns $\nabla g_i(\bar{x})$, $i = \alpha + 1, \ldots, m$. Finally, the symbol D will be used to denote briefly the matrix $D(\bar{x},\bar{u})$. With these arrangements and notations $\bar{J}$ can be put into the form

$$\bar{J} = \begin{pmatrix} D & -H_1 & -H_2 \\ \bar{U}H_1{}^T & 0 & 0 \\ 0 & 0 & G \end{pmatrix}. \tag{2.2.4}$$

Clearly, we have to guarantee that the submatrix

$$\begin{pmatrix} D & -H_1 \\ \bar{U}H_1{}^T & 0 \end{pmatrix}$$

is nonsingular. We shall demonstrate that the system

$$\left.\begin{aligned} Dy \;-\; H_1 v &= 0 \qquad (2.2.5) \\ \bar{U}H_1{}^T y \qquad &= 0 \qquad (2.2.6) \end{aligned}\right\}$$

has the trivial solution only. Condition 2.1 implies that $\bar{U}$ is nonsingular. It follows then from (2.2.6) that $H_1{}^T y = 0$. Premultiplying (2.2.5) by y^T we obtain

$$y^T Dy - y^T H_1 v = 0,$$

whence

$$y^T Dy = 0.$$

Using condition 2.3 we can write $y = 0$. Now $H_1 v = 0$, and it follows from condition 2.2 that $v = 0$. Hence $\bar{J}$ is nonsingular, and accordingly a neighbourhood of $(\bar{x},\bar{u})$ can be found where $(\bar{x},\bar{u})$ is the unique solution of the Kuhn–Tucker relations.

Using theorem 2.2.2 we may conclude, on the basis of conditions 2.1 and 2.3, that $\bar{x}$ is an isolated local minimum of (1.1.1). The uniqueness of $\bar{u}$ is implied by condition 2.2.

The above theorem can also be applied if the problem under consideration is one of linear programming. Then $D(\bar{x},\bar{u}) = 0$, but if there are exactly n active constraints at $\bar{x}$ satisfying conditions 2.1 and 2.2, then the set of all $y \in E_n$, $y \neq 0$, such that

$$\nabla g_i(\bar{x})^T y = 0; \quad i \in A(\bar{x}),$$

is empty. Hence, condition 2.3 of the theorem is also satisfied although $D(\bar{x},\bar{u}) = 0$.

2.3. The boundary and the interior of the constraint set

In this section we are concerned with the interior R^0 of the constraint set R defined by (1.1.2), and with the set

$$P(R) = \{x \mid g_i(x) > 0; \quad i = 1, \ldots, m\}.$$

It is important for interior-point methods that R can be characterized as the *closure* of $P(R)$. Then, namely, any point $x \in R$ can be attained via a sequence $\{x_k\}$ of points each of which satisfies the constraints of the problem with strict inequality sign.

It will be convenient to define a function $\tilde{g}$ by

$$\tilde{g}(x) = \min\,[g_1(x), \ldots, g_m(x)]. \tag{2.3.1}$$

Then

$$R = \{x \,|\tilde{g}(x) \geqslant 0\},$$

and

$$P(R) = \{x \,|\tilde{g}(x) > 0\}. \tag{2.3.2}$$

Lastly, we introduce the set $Z(R)$ by defining

$$Z(R) = \{x \,|\tilde{g}(x) = 0\}. \tag{2.3.3}$$

If we assume continuity of the constraint functions $g_1, \ldots, g_m$, then $\tilde{g}$ is continuous in E_n, the set R is a *closed* subset of E_n, and $P(R)$ is contained in R^0.

The results to follow, which are closely connected with local minima and maxima of $\tilde{g}$, are largely due to Bui Trong Lieu and Huard (1966), and Tremolières (1968). They presented a necessary and sufficient condition for $Z(R)$ to be the boundary of R (implying that $P(R)$ is the interior of R), as well as a necessary and sufficient condition for R to be the closure of $P(R)$.

Theorem 2.3.1. Let the constraint functions $g_1, \ldots, g_m$ be continuous in E_n. Then the set $Z(R)$ is the boundary of R if, and only if, no local, unconstrained minimum of $\tilde{g}$ belongs to $Z(R)$.

Proof. Let us start by proving the if-part of the theorem. First, we show that $Z(R)$ is contained in the boundary of R. Let $x_0 \in Z(R)$ and let $N(x_0,\varepsilon)$ denote an ε-neighbourhood of x_0. The set $N(x_0,\varepsilon) \cap R$ is nonempty since x_0 is contained in it. On the other hand, a point $x_1 \in N(x_0,\varepsilon)$ can be found such that $\tilde{g}(x_1) < \tilde{g}(x_0)$ since x_0 is not a local, unconstrained minimum of $\tilde{g}$. The set $N(x_0,\varepsilon)$ contains an element of R as well as a point which does not belong to R for arbitrary, positive values of ε. Hence, x_0 is a boundary point of R. Second, we consider a boundary point x_2 of R and we suppose that $\tilde{g}(x_2) \neq 0$. If $\tilde{g}(x_2) > 0$, then x_2 is an interior point of R. If $\tilde{g}(x_2) < 0$, then x_2 is an interior point of the complement of R. In both cases we have a contradiction, and it must be true that $\tilde{g}(x_2) = 0$. Combination of the results leads to the conclusion that $Z(R)$ is the boundary of R.

To show the reverse, we start from the assumption that $Z(R)$ is the boundary of R. Consider an arbitrary $x_0 \in Z(R)$ and an ε-neighbourhood $N(x_0,\varepsilon)$ of x_0. Then a point $x_1 \in N(x_0,\varepsilon)$ can be found such that $\tilde{g}(x_1) < \tilde{g}(x_0)$. Hence, x_0 cannot be a local unconstrained minimum of $\tilde{g}$, which completes the proof.

Corollary. The set $P(R)$ defined by (2.3.2) is the interior R^o of R if, and only if, no local unconstrained minimum of $\tilde{g}$ belongs to $Z(R)$.

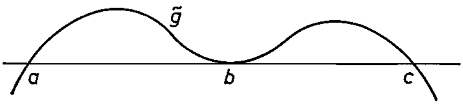

Fig. 2.1.

Figure 2.1 shows a situation which is ruled out if no local minimum of $\tilde{g}$ belongs to $Z(R)$. Here, the interior R^o of the constraint set R is given by the open interval (a,c). The point $b \in R^o$ belongs to $Z(R)$.

Lastly, we find a condition which ensures that R is the closure of $P(R)$.

Theorem 2.3.2. Let the constraint functions $g_1, \ldots, g_m$ be continuous in E_n and suppose that $P(R)$ is nonempty. Then R is the closure of $P(R)$ if, and only if, no local, unconstrained maximum of $\tilde{g}$ belongs to $Z(R)$.

Proof. We start by proving the if-part of the theorem. It is sufficient to consider a point $x_0 \in Z(R)$. Suppose that a positive δ can be found such that $N(x_0,\delta)$ does not contain any point of $P(R)$. Then $\tilde{g}(x) \leqslant \tilde{g}(x_0)$ for any $x \in N(x_0,\delta)$, which implies that x_0 is a local, unconstrained maximum of $\tilde{g}$.

Conversely, if R is the closure of $P(R)$, we suppose that a local, unconstrained maximum x_1 of $\tilde{g}$ belongs to $Z(R)$. We can then find a neighbourhood $N(x_1,\delta)$ of x_1 such that $\tilde{g}(x) \leqslant \tilde{g}(x_1) = 0$ for any $x \in N(x_1,\delta)$, contradicting that an element of $P(R)$ can be found in any neighbourhood of x_1.

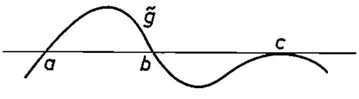

Fig. 2.2.

Figure 2.2 is given in order to illustrate theorem 2.3.2. Here, the set R is the union of the closed interval $[a,b]$ and the point c. The interior R^o of R is given by (a,b); the closure of R^o consists of $[a,b]$ only.

2.4. Convex sets and convex functions

In this section we shall briefly sum up the properties of convex sets and convex functions that we need in subsequent chapters. The proofs will be omitted. They can be found in many textbooks such as, for example, Berge (1951) or Berge and Ghouila-Houri (1962).

Definition. A set $C \subset E_n$ is *convex* if $\lambda x_1 + (1-\lambda) x_2 \in C$ for every two points $x_1 \in C$ and $x_2 \in C$ and every λ, $0 \leqslant \lambda \leqslant 1$.

Theorem 2.4.1. The intersection of two convex sets is a convex set.

Definition. Let C be a convex set and f a function defined in C. Then f is *convex* in C if

$$f[\lambda x_1 + (1-\lambda) x_2] \leqslant \lambda f(x_1) + (1-\lambda) f(x_2) \qquad (2.4.1)$$

for every two points $x_1 \in C$ and $x_2 \in C$ and every λ, $0 \leqslant \lambda \leqslant 1$. The function f is *strictly convex* in C if strict inequality holds in (2.4.1) when $0 < \lambda < 1$ and $x_1 \neq x_2$. If f is (strictly) convex in E_n, it will briefly be referred to as a (strictly) convex function.

In the remainder of this section the symbols C and C^o will invariably be used to denote, respectively, a convex set in E_n and its interior.

Theorem 2.4.2. If $f_1, \ldots, f_p$ are convex functions in C, then any nonnegative linear combination of these functions is convex in C. The function f defined by

$$\bar{f}(x) = \max\,[f_1(x), \ldots, f_p(x)]$$

is also convex in C.

Theorem 2.4.3. If f is a convex function in C, then the set

$$\{x \mid f(x) \leqslant a, \quad x \in C\}$$

is convex (possibly empty) for any a.

Theorem 2.4.4. If f is a convex function in C, and if h is a nondecreasing, convex function in E_1, then $h(f)$ is convex in C.

Theorem 2.4.5. If f is a convex function in C, then f is continuous in the interior C^o of C.

Theorem 2.4.6. If f has continuous first-order partial derivatives in C, then f is convex in C if, and only if,

$$f(x_2) - f(x_1) \leqslant \nabla f(x_2)^T (x_2 - x_1) \qquad (2.4.2)$$

for every two points $x_1 \in C$ and $x_2 \in C$.

Theorem 2.4.7. If f has continuous second-order partial derivatives in C, then f is convex in C if, and only if, $\nabla^2 f(x)$ is positive semi-definite in C. If $\nabla^2 f(x)$ is positive definite for any $x \in C$, then f is strictly convex in C. (The reverse of the last statement is not necessarily true.)

Theorem 2.4.8. If f is a convex function in C, then any local minimum of f in C is a global minimum of f in C. If f is strictly convex in C, then a minimum of f in C is unique.

Theorem 2.4.9. If f is convex in C and if it possesses continuous first-order partial derivatives in C, then a point $\bar{x} \in C^o$ is a minimum of f in C if, and only if, $\nabla f(\bar{x}) = 0$.

Definition. A function g defined in C is *concave* in C if $-g$ is convex in C.

It will be convenient to sum up a number of properties of concave functions which follow from the above theorems.

Theorem 2.4.10. Every nonnegative linear combination of functions $g_1, \ldots, g_m$ which are concave in C is concave in C. The function $\bar{g}$ defined by

$$\bar{g}(x) = \min\,[g_1(x), \ldots, g_m(x)]$$

is also concave in C.

Theorem 2.4.11. If g is a concave function in C, then the set

$$\{x \,|g(x) \geqslant a, \quad x \in C\}$$

is convex (possibly empty) for any a.

Theorem 2.4.12. If g is concave in C, and if h is a nondecreasing, concave function in E_1, then $h(g)$ is concave in C.

Theorem 2.4.13. If g has continuous first-order partial derivatives in C, then g is concave in C if, and only if,

$$g(x_2) - g(x_1) \geqslant \nabla g(x_2)^T\,(x_2 - x_1) \tag{2.4.3}$$

for every two points $x_1 \in C$ and $x_2 \in C$.

The counterparts of the theorems 2.4.8 and 2.4.9 can readily be obtained if one replaces the concepts "convex function" and "minimum" by "concave function" and "maximum". Lastly, we have:

Theorem 2.4.14. If a local minimum $\bar{x}$ of a concave function g in C belongs to C^o, then $\bar{x}$ is also a maximum of g.
Proof. There is an ε-neighbourhood $N(\bar{x},\varepsilon) \subset C$, such that $g(x) \geqslant g(\bar{x})$ for any $x \in N(\bar{x},\varepsilon)$. Select two points x_1 and $x_2 \in N(\bar{x},\varepsilon)$ such that $\bar{x} = \frac{1}{2}(x_1 + x_2)$. Then, by the concavity of g,

$$g(\bar{x}) \geqslant \tfrac{1}{2}\,g(x_1) + \tfrac{1}{2}\,g(x_2) \geqslant g(\bar{x}).$$

It follows that $g(x) = g(\bar{x})$ for any $x \in N(\bar{x},\varepsilon)$. Hence, $\bar{x}$ is a local maximum and accordingly a global maximum of g in C.

2.5. Convex programming

The original problem (1.1.1) is said to be one of *convex programming* if the objective function f is convex, and if the constraint functions $g_1, \ldots, g_m$ are concave in E_n.

Theorem 2.5.1. The constraint set R of the convex-programming problem (1.1.1) is convex.
Proof. This follows directly from the theorems 2.4.11 and 2.4.1.

Theorem 2.5.2. Any local minimum of the convex-programming problem (1.1.1) is a global minimum.
Proof. See theorem 2.5.1 and use theorem 2.4.8 with $C = R$.

Theorem 2.5.3. If the constraint functions $g_1, \ldots, g_m$ of problem (1.1.1) are concave, and if a point x_0 exists which satisfies the constraints with strict inequality sign, then
(a) the interior R^o of R is given by the set

$$P(R) = \{x \,|g_i(x) > 0; \quad i = 1, \ldots, m\};$$

(b) the boundary of R is given by the set

$$Z(R) = R - P(R);$$

(c) the set R is the closure of its interior.
Proof. Let $\tilde{g}$ be defined by (2.3.1). Then, by theorem 2.4.10, $\tilde{g}$ is concave in E_n. Moreover, by theorem 2.4.5, $\tilde{g}$ is continuous in E_n.

We note, firstly, that a local, unconstrained maximum of $\tilde{g}$ cannot belong to the set $Z(R) = \{x \,|\tilde{g}(x) = 0\}$, since a point x_0 exists such that $\tilde{g}(x_0) > 0$.

Using theorem 2.4.14 with $g = \tilde{g}$ and $C = E_n$ we find that a local, unconstrained minimum of $\tilde{g}$ cannot belong to $Z(R)$ either. Now, the theorem follows immediately from theorems 2.3.1 and 2.3.2.

The proof that R is the closure of $P(R)$ can also be given in a more direct way. Consider an arbitrary $x \in R$ and the line segment connecting x and x_0. Let

$$x(\lambda) = (1 - \lambda)\, x + \lambda\, x_0, \quad 0 \leqslant \lambda \leqslant 1.$$

By concavity of $\tilde{g}$ we obtain

$$\tilde{g}\,[x(\lambda)] \geqslant (1 - \lambda)\, \tilde{g}(x) + \lambda\, \tilde{g}(x_0), \quad 0 \leqslant \lambda \leqslant 1,$$

so that $\tilde{g}\,[x(\lambda)] > 0$ for any $0 < \lambda \leqslant 1$. Hence, $x(\lambda) \in P(R)$ for any $0 < \lambda \leqslant 1$, which completes the proof.

Theorem 2.5.4. If the constraint functions $g_1, \ldots, g_m$ are concave and if R is nonempty and compact, then the set

$$R(b) = \{x \,|g_i(x) \geqslant -b_i; \quad i = 1, \ldots, m\}$$

is compact (possibly empty) for any *perturbation* $b = (b_1, \ldots, b_m)^T$ of the constraints.

Proof. Theorem 2.4.5 implies that $R(b)$ is closed for any perturbation $b \in E_m$. It is sufficient to show that $R(b)$ is bounded for $b = \bar{b} = (\bar{b}_1, 0, \ldots, 0)$, $\bar{b}_1 > 0$. Let us assume the contrary, that $R(\bar{b})$ is unbounded, and let us choose a point $x_1 \in R$. A straight line emanating from x_1 can then be found which intersects the boundary of R but not the boundary of $R(\bar{b})$. Let x_2 be a point on that line such that

$$\left.\begin{array}{l} g_1(x_2) = -\delta < 0, \\ g_i(x_2) \geqslant 0; \quad i = 2, \ldots, m. \end{array}\right\}$$

Lastly, we consider a point w on that line such that x_2 is a convex combination of w and x_1:

$$x_2 = \lambda\, w + (1 - \lambda)\, x_1, \quad 0 < \lambda \leqslant 1.$$

By the concavity of g_1 we have

$$g_1(x_2) \geqslant \lambda\, g_1(w) + (1 - \lambda)\, g_1(x_1) \geqslant \lambda\, g_1(w),$$

whence

$$g_1(w) \leqslant \frac{-\delta}{\lambda}.$$

The point w belongs to $R(\bar{b})$ for any λ, $0 < \lambda \leqslant 1$. However, by choosing λ sufficiently small, we can obtain the contradictory result

$$g_1(w) < -\bar{b}_1.$$

Hence, $R(b)$ is compact for any perturbation b.

Having established some desirable topological properties of R, we shall now move on to necessary and sufficient conditions for constrained minima of a convex-programming problem.

Theorem 2.5.5. If (a) problem (1.1.1) is a convex-programming problem, and (b) the problem functions f, $g_1, \ldots, g_m$ have continuous first-order partial derivatives in E_n, then a sufficient condition for $\bar{x}$ to be a minimum solution of (1.1.1) is that a vector $\bar{u} \in E_m$ can be found such that $(\bar{x},\bar{u})$ is a Kuhn–Tucker point.

Proof. It follows from (2.1.8), (2.1.10) and theorems 2.4.10 and 2.4.9 that $\bar{x}$ is a point minimizing the *convex* function

$$f(x) - \sum_{i=1}^{m} \bar{u}_i\, g_i(x)$$

over E_n. Using (2.1.9) we can obtain

$$f(x) - \sum_{i=1}^{m} \bar{u}_i\, g_i(x) \geqslant f(\bar{x}) \quad \text{for any} \quad x \in E_n,$$

which implies

$$f(x) \geqslant f(\bar{x}) \quad \text{for any} \quad x \in R.$$

This completes the proof of the theorem.

A convex-programming problem admits of an easy criterion for deciding whether a feasible solution is qualified. This is expressed by the next theorem:

Theorem 2.5.6. If the constraint functions $g_1, \ldots, g_m$ are concave and if the interior R^o of the constraint set is nonempty, then any feasible solution is qualified.
Proof. Let $x_0 \in R^o$. Then, by theorem 2.5.3, it must be true that $g_i(x_0) > 0$, $i = 1, \ldots, m$. Consider an arbitrary $x \in R$ and define $s_0 = x_0 - x$. For any $i \in A(x)$ we have, by theorem 2.4.13,

$$\nabla g_i(x)^T s_0 \geqslant g_i(x_0) - g_i(x) = g_i(x_0) > 0.$$

Application of theorem 2.1.3 completes the proof.

Theorem 2.5.7. If (a) problem (1.1.1) is a convex-programming problem, (b) the problem functions f, $g_1, \ldots, g_m$ have continuous first-order partial derivatives in E_n, and (c) the interior of the constraint set R is nonempty, then a feasible solution $\bar{x}$ is a minimum solution of (1.1.1) if, and only if, a vector $\bar{u} \in E_m$ exists such that $(\bar{x},\bar{u})$ is a Kuhn–Tucker point.
Proof. The theorem follows easily from a combination of theorems 2.5.5, 2.5.6, and 2.1.2.

For a convex-programming problem the Kuhn–Tucker points can be characterized in a different way. First of all we introduce:

Definition. The *Lagrangian function* associated with problem (1.1.1) is given by

$$L(x,u) = f(x) - \sum_{i=1}^{m} u_i\, g_i(x). \tag{2.5.1}$$

Definition. $E_m^+ = \{u \mid u \in E_m, \quad u \geqslant 0\}.$

Definition. A point $(\bar{x},\bar{u}) \in E_n \times E_m^+$ is a *saddle point* of L in $E_n \times E_m^+$ if

$$L(\bar{x},u) \leqslant L(\bar{x},\bar{u}) \leqslant L(x,\bar{u}) \tag{2.5.2}$$

for any $x \in E_n$ and any $u \in E_m^+$.

Theorem 2.5.8. If (a) problem (1.1.1) is a convex-programming problem, and (b) the problem functions f, $g_1, \ldots, g_m$ have continuous first-order partial derivatives in E_n, then $(\bar{x},\bar{u})$ is a Kuhn–Tucker point of the problem if, and only if, it is a saddle point of the associated Lagrangian function in $E_n \times E_m^+$.
Proof. Let us, first, prove the if-part. If $(\bar{x},\bar{u})$ is a saddle point of the Lagrangian

in $E_n \times E_m{}^+$, then (2.1.8) must hold and (2.1.10) follows easily from the inequality $L(\bar{x},\bar{u}) \leqslant L(x,\bar{u})$. The left-hand inequality $L(\bar{x},u) \leqslant L(\bar{x},\bar{u})$ implies

$$\sum_{i=1}^{m} u_i\, g_i(\bar{x}) \geqslant \sum_{i=1}^{m} \bar{u}_i\, g_i(\bar{x})$$

for any $u \in E_m{}^+$. This can only be true if (2.1.7) and (2.1.9) are satisfied.

Second, we consider the case that $(\bar{x},\bar{u})$ is a Kuhn–Tucker point. Then $\bar{u} \geqslant 0$, and $L(x,\bar{u})$ is a convex function of x the gradient of which vanishes at $\bar{x}$. Hence we obtain $L(\bar{x},\bar{u}) \leqslant L(x,\bar{u})$ for any $x \in E_n$. The relations (2.1.7) and (2.1.9) imply that $L(\bar{x},u) \leqslant L(\bar{x},\bar{u})$ for any $u \in E_m{}^+$.

The last subject to be treated here is a *dual (programming) problem* of (1.1.1). *Duality* in nonlinear programming is a relationship between two problems — one of which, the *primal*, is a constrained-minimization problem and the other, the *dual*, is a constrained-maximization problem — with the following properties.

1. The primal problem has a minimum solution if, and only if, the dual has a maximum solution, and the extreme values are equal.
2. If the constraints of the primal (dual) problem are consistent and those of the dual (primal) are not, then the primal (dual) problem has no finite minimum (maximum).

For convex-programming problems many results in the above sense have been obtained in the last decade. Here the Lagrangian function plays a prominent part.

A dual problem of (1.1.1) is given by

$$\left.\begin{array}{c} \text{maximize } L(x,u) \text{ subject to} \\ \nabla_x L(x,u) = 0 \text{ and} \\ u \geqslant 0, \end{array}\right\} \tag{2.5.3}$$

where $\nabla_x L$ symbolizes the gradient of L with respect to x. Any point (x,u) satisfying the constraints of (2.5.3) is a *dual-feasible solution.* The feasible solutions of the original problem (1.1.1) are referred to as *primal-feasible solutions*. We shall here confine ourselves to the proof of the following two theorems.

Theorem 2.5.9. If (a) problem (1.1.1) is a convex-programming problem, and (b) the problem functions have continuous first-order partial derivatives in E_n, then

$$L(\approx{x},\approx{u}) \leqslant f(\tilde{x}) \tag{2.5.4}$$

for any primal-feasible solution $\tilde{x}$ and any dual-feasible solution $(\approx{x},\approx{u})$.

Proof. Using theorems 2.4.6 and 2.4.13 we obtain

$$f(\overset{\approx}{x}) - \sum_{i=1}^{m} \overset{\approx}{u}_i\, g_i(\overset{\approx}{x}) \leqslant$$

$$\leqslant f(\tilde{x}) + (\overset{\approx}{x} - \tilde{x})^T\, \nabla f(\overset{\approx}{x}) - \sum_{i=1}^{m} \overset{\approx}{u}_i\, g_i(\tilde{x}) - \sum_{i=1}^{m}\ \overset{\approx}{u}_i\, (\overset{\approx}{x} - \tilde{x})^T\, \nabla g_i(\overset{\approx}{x}) =$$

$$= f(\tilde{x}) - \sum_{i=1}^{m} \overset{\approx}{u}_i\, g_i(\tilde{x}) \leqslant f(\tilde{x}).$$

Theorem 2.5.10. If (a) problem (1.1.1) is a convex-programming problem, (b) the problem functions have continuous first-order partial derivatives in E_n, and (c) $\bar{x}$ is a qualified minimum solution of (1.1.1), then the dual problem (2.5.3) has a maximum solution and the extreme values are equal.
Proof. There is a vector $\bar{u} \in E_m$ such that $(\bar{x},\bar{u})$ is a Kuhn–Tucker point. Then $(\bar{x},\bar{u})$ is a dual-feasible solution. The complementary slack relations (2.1.9) imply

$$f(\bar{x}) = L(\bar{x},\bar{u}).$$

Application of the preceding theorem completes the proof.

We shall proceed no further into the duality theory. All the material we need is contained in the last two theorems. We have, in fact, the asymmetric result that a maximum of the dual problem exists if the primal has a minimum. For more details on the symmetry of a pair of dual problems reference may be made to Dantzig, Eisenberg and Cottle (1965).

In what follows, the components of the vector $\bar{u}$ appearing in the Kuhn–Tucker relations will be called *Lagrangian multipliers.* There is a well-known, interesting interpretation: the ith multiplier $\bar{u}_i$ expresses the effect of relaxing the ith constraint on the minimum value of the objective function. More details may be found in Hadley (1964), and Fiacco and McCormick (1968).

3. PARAMETRIC PENALTY-FUNCTION TECHNIQUES

3.1. Mixed parametric penalty functions

A rough sketch of barrier-function techniques and loss-function techniques for solving problem (1.1.1) is contained in sec. 2.1. A detailed analysis will be presented in this chapter.

It is convenient to consider here a *mixed* penalty function so that many properties can be established simultaneously for both classes of techniques. We shall think of the set $I = \{1, \ldots, m\}$ of constraint indices as partitioned into two *disjunct* subsets I_1 and I_2. The partitioning is arbitrary and either I_1 or I_2 may be empty. Furthermore, we introduce

$$R_k = \{x \mid g_i(x) \geqslant 0; \quad i \in I_k\}, \quad k = 1, 2, \tag{3.1.1}$$

so that $R = R_1 \cap R_2$. We shall use $R_1{}^\circ$ to denote the interior of R_1. With these notations we formulate:

Condition 3.1. Problem (1.1.1) is a convex-programming problem. The constraint set R is compact. The set $R_1{}^\circ \cap R_2$ is nonempty.

Under this condition problem (1.1.1) has a minimum solution $\bar{x}$ with minimum value $\bar{v} = f(\bar{x})$, since f is continuous over the nonempty, compact set R. By theorem 2.5.3 we have

$$R_1{}^\circ = \{x \mid g_i(x) > 0; \quad i \in I_1\}. \tag{3.1.2}$$

The set R is the closure of $R_1{}^\circ \cap R_2$. This can easily be demonstrated if we consider the line segment connecting an arbitrary point $x \in R$ with a point $x_0 \in R_1{}^\circ \cap R_2$: then any point, different from x, in the line segment is an element of $R_1{}^\circ \cap R_2$.

The *mixed penalty function* to be considered is given by

$$M_{rs}(x) = f(x) + r^\lambda\, b(x) + s^{-\mu}\, l(x), \tag{3.1.3}$$

which contains a *barrier term* defined by

$$b(x) = -\sum_{i \in I_1} \varphi[g_i(x)], \tag{3.1.4}$$

and a *loss term* defined by

$$l(x) = -\sum_{i \in I_2} \psi[g_i(x)]. \tag{3.1.5}$$

Here, r and s are positive controlling parameters; λ and μ denote positive numbers the choice of which will be discussed in sec. 3.4. The functions φ and ψ appearing in (3.1.4) and (3.1.5) respectively are functions of one variable, say η. We impose the following conditions:

Condition 3.2. The function φ is concave and nondecreasing in the interval $(0,\infty)$, and $\varphi(0+) = -\infty$.

Condition 3.3. The function ψ is concave and nondecreasing in the interval $(-\infty,\infty)$; $\psi(\eta) = 0$ for $\eta \geqslant 0$ and $\psi(\eta) < 0$ for $\eta < 0$.

A partial explanation of these conditions may be found in sec. 1.2. We have imposed them in order to ensure that the mixed penalty function of (3.1.3) has the following, desirable properties.

(a) Preservation of convexity. By theorem 2.4.12, the function M_{rs} is convex in $R_1{}^\circ$ for any $r > 0$ and $s > 0$.

(b) Generation of a barrier. If $\{x_k\}$ denotes a sequence of points in $R_1{}^\circ$ converging to a point in $R_1 - R_1{}^\circ$, then

$$\lim_{k\to\infty} b(x_k) = +\infty. \tag{3.1.6}$$

(c) Penalization of constraint violation. For the loss term we have

$$\left.\begin{aligned} l(x) &= 0, \quad \text{for all} \quad x \in R_2, \\ l(x) &> 0, \quad \text{for all} \quad x \notin R_2. \end{aligned}\right\} \tag{3.1.7}$$

It will immediately be clear that M_{rs} reduces to the barrier function B_r of (1.2.6) if I_2 is empty, and to the loss function L_s of (1.2.7) if I_1 is empty. In the next sections we consider the convergence of mixed-penalty-function techniques. The results so obtained fall apart into similar results for barrier- and loss-function techniques. The introduction of mixed penalty functions might therefore seem to be a purely theoretical trick in order to avoid a separate treatment of barrier and loss functions. In addition to that, however, a mixed penalty function also presents some computational advantages that will be discussed in chapter 5.

