INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

Bell & Howell Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

Variational inequalities with the analytic center cutting plane method

Michel Denault
Faculty of Management
McGill University, Montréal

February 1998

A thesis submitted to the
Faculty of Graduate Studies and Research
in partial fulfilment of the requirements of
the degree of Doctor of Philosophy

Abstract

This thesis concerns the solution of variational inequalities (VIs) with analytic center cutting plane methods (ACCPMs). A convex feasibility problem reformulation of the variational inequality is used; this reformulation applies to VIs defined with pseudo-monotone, single-valued mappings or with maximal monotone, multi-valued mappings.

Two cutting plane methods are presented: the first is based on linear cuts while the second uses quadratic cuts. The first method, ACCPM-VI (linear cuts), requires mapping evaluations but no Jacobian evaluations; in fact, no differentiability assumption is needed. The cuts are placed at approximate analytic centers that are tracked with infeasible primal-dual Newton steps. Linear equality constraints may be present in the definition of the VI's set of reference, and are treated explicitly. The set of reference is assumed to be polyhedral, or is convex and iteratively approximated by polyhedra. Alongside of the sequence of analytic centers, another sequence of points is generated, based on convex combinations of the analytic centers. This latter sequence is observed to converge to a solution much faster than the former sequence.

The second method, ACCPM-VI (quadratic cuts), has cuts based on both mapping evaluations and Jacobian evaluations. The use of such a richer information set allows cuts that guide more accurately the sequence of analytic centers towards a solution. Mappings are assumed to be strongly monotone.

However, Jacobian approximations, relying only on mapping evaluations, are observed to work very well in practice, so that differentiability of the mappings may not be required. There are two versions of the ACCPM-VI (quadratic cuts), that differ in the way a new analytic center is reached after the introduction of a cut. One version uses a curvilinear search followed by dual Newton centering steps. The search entails a full eigenvector-eigenvalue decomposition of a dense matrix of the order of the number of variables. The other version uses two line searches, primal-dual Newton steps, but no eigenvector-eigenvalue decomposition.

The algorithms described in this thesis were implemented in the MATLAB environment. Numerical tests were performed on a variety of problems, some new and some traditional applications of variational inequalities.

Résumé

Le sujet de cette thèse est la résolution d'inégalités variationnelles (IVs) à l'aide de méthodes de centres analytiques et plans coupants (ACCPM). L'inégalité variationnelle est transformée en un problème de réalisabilité convexe; cette transformation est valide pour les IVs définies avec des applications soit univoques et pseudo-monotones, ou multivoques et maximales monotones.

Deux méthodes de plans coupants sont présentées, la première fondée sur des coupes linéaires, et la seconde, sur des coupes quadratiques. La méthode à coupes linéaires, appelée ACCPM-VI (coupes linéaires), exige des évaluations de l'application, mais aucunes du jacobien; de fait, aucune hypothèse de différentiabilité n'est requise. Les coupes sont placées sur des centres analytiques approximatifs, que l'on trouve grâce à une méthode de Newton primale-duale non-réalisable. Des égalités linéaires peuvent être présentes dans la définition de l'ensemble de référence de l'IV, et sont traitées de façon explicite. On fait l'hypothèse que cet ensemble de référence est polyhédral, ou alors il est simplement convexe et une approximation polyhédrale est bâtie par l'algorithme à l'aide d'un oracle. Au fur et à mesure que la suite de centres analytiques est fabriquée, une autre suite de points est produite en parallèle, consistant de combinaisons convexes de centres analytiques. Cette deuxième suite démontre des propriétés de convergence beaucoup plus rapide que la première.

La deuxième méthode de plans coupants est appelée ACCPM-VI (coupes quadratiques), et ses coupes sont fondées à la fois sur des évaluations de l'application et de son jacobien. L'utilisation de cette information plus détaillée permet la production de coupes qui orientent plus précisément la suite de centres analytiques vers une solution. L'hypothèse est faite que les applications sont fortement monotones. Cependant, il n'a pas encore été déterminé si l'hypothèse de différentiabilité est nécessaire, car en pratique, des approximations du jacobien qui n'utilisent que les évaluations de l'application fonctionnent très adéquatement. Nous présentons deux versions de ACCPM-VI (coupes quadratiques), qui diffèrent par leur façon de retrouver un centre analytique après l'introduction d'une coupe. Une version utilise une recherche sur une courbe, suivie de pas de centrage duaux de Newton. Cette recherche exige une décomposition complète en valeurs propres et vecteurs propres d'une matrice dense de l'ordre du nombre de variables. L'autre version remplace la recherche sur une courbe par deux recherches linéaires, éliminant le besoin de la décomposition; aussi, les pas de centrage sont primaux-duaux plutôt que duaux.

Remerciements

S'il est une personne sans qui je n'aurais pu faire de cette thèse ce qu'elle est, c'est bien mon directeur, Jean-Louis Goffin, que je remercie de tout coeur. Son expérience, ses idées, ses conseils mais aussi la liberté qu'il m'accordait, ont été primordiaux pour l'élaboration de ce travail. Je le remercie aussi pour l'aide financière qu'il m'a offerte au long de mes études.

Plusieurs autres personnes m'ont aidé au cours des dernières années, mais parmi elles, Patrice Marcotte mérite une reconnaissance particulière. Non seulement a-t-il lu et commenté la plus grande partie de ma thèse, mais il m'a aussi et surtout dévoué de nombreuses et longues conversations sur les subtilités de la théorie des inégalités variationnelles. Les professeurs Richard Loulou et Gilbert Laporte, membres de mon comité de doctorat, m'ont aussi fait bénéficier de leurs conseils et de leur expérience, et je les en remercie. Je veux témoigner ma gratitude au professeur Jean-Philippe Vial qui lors de ses courts séjours à Montréal a toujours trouvé le temps de m'écouter et de me prodiguer ses conseils; la clarté de sa vision de l'optimisation demeure pour moi une source d'inspiration. Son collaborateur, Yuri Nesterov, aura aussi, quoiqu'indirectement, contribué au succès de mon travail, et je lui en suis reconnaissant.

Trois chercheurs me sont venus en aide sur des sujets plus particuliers, mais je ne leur en suis pas moins reconnaissant. Le professeur Hans-Jakob Lüthi et le docteur Benno Büeler, de l'ETH à Zurich, ont gracieusement mis

à ma disposition leur modèle d'équilibre économique et énergétique, de même leur ordinateur parallèle. Je dois souligner les efforts que le Dr. Büeler a dû déployer pour faire comprendre à un néophyte d'outre-atlantique les bases du modèle MM^{mr}, et ce, alors que lui-même finissait sa thèse de doctorat. Le professeur Jérôme Detemple m'a quant à lui introduit à l'évaluation d'options par méthodes variationnelles, et est donc à l'origine des problèmes d'options des chapitres 3 et 4.

Je tiens à remercier certaines personnes qui n'ont ni lu ma thèse ni apporté de conseils d'ordre scientifique, mais qui auront rendu ces années combien plus agréables et qui, par le fait même, ont contribué au succès de l'entreprise. Il s'agit premièrement de mes collègues de bureau et amis Goran et Mirela Stojkovic, au centre de recherche GERAD, et Stelios Zyglidopoulos, de l'université McGill; ils ont partagé mes joies et mes frustrations, et je leur en sais gré. Il s'agit aussi de ceux grâce à qui, en grande partie, le GERAD est un endroit aussi plaisant et propice à la recherche: les dames du soutien administratif Anita Beauchamp, Francine Benoît, Carole Dufour, Nicole Paradis et Christine Picard, et les messieurs du soutien informatique Guy Bordeleau, Daniel Charbonneau, et Pierre Girard. Ces gens apportent un dévouement à leur travail qu'on rencontre trop rarement.

Je veux témoigner toute ma reconnaissance à mes parents, Pierre et Thérèse Denault. Je leur dois le goût d'apprendre et celui de bien faire les choses, peut-être les qualités les plus importantes pour un chercheur. Finalement, c'est beaucoup grâce à eux que je suis qui je suis.

Le dernier mais non le moindre de mes mercis revient à la personne la plus importante dans ma vie, mon épouse Joelle. Elle comprend le chercheur en moi comme seule une chercheuse peut le faire, mais surtout, elle comprend l'homme en moi comme seule une âme soeur peut le faire. Elle m'a soutenu, depuis le début, de ses conseils, de sa confiance, de son amour.

Contents

List of Figures

List of Tables

Introduction

This thesis is about solving variational inequalities. This is an old problem, which was for a long time considered mainly in infinite-dimensional spaces. However, in the last years, a large amount of interest has been devoted to *finite dimensional* variational inequality problems (VI), nonlinear complementarity problems (NCP) and mixed complementarity problems (MCP) (i.e. box-constrained variational inequalities). The application domain of these problems is wide, ranging from the traditional equilibrium models for road traffic and economic markets, to game theory, to environmental, stochastic models, and to the solution of differential equations with applications in finance and optimal control.

The algorithms for variational inequalities that constitute the core of this thesis are based on two complementary concepts: the cutting plane and the analytic center. The cutting plane is used to separate a set in two: one part that is of interest, and one that is not. For our purposes, the part of interest is known to contain the solution(s) of the variational inequality, and the cut is used to remove a part which does not contain the solution(s). After several cuts, the part of interest becomes smaller and smaller, to the point of containing just about only the solution(s). The cutting plane is defined with respect to a specific point in the set, and this is where the second concept, analytic center, comes into play. Intuitively, without knowing in advance on which side of the cutting plane the part of interest will be, we should strive

to cut the set roughly in two equal parts, so that roughly half of it is deemed without interest; this to avoid useless cuts that only see a tiny portion of the set as without interest. It is precisely the role of the analytic center to guide the placing of the cut. Among other available definitions of centers, analytic centers are particularly attractive because their computation, with techniques derived from interior point methods, is relatively cheap.

The thesis comprises four main chapters. The first one discusses the Analytic Center Cutting Plane Method (ACCPM), as the combination of cutting planes and analytic centers has come to be known. The second chapter is a review of the field of variational inequalities. The last two chapters present the algorithms to which we refer, as a whole, as ACCPM-VI. The third chapter covers a linear cut method, which includes the possibility of treating variational inequalities with linear equality constraints. The fourth chapter refines the linear cut approach to improve its speed of convergence; this is achieved by using quadratic cuts. In both Chapters 3 and 4, we present the results of several numerical experiments with the algorithms. An appendix treats in more detail one application from the previous chapters, the pricing of options in finance.

NOTATION: We use the convention, common in interior point methods, that for a variable x, the capitalized X is a matrix with the values of x on its diagonal and zeros elsewhere. Also, e is the vector $[1, 1, \ldots, 1]^t$ of appropriate dimension, so that for example $Xe = x$.