3.2. Primal convergence

This section is concerned with the existence and the convergence of points $x(r,s)$ minimizing M_{rs} over $R_1{}^\circ$ for positive values of the controlling parameters r and s. It will intuitively be clear that the *existence* of such a point $x(r,s)$ is rather easy to show if R_1 is compact. This is due to the barrier at the boundary of R_1, generated by the mixed penalty function M_{rs}. Generally, however, R_1 is not compact. We begin by proving the existence of a point minimizing M_{rs} over a *truncation* $R_1{}^\circ \cap S$ such that $R_1 \cap S$ is compact. This is carried out in the following lemma.

Lemma 3.2.1. If (a) the sets R_1 and S are closed subsets of E_n such that $R_1{}^\circ \cap S$ is nonempty and $R_1 \cap S$ compact, (b) the function $h(x)$ is continuous in

$R_1^\circ \cap S$, and (c) for every sequence $\{x_k\}$ in $R_1^\circ \cap S$ converging to a point in $(R_1 - R_1^\circ) \cap S$ it is true that

$$\lim_{k \to \infty} h(x_k) = +\infty,$$

then there exists a point x^* in $R_1^\circ \cap S$ minimizing h over $R_1^\circ \cap S$.
Proof. Consider an arbitrary $w_0 \in R_1^\circ \cap S$ and the set

$$W_0 = \{\, x \mid h(x) \leqslant h(w_0)\,;\, x \in R_1^\circ \,\cap S \,\}.$$

This is a bounded set since it is contained in the compact set $R_1 \cap S$. It is nonempty since $w_0 \in W_0$. In order to show that W_0 is compact we consider a sequence $\{w_k\}$ of points in W_0 converging to a point $\bar{w} \in R_1 \cap S$. Suppose that $\bar{w} \in (R_1 - R_1^\circ) \cap S$. Then

$$\lim_{k \to \infty} h(w_k) = +\infty.$$

On the other hand, $h(w_k) \leqslant h(w_0)$, whereas h is continuous on $R_1^\circ \cap S$. Hence, $\bar{w} \in R_1^\circ \cap S$ and, moreover, $h(\bar{w}) \leqslant h(w_0)$ so that $\bar{w} \in W_0$. Consequently, W_0 is compact and a point $x^* \in W_0$ exists which minimizes h over W_0. From the construction of W_0, however, it follows that x^* is a point minimizing h over $R_1^\circ \cap S$.

Theorem 3.2.1. Let $\{r_k\}$ and $\{s_k\}$ denote monotonic, decreasing null sequences as $k \to \infty$. Under the conditions 3.1 to 3.3 a point $x(r_k,s_k)$ minimizing $M_{r_k s_k}$ over R_1° can be found for k large enough. Any limit point of the sequence $\{x(r_k,s_k)\}$ is a minimum solution of problem (1.1.1).
Proof. To prove this theorem we introduce a perturbation S_2 of R_2 by taking

$$S_2 = \{x \,|g_i(x) \geqslant -a; \quad i \in I_2\}, \tag{3.2.1}$$

where a denotes a positive number. By theorem 2.5.4 the set $R_1 \cap S_2$ is compact since R is compact. Invoking lemma 3.2.1 we find that a point $z(r,s)$ minimizing M_{rs} over $R_1^\circ \cap S_2$ exists for any $r > 0$ and $s > 0$. It is clear that $z(r,s)$ may be a boundary point of S_2.

Now we proceed as follows. We demonstrate that any limit point of the sequence $\{z(r_k,s_k)\}$ is a minimum solution of (1.1.1). This implies that the points $z(r_k,s_k)$ do not belong to the boundary of S_2, but to $R_1^\circ \cap S_2^\circ$, for k large enough. From the construction of S_2 and the convexity of M_{rs} and R_1° it follows that $z(r_k,s_k)$ minimizes $M_{r_k s_k}$ over R_1° for k large enough.

Thus, defining $z_k = z(r_k,s_k)$ we concern ourselves with the convergence of the sequence $\{z_k\}$. A limit point $\bar{z}$ of $\{z_k\}$ exists by the compactness of $R_1 \cap S_2$, and there is a subsequence of $\{z_k\}$ converging to $\bar{z}$. For convenience we also take $\{z_k\}$ to denote this subsequence. Let

$$M_k = M_{r_k s_k},$$

and suppose that $\bar{z} \notin R_2$. Then $l(\bar{z}) > 0$, and we can accordingly write

$$\lim_{k \to \infty} M_k(z_k) = \infty,$$

since $\{f(z_k)\}$ and $\{r_k^\lambda b(z_k)\}$ are bounded below in $R_1^\circ \cap S_2$. On the other hand, there is a point $x_0 \in R_1^\circ \cap R_2$ and we obtain

$$\lim_{k \to \infty} M_k(x_0) = f(x_0) < \infty.$$

This contradicts the statement that z_k minimizes M_k for any k, and hence $\bar{z} \in R_2$. Let us now assume that $f(\bar{z})$ is greater than the minimum value $\bar{v}$ of problem (1.1.1). The set R is the closure of $R_1^\circ \cap R_2$, and hence there is a point $\tilde{x} \in R_1^\circ \cap R_2$ such that

$$f(\bar{z}) > f(\tilde{x}) > \bar{v}.$$

We obtain straightaway

$$\lim_{k \to \infty} M_k(z_k) \geqslant f(\bar{z}) > f(\tilde{x}) = \lim_{k \to \infty} M_k(\tilde{x}).$$

And here we are again led to a contradiction for k large enough. Thus $f(\bar{z}) = \bar{v}$.

The sequence $\{z_k\}$ converges to a minimum solution of (1.1.1). This implies that $z_k \in R_1^\circ \cap S_2^\circ$ for k large enough. Then z_k is an *unconstrained* minimum of M_k or, and this is exactly what we want to show, z_k is a point minimizing M_k over R_1°. Taking $x(r_k,s_k) = z_k$ for k large enough we can complete the proof of the theorem.

Theorem 3.2.2. Let $\bar{v}$ denote the minimum value of problem (1.1.1). Under the conditions of theorem 3.2.1

$$\lim_{k \to \infty} f[x(r_k,s_k)] = \bar{v}, \tag{3.2.2}$$

$$\lim_{k \to \infty} r_k^\lambda b[x(r_k,s_k)] = 0, \tag{3.2.3}$$

$$\lim_{k \to \infty} s_k^{-\mu} l[x(r_k,s_k)] = 0. \tag{3.2.4}$$

Proof. The first formula follows directly from the preceding theorem. In order to show the remaining ones, we shall also be working in the compact set $R_1 \cap S_2$. Here the barrier term is bounded below by a value which we may denote by b_0. Let $x_k = x(r_k,s_k)$. Choose a $\delta > 0$ and a point $\tilde{x} \in R_1^\circ \cap R_2$ such that

$$f(\tilde{x}) < \bar{v} + \delta.$$

Using the property that x_k minimizes M_k over R_1° for k large enough we obtain

$$f(x_k) + r_k^\lambda b(x_k) + s_k^{-\mu} l(x_k) \leqslant f(\tilde{x}) + r_k^\lambda b(\tilde{x}) + s_k^{-\mu} l(\tilde{x}),$$

which leads to

$$f(x_k) + r_k^\lambda\, b(x_k) \leqslant f(\tilde{x}) + r_k^\lambda\, b(\tilde{x}).$$

Hence

$$r_k^\lambda\, b_0 \leqslant r_k^\lambda\, b(x_k) \leqslant f(\tilde{x}) - f(x_k) + r_k^\lambda\, b(\tilde{x}),$$

which proves (3.2.3) since $|f(\tilde{x}) - \bar{v}| < \delta$ and $|f(x_k) - \bar{v}| < \delta$ for k large enough. Moreover,

$$0 \leqslant s_k^{-\mu}\, l(x_k) \leqslant f(\tilde{x}) + r_k^\lambda\, b(\tilde{x}) - f(x_k) - r_k^\lambda\, b(x_k),$$

which can be used to prove (3.2.4).

If the conditions of theorem 3.2.1 are satisfied and if the set R_1 is compact, then a point minimizing M_{rs} over $R_1{}^\circ$ exists for any positive r and s. Computationally, this is a more pleasant situation; it is difficult for a minimization procedure to decide whether a minimum exists or not. We prefer to minimize a penalty function which is *a priori known* to possess an unconstrained minimum in $R_1{}^\circ$.

It is therefore interesting to note that the existence of $x(r,s)$ for any $r > 0$ and $s > 0$ can also be shown if the loss term increases rapidly enough outside the constraint set. This is expressed in the following additional condition.

Condition 3.4. There are positive numbers P and p such that $\psi(\eta) < -P\, |\eta|^{1+p}$ for any $\eta < 0$.

Theorem 3.2.3. Under the conditions 3.1 to 3.4 a point $x(r,s) \in R_1{}^\circ$ minimizing M_{rs} over $R_1{}^\circ$ exists for any $r > 0$ and $s > 0$.
Proof. It is sufficient to show that the set

$$T_0 = \{x\,|M_{rs}(x) \leqslant M_{rs}(w_0);\quad x \in R_1{}^\circ\}$$

is compact for an arbitrarily chosen $w_0 \in R_1{}^\circ$. The proof will be given by contradiction. Assume that T_0 is unbounded. Then a sequence $\{w_k\}$ of points in T_0 can be found such that $||w_k|| \to \infty$ as $k \to \infty$. This can only be true if an $i \in I_2$ exists such that

$$\lim_{k\to\infty} g_i(w_k) = -\infty, \tag{3.2.5}$$

since, by theorem 2.5.4, the set

$$\{x\,|g_i(x) \geqslant -a_i,\quad i \in I_2,\quad x \in R_1\}$$

is bounded for any choice of a_i, $i \in I_2$. Defining F by

$$F(x) = f(x) + r^\lambda\, b(x)$$

we obtain

$$F(w_k) + s^{-\mu}\, l(w_k) = M_{rs}(w_k) \leqslant M_{rs}(w_0).$$

This yields

$$\lim_{k\to\infty} F(w_k) = -\infty,$$

since, by (3.2.5),

$$\lim_{k\to\infty} l(w_k) = +\infty.$$

By lemma 3.2.1, there is a point $v_0 \in R_1{}^\circ \cap R_2$ minimizing F over $R_1{}^\circ \cap R_2$. Now we introduce a perturbation S_2 of R_2 defined by

$$S_2 = \{x \mid g_i(x) \geqslant -a; \quad i \in I_2\},$$

where a denotes a positive number, and we consider the points w_k such that

$$w_k \notin S_2,$$
$$F(w_k) \leqslant F(v_0).$$

Let $\tilde{w}_k$ denote the point where the line segment connecting w_k and v_0 intersects the boundary of $R_1 \cap S_2$. There is an $i_k \in I_2$ such that

$$g_{i_k}(\tilde{w}_k) = -a.$$

Lastly, we introduce a point v_a minimizing F over $R_1{}^\circ \cap S_2$. Such a point exists by theorem 2.5.4 and lemma 3.2.1. Then

$$F(v_0) \geqslant F(v_a).$$

Now, we can write

$$\tilde{w}_k = \lambda_k\, w_k + (1-\lambda_k)v_0, \quad 0 < \lambda_k < 1. \tag{3.2.6}$$

By convexity, $\qquad F(\tilde{w}_k) \leqslant \lambda_k\, F(w_k) + (1-\lambda_k)\, F(v_0),$

whence

$$\begin{aligned} F(w_k) &\geqslant \lambda_k{}^{-1}\, [F(\tilde{w}_k) - (1-\lambda_k)\, F(v_0)] \geqslant \\ &\geqslant \lambda_k{}^{-1}\, [F(v_a) - (1-\lambda_k)\, F(v_0)] = \\ &= F(v_0) - \lambda_k{}^{-1}\, [F(v_0) - F(v_a)]. \end{aligned} \tag{3.2.7}$$

Moreover,

$$0 > -a = g_{i_k}(\tilde{w}_k) \geqslant \lambda_k\, g_{i_k}(w_k) + (1-\lambda_k)\, g_{i_k}(v_0) \geqslant \lambda_k\, g_{i_k}(w_k),$$

since v_0 is feasible. This leads to

$$g_{i_k}(w_k) \leqslant \frac{-a}{\lambda_k}.$$

By condition 3.4, we have

$$l(w_k) = -\sum_{i \in I_2} \psi[g_i(w_k)] \geqslant P\left(\frac{a}{\lambda_k}\right)^{1+p}. \tag{3.2.8}$$

Combination of (3.2.7) and (3.2.8) yields

$$M_{rs}(w_k) = F(w_k) + s^{-\mu}\, l(w_k) \geqslant$$
$$\geqslant F(v_0) - \frac{F(v_0) - F(v_a)}{\lambda_k} + s^{-\mu}\, P\left(\frac{a}{\lambda_k}\right)^{1+p}.$$

It follows from the behaviour of $\{w_k\}$ and from (3.2.6) that

$$\lim_{k \to \infty} \lambda_k = 0,$$

so that

$$\lim_{k \to \infty} M_{rs}(w_k) = +\infty.$$

This, however, contradicts the statement that the points w_k belong to T_0 for any k. Hence, T_0 is bounded, and the proof of theorem 3.2.3 can be completed.

For the case that I_2 is empty we have the following theorem concerning the convergence of *barrier-function techniques.*

Theorem 3.2.4. Let problem (1.1.1) be a convex-programming problem, let the constraint set R be compact and let the interior R° of R be nonempty. Under condition 3.2 a point $x(r)$ minimizing the barrier function (1.2.6) over R° exists for any $r > 0$. Any limit point of the sequence $\{x(r_k)\}$, where $\{r_k\}$ denotes a monotonic, decreasing null sequence, is a minimum solution of (1.1.1). The sequences $\{f\,[x(r_k)]\}$ and $\{b\,[x(r_k)]\}$ are monotonic nonincreasing and nondecreasing respectively.

Proof. We only need to show the last statement. Let f_k denote $f\,[x(r_k)]$ and let $b_k = b[x(r_k)]$. Then

$$f_k + r_k{}^\lambda\, b_k \leqslant f_{k+1} + r_k{}^\lambda\, b_{k+1},$$
$$f_{k+1} + r_{k+1}{}^\lambda\, b_{k+1} \leqslant f_k + r_{k+1}{}^\lambda\, b_k.$$

Adding the first inequality to the second, we obtain

$$(r_k{}^\lambda - r_{k+1}{}^\lambda)\,(b_k - b_{k+1}) \leqslant 0,$$

whence

$$b_k \leqslant b_{k+1}.$$

The inequality

$$f_k \geqslant f_{k+1}$$

is obtained in a similar way.

If I_1 is empty, theorem 3.2.1 and 3.2.3 yield the following theorem concerning the pure loss-function techniques.

Theorem 3.2.5. Let problem (1.1.1) be a convex-programming problem and let the constraint set R be nonempty and compact. Under conditions 3.3 and 3.4 a point $x(s)$ minimizing the loss function (1.2.7) over E_n exists for any $s > 0$. Any limit point of the sequence $\{x(s_k)\}$ where $\{s_k\}$ denotes a monotonic, decreasing null sequence, is a minimum solution of (1.1.1). The sequences $\{f[x(s_k)]\}$ and $\{l[x(s_k)]\}$ are monotonic nondecreasing and nonincreasing, respectively.

A proof of theorem 3.2.5 will be omitted; the monotonicity can be established in a similar way as in theorem 3.2.4.

3.3. Dual convergence

If the mixed penalty function (3.1.3) has continuous first-order partial derivatives in E_n, a solution of the *dual problem* (2.5.3) can easily be constructed, as we shall demonstrate in the present section. We shall, first, impose the following conditions on the functions φ and ψ.

Condition 3.5. The function φ has a continuous first-order derivative φ' in the interval $(0,\infty)$.

Condition 3.6. The function ψ has a continuous first-order derivative ψ' in the interval $(-\infty,\infty)$.

Theorem 3.3.1. If (a) the conditions 3.1 to 3.6 are satisfied, and (b) the problem functions have continuous first-order partial derivatives in E_n, then a feasible solution of the dual problem of (1.1.1) is given by $[x(r,s), u(r,s)]$, where $x(r,s)$ is a point minimizing M_{rs} over $R_1{}^\circ$ for positive r and s, and $u(r,s)$ is taken to be the m vector with components

$$u_i(r,s) = r^\lambda\, \varphi'\{g_i[x(r,s)]\}, \quad i \in I_1, \tag{3.3.1}$$

$$u_i(r,s) = s^{-\mu}\, \psi'\{g_i[x(r,s)]\}, \quad i \in I_2. \tag{3.3.2}$$

Proof. By conditions 3.2, 3.3, 3.5 and 3.6 the functions φ' and ψ' are nonnegative in their respective definition areas. The mixed penalty function M_{rs} possesses continuous first-order partial derivatives in $R_1{}^\circ$. Then the gradient of M_{rs} vanishes at a minimizing point $x(r,s)$, which exists by theorem 3.2.3. Hence

$$\nabla f[x(r,s)] - \sum_{i=1}^{m} u_i(r,s)\, \nabla g_i[x(r,s)] = 0. \tag{3.3.3}$$

Moreover,

$$u_i(r,s) \geqslant 0; \quad i = 1, \ldots, m, \tag{3.3.4}$$

which completes the proof of the theorem.

The next theorem is concerned with values of the dual objective function L (the Lagrangian function defined by (2.5.1)) and their convergence to the minimum value $\bar{v}$ of problem (1.1.1).

Theorem 3.3.2. Let the conditions 3.1 to 3.6 be satisfied and suppose that the problem functions admit of continuous first-order partial derivatives in E_n. Under the additional condition that the ratio

$$\gamma(\eta) = \eta\ \varphi'(\eta)/\varphi(\eta)$$

has a finite limit as $\eta \downarrow 0$, it must be true that

$$\lim_{k\to\infty} L[x(r_k,s_k),\ u(r_k,s_k)] = \bar{v}, \tag{3.3.5}$$

for monotonic, decreasing null sequences $\{r_k\}$ and $\{s_k\}$.
Proof. Defining $x_k = x(r_k,s_k)$ we can use (3.2.3) to obtain

$$\lim_{k\to\infty} r_k^{\lambda} \sum_{i\epsilon A_1} \varphi\{g_i(x_k)\} = 0.$$

It can then be shown that

$$\lim_{k\to\infty} \sum_{i\epsilon I_1} u_i(r_k,s_k)\, g_i(x_k) =$$

$$\lim_{k\to\infty} r_k^{\lambda} \sum_{i\epsilon I_1} \varphi'\{g_i(x_k)\}\, g_i(x_k) =$$

$$\lim_{k\to\infty} r_k^{\lambda} \sum_{i\epsilon A_1} \varphi\{g_i(x_k)\}\, \gamma\{g_i(x_k)\} = 0.$$

Conditions 3.3 and 3.6 imply

$$u_i(r,s) g_i[x(r,s)] \leqslant 0; \quad i \in I_2.$$

By theorem 2.5.9 and the above results we find that

$$\bar{v} \geqslant L[x_k,\ u(r_k,s_k)] \geqslant f(x_k) - \sum_{i\epsilon I_1} u_i(r_k,s_k)\, g_i(x_k).$$

Using (3.2.2) one can now complete the proof of the theorem.

Interesting results from these duality considerations follow for the barrier-function techniques, which operate in the interior of the constraint set. Any point $x(r)$ minimizing the barrier function B_r over R^0 is *primal-feasible*. A *dual-feasible* solution is given by $[x(r), u(r)]$, where $u(r)$ denotes the m vector with components

$$u_i(r) = r^{\lambda}\ \varphi'\{g_i[x(r)]\}; \quad i = 1, \ldots, m.$$

On the basis of (2.5.4) and keeping in mind that $x(r)$ is primal-feasible we can write the inequalities

$$f[x(r)] - E[r,x(r)] \leqslant \bar{v} \leqslant f[x(r)], \tag{3.3.6}$$

where

$$E[r,x(r)] = \sum_{i=1}^{m} u_i(r)\, g_i[x(r)] = r^{\lambda} \sum_{i=1}^{m} \varphi'\{g_i[x(r)]\}\, g_i[x(r)]. \tag{3.3.7}$$

The expression (3.3.7) may be regarded as an *error term*. It is positive for any $r > 0$. Obviously, it is desirable that the error term should be as small as possible, independently of $x(r)$, so that the error in the approximation of $\bar{v}$ can be estimated a priori. Let us try to choose the function φ in such a way that the error term is equal to or smaller than an arbitrary, positive δ. As a matter of course this function has to meet the additional requirements of conditions 3.2 and 3.5. Keeping in mind that (3.3.7) is a sum of nonnegative terms one could also impose the requirement

$$\varphi'\{g_i[x(r)]\}\, g_i[x(r)] \leqslant \frac{r^{-\lambda}\,\delta}{m}\,; \quad i = 1, \ldots, m.$$

A simple function φ satisfying the above inequality is given by

$$\varphi(\eta) = \frac{r^{-\lambda}\,\delta}{m} \ln \eta,$$

which yields, after substitution into (3.3.7),

$$E[r,x(r)] = \delta.$$

Similarly, substitution of

$$\varphi(\eta) = \frac{r^{-\lambda}\,\delta}{m} \ln\left(\frac{\eta}{\eta + 1}\right)$$

into (3.3.7) leads to

$$E[r,x(r)] = \frac{\delta}{m} \sum_{i=1}^{m} \frac{1}{g_i[x(r)] + 1} < \delta.$$

In what follows we shall frequently refer to the *logarithmic barrier function*

$$f(x) - r \sum_{i=1}^{m} \ln g_i(x), \tag{3.3.8}$$

which is obtained by substituting $\varphi(\eta) = \ln \eta$ and $\lambda = 1$ into (1.2.6). Then

(3.3.6) and (3.3.7) reduce to

$$0 \leqslant f\,[x(r)] - \bar{v} \leqslant m\,r. \tag{3.3.9}$$

Thus, the numerical problem of how to choose the parameter r is facilitated considerably; it can be given such a value that $\bar{v}$ is approximated with a prescribed accuracy. This is a particular feature of some *barrier functions*. For more details one is referred to the next section. There we shall discuss a first-order approximation to $f\,[x(r,s)] - \bar{v}$ for the more general case that the mixed penalty function M_{rs} is employed.

We conclude this section by a theorem concerning the dual convergence if the problem admits of a unique minimum $\bar{x}$ with a uniquely determined vector $\bar{u}$ of associated Lagrangian multipliers.

Theorem 3.3.3. If (a) the problem functions f, $g_1, \ldots, g_m$ of (1.1.1) have continuous second-order partial derivatives in E_n, (b) a Kuhn–Tucker point $(\bar{x},\bar{u})$ of (1.1.1) exists which satisfies the Jacobian uniqueness conditions 2.1 to 2.3, and (c) the conditions 3.1 to 3.6 are satisfied, then

$$\lim_{k \to \infty} [x(r_k,s_k),\, u(r_k,s_k)] = (\bar{x},\bar{u}), \tag{3.3.10}$$

for monotonic, decreasing null sequences $\{r_k\}$ and $\{s_k\}$.

Proof. By theorem 2.2.3, $\bar{x}$ is the unique minimum solution of (1.1.1), so that

$$\lim_{k \to \infty} x(r_k,s_k) = \bar{x}.$$

Let us define

$$d_k = \sum_{i=1}^{m} u_i(r_k,s_k) \geqslant 0,$$

and let us assume that

$$\lim_{k \to \infty} d_k = \infty.$$

With the additional definition

$$w_i(r_k s_k,) = u_i(r_k,s_k)\, d_k^{-1}; \quad i = 1, \ldots, m,$$

so that

$$\sum_{i=1}^{m} w_i(r_k,s_k) = 1, \tag{3.3.11}$$

and taking $w(r_k,s_k)$ to denote the m vector with components $w_i(r_k,s_k)$, $i = 1, \ldots, m$, we find that a limit point $\bar{w}$ of the sequence $\{w(r_k,s_k)\}$ exists. Let us take $\{w(r_k,s_k)\}$ to denote a subsequence converging to $\bar{w}$. Using (3.3.3) and dividing by d_k we obtain

$$d_k^{-1} \nabla f[x(r_k,s_k)] - \sum_{i=1}^{m} w_i(r_k,s_k) \nabla g_i[x(r_k,s_k)] = 0.$$

Taking the limit as $k \to \infty$ yields

$$\sum_{i=1}^{m} \bar{w}_i \nabla g_i(\bar{x}) = 0. \tag{3.3.12}$$

It is clear that

$$\lim_{k\to\infty} u_i(r_k,s_k) = 0, \quad i \notin A(\bar{x}),$$

which implies

$$\bar{w}_i = 0, \quad i \in A(\bar{x}).$$

Now, (3.3.12) reduces to

$$\sum_{i\in A(\bar{x})} \bar{w}_i \nabla g_i(\bar{x}) = 0,$$

so that, by condition 2.2, we must have

$$\bar{w}_i = 0, \quad i \in A(\bar{x}),$$

contradicting (3.3.11). Hence, it must be true that the d_k are bounded, and accordingly the sequence $\{u(r_k,s_k)\}$ has a limit point $\tilde{u}$. Using (3.3.3) and taking the limit as $k \to \infty$ we obtain

$$\nabla f(\bar{x}) - \sum_{i\in A(\bar{x})} \tilde{u}_i \nabla g_i(\bar{x}) = 0.$$

From (2.1.10) and condition 2.2 it follows that

$$\tilde{u}_i = \bar{u}_i, \quad i \in A(\bar{x}),$$

whereas

$$\tilde{u}_i = 0 = \bar{u}_i, \quad i \notin A(\bar{x}),$$

which completes the proof of theorem 3.3.3.

3.4. Series expansion of the minimizing function

Let us now turn to the question of how the pair

$$[x(r,s), u(r,s)] \tag{3.4.1}$$

behaves as a function of r and s in a neighbourhood of $(r,s) = (0,0)$. We shall be operating under conditions which, if satisfied, guarantee that problem (1.1.1) has a unique Kuhn–Tucker point $(\bar{x},\bar{u})$. Furthermore, we assume that the problem functions admit of continuous $(k + 1)$th-order partial derivatives $(k \geqslant 1)$ in E_n. It is a matter of course that the conditions 3.1 to 3.6 are satisfied by assumption.

For numerical purposes (extrapolation towards the minimum solution) it is desirable that (3.4.1) should be differentiable in a neighbourhood of $(r,s) = (0,0)$, preferably as many times as the problem functions admit. These requirements lead to additional conditions to be imposed on the functions φ and ψ; for reasons of convenience, however, we formulate conditions involving the first-order derivatives of φ and ψ. The analysis of the present section will eventually lead to an appropriate choice of the parameters λ and μ appearing in (3.1.3).

From (3.3.1) to (3.3.3) we can infer that (3.4.1) solves the system

$$\left.\begin{aligned} \nabla f(x) - \sum_{i=1}^{m} u_i \nabla g_i(x) &= 0, & \\ u_i - r^\lambda \varphi'\{g_i(x)\} &= 0, & i \in I_1, \\ s^\mu u_i - \psi'\{g_i(x)\} &= 0, & i \in I_2, \end{aligned}\right\} \qquad (3.4.2)$$

for any $r > 0$ and $s > 0$. We shall, first, show that we only have to deal with the behaviour of ψ for *nonpositive* values of its argument. By (3.3.10) we have

$$\lim_{k\to\infty} u_i(r_k,s_k) = \bar{u}_i; \quad i = 1, \ldots, m.$$

Then there exist positive numbers ϱ_0 and σ_0 such that $u_i(r,s) > 0$, $i \in A(\bar{x})$, for all $0 < r < \varrho_0$ and $0 < s < \sigma_0$. It follows then from condition 3.3 and from (3.3.2) that $g_i[x(r,s)] < 0$, $i \in A_2(\bar{x})$, for all $0 < r < \varrho_0$ and $0 < s < \sigma_0$. On the other hand, it must be true that $g_i[x(r,s)] > 0$, $i \notin A(\bar{x})$, for sufficiently small, positive values of r and s, whence $u_i(r,s) = 0$, $i \in I_2 - A_2(\bar{x})$. Summarizing the results we find that a positive ϱ and σ exist such that for all $0 < r < \varrho$ and $0 < s < \sigma$

$$\left.\begin{aligned} u_i(r,s) &> 0 \\ g_i[x(r,s)] &< 0 \end{aligned}\right\} \; i \in A_2(\bar{x}), \qquad (3.4.3)$$

$$\left.\begin{aligned} g_i[x(r,s)] &> 0 \\ u_i(r,s) &= 0 \end{aligned}\right\} \; i \in I_2 - A_2(\bar{x}). \qquad (3.4.4)$$

It is now sufficient to confine our attention to the constraints with indices in $I_1 \cap A_2(\bar{x})$, since the behaviour of $u_i(r,s)$, $i \in I_2 - A_2(\bar{x})$ is known. Then we are only concerned with the behaviour of ψ for *nonpositive* values of its argument. We shall accordingly introduce a function ω which has the property

$$\omega(\eta) = \psi(\eta) \quad \text{for} \quad \eta \leqslant 0. \qquad (3.4.5)$$

For numerical purposes we want to establish differentiability of (3.4.1), as many times as the problem functions admit. To that end we shall be using the system (3.4.2), where the derivatives φ' and ψ' (or, in point of fact, the derivatives φ' and ω' if we omit some equations) appear. Hence, we shall impose the additional requirements that φ' and ω' be analytic functions. With these introductory remarks we formulate the following conditions.

Condition 3.7. There is a positive number φ_0 such that φ' is analytic in the interval $(-\varphi_0,\infty)$, except at $\eta = 0$; it has a pole of order λ at $\eta = 0$.

Condition 3.8. There is a positive number ω_0 such that ω' is analytic in the interval $(-\infty,\omega_0)$; it has a zero of order μ at $\eta = 0$.

If these conditions are satisfied we shall say, in what follows, that the barrier term $b(x)$ of (3.1.4) has order λ, and that the loss term $l(x)$ of (3.1.5) has order μ.

Lemma 3.4.1. Conditions 3.2 and 3.7 imply

$$\varphi'(\eta) > 0 \quad \text{for any} \quad \eta > 0.$$

Conditions 3.3 and 3.8 imply

$$\omega'(\eta) > 0 \quad \text{for any} \quad \eta < 0.$$

Proof. It must be true that $\varphi'(\eta) \geqslant 0$ for any $\eta > 0$. Suppose that a positive η_0 exists such that $\varphi'(\eta_0) = 0$. Then, by concavity,

$$\varphi(\eta_0) \geqslant \varphi(\eta) \quad \text{for any} \quad \eta > 0.$$

On the other hand, φ: s a monotonic, nondecreasing function whence

$$\varphi(\eta) = \varphi(\eta_0) \quad \text{for any} \quad \eta \geqslant \eta_0.$$

Then, φ is a constant in the interval $[\eta_0,\infty)$, contradicting the statement that it is an analytic function in the interval $(0,\infty)$ with $\varphi(0+) = -\infty$.

The proof of the second statement proceeds along the same lines, so that it can be omitted.

On the basis of the conditions 3.7 and 3.8 we can write

$$\varphi'(\eta) = \eta^{-\lambda}\, \xi(\eta), \tag{3.4.6}$$

$$\omega'(\eta) = (-\eta)^{\mu}\, \theta(\eta), \tag{3.4.7}$$

where ξ is analytic in the interval $(-\varphi_0,\infty)$, $\xi(0) \neq 0$, and θ is analytic in the interval $(-\infty,\omega_0)$, $\theta(0) \neq 0$. Invoking lemma 3.4.1 we find that

$$\xi(\eta) > 0 \quad \text{for} \quad \eta \geqslant 0,$$

$$\theta(\eta) > 0 \quad \text{for} \quad \eta \leqslant 0.$$

Theorem 3.4.1. If (a) the problem functions f, $g_1, \ldots, g_m$ have continuous $(k+1)$th-order partial derivatives $(k \geqslant 1)$ in E_n, (b) a Kuhn–Tucker point $(\bar{x},\bar{u})$ of problem (1.1.1) exists which satisfies the Jacobian uniqueness conditions 2.1 to 2.3, and (c) the conditions 3.1 to 3.8 are satisfied, then the pair

$[x(r,s), u(r,s)]$ is unique with continuous kth-order partial derivatives in a neighbourhood of $(r,s) = (0,0)$.

Proof. We arrange the constraints in such a way that we obtain

$$A(\bar{x}) = \{1, \ldots, \alpha\},$$

as we have done in the proof of theorem 2.2.3. We take A_1 and A_2 to denote $A_1(\bar{x})$ and $A_2(\bar{x})$ respectively. Finally, we think of the constraints which are inactive at $\bar{x}$ as arranged in such a way that

$$\begin{array}{lll} i \in I_1 & \text{for all} & \alpha + 1 \leqslant i \leqslant \beta, \\ i \in I_2 & \text{for all} & \beta + 1 \leqslant i \leqslant m. \end{array}$$

We have pointed out that the constraints numbered from $\beta + 1$ to m can be dropped from consideration. These are precisely the constraints in $I_2 - A_2(\bar{x})$. Employing these notations we replace the system (3.4.2) by the slightly reduced system

$$\left.\begin{array}{l} \nabla f(x) - \sum\limits_{i=1}^{\beta} u_i \nabla g_i(x) = 0, \\ u_i\, g_i^{\lambda}(x) - r^{\lambda}\, \xi\{g_i(x)\} = 0, \quad i \in I_1, \\ s^{\mu}\, u_i - \{-g_i(x)\}^{\mu}\, \theta\{g_i(x)\} = 0, \quad i \in A_2, \end{array}\right\} \tag{3.4.8}$$

a solution of which is given by

$$[x(r,s), u_1(r,s), \ldots, u_{\beta}(r,s)] \tag{3.4.9}$$

for all $0 < r < \varrho$ and $0 < s < \sigma$. Furthermore, it can be verified that $(\bar{x}, \bar{u}_1, \ldots, \bar{u}_{\beta})$ solves (3.4.8) for $r = 0$ and $s = 0$ so that, if we take

$$\begin{aligned} x(0,0) &= \bar{x}, \\ u(0,0) &= \bar{u}, \end{aligned}$$

we obtain straightaway that (3.4.9) is a solution of (3.4.8) for any $0 \leqslant r < \varrho$ and $0 \leqslant s < \sigma$. With the additional definitions

$$\begin{array}{ll} y_i = u_i^{1/\lambda}, & i \in I_1, \\ y_i = u_i^{1/\mu}, & i \in A_2, \end{array}$$

and with similar definitions of $\bar{y}_i$ and $y_i(r,s)$ for $i \in I_1 \cup A_2$, the system (3.4.8) can be rewritten as

$$\left.\begin{array}{l} \nabla f(x) - \sum\limits_{i \in I_1} y_i^{\lambda} \nabla g_i(x) - \sum\limits_{i \in A_2} y_i^{\mu} \nabla g_i(x) = 0, \\ y_i\, g_i(x) - r\, [\xi\{g_i(x)\}]^{1/\lambda} = 0, \quad i \in I_1, \\ s\, y_i + g_i(x)\, [\theta\{g_i(x)\}]^{1/\mu} = 0, \quad i \in A_2. \end{array}\right\} \tag{3.4.10}$$

For any $0 \leqslant r < \varrho$ and $0 \leqslant s < \sigma$ a solution of (3.4.10) is obviously given by

$$[x(r,s), y(r,s)], \tag{3.4.11}$$

where $y(r,s)$ denotes the vector with components $y_i(r,s)$, $i = 1, \ldots, \beta$. Similarly, we take y and $\bar{y}$ to denote vectors with components $y_1, \ldots, y_\beta$ and $\bar{y}_1, \ldots, \bar{y}_\beta$ respectively. The system (3.4.10) represents a system of $n + \beta$ nonlinear equations involving $n + \beta + 2$ variables: the components of x and y, and the controlling parameters r and s.

The functions appearing in (3.4.10) have continuous kth-order partial derivatives in a neighbourhood of the point $(x, y, r, s) = (\bar{x}, \bar{y}, 0, 0)$. Let J denote the Jacobian matrix of (3.4.10) with respect to x and y evaluated at $(\bar{x}, \bar{y}, 0, 0)$. We shall verify that J is nonsingular under the uniqueness conditions 2.1 to 2.3. In order to do this we introduce a convenient notation.

Let H_1 denote the matrix with the *linearly independent* columns $\nabla g_i(\bar{x})$, $i = 1, \ldots, \alpha$, and H_2 the matrix with the columns $\nabla g_i(\bar{x})$, $i = \alpha + 1, \ldots, \beta$. The matrix G represents a diagonal matrix with *positive* elements $g_i(\bar{x})$, $i = \alpha + 1, \ldots, \beta$. The symbol D denotes the matrix $D(\bar{x},\bar{u})$ of (2.2.1). The matrices Y_1 and Y_3 are taken to be diagonal matrices. The diagonal elements of Y_1 are given by

$$\begin{array}{lll} \lambda\, \bar{y}_i^{\lambda-1}, & \text{if} & i \in A_1, \\ \mu\, \bar{y}_i^{\mu-1}, & \text{if} & i \in A_2. \end{array}$$

The diagonal elements of Y_3 are

$$\begin{array}{lll} \bar{y}_i \quad , & \text{if} & i \in A_1, \\ [\theta(0)]^{1/\mu}, & \text{if} & i \in A_2. \end{array}$$

Lastly, the matrix Y_2 is the unit matrix if $\lambda = 1$, and the null matrix if $\lambda > 1$. With these notations and arrangements the matrix J can be written as

$$J = \begin{pmatrix} D & -H_1\, Y_1 & -H_2\, Y_2 \\ Y_3\, H_1{}^T & 0 & 0 \\ 0 & 0 & G \end{pmatrix}. \tag{3.4.12}$$

Comparison of (3.4.12) and (2.2.4) leads to the conclusion that J must be nonsingular.

By the implicit-function theorem (De la Vallée Poussin (1946)) there is a neighbourhood of $(r,s) = (0,0)$ such that x and y can be solved *uniquely* from the system (3.4.10) in terms of the remaining variables r and s. The solution so obtained has continuous kth-order partial derivatives *at* $(r,s) = (0,0)$. We have already constructed the solution (3.4.11) of the system under consideration, for $0 \leqslant r < \varrho$ and $0 \leqslant s < \sigma$. Hence, there is a neighbourhood of $(r,s) = (0,0)$ such that (3.4.11) is the unique solution of (3.4.10) with continuous kth-order partial derivatives *at* $(r,s) = (0,0)$. Moreover, these derivatives exist and are continuous *in a neighbourhood of* $(r,s) = (0,0)$ since the functions appearing in the system (3.4.10) have continuous kth-order partial derivatives around the point $(\bar{x}, \bar{y}, 0, 0)$. The observation that

$$u_i(r,s) = \begin{cases} [y_i(r,s)]^\lambda, & i \in I_1, \\ [y_i(r,s)]^\mu, & i \in A_2, \\ 0, \; i \in I_2 - A_2, \end{cases}$$

in a sufficiently small neighbourhood of $(r,s) = (0,0)$ completes the proof of theorem 3.4.1.

In what follows we shall refer to the vector function $[x(r,s), u(r,s)]$ of (3.4.1) as the *minimizing function* associated with the mixed penalty function M_{rs} of (3.1.3). It is worth noting that this function is defined in a *full* neighbourhood of the origin; only for sufficiently small, *positive* values of r and s can it be thought of as related to the mixed penalty function M_{rs}.

Let us now discuss a number of examples in order to illustrate the results of theorem 3.4.1. We shall be dealing with the problem

$$\begin{array}{ll} \text{minimize} & 4\,x_1 + x_2 \\ \text{subject to} & x_1 \qquad \geqslant 1, \\ & \qquad x_2 \geqslant 2, \end{array}$$

and we start off with the mixed penalty function

$$4x_1 + x_2 - r^\lambda\, \varphi(x_1 - 1) - s^{-\mu}\, \psi(x_2 - 2). \tag{3.4.13}$$

We take φ and ψ such that

$$\begin{aligned} \varphi'(\eta) &= \eta^{-\lambda}, \\ \omega'(\eta) &= (-\eta)^\mu. \end{aligned}$$

A point $x(r,s)$ minimizing (3.4.13) can then be obtained by solving

$$\begin{array}{ll} 4 - r^\lambda\,(x_1 - 1)^{-\lambda} = 0, & x_1 \geqslant 1, \\ 1 - s^{-\mu}\,(-x_2 + 2)^\mu = 0, & x_2 \leqslant 2, \end{array}$$

which leads to

$$\begin{aligned} x_1(r,s) &= 1 + 4^{-1/\lambda}\, r, \\ x_2(r,s) &= 2 - s. \end{aligned}$$

Here, we have a minimizing function which is clearly differentiable in a neighbourhood of $(r,s) = (0,0)$. The above example can also be used to demonstrate the convenience of raising r and s to the powers λ and μ respectively. Let us, instead of (3.4.13), apply the mixed penalty function

$$4x_1 + x_2 - r^p\, \varphi(x_1 - 1) - s^{-q}\, \psi(x_2 - 2), \tag{3.4.14}$$

where φ and ψ are taken as before, and p and q denote positive numbers. Then $x(r,s)$ can be solved from

$$\begin{array}{ll} 4 - r^p\,(x_1 - 1)^{-\lambda} = 0, & x_1 \geqslant 1, \\ 1 - s^{-q}\,(-x_2 + 2)^\mu = 0, & x_2 \leqslant 2, \end{array}$$

yielding
$$x_1(r,s) = 1 + 4^{-1/\lambda}\, r^{p/\lambda},$$
$$x_2(r,s) = 2 - s^{q/\mu}.$$

In this example the minimizing function is only differentiable at $(r,s) = (0,0)$ if $p \geqslant \lambda$ and $q \geqslant \mu$. The choice $p = \lambda$ and $q = \mu$ is a convenient one, since it leads to an order of differentiability for the minimizing function which is as high as the problem functions admit.

Let us, finally, employ the mixed penalty function (3.4.14) with φ and ψ chosen in such a way that

$$\varphi'(\eta) = \omega'(\eta) = \exp\left(\frac{1}{\eta}\right).$$

Then φ and ψ satisfy the requirement of conditions 3.2, 3.3, 3.5 and 3.6, but not of 3.7 and 3.8. We solve $x(r,s)$ from the system

$$4 - r^p \exp\left(\frac{1}{x_1 - 1}\right) = 0, \qquad x_1 \geqslant 1,$$

$$1 - s^{-q} \exp\left(\frac{1}{x_2 - 2}\right) = 0, \quad x_2 \leqslant 2.$$

In so doing, we obtain

$$x_1(r,s) = 1 + \frac{1}{\ln 4 - p \ln r},$$

$$x_2(r,s) = 2 + \frac{1}{q \ln s}.$$

Here, the minimizing function is *not* differentiable at $(r,s) = (0,0)$, for any positive value of p and q.

We conclude this section by discussing a first-order approximation to the expression

$$f\,[x(r,s)] - f(\bar{x}).$$

Observing that the minimizing function solves (3.4.8) we obtain

$$\frac{g_i[x(r,s)]}{r} = \left(\frac{\xi\{g_i[x(r,s)]\}}{u_i(r,s)}\right)^{1/\lambda}, \quad i \in I_1,$$

$$\frac{-g_i[x(r,s)]}{s} = \left(\frac{u_i(r,s)}{\theta\{g_i[x(r,s)]\}}\right)^{1/\mu}, \quad i \in A_2.$$

We can now obtain

$$\lim_{r \downarrow 0} \frac{g_i[x(r,0)] - g_i(\bar{x})}{r} = \left(\frac{\xi(0)}{\bar{u}_i}\right)^{1/\lambda}, \quad i \in A_1,$$

or, equivalently,

$$\nabla g_i(\bar{x})^T \frac{\partial x(0,0)}{\partial r} = \left(\frac{\xi(0)}{\bar{u}_i}\right)^{1/\lambda}, \quad i \in A_1.$$

Similarly,

$$\nabla g_i(\bar{x})^T \frac{\partial x(0,0)}{\partial s} = 0, \quad i \in A_1,$$

$$\nabla g_i(\bar{x})^T \frac{\partial x(0,0)}{\partial r} = 0, \quad i \in A_2,$$

$$\nabla g_i(\bar{x})^T \frac{\partial x(0,0)}{\partial s} = -\left(\frac{\bar{u}_i}{\theta(0)}\right)^{1/\mu}, \quad i \in A_2.$$

With the above formulas and the Kuhn–Tucker relations we can write

$$f\,[x(r,s)] - f(\bar{x}) \simeq \nabla f(\bar{x})^T \left[r \frac{\partial x(0,0)}{\partial r} + s \frac{\partial x(0,0)}{\partial s} \right] =$$

$$= \sum_{i=1}^{m} \bar{u}_i \nabla g_i(\bar{x})^T \left[r \frac{\partial x(0,0)}{\partial r} + s \frac{\partial x(0,0)}{\partial s} \right] =$$

$$= r \sum_{i \epsilon A_1} \bar{u}_i \left(\frac{\xi(0)}{\bar{u}_i}\right)^{1/\lambda} - s \sum_{i \epsilon A_2} \bar{u}_i \left(\frac{\bar{u}_i}{\theta(0)}\right)^{1/\mu}. \tag{3.4.15}$$

It is interesting to consider the logarithmic barrier function (3.3.8). Then A_2 is empty, $\lambda = 1$, and $\xi(\eta) = 1$ for all η. Hence, if $x(r)$ denotes the minimizing function associated with (3.3.8), then (3.4.15) reduces to

$$f\,[x(r)] - f(\bar{x}) \simeq \alpha\, r,$$

where α stands for the *number* of active constraints at $\bar{x}$. One may compare this approximation with (3.3.9). The property that the minimum value of (1.1.1) can be approximated with a prescribed accuracy is apparently a particular property of first-order barrier-function techniques (where A_2 is empty and $\lambda = 1$). For the remaining methods the first-order approximation (3.4.15) depends on the Lagrangian multipliers, which are generally unknown before the problem is solved.

In view of the results obtained so far, there is virtually no need for using *two* separate controlling parameters. In the considerations to follow we shall accordingly be dealing with the mixed penalty function

$$M_r(x) = f(x) + r^{\lambda}\, b(x) + r^{-\mu}\, l(x), \tag{3.4.16}$$

where $b(x)$ and $l(x)$ denote the barrier term (3.1.4) and the loss term (3.1.5) respectively. We take $x(r)$ to denote a point minimizing M_r over $R_1{}^{\circ}$. The minimizing function associated with (3.4.16) will be represented by $[x(r), u(r)]$, where $u(r)$ is the m vector with components

$$u_i(r) = \begin{cases} r^{\lambda}\, \varphi'\{g_i[x(r)]\}, & i \in I_1, \\ r^{-\mu}\, \psi'\{g_i[x(r)]\}, & i \in I_2. \end{cases} \tag{3.4.17}$$

In what follows we shall refer to the vector function $[x(r), u(r)]$ as the *minimizing trajectory.*

Now, a consequence of theorem 3.4.1 is that the minimizing trajectory can be expanded in a Taylor series about $r = 0$. This provides, as an important numerical application, a basis for extrapolation towards $(\bar{x},\bar{u})$. For more details we may, for instance, refer to Bulirsch (1964), Bulirsch and Stoer (1964, 1966) and Veltkamp (1969).

The results of this section suggest that *first-order* barrier and loss terms in the mixed penalty function (3.4.16) are preferable to higher-order terms, since they provide a more rapid convergence. The approximation (3.4.15), namely, reduces to

$$f\,[x(r)] - f\,(\bar{x}) \simeq r \left[\sum_{i \in A_1} \bar{u}_i \left(\frac{\xi(0)}{\bar{u}_i} \right)^{1/\lambda} - \sum_{i \in A_2} \bar{u}_i \left(\frac{\bar{u}_i}{\theta(0)} \right)^{1/\mu} \right].$$

Thus, it varies with r for small values of r, whereas the controlling parameter in (3.4.16) is raised to the powers λ and μ respectively. The situation is more complicated, however. The next section is concerned with the question of whether the orders λ and μ affect the degree of difficulty in minimizing the penalty function (3.4.16). A discussion of the choice of a penalty function for computational purposes is postponed until that subject is reached in chapter 5.

In order to simplify matters we restrict ourselves henceforth to functions φ and ψ such that

$$\varphi'(\eta) = \eta^{-\lambda}, \tag{3.4.18}$$

$$\omega'(\eta) = (-\eta)^{\mu}. \tag{3.4.19}$$

The results of the sections to follow can, however, be generalized and applied to the cases where (3.4.6) and (3.4.7) are used.

3.5. Eigenvalues of the principal Hessian matrix

Numerically, problem (1.1.1) can be solved by unconstrained minimization of a penalty function for a sequence of positive, decreasing values of the controlling parameter. It is obvious that computational success depends critically on the power of unconstrained-minimization techniques. This, however, introduces the question of whether we can facilitate the computational process by an appropriate choice of the orders λ and μ of the barrier and the loss term respectively.

In this section our concern will acordingly be the *Hessian matrix* of a penalty function, and particularly its eigenvalues, in the limiting case where r decreases to 0. The motivation for the study is the idea that failures of unconstrained-minimization techniques may be due to ill-conditioning (for symmetric, positive definite matrices an excessive ratio of the greatest to the smallest eigenvalue) of the Hessian matrix at some iteration points. The idea is quite plausible for the Newton–Raphson technique: here, the Hessian matrix is evaluated at the current iteration point; thereafter a system of linear equations, *with the Hessian matrix as coefficient matrix*, is solved in order to obtain the direction to the next iteration point. The successful Davidon–Fletcher–Powell algorithm (Davidon (1959), Fletcher and Powell (1963)) and the related quasi-Newton or variable-metric methods (Broyden (1967), Fiacco and McCormick (1968), Pearson (1969)) may also be affected by ill-conditioning (Murray (1969)). In these methods the Hessian matrix is not explicitly evaluated. In every iteration a so-called direction matrix is updated on the ground of information which is due to the difference of two successive iteration points (change in position) and the difference of the corresponding gradients of the function to be minimized. The updating is such that, if a quadratic function of n variables is minimized, the direction matrix equals the inverse Hessian matrix after n iterations. Ill-conditioning was discussed by Bard (1968) who investigated a numerical instability arising if the difference of two successive iteration points *is very large or very small with respect to* the difference of the corresponding gradients. A successful attempt, however, to analyze the effect of ill-conditioning on the iterative course of the above methods (these are probably the most efficient ones) has never been made, at least to our knowledge. It is nevertheless interesting to deal with the question of conditioning of penalty functions: we shall presently demonstrate that a certain condition number varies with r^{-1}, for small values of r, independently of the behaviour of a mixed penalty function at the boundary of the constraint set. It is therefore unlikely that some penalty functions would generally be easier or harder to minimize than other ones.

We shall be assuming that the functions in problem (1.1.1) possess continuous third-order partial derivatives in E_n, and that a Kuhn–Tucker point $(\bar{x},\bar{u})$ exists satisfying the Jacobian uniqueness conditions 2.1 to 2.3. Furthermore, the conditions 3.1 to 3.8 are satisfied by assumption.

We shall primarily be concerned with the Hessian matrix $H(r)$ of the mixed penalty function M_r of (3.4.16), evaluated at the point $x(r)$ which minimizes M_r over $R_1{}^{\circ}$; for small values of r, the point $x(r)$ is unique by theorem 3.4.1. In what follows we shall refer to $H(r)$ as the *principal* Hessian matrix of M_r. Since any method for minimizing M_r approaches $x(r)$ it is reasonable to assume that unconstrained minimization may be obstructed by ill-conditioning of $H(r)$.

Conditioning of a matrix is measured by the *condition number*: for symmetric, positive definite matrices defined as the ratio of the greatest to the smallest eigenvalue. We are particularly interested in variations of the condition number $\chi(r)$ of $H(r)$ in the case where r decreases to 0.

Let us now proceed to the analysis of $H(r)$. We recall from (3.4.3) and (3.4.4) that a positive number ϱ exists such that for $0 \leqslant r < \varrho$:

$$\left.\begin{aligned} u_i(r) &> 0 \\ g_i[x(r)] &< 0 \end{aligned}\right\} \; i \in A_2, \tag{3.5.1}$$

$$\left.\begin{aligned} g_i[x(r)] &> 0 \\ u_i(r) &= 0 \end{aligned}\right\} \; i \in I_2 - A_2. \tag{3.5.2}$$

Furthermore, we introduce the rank-one matrices

$$N_i(x) = \nabla g_i(x) \nabla g_i(x)^T, \quad i \in I. \tag{3.5.3}$$

For any r, $0 < r < \varrho$, the principal Hessian matrix $H(r)$ of M_r can now be written as

$$\begin{aligned} H(r) = D[x(r), u(r)] - r^{\lambda} \sum_{i \in I_1} \varphi''\{g_i[x(r)]\} N_i[x(r)] + \\ - r^{-\mu} \sum_{i \in A_2} \omega''\{g_i[x(r)]\} N_i[x(r)], \end{aligned} \tag{3.5.4}$$

where D is the matrix defined by (2.2.1) and φ'' and ω'' represent the second-order derivatives of φ and ω respectively. We can infer from (3.4.18) and (3.4.19) that

$$H(r) = D[x(r), u(r)] + r^{-1} G[x(r), u(r)] \tag{3.5.5}$$

with

$$G(x,u) = \lambda \sum_{i \in I_1} u_i^{1+1/\lambda} N_i(x) + \mu \sum_{i \in A_2} u_i^{1-1/\mu} N_i(x). \tag{3.5.6}$$

Using the assumption that the functions in problem (1.1.1) admit of continuous third-order partial derivatives, we can expand the elements of $D[x(r), u(r)]$ and $G[x(r), u(r)]$ in a Taylor series about $r = 0$. Hence

$$D[x(r), u(r)] = D(\bar{x},\bar{u}) + r\, D_1(r), \tag{3.5.7}$$

$$G[x(r), u(r)] = G(\bar{x},\bar{u}) + r\, G_1(0) + \tfrac{1}{2} r^2\, G_2(r). \tag{3.5.8}$$

In the above expressions $D_1(r)$ and $G_2(r)$ denote matrices with elements which are due to truncation of the Taylor series expansions; furthermore, $G_1(r)$ is defined by

$$G_1(r) = \frac{\mathrm{d}}{\mathrm{d}r} G[x(r), u(r)]. \tag{3.5.9}$$

If we take $[x'(r), u'(r)]$ to denote the first-order derivative of the minimizing trajectory, then

$$G_1(r) = (\lambda + 1) \sum_{i \epsilon I_1} [u_i(r)]^{1/\lambda} u_i'(r) N_i[x(r)] +$$

$$+ \lambda \sum_{i \epsilon I_1} [u_i(r)]^{1+1/\lambda} \frac{\mathrm{d}}{\mathrm{d}r} N_i[x(r)] +$$

$$+ (\mu - 1) \sum_{i \epsilon A_2} [u_i(r)]^{-1/\mu} u_i'(r) N_i[x(r)] +$$

$$+ \mu \sum_{i \epsilon A_2} [u_i(r)]^{1-1/\mu} \frac{\mathrm{d}}{\mathrm{d}r} N_i[x(r)], \tag{3.5.10}$$

where

$$\frac{\mathrm{d}}{\mathrm{d}r} N_i[x(r)] = \nabla^2 g_i[x(r)]\, x'(r) \nabla g_i[x(r)]^T +$$

$$+ \nabla g_i[x(r)] x'(r)^T \nabla^2 g_i[\, x(r)]. \tag{3.5.11}$$

For convenience, we shall henceforth employ the symbol F to denote briefly the matrix $G_1(0)$. Furthermore, we define a matrix $K(r)$ by

$$K(r) = D_1(r) + \tfrac{1}{2}\, G_2(r). \tag{3.5.12}$$

The matrix $K(r)$ has a finite limiting matrix as $r \downarrow 0$. Substituting (3.5.7), (3.5.8) and (3.5.12) into (3.5.5) we obtain straightaway

$$H(r) = D(\bar{x},\bar{u}) + G_1(0) + r^{-1}\, G(\bar{x},\bar{u}) + r\, [D_1(r) + \tfrac{1}{2}\, G_2(r)] =$$

$$= D(\bar{x},\bar{u}) + F + r^{-1}\, G(\bar{x},\bar{u}) + r\, K(r). \tag{3.5.13}$$

The matrices in (3.5.13) are real, symmetric matrices so that their eigenvalues are real. Furthermore, one can verify that

$$y^T\, D(\bar{x},\bar{u})\, y > 0, \tag{3.5.14}$$

$$y^T\, F\, y = 0, \tag{3.5.15}$$

$$G(\bar{x},\bar{u})\, y = 0, \tag{3.5.16}$$

for any vector $y \in E_n$ satisfying

$$y^T \nabla g_i(\bar{x}) = 0, \quad i \in A(\bar{x}).$$

This is due to condition 2.3 (satisfied by assumption) and to the particular form of $F = G_1(0)$ and $G(\bar{x},\bar{u})$, involving only the (linearly independent) gradients of the constraints which are *active* at $\bar{x}$. Lastly, $G(\bar{x},\bar{u})$ is a matrix with rank α.

The eigenvalues of a matrix are not affected by a coordinate transformation. We use this property in order to transform the matrices $H(r)$, $D(\bar{x},\bar{u})$, $F = G_1(0)$, $G(\bar{x},\bar{u})$ and $K(r)$ into matrices $H^*(r)$, D^*, F^*, G^* and $K^*(r)$ respectively, in such a way that G^* is a diagonal matrix. The new coordinate system can be characterized as follows. We take W to denote the subspace of E_n spanned by the gradients $\nabla g_i(\bar{x})$, $i = 1, \ldots, \alpha$. The symbol Z represents the orthogonal complement of W in E_n. Let $w_1, \ldots, w_\alpha$ denote normalized, orthogonal eigenvectors corresponding to the *positive* eigenvalues of $G(\bar{x},\bar{u})$. These eigenvectors span the subspace W. Finally, we take $z_{\alpha+1}, \ldots, z_n$ to represent a set of normalized, orthogonal vectors spanning the subspace Z. With the vectors $w_1, \ldots, w_\alpha, z_{\alpha+1}, \ldots, z_n$ as a new coordinate system, we find that $H^*(r)$ can be written as

$$\begin{pmatrix} D_{11}^* & D_{12}^* \\ D_{21}^* & D_{22}^* \end{pmatrix} + \begin{pmatrix} F_{11}^* & F_{12}^* \\ F_{21}^* & 0 \end{pmatrix} + r^{-1} \begin{pmatrix} G_{11}^* & 0 \\ 0 & 0 \end{pmatrix} + r\, K^*(r). \tag{3.5.17}$$

Here, G_{11}^* is a diagonal matrix with α rows and α columns, and *positive* diagonal elements g_{ii}^*, $i = 1, \ldots, \alpha$. The partitioning of D^* and F^* is similar to that of G^* so that D_{11}^* has α rows and α columns, etc. The vanishing of F_{22}^* needs some explanation. By (3.5.15), F_{22}^* is antisymmetric, but we have seen that F, and consequently F^*, is symmetric; from these arguments $F_{22}^* = 0$. We still have some degree of freedom in the choice of the coordinate axes $z_{\alpha+1}, \ldots, z_n$, and we shall take them to be normalized, orthogonal eigenvectors of D_{22}^*. Then D_{22}^* is a diagonal matrix with diagonal elements d_{ii}^*, $i = \alpha + 1, \ldots, n$. By (3.5.14), these elements must be *positive.*

We are now in a position to apply Gerschgorin's theorem (Wilkinson (1965)) which states that any eigenvalue of a real or complex matrix A is contained in one of the circular disks with centre a_{ii} and radius $\sum\limits_{i \neq j} |a_{ij}|$; if s of these disks form a connected set isolated from the remaining disks, then there are exactly s eigenvalues of A within this connected domain. The disks in question are commonly referred to as "Gerschgorin disks". We are, however, concerned with real, symmetric matrices so that we can restrict ourselves to "Gerschgorin intervals".