The notation $2^{\mathbb{R}^m}$ denotes the set of all subsets of $\mathbb{R}^m$, while $\mathbb{R}^m_+$ is the positive orthant in $\mathbb{R}^m$. The Loewner ordering of matrices $A \succeq B$ means that $A - B$ is a positive semidefinite matrix.

Chapter 1

The Analytic Center Cutting Plane Method

1.1 Introduction

The analytic center cutting plane method, or ACCPM, was developped as a nondifferentiable optimization algorithm by J.-L. Goffin and J.-Ph. Vial [31]. It is an amalgamation of concepts from different branches of mathematical programming: intrinsically a convex programming tool, it is chiefly geared at nondifferentiable optimization, but draws heavily from the recent advances in interior-point linear programming; and one of its best known applications is the solution of very large linear or nonlinear programs through decomposition techniques.

This chapter reviews the theory necessary to understand the ACCPM. The first section, on nondifferentiable optimization, builds the backbone of ACCPM: the cutting plane approach to convex and nondifferentiable optimization. Several other nondifferentiable optimization methods are introduced, as reference points.

The second section discusses some concepts of interior point methods, and uncovers the heart of ACCPM: the analytic centers. The technicalities

of interior point methods are hardly touched upon; we rather concentrate on the aspects that are more important for the remainder of the thesis.

The third section explains the analytic center cutting plane method *per se*. Cutting-planes and analytic centers meld into an efficient, reliable non-differentiable optimization algorithm. We also introduce in the third section a variation of the current implementation, which further involves the interior point methods.

1.2 Nondifferentiable Optimization

1.2.1 Introduction

The purpose of discussing nondifferentiable optimization is two-fold: first, many concepts from this field are crucial to the analytic center cutting plane method, a focal point of this thesis. Second, the ACCPM is in itself a method for nondifferentiable optimization, so that fundamentals of nondifferentiable optimization are needed to situate the ACCPM amongst other methods of nonsmooth optimization.

Nondifferentiable optimization is concerned with the minimization of a real function f:

$$\text{minimize } f(x),\ x \in \mathbb{R}^n \tag{1.1}$$

where f may **not** have derivatives. Although the unconstrained problem above is the norm, differentiable and nondifferentiable constraints can be applied. For our purposes, we require all functions to be convex.

In this setting, the **directional derivative** of f at x in direction d

$$f'(x;d) := \lim_{t \downarrow 0} \frac{f(x+td) - f(x)}{t}$$

exists for all x and all d.

It is usually the case that f is *piecewise-C^1*, i.e. that $\mathbb{R}^n$ can be divided in parts over each of which the gradient $\nabla f = [\frac{\partial f}{\partial x_1}, \ldots, \frac{\partial f}{\partial x_n}]^t$ exists and is continuous. At points where the gradient does not exist, the **subdifferential** $\partial f(x)$ of f at x can be used in its stead:

$$\partial f(x) := \{\gamma \in \mathbb{R}^n \mid f(x) + \gamma(y - x) \leq f(y), \forall y \in \mathbb{R}^n\}$$

Elements of the convex set $\partial f(x)$ are called **subgradients**; at differentiable points, it is clear that ∂f reduces to a unique element, the gradient ∇f.

One can derive necessary and sufficient first order conditions for problem (1.1):

$$f(x^*) \leq f(x)\ \forall x \in \mathbb{R}^n \Leftrightarrow \mathbf{0} \in \partial f(x^*)$$

Finally, we make the assumption, common in nondifferentiable optimization, that a **black box** or **oracle** can compute $f(x)$ and one arbitrary subgradient $\gamma \in \partial f(x)$, for any $x \in \mathbb{R}^n$. The oracle could be an explicit formula, or a mathematical program itself.

1.2.2 Examples of nondifferentiable problems

Nondifferentiable optimization problems can arise through the mathematical modeling of a situation which has inherent nondifferentiability. For example, a tax-and-surtax system could be modelled by a function T of income level x:

$$T(x) = \begin{cases} r_0 x & \text{when } 0 \leq x \leq a \\ r_0 a + r_1(x - a) & \text{when } x > a \end{cases}$$

where r_0 is the nominal tax rate, $r_1 (> r_0)$ the tax-and-surtax rate, and a the level of income over which surtax is levied. Clearly $T(x)$ is continuous and is nondifferentiable at $x = a$.

This example is one case of a general scheme: the minimization of a function of the form

$$f(x) = \max\{f_i(x) \mid i = 1, 2, ..., m\}$$

where the functions $f_1(x), ..., f_m(x)$ are smooth. A typical occurrence of that scheme is the minimization of the l_1- or l_∞-norm of a vector-valued function:

$$\min_x \|f_1(x), ..., f_m(x)\|_1$$

and

$$\min_x \|f_1(x), ..., f_m(x)\|_\infty$$

are respectively equivalent to the minmax problems:

$$\min_x \max_{+-}(\pm f_1(x) \pm f_2(x) ... \pm f_m(x))$$

and

$$\min_x \max\{-f_1(x), f_1(x), -f_2(x), ..., f_m(x)\}$$

Lagrangian relaxation is one of the most abundant source of nondifferentiable problems. This technique is based on Lagrangian duality and is used on constrained problems:

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g(x) \leq 0 \\ & x \in X \end{array} \tag{1.2}$$

In a nutshell, "complicating" constraints are eliminated through a relaxation of the problem; ultimately, the price to pay is the loss of differentiability. In (1.2) assume that $g(x) \leq 0$ are the complicating constraints, and that $x \in X$ is much easier to handle. Then

$$\begin{array}{ll} \text{minimize} & f(x) + \lambda^t g(x) \\ \text{subject to} & x \in X \end{array} \tag{1.3}$$

is a relaxation of (1.2). Denoting by $\phi(\lambda)$ the optimal value of (1.3) ($\phi(\lambda)$ is a function of λ), the dual problem

$$\text{maximize } \phi(\lambda), \ \lambda \in \mathbb{R}^n_+$$

is a convex problem, nondifferentiable, which often leads to an optimal solution of (1.2) more easily than by direct attack.

This technique is very closely related to **decomposition schemes**. Almost every large-scale mathematical program possesses a structure which allows to "split" it into many smaller programs, if it were not for some complicating constraints. One can get rid of the latter by Lagrangian relaxation, and the evaluation of the nondifferentiable function $\phi(\lambda)$ means solving

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^{m} (f_i(x_i) + \lambda^t g_i(x_i)) \\ \text{subject to} & x_i \in X_i \ (i = 1, 2, \ldots, m) \end{array}$$

which is just m smaller problems of the form

$$\begin{array}{ll} \text{minimize} & f_i(x_i) + \lambda^t g_i(x_i) \\ \text{subject to} & x_i \in X_i \end{array}$$

1.2.3 First methods in nondifferentiable optimization

Applying "smooth" methods (steepest descent, Newton method etc.) to nondifferentiable problems often has disastrous consequences. For example, it is easy to build problems where a smooth method converges to a nonoptimal point; see section 3.1 of [58]. There are then two ways of tackling a nondifferentiable optimization problem: by using a method which can handle nondifferentiability (direct methods), or by *smoothing* the problem so that ordinary smooth methods can be applied to it (indirect methods). With the current state of knowledge, the latter approach is the least attractive, if nothing else because only certain types of problems can be smoothed. Min-max, l_1-norm and l_∞- norm problems can be smoothed; however, numerical instabilities are frequent in practice. We will not discuss indirect methods any further in this work.

The **subgradient method** is one of the first and simplest direct methods of nondifferentiable optimization. The direction of the subgradient is used

to find the point x_{k+1}, with the help of a steplength t_k:

$$x_{k+1} := x_k - t_k \frac{\gamma_k}{\|\gamma_k\|}$$

Remember that by our "oracle assumption", a subgradient γ_k is available at any point $x_k \in \mathbb{R}^n$. A major drawback of γ_k is that it may well not be a descent direction from x_k. However, it can be shown that with t_k chosen off-line and such that $\lim_{k\to\infty} t_k = 0$ and $\sum_{k=0}^{\infty} t_k = \infty$, x_k converges to the optimal set X^*. But the great simplicity of this convergent method bears a high price: the speed of convergence is usually poor, and there exists no good stopping criterion. Much research has gone into the steplength adjustments and acceleration of the subgradient method; unfortunately it remains quite mediocre in practice. See [58] and the recent book [46] and the references therein for more information.

The **ellipsoid method**, developed by Nemirovskii and Yudin [78] on the basis of work by Levin [61] and Shor [97], is another nondifferentiable optimization method. An improving approximation of the optimal set X^* is obtained with a sequence of ellipsoids containing X^*. Once again, the performance, in practice, is mediocre.

Finally, the cutting plane method, discovered independently by Kelley [51] and Cheney and Goldstein [12], is also an early method for convex (not necessarily differentiable) minimization. It bears particular importance to us, so that the next section is devoted to it.

1.2.4 Cutting Plane Method

Cutting plane methods were originally devised for general, convex programs:

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_i(x) \leq 0,\ i = 1, 2, ..., m \\ & f, g_i \text{ convex functions} \end{array}$$

However, these methods have seen their main application in a more specific area, the optimization of convex, nondifferentiable programs whose functions are linear by parts; such programs occur naturally when dual or decomposition methods are used.

Fundamentally, cutting plane methods rely on the following result for convex functions (see theorem XI 1.3.8 in [46]). For $f : \mathbb{R}^n \rightarrow \mathbb{R}$ closed and convex,

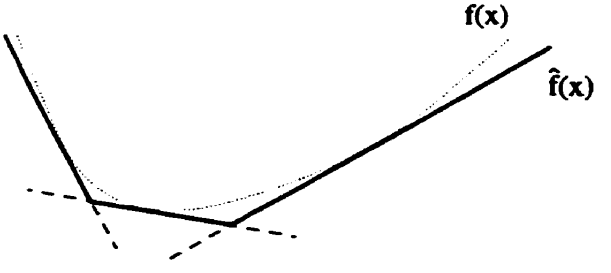
$$f(x) = \max_{y \in \mathbb{R}^n} \left\{ f(y) + \gamma^t (x - y),\ \forall x \in \mathbb{R}^n \right\}$$

where $\gamma \in \partial f(y)$.