A second, useful device is obtained from the following observation: if the ith column of a matrix A is multiplied by some number $p \neq 0$, and the ith row by p^{-1}, then the eigenvalues of A remain unchanged.

We shall now demonstrate that the eigenvalues of $H^*(r)$, and consequently the eigenvalues of $H(r)$, are given by

$$\left.\begin{aligned} r^{-1}\, g_{ii}^* + \varepsilon_i(r), &\quad i = 1, \ldots, \alpha, \\ d_{ii}^* + \varepsilon_i(r), &\quad i = \alpha + 1, \ldots, n, \end{aligned}\right\} \tag{3.5.18}$$

where

$$\lim_{r \downarrow 0} r\, \varepsilon_i(r) = 0, \quad i = 1, \ldots, \alpha,$$

$$\lim_{r \downarrow 0} \varepsilon_i(r) = 0, \quad i = \alpha + 1, \ldots, n.$$

To start with, we infer from (3.5.17) that the diagonal elements of $H^*(r)$ can be written as

$$h_{ii}^*(r) = \begin{cases} d_{ii}^* + f_{ii}^* + r^{-1}\, g_{ii}^* + r\, k_{ii}^*(r), & i = 1, \ldots, \alpha, \\ d_{ii}^* + r\, k_{ii}^*(r), & i = \alpha + 1, \ldots, n. \end{cases}$$

Next, we try to obtain Gerschgorin intervals with radii which are small with respect to the centres $h_{ii}^*(r)$, $i = 1, \ldots, n$. If we multiply the rows $\alpha + 1$ to n of $H^*(r)$ by $r^{1/2}$ and the columns $\alpha + 1$ to n by $r^{-1/2}$, then $H^*(r)$ is reduced to a matrix possessing Gerschgorin intervals $\Gamma_i(r)$ with centres $h_{ii}^*(r)$ and radii $\varrho_i(r)$ given by

$$\varrho_i(r) = \sum_{\substack{j=1 \\ j \neq i}}^{\alpha} |d_{ij}^* + f_{ij}^* + r\, k_{ij}^*(r)| + r^{-1/2} \sum_{j=\alpha+1}^{n} |d_{ij}^* + f_{ij}^* + r\, k_{ij}^*(r)|;$$

$$i = 1, \ldots, \alpha,$$

$$\varrho_i(r) = r^{1/2} \sum_{j=1}^{\alpha} |d_{ij}^* + f_{ij}^* + r\, k_{ij}^*(r)| + \sum_{\substack{j=\alpha+1 \\ j \neq i}}^{n} |r\, k_{ij}^*(r)|; \quad i = \alpha + 1, \ldots, n.$$

It is obvious that

$$\lim_{r \downarrow 0} r\, \varrho_i(r) = 0; \quad i = 1, \ldots, \alpha,$$

$$\lim_{r \downarrow 0} \varrho_i(r) = 0; \quad i = \alpha + 1, \ldots, n.$$

Any eigenvalue of $H^*(r)$ is contained in at least one of the intervals $\Gamma_i(r)$. If the diagonal elements g_{ii}^*, $i = 1, \ldots, \alpha$, and d_{ii}^*, $i = \alpha + 1, \ldots, n$ are mutually different, then the intervals $\Gamma_i(r)$ are disjunct for r small enough, and this can readily be used to prove (3.5.18). If two or more of the values just named coincide, one only has to consider a number of connected domains (unions of some intervals $\Gamma_i(r)$) in order to establish (3.5.18). The mode of operation will be clear so that detailed calculations can be omitted.

Suppose that the diagonal elements of $H^*(r)$ are arranged in such a way that

$$g_{11}^* \geqslant g_{22}^* \geqslant \ldots \geqslant g_{\alpha\alpha}^*,$$

$$d_{\alpha+1,\alpha+1}^* \geqslant \ldots \geqslant d_{nn}^*.$$

For sufficiently small values of r and $1 \leqslant \alpha < n$ the condition number $\chi(r)$ of $H(r)$ is given by

$$\chi(r) = \frac{r^{-1}\, g_{11}^* + \varepsilon_1(r)}{d_{nn}^* + \varepsilon_n(r)} = r^{-1} \frac{g_{11}^* + r\, \varepsilon_1(r)}{d_{nn}^* + \varepsilon_n(r)} \simeq r^{-1} \frac{g_{11}^*}{d_{nn}^*}, \qquad (3.5.19)$$

so that the condition number is proportional to r^{-1}, *regardless of the orders λ and μ of the barrier and the loss term respectively.* Similarly, the determinant of $H(r)$, which can be written as the product of the eigenvalues, varies with $r^{-\alpha}$ where α stands for the number of active constraints at $\bar{x}$. If $\alpha = n$, then $\chi(r)$ converges to the finite value

$$\frac{g_{11}{}^*}{g_{nn}{}^*}$$

as r decreases to 0; this may happen if, for example, the problem is one of linear programming with a nondegenerate minimum solution.

The behaviour of $\chi(r)$ is an indication that first-order penalty functions are not easier or harder to minimize than the higher-order ones, *provided that the condition number is an appropriate measure of the degree of difficulty.* The last hypothesis has not been thoroughly investigated and it is beyond the scope of the present thesis to do so. We only want to show that, in choosing values of λ and μ for computational purposes, one does not run up against difficulties which may be due to an *excessive* rate of ill-conditioning: we have obtained that conditioning varies with r^{-1} for *any* choice of λ and μ, and for *any* partitioning of the set of constraint indices into subsets I_1 and I_2. We have in fact the even stronger result that α eigenvalues vary with r^{-1} and that the remaining eigenvalues converge to finite, positive values as r decreases to 0, *independently of λ, μ, I_1 and I_2.*

For similar, speculative reasons it is interesting to analyze the "coefficient" $G(\bar{x},\bar{u})$ of r^{-1} in formula (3.5.13), and its positive eigenvalues $g_{ii}{}^*$, $i = 1, \ldots, \alpha$. A penalty function, namely, is designed to *identify* the active constraints and to *solve* them. Presumably, solution of the active constraints is easier if the eigenvalues $g_{ii}{}^*$, $i = 1, \ldots, \alpha$, are of the same order of magnitude. We concern ourselves with the manner in which these eigenvalues are affected by the choice of λ and μ. The matrix $G(\bar{x},\bar{u})$ can, by (3.5.6), be written as

$$G(\bar{x},\bar{u}) = \lambda \sum_{i \epsilon A_1} (\bar{u}_i)^{1+1/\lambda} N_i(\bar{x}) + \mu \sum_{i \epsilon A_2} (\bar{u}_i)^{1-1/\mu} N_i(\bar{x}).$$

We focus our attention on the case where the gradients $\nabla g_i(\bar{x})$, $i = 1, \ldots, \alpha$, are orthogonal. This is not the general case, but the orthogonality hypothesis leads to an interesting interpretation. It follows that the positive eigenvalues of $G(\bar{x},\bar{u})$ are then given by

$$\left.\begin{array}{ll} \lambda\, (\bar{u}_i)^{1+1/\lambda}\, \|\nabla g_i(\bar{x})\|^2, & i \in A_1, \\ \mu\, (\bar{u}_i)^{1-1/\mu}\, \|\nabla g_i(\bar{x})\|^2, & i \in A_2. \end{array}\right\} \qquad (3.5.20)$$

Using the Kuhn–Tucker relations (2.1.10) and the orthogonality relations one can show that

$$\bar{u}_i = \frac{\beta_i \, \|\nabla f(\bar{x})\|}{\|\nabla g_i(\bar{x})\|}, \quad i = 1, \ldots, \alpha,$$

where β_i stands for the cosine of the angle between $\nabla f(\bar{x})$ and $\nabla g_i(\bar{x})$. The positive eigenvalues g_{ii}^* of $G(\bar{x},\bar{u})$ can then be reduced to

$$\left.\begin{array}{ll} \lambda\,(\beta_i\,||\nabla f(\bar{x})||)^{1+1/\lambda}\,||\nabla g_i(\bar{x})||^{1-1/\lambda}, & i \in A_1, \\ \mu\,(\beta_i\,||\nabla f(\bar{x})||)^{1-1/\mu}\,||\nabla g_i(\bar{x})||^{1+1/\mu}, & i \in A_2. \end{array}\right\} \qquad (3.5.21)$$

General remarks can hardly be made, but we restrict ourselves to the first-order case where $\lambda = \mu = 1$. Then the eigenvalues are

$$\left.\begin{array}{rl} \beta_i^2\,||\nabla f(\bar{x})||^2, & i \in A_1, \\ ||\nabla g_i(\bar{x})||^2, & i \in A_2. \end{array}\right\} \qquad (3.5.22)$$

This demonstrates that the eigenvalues corresponding to constraints which are incorporated in a first-order *barrier* term depend on *angles* between gradients only. Eigenvalues corresponding to constraints in a first-order *loss* term are determined by *lengths* of gradients. Such a complete separation is apparently not a feature of higher-order penalty functions. For increasing values of λ and μ they only show the tendency of attaching similar weights to angles and lengths of the gradients involved.

We are thus brought back to the question of whether a barrier function is harder to minimize than a loss function. This point has also been discussed by Murray (1967), and Fiacco and McCormick (1968). They have some preference for (first-order) barrier functions. In view of the results in this section, however, we cannot give an answer which is favourable to either first-order barrier functions or loss functions. We have been dealing with a phenomenon which we may call *constraint balance*: any of the active constraints is associated with precisely one of the positive eigenvalues of $G(\bar{x},\bar{u})$ which we want to equilibrate. Apparently, a particular problem may be highly unbalanced with respect to a first-order barrier function, but conveniently balanced with respect to a first-order loss function, and vice versa.

These results are obtained, it is true, on the assumption that the gradients $\nabla g_i(\bar{x})$, $i = 1, \ldots, \alpha$, are orthogonal. The continuity of the eigenvalues, however, ensures that small perturbations of the orthogonality will only lead to small deviations of the eigenvalues from (3.5.21) and (3.5.22), so that the results just sketched have a slightly wider validity.

4. PENALTY-FUNCTION TECHNIQUES WITH MOVING TRUNCATIONS

4.1. Basic concepts

The preceding chapter was concerned with penalty functions containing one or two parameters which control the convergence of a computational process to a minimum solution of problem (1.1.1). Recently, however, several authors (see sec. 1.1) have observed that parametric barrier-function and loss-function techniques can be modified into methods which do not (explicitly) operate with controlling parameters, but with *moving truncations of the constraint set*. We shall not refer to these methods by the usual name of "parameter-free" versions. In our opinion this name is misleading. Convergence to a minimum solution of (1.1.1) is here controlled by a sequence $\{t_k\}$ of *truncation levels* converging to the unknown minimum value $\bar{v}$ of the problem. In using a parametric technique, however, one employs a null sequence $\{r_k\}$ of values assigned to the controlling parameter.

We shall presently see that the methods of this chapter have a close relationship with Huard's method of centres: it provides the moving-truncations counterpart of the logarithmic barrier-function technique.

We begin by imposing a number of requirements which are summarized in:

Condition 4.1. Problem (1.1.1) is a convex-programming problem. The constraint set R is compact and its interior R^0 is nonempty.

Throughout this chapter we shall be dealing with barrier functions and loss functions *separately*. We did not succeed in finding a moving-truncations version of the mixed-penalty-function technique treated in the previous chapter.

Let us first introduce some terminology and sketch the basic ideas. We shall be operating in intersections of the constraint set R and the *truncations*

$$F(t) = \{x \mid f(x) \leqslant t; \quad x \in E_n\}$$

for values of the *truncation level t* which are not less than the minimum value $\bar{v}$ of problem (1.1.1). Then the *truncated constraint set*

$$T(t) = R \cap F(t) = \{x \mid f(x) \leqslant t; \quad x \in R\} \tag{4.1.1}$$

is nonempty. We define a *moving-truncations barrier function* by

$$B_t{}^*(x) = p\,\varphi[t - f(x)] + \sum_{i=1}^{m} \varphi[g_i(x)], \tag{4.1.2}$$

where φ is a function of one variable satisfying condition 3.2 and possibly 3.7. Furthermore, a positive weight factor p is attached to the term which contains the objective function, for reasons that will become clear at the end of sec. 4.3.

If $t > \bar{v}$, the interior $T^{\circ}(t)$ of $T(t)$ is nonempty, and the function $B_t{}^*$ is apparently concave in $T^{\circ}(t)$. Moreover, if $\{y_j\}$ denotes a sequence of points in $T^{\circ}(t)$ converging to a boundary point of $T(t)$, then

$$\lim_{j \to \infty} B_t{}^*(y_j) = -\infty.$$

Under these conditions a point $c(t)$ exists which maximizes $B_t{}^*$ over $T^{\circ}(t)$. The proof will presently be given. Let $\{t_k\}$ denote a sequence of monotonic, decreasing truncation levels converging to $\bar{v}$ as $k \to \infty$. It will intuitively be clear that any limit point of the sequence $\{c(t_k)\}$ is then a minimum solution of problem (1.1.1).

There are several algorithms which operate along these lines: the method of centres (Huard (1964, 1967)), some variants of it with relaxation facilities (Tremolières (1968)), and SUMT without parameters (Fiacco and McCormick (1967b)). Generally, a sequence $\{t_k\}$ as mentioned above is obtained as follows. The first step starts with a truncation level $t_1 = f(x_0)$, where x_0 is some feasible solution of (1.1.1). Hence $t_1 \geqslant \bar{v}$. At the beginning of the kth step the truncation level t_{k-1} of the previous step and the iteration point $c(t_{k-1})$ are available whereas, by construction,

$$t_{k-1} > f\,[c(t_{k-1})]. \tag{4.1.3}$$

The truncation level t_k is then taken to be

$$t_k = t_{k-1} - \varrho\,\{t_{k-1} - f\,[c(t_{k-1})]\}. \tag{4.1.4}$$

Here, ϱ stands for a *relaxation factor* such that $0 < \varrho \leqslant 1$, in order to ensure that $t_{k-1} > t_k \geqslant \bar{v}$. The reason for introducing this factor will be explained at the end of sec. 4.3. The proof that the sequence $\{t_k\}$ converges to $\bar{v}$ is postponed until the next section.

An interesting example of these techniques is obtained by substituting $\varphi(\eta) = \ln \eta$ into (4.1.2). A point maximizing

$$p \ln\,[t - f(x)] + \sum_{i=1}^{m} \ln g_i(x)$$

over $T^{\circ}(t)$ can also be found by the maximizing over $T(t)$ of the function

$$d_t(x) = [t - f(x)]^p \prod_{i=1}^{m} g_i(x).$$

This function is an example of the general *distance function* appearing in the method of centres: one of the properties of a distance function is that it vanishes on the boundary of a truncated constraint set; another is that it is positive in every interior point of the truncated constraint set under consideration. A point maximizing d_t over $T(t)$ was referred to as a *centre* of $T(t)$. In what follows we shall be using the name "centre" for the points $c(t)$ in the more general case where the function $B_t{}^*$ of (4.1.2) is employed.

4.2. Barrier-function techniques with moving truncations

In this section we shall variously use the symbols c_k, f_k and T_k to denote $c(t_k)$, $f\,[c(t_k)]$ and $T(t_k)$ respectively. We can now establish:

Theorem 4.2.1. If (a) problem (1.1.1) satisfies condition 4.1, and (b) the function φ appearing in the moving-truncations barrier function (4.1.2) satisfies condition 3.2, then the truncation levels t_k generated in accordance with (4.1.4), and the truncated constraint sets T_k, $k = 1, 2, \ldots$, have the following properties.
(1) If $T_k{}^\circ$ is empty for some k, then c_{k-1} is an unconstrained minimum of f.
(2) If $T_k{}^\circ$ is nonempty for some k, there is a point c_k maximizing $B_{t_k}{}^*$ over $T_k{}^\circ$.
(3) If $T_k{}^\circ$ is nonempty for every $k = 1, 2, \ldots$, then

$$\lim_{k\to\infty} t_k = \overline{v}.$$

Property (3) implies that every limit point of the sequence $\{c_k\}$ is a minimum solution of (1.1.1).
Proof (1). If $T_k{}^\circ$ happens to be empty, then $f(x) \geqslant t_k$ for any $x \in R^\circ$, whereas $t_k \geqslant \overline{v}$. Hence, $t_k = \overline{v}$, and from (4.1.4) we can now infer that

$$t_{k-1} - \overline{v} = \varrho\,[t_{k-1} - f_{k-1}] \leqslant t_{k-1} - f_{k-1},$$

which implies $f_{k-1} \leqslant \overline{v}$. On the other hand, c_{k-1} is feasible. It must accordingly be true that $f_{k-1} = \overline{v}$, and this proves the first part.

(2) The set T_k is obviously compact. If its interior is nonempty, then lemma 3.2.1 can be invoked (with $R_1 = S = T_k$ and $h = -B_{t_k}{}^*$) in order to establish the existence of a point c_k maximizing $B_{t_k}{}^*$ over $T_k{}^\circ$. This proves the second part of the theorem.

(3) Lastly, we consider the infinite sequences $\{t_k\}$ and $\{f_k\}$. By (4.1.3) and (4.1.4) we have

$$t_{k-1} > t_k > f_k \geqslant \overline{v}, \quad k = 1, 2, \ldots . \tag{4.2.1}$$

Thus, the sequence $\{t_k\}$ is monotonic, decreasing and bounded below. We can accordingly write

$$\lim_{k\to\infty} t_k = \bar{t} \geqslant \overline{v}.$$

Assume $\bar{t} > \overline{v}$. There is a point $y_0 \in R^\circ$ such that

$$\overline{v} < f(y_0) < \bar{t},$$

and one can verify that $y_0 \in T_k{}^\circ$ for any $k = 1, 2, \ldots$. Furthermore, substituting y_0 into (4.1.2) one obtains

$$\lim_{k\to\infty} B_{t_k}{}^*(y_0) = \lim_{k\to\infty} \{p\,\varphi[t_k - f(y_0)] + \sum_{i=1}^{m} \varphi[g_i(y_0)]\} > -\infty.$$

It follows from the convergence of the sequence $\{t_k\}$ that

$$\lim_{k\to\infty} (t_{k-1} - t_k) = 0.$$

From (4.1.4) we can infer

$$t_{k-1} - f_{k-1} = \varrho^{-1}\,(t_{k-1} - t_k),$$

so that

$$\lim_{k\to\infty} (t_k - f_k) = 0, \tag{4.2.2}$$

and hence

$$\lim_{k\to\infty} \varphi(t_k - f_k) = -\infty.$$

Then it follows from inspection of (4.1.2) that

$$\lim_{k\to\infty} B_{t_k}{}^*(c_k) = \lim_{k\to\infty} \{p\,\varphi(t_k - f_k) + \sum_{i=1}^{m} \varphi[g_i(c_k)]\} = -\infty.$$

Thus, for k sufficiently large,

$$B_{t_k}{}^*(c_k) < B_{t_k}{}^*(y_0).$$

This contradicts the statement that c_k maximizes $B_{t_k}{}^*$ over $T_k{}^o$ for *any* $k = 1, 2, \ldots$. Thus, $\bar{t} = \bar{v}$, and one can infer from (4.2.1) that

$$\lim_{k\to\infty} f_k = \bar{v}.$$

This completes the proof of theorem 4.2.1.

The precise relationship between parametric barrier-function techniques and barrier-function techniques with moving truncations is expressed by:

Theorem 4.2.2. If (a) problem (1.1.1) satisfies condition 4.1, (b) the problem functions have continuous first-order partial derivatives in E_n, and (c) the function φ appearing in (4.1.2) satisfies condition 3.2 and 3.7, then a centre c_k minimizes the parametric barrier function

$$B_r(x) = f(x) - r^{\lambda} \sum_{i=1}^{m} \varphi[g_i(x)] \tag{4.2.3}$$

over R^o for r equal to

$$r_k = [p\,\varphi'(t_k - f_k)]^{-1/\lambda}. \tag{4.2.4}$$

The sequence $\{r_k\}$ generated in this manner is a monotonic, nonincreasing null sequence.

Proof. The gradient of $B_{t_k}{}^*$ vanishes at a centre c_k whence

$$\nabla f(c_k) - \sum_{i=1}^{m} \left(\frac{\varphi' \{g_i(c_k)\}}{p\, \varphi'(t_k - f_k)} \right) \nabla g_i(c_k) = 0.$$

Obviously, the point c_k solves the equation

$$\nabla f(x) - r^{\lambda} \sum_{i=1}^{m} \varphi' \{g_i(x)\} \nabla g_i(x) = 0$$

for r equal to the value r_k of (4.2.4). We may note that r_k is positive since $t_k > f_k$ and $\varphi'(\eta) > 0$ for any $\eta > 0$ by lemma 3.4.1. Hence, B_{r_k} is convex in R^o, and c_k minimizes this function over R^o since its gradient vanishes at c_k.

The monotonic behaviour of $\{r_k\}$ is shown as follows. Writing

$$b_k = - \sum_{i=1}^{m} \varphi \{g_i(c_k)\}$$

and keeping in mind that c_k minimizes B_{r_k} over R^o we have

$$f_k + r_k{}^{\lambda}\, b_k \leqslant f_{k-1} + r_k{}^{\lambda}\, b_{k-1},$$

$$f_{k-1} + r_{k-1}{}^{\lambda}\, b_{k-1} \leqslant f_k + r_{k-1}{}^{\lambda}\, b_k,$$

whence

$$(r_{k-1}{}^{\lambda} - r_k{}^{\lambda})\,(f_k - f_{k-1}) \leqslant 0.$$

Thus $r_{k-1} \geqslant r_k$ since $f_k < f_{k-1}$. Finally, (4.2.2), (4.2.4), and the behaviour of φ' as its argument decreases to 0 lead to

$$\lim_{k \to \infty} r_k = 0,$$

which completes the proof of theorem 4.2.2.

The relationship between the two classes of methods can also be clarified if we continue the introductory sketch presented in sec. 4.1.

We have to construct a sequence of truncation levels converging to the minimum value $\bar{v}$ of (1.1.1). Thus, we are facing the problem of minimizing some variable t subject to the constraints

$$\left. \begin{array}{c} t - f(x) \geqslant 0, \\ g_i(x) \geqslant 0, \quad i = 1, \ldots, m. \end{array} \right\}$$

In order to solve this "extended" problem one may introduce the parametric barrier function

$$B_r(t,x) = t - r^\lambda \{p\, \varphi[t - f(x)] + \sum_{i=1}^{m} \varphi[g_i(x)]\}, \tag{4.2.5}$$

which differs from the ordinary barrier functions in the sense that one of the constraints is weighted by a positive factor p. It will immediately be clear that

$$B_r(t,x) = t - r^\lambda\, B_t^*(x). \tag{4.2.6}$$

Let $[t(r), x(r)]$ denote a point minimizing (4.2.5) over the interior of the constraint set of the extended problem for $r > 0$. Then

$$B_r[t(r), x(r)] \leqslant B_r[t(r), x]$$

for any $x \in R^\circ$. Employing this result in (4.2.6) one will observe that $x(r)$ is a centre of the truncated constraint set $T[t(r)]$.

The results of theorem 4.2.2 may be summarized as follows: a moving-truncations barrier-function technique is equivalent to a parametric barrier-function technique *adjusting the controlling parameter automatically.* At first sight this is a particularly welcome feature, not only from a theoretical standpoint. For, under certain conditions the parametric technique based on the barrier function B_r of (4.2.3) admits of a minimizing trajectory $[x(r), u(r)]$ which can be expanded in a Taylor series about $r = 0$. The centres c_k generated by the moving-truncations method treated here can be written as

$$c_k = x(r_k), \quad k = 1, 2, \ldots, \tag{4.2.7}$$

with r_k given by (4.2.4). Thus, we obtain a sequence $[x(r_k), u(r_k)]$ on the minimizing trajectory. This sequence is clearly amenable to *extrapolation* towards a minimum solution $\bar{x}$ of the problem. The crucial point, however, is the *rate of convergence*, and we shall accordingly be dealing with that subject in the next section.

4.3. Rate of convergence

On the ground of the Taylor series expansion of $f[x(r)]$ in terms of r we can write

$$\frac{f(c_k) - f(\bar{x})}{f(c_{k-1}) - f(\bar{x})} = \frac{f[x(r_k)] - f(\bar{x})}{f[x(r_{k-1})] - f(\bar{x})} \approx \frac{r_k}{r_{k-1}}$$

for small values r_{k-1} and r_k. Particularly the quantity

$$\lim_{k \to \infty} \frac{r_k}{r_{k-1}}$$

is an appropriate measure of the ultimate rate of convergence of a method with moving truncations. A value r_k generated by such a method, in accordance with (4.2.4), will henceforth be termed an *equivalent r-value.*

Another appropriate measure of the rate of convergence could of course be the quantity

$$\lim_{k\to\infty} \frac{t_k - \bar{v}}{t_{k-1} - \bar{v}},$$

which refers immediately to the truncations levels. The next theorem, however, shows that these two measures of efficiency are equivalent.

Theorem 4.3.1. If (a) problem (1.1.1) satisfies condition 4.1, (b) the problem functions have continuous second-order partial derivatives in E_n, (c) a Kuhn–Tucker point $(\bar{x},\bar{u})$ of (1.1.1) exists satisfying the Jacobian uniqueness conditions 2.1 to 2.3, (d) the point $\bar{x}$ is a boundary point of the constraint set R, (e) the sequence $\{r_k\}$ denotes a sequence of equivalent r-values generated by the moving-truncations method based on (4.1.2), and (f) the function φ appearing in (4.1.2) satisfies the conditions 3.2 and 3.7, then

$$\lim_{k\to\infty} \frac{t_k - \bar{v}}{t_{k-1} - \bar{v}} = \lim_{k\to\infty} \frac{r_k}{r_{k-1}} = 1 - \frac{\varrho}{\bar{\beta} + 1} \tag{4.3.1}$$

with

$$\bar{\beta} = p^{-1/\lambda} \sum_{i=1}^{\alpha} (\bar{u}_i)^{1-1/\lambda}. \tag{4.3.2}$$

Proof. Without loss of generality we confine ourselves to the case where $\varphi'(\eta) = \eta^{-\lambda}$, so that, by (4.2.4),

$$r_k = p^{-1/\lambda}(t_k - f_k). \tag{4.3.3}$$

The parametric technique based on (4.2.3) has, by theorem 3.4.1, a minimizing function $[x(r), u(r)]$ with a continuous first-order derivative $[x'(r), u'(r)]$ in a neighbourhood of $r = 0$. The vector $u(r)$ is, of course, the m vector with components

$$u_i(r) = \frac{r^\lambda}{g_i{}^\lambda[x(r)]}, \quad i = 1, \ldots, m.$$

Hence,

$$g_i[x(r)] = r[u_i(r)]^{-1/\lambda}, \quad i = 1, \ldots, m,$$

and, using (4.3.3) and the relation $c_k = x(r_k)$, we obtain

$$g_i(c_k) = (t_k - f_k)\,[p\, u_i(r_k)]^{-1/\lambda}. \tag{4.3.4}$$

A second relation to be used here is obtained by application of the Kuhn–

Tucker relations (2.1.10). Then

$$\lim_{r\downarrow 0} \frac{\sum_{i=1}^{\alpha} \bar{u}_i\, g_i[x(r)]}{f[x(r)]-f(\bar{x})} = \lim_{r\downarrow 0} \frac{r^{-1}\sum_{i=1}^{\alpha} \bar{u}_i\, g_i[x(r)]}{r^{-1}\,\{f[x(r)]-f(\bar{x})\}} =$$

$$= \frac{\sum_{i=1}^{\alpha} \bar{u}_i\, \nabla g_i(\bar{x})^T\, x'(0)}{\nabla f(\bar{x})^T\, x'(0)} = 1. \tag{4.3.5}$$

It should be noticed that the assumption of $\bar{x}$ being a boundary point of R is essential. Then $\alpha \geqslant 1$, and for any $i = 1, \ldots, \alpha$ one has

$$\nabla g_i(\bar{x})^T\, x'(0) = \lim_{r\downarrow 0} \frac{g_i[x(r)]}{r} = (\bar{u}_i)^{-1/\lambda} > 0.$$

The result of (4.3.5) is accordingly obtained by dividing two *nonzero* quantities. If $\bar{x}$ is an interior minimum (an exceptional case in practical circumstances), it may happen that $c_k = \bar{x}$ for some k; then the process terminates after a finite number of steps.

One can infer from (4.3.5) that

$$\sum_{i=1}^{\alpha} \bar{u}_i\, g_i[x(r)] = \{f[x(r)]-f(\bar{x})\}\,(1+\varepsilon_r),$$

with

$$\lim_{r\downarrow 0} \varepsilon_r = 0.$$

Keeping in mind that $c_k = x(r_k)$, we can immediately write

$$\sum_{i=1}^{\alpha} \bar{u}_i\, g_i(c_k) = (f_k - \bar{v})\,(1+\varepsilon_k), \tag{4.3.6}$$

with

$$\lim_{k\to\infty} \varepsilon_k = 0.$$

Lastly, we define

$$\beta_k = \sum_{i=1}^{\alpha} \bar{u}_i\, [p\, u_i(r_k)]^{-1/\lambda}.$$

Combination of (4.3.4) and (4.3.6) yields

$$(t_k - f_k)\,\beta_k = (f_k - \overline{v})\,(1 + \varepsilon_k),$$

and we can now readily establish

$$(t_k - \overline{v})\,\beta_k = (f_k - \overline{v})\,(\beta_k + 1 + \varepsilon_k). \tag{4.3.7}$$

Using (4.1.3) we can write

$$(t_k - \overline{v}) = (1 - \varrho)\,(t_{k-1} - \overline{v}) + \varrho\,(f_{k-1} - \overline{v}). \tag{4.3.8}$$

We are now in a position to derive (4.3.1) and (4.3.2). Let us first employ (4.3.7) and (4.3.8) in order to eliminate the factors $(f_{k-1} - \overline{v})$ and $(f_k - \overline{v})$. Then

$$\begin{aligned}\varrho(t_{k-1} - \overline{v})\,\beta_{k-1} &= \varrho\,(f_{k-1} - \overline{v})\,(\beta_{k-1} + 1 + \varepsilon_{k-1}) = \\ &= [(t_k - \overline{v}) - (1 - \varrho)\,(t_{k-1} - \overline{v})]\,(\beta_{k-1} + 1 + \varepsilon_{k-1}).\end{aligned}$$

This yields

$$(t_k - \overline{v})\,(\beta_{k-1} + 1 + \varepsilon_{k-1}) = (t_{k-1} - \overline{v})\,[\beta_{k-1} + (1 - \varrho)\,(1 + \varepsilon_{k-1})],$$

whence

$$\lim_{k\to\infty} \frac{t_k - \overline{v}}{t_{k-1} - \overline{v}} = \frac{\overline{\beta} + 1 - \varrho}{\overline{\beta} + 1} = 1 - \frac{\varrho}{\overline{\beta} + 1}.$$

This is a measure of the rate of convergence of the *truncation levels*. We can now obtain

$$\begin{aligned}\frac{r_k}{r_{k-1}} = \frac{t_k - f_k}{t_{k-1} - f_{k-1}} &= \frac{(f_k - \overline{v})\,(1 + \varepsilon_k)\,\beta_{k-1}}{(f_{k-1} - \overline{v})\,(1 + \varepsilon_{k-1})\,\beta_k} = \\ &= \frac{(t_k - \overline{v})\,(1 + \varepsilon_k)\,(\beta_{k-1} + 1 + \varepsilon_{k-1})}{(t_{k-1} - \overline{v})(1 + \varepsilon_{k-1})\,(\beta_k + 1 + \varepsilon_k)}.\end{aligned}$$

Taking the limit as $k \to \infty$ completes the proof of theorem 4.3.1.

We can now explain the purposes of introducing a relaxation factor ϱ and a weight factor p in the barrier function of (4.1.2). In choosing $\varrho = 1$ we obtain

$$t_k = f\,(c_{k-1}), \quad k = 1, 2, \ldots .$$

Hence, maximization of $B_{t_k}{}^*$ cannot start from c_{k-1} since it is a point on the boundary of T_k where $B_{t_k}{}^*$ is undefined. It is therefore easier to use a relaxation $\varrho < 1$. Then $t_k > f\,(c_{k-1})$, and the search for c_k can immediately depart from the previous centre c_{k-1}. The computational process is slowed down to an extent displayed by (4.3.1) and (4.3.2). An increase of p speeds up the convergence, and this can be made plausible by inspection of the barrier function $B_t{}^*$. Any point maximizing the second term

$$\sum_{i=1}^{m} \varphi[g_i(x)]$$

is a centre of R^o. It is the first term which forces convergence to a minimum solution of (1.1.1). Accordingly as the weight of the first term is increased, the rate of convergence is improved. The influence of p is precisely expressed by (4.3.1) and (4.3.2).

There is a remarkable difference between the method of centres (a first-order technique) and the higher-order moving-truncations barrier-function techniques. In the first-named case we obtain, by substituting $\lambda = 1$ into (4.3.2),

$$\bar{\beta} = p^{-1}\,\alpha.$$

Then

$$\lim_{k\to\infty} \frac{t_k - \bar{v}}{t_{k-1} - \bar{v}} = \lim_{k\to\infty} \frac{r_k}{r_{k-1}} = 1 - \frac{\varrho\, p}{\alpha + p}.$$

This implies that one can predict the rate of convergence, or at least its order of magnitude, since α and the number n of variables are of the same order of magnitude: the number of active constraints with *linearly independent* gradients at $\bar{x}$ cannot exceed n. For the higher-order techniques, however, the rate of convergence is unpredictable since the Lagrangian multipliers appearing in (4.3.2) are unknown at the beginning of the computations. It is worth noting that a similar phenomenon was found in studying the parametric barrier-function techniques (sec. 3.4): the first-order approximation of $f\,[x(r)] - \bar{v}$ depends in general on the Lagrangian multipliers; if $\lambda = 1$, however, this approximation depends only on α.

It is obvious from the above arguments that a reasonable choice for the weight factor p can only be made if the method of centres is employed. By taking p equal to n, for example, and $\varrho = 1$ one ensures that the rate of convergence is less than or equal to $\frac{1}{2}$.

The formulas (4.3.1) and (4.3.2) suggest the possibility of speeding up the convergence by choosing $\varrho > 1$ so that there is some *overrelaxation.* One has to be sure, of course, that none of the truncation levels so obtained is less than $\bar{v}$. This can be achieved by adjusting the relaxation factor in every step of the process. In the kth step, for instance, one may explore the direction $-\nabla f\,(c_{k-1})$ emanating from the centre c_{k-1} in order to find some *feasible* point ξ_{k-1} such that

$$f\,(\xi_{k-1}) < f\,(c_{k-1}).$$

An overrelaxed truncation level is then obtained by setting

$$t_k = f\,(\xi_{k-1}).$$

The relaxation factor ϱ_k for the kth step, which can be calculated from

$$f(\xi_{k-1}) = t_{k-1} - \varrho_k\,(t_{k-1} - f_{k-1}),$$

is then greater than unity.

We shall not go into further details. We only want to demonstrate that automatic adjustment of the controlling parameter, as it is performed by barrier-function techniques with moving truncations, provides a doubtful advantage over the corresponding parametric techniques, particularly if the order is greater than 1. Then the rate of convergence depends critically on the Lagrangian multipliers $\bar{u}_1, \ldots, \bar{u}_m$ which are unknown at the start of the computational process for solving (1.1.1). Hence, it is difficult to find an appropriate value for the weight factor p in order to improve the convergence.

To do justice to the moving-truncations techniques one has to consider the principal Hessian matrix of (4.1.2), and its behaviour in the limiting case as the truncation levels t_k decrease to $\bar{v}$, and the equivalent r-values r_k decrease to 0. It can be shown that this matrix has a condition number which varies with r_k^{-1} for k large enough, just as in the parametric case. It is therefore unlikely that maximization of the barrier function $B_{t_k}{}^*$ would be easier than minimization of the parametric barrier function B_{r_k}, for large values of k.

4.4. Loss-function techniques with moving truncations

The development of loss-function techniques with moving truncations proceeds in analogy with the mode of operation in the previous sections. There are some minor differences: we have to deal with a monotonic, *increasing* sequence of truncation levels converging to $\bar{v}$ *from below*, in contrast to the convergence from above provided by the barrier-function techniques with moving truncations.

Condition 4.1 can be weakened. The requirement that the interior of the constraint set R be nonempty is essential for barrier-function techniques, but it can be dropped if the technique for solving (1.1.1) is concerned with a loss function. One may compare, for example, the formulation of theorems 3.2.4 and 3.2.5. In this section we shall accordingly be working under:

Condition 4.2. Problem (1.1.1) is a convex-programming problem with a nonempty, compact constraint set.

Let us now, first, describe the basic concepts and the iterative procedure. We begin by introducing the *moving-truncations loss function*

$$L_t{}^*(x) = p\,\psi[t - f(x)] + \sum_{i=1}^{m} \psi[g_i(x)], \qquad (4.4.1)$$

where ψ is a function of one variable satisfying the conditions 3.3, 3.4 and possibly 3.8. We have again attached a positive weight factor p to the first term in the right-hand side of (4.4.1). From these arguments L_t^* is a concave function in E_n for any value of the truncation level t. The truncation $F(t)$ and the truncated constraint set $T(t)$ to be considered in this section are again defined by

$$\begin{aligned} F(t) &= \{x \mid f(x) \leqslant t; \quad x \in E_n\}, \\ T(t) &= R \cap F(t). \end{aligned} \tag{4.4.2}$$

Using the properties that

$$\psi[t - f(x)] \begin{cases} = 0 & \text{for all} \quad x \in F(t), \\ < 0 & \text{for all} \quad x \notin F(t), \end{cases} \tag{4.4.3}$$

$$\sum_{i=1}^{m} \psi[g_i(x)] \begin{cases} = 0 & \text{for all} \quad x \in R, \\ < 0 & \text{for all} \quad x \notin R, \end{cases} \tag{4.4.4}$$

we obtain straightaway

$$L_t^*(x) \begin{cases} = 0 & \text{for all} \quad x \in T(t), \\ < 0 & \text{for all} \quad x \notin T(t). \end{cases} \tag{4.4.5}$$

The set $T(t)$ is nonempty if, and only if, the truncation level $t \geqslant \bar{v}$. If $t < \bar{v}$, it must be true that

$$L_t^*(x) < 0 \quad \text{for all} \quad x \in E_n.$$

We shall presently demonstrate that a point $c(t)$ maximizing L_t^* over E_n exists under certain conditions for any t. One may now distinguish a number of cases.

If $t \geqslant \bar{v}$, then $T(t)$ is nonempty. By (4.4.5), any point in $T(t)$ is a maximizing point.

If $t = \bar{v}$, then $T(t)$ is precisely the set of minimum solutions of (1.1.1); this is a welcome feature if $\bar{v}$ is known at the beginning of the computations for solving (1.1.1); generally, however, $\bar{v}$ is unknown. The two cases share the property

$$L_t^*[c(t)] = 0.$$

If $t < \bar{v}$ it must be true that

$$L_t^*[c(t)] < 0,$$

and, moreover, we can show that

$$t < f\,[c(t)] \leqslant \bar{v}.$$

Thus, $c(t) \notin F(t)$. Furthermore, the next theorem shows that $c(t) \notin R$ in the (usual) case that f does not have an unconstrained minimum in R. However, we may think of $c(t)$ as the (*common*) *centre* of the two disjunct sets R and $F(t)$.

The basic idea of the techniques to be treated here will now be evident: if $\{t_k\}$ is a monotonic, *increasing* sequence of truncation levels converging to $\bar{v}$, then any limit point of the sequence $\{c(t_k)\}$ is a minimum solution of (1.1.1).

In what follows we shall variously employ the symbols c_k, f_k and T_k to denote $c(t_k)$, $f[c(t_k)]$ and $T(t_k)$ respectively. We consider the following construction of a sequence $\{t_k\}$ as mentioned above. In the first step the truncation level t_1 is taken to be a lower estimate of $\bar{v}$. At the beginning of the kth step the truncation level t_k is generated according to

$$t_k = t_{k-1} + \varrho\,\{f[c(t_{k-1})] - t_{k-1}\}, \tag{4.4.6}$$

where ϱ is a relaxation factor such that $0 < \varrho \leqslant 1$ in order to guarantee that $t_{k-1} < t_k \leqslant \bar{v}$. This procedure is validated by:

Theorem 4.4.1. If (a) problem (1.1.1) satisfies condition 4.2, (b) the problem functions have continuous first-order partial derivatives in E_n, and (c) the function ψ appearing in the moving-truncations loss function (4.4.1) satisfies the conditions 3.3, 3.4 and 3.8, then the following properties must hold.

(1) A point $c(t)$ maximizing $L_t{}^*$ over E_n exists for any t.

(2) If $t < \bar{v}$, then $t < f[c(t)] \leqslant \bar{v}$.

(3) If f has an unconstrained minimum in R, and if $t < \bar{v}$, then $c(t)$ is a minimum solution of problem (1.1.1).

(4) If there is no unconstrained minimum of f in R, so that any minimum solution of (1.1.1) must be a boundary point of R, and if $t_1 < \bar{v}$, then the sequence $\{t_k\}$ generated by (4.4.6) is a monotonic, increasing sequence converging to $\bar{v}$. Any limit point of the sequence $\{c_k\}$ is a minimum solution of (1.1.1).

(5) Each point c_k minimizes the parametric loss function

$$L_s(x) = f(x) - s^{-\mu} \sum_{i=1}^{m} \psi[g_i(x)] \tag{4.4.7}$$

for s equal to the *equivalent s-value*

$$s_k = [p\,\psi'(t_k - f_k)]^{1/\mu}. \tag{4.4.8}$$

The sequence $\{s_k\}$ is a monotonic, nonincreasing null sequence.

Proof. (1) We can invoke theorem 3.2.5 in order to demonstrate the existence of a maximizing point $c(t)$. The function $-L_t{}^*$ is precisely the parametric loss function for solving the problem

$$\left.\begin{array}{c} \text{minimize } -\psi[t - f(x)] \text{ subject to} \\ g_i(x) \geqslant 0; \quad i = 1, \ldots, m, \end{array}\right\}$$

with the controlling parameter s equal to p. Then theorem 3.2.5 ensures the existence of a point minimizing $-L_t{}^*$ over E_n for any positive p.

In proving the remainder of theorem 4.4.1 we shall frequently consider a value $t_0 < \bar{v}$ and the centre $c_0 = c(t_0)$. The gradient of $L_{t_0}{}^*$ vanishes at c_0 whence

$$p\, \psi'[t_0 - f(c_0)]\, \nabla f(c_0) - \sum_{i=1}^{m} \psi'[g_i(c_0)]\, \nabla g_i(c_0) = 0. \tag{4.4.9}$$

Moreover, if we take $\bar{x}$ to denote a minimum solution of (1.1.1),

$$p\, \psi[t_0 - f(c_0)] + \sum_{i=1}^{m} \psi[g_i(c_0)] \geqslant p\, \psi[t_0 - f(\bar{x})] + \sum_{i=1}^{m} \psi[g_i(\bar{x})] = p\, \psi(t_0 - \bar{v}),$$

whence

$$p^{-1} \sum_{i=1}^{m} \psi[g_i(c_0)] \geqslant \psi(t_0 - \bar{v}) - \psi[t_0 - f(c_0)]. \tag{4.4.10}$$

Let us now move on to the proof of the remaining parts.

(2) Assume that a value $t_0 < \bar{v}$ exists such that $f(c_0) \leqslant t_0$. Then (4.4.9) reduces to

$$\sum_{i=1}^{m} \psi'[g_i(c_0)]\, \nabla g_i(c_0) = 0. \tag{4.4.11}$$

Thus, the point c_0 maximizes

$$\sum_{i=1}^{m} \psi[g_i(x)]$$

over E_n, which can only be true if $c_0 \in R$. This leads to the contradictory result $f(c_0) \geqslant \bar{v}$. Hence, we can only have $f(c_0) > t_0$.

Secondly, assume $f(c_0) > \bar{v}$. Then (4.4.10) yields

$$\sum_{i=1}^{m} \psi[g_i(c_0)] > 0,$$

since $0 > t_0 - \bar{v} > t_0 - f(c_0)$, and $\psi(\eta)$ *strictly* increasing for $\eta < 0$. This contradicts (4.4.4) so that $f(c_0) \leqslant \bar{v}$.

(3) Let us again consider a value $t_0 < \bar{v}$. The assumption that f has an unconstrained minimum in R implies $f(x) \geqslant \bar{v}$ for any $x \in E_n$, whence $f(c_0) = \bar{v}$. Using (4.4.10) and (4.4.4) we can easily see that c_0 must be feasible. This proves that c_0 is a minimum solution of (1.1.1).

(4) Let $t_0 < \bar{v}$ and suppose that $f(c_0) = \bar{v}$. We can then infer from (4.4.10) that c_0 must be feasible. By (4.4.4), formula (4.4.9) reduces to

$$p\, \psi'(t_0 - \bar{v})\, \nabla f(c_0) = 0,$$

so that $\nabla f(c_0) = 0$. Thus, if there is no unconstrained minimum of f in R, it must be true that $f(c_0) < \bar{v}$. Moreover, starting with a truncation level $t_1 < \bar{v}$ and generating the sequence $\{t_k\}$ according to (4.4.6) we obtain straightaway

$$t_{k-1} < t_k < f_k < \bar{v}. \tag{4.4.12}$$

The sequence $\{t_k\}$ has apparently a finite limit $\bar{t} \leqslant \bar{v}$. Assume $\bar{t} < \bar{v}$. Then

$$\bar{t} < f\,[c(\bar{t})] < \bar{v}. \tag{4.4.13}$$

From (4.4.6) we can infer

$$\lim_{k\to\infty} f_k = \bar{t},$$

so that, by continuity, $f\,[c(\bar{t})] = \bar{t}$, contradicting (4.4.13). Hence, we must have $\bar{t} = \bar{v}$ and

$$\lim_{k\to\infty} f_k = \bar{v}. \tag{4.4.14}$$

Finally, by (4.4.10),

$$p^{-1} \sum_{i=1}^{m} \psi[g_i(c_k)] \geqslant \psi(t_k - \bar{v}) - \psi(t_k - f_k),$$

leading to

$$\lim_{k\to\infty} \sum_{i=1}^{m} \psi[g_i(c_k)] = 0. \tag{4.4.15}$$

By theorem 2.5.4, the set

$$R_\varepsilon = \{x \,|g_i(x) \geqslant -\varepsilon;\quad i = 1, \ldots, m\}$$

is a compact (possibly empty) subset of E_n for any ε. Let ε be positive. Then, by (4.4.15), there is a number K such that $c_k \in R_\varepsilon$ for all $k \geqslant K$. Consequently, the sequence $\{c_k\}$ has a limit point $\bar{c}$. Combination of (4.4.14) and (4.4.15) leads to the result that $\bar{c}$ is a minimum solution of (1.1.1).

(5) The proof that c_k minimizes the parametric loss function L_s of (4.4.7) for s equal to the value s_k of (4.4.8) rests on the observation that the gradient of $L_{t_k}{}^*$ vanishes at c_k. Reasoning along the same lines as in theorem 4.2.2 one can readily establish the monotonic behaviour of the sequence $\{s_k\}$.

The last part of this theorem shows the relationship between loss-function techniques with moving truncations and the parametric techniques of the previous chapter. Here, we have a *loss-function technique* adjusting the controlling parameter automatically. The question of whether it is a workable method depends critically on the rate of convergence, which is treated in the next theorem.

Theorem 4.4.2. If (a) problem (1.1.1) satisfies condition 4.2, (b) the problem functions have continuous second-order partial derivatives in E_n, (c) a Kuhn–Tucker point $(\bar{x},\bar{u})$ of (1.1.1) exists satisfying the Jacobian uniqueness conditions 2.1 to 2.3, (d) the point $\bar{x}$ is a boundary point of the constraint set R, (e) the sequence $\{s_k\}$ denotes a sequence of equivalent s-values generated by the moving-truncations loss-function technique based on (4.4.1), and (f) the function ψ appearing in (4.4.1) satisfies the conditions 3.3, 3.4 and 3.8, then

$$\lim_{k\to\infty} \frac{t_k - \bar{v}}{t_{k-1} - \bar{v}} = \lim_{k\to\infty} \frac{s_k}{s_{k-1}} = 1 - \frac{\varrho}{\bar{\gamma} + 1}, \tag{4.4.16}$$

with

$$\bar{\gamma} = p^{1/\mu} \sum_{i=1}^{\alpha} (\bar{u}_i)^{1+1/\mu}. \tag{4.4.17}$$

Proof. Without loss of generality we confine ourselves to a function ψ such that

$$\omega'(\eta) = (-\eta)^{\mu}.$$

By (4.4.8) and (4.4.12) we can immediately write

$$s_k = p^{1/\mu}\,(f_k - t_k). \tag{4.4.18}$$

Taking $[x(s), u(s)]$ to denote the minimizing trajectory of the loss-function technique based on the function L_s of (4.4.7), we have

$$u_i(s) = s^{-\mu}\,\psi'\{g_i[x(s)]\}, \quad i = 1, \ldots, m. \tag{4.4.19}$$

As we have seen in sec. 3.4, there is a positive number K such that

$$\left.\begin{array}{l} g_i[x(s_k)] < 0, \\ u_i(s_k) \;\; > 0, \end{array}\right\} \; i = 1, \ldots, \alpha,$$

$$\left.\begin{array}{l} g_i[x(s_k)] > 0, \\ u_i(s_k) \;\; = 0, \end{array}\right\} \; i = \alpha + 1, \ldots, n,$$

for all $k > K$. Then (4.4.19) may be used to obtain

$$u_i(s_k) = s_k{}^{-\mu}\,\omega'\{g_i[x(s_k)]\}, \quad i = 1, \ldots, \alpha,$$

for $k > K$, whence

$$s_k = -[u_i(s_k)]^{-1/\mu} g_i[x(s_k)], \quad i = 1, \ldots, \alpha. \tag{4.4.20}$$

Substituting this result into (4.4.18) and writing $c_k = x(s_k)$ we obtain

$$g_i(c_k) = (t_k - f_k)\,[p\,u_i(s_k)]^{1/\mu}, \quad i = 1, \ldots, \alpha. \tag{4.4.21}$$

Moreover,

$$\sum_{i=1}^{\alpha} \bar{u}_i\, g_i(c_k) = (f_k - \bar{v})\,(1 + \varepsilon_k), \tag{4.4.22}$$

with

$$\lim_{k\to\infty} \varepsilon_k = 0.$$

The last relation can be derived by application of the Kuhn–Tucker relations, just as in theorem 4.3.1. Lastly, we define

$$\gamma_k = \sum_{i=1}^{\alpha} \bar{u}_i\, [p\, u_i(s_k)]^{1/\mu}.$$

Combination of (4.4.21) and (4.4.22) leads then to

$$(t_k - f_k)\,\gamma_k = (f_k - \bar{v})\,(1 + \varepsilon_k),$$

which can be rewritten as

$$(t_k - \bar{v})\,\gamma_k = (f_k - \bar{v})\,(\gamma_k + 1 + \varepsilon_k). \tag{4.4.23}$$

This result is similar to that of (4.3.7) so that the proof of theorem 4.4.2 can be completed in the same way.

Let us finally discuss the results of theorem 4.4.2. It will be clear that relaxation does not provide any advantage. It is not even *necessary* to apply some relaxation in order to obtain a starting point for the next step in the iterative procedure: the moving-truncations loss function $L_t{}^*$ is defined for any $x \in E_n$ so that the kth step can immediately start from $c(t_{k-1})$. Overrelaxation is difficult to apply. One does not have a workable criterion for deciding whether $\bar{v}$ is overshot. Such a criterion (feasibility) exists if one employs a barrier-function technique.

The rate of convergence depends clearly on the Lagrangian multipliers $\bar{u}_1$, . . . , $\bar{u}_m$ for any order μ of the loss function. It is evident from (4.4.17) that a decrease of the weight factor p will speed up the convergence. This acceleration can be made plausible from inspection of the loss function $L_t{}^*$. Any point maximizing the second term

$$\sum_{i=1}^{m} \psi[g_i(x)]$$

will be feasible. It is the first term in (4.4.1) which causes infeasibility of the next centre $c(t)$. A smaller weight of this term will apparently lead to an accelerated reduction of the constraint violation.

Some numerical experiences with a moving-truncations loss-function technique are reported by the author in a previous paper (1968c). They show how sensitive the computational process is to changes in the weight factor p. It is evident that an appropriate choice of p cannot be made at the beginning of the computations for solving (1.1.1), since the Lagrangian multipliers are then unknown. This is, at least in our opinion, a serious disadvantage of the loss-function techniques with moving truncations.

5. EVALUATION AND CONCLUSIONS

5.1. Choice of a penalty function

In this chapter we shall finally be dealing with the choice of a penalty function for computational purposes. The preceding chapters provide more material for supporting a particular choice than the arguments brought forward by Fiacco and McCormick (1968), who have almost exclusively been dealing with parametric penalty functions. They motivate their preference for *first-order* penalty functions (logarithmic barrier function, quadratic loss function, and the mixed penalty function which combines their properties) with the argument that first-order penalty functions are easier to differentiate than the higher-order ones. The argument is obviously true, but is it sufficient to base such a decision on these grounds only?

Zangwill (1967a) and Roode (1968) have only been concerned with parametric loss functions. They did not deal with the actual problem of choosing a computationally workable loss function.

It is still an open question whether barrier functions are easier to minimize than loss functions. The question was raised by Murray (1967) and has been left unanswered since that time.

The relationship between parametric penalty-function techniques and the corresponding versions with moving truncations has been studied by Fiacco and McCormick (1967b) and Fiacco (1967). They have indeed pointed to the automatic adjustment of the controlling parameter as a striking feature of methods with moving truncations. What they did *not* consider, however, is the rate of convergence (the efficiency of the adjustment).

The studies of Faure and Huard (1965, 1966), Bui Trong Lieu and Huard (1966), Huard (1964, 1967, 1968) and Tremolières (1968), on the other hand, are entirely devoted to the method of centres and not to the relationship with parametric techniques.

In the light of the results obtained in the previous chapters we may draw the following conclusions for convex-programming problems.

5.2. Choice of the order

There is no obvious reason for not using first-order techniques. Let us consider the parametric and the moving-truncations techniques separately in order to give a motivation.

Parametric penalty-function techniques. The computational process is the same for all methods under consideration: a sequence of minimizing points $x(r_k)$ is generated for monotonic, decreasing, positive values r_k of the controlling parameter r. These points are employed as grid points for extrapolation towards $\bar{x}$. The basis for extrapolation is the Taylor series expansion of the vector function $x(r)$ about $r = 0$. Under certain conditions this is a series expansion in terms of r, *regardless of the order λ of the barrier term and the order μ of the loss term.* The condition number of the Hessian matrix, on the other hand, varies with r^{-1} *independently of λ and μ.* This may be an indication that first-order penalty functions are not easier or harder to minimize than the higher-order ones. The first-order (logarithmic) barrier function, however, offers the additional advantage that the minimum value of problem (1.1.1) is approximated with an a priori given accuracy.

Penalty-function techniques with moving truncations. Within this class of methods the computational processes are also mutually the same: find a series $\{c_k\}$ of centres converging to a minimum solution $\bar{x}$ of (1.1.1) by constructing a sequence of truncation levels decreasing (barrier functions) or increasing (loss functions) monotonically and converging to the minimum value $\bar{v}$ of (1.1.1). The crucial point, however, is the rate of convergence. For the first-order barrier-function technique (method of centres) the rate of convergence is roughly predictable since it depends on the number of active constraints at the minimum solution $\bar{x}$. For the remaining techniques (higher-order barrier-function techniques and all the loss-function techniques) the rate of convergence depends on the unknown Lagrangian multipliers associated with $\bar{x}$, so that it is unpredictable. Thus, higher-order barrier-function techniques do not provide any significant advantage over the first-order ones. A similar, somewhat weaker verdict can be given upon loss-function techniques.

A disadvantage of higher-order techniques is the increasing effort necessary for evaluating the penalty function and its derivatives.

5.3. Controlling parameter or moving truncations?

The relation between methods operating with moving truncations and parametric methods was given by the observation that any centre c_k is a point on the trajectory $\{x(r)|r > 0\}$ originating from a corresponding parametric method. We can write $c_k = x(r_k)$ where r_k denotes the equivalent r-value which can be calculated as soon as c_k is obtained.

Computational success with moving-truncations techniques depends critically on the rate of convergence or, as we have mentioned in the previous section, on the Lagrangian multipliers associated with the minimum solution of (1.1.1). For a parametric technique, however, the rate of convergence (the rate of two successive values of the controlling parameter) can *freely be chosen.*

The rate of convergence of a moving-truncations method can, it is true, be affected by a weight factor attached to the objective function. The present author (1968c) has given a numerical example which demonstrates the effect of weighting. Generally, however, the choice of an appropriate weight factor can readily be made for the method of centres only.

We are accordingly led to *parametric* first-order penalty functions as the most desirable ones for computational purposes. The discussion of the methods involved has so many aspects that we devote a special section to it.

5.4. Parametric first-order penalty functions

This group of methods comprises a pure barrier-function technique based on the *logarithmic barrier function*

$$f(x) - r \sum_{i=1}^{m} \ln g_i(x), \tag{5.4.1}$$

a pure loss-function technique based on the *quadratic loss function*

$$f(x) + r^{-1} \sum_{i=1}^{m} \{\min [0, g_i(x)]\}^2, \tag{5.4.2}$$

and a technique operating with the *mixed penalty function*

$$f(x) - r \sum_{i \epsilon I_1} \ln g_i(x) + r^{-1} \sum_{i \epsilon I_2} \{\min [0, g_i(x)]\}^2. \tag{5.4.3}$$

The pure techniques have particular advantages and disadvantages, and they present an entirely different approach to a minimum solution of the problem. Therefore, we start by summarizing the differences between both of them.

1. *Constraint satisfaction.* A barrier is impenetrable so that the imposed constraints remain satisfied throughout the computational process if (5.4.1) is employed. A technique based on (5.4.2), however, will invariably lead out of the constraint set (unless the objective function has an unconstrained minimum in it).