The elements required in the right-hand side term are the values of the function f and a subgradient γ, for each $y \in \mathbb{R}^n$. Remember the assumption of (1.2.1) that an oracle exists, which provides these two elements for any $y \in \mathbb{R}^n$ supplied to it. While the right-hand side term in itself is most unwieldy, it opens the door for an approximation $\hat{f}$ of f:

$$\hat{f}(x) = \max_{i=1,\ldots,k} \{ f(x_i) + \gamma_i (x - x_i) \} \tag{1.4}$$

where $\{x_1, x_2, \ldots, x_k\}$ is a finite set of points in $\mathbb{R}^n$ and $\gamma_i \in \partial f(x_i)$. This function is sometimes called a linear tangential (outer) approximation of the convex function f. It is clear that $\hat{f}(x) \leq f(x)$



Consider the unconstrained, convex program:

$$\min_{x \in \mathbb{R}^n} f(x) \tag{1.5}$$

where f is convex. Replacing $f(x)$ by $\hat{f}(x)$, we obtain a relaxation $\min_{x\in\mathbb{R}^n} \hat{f}(x)$ of problem (1.5). An equivalent formulation is

$$\begin{array}{ll} \min_{v,x} & v \\ \text{subject to} & v \geq f(x_1) + \gamma_1^t(x - x_1) \\ & \vdots \\ & v \geq f(x_k) + \gamma_k^t(x - x_k) \end{array} \tag{1.6}$$

The original cutting plane method, usually refered to as Kelley's cutting plane method, solves (1.6) to obtain (v_{k+1}, x_{k+1}); the oracle yields $f(x_{k+1})$ and $\gamma_{k+1} \in \partial f(x_{k+1})$. If $v_{k+1} = f(x_{k+1})$, the solution of (1.5) has been found. Otherwise,

$$v \geq f(x_{k+1}) + \gamma_{k+1}^t(x - x_{k+1})$$

is a valid cut to add to the constraints of (1.6). This is called an **optimality cut**.

The algorithm thus described is convergent (see theorem XII 4.2.3 in [46]); when f consists of a finite number of linear pieces, the algorithm converges in finite time. Note that it is necessary to bound x artificially at the outset of the algorithm, until enough constraints have been generated.

Consider now the *constrained* program with convex constraint $g(x)$: $\mathbb{R}^n \to \mathbb{R}$:

$$\begin{array}{ll} \min_{x\in\mathbb{R}^n} & f(x) \\ \text{subject to} & g(x) \leq 0 \end{array} \tag{1.7}$$

By using approximations $\hat{f}, \hat{g}$ of the type (1.4), we have the relaxed program

$$\begin{array}{ll} \min & \hat{f}(x) \\ \text{subject to} & \hat{g}(x) \leq 0 \end{array} \tag{1.8}$$

and its equivalent form

$$\begin{array}{ll} \min_{v,x} & v \\ \text{subject to} & f(x_1) + \gamma_1^f (x - x_1) \leq v \\ & \qquad \vdots \\ & f(x_k) + \gamma_k^f (x - x_k) \leq v \\ & g(x_1) + \gamma_1^g (x - x_1) \leq 0 \\ & \qquad \vdots \\ & g(x_k) + \gamma_k^g (x - x_k) \leq 0 \end{array} \tag{1.9}$$

In the classical cutting-plane method, the relaxation (1.9) is solved, producing solution (v_{k+1}, x_{k+1}). There are two functions involved here, hence two oracles. The oracle for f yields $f(x_{k+1})$ and $\gamma_{k+1}^f \in \partial f(x_{k+1})$. As before, if $f(x_{k+1}) = v_{k+1}$ and x_{k+1} is feasible for (1.7) then x_{k+1} is a solution of (1.7); otherwise, an optimality cut $f(x_{k+1}) + \gamma_{k+1}^f (x - x_{k+1}) \leq v$ is added to (1.9).

The oracle for g yields $g(x_{k+1})$ and $\gamma_{k+1}^g \in \partial g(x_{k+1})$. If $g(x_{k+1}) > 0$, then the i^{th} constraint of (1.7) is violated at x_{k+1}, and

$$g(x_{k+1}) + \gamma_{k+1}^g (x - x_{k+1}) \leq 0$$

is a valid constraint to introduce in (1.9). Such cuts are called **feasibility cuts**. Again, convergence to the optimum of (1.7) can be proved. Note that the method above applies when several constraints $g_1, \ldots, g_m$ are present, at the cost of a more cumbersome notation: it is then possible to either treat each function g_i separately, with its own approximation, or to lump all constraints into the single $g(x) = \max_{i=1,\ldots,m} g_i(x)$.

The classical cutting plane method described in this section iterates to a new point $x_{k+1} \in \mathbb{R}^n$ by *solving the current relaxation*; but there are other ways of selecting x_{k+1}. The *center methods*, which will be a centerpiece of chapter 1.4, are the best examples of these alternative ways. Their study is deferred until then.

Finally, let us recall that in large scale linear programming, the Dantzig–Wolfe decomposition technique is the dual equivalent of the classical cutting

plane method above.

1.3 Interior Point Methods (IPM)

Since the projective algorithm of Karmarkar [50] for linear optimization problems, interior point methods (IPM) have been the source of renewed and sustained interest for the mathematical programming community. We say renewed because the use of logarithmic barriers to work in the interior of a feasible area is not a new concept; consider [27] and [25]. The literature on the subject is immense, but [40] provides a good review.

In linear programming, interior point methods have come in the last decade to compete with the simplex approach, especially on very large, sparse problems. Interior points are also being used in increasingly diverse contexts, such as nonlinear programming, semi-definite programming and complementarity problems. What draws our interest to IPMs is the Analytic center, readily computed by interior points techniques and fundamental to the analytic center cutting plane method.

1.3.1 Interior points and path-following

We shall discuss here only a small part of the huge body of theory and numerous algorithms within interior points methods. More specifically, we are interested in path-following IPMs for linear programming. The underlying philosophy of path following is that iterations follow, more or less closely, the **central path** associated with every linear program. There are different ways of doing this, some of which appear hereafter; all describe the same central paths, but the parameterizations are different. In all cases we use the

primal-dual pair of linear programs:

$$\begin{array}{ll} \text{minimize} & c^t x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array} \tag{1.10}$$

and

$$\begin{array}{ll} \text{maximize} & b^t y \\ \text{subject to} & A^t y \leq c \end{array} \tag{1.11}$$

Methods using the Log barrier The Primal Log Barrier method is quite intuitive. The problem (1.10) is transformed to

$$\begin{array}{ll} \text{minimize}_{x>0} & c^t x - \mu \sum_j \ln x_j \\ \text{subject to} & Ax = b \end{array}$$

where the logarithmic term acts as a barrier to keep x away from the boundary $x = 0$; μ is a positive parameter. The Lagrangian for that problem is

$$L(x, y; \mu) = c^t x - \mu \sum_j \ln x_j - y^t (Ax - b)$$

and the first order optimality conditions are

$$\begin{array}{rcccl} \nabla_x L & = & c - \mu X^{-1} e - A^t y & = & 0 \\ \nabla_y L & = & -Ax + b & = & 0 \\ & & x & > & 0 \end{array} \tag{1.12}$$

One can similarly work on the dual linear program (1.11), forming the dual log-barrier problem

$$\begin{array}{ll} \text{maximize}_{z>0,y} & b^t y + \mu \sum_j \ln z_j \\ \text{subject to} & A^t y + z = c \end{array}$$

The first order conditions are

$$\begin{array}{rcl} Ax & = & b \\ A^t y + z & = & c \\ XZe & = & \mu e \end{array} \tag{1.13}$$

with $x > 0, z > 0$, which are the usual linear programming optimality conditions, except the complementary slackness where μe has replaced the usual 0. It is important to note that the duality gap is known explicitly in this case:

$$c^t x - b^t y = x^t z = \mu n$$

If we make the assumptions that the feasible areas of both (1.10) and (1.11) have non-empty relative interiors, and that A has full rank m, it follows easily that the systems (1.12) and (1.13) fully characterize the unique solution of their respective problems. Furthermore the systems are equivalent in the sense that the solution of one system is also the solution of its corresponding part in the other systems.

The **central paths** are precisely those points that are solutions of the first order systems, *for all values of* $\mu > 0$. To make explicit the relation to μ, call those solutions $\{x(\mu), y(\mu), z(\mu)\}$. The set $\{x(\mu), \mu > 0\}$ forms the primal central path, while $\{y(\mu), z(\mu), \mu > 0\}$ is the dual central path. It is well established that

$$\lim_{\mu \to 0} (x(\mu), y(\mu), z(\mu)) = (x, y, z)$$

is indeed a solution to the original problems (1.10) and (1.11). This confirms the intuition —at the heart of path-following methods— that solving

$$\begin{aligned} Ax &= b \\ A^t y + z &= c \\ XZe &= \mu e \\ x, z &> 0 \end{aligned}$$

for small values of μ, is a valid approach to solve

$$\begin{aligned} Ax &= b \\ A^t y + z &= c \\ XZe &= 0 \\ x, z &> 0 \end{aligned}$$

Let us outline how log barrier methods proceed from the KKT conditions onward. For any of the systems (1.12) and (1.13) with a parameter μ_0, a

Newton step is taken, with an appropriate steplength. In the case of (1.12), the step takes place in the primal space only; with (1.13), the step is in the dual space. Simultaneous steps in the primal and dual spaces can also be taken in (1.13), justifying the name primal-dual method. A new value $\mu_+, 0 < \mu_+ < \mu_0$ is then chosen and another Newton step taken for the new systems using μ_+.

The advantage of choosing μ_+ sufficiently close to μ_0, is that the current point $(x(\mu_0), y(\mu_0), z(\mu_0))$ then remains within the quadratic convergence area of the Newton method for the system with μ_+. This is the philosophy of the "short-step path-following method"; it is also in this manner that the polynomiality of path-following algorithms is proved. On the other hand, "long-step" methods which do not pay the same attention to the quadratic convergence area are known to work faster in practice.

Using the potential function Another interior points method in the path-following philosophy directly attempts to decrease the potential function associated with (1.10) and a lower bound β:

$$\varphi_P(x;\beta) = (n+1)\ln(c^t x - \beta) - \sum_j \ln x_j$$

the factor $n+1$ will find justification below. References are for example [16] and [103].

The problem

$$\begin{array}{ll} \text{minimize} & \varphi_P(x;\beta) \\ \text{subject to} & Ax = b \\ & x > 0 \end{array} \tag{1.14}$$

then has 1st order conditions

$$Ax = b \tag{1.15}$$

$$Zx = \frac{c^t x - \beta}{n+1} e \tag{1.16}$$

$$A^t y + z = c \tag{1.17}$$

with $x > 0, z > 0$. Clearly, as β is updated to better and better lower bounds, this system of equations approaches the optimality conditions of the original LP. Also, this system defines for all possible lower bounds β the central paths, although the parameterization is obviously different from (1.13).