2. *Evaluation of constraint functions.* The barrier function (5.4.1) requires evaluation of all the constraints appearing in the barrier term. Employing the loss function (5.4.2) one has the following computational advantage: in differentiating it one only has to evaluate the derivatives of the constraints which are *violated* at the current point.

3. *The interior of the constraint set.* A loss-function technique does not require that the interior of the constraint set be nonempty. Hence, it can also be used to handle equality constraints. We shall be dealing with that subject in a following section. It is obvious that barrier-function techniques which operate in the *interior* of the constraint set are not appropriate for handling equalities.

4. *Starting facilities.* Unconstrained minimization of (5.4.2) can start from any point, feasible or not. For a technique using (5.4.1), however, special

starting procedures have to be developed which can be applied if an interior starting point is not available.

5. *Conditioning of the principal Hessian matrix.* We have assumed that the condition number of the principal Hessian matrix of a penalty function, and particularly its rate of change as r decreases to 0, is an appropriate measure of the degree of difficulty in minimizing the penalty function concerned. On the ground of this assumption there is no obvious reason that (5.4.1) should be easier or harder to minimize than (5.4.2).

There are clearly more arguments (2, 3, 4) in favour of the loss-function technique. Nevertheless, we have strong reasons to believe that the advantage of constraint satisfaction (1) offered by the barrier-function technique is most important as soon as one departs from the convexity assumptions. This subject will presently be discussed.

A natural way out of the dilemma seems to be a combination of these methods by using the mixed penalty function (5.4.3). There are many ways of partitioning the index set $I = \{1, \ldots, m\}$ into two disjunct sets I_1 and I_2. However, it is reasonable that the starting point x_0 of the computational process should indicate whether a constraint is to be incorporated in the barrier term or in the loss term. One could think of the constraints as partitioned in such a way that

$$I_1 = \{i | g_i(x_0) > 0; \quad 1 \leqslant i \leqslant m\},$$
$$I_2 = \{i | g_i(x_0) \leqslant 0; \quad 1 \leqslant i \leqslant m\}.$$

The mixed penalty-function technique based on (5.4.3) preserves then the easy starting facilities of a loss-function technique. Furthermore, it guarantees that any constraint which is strictly satisfied in the starting point remains satisfied throughout the computations for solving (1.1.1). The mixed technique can also be used for handling equality constraints. Finally, the principal Hessian matrix of (5.4.3) has a condition number which varies with r^{-1}, just as well as the principal Hessian matrix of a pure barrier or loss function. Thus, we do not expect more difficulties in minimizing the mixed function of (5.4.3) than in minimizing the pure penalty functions (5.4.1) and (5.4.2).

5.5. The convexity assumptions

In this study the objective function f and the constraint functions $g_1, \ldots, g_m$ are supposed to be continuously differentiable, as many times as it was desirable in the given circumstances. This is not a serious restriction: many constrained-minimization problems arising in practice have these properties; as important as anything else is that these properties can easily be verified.

Furthermore, we have required boundedness of the constraint set. This seems to be equally acceptable. The minimum solution of a practical problem can frequently be enclosed by a number of constraints imposed on the ground of a priori knowledge of the problem underlying the mathematical formulation.

The situation is entirely different as soon as we turn to the convexity assumptions. These are generally not easy to verify. On the rather scarce occasions that some verification is possible one may come to the conclusion that the objective function and/or some of the constraint functions violate the convexity conditions. Nevertheless, convexity plays an important role at several places in the preceding analysis. We may briefly summarize the critical points in the proof of the existence and the convergence of penalty-function minima.

1. Characterization of the interior and the boundary of the constraint set R. Concavity of the constraint functions implies that the interior of R is given by the set of points satisfying the constraints with strict inequality sign. Furthermore, R is the closure of its interior. These results are important for the convergence of techniques with pure barrier functions or mixed penalty functions (theorems 3.2.1, 3.2.4 and 4.2.1).

2. Compactness of perturbations of the compact constraint set. This property has extensively been used (in theorems 3.2.3, 3.2.5 and 4.4.1) in order to show the existence of penalty-function minima for any positive value of the controlling parameters.

3. Global convergence. Convexity implies that any local minimum solution of the constrained problem is a global minimum solution. Any limit point of a sequence of penalty-function minima is a global minimum solution.

Abandoning the convexity assumptions one might adopt the mode of operation of Fiacco and McCormick (1968). They considered the nonconvex case and established a number of results which are of course of a local nature: existence of penalty-function minima for sufficiently small, positive values of the controlling parameter(s), and convergence to local minima of problem (1.1.1). Instead of the above-named characterization of the interior of R, one needs an additional hypothesis on the sets of local minima.

These results imply that, in the nonconvex case, a constrained-minimization procedure can only be used for some exploration in the vicinity of the starting point (where the problem functions may have the desired convexity properties). It is a sound strategy to supply then a number of constraints which enclose a minimum solution; the starting point has to be a good guess of a minimum solution. If this starting point satisfies the imposed constraints with strict inequality sign, the particular choice (see sec. 5.3) of the mixed penalty function guarantees that minimization is carried out within the enclosed area. With these precautions a penalty-function technique may be an effective tool for improving the guess.

The convexity assumptions seem to be a suitable hypothesis for ensuring that a problem is well-behaved: then, penalty-function minima exist for any positive value of the controlling parameter and they converge to a *desired* global minimum of the problem under consideration.

5.6. Equality constraints

The results obtained so far can be extended to the case where one comes across inequality as well as equality constraints. Let us turn to the problem

$$\left.\begin{aligned} &\text{minimize } f(x) \text{ subject to} \\ &g_i(x) \geqslant 0; \quad i = 1, \ldots, m, \\ &h_j(x) = 0; \quad j = 1, \ldots, p. \end{aligned}\right\} \tag{5.6.1}$$

This problem can equivalently be written as

$$\left.\begin{aligned} &\text{minimize } f(x) \text{ subject to} \\ &g_i(x) \geqslant 0; \quad i = 1, \ldots, m, \\ &h_j(x) \geqslant 0; \quad j = 1, \ldots, p, \\ -&h_j(x) \geqslant 0; \quad j = 1, \ldots, p. \end{aligned}\right\}$$

It is obvious that a point satisfying the constraints with strict inequality sign does not exist, but we can readily employ a mixed penalty function in order to solve the problem. We incorporate the inequalities $h_j(x) \geqslant 0$ and $-h_j(x) \geqslant 0$, $j = 1, \ldots, p$, in the loss term. A penalty function for solving (5.6.1) is then given by

$$P_r(x) = f(x) - r \sum_{i \in I_1} \ln g_i(x) + r^{-1} \sum_{i \in I_2} \{\min\,[0, g_i(x)]\}^2 + r^{-1} \sum_{j=1}^{p} h_j^2(x). \tag{5.6.2}$$

Now let $x(r)$ be a minimizing point of (5.6.2), and let $u(r)$ and $w(r)$ denote vectors with components

$$u_i(r) = \begin{cases} \dfrac{r}{g_i[x(r)]}, & i \in I_1, \\ -2\,r^{-1} \min\,\{0, g_i[x(r)]\}, & i \in I_2, \end{cases}$$

and

$$w_j(r) = 2\,r^{-1}\,h_j[x(r)], \quad j = 1, \ldots, p,$$

respectively. The behaviour of the minimizing trajectory $[x(r), u(r), w(r)]$ has been thoroughly studied by Fiacco and McCormick (1968), so that we do not need to go into details. If the functions h_j, $j = 1, \ldots, p$, are linear, theorem 3.2.3 can be invoked in order to show that a minimizing point $x(r)$ exists for any $r > 0$. If $\{r_k\}$ is a monotonic, decreasing null sequence, any limit point of the sequence $\{x(r_k)\}$ is a minimum solution of problem (5.6.1). Finally, theorem 3.4.1 can be extended in order to establish the existence of continuous first-order and possibly higher-order derivatives of the vector function $[x(r), u(r), w(r)]$.

The appendix to this thesis contains an ALGOL 60 procedure designed for solving problem (5.6.1) with a technique based on the mixed penalty function of (5.6.2).

5.7. Other developments

The basic idea of solving problem (1.1.1) by sequential unconstrained minimization of a *combination* of the problem functions has been a fruitful one. Particularly the penalty-function approach has led to many techniques and to a large number of computational experiments. It is our purpose to present here a brief survey of methods and refinements which are also based on the above-named idea but fall beyond the scope of this thesis.

First, there is an intriguing loss function, proposed by Zangwill (1967a), which has the form

$$f(x) - r^{-1} \sum_{i=1}^{m} \min\,[0, g_i(x)]. \tag{5.7.1}$$

It has the following, remarkable property (see also Roode (1968)). If $x(r)$ denotes a point minimizing (5.7.1) over E_n for $r > 0$, then a positive ϱ_0 can be found such that $x(r)$ is a minimum solution of the original problem (1.1.1) for any $0 < r < \varrho_0$. Thus the computational process for solving (1.1.1) with a method based on (5.7.1) would be as follows. Generate a sequence $x(r_1)$, $x(r_2)$, . . . of r-minima for a positive decreasing null sequence r_1, r_2, . . . , until a point $x(r_k)$ is found which satisfies the constraints of (1.1.1). Such a number k exists, and $x(r_k)$ must be a required minimum. A serious drawback of (5.7.1) is that its first-order derivatives do not exist at the boundary of the constraint set. Hence, it is doubtful whether the gradient methods for unconstrained minimization can be used to minimize (5.7.1). Recently, Pietrzykowski (1969) sketched a new algorithm for finding the r-minima. He does not report any computational experience, however.

In the preceding chapters we have only been dealing with methods incorporating all the constraints in a penalty function, regardless of whether they are linear or not. It might of course be attractive to treat the linear constraints separately. If only the nonlinear constraints are included in a penalty function, the computational method for solving (1.1.1) reduces to one of sequential *linearly constrained* minimization. There are several algorithms which can be used for minimization under linear constraints. We may for instance refer to the methods of feasible directions proposed by Zoutendijk (1960), the gradient-projection method of Rosen (1961), the reduced-gradient method of Faure and Huard (1965), and a modification of the Davidon–Fletcher–Powell algorithm proposed by Goldfarb (1969). Computational experiences with penalty-function techniques using Goldfarb's method are reported by Davies (1968). These experiences are not encouraging, however. It is true that Goldfarb's method leads to an improvement if there are only linear constraints in problem (1.1.1). When nonlinear constraints are present, one has to enter a *sequential* linearly constrained minimization process. Then Goldfarb's method does not give any substantial improvement over the original technique (unconstrained minimization of a penalty function which also includes the linear constraints).

Several authors have proposed a method for solving (1.1.1) by sequential unconstrained minimization of the *Lagrangian function* associated with the problem. The basic idea may be described as follows. Let x_k denote a point minimizing the function

$$(x) - \sum_{i=1}^{m} u_i{}^{(k)}\, g_i(x)$$

over E_n. If the sequence $\{u^{(k)}\}$ of multipliers is adjusted in an appropriate way the sequence $\{x_k\}$ converges to a minimum solution of (1.1.1). Under certain uniqueness conditions the sequence $\{u^{(k)}\}$ converges to the vector $\bar{u}$ of Lagrangian multipliers associated with the minimum solution $\bar{x}$ of (1.1.1). An algorithm of this kind was first proposed by Benders (1960), later on by Everett (1963) and Falk (1967a). Roode (1968) introduced the concept of a generalized Lagrangian function in order to find a unifying approach to these methods and the loss-function techniques. Recently, Fletcher (1969b) has proposed a method which is also based on the Lagrangian function. His method is not one of sequential minimization, however. It operates with a continuously varying approximation to the Lagrangian multipliers. Unfortunately, whereas theoretical work has been extensive, all these Lagrangian methods suffer from a serious lack of computational experience.

The idea of *sequential* minimization was also dropped in the modified interior-point method of Zoutendijk (1966). This is a variant of the barrier-function techniques, but it does not approach a constrained minimum via a sequence of unconstrained r-minima. It presents a more direct approach: one has to perform a sequence of univariate searches each of which starts with a smaller value of the controlling parameter than the previous one. A similar method based on loss functions was proposed by Butz (1967). As far as we know, however, these methods have not been tested on practical usefulness. Even the authors did not mention any computational experience.

This concludes our survey. It is clear that there are still many promising directions for future research, particularly if one is concerned with the computational efficiency of these methods.

List of conditions

Condition 2.1. The multipliers $\bar{u}_i$, $i \in A(\bar{x})$, are positive.

Condition 2.2. The gradients $\nabla g_i(\bar{x})$, $i \in A(\bar{x})$, are linearly independent.

Condition 2.3. For any $y \in E_n$, $y \neq 0$, such that $\nabla g_i(\bar{x})^T y = 0, i \in A(\bar{x})$, it must be true that $y^T D(\bar{x},\bar{u}) y > 0$.

Condition 3.1. Problem (1.1.1) is a convex-programming problem. The constraint set R is compact. The set $R_1{}^{\circ} \cap R_2$ is nonempty.

Condition 3.2. The function φ is concave and nondecreasing in the interval $(0,\infty)$, and $\varphi(0+) = -\infty$.

Condition 3.3. The function ψ is concave and nondecreasing in the interval $(-\infty,\infty)$; $\psi(\eta) = 0$ for $\eta \geqslant 0$ and $\psi(\eta) < 0$ for $\eta < 0$.

Condition 3.4. There are positive numbers P and p such that $\psi(\eta) < -P|\eta|^{1+p}$ for any $\eta < 0$.

Condition 3.5. The function φ has a continuous first-order derivative φ' in the interval $(0,\infty)$.

Condition 3.6. The function ψ has a continuous first-order derivative ψ' in the interval $(-\infty,\infty)$.

Condition 3.7. There is a positive number φ_0 such that φ' is analytic in the interval $(-\varphi_0,\infty)$, except at $\eta = 0$; it has a pole of order λ at $\eta = 0$.

Condition 3.8. There is a positive number ω_0 such that ω' is analytic in the interval $(-\infty,\omega_0)$; it has a zero of order μ at $\eta = 0$.

Condition 4.1. Problem (1.1.1) is a convex-programming problem. The constraint set R is compact and its interior R° is nonempty.

Condition 4.2. Problem (1.1.1) is a convex-programming problem with a nonempty, compact constraint set R.

Appendix

An ALGOL 60 procedure for constrained minimization via a mixed parametric firstorder penalty function

The procedure "minimize" is an ALGOL 60 procedure designed for solving the constrained-minimization problem

$$\left.\begin{array}{l} \text{minimize } f(x) \text{ subject to} \\ \qquad g_i(x) \geqslant 0; \quad i = 1, \ldots, m, \\ \qquad h_j(x) = 0; \quad j = 1, \ldots, p, \end{array}\right\} \tag{1}$$

where x denotes a vector in the n-dimensional vector space E_n. It can also be used for unconstrained minimization, in which case the problem reads

$$\text{minimize } f(x). \tag{2}$$

In addition to this, "minimize" can be used merely to find a solution of the system of (in)equalities

$$\left.\begin{array}{l} g_i(x) \geqslant 0; \quad i = 1, \ldots, m, \\ h_j(x) = 0; \quad j = 1, \ldots, p. \end{array}\right\} \tag{3}$$

The penalty function

In solving problem (1) the procedure deals with the penalty function

$$P_r(x) = f(x) + r\, b(x) + r^{-1}\, [l(x) + e(x)] \tag{4}$$

containing the logarithmic barrier term

$$b(x) = -\sum_{i \epsilon I_1} \ln g_i(x),$$

and the quadratic loss terms

$$l(x) = \sum_{i \epsilon I_2} \{\min\, [0, g_i(x)]\}^2$$

and

$$e(x) = \sum_{j=1}^{p} h_j^2(x).$$

The index sets I_1 and I_2 are defined by

$$\begin{array}{l} I_1 = \{i \,| g_i(x_0) > 0; \quad 1 \leqslant i \leqslant m\}, \\ I_2 = \{i \,| g_i(x_0) \leqslant 0; \quad 1 \leqslant i \leqslant m\}, \end{array}$$

where x_0 denotes the (user-supplied) starting point of the computational process for solving problem (1). The parameter r appearing in (4) controls the convergence to a minimum solution $\bar{x}$ of problem (1). Let $x(r_k)$ be a point minimizing (4) over the set

$$R_1{}^{\circ} = \{x \mid g_i(x) > 0; \quad i \in I_1\}$$

for a fixed, positive value r_k of r. Under mild conditions one has

$$\lim_{k \to \infty} f\,[x(r_k)] = f(\bar{x})$$

and

$$\lim_{k \to \infty} \{l[x(r_k)] + e[x(r_k)]\} = 0,$$

provided that $\{r_k\}$ is a monotonic, decreasing null sequence for $k \to \infty$. Hence, any limit point of the sequence $\{x(r_k)\}$ is a minimum solution of (1). This provides the framework of the algorithm.

If certain uniqueness conditions are satisfied at a minimum solution $\bar{x}$, and if the problem functions f, $g_1, \ldots, g_m$, $h_1, \ldots, h_p$ have continuous second-order partial derivatives in E_n, then the trajectory $x(r)$ of points minimizing (4) is a continuously differentiable vector function of r in a neighbourhood of $r = 0$. The kth-order derivative of $x(r)$ exists accordingly as the problem functions admit of $(k + 1)$th-order partial derivatives $(k \geqslant 1)$. Then $x(r)$ can be expanded in a Taylor series about $\bar{x}$. This provides the basis for an extrapolation device in order to obtain a more accurate approximation of $\bar{x}$.

The controlling parameter r is initially given the value r_0 defined by

$$r_0 = \max\left(10^{-2}, \frac{|v^*|}{100}\right), \tag{5}$$

where v^* denotes an estimate of the minimum value of problem (1). Successive values $r_1, r_2, \ldots$ assigned to r are generated in accordance with the rule

$$r_k = \frac{r_{k-1}}{10^{1/3}}.$$

"Minimize" does not go beyond sixth-order extrapolation. Let $X^{(k,l)}$ denote the approximation of $\bar{x}$ based on $x(r_{k-l}), \ldots, x(r_k)$. The computations will be stopped as soon as two successive approximations differ in each component by less than a relative accuracy ε_1 and an absolute accuracy ε_2, i.e.

$$|X_j{}^{(k,l)} - X_j{}^{(k-1,l-1)}| < \varepsilon_1 |X_j{}^{(k,l)}| + \varepsilon_2; \quad j = 1, \ldots, n,$$

whereafter $X^{(k,l)}$ is delivered as an approximation of the minimum solution $\bar{x}$. The accuracies ε_1 and ε_2 are to be supplied by the user.

The algorithm just sketched is clearly one whereby solving a constrained-minimization problem is reduced to (sequential) unconstrained minimization of a penalty function. It is convenient to extend the concept of a penalty function to the cases where problems (2) or (3) are involved. If unconstrained minimization of one single function f is required the penalty function reduces to f. If "minimize" is used to find a solution of the system (3) the penalty function is given by

$$P(x) = \sum_{i=1}^{m} \{\min\,[0, g_i(x)]\}^2 + \sum_{j=1}^{p} h_j^2(x).$$

Unconstrained minimization

Unconstrained minimization is largely carried out in accordance with the algorithm of Davidon (1959) as amended and described by Fletcher and Powell (1963). Only two (minor) modifications have been introduced.

(1) The direction matrix H_i is reset to the unit matrix if the iteration number i is a multiple of $2n$ and if the length of the gradient of the penalty function at the corresponding iteration point happens to be greater than 1. A similar (although more frequent) resetting strategy was recommended by Pearson (1969). We are more reluctant, however, to reset the direction matrix.

(2) The line search uses penalty-function values only. The gradient of the penalty function is not evaluated at the trial points along the direction of search.

As soon as during n iterations two successive iteration points differ in each component by less than the relative accuracy ε_1 and the absolute accuracy ε_2, the process terminates. The last iteration point is then delivered as an approximation of a (local) unconstrained minimum.

Line searches

A minimum along a line is obtained by repeated application of quadratic interpolation. To start with, a step is taken in the direction of search and the penalty function is recalculated. If this value is less than or equal to the initial value at the current iteration point, the step length is multiplied by 2 and a further move is made in the direction of search. This process is repeated until a step is performed resulting in an increase of the penalty function, indicating

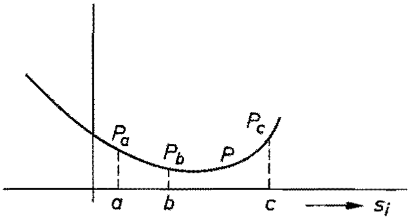

Fig. A.1.

that a line minimum has been overshot. A typical situation is illustrated in fig. A.1. Here one has three "line coordinates" a, b and c with the corresponding penalty-function values P_a, P_b and P_c. The value P_a, for instance, is given by

$$P_a = P(x_i + a\, s_i),$$

where x_i denotes the current iteration point, s_i the direction of search from x_i, and P the penalty function. Let d be the coordinate of the point which minimizes the interpolating quadratic. One of the two extremes a and c is then removed according to a rule which guarantees that the remaining coordinates embracket a minimum. After this rearrangement quadratic interpolation is repeated until two line coordinates b and d are obtained such that the corresponding points $x_i + b\, s_i$ and $x_i + d\, s_i$ differ in each component by less than the relative accuracy ε_1 and the absolute accuracy ε_2. Then the point with the smallest penalty-function value is delivered as an approximation of a line minimum.

If the first step in the search direction yields a point c with a penalty-function value $P_c = P(x_i + c\, s_i)$ greater than the initial value $P_0 = P(x_i)$, quadratic interpolation starts at once. The information being used consists of the function values P_0 and P_c, and the slope of the penalty function at x_i (the inner product of the gradient of P at x_i and the search direction s_i).

Directions for use

When "minimize" is called the actual parameters, which, for convenience, are denoted by the corresponding formal parameters, have the following meaning:

x real array with elements $x[1], \ldots, x[n]$; before calling "minimize" the starting point of the computations must be stored in x; on return x contains the solution produced by "minimize";

functions procedure with two parameters x and gx; by a call the elements of the real array $gx[1:m]$ are evaluated for the current value of the elements of the real array $x[1:n]$; via this procedure the user supplies the problem functions $gx[1], \ldots, gx[m]$ as functions of the independent variables $x[1], \ldots, x[n]$; the declaration reads

```
procedure functions (x,gx);
real array x, gx; ⟨body⟩;
```

xtype integer array with elements $xtype[1], \ldots, xtype[n]$; before calling "minimize" $xtype[j]$ must be assigned the value 0, 1 or 2; if $xtype[j] = 0$ then $x[j]$ remains unchanged during the computations and the derivative of the penalty function with respect to $x[j]$ is set to 0; if $xtype[j] = 1$ then $x[j]$ is a free variable; if *xtype*[*j*] = 2 then *x*[*j*] is required to be nonnegative;

gtype integer array with elements *gtype*[1], . . . , *gtype*[*m*]; before calling “minimize” these elements must be assigned the values 0, 1, 2 or 3; if *gtype*[*i*] = 0 then *gx*[*i*] will be dropped from the problem; if *gtype*[*i*] = 1 then *gx*[*i*] is taken to be the objective function; if *gtype*[*i*] = 2 then the constraint $gx[i] \geqslant 0$ is imposed; if *gtype*[*i*] = 3 then the constraint $gx[i] = 0$ is imposed;

analytic boolean expression; if *analytic* is **true** it is assumed that the first-order derivatives of the problem functions are supplied by the user; if it is **false** “minimize” will compute the first-order derivatives by differencing;

derivatives procedure with two parameters *x* and *dgdx*; by a call the elements of the real array *dgdx*[1 : *m*, 1 : *n*] are evaluated for the current value of the elements of the real array *x*[1 : *n*]; it is assumed that *dgdx*[*i,j*] is the first-order partial derivative of *gx*[*i*] with respect to *x*[*j*]; only non-zero derivatives need be supplied; the declaration reads

procedure *derivatives* (*x*,*dgdx*);
real array *x*, *dgdx*; ⟨body⟩;

n integer expression; dimension of the arrays *x*[1 : *n*] and *xtype* [1 : *n*];

m integer expression; dimension of the array *gtype*[1 : *m*];

two boolean expression; if *two* is **true** numerical differentiation takes two function values per derivative; otherwise it will take four function values;

raxmin real expression; relative accuracy of the solution to be produced; a value of 10^{-5} is suggested;

aaxmin real expression; absolute accuracy of the solution to be produced; a value of 10^{-5} is suggested;

estimate real expression; an estimate of the value of the objective function at a minimum solution; it is only used in **constrained** minimization in order to find the value r_0 of formula (5);

converged boolean variable; *converged* is set to **true** by “minimize” if the above-named convergence criteria are satisfied; otherwise *converged* is set to **false**; this may occur, for instance, if the process would take more than the maximum number of iterations permitted by the user or if constrained minimization would involve more than 10 unconstrained-minimization cycles;

coc integer expression; conditional output-control parameter; if $coc = 0$ then no output will be given; if $coc = 1$ then "minimize" produces short output in every iteration, and x-output and g-output in the first and the last iteration of an unconstrained-minimization cycle; in addition to this, x-output appears in every iteration if $coc = 2$; an explanation of the output will be given below;

imax integer expression; the maximum number of iterations permitted by the user; if $imax = 0$ then no restriction is imposed on the number of iterations.

Output

"Minimize" uses some output procedures of the ALGOL 60 system designed in the Mathematical Centre in Amsterdam for the Electrologica X8 computer. They can readily be removed from "minimize" or replaced by the output procedures of other systems.

Short output is a single-line summary of the current iteration presenting the iteration number, the value of the penalty function and the length of the gradient of the penalty function both evaluated at the current iteration point, and the distance between the current and the preceding iteration point.

***X*-output** is a print of the current iteration point (solution vector) and the gradient of the penalty function evaluated at this point. If *analytic* is **true** the gradient is obtained by evaluating the user-supplied first-order derivatives of the problem functions; otherwise the gradient is approximated by differences of function values. In iteration 0, if *analytic* is **true,** this approximation is also computed and printed in order that users derivatives can be checked.

***G*-output** is a print of the values of the problem functions at the current iteration point. If the problem is one of constrained minimization, it also comprises the current approximation of the Lagrangian multipliers.

The output procedures being used are the following ones:

ABSFIXT(n,m,x) when this procedure is called the absolute value of x will be printed in fixed-point representation: one space, n decimal digits (leading zeroes being replaced by spaces), decimal point, m decimal digits, one space; if $m = 0$ the decimal point is not printed;

CARRIAGE(n) causes the printer to advance the paper n lines and to take the print position at the beginning of the line;

FLOT(n,m,x) when this procedure is called the value of x will be printed in floating-point representation: sign, decimal point, n decimal digits, the symbol $_{10}$, sign, m decimal digits (leading zeroes being replaced by spaces), one space;

NLRC New Line Carriage Return;

PRINTTEXT(*s*) when this procedure is called the string *s* will be printed without the outermost quotes;

SPACE(*n*) causes the printer to move over *n* spaces on the current line.

```
comment page 1 of minimize;
procedure minimize(x,functions,xtype,gtype,analytic,derivatives,
  n,m,two,raxmin,aaxmin,estimate,converged,coc,imax);
value n,m,raxmin,aaxmin,estimate,coc,imax; integer n,m,coc,imax;
real raxmin,aaxmin,estimate; boolean analytic,two,converged;
real array x; integer array xtype,gtype;
procedure functions,derivatives;
begin real initr,factor;
  integer i,j,k,nv,ieff,cycle,maxcycle,nc1,nc2,nc3,nsearch,nrp;
  boolean objective,constraints,nlp,heading,nonneg;
  real array xrmin,xmin[1:n],urmin,umin[1:m],
  xtable[1:n,0:6],utable[1:m,0:6];
  boolean array logp[1:m+n];

  procedure rcycle(r,xr,ur,raxr,aaxr); value r;
  real r,raxr,aaxr; real array xr,ur;
  comment unconstrained-minimization cycle for computing an r-minimum
  xr[1:n] and the associated dual multipliers ur[1:m] with relative
  and absolute accuracy raxr and aaxr respectively;
  begin integer itercnt,counter,reset; real prxr,gradl,distance;
    real array grad,dgrad,dir,sigma,yvec[1:n],h[1:n,1:n],gxr[1:m];

    real procedure penalty(p,t,q,gt,reject);
    integer reject; real t; real array p,q,gt;
    comment computes the problem functions gt[1:m] and the penalty
    function at the point p[1:n] + t × q[1:n] — reject indicates
    whether this point is feasible or not: it is the index of the
    first encountered, violated constraint;
    begin real barrier,loss,pen; real array xt[1:n];
      penalty := pen := loss := 0; reject := 0; barrier := 1;
      for j := 1 step 1 until n do xt[j] := p[j] + t × q[j];
      if  nonneg then
      begin for j := 1 step 1 until n do if xtype[j] = 2 then
        begin if  logp[m + j] then
          begin if xt[j] > ₁₀-10 then barrier := barrier × xt[j] else
            begin reject := m + j; nrp := nrp + 1; goto fin end
          end of logarithmic transformation
          else if xt[j] < 0 then loss := loss + xt[j]∧2
        end of j loop
      end  of handling nonnegative variables;
      functions(xt,gt); nc2 := nc2 + 1;
      for i := 1 step 1 until m do
      begin if gtype[i] = 1 then pen := pen + gt[i] else
        begin if gtype[i] = 2 then
          begin if logp[i] then
            begin if gt[i] > ₁₀-10 then barrier := barrier × gt[i] else
              begin reject := i; nrp := nrp + 1; goto fin end
            end of logarithmic transformation
            else if gt[i] < 0 then loss := loss + gt[i]∧2
          end of handling inequality constraints
          else if gtype[i] = 3 then loss := loss + gt[i]∧2
        end of transforming constraints
      end of i loop for generating barrier and loss term;
      penalty := pen - r × ln(barrier) + loss/r;
    fin:
    end of penalty for computing the problem functions
    and the mixed penalty function;
```

```
comment page 2 of minimize — continuation 1 of rcycle;

    procedure gradient(xt,gt,dptdx,analytic);
    real array xt,gt,dptdx; boolean analytic;
    comment computes the gradient dptdx[1:n] of the penalty
    function at the point xt[1:n] — before calling the procedure
    gradient the array gt[1:m] must contain the values of the
    problem functions at this point — analytic indicates whether
    user-supplied derivatives are available or not;
    begin real hj,dj,xtj,dpj; integer upper;
      real array gdelta,dgdxj[1:m],diff[1:m,1:2],dgdx[1:m,1:n];
      upper := if two then 1 else 2;
      if  analytic then
      begin for  j := 1 step  1 until  n do
        for  i := 1 step  1 until  m do  dgdx[i,j] := 0;
        derivatives(xt,dgdx); nc3  := nc3  + 1
      end of computing users derivatives at xt[1:n];
      for j := 1 step 1 until n do
      if xtype[j] = 0 then dptdx[j] := 0 else
      begin xtj  := xt[j];
        if analytic then
        begin for  i := 1 step  1 until  m do
          dgdxj[i] := dgdx[i,j]
        end of storing derivatives with respect to xtj else
        begin comment compute derivatives with respect to xtj;
          hj := ₁₀-2 × abs(xtj) + ₁₀-5;
          for k := 1 step 1 until upper do
          begin dj  := hj/k; nc1  := nc1  + 2;
            xt[j]  := xtj + dj; functions(xt,gdelta);
            for i  := 1 step 1 until m do diff[i,k]  := gdelta[i];
            xt[j]  := xtj - dj; functions(xt,gdelta);
            for  i := 1 step  1 until  m do
            diff[i,k] := (diff[i,k] - gdelta[i])/(2 × dj)
          end of k loop; xt[j]  := xtj;
          for i  := 1 step 1 until m do dgdxj[i]  :=
          if two then diff[i,1] else (4 × diff[i,2] - diff[i,1])/3
        end  of generating dgi/dxj;
        dpj := 0;
        for i  := 1 step 1 until m do
        begin if gtype[i] = 1 then dpj  := dpj + dgdxj[i] else
          begin if gtype[i] = 2 then
            begin if logp[i] then dpj  := dpj - r × dgdxj[i]/gt[i] else
              if gt[i] < 0 then dpj  := dpj + 2 × dgdxj[i] × gt[i]/r
            end of handling inequality constraints
            else if gtype[i] = 3 then dpj := dpj + 2 × dgdxj[i] × gt[i]/r
          end of handling constraint derivatives
        end of i loop;
        if xtype[j] = 2 then
        begin if logp[m + j] then dpj  := dpj - r/xtj else
          if xtj < 0 then dpj  := dpj + 2 × xtj/r
        end of handling variable constrained to nonnegative values;
        dptdx[j]  := dpj
      end of j loop for computing j-th component of gradient
    end of gradient for computing the first-order derivatives
    of the penalty function;
```

```
comment page 3 of minimize — continuation 2 of rcycle;
    procedure linemin(point,prpoint,gpoint,gradpr,s,diffp,diffg,
    racc,aacc); value racc,aacc; real prpoint,racc,aacc;
    real array point,gpoint,gradpr,s,diffp,diffg;
    comment procedure for searching a minimum with relative and
    absolute accuracy racc and aacc respectively along the direction
    s[1:n] emanating from point[1:n] — prpoint and gpoint[1:m] must
    contain the values of the penalty function and the problem
    functions at point[1:n] — gradpr[1:n] is the gradient of the
    penalty function at point[1:n] — on return from linemin the line
    minimum is contained in point[1:n], and prpoint,gpoint[1:m] and
    gradpr[1:n] contain the corresponding data — the difference
    between the line minimum and the starting point of the search is
    stored in diffp[1:n] — the difference of the corresponding
    gradients is contained in diffg[1:n];
    begin integer nsteps,idle;
      real lambda,pra,prb,prc,prd,a,b,c,d,e,multiplier,descent,pa,pc;
      real array ga,gb,gc,gd[1:m];

      boolean procedure ready(new,old); real new,old;
      begin real sj,r1,a1;
        ready := true ; r1 := racc/abs(old − new); a1 := aacc/abs(old − new);
        for  j := 1 step  1 until  n do if  xtype[j] ≠ 0 then
        begin sj := s[j]; if abs(sj) > r1 × abs(point[j] + new × sj) + a1 then
          begin ready := false ; j := n end
        end of j loop
      end of ready for comparing two successive approximations
      of a minimum along the direction s[1:n];

      procedure shift1(y,z,pry,prz,gy,gz);
      real y,z,pry,prz; real array gy,gz;
      begin y := z; pry := prz;
        for i := 1 step 1 until m do gy[i] := gz[i]
      end of shift1;

      nsearch := nsearch + 1;
      descent := vecvec(1,n,0,gradpr,s);
      multiplier := 2;
      lambda := abs(1/descent);
      if lambda > 1 then lambda := 1;
      a := b := c := 0; pra := prb := prc := prpoint;
      for i := 1 step 1 until m do
      ga[i] := gb[i] := gc[i] := gpoint[i];
    forward: for nsteps := 1 step 1 until 50 do
      begin c := b + lambda;
        prc := penalty(point,c,s,gc,idle); if idle > 0 then
        begin lambda := lambda/2;
          multiplier := 0.5
        end  of backward step
        else
        begin if prc ≤ prb then
          begin lambda := multiplier × lambda;
            shift1(a,b,pra,prb,ga,gb);
            shift1(b,c,prb,prc,gb,gc);
          end of forward step
          else goto inter1
        end of testing the feasible trial point c
      end of nsteps loop for moving in search direction;
```

```
comment page 4 of minimize — cont. 3 of rcycle — cont. 1 of linemin;
    inter1: if a = 0 ∧ b = 0 ∧ c > 0 ∧ descent < 0 ∧ prb < prc then
      begin comment quadratic interpolation using descent of
        penalty function in the starting point of the search;
        for nsteps := 1 step 1 until 50 do
        begin comment d minimizes interpolating quadratic;
          d := -(0.5 × descent × c↑2)/(prc - prb - descent × c);
          prd := penalty(point,d,s,gd,idle); if idle > 0 then
          begin lambda := d/2; multiplier := 0.5; goto forward end ;
          if prd > prb then
          begin if ready(c,0) then goto last;
            shift1(c,d,prc,prd,gc,gd)
          end of reducing search interval (0,c)
          else
          begin shift1(b,d,prb,prd,gb,gd); goto inter2 end
        end of nsteps loop for sequential interpolation
      end of quadratic interpolation using descent of penf;
    inter2: if a < b ∧ b < c ∧ pra ≥ prb ∧ prb ≤ prc then
      begin comment quadratic interpolation on three points a,b,c;
        for nsteps := 1 step 1 until 50 do
        begin comment d minimizes interpolating quadratic;
          pc := (a - b) × (prc - prb);
          if pc = 0 then d := (b + c)/2 else
          begin pa := (b - c) × (pra - prb);
            d := 0.5 × ((a + b) × pc + (b + c) × pa)/(pa + pc)
          end of computing d;
          e := b; comment save best approximation so far;
          prd := penalty(point,d,s,gd,idle); if idle > 0 then
          begin if d > b then lambda := (d - b)/2 else
            begin lambda := (d - a)/2;
              shift1(b,a,prb,pra,gb,ga);
              shift1(a,0,pra,prpoint,ga,gpoint)
            end of shifting towards a;
            multiplier := 0.5; goto forward
          end of backward step due to constraint violation;
          if d < b then
          begin if  prd < prb then
            begin shift1(c,b,prc,prb,gc,gb);
              shift1(b,d,prb,prd,gb,gd)
            end else shift1(a,d,pra,prd,ga,gd)
          end else
          begin if  prd < prb then
            begin shift1(a,b,pra,prb,ga,gb);
              shift1(b,d,prb,prd,gb,gd)
            end else shift1(c,d,prc,prd,gc,gd)
          end of rearranging a,b,and c;
          if ready(b,e) then goto last
        end of nsteps loop for sequential interpolation
      end of quadratic interpolation on three points a,b,c;
    last: prpoint := prb;
      for j := 1 step 1 until n do
      begin diffp[j] := b × s[j]; diffg[j] := gradpr[j];
        point[j] := point[j] + diffp[j]
      end of moving iteration point;
      for i := 1 step 1 until m do gpoint[i] := gb[i];
      gradient(point,gpoint,gradpr,analytic);
      for j := 1 step 1 until n do diffg[j] := gradpr[j] - diffg[j]
    end of linemin for one-dimensional minimization;
```

```
comment page 5 of minimize — continuation 4 of rcycle;

    procedure output1(cycle,r,itercnt,ieff,analytic,converged,nlp,
    heading,prxr,gradl,distance,xr,grad,dgrad,gxr,ur,n,m,coc);
    real r,prxr,gradl,distance;
    integer cycle,itercnt,ieff,n,m,coc;
    boolean analytic,converged,nlp,heading;
    real array xr,grad,dgrad,gxr,ur;
    begin integer i,j;
      if itercnt = 0 ∧ nlp then
      begin CARRIAGE(2);
        PRINTTEXT(≮ begin of cycle ≯); ABSFIXT(2,0,cycle);
        PRINTTEXT(≮ for r equal to ≯); FLOT(3,3,r)
      end ;
      comment short output;
      if heading then
      begin CARRIAGE(2); PRINTTEXT(≮ iteration   ≯);
        PRINTTEXT(≮ penalty value      gradient length ≯);
        SPACE(8); PRINTTEXT(≮ distance ≯)
      end of printing the heading;
      NLCR; ABSFIXT(6,0,ieff); SPACE(6); FLOT(6,3,prxr); SPACE(7);
      FLOT(6,3,gradl); SPACE(7); FLOT(6,3,distance);
      heading := false ;
      comment end of short output;
      if ieff = 0 ∨ converged ∨ coc = 2 then
      begin comment x-output; CARRIAGE(2);
        PRINTTEXT(≮  variable   solution vector    ≯);
        PRINTTEXT(≮ gradient of penf ≯); SPACE(4);
        if  analytic ∧ ieff = 0 then
        PRINTTEXT(≮ grad(differences) ≯);
        for j := 1 step 1 until n do
        begin NLCR; PRINTTEXT(≮x ≯); ABSFIXT(4,0,j); SPACE(6);
          FLOT(6,3,xr[j]); SPACE(7); FLOT(6,3,grad[j]); SPACE(7);
          if analytic ∧ ieff = 0 then FLOT(6,3,dgrad[j])
        end of printing iteration point and gradient;
        heading := true
      end of x-output;
      if  ieff = 0 ∨ converged then
      begin comment g-output; CARRIAGE(2);
        PRINTTEXT(≮  function   function values      ≯);
        if nlp ∧ converged then
        PRINTTEXT(≮ dual solution ≯);
        for i := 1 step 1 until m do
        begin NLCR; PRINTTEXT(≮gx≯); ABSFIXT(4,0,i); SPACE(6);
          FLOT(6,3,gxr[i]); SPACE(7);
          if nlp ∧ converged then FLOT(6,3,ur[i])
        end of printing function values and dual solution;
        heading := true
      end of g-output;
      if  nlp ∧ converged then
      begin CARRIAGE(2);
        PRINTTEXT(≮ end of cycle ≯); ABSFIXT(2,0,cycle);
        PRINTTEXT(≮ for r equal to ≯); FLOT(3,3,r)
      end
    end of output1 for printing iteration data;
```

```
comment page 6 of minimize — continuation 5 of rcycle;
    comment start of main program of rcycle;
    itercnt := counter := 0; distance := 0; converged := false ;
    for j := 1 step 1 until n do
    grad[j] := dir[j] := sigma[j] := yvec[j] := 0;
    for i := 1 step 1 until m do gxr[i] := ur[i] := 0;
    restart: prxr := penalty(xr,0,xr,gxr,reset);
    if reset > 0 then begin logp[reset] := false ; goto  restart end ;
    gradient(xr,gxr,grad,analytic);
    if analytic ∧ ieff = 0 then gradient(xr,gxr,dgrad,false );
    gradl := sqrt(vecvec(1,n,0,grad,grad));
    for itercnt := 0 step 1 until counter + nv do
    begin comment unconstrained-minimization cycle;
      if (itercnt = 0)∨(itercnt = (itercnt ÷ (2 × nv))
      × 2 × nv ∧ gradl > 1) then
      begin for i := 1 step 1 until n do
        for k := 1 step 1 until n do
        h[i,k] := if i = k then 1 else 0
      end of resetting direction matrix else
      begin real si,hi,sigmay,yhy; real array hy[1:n];
        for i := 1 step 1 until n do
        hy[i] := matvec(1,n,i,h,yvec);
        sigmay := vecvec(1,n,0,sigma,yvec);
        yhy := vecvec(1,n,0,yvec,hy);
        for i := 1 step 1 until n do
        begin si := sigma[i]/sigmay; hi := hy[i]/yhy;
          for k := 1 step 1 until i do
          h[k,i] := h[i,k] := h[i,k] + si × sigma[k] - hi × hy[k]
        end of adding corrections
      end of updating direction matrix;
      for j := 1 step 1 until n do dir[j] := matvec(1,n,j,h,grad);
      if vecvec(1,n,0,grad,dir) > 0 then
      for j := 1 step 1 until n do dir[j] := -dir[j];
      if coc ≠ 0 then output1(cycle,r,itercnt,ieff,analytic,converged,
      nlp,heading,prxr,gradl,distance,xr,grad,dgrad,gxr,ur,n,m,coc);
      ieff := ieff + 1; if ieff = imax then goto last;
      linemin(xr,prxr,gxr,grad,dir,sigma,yvec,raxr,aaxr);
      gradl := sqrt(vecvec(1,n,0,grad,grad));
      distance := sqrt(vecvec(1,n,0,sigma,sigma));
      if distance = 0 then goto endpoint;
      for  j := 1 step  1 until  n do if  xtype[j] ≠ 0 then
      begin if abs(sigma[j]) > raxr × abs(xr[j]) + aaxr then
        begin counter := itercnt; j := n end
      end of testing the accuracy
    end  of unconstrained minimization;
  endpoint: converged := true ; if nlp then
    for i := 1 step  1 until  m do
    begin if gtype[i] = 1 then ur[i] := 1 else
      begin if gtype[i] = 2 then
        begin if logp[i] then ur[i] := r/gxr[i] else
          ur[i] := if gxr[i] < 0 then -2 × gxr[i]/r else 0
        end of computing multipliers for inequalities
        else if gtype[i] = 3 then ur[i] := 2 × gxr[i]/r
      end of computing constraint multipliers
    end  of generating dual solution;
    if coc ≠ 0 then output1(cycle,r,itercnt,ieff,analytic,converged,
    nlp,heading,prxr,gradl,distance,xr,grad,dgrad,gxr,ur,n,m,coc);
  last:
  end of rcycle for unconstrained minimization of penalty function;
```

```
comment page 7 of minimize;

  procedure extrapol(t,order,dim,new,result,conv);
  integer order,dim; boolean conv; real array t,new,result;
  begin integer k; real beta; real array aid[1:dim,0:order];
    conv := false ;
    for k := 0 step 1 until order do
    for j := 1 step 1 until dim do aid[j,k] := t[j,k];
    for j := 1 step 1 until dim do t[j,0] := new[j];
    for k := 1 step 1 until order do
    begin beta := 1/(1 - factor↑k); conv := true ;
      for j := 1 step 1 until dim do
      t[j,k] := beta × t[j,k - 1] + (1 - beta) × aid[j,k - 1];
      for j := 1 step 1 until dim do if xtype[j] ≠ 0 then
      begin if abs(t[j,k] - aid[j,k - 1]) > raxmin × abs(t[j,k]) +
        aaxmin then begin conv := false ; j := dim end
      end of testing j-th order approximation;
      if conv then order := k
    end of k loop for table updating and testing;
    for j := 1 step 1 until dim do result[j] := t[j,order]
  end of extrapol for extrapolation towards constrained minimum;

  procedure output2(nc1,nc2,nc3,nsearch,nrp,nv);
  integer nc1,nc2,nc3,nsearch,nrp,nv;
  begin CARRIAGE(2);ABSFIXT(5,0,nc1);
    PRINTTEXT(‹ evaluations of functions for num. diff. ›);
    NLCR; ABSFIXT(5,0,nc2);
    PRINTTEXT(‹ evaluations of functions for line searches ›);
    NLCR; ABSFIXT(5,0,nc3);
    PRINTTEXT(‹ evaluations of derivatives ›);
    NLCR; ABSFIXT(5,0,nc1 + nc2 + nc3 × nv);
    PRINTTEXT(‹ aequivalent function evaluations ›);
    NLCR; ABSFIXT(5,0,nc2/nsearch);
    PRINTTEXT(‹ evaluations of functions per line minimum ›);
    NLCR; ABSFIXT(5,0,nrp);
    PRINTTEXT(‹ rejected points because of constraint violation ›)
  end of output2 for printing number of function evaluations;

  procedure output3(cycle,order,n,m,xmin,umin,functions,heading);
  integer cycle,order,n,m; boolean heading;
  real array xmin,umin; procedure functions;
  begin integer i,j; real array gmin[1:m]; CARRIAGE(2);
    PRINTTEXT(‹ extrapolation, cycle ›); ABSFIXT(2,0,cycle);
    PRINTTEXT(‹ , order ›); ABSFIXT(2,0,order);
    CARRIAGE(2);PRINTTEXT(‹  variable   solution vector ›);
    for j := 1 step 1 until n do
    begin NLCR; PRINTTEXT(‹x ›);
      ABSFIXT(4,0,j); SPACE(6); FLOT(6,3,xmin[j])
    end of x-output;
    CARRIAGE(2);PRINTTEXT(‹  function   function values
    PRINTTEXT(‹ dual solution ›); functions(xmin,gmin);
    for i := 1 step 1 until m do
    begin NLCR; PRINTTEXT(‹gx›); ABSFIXT(4,0,i); SPACE(6);
      FLOT(6,3,gmin[i]); SPACE(7); FLOT(6,3,umin[i])
    end of g-output; heading := true
  end of output3 for printing extrapolated solutions;
```

```
comment page 8 of minimize;

  real procedure vecvec(l,u,shift,a,b); value l,u,shift;
  integer l,u,shift; real array a,b;
  begin integer k; real s; s := 0; for k := l step 1 until u do
    s := a[k] × b[shift + k] + s; vecvec := s
  end of vecvec for computing the inner product of vectors a and b;

  real procedure matvec(l,u,i,a,b); value l,u,i;
  integer l,u,i; real array a,b;
  begin integer k; real s; s := 0; for k := l step 1 until u do
    s := a[i,k] × b[k] + s; matvec := s
  end of matvec for computing i-th element of matrix a × vector b;

  comment start of main program of minimize;
  nv := ieff := nc1 := nc2 := nc3 := nsearch := nrp := 0;
  heading := true ; maxcycle := 0;
  objective := constraints := nlp := nonneg := false ;
  initr := 1; factor := 10∧(-1/3);
  for i := 1 step 1 until m + n do logp[i] := false ;
  for j := 1 step 1 until n do xrmin[j] := xmin[j] := x[j];
  for j := 1 step 1 until n do
  begin if xtype[j] < 0 ∨ xtype[j] > 2 then goto terminal;
    if xtype[j] ≠ 0 then nv := nv + 1;
    if xtype[j] = 2 then nonneg := constraints := true ;
    for k := 0 step 1 until 6 do xtable[j,k] := 0
  end of checking types of variables;
  for i := 1 step 1 until m do
  begin if gtype[i] < 0 ∨ gtype[i] > 3 then goto terminal;
    if gtype[i] = 1 then objective := true else
    if gtype[i] = 2 ∨ gtype[i] = 3 then constraints := true ;
    for k := 0 step 1 until 6 do utable[i,k] := 0
  end of checking constraint types;
  if objective ∧ constraints then
  begin initr := abs(estimate)/100;
    if initr < ⏨-2 then initr := ⏨-2;
    maxcycle := 9; nlp := true ;
    for i := 1 step 1 until m + n do logp[i] := true
  end of initiating constrained minimization;
  for cycle := 0 step 1 until maxcycle do
  begin comment  sequential unconstrained minimization;
    integer orderx,orderu; boolean convx,convu;
    rcycle(initr × factor∧cycle,xrmin,urmin,raxmin,saxmin);
    if ¬ converged then
    begin if cycle = 0 then
      for j := 1 step 1 until n do xmin[j] := xrmin[j];
      goto terminal
    end if there are too many iterations;
    orderx := orderu := if cycle < 6 then cycle else 6;
    extrapol(xtable,orderx,n,xrmin,xmin,convx);
    extrapol(utable,orderu,m,urmin,umin,convu);
    if coc ≠ 0 then output2(nc1,nc2,nc3,nsearch,nrp,nv);
    if cycle > 0 then
    begin if coc ≠ 0 then
      output3(cycle,orderx,n,m,xmin,umin,functions,heading);
      if convx then goto terminal else converged := false
    end of comparing extrapolated solutions
  end of sequential unconstrained minimization;
terminal: for j := 1 step 1 until n do x[j] := xmin[j]
end of minimize;
```

```
comment an example where minimize is called to compute the smallest
distance between the set of points (x[1],x[2],x[3]) such that

          x[1]↑2 + x[2]↑2 + x[3]↑2 ≤ 5,

and the set of points (x[4],x[5],x[6]) satisfying the constraints

           (x[4] - 3)↑2 + x[5]↑2 ≤ 1,
                 4 ≤ x[6] ≤ 8.

— the starting point of the search is chosen to be (1,1,1,3,0,5);


procedure smdist(x,gx); real array x,gx;
comment this procedure is used to supply the objective function
and the constraint functions of the problem;
begin gx[1] := - x[1]↑2 - x[2]↑2 - x[3]↑2 + 5;
  gx[2] := - (x[4] - 3)↑2 - x[5]↑2 + 1;
  gx[3] := - x[6] + 8;
  gx[4] := x[6] - 4;
  gx[5] := (x[1] - x[4])↑2 + (x[2] - x[5])↑2 + (x[3] - x[6])↑2
end of smdist;


procedure smdist1(x,dgdx); real array x,dgdx;
comment this procedure is used to supply the first-order
derivatives of the problem functions;
begin for j := 1,2,3 do dgdx[1,j] := - 2 × x[j];
  dgdx[2,4] := - 2 × (x[4] - 3); dgdx[2,5] := - 2 × x[5];
  dgdx[3,6] := -1; dgdx[4,6] := 1;
  for j := 1,2,3 do dgdx[5,j] := 2 × (x[j] - x[j + 3]);
  for j := 1,2,3 do dgdx[5,j + 3] := - 2 × (x[j] - x[j + 3])
end of smdist1;


for j := 1,2,3 do x[j] := 1; x[4] := 3; x[5] := 0; x[6] := 5;
for j := 1 step 1 until 6 do xtype[j] := 1;
for i := 1,2,3,4 do gtype[i] := 2; gtype[5] := 1;


minimize(x,smdist,xtype,gtype,true ,smdist1,
6,5,true ,₁₀-5,₁₀-5,5,converged,1,100);
```

REFERENCES and LITERATURE

J. Abadie (1967), Nonlinear programming. North-Holland Publ. Comp., Amsterdam.

C. M. Ablow and G. Brigham (1955), An analog solution of programming problems. Opns. Res. **3**, 388-394.

L. V. Ahlfors (1953), Complex analysis. McGraw-Hill, New York.

T. M. Apostol (1957), Mathematical analysis. Addison-Wesley, Reading, Mass.

K. J. Arrow, L. Hurwicz and H. Uzawa (1961), Constraint qualifications in maximization problems. Naval Res. Log. Qu. **8**, 175-191.

Y. Bard (1968), On a numerical instability of Davidonlike methods. Math. of Comp. **22**, 665-666.

E. J. Beltrami (1967), A computational approach to necessary conditions in mathematical programming. ICC Bulletin **6**, 265-273.

E. J. Beltrami (1969a), A constructive proof of the Kuhn-Tucker multiplier rule. J. Math. Anal. Appl. **26**, 297-306.

E. J. Beltrami (1969b), A comparison of some recent iterative methods for the numerical solution of nonlinear programs, in M. Beckmann (ed.), Computing methods in optimization problems. Springer, Berlin, pp. 20-29.

J. F. Benders (1960), Partitioning in mathematical programming. Avanti, Delft.

C. Berge (1959), Espaces topologiques et fonctions multivoques. Dunod, Paris.

C. Berge et A. Ghouila-Houri (1962), Programmes, jeux et réseaux de transport. Dunod, Paris.

M. J. Box (1966), A comparison of several current optimization methods and the use of transformations in constrained problems. The Comp. J. **9**, 67-77.

M. J. Box, D. Davies and W. H. Swann (1969), Nonlinear optimization techniques. ICI Monograph, Oliver and Boyd, Edinburgh.

J. Bracken and G. P. McCormick (1968), Selected applications of nonlinear programming. Wiley, New York.

C. G. Broyden (1965), A class of methods for solving nonlinear simultaneous equations. Math. of Comp. **19**, 577-593.

C. G. Broyden (1967), Quasi-Newton methods and their application to function minimization. Math. of Comp. **21**, 368-381.

Bui Trong Lieu et P. Huard (1966), La méthode des centres dans un espace topologique. Numer. Math. **8**, 56-67.

R. Bulirsch (1964), Bemerkungen zur Romberg-Integration. Numer. Math. **6**, 6-16.

R. Bulirsch und J. Stoer (1964), Fehlerabschätzungen und Extrapolation mit rationalen Funktionen bei Verfahren vom Richardson-Typus. Numer. Math. **6**, 413-427.

R. Bulirsch and J. Stoer (1966), Asymptotic upper and lower bounds for results of extrapolation methods. Numer. Math. **8**, 93-104.

T. Butler and A. V. Martin (1962), On a method of Courant for minimizing functionals. J. Math. Phys. **41**, 291-299.

A. R. Butz (1967), Iterative saddle point techniques. SIAM J. Appl. Math. **15**, 719-725.

G. D. Camp (1955), Inequality-constrained stationary value problems. Opns. Res. **3**, 548-550.

J. Carpentier (1962), Contribution à l'étude du dispatching économique. Bull. Soc. Fr. des El., 8ième série, tome III, 431-447.

C. W. Carroll (1961), The created response surface technique for optimizing nonlinear restrained systems. Opns. Res. **9**, 169-184.

L. Collatz und W. Wetterling (1966), Optimierungsaufgaben. Springer, Berlin.

A. R. Colville (1968a), Mathematical programming codes. IBM New York Scientific Center, Techn. Rep. 320-2925.

A. R. Colville (1968b), A comparative study of nonlinear programming codes. IBM New York Scientific Center, Techn. Rep. 320-2949.

R. W. Cottle (1963), Symmetric dual quadratic programs. Quart. Appl. Math. **21**, 237-243.

R. Courant (1943), Variational methods for the solution of problems of equilibrium and vibrations. Bull. Am. Math. Soc. **49**, 1-23.

J. B. Crockett and H. Chernoff (1955), Gradient methods of maximization. Pac. J. Math. **5**, 33-50.

H. B. Curry (1944), The method of steepest descent for nonlinear minimization problems. Quart. Appl. Math. **2**, 258-261.

J. W. Daniel (1967a), The conjugate gradient method for linear and nonlinear operator equations. SIAM J. Numer. Anal. **4**, 10-26.

J. W. Daniel (1967b), Convergence of the conjugate gradient method with computationally convenient modifications. Numer. Math. **10**, 125-131.

G. B. Dantzig, E. Eisenberg and R. W. Cottle (1965), Symmetric dual nonlinear programs. Pac. J. Math. **15**, 809-812.

W. C. Davidon (1959), Variable metric method for minimization. A.E.C. Research and Development report ANL-5990.

W. C. Davidon (1968), Variance algorithm for minimization. The Comp. J. **10**, 406-410.

D. Davies (1968), The use of Davidon's method in nonlinear programming. Technical paper, Imperial Chemical Industries Ltd, Wilmslow, UK.

W. S. Dorn (1960a), Duality in quadratic programming. Quart. Appl. Math. **18**, 155-162.

W. S. Dorn (1960b), A duality theorem for convex programming. IBM J. Res. Dev. **4**, 407-413.

W. S. Dorn (1961), On Lagrange multipliers and inequalities. Opns. Res. **9**, 95-104.

H. Everett (1963), Generalized Lagrange multiplier method for solving problems of optimum allocation of resources. Opns. Res. **11**, 399-417.

J. E. Falk (1967a), Lagrange multipliers and nonlinear programming. J. Math. Anal. Appl. **19**, 141-159.

J. E. Falk (1967b), A relaxed interior approach to nonlinear programming. Technical paper RAC-TP-279, Research Analysis Corporation, McLean, Va., U.S.A.

P. Faure et P. Huard (1965), Résolution des programmes mathématiques à fonction nonlinéaire par la méthode du gradient réduit. Rev. Fr. Recherche Opér. **9**, 167-205.

P. Faure et P. Huard (1966), Résultats nouveaux relatifs à la méthode des centres. Quatrième conférence de recherche opérationelle, Cambridge, Mass.

A. V. Fiacco (1967), Sequential unconstrained minimization methods for nonlinear programming. Thesis, Northwestern University, Evanston, Illinois.

A. V. Fiacco and G. P. McCormick (1964a), The sequential unconstrained minimization technique for nonlinear programming, a primal-dual method. Management Science **10**, 360-366.