As would be expected, a potential function can also be devised for the dual problem (1.11) (see [89])

$$\varphi_D(y, z; \beta) = -\ln(b^t y - \beta) - \sum_{i=1}^{n} \ln z_i$$

where the z variables are the slacks in the dual constraints. The potential reduction problem is then

$$\begin{array}{ll} \text{minimize} & \varphi_D(y, z; \beta) \\ \text{subject to} & A^t y + z = c \\ & z > 0 \end{array} \tag{1.18}$$

with 1st order conditions

$$Ax = b \tag{1.19}$$

$$Zx = (b^t y - \beta)e \tag{1.20}$$

$$A^t y + z = c \tag{1.21}$$

again with $x > 0, z > 0$. When the coefficient $(n+1)$ is used in the primal potential, these conditions are if fact equivalent to (1.15) (see [31]): from (1.20), we have

$$e^t Z x = z^t x = n(by - \beta)$$

and from (1.19) and (1.21) we find

$$z^t x = c^t x - b^t y$$

so that

$$c^t x - b^t y = n(by - \beta).$$

This equation can be rearranged into

$$\frac{c^t x - \beta}{n+1} = b^t y - \beta$$

showing that (1.15)–(1.17) follows from (1.19)–(1.21); the reverse relation clearly holds as well.

Therefore either set of equations trace the central paths for the possible values of the lower bound β. As in the previous section, primal, dual and primal-dual approaches are possible: it is again mostly a matter of whether steps are taken in the primal space, the dual space, or both.

One very important property is common to all central points, independently of their parameterization: feasible points in both the primal and dual spaces are always available. In opposition, the simplex method yield a dual feasible point only once optimality of (1.10) and (1.11) has been reached.

1.3.2 Analytic centers

Analytic centers are in essence defined as centers of polyhedrons. They were formally introduced by Sonnevend [99]. If the polyhedron $P = \{y \in \mathbb{R}^m : \alpha_k y \leq \beta_k, k = 1, \ldots, l\}$ has a nonempty interior, then its analytic center is

$$\bar{y} = \arg \max_{y \in \text{int}(P)} \log \prod_{k=1}^{l} (\beta_k - \alpha_k y).$$

With a short proof (see [89]), one can verify that if P is bounded, $\log \prod_{k=1}^{l} (\beta_k - \alpha_k y)$ is strictly concave on $y \in \text{int}(P)$ and $\bar{y}$ is consequently a unique analytic center for P.

Note that in the next chapters, on variational inequalities, the definition of analytic center will be extended to sets defined by linear inequalities *and* linear equalities, as well as to sets defined by linear and quadratic inequalities.

Although its definition involves the maximization of a function, the analytic center's original context, the polyhedron, is unrelated to optimization. The relation comes via the concept of *set of localization of the optimum*; see [37]. Broadly speaking, the set of localization is, for a mathematical program

(MP) in $\mathbb{R}^m$, a bounded polyhedral set $Y \subset \mathbb{R}^m$ which contains y^*, the optimum of (MP). Alternatively, the set of localization can also be placed in $\mathbb{R}^m \times \mathbb{R}$ in which case it contains (y^*, z^*) where z^* is the optimal value of (MP).

Consider once again the primal and dual problems (1.10) and (1.11)

$$\begin{array}{ll} \text{minimize} & c^t x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array} \tag{1.22}$$

and

$$\begin{array}{ll} \text{maximize} & b^t y \\ \text{subject to} & A^t y \leq c \end{array} \tag{1.23}$$

and v_l, a lower bound for the optimal value of the primal, and hence for that of the dual. Assume for simplicity that the feasible sets are bounded. Then we define the set of localization for the dual problem (1.23) with lower bound v_l as

$$\mathcal{L}_{v_l} = \{y \in \mathbb{R}^m : by \geq v_l, A^t y \leq c\}$$

The analytic center of the set of localization is then

$$\begin{aligned} \bar{x} &= \arg\max_{y \in \mathcal{L}_{v_l}} \log(b^t y - v) \prod_{i=1}^{n} (c_i - a_i^t y) \\ &= \arg\max_{y \in \mathcal{L}_{v_l}} \left\{ \log(b^t y - v) + \sum_{i=1}^{n} \log(c_i - a_i^t y) \right\} \end{aligned}$$

where a_i is the i^{th} column of A and c_i is the i^{th} element of c.

As mentioned, the set of localization could also be defined in the space $\mathbb{R}^m \times \mathbb{R}$:

$$\mathcal{L}_{v_l} = \{(y, v) \in \mathbb{R}^m \times \mathbb{R} : by \geq v, v \geq v_l, A^t y \leq c\}$$

with a corresponding analytic center in $\mathbb{R}^m \times \mathbb{R}$ as well.

In view of the previous section, the rationale for discussing analytic centers in a chapter on interior point methods is now clear: finding the analytic

center $\bar{x}$ amounts to solving the problem

$$\begin{array}{ll} \text{maximize} & \log(b^t y - v_l) + \sum_{i=1}^{n} \log(c_i - a_i^t y) \\ \text{subject to} & A^t y \leq c \\ & b^t y \geq v_l \end{array}$$

which is the problem (1.18) of the last section; therefore, the optimality conditions (1.19)

$$\begin{aligned} Ax &= b \\ Zx &= (b^t y - v_l)e \\ A^t y + z &= c \\ x &> 0 \\ z &> 0 \end{aligned}$$

define uniquely the analytic center of $\mathcal{L}_{v_l}$, and the analytic center lies on the (dual) central path.

Furthermore, let us recall that the conditions (1.15) of the *primal* potential function method are equivalent to (1.19) so that either of the primal and dual potential methods leads directly to the analytic center. As a matter of fact, any interior point method following the central path, e.g. the log barrier methods, could be used, once the relation between the parameterizations is made explicit.

1.4 Analytic Center Cutting Plane Method

In this section, the pieces of theory of the previous sections are put together: in a cutting plane framework, analytic centers are advantageously applied to large-scale linear programs through a nondifferentiable optimization formulation. Note that the actual techniques for finding analytic centers are not covered here but in chapters 3 and 4 on variational inequalities.

The analytic center cutting plane method, or ACCPM, is a cutting plane approach to convex mathematical programming. The main improvement concerns choosing a point at which the next cutting plane is generated; indeed, the classical approach is to use the point optimizing the current relaxation of the objective function. It is then implicitly assumed that the relaxation is a good approximation of the function, so that the optimum point of one is close to the optimum point of the other. This needs not be true, and it is why the ACCPM uses a more balanced approach: the point used to generate the next cutting plane is the center of the set of localization, set in which the true optimum of the original function is known to lie.

1.4.1 Nondifferentiable optimization with the ACCPM

The ACCPM is a cutting plane method, and in this respect its main function is to solve nondifferentiable, convex mathematical programs. This naturally includes any problem that can be rewritten as an nondifferentiable convex program, like the decomposable linear programs of section 1.2.2. The only reason why smooth convex cases are not often good candidates, is that methods better than cutting planes frequently exist for them.

It is most natural to use the framework of section 1.2 on nondifferentiable optimization to describe the ACCPM. Consider the problem

$$\begin{array}{ll} \text{minimize}_{x\in\mathbb{R}^n} & f(x) \\ \text{subject to} & g(x) \leq 0 \end{array} \tag{1.24}$$

with f, g convex, not necessarily differentiable functions. The usual cutting planes approach substitutes a linear tangential approximation $\hat{f}, \hat{g}$ for each function f, g

$$\hat{f}(x) := \max_{i=1,\ldots,k} \{f(y_i) + \gamma_i^f (x - y_i)\} \tag{1.25}$$

$$\hat{g}(x) := \max_{i=1,\ldots,k} g(y_i) + \gamma_i^g (x - y_i) \tag{1.26}$$

As was already pointed out in section (1.2.4), the relaxation of problem (1.24)

$$\begin{array}{ll} \text{minimize}_{x \in \mathbb{R}^n} & \hat{f}(x) \\ \text{subject to} & \hat{g}(x) \leq 0 \end{array}$$

has the linear programming equivalent

$$\begin{array}{ll} \text{minimize}_{x,v} & v \\ \text{subject to} & f(y_i) + \gamma_i^f (x - y_i) \leq v \\ & g(y_i) + \gamma_i^g (x - y_i) \leq 0 \\ & \forall i = 1, \ldots, k \end{array} \tag{1.27}$$

The classical method generates further cuts at the point $(\hat{x}, \hat{v})$ which minimizes (1.27). The ACCPM uses instead the *analytic center* $(\bar{x}, \bar{v})$ of the localization set $\mathcal{L}_{v_u} \subset \mathbb{R}^n \times \mathbb{R}$ associated with the upper bound v_u of (1.24)

$$\mathcal{L}_{v_u} = \{(x, v) \mid v \leq v_u, f(y_i) + \gamma_i^f (x - y_i) \leq v, g(y_i) + \gamma_i^g (x - y_i) \leq 0; i = 1, \ldots, k\}$$

The upper bound v_u is typically the lowest value of $f(x)$ over the points at which is was evaluated so far. The analytic center is then

$$(\bar{x}, \bar{v}) = \arg \max_{(x,v) \in \mathcal{L}_{v_u}} \left\{ \log(v_u - v) + \sum_{i=1}^{k} \log(v - f(y_i) - \gamma_i^f (x - y_i) \right. \tag{1.28}$$

$$\left. + \sum_{i=1}^{k} \log(-g(y_i) + \gamma_i^g (x - y_i)) \right\} \tag{1.29}$$

See a depiction of the above concepts in Figure 1.1.

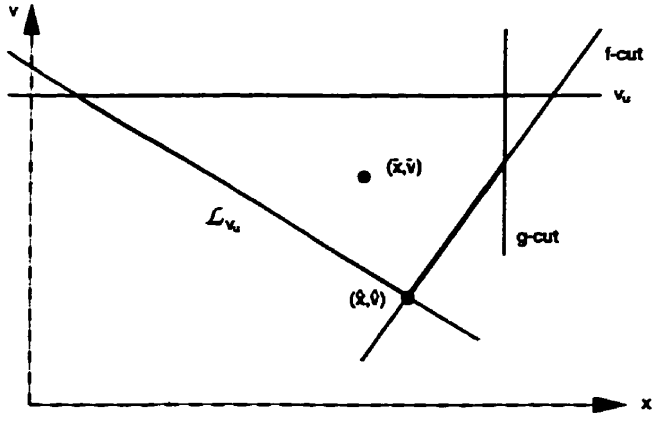


Figure 1.1: Localization set and analytic center

Interior point methods contribute to the ACCPM by providing an efficient way of computing the analytic center. This is far from a trivial contribution, since other centers of polyhedra, for example the center of mass, are computationally difficult or expensive to obtain.

The problem defining the analytic center can be solved by a variety of interior point methods. The one used in the current implementation of ACCPM is a primal projective method; in our notation, this method works on the dual of problem (1.27). See [16].

The $\bar{x}$–part of the analytic center is then passed on to the oracles to generate new cutting planes, as is done in the classical cutting plane method. The oracle for the function f computes $f(\bar{x})$ and a subgradient $\bar{\gamma}^f \in \partial f(\bar{x})$

$$\bar{x} \longrightarrow \boxed{\text{oracle for } f} \longrightarrow f(\bar{x}), \bar{\gamma}^f \in \partial f(\bar{x})$$

The output of the oracle is used to improve the approximation $\hat{f}(x)$ in (1.25).

Similarly, the constraint function g also has its oracle, which computes $g(\bar{x})$ and a subgradient $\bar{\gamma}^g \in \partial g(\bar{x})$

$$\bar{x} \longrightarrow \boxed{\text{oracle for } g} \longrightarrow g(\bar{x}), \bar{\gamma}^g \in \partial g(\bar{x})$$

Again, the oracle's output is used to improve the approximation (1.26) of g. Denoting $x_{k+1} := \bar{x}, \gamma^f_{k+1} := \bar{\gamma}^f$ and $\gamma^g_{k+1} := \bar{\gamma}^g$, we then have a better relaxation

$$\begin{array}{ll} \text{minimize}_{x,v} & v \\ \text{subject to} & f(y_i) + \gamma^f_i (x - y_i) \leq v \\ & g(y_i) + \gamma^g_i (x - y_i) \leq 0 \\ & \forall i = 1, \ldots, k+1 \end{array}$$

of the original problem.

This is the simplest approach: insert the newly generated cuts, without any further consideration. It has an important disadvantage: the number of

linear constraints involved in (1.27) may become unmanageably high. Therefore, an add–and–drop scheme which periodically removes constraints that have become useless, may be necessary. [32] discusses the use of ellipsoids containing the set of localization; clearly, any cutting plane which does not cut the ellipsoid is superfluous and should be deleted.

1.4.2 Weighted Analytic Centers

As already mentioned in section 1.3 on interior point methods, another set of localization can be defined, this time in $\mathbb{R}^n$ and not in $\mathbb{R}^n \times \mathbb{R}$ as above:

$$\mathcal{L}_{v_u} = \{x \in \mathbb{R}^n \mid f(y_i) + \gamma_i^f (x - y_i) \leq v_u, g(y_i) + \gamma_i^g (x - y_i) \leq 0, \forall i = 1, \ldots, k\}$$

Clearly, the analytic center of the polyhedron $\mathcal{L}_{v_u}$ is then also in $\mathbb{R}^n$:

$$\bar{x}' = \arg\max_{x \in \mathcal{L}_{v_u}} \left\{ \sum_{i=1}^{k} \left(\log(v_u - f(y_i) - \gamma_i^f (x - y_i) + \log(-g(y_i) + \gamma_i^g (x - y_i)) \right) \right\}$$

This $\bar{x}'$ is not equal to the $\bar{x}$–part of (1.28); however, there is a "path of centers" which links the two.

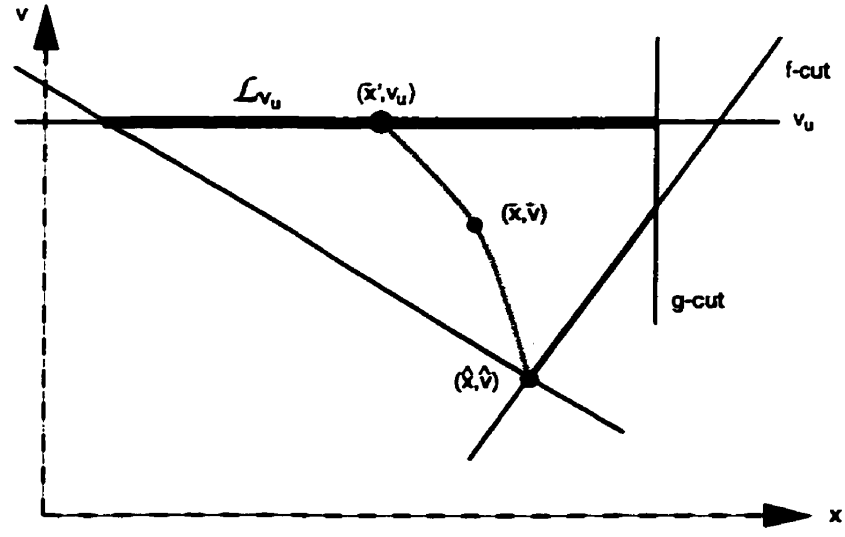


Figure 1.2: A path of centers

This path can be found by considering the maximum in $\mathcal{L}_{v_u}$ of the function

$$\nu\left(\sum_{i=1}^{k}\log\left((v_u - f(y_i) - \gamma_i^f(x-y_i)\right)\right) + \sum_{i=1}^{k}\log(-g(y_i) + \gamma_i^g(x-y_i))$$

The parameter $\nu \geq 1$ is used to vary the "repulsion effect" of the constraints derived from the objective value. For $\nu = 1$, the center defined is $\bar{x}'$; for $\nu = +\infty$, the solution is the point that minimizes the current relaxation of $f(x)$ over $\mathcal{L}_{v_u}$, i.e. the traditional cutting plane point $\hat{x}, \hat{v}$. There is an intermediate value of ν which yields the $\bar{x}$ discussed in the previous section. See [72] and the weighted projective algorithm for weighted analytic centers of [38].

1.4.3 Termination and convergence of the algorithm

In the traditional Cutting Plane method one computes the point $\hat{x}$ which minimizes the current relaxation of the problem with a value $\hat{v}$, as well as $f(\hat{x})$, the actual objective's value at $\hat{x}$. Provided $\hat{x}$ is feasible, $\hat{v}$ is a lower bound and $f(\hat{x})$ an upper bound for the optimal value of the original problem. Assuming that a finite number of cutting planes suffice to describe f in a small area around x^* (the optimal point of the original problem), it is at least theoretically possible to iterate until $f(\hat{x}) - \hat{v} = 0$: then, all cutting planes necessary to describe f at $\hat{x}$ have been generated; since

$\hat{f}(x)$ is a relaxation of $f(x)$,
$f(\hat{x}) = \hat{f}(\hat{x})$,

and

$\hat{x}$ minimizes $\hat{f}(x)$,

$\hat{x}$ is clearly the sought minimum. In practice one is satisfied as soon as $f(\hat{x}) - \hat{v} < \epsilon$ for some small ϵ, allowing to cut down on computation time and difficulty.

On the other hand, the ACCPM must use a termination criterion of the type upper bound - lower bound $< \epsilon$. Indeed, even if $\hat{f}(x)$ has been refined to perfectly represent $f(x)$ on a small area around x^*, the very fact that the ACCPM does not minimize $\hat{f}(x)$ to determine the next x_{k+1}, implies that the algorithm may not find x^* itself, although it is able to get arbitrarily close to it. The bounds on the original problem are as follows.

The upperbound is maintained through the calls to the oracle for f, the objective function. At each call, the actual value of f is computed for an $\bar{x}$, and the upperbound is updated if $f(\bar{x})$ is lower than the current upperbound.

The lower bound is provided by the interior point method that computes the analytic center. The dual of the relaxation of the original problem is a restriction of the dual of the original problem. Therefore the *feasible* dual solution, available when the analytic center is computed, directly yields a lower bound for the optimum of the dual of the original problem, hence for the optimum of the original problem too. Every time an analytic center is computed, the current lower bound is replaced by the newly computed one if the latter is higher.

Results on the convergence and complexity of the ACCPM have appeared in the recent years. For example, the "practical" version of the algorithm, which computes *approximate* analytic centers and eliminates constraints that have become useless, is extensively treated by Atkinson and Vaidya in [3].

While that article treats the convex feasibility problem (finding a point in the interior of a convex set), that of Nesterov [79] deals directly with the optimization problem. His convergence results served as a basis for some of the work by Goffin, Luo and Ye [33]. This paper studied a version of ACCPM applied to the convex feasibility problem; in opposition to Atkinson and Vaidya, they do not rely on cutting planes being dropped along the way to achieve convergence.

Kiwiel and Altman ([52] and [1]) consider the ACCPM for minimizing an unconstrained convex function with exact analytic center computations. They prove convergence and give a complexity bound. See the articles themselves or the thesis [22] for details.

1.4.4 Variation on a theme: IPM in subproblems

In this section, we digress somewhat from the theory that underlies ACCPM-VI, to introduce an idea that was not explored in detail within this doctoral work, but that nevertheless is promising. It must be added that since the time of our work on this topic, Gondzio and Vial have written the report [39] on the subject.

The analytic center cutting plane method, as a cutting plane method, requires the solution of two distinct problems alternately: a master problem (e.g. (1.27)) and subproblem, also called oracle. An interior point method is used at the master problem level in the ACCPM, to find analytic centers. In many important applications, the subproblem, which remained a "black box" (or oracle) so far in this chapter, is in fact a linear program. Traditionally, such a subproblem is solved with the Simplex method; however, an interior point method may be a better choice, although the motivation here is different from the reasons which lead to interior point methods at the Master level.

One foremost technique which yields subproblems that are linear programs, is the Dantzig-Wolfe decomposition (see [73] for an excellent description). We use that framework to expose the concepts of interior point methods in the subproblems. The typical setup of Dantzig-Wolfe decomposition is written as:

Relaxed master problem

$$\begin{array}{ll} \min_{v,\pi} & v \\ \text{subject to} & v \geq z_1 + \gamma_1 \pi \\ & \vdots \\ & v \geq z_k + \gamma_k \pi \end{array} \tag{1.30}$$

Subproblem

$$\begin{array}{ll} \max_{x} & (c + \pi A)^t x \\ \text{subject to} & Dx = d \\ & x \geq 0 \end{array} \tag{1.31}$$

The subproblem stems from the desire to generate a cut as "useful" as possible in the Master: that is, to find scalar $\bar{z}$ and (sub)gradient $\bar{\gamma}$ such that $\bar{z} + \bar{\gamma}^t \bar{\pi}$ is largest, for a given $\bar{\pi}$. Since $z := c^t y_i$ and $\gamma := b - A y_i$, where $y_i, i \in \{1, \ldots, q\}$ is one of q vertices of $\mathcal{D} = \{x | Dx = d, x \geq 0\}$, we have

$$z + \gamma \bar{\pi} = c^t y_i + (b - A y_i)^t \bar{\pi} = b^t \bar{\pi} + (c - \bar{\pi} A)^t y_i$$

and the subproblem above with $\pi = \bar{\pi}$ will yield the desired answer. By definition, the cut thus generated $\bar{z} + \bar{\gamma} \bar{\pi}$ is tangent to the epigraph of $f(\pi) = \max_{i \in 1, \ldots, q} z_i + \gamma_i \pi$ at $\bar{\pi}$.

Our proposal is to generate "sub-optimal cuts" from sub-optimal solutions to the subproblem. Consider any point $\tilde{x} \in \text{int}(\mathcal{D})$. Clearly, since $\tilde{x}$ is a convex combination of vertices of $\mathcal{D}$, $\tilde{x} = \sum_{i=1}^{q} \lambda_i y_i$, with $\sum_{i=1}^{q} \lambda_i = 1$ and $\lambda_i \geq 0$, the inequality

$$v \geq c^t \tilde{x} + (b - A\tilde{x})^t \pi \tag{1.32}$$

is redundant to the system

$$\begin{array}{rcl} v & \geq & c^t y_1 + (b - A y_1)^t \pi \\ & \vdots & \\ v & \geq & c^t y_q + (b - A y_q)^t \pi \end{array}$$

In other words, adjoining that inequality to the relaxed Master problem can only help.

Assume furthermore that $\tilde{x}$ is on the central path of the subproblem with $\bar{\pi}$, i.e. $\exists\, \tilde{\omega}$ and $\tilde{\zeta} > 0$ such that

$$\begin{aligned} D\tilde{x} &= d \\ D^t\tilde{\omega} + \tilde{\zeta} &= c + \bar{\pi}A \\ \tilde{X}\tilde{\zeta} &= \mu e \end{aligned}$$

At such a point, the duality gap is known exactly: it is $e^t\tilde{X}\tilde{\zeta} = n\mu$. Therefore

$$(c + \bar{\pi}A)^t x^* - (c + \bar{\pi}A)^t\tilde{x} \leq n\mu$$

where x^* is the subproblem's optimal answer and

$$f(\bar{\pi}) - (c^t\tilde{x} + (b - A\tilde{x})^t\bar{\pi}) = (c + \bar{\pi}A)^t(x^* - \tilde{x}) \leq n\mu \qquad (1.33)$$

Furthermore,

$$0 \leq f(\pi) - (c^t\tilde{x} + (b - A\tilde{x})^t\pi),\ \forall\pi$$

by the very definition of $f(\pi)$; equivalently,

$$(c^t\tilde{x} + (b - A\tilde{x})^t\bar{\pi}) + (b - A\tilde{x})^t(\pi - \bar{\pi}) \leq f(\pi),\ \forall\pi \qquad (1.34)$$

Adding (1.33) and (1.34)

$$f(\bar{\pi}) + (b - A\tilde{x})^t(\pi - \bar{\pi}) \leq f(\pi) + n\mu,\ \forall\pi$$

shows that $(b - A\tilde{x})$ is actually an $n\mu$–subgradient of $f(\pi)$ at $\bar{\pi}$.

To sum up, such a point $\tilde{x}$ yields a valid cut in the master problem, cut whose "distance" from the tangent cut $\bar{z} + \bar{\gamma}\pi$, at $\bar{\pi}$, is a direct function of the quality of $\tilde{x}$ as an answer to SP($\bar{\pi}$). This quality is parameterized by μ: a low μ means a $\tilde{x}$ which is close to optimality in the subproblem; the cut generated with such a $\tilde{x}$ will be near the epigraph. A higher value of μ

renders the subproblem easier to solve, but the derived cut will be "looser", i.e. not as close to the epigraph. In any case it is clear that the value of the parameter μ should be inferior to the difference between the current upper bound and lower bound of the master problem (1.30).

Bounding One unpleasant effect of using suboptimal $\bar{x}$ is that the value $f(\bar{\pi})$ is not available as it would be if optimal x^* was computed. Since $f(\bar{\pi})$ is normally used for upperbounding in the Master problem, a substitute must be found. We suggest using

$$(c + \bar{\pi}A)^t \bar{x} + n\mu$$

which, from (1.33), is at least as large as $f(\bar{\pi})$.

As far as lower bounding of the Master problem is concerned, suboptimal solutions of the subproblem open up a possibility. The cut (1.32) that is being proposed

$$v \geq c^t \bar{x} + (b - A\bar{x})^t \pi$$

becomes a plain lower bound on the objective v if $A\bar{x} = b$. The following feasibility problem can then be attempted:

$$\begin{array}{ll} \text{Find} & x \\ \text{subject to} & Ax = b \\ & Dx = d \\ & x \geq 0 \\ & (c - \bar{\pi}A)^t x \geq (c - \bar{\pi}A)^t \bar{x} \end{array}$$

The last constraint ensures that, should the problem have a solution, the difference between upper and lower bounds is at most $n\mu$; i.e. $\bar{\pi}$ is $n\mu$–optimal for the original problem. Needless to say, this lower-bounding attempt is likely to be a jump from the frying pan into the fire: the feasibility problem contains all the constraints of the original problem, a problem that was itself worth being decomposed.

To sum up, the main advantage of the interior point methods and sub-optimal solutions in the subproblem, is to allow to shift the computational

burden from the Master Problem to the subproblem, and vice-versa, by adjusting the parameter μ: a high μ simplifies solving the subproblem, but yields loose cuts in the Master; a lower μ gives better cuts, at the expense of more work in the subproblem.

However, until some practical experience is gained with this approach, it remains difficult to establish if it can compete favorably with the standard approach.

Chapter 2

Variational Inequalities

This chapter reviews the field of variational inequalities. We present fundamental definitions and results in Sections 2.1 and 2.2. Section 2.3 discusses complementarity problems, an important subset of variational inequalities, while Section 2.4 is devoted to classical reformulations of the variational inequality problem. We review in Section 2.5 the main classes of algorithms for variational inequalities. The last section of the chapter discusses extensions to variational inequalities with multi-valued (point-to-set) functions; until this section, all mappings are assumed to be single-valued.

For general references on variational inequalities, the reader can consult the books by Harker [43] and Nagurney [77], the review article of Harker and Pang [44], the short course of Marcotte [69], and the recent proceedings book of Ferris and Pang [24]. See also the various articles cited in this chapter.

2.1 Variational Inequalities: Fundamentals

This section serves to define the variational inequality (VI, or VIP for variational inequality problem). A short example is given to illustrate it.

2.1.1 Basic Concepts

Definition 1 *Let F be a continuous mapping from $\mathbb{R}^m$ into $\mathbb{R}^m$ and let Y be a nonempty subset of $\mathbb{R}^m$. Then the* variational inequality problem, *denoted $VI(F,Y)$, is to find a point $y^* \in Y$ such that*

$$F(y^*)^t(y - y^*) \geq 0 \quad \forall y \in Y \qquad VI(F,Y)$$

Geometrically, a solution y^* is a point of Y such that $F(y^*)$ makes an acute angle with all feasible directions, i.e. $-F(y^*)$ belongs to the *normal cone* to Y at y^*:

$$-F(y^*) \in \{z : (y - y^*)^t z \leq 0 \quad \forall y \in Y\}$$

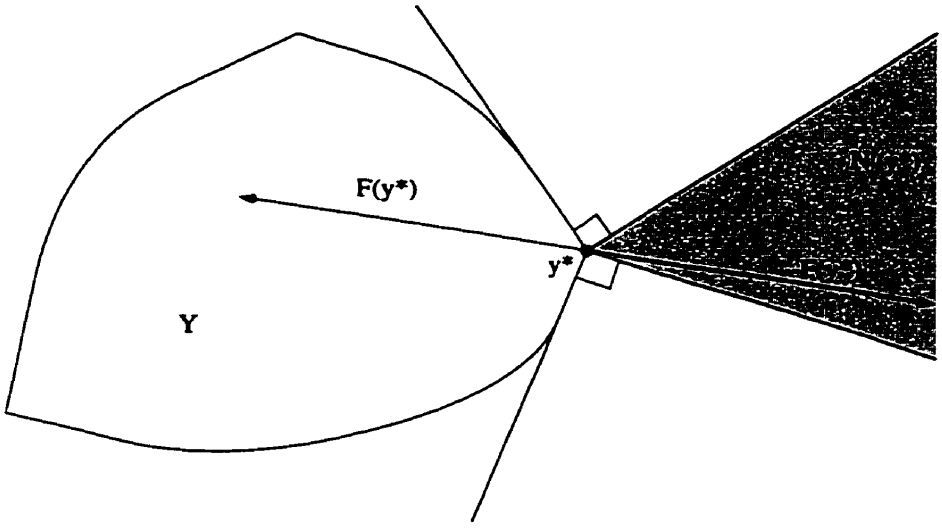


Figure 2.1: Geometric view of VI, with normal cone

Whenever the mapping F is a *gradient mapping*, i.e. $F(y) = \nabla f(y)$ for some function $f : \mathbb{R}^m \to \mathbb{R}$, any solution of VI(F,Y) satisfies the first-order conditions of the mathematical program:

$$\begin{array}{ll} \text{minimize}_y & f(y) \\ \text{subject to} & y \in Y \end{array}$$

In other words, the steepest descent direction $-\nabla f(y)$ makes an obtuse angle with any feasible direction. If f is a convex function and Y is a convex set, this also implies global optimality. A sufficient condition for F to be a gradient mapping is that F be continuously differentiable with symmetric Jacobian ∇F. For this reason, gradient mappings are also called *symmetric* mappings. This is made more precise in the following theorem.

Theorem 1 ([82]) *Let $F : Y \rightarrow \mathbb{R}^m$ be continuously differentiable over the open convex set $Y \subseteq \mathbb{R}^m$. If the Jacobian matrix $\nabla F(y)$ is symmetric for all $y \in Y$, then $F = \nabla f$ where*

$$f(y) = \int_0^1 F(y_0 + t(y - y_0))^t (y - y_0) dt$$

and y_0 is an arbitrary point in Y.

Often used concepts are the **elliptic norm** and the **projection**, which we now define.

Definition 2 *For a symmetric, positive definite $m \times m$ matrix Q and for a vector $y \in \mathbb{R}^m$, let the* elliptic norm induced by the matrix Q (the Q-norm) *be defined as $\|y\|_Q = (y^t Q y)^{\frac{1}{2}}$.*

Also, for a closed, non-empty, convex set Y in $\mathbb{R}^m$, define the projection of y on Y under the Q-norm *as the (unique) solution of*

$$\begin{array}{ll} \textit{minimize}_x & \frac{1}{2}\|y - x\|_Q^2 \\ \textit{subject to} & x \in Y \end{array}$$

The projection is denoted by $\Pi_{Y,Q}(y)$, or simply $\Pi_Y(y)$ when Q is the identity matrix, i.e. under the Euclidian norm.

2.1.2 A Short Example: Nash Equilibrium

Consider two players who must each select a real number between 1 and 3. Let x_i denote the number selected by player i, $i = 1, 2$, and u_1 and u_2 the

utility derived from it. The utility could be an amount of money, of leisure time, etc. A Nash equilibrium is achieved when the decisions x_1^* and x_2^* are such that neither player can improve his or her utility unilaterally, i.e.

$$x_1^* \in \arg\max_{x_1 \in [1,3]} u_1(x_1, x_2^*) \tag{2.1}$$

$$x_2^* \in \arg\max_{x_2 \in [1,3]} u_2(x_1^*, x_2). \tag{2.2}$$

Let us make the assumptions that the set of feasible decisions is closed, convex, and nonempty, and that, for each i, the utility u_i is both continuously differentiable and pseudo-concave with respect to x_i with x_j, $j \neq i$ fixed. Then the Nash equilibrium conditions (2.1) are equivalent to

$$-(\partial u_1(x_1^*, x_2^*)/\partial x_1)(x_1 - x_1^*) \geq 0 \quad \forall x_1 \in [1,3]$$
$$-(\partial u_2(x_1^*, x_2^*)/\partial x_2)(x_2 - x_2^*) \geq 0 \quad \forall x_2 \in [1,3]$$

This follows directly from the variational principle for optimality, applied to (2.1).

We will consider in turn three possible utility functions, and their impact on the difficulty of finding an equilibrium point.

First, let

$$u_1(x_1, x_2) = -(x_1 - 2)^2 + 1$$
$$u_2(x_1, x_2) = -x_2^2.$$

Clearly, since u_1 is a function of x_1 only, and u_2 is a function of x_2 only, the maximization problems above can be solved independently, yielding a unique Nash equilibrium at (2,1).

Second, consider the functions

$$u_1(x_1, x_2) = -3(x_1 - 2)^2 + 1 - x_1 x_2$$
$$u_2(x_1, x_2) = -2x_2^2 - x_1 x_2.$$

Clearly, finding an equilibrium point, i.e. a pair (x_1^*, x_2^*) such that (2.1) holds, is not as easy as the utility of each player depends also on the decision

of the other. An example of such a case is a duopoly market, where two producers must decide on a production level of a product; their profits, — their utility levels—, depend on both their production and the production of the competitor. (Negative profits are simply losses.)

Here the maximization problems cannot be solved independently. However, with Theorem 1 of the previous section, an optimization formulation is still possible, thanks to the fact that the Jacobian of the mapping

$$U(x_1, x_2) := -\begin{bmatrix} \partial u_1(x_1, x_2)/\partial x_1 \\ \partial u_2(x_1, x_2)/\partial x_2 \end{bmatrix}$$

is $\begin{bmatrix} -6 & -1 \\ -1 & -4 \end{bmatrix}$ and thus symmetric. Indeed, here, the problem

$$\max_{(x_1,x_2)\in[1,3]^2} 3x_1^2 + 2x_2^2 + x_1x_2 - 12x_1 + 6,$$

yields a unique solution $(11/6, 1)$ that is an equilibrium point.

Let us consider a final set of utility functions, very similar to the last:

$$\begin{aligned} u_1(x_1, x_2) &= -3(x_1 - 2)^2 + 1 + 3x_1x_2 \\ u_2(x_1, x_2) &= -2x_2^2 + x_1x_2. \end{aligned}$$

Here, the Jacobian $\nabla U(x_1, x_2)$ is *not* symmetric, so that there is no "easy" optimization problem equivalent to the Nash equilibrium problem. However, a Nash point can be found as the solution of the variational inequality problem

$$\text{Find } (x_1^*, x_2^*) \in [1,3]^2 \text{ such that}$$
$$U(x_1^*, x_2^*)^t \begin{bmatrix} x_1 - x_1^* \\ x_2 - x_2^* \end{bmatrix} \geq 0 \ \ \forall\, (x_1, x_2) \in [1,3]^2$$

whose unique solution is $(3/2, 1)$.

2.2 Properties of F. Existence and uniqueness of a solution

First, some commonly used definitions:

Definition 3 *Over a set Y, a mapping $F : \mathbb{R}^m \to \mathbb{R}^m$ is*

- monotone *if*

$$(F(y_1) - F(y_2))^t(y_1 - y_2) \geq 0 \quad \forall y_1, y_2 \in Y$$

- strictly monotone *if*

$$(F(y_1) - F(y_2))^t(y_1 - y_2) > 0 \quad \forall y_1, y_2 \in Y, \quad y_1 \neq y_2$$

- strongly monotone with modulus α *(or α−strongly monotone) if*

$$(F(y_1) - F(y_2))^t(y_1 - y_2) \geq \alpha \|y_1 - y_2\|^2 \quad \forall y_1, y_2 \in Y, \quad y_1 \neq y_2$$

- pseudo-monotone *if*

$$F(y_2)^t(y_1 - y_2) \geq 0 \Rightarrow F(y_1)^t(y_1 - y_2) \geq 0 \quad \forall y_1, y_2 \in Y$$

- co-coercive with modulus α *(or α-co-coercive) if*

$$(F(y_1) - F(y_2))^t(y_1 - y_2) \geq \alpha \|F(y_1) - F(y_2)\|^2 \quad \forall y_1, y_2 \in Y, \quad y_1 \neq y_2$$

- pseudo-co-coercive with modulus α *(or α-pseudo-co-coercive) if*

$$F(y_2)^t(y_1 - y_2) \geq 0 \Rightarrow$$
$$F(y_1)^t(y_1 - y_2) \geq \alpha \|F(y_1) - F(y_2)\|^2 \quad \forall y_1, y_2 \in Y, \quad y_1 \neq y_2$$

- Lipschitz continuous with modulus L *if*

$$\|F(y_1) - F(y_2)\| \leq L\|y_1 - y_2\| \quad \forall y_1, y_2 \in Y$$

for some vector norm $\|\cdot\|$.

Clearly, of the monotonicity properties, pseudo-monotonicity is the weakest and strong monotonicity the strongest. Also, co-coercivity implies monotonicity, and pseudo-co-coercivity implies pseudo-monotonicity. See Figure 2.2. Various other properties of mappings, and relationships between them, can be found in Zhu and Marcotte [113]. Note that co-coercivity has also been called *strong-F-monotonicity* by some authors, e.g. [66].

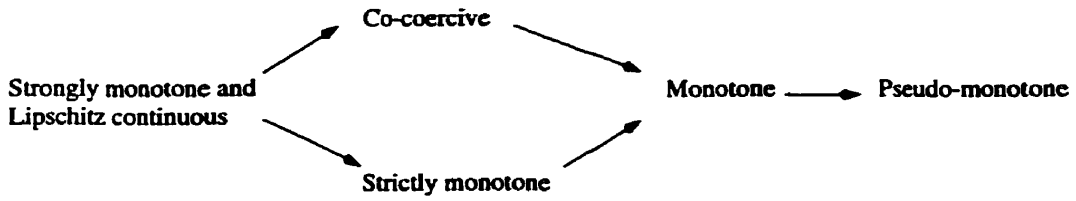


Figure 2.2: Relationships between some monotonicity properties

When F is continuously differentiable, then the monotonicity of F can be related to the Jacobian ∇F as follows.

Lemma 1 ([82]) *If a mapping* $F : \mathbb{R}^m \to \mathbb{R}^m$ *is once continuously differentiable over an open set* Y*, then*

- F *is monotone if and only if* $\nabla F(y)$ *is positive semidefinite for all* $y \in Y$.
- F *is strictly monotone if* $\nabla F(y)$ *is positive definite for all* $y \in Y$.
- F *is* $\alpha-$*strongly monotone if and only if* $\nabla F(y)$ *is* α*-strongly positive definite for all* $y \in Y$*, i.e. there exists a constant* $\alpha > 0$ *such that*

$$z^t \nabla F(y) z \geq \alpha z^t z \; \forall y \in Y, \; \forall z \in \mathbb{R}^m.$$

Remark that the strict monotonicity does not imply the positive definiteness, as the example $F(y) = y^2$ over $Y = \mathbb{R}$ shows when $y = 0$.

One complete reference on the topic of monotonicity is the book of Ortega and Rheinboldt [82]. Note that by a slight abuse of language, we sometimes refer to a variational inequality $VI(F, Y)$ with (strongly-, strictly-, pseudo-)monotone mapping F as, simply, a (strongly-, strictly-, pseudo-)monotone variational inequality.

Equipped with these definitions, we can discuss the conditions of existence and uniqueness of a solution of the variational inequality problem. The **existence** of a solution of $VI(F, Y)$ when Y is compact and convex and F is continuous is a well known result, whose proof relies on the continuity of the projection mapping and on Brouwer's fixed-point theorem (see [54]). When Y is not compact, e.g. for lack of boundedness, stronger conditions are required of F. For example, when Y is closed, convex, and nonempty, the strong monotonicity of F guarantees the existence of a unique solution to $VI(F, Y)$.

Under the pseudo-monotonicity of F the solution set is convex while under the strict monotonicity of F, the solution is unique if there is one. See the general references [43] and [77] for more details.

2.3 Box-Constrained Variational Inequalities and Complementarity Problems

The problems presented in this section are restrictions of variational inequalities. They nevertheless form a very important class, whose development was, and is still, intimately linked to that of general VIP's. Like variational inequalities, finite-dimensional complementarity problems have become a topic of intense interest in the last decade; this is well illustrated by the award of the Mathematical Programming Society's 1997 Beale-Orchard-Hayes prize to S. Dirkse and M. Ferris for a mixed-complementarity problem algorithm and

solver.

We first define the nonlinear and linear complementarity problems.

Definition 4 *Let F be a mapping from $\mathbb{R}^m$ into $\mathbb{R}^m$. The* nonlinear complementarity problem, NCP(F), *is to find a point y^* in $\mathbb{R}^m$ such that*

$$y^* \geq 0 \quad F(y^*) \geq 0 \quad F(y^*)^t y^* = 0 \qquad NCP(F)$$

In the special case where the mapping F is linear, $F(Y) = My + v$, the problem is called a linear complementarity problem, *and denoted* LCP(F).

These problems were defined 30 years ago, as a way to model and solve matrix games and economic equilibria. An efficient algorithm for the LCP, Lemke's pivotal algorithm, was discovered early, and has kept a certain level of popularity as a subproblem solver for linearized VI methods. This pivotal algorithm was also extended to the NCP. General references on the LCP are Cottle, Pang and Stone [14] and Murty [76], and Chan and Pang [85] for the NCP.

We now introduce the more general class of box-constrained variational inequalities, also known as mixed complementarity problems. This class has been described in at least three different ways, outlined below.

Let $F : \mathbb{R}^m \to \mathbb{R}^m$, $G : \mathbb{R}^{p+q} \to \mathbb{R}^p$ *and* $L : \mathbb{R}^{p+q} \to \mathbb{R}^q$ *be mappings. Let* l *and* u *be vectors in the extended reals,* $\{\mathbb{R} \cup -\infty \cup \infty\}^m$, *with* $l < u$, *thus defining a* box $[l, u]$ *in* $\{\mathbb{R} \cup -\infty \cup \infty\}^m$. *Then,*

Definition 5 *The* **box-constrained variational inequality problem** *is*

$$\text{Find } \; y^* \in [l, u] \; \text{ such that } \; F(y^*)^t(y - y^*) \geq 0 \quad \forall\, y \in [l, u] \tag{2.3}$$

Definition 6 *The* **mixed complementarity problem (MCP)** *is*

Find $(x, z) \in \mathbb{R}^p \times \mathbb{R}^q$ such that

$$\begin{array}{rcl} L(x, z) & = & 0 \\ G(x, z) & \geq & 0 \\ x & \geq & 0 \\ x^t G(x, z) & = & 0 \end{array} \tag{2.4}$$

Definition 7 *The* **box-constrained VI in KKT form** *is*

Find $(y, v, w) \in \mathbb{R}^m \times \mathbb{R}^m_+ \times \mathbb{R}^m_+$ such that

$$\begin{array}{rcl} F(y) + v - w & = & 0 \\ l \leq y \leq u & & \\ (y - l)^t w & = & 0 \\ (u - y)^t v & = & 0 \end{array} \tag{2.5}$$

A box-constrained variational inequality, also called rectangular variational inequality, is a $VI(F, Y)$ whose set Y is a box. The MCP derives its name from the fact that some variables have a nonnegativity constraint and complementarity condition, while the others are free; it is thus a "mix" of complementarity conditions and usual equations. The rationale for the third form's name will be clear from Theorem 2 below.

We warn immediately that the above notation reflects our personal preference. Although many authors would agree with them, others prefer to call (2.5) the mixed complementarity problem. Disagreements are largely due to

the fact that any of the three forms above can be rewritten in the other two formats, although the number of variables may vary.

For example, an MCP can be written as a box-constrained VI by setting

$$y := \begin{bmatrix} x \\ z \end{bmatrix} \quad F(y) := \begin{bmatrix} G(x,z) \\ L(x,z) \end{bmatrix} \quad l := \begin{bmatrix} 0_{p\times 1} \\ -\infty_{q\times 1} \end{bmatrix} \quad u := \begin{bmatrix} \infty_{p\times 1} \\ \infty_{q\times 1} \end{bmatrix} \tag{2.6}$$

Similarly, an MCP can be written as a box-constrained VI in KKT form by using (2.6) and introducing new variables $v \in \mathbb{R}^m_+$ and $w \in \mathbb{R}^m_+$ in (2.4). Conversely, (2.5) can be written in MCP form with

$$x := \begin{bmatrix} v \\ w \end{bmatrix} \quad z := y \quad L(x,z) := F(y) + v - w \quad G(x,z) := \begin{bmatrix} u - y \\ y - l \end{bmatrix}.$$

The equivalence between the box-constrained VI and its KKT formulation follows as a special case of Theorem 2 below. In terms of number of variables, it is clear that the same problem, expressed as a box-constrained VI in KKT form, will have 3 times as many variables as if it were cast in the forms (2.3) or (2.4).

We conclude this section with an important theorem that can be found in [92]. For a variational inequality problem $VI(F, Y)$ satisfying some assumptions, this theorem gives a set of equations and inequalities that is equivalent to the variational problem. In view of the very close link to the Karush-Kuhn-Tucker conditions of optimization problems, this set of conditions has been referred to as "KKT conditions" of the variational inequality problem.

Theorem 2 *Let F be a mapping from $\mathbb{R}^m$ into $\mathbb{R}^m$. Let Y be defined as*

$$Y = \{y \in \mathbb{R}^m \mid G(y) \geq 0,\ H(y) = 0\}$$

where $G : \mathbb{R}^m \to \mathbb{R}^p$ is differentiable and $H : \mathbb{R}^m \to \mathbb{R}^r$ is affine. Then:

1. *Under some constraint qualification, if y^* solves $VI(F,Y)$ then there exists $\pi \in \mathbb{R}^p$ and $\mu \in \mathbb{R}^r$ such that*

$$F(y^*) - \nabla G(y^*)\pi - \nabla H(y^*)\mu = 0 \quad (2.7)$$

$$\pi^t G(y^*) = 0 \quad (2.8)$$

$$\pi \geq 0 \quad (2.9)$$

2. *If the components of $G(y)$, $G_i(y)$, $i = 1, \ldots, p$, are concave functions, $y^* \in Y$, and (y^*, π^*, μ^*) satisfies (2.7)–(2.9), then y^* solves $VI(F,Y)$.*

The constraint qualification evoked in *1.* corresponds to one of the usual qualifications of nonlinear programming: linearity of the constraint functions or linear independence of the gradients of the active constraints at y^*, for example.

Note that if both G and H are affine mappings, then (2.7)–(2.8) and $y^* \in Y$ form a **necessary and sufficient set of conditions** for y^* to solve $VI(F,Y)$. If in particular, there are no equality constraints, and

$$G(y) = \begin{bmatrix} u - y \\ y - l \end{bmatrix} \geq 0,$$

then (2.7)–(2.8) reduces to (2.5). This justifies the name box-constrained VI in KKT form given to 2.5.

2.4 Reformulations of variational inequalities

The variational inequality problem can be rewritten in different ways: as a fixed-point problem, as a system of equations, as an optimization or convex feasibility problem, as a nonlinear or mixed complementarity problem, etc. Some reformulations are general, some require assumptions on the mapping F or the set Y.

Each formulation opens the door to several algorithmic methods. In that sense, the fixed-point problem and the system of equations approaches have been the most popular. On the other hand, the algorithms developed in this thesis are derived from the convex feasibility problem formulation.

2.4.1 Fixed-point formulation

The variational inequality problem can be reformulated as a **fixed-point problem** for the mapping $M : \mathbb{R}^m \rightarrow \mathbb{R}^m$:

$$\text{Find a point } y^* \text{ such that } y^* = M(y^*). \tag{2.10}$$

The fixed-point formulation is the cornerstone of several existence results as well as algorithms. It can be described very succinctly, using the general iterative scheme of Dafermos [15].

Consider the strictly monotone variational inequality problem $VI(F, Y)$ with Y compact and convex. Let $\tilde{F}(z, y) : \mathbb{R}^m \times \mathbb{R}^m \rightarrow \mathbb{R}^m$ be a strictly monotone mapping with the property

$$\tilde{F}(y, y) = F(y)$$

Let us also fix y in $\tilde{F}(z, y)$ and call $M(y)$ the solution set of $VI(\tilde{F}, Y)$:

$$M(y) = \{z^* \in Y \mid \tilde{F}(z^*, y)^t(z - z^*) \geq 0 \quad \forall z \in Y\}$$

By the strict monotonicity of $\tilde{F}$, $M(y)$ is a unique point. Clearly then y^* is a solution of $VI(F, Y)$ if and only if y^* is a fixed-point for the mapping M, $y^* \in M(y^*)$.

There is considerable latitude in choosing $\tilde{F}$ and some of the best known choices will be reviewed in Section 2.5. Let us however discuss here one of these choices, based on the concept of projection. The *projection formulation*

is the archetype of fixed-point formulations and is obtained by using the mapping

$$\tilde{F}(z,y) = F(y) + M(z-y),$$

where M is a constant, symmetric, positive-definite matrix. The rationale for the name "projection" will become clear shortly.

We first consider a lemma that gives a basic property of projections.

Lemma 2 *Let Y and Q be defined as above. Then $\bar{y} = \Pi_{Y,Q}(x)$ if and only if*

$$(\bar{y} - x)^t Q(y - \bar{y}) \geq 0 \quad \forall\, y \in Y \tag{2.11}$$

Proof: The point $\bar{y}$ minimizes the function $d(y) = \frac{1}{2}\|y - x\|_Q^2$ over all $y \in Y$; since $\nabla d(y) = Q(y - x)$, the inequality above expresses the (necessary and sufficient) optimality condition of the projection problem. □

Then the following theorem spells out the fixed-point result above for the specific case of a projection mapping (see for example [77]):

Theorem 3 *Let Y be a closed, non-empty and convex subset of $\mathbb{R}^m$, and Q be a symmetric, positive definite $m \times m$ matrix. Then y^* solves $VI(F,Y)$ if and only if*

$$y^* = \Pi_{Y,Q}(y^* - Q^{-1}F(y^*)); \tag{2.12}$$

that is, if and only if y^* is a fixed-point of the mapping $M : \mathbb{R}^m \to \mathbb{R}^m$

$$M(y) := \Pi_{Y,Q}(y - Q^{-1}F(y)).$$

Proof: The point y^* solves $VI(F,Y)$ if and only if

$$F(y^*)^t(y - y^*) \geq 0 \quad \forall\, y \in Y$$

which, multiplying by -1 and adding $(Qy^*)^t(y-y^*)$, is equivalent to

$$\begin{aligned}(Qy^*)^t(y-y^*) &\geq (Qy^*)^t(y-y^*) - F(y^*)^t(y-y^*) \\ &= (Qy^* - F(y^*))^t(y-y^*) \\ &= \left(y^* - Q^{-1}F(y^*)\right)^t Q(y-y^*) \;\; \forall\, y \in Y\end{aligned}$$

and

$$\left(y^* - (y^* - Q^{-1}F(y^*))\right)^t Q(y-y^*) \geq 0 \quad \forall\, y \in Y. \tag{2.13}$$

By Lemma 2, (2.13) holds if and only if

$$y^* = \Pi_{Y,Q}(y^* - Q^{-1}F(y^*))$$

which concludes the proof. □

Using Lemma 2, we see that $\bar{y}$ is the projection point $\Pi_{Y,Q}(y_k - Q^{-1}F(y_k))$ if and only if $\bar{y}$ solves the VIP

$$\left(\bar{y} - (y_k - Q^{-1}F(y_k))\right)^t Q(y-\bar{y}) \geq 0 \quad \forall\, y \in Y,$$

equivalent to

$$\left(F(y_k) + Q(\bar{y} - y_k)\right)^t (y-\bar{y}) \geq 0 \quad \forall\, y \in Y$$

which explains the name "projection method" associated to the fixed-point iterations with $\tilde{F}(z,y) = F(y) + M