A. V. Fiacco and G. P. McCormick (1964b), Computational algorithm for the sequential unconstrained minimization technique for nonlinear programming. Management Science **10**, 601-617.

A. V. Fiacco and G. P. McCormick (1966), Extensions of SUMT for nonlinear programming: equality constraints and extrapolation. Management Science **12**, 816-828.

A. V. Fiacco and G. P. McCormick (1967a), The slacked unconstrained minimization technique for convex programming. SIAM J. Appl. Math. **15**, 505-515.

A. V. Fiacco and G. P. McCormick (1967b), The sequential unconstrained minimization technique without parameters. Opns. Res. **15**, 820-827.

A. V. Fiacco and G. P. McCormick (1968), Nonlinear programming, sequential unconstrained minimization techniques. Wiley, New York.

A. V. Fiacco and A. P. Jones (1969), Generalized penalty methods in topological spaces. SIAM J. Appl. Math. **17**, 996-1000.

R. Fletcher (1965), Function minimization without evaluating derivatives — a review. The Comp. J. **8**, 33-41.

R. Fletcher (1969a), Optimization. Academic Press, London.

R. Fletcher (1969b), A class of methods for nonlinear programming with termination and convergence properties. Technical paper 368, Atomic Energy Research Establishment, Harwell.

R. Fletcher (1969c), A new approach to variable metric algorithms. Technical paper 383, Atomic Energy Research Establishment, Harwell.

R. Fletcher and M. J. D. Powell (1963), A rapidly convergent descent method for minimization. The Comp. J. **6**, 163-168.

R. Fletcher and C. M. Reeves (1964), Function minimization by conjugate gradients. The Comp. J. **7**, 149-153.

R. Fletcher and A. P. McCann (1969), Acceleration techniques for nonlinear programming, in R. Fletcher (1969a), pp. 203-214.

R. Frisch (1955), The logarithmic potential method for solving linear programming problems. Memorandum of the University Institute of Economics, Oslo.

D. Goldfarb (1969), Extension of Davidon's variable metric method to maximization under linear inequality and equality constraints. SIAM J. Appl. Math. **17**, 739-764.

A. A. Goldstein (1962), Cauchy's method of minimization. Numer. Math. **4**, 146-150.

A. A. Goldstein and B. R. Kripke (1964), Mathematical programming by minimizing differentiable functions. Numer. Math. **6**, 47-48.

A. A. Goldstein and J. F. Price (1967), An effective algorithm for minimization. Numer. Math. **10**, 184-189.

R. L. Graves and P. Wolfe (1963), Recent advances in mathematical programming. McGraw-Hill, New York.

J. Greenstadt (1967), On the relative efficiencies of gradient methods. Math. of Comp. **21**, 360-367.

G. Hadley (1964), Nonlinear and dynamic programming. Addison-Wesley, Reading, Mass.

H. Hancock (1960), Theory of maxima and minima. Dover, New York.

M. R. Hestenes and E. Stiefel (1952), Methods of conjugate gradients for solving linear systems. J. Res. Nat. B. Standards **6**, 409-436.

P. Huard (1962), Dual programs. IBM J. Res. Dev. **6**, 137-139.

P. Huard (1963), Dual programs, in R. L. Graves and P. Wolfe (1963), pp. 55-62.

P. Huard (1964), Résolution de programmes mathématiques à contraintes non-linéaires par la méthode des centres. Note E.D.F., HR 5690/3/317.

P. Huard (1967), Resolution of mathematical programming with nonlinear constraints by the method of centres, in J. Abadie (ed.), Nonlinear programming, North-Holland Publ. Comp., Amsterdam, pp. 207-219.

P. Huard (1968), Programmation mathématique convexe. R.I.R.O. **7**, 43-59.

F. John (1948), Extremum problems with inequalities as subsidiary conditions, in Studies and Essays, Courant Anniversary Volume, Interscience, New York, pp. 187-204.

K. Kleibohm (1967), Äquivalenz eines Optimierungsproblems mit Restriktionen und einer Folge von Optimierungsproblemen ohne Restriktionen. Unternehmensforschung **11**, 111-118.

J. Kowalik (1966), Nonlinear programming procedures and design optimization. Acta Polytechnica Scandinavica **13**, Trondheim.

H. W. Kuhn and A. W. Tucker (1951), Nonlinear programming, in J. Neyman (ed.), Proceedings of the second symposium on mathematical statistics and probability. University of California Press, Berkeley, pp. 481-493.

H. P. Künzi und W. Krelle (1962), Nichtlineare Programmierung. Springer, Berlin.

H. P. Künzi, H. G. Tzschach und C. A. Zehnder (1967), Numerische Methoden der mathematischen Optimierung mit ALGOL und FORTRAN Programmen. Teubner, Stuttgart.

P. J. Laurent (1963), Un théorème de convergence pour le procédé d'extrapolation de Richardson. C.R. Acad. Sci. Paris **256**, 1435-1437.

A. Lavi and T. P. Vogl (1966), Recent advances in optimization techniques. Wiley, New York.

G. Leitmann (1966), Topics in optimization. Academic Press, New York.

F. A. Lootsma (1967), Logarithmic programming: a method of solving nonlinear-programming problems. Philips Res. Repts **22**, 329-344.

F. A. Lootsma (1968a), Extrapolation in logarithmic programming. Philips Res. Repts **23**, 108-116.

F. A. Lootsma (1968b), Constrained optimization via penalty functions. Philips Res. Repts **23**, 408-423.

F. A. Lootsma (1968c), Constrained optimization via parameter-free penalty functions. Philips Res. Repts **23**, 424-437.

F. A. Lootsma (1969), Hessian matrices of penalty functions for solving constrained-optimization problems. Philips Res. Repts **24**, 322-331.

O. L. Mangasarian (1962), Duality in nonlinear programming. Quart. Appl. Math. **20**, 300-302.

O. L. Mangasarian (1965), Pseudo-convex functions. SIAM J. Control **3**, 281-290.

O. L. Mangasarian and S. Fromowitz (1967), The Fritz John necessary optimality conditions in the presence of equality and inequality constraints. J. Math. Anal. Appl. **17**, 37-47.

G. P. McCormick, W. C. Mylander and A. V. Fiacco (1965), Computer program implementing the sequential unconstrained minimization technique for nonlinear programming. Technical paper, RAC-TP-151, Research Analysis Corporation, McLean, Va., U.S.A.

E. J. Messerli and E. Polak (1969), On second order necessary conditions of optimality. SIAM J. Control **7**, 272-291.

R. J. Meyer (1969), The Philips Stirling Engine. De Ingenieur **81**, W 69-W 93.

W. Murray (1967), Ill-conditioning in barrier and penalty functions arising in constrained nonlinear programming. Sixth int. symp. on math. programming, Princeton, N.J.

W. Murray (1969), Constrained optimization. Working paper, National Physical Laboratory, Teddington.

Geraldine E. Myers (1968), Properties of the conjugate-gradient and Davidon methods. J. Opt. Th. and Appl. **2**, 209-219.

J. A. Nelder and R. Mead (1964), A simplex method for function minimization. The Comp. J. **7**, 308-313.

G. R. Parisot (1961), Résolution numérique approchée du problème de programmation linéaire par application de la programmation logarithmique. Rev. Fr. Recherche Opérationelle **20**, 227-259.

J. D. Pearson (1968), Powell's method for equality constraints, quadratic programming. Working paper RAC-S-1994, Research Analysis Corporation, McLean, Va., U.S.A.

J. D. Pearson (1969), On variable metric methods of minimization. The Comp. J. **12**, 171-178.

T. Pietrzykowski (1962), Application of the steepest descent method to concave programming, in Proceedings of the IFIPS Congress, Munich, 1962, North-Holland, Publ. Comp., Amsterdam, pp. 185-189.

T. Pietrzykowski (1969), An exact potential method for constrained maxima. SIAM J. Numer. Anal. **6**, 299-304.

E. Polak (1969), On the convergence of optimization algorithms. R.I.R.O. **3**, 17-34.

E. Polak et G. Ribière (1969), Note sur la convergence de méthodes de directions conjuguées. R.I.R.O. **3**, 35-43.

T. Pomentale (1965), A new method for solving conditioned maxima problems. J. Math. Anal. Appl. **10**, 216-220.

M. J. D. Powell (1964), An efficient method for finding the minimum of a function of several variables without calculating derivatives. The Comp. J. **7**, 155-162.

M. J. D. Powell (1967), A method for nonlinear constraints in minimization problems. Technical paper **310**, Atomic Energy Research Establishment, Harwell.

M. J. D. Powell (1969), On the convergence of the variable metric algorithm. Technical paper 382, Atomic Energy Research Establishment, Harwell.

J. Rissanen (1967), On duality without convexity. J. Math. Anal. Appl. **18**, 269-275.

J. D. Roode (1968), Generalized Lagrangian functions in mathematical programming. Thesis, Leiden, The Netherlands.

E. M. Rosen (1966), A review of quasi-Newton methods in nonlinear equation solving and unconstrained optimization. Proc. 21st National Conference of the ACM, Thompson Book Cy, Washington, pp. 37-41.

J. B. Rosen (1960), The gradient projection method for nonlinear programming, part I, linear constraints. SIAM J. **8**, 181-217.

J. B. Rosen (1961), The gradient projection method for nonlinear programming, part II, nonlinear constraints. SIAM J. **9**, 514-532.

J. B. Rosen and S. Suzuki (1965), Construction of nonlinear programming test problems. Comm. ACM **8**, 113.

H. H. Rosenbrock (1960), An automatic method for finding the greatest or least values of a function. The Comp. J. **3**, 175-184.

A. M. Sasson (1969), Nonlinear programming solutions for load-flow, minimum-loss, and economic dispatching problems. IEEE Trans. PAS-88, 399-409.

B. V. Shah, R. J. Buehler and O. Kempthorne (1964), Some algorithms for minimizing a function of several variables. SIAM J. **12**, 74-92.

H. A. Spang (1962), A review of minimization techniques for nonlinear functions. SIAM Review **4**, 343-365.

G. W. Stewart (1967), A modification of Davidon's minimization method to accept difference approximations of derivatives. Journal ACM **14**, 72-83.

R. E. Stong (1965), A note on the sequential unconstrained minimization technique for nonlinear programming. Management Sci. **12**, 142-144.

D. M. Topkis and A. F. Veinott (1967), On the convergence of some feasible direction algorithms for nonlinear programming. SIAM J. Control **5**, 268-279.

R. Tremolières (1968), La méthode des centres à troncature variable. Thèse, Paris.

Ch.-J. de la Vallée Poussin (1946), Cours d'analyse infinitésimale. Dover, New York.

G. W. Veltkamp (1969), Extrapolatie volgens het Richardson-Romberg principe. TH report 69-WSK-02, Technological University Eindhoven, The Netherlands.

D. J. Wilde and C. S. Beightler (1967), Foundations of optimization. Prentice-Hall, Englewood Cliffs, N.J.

J. H. Wilkinson (1965), The algebraic eigenvalue problem. Clarendon Press, Oxford.

P. Wolfe (1961), A duality theorem for nonlinear programming. Quart. Appl. Math. **19**, 239-244.

W. I. Zangwill (1967a), Nonlinear programming via penalty functions. Management Science **13**, 344-358.

W. I. Zangwill (1967b), The convex simplex method. Management Science **14**, 221-238.

W. I. Zangwill (1967c), Minimizing a function without calculating derivatives. The Comp. J. **10**, 293-296.

W. I. Zangwill (1968), Convergence conditions for nonlinear programming algorithms. Working paper 197, Center for Research in Management Science, University of California, Berkeley, Calif.

F. J. Zeleznik (1968), Quasi-Newton methods for nonlinear equations. Journal ACM **15**, 265-271.

G. Zoutendijk (1960), Methods of feasible directions. Elsevier, Amsterdam.

G. Zoutendijk (1966), Nonlinear programming: a numerical survey. SIAM J. Control **4**, 194-210.

Summary

This thesis is concerned with a number of methods for solving a constrained-minimization or nonlinear-programming problem. The methods under consideration have the following, common feature: they reduce the computational process for solving a constrained-minimization problem to sequential unconstrained minimization of a penalty function combining in a particular way the objective function, the constraint functions, and possibly one or more controlling parameters. Well-known examples of such methods are the logarithmic-potential method of Frisch and Parisot, the sequential unconstrained minimization technique of Carroll, Fiacco and McCormick, the exterior-point methods of Courant, Pietrzykowski and Zangwill, and Huard's method of centres.

Penalty-function techniques are designed to take into account the constraints of a minimization problem or, since almost none of the problems arising in practice have interior minima, to approach the boundary in a specifically controlled manner. The thesis starts therefore by classifying penalty functions according to their behaviour in the neighbourhood of that boundary.

A separate treatment of interior- and exterior-point methods is avoided by the study of mixed penalty-function techniques. Appropriate convexity and differentiability conditions are imposed on the problem under consideration. Furthermore, certain uniqueness conditions involving the Jacobian matrix of the Kuhn–Tucker relations are satisfied by assumption. This implies that the problem has a unique minimum $\bar{x}$ with a unique vector $\bar{u}$ of associated Lagrangian multipliers.

Under these conditions the minimizing trajectory generated by a mixed penalty-function technique can be expanded in a Taylor series about $(\bar{x},\bar{u})$. This provides, as an important numerical application, a basis for extrapolation towards $(\bar{x},\bar{u})$. The series expansion is always one in terms of the controlling parameter, independently of the behaviour of the mixed penalty function at the boundary of the constraint set.

Next, there is the intriguing question of whether some penalty functions are easier or harder to minimize than other ones. Accordingly, the condition number of the principal Hessian matrix of a penalty function is studied. It comes out that the condition number varies with the inverse of the controlling parameter, independently of the behaviour of the mixed penalty function at the boundary of the constraint set.

The parametric penalty-function techniques just named can be modified into methods which do not explicitly operate with a controlling parameter. These parameter-free versions, which are based on moving truncations of the constraint set, may be considered as penalty-function techniques adjusting the controlling parameter automatically. The crucial point is the efficiency of the adjustment. It is established how the rate of convergence depends on the vector $\bar{u}$

of Lagrangian multipliers associated with $\bar{x}$, on the boundary properties of a penalty function, on a weight factor p attached to the objective function, and on a relaxation factor ϱ. Huard's method of centres is a remarkable exception: its rate of convergence depends on the number of active constraints at $\bar{x}$, and on p and ϱ.

The computational advantages and disadvantages of the penalty-function techniques treated in the thesis are discussed in the last chapter. The parameter-free methods do not provide a significant advantage with respect to the parametric techniques which have a controlling parameter in the penalty function. Within the class of parametric techniques, there is no obvious reason for not using a so-called "first-order" method with a logarithmic barrier function, a quadratic loss function, or a mixture of these penalty functions.

An appendix wh ichpresents an ALGOL 60 procedure for constrained minimization via a mixed parametric first-order penalty function concludes the thesis.

Samenvatting

In dit proefschrift worden enkele methoden behandeld voor het oplossen van een niet-lineair programmeringsprobleem. De onderhavige methoden hebben gemeen dat zij het minimaliseren onder nevenvoorwaarden terugbrengen tot het oplossen van een reeks minimaliseringsproblemen zonder nevenvoorwaarden. Hiertoe worden de doelfunctie en de grensfuncties van het probleem en eventueel één of meer stuurparameters gecombineerd tot een te minimaliseren boetefunctie van zodanige vorm dat schending van de nevenvoorwaarden wordt verhinderd (inwendige methoden) of bestraft (uitwendige methoden). Bekende voorbeelden van dergelijke methoden zijn de logarithmische potentiaalmethode van Frisch en Parisot, de inwendige methode van Carroll, Fiacco en McCormick, de uitwendige methoden van Courant, Pietrzykowski en Zangwill, en de middelpuntsmethode van Huard.

Deze en andere, daarmee verwante methoden waarin een boetefunctie optreedt zijn ontwikkeld om de nevenvoorwaarden van een niet-lineair programmeringsprobleem te behandelen ofwel, omdat bijna geen enkel praktijkprobleem een inwendig minimum heeft, om de rand van het toegelaten gebied op een speciale manier te naderen. Dit proefschrift begint daarom met een classificatie van boetefuncties naar hun gedrag bij die rand.

Een afzonderlijke behandeling van inwendige en uitwendige methoden is overbodig. Het onderzoek richt zich op gemengde boetefuncties; de bereikte resultaten leiden onmiddellijk tot overeenkomstige resultaten voor inwendige en uitwendige methoden. Aan het niet-lineair programmeringsprobleem worden verder bepaalde convexiteits- en differentieerbaarheidsvoorwaarden opgelegd. Tenslotte is er, volgens een veronderstelling, voldaan aan éénduidigheidsvoorwaarden, die verband houden met de Jacobi-matrix van de Kuhn–Tucker relaties. Dit heeft o.a. tot gevolg dat het probleem precies één minimum $\bar{x}$ heeft met daarbij een éénduidig bepaalde vector $\bar{u}$ van Lagrangemultiplicatoren.

Onder deze veronderstellingen kan de minimaliserende weg voortgebracht door een gemengde boetefunctiemethode worden ontwikkeld in een Taylorreeks rondom $(\bar{x},\bar{u})$. Voor numerieke doeleinden is dit een belangrijk resultaat; men heeft hiermee een basis voor extrapolatie naar $(\bar{x},\bar{u})$. De Taylorreeks is steeds een reeks in termen van de stuurparameter, hoe de boetefunctie zich ook gedraagt bij de rand van het toegelaten gebied.

Dan is er de belangrijke vraag of sommige boetefuncties moeilijker of gemakkelijker te minimaliseren zijn dan andere. Daartoe werd onderzocht de matrix van tweede-orde afgeleiden van een boetefunctie, berekend in het punt waar de boetefunctie zijn minimum aanneemt; in het bijzonder werd aandacht besteed aan het conditiegetal van deze matrix. Het blijkt dat dit conditiegetal varieert met het omgekeerde van de stuurparameter, hoe de boetefunctie zich ook gedraagt bij de rand van het toegelaten gebied.

De hierboven genoemde, parametrische methoden kunnen gewijzigd worden in methoden waarin een stuurparameter niet expliciet voorkomt. Deze parametervrije versies, die gebaseerd zijn op voortschrijdende afknottingen van het toegelaten gebied, kunnen beschouwd worden als boetefunctiemethoden waarin de stuurparameter automatisch wordt bijgeregeld. Hoe efficient verloopt dit proces? Getoond wordt hoe de convergentiesnelheid afhangt van de vector $\bar{u}$ van Lagrangemultiplicatoren, van het gedrag van een boetefunctie bij de rand van het toegelaten gebied, van een gewichtsfactor p waarmee de doelfunctie wordt gewogen, en van een relaxatiefactor ϱ. De middelpuntsmethode van Huard blijkt een opmerkelijke uitzondering te zijn: de convergentiesnelheid van deze methode hangt af van het aantal actieve beperkingen in $\bar{x}$ en van p en ϱ.

De rekentechnische voor- en nadelen van de boetefunctiemethoden die in dit proefschrift worden behandeld komen ter sprake in het laatste hoofdstuk. De parametervrije methoden geven geen significante voordelen ten opzichte van de parametrische methoden met een stuurparameter in de boetefunctie. Er is geen duidelijke reden om, binnen de klasse van parametrische methoden, andere dan z.g. “eerste-orde” methoden te gebruiken, met een logarithmische barrierefunctie, een kwadratische verliesfunctie, of een mengsel van deze boetefuncties.

Een appendix met een Algol procedure voor niet-lineaire programmering via een gemengde parametrische eerste-orde boetefunctie besluit het proefschrift.

Curriculum vitae

De schrijver van dit proefschrift werd geboren op 21 januari 1936 te Midlum (Fr.). Hij behaalde het diploma gymnasium-β in 1953 en gymnasium-α in 1954. Vervolgens studeerde hij wis- en natuurkunde aan de Rijks-Universiteit te Utrecht, waar hij in februari 1961 het doctoraal examen wiskunde met de bijvakken theoretische en experimentele natuurkunde aflegde. Van maart 1961 tot december 1962 vervulde hij zijn dienstplicht bij de Koninklijke Marine; hij werd gedetacheerd bij het Physisch Laboratorium van de Rijksverdedigingsorganisatie TNO te Den Haag, in de groep systeemresearch. Sinds 1 december 1962 is hij werkzaam bij de N.V. Philips' Gloeilampenfabrieken, in de rekengroep van het Natuurkundig Laboratorium te Eindhoven.

STELLINGEN

bij het proefschrift van F. A. Lootsma

15 mei 1970
T. H. Eindhoven

I

Beschouw het programmeringsprobleem

$$\min \{ f(x) \, | g_i(x) \geqslant 0; \qquad i = 1, \ldots, m; \; x \in En \}$$

met kwadratische doelfunctie f, concave en continu differentieerbare grensfuncties $g_1, \ldots, g_m$, en compact toegelaten gebied R met niet-leeg inwendige R^o. Laat de matrix H van tweede afgeleiden van f positief definiet zijn en laat $\lambda_{\min}$ de kleinste eigenwaarde van H voorstellen. Zij $\bar{x}$ de oplossing van het probleem. Wanneer $x(r) \in R^o$ het punt is waar de logarithmische barrièrefunctie

$$f(x) - r \sum_{i=1}^{m} \ln g_i(x)$$

zijn minimumwaarde over R^o aanneemt, dan is

$$\lambda_{\min} \, [x(r) - \bar{x}]^T \, [x(r) - \bar{x}] \leqslant m \, r$$

voor elke positieve waarde van r.

II

Laat $\xi(r)$ voor $r > 0$ gedefinieerd zijn als een punt waar een barrièrefunctie van de vorm

$$f(x) + r \sum_{i=1}^{m} g_i^{-2}(x)$$

zijn minimumwaarde aanneemt over de verzameling

$$\{x \, | g_i(x) > 0; \qquad i = 1, \ldots, m\}.$$

De bewering van Fletcher en McCann, dat de vectorfunctie $\xi(r)$ onder de door hen genoemde omstandigheden in de buurt van $r = 0$ ontwikkeld kan worden in een machtreeks in termen van $r^{2/3}$, is onjuist.

R. Fletcher and A. P. McCann, Acceleration techniques for nonlinear programming, in R. Fletcher (ed.), Optimization. Academic Press, London, 1969, pp. 203-214.

III

Laat $x(r)$ een punt voorstellen waar, voor positieve r, de geregulariseerde barrièrefunctie

$$f(x) - r^{\lambda} \sum_{i=1}^{m} \varphi[g_i(x) + r]$$

zijn minimumwaarde aanneemt over de verzameling

$$S_r^o = \{x \, | g_i(x) + r > 0; \qquad i = 1, \ldots, m\},$$

en laat $u(r)$ een m vector zijn met componenten

$$u_i(r) = r^\lambda \, \varphi'\{g_i[x(r)] + r\}; \qquad i = 1, \ldots, m.$$

Onder de voorwaarden van stelling 3.4.1 van dit proefschrift is er een omgeving van $r = 0$ te vinden waar de vectorfunctie $[x(r), u(r)]$ eenduidig bepaald is en continue k-de afgeleiden bezit.

A. V. Fiacco, A general regularized sequential unconstrained minimization technique, SIAM J. Appl. Math. **17**, 1239-1245, 1969.

IV

De zwakke en de sterke dualiteitstelling van Dantzig, Eisenberg en Cottle zijn ook geldig voor het primaire probleem

$$\min_{x,y} \, [K(x,y) - y^T D_y K(x,y)]$$

onder de voorwaarden

$$D_y K(x,y) \leqslant 0, \qquad x \geqslant 0,$$

en het duale probleem

$$\max_{x,y} \, [K(x,y) - x^T D_x K(x,y)]$$

onder de voorwaarden

$$D_x K(x,y) \geqslant 0, \qquad y \geqslant 0.$$

Het is dus niet nodig om aan het primaire probleem de voorwaarde $y \geqslant 0$ en aan het duale probleem de voorwaarde $x \geqslant 0$ op te leggen.

G. B. Dantzig, E. Eisenberg and R. W. Cottle, Symmetric dual nonlinear programs. Pac. J. Math. **15**, 809-812, 1965.

F. A. Lootsma, Congruent, half-congruent and acongruent duality theorems in concave programming. Nat. Lab. report 3979, Philips Research Laboratories, Eindhoven, The Netherlands, 1965.

V

Pearson heeft zijn onderzoek van de gevolgen, die herhaald initieren van de richtingsmatrix heeft voor de algorithme van Davidon, Fletcher en Powell om een functie van n variabelen te minimaliseren, ten onrechte beperkt tot de gevolgen van initieren na elk $(n + 1)$tal iteraties.

J. D. Pearson, Variable metric methods of minimization. The Computer J. **12**, 171-178, 1969.

VI

De door Fletcher voorgestelde methode met variabele metriek, waarbij tijdens het iteratieproces een lijnminimum in de zoekrichting niet gelocaliseerd behoeft te worden, is voor het minimaliseren van boetefuncties niet aan te bevelen boven de oorspronkelijke algorithme van Davidon, Fletcher en Powell.

R. Fletcher, A new approach to variable metric algorithms. Technical paper 383, Atomic Energy Research Establishment, Harwell, 1969.

VII

In een plat vlak is een rechthoekig trefplaatje T gelegen met de ribben evenwijdig aan de X-as, resp. de Y-as van een coordinatenstelsel in dat vlak. De positie van het plaatje is niet nauwkeurig waar te nemen; het snijpunt (x, y) van de diagonalen van T is een normaal verdeelde stochastische grootheid (de waarneemfout) met verwachting (0,0) en momentenmatrix

$$\begin{pmatrix} D_x^2 & 0 \\ 0 & D_y^2 \end{pmatrix}.$$

Het plaatje wordt getroffen door een salvo van n gelijktijdig afgeschoten deeltjes. Het trefpunt (u_i, v_i) van het i-de deeltje is een normaal verdeelde stochastische grootheid met verwachting het mikpunt (ξ_i, η_i) en momentenmatrix

$$\begin{pmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{pmatrix}.$$

De trefpunten (u_i, v_i), $i = 1, \ldots, n$, en de waarneemfout (x, y) zijn onafhankelijk verdeeld. Laat $P_n(\xi_1, \eta_1, \ldots, \xi_n, \eta_n)$ de kans zijn dat tenminste één van de deeltjes in het salvo afgeschoten volgens het patroon $\{(\xi_1, \eta_1), \ldots, (\xi_n, \eta_n)\}$ het trefplaatje raakt. Wanneer σ_x groot is t.o.v. de halve lengte l_x van het plaatje en/of σ_y groot t.o.v. de halve breedte l_y, wordt het maximum van P_n over alle patronen gegeven door

$$P_n^* = 1 - (1 + s)\,\mathrm{e}^{-s} + O\left(\left(\frac{l_x\, l_y}{\sigma_x\, \sigma_y}\right)^2\right).$$

met

$$s = 2\left(\frac{l_x\, l_y\, n}{D_x\, D_y\, \pi}\right)^{1/2}.$$

VIII

Het verdient aanbeveling om het snijprobleem in een golfkartonfabriek, waar men rechthoekige platen snijdt uit een voortlopende baan golfkarton, te formuleren als een z.g. overdekkingsprobleem met gelijkheidsbeperkingen. In de formulering treden de complete of gesloten snijpatronen, en eventueel andere patronen met bijzondere eigenschappen, op als activiteiten die al dan niet uitgevoerd moeten worden. Dit kan leiden tot een rekentechnisch aanvaardbare probleemstelling waarin ook de vaste kosten van een omstelling der messen, de toleranties op de lengte en breedte van de gevraagde platen golfkarton, en de gewenste verdeling van het orderpakket over de snij-inrichtingen zijn verdisconteerd.

F. A. Lootsma, An algorithm for the cutting-stock problem in the corrugated-cardboard factory. Nat. Lab. Technical Note 43/66, Philips Research Laboratories, Eindhoven, Netherlands.

R. S. Garfinkel and G. L. Nemhauser, The set-partitioning problem: set covering with equality constraints. Operations Research **17**, 848-856, 1969.

IX

Als model voor een activiteitsduur in planningstechnieken is een gammaverdeling te verkiezen boven een betaverdeling.

F. A. Lootsma, A gamma distribution as a model of an activity duration. Méthodes à chemin critique. Actes du Congrès Internet I, Vienne, 1967. Dunod, Paris, 1969.

X

Het voorstel van de commissie-Braun tot het instellen van een centraal orgaan post-academisch onderwijs, met als taak het stimuleren, coordineren en financieren van post-academisch onderwijs aan de universiteiten en hogescholen, is door de Academische Raad ten onrechte verworpen.

Rapport post-academisch onderwijs (uitgebracht door de commissie-Braun van het verbond van Wetenschappelijke Onderzoekers). Wetenschap en Samenleving, supplement op de 18de jaargang, april/mei 1964.

Brief van de Academische Raad aan de Minister van Onderwijs en Wetenschappen, nr. AR - 1627, 31 december 1966, met bijlagen.

XI

Terugziende op de periode waarin hij lid was van het Air Defence Research Committee (1935-1939) schrijft Churchill: "*It is often possible in England for experienced politicians to reconcile functions of this kind* (felle kritiek op, en waar mogelijk loyale medewerking aan het regeringsbeleid, L.) *in the same way as the sharpest political differences are sometimes found not incompatible with personal friendship. Scientists are however a far more jealous society*". Een dergelijke bewering waarmee de loyaliteit en onderlinge communicatie van de beoefenaars der exacte wetenschappen ongemotiveerd in twijfel getrokken worden en die ook in onze tijd de verstandhouding tussen politici en wetenschapsmensen kan vertroebelen, is verwerpelijk.

W. S. Churchill, The second world war. Vol. I, Cassell & Co., London, 1948, p. 120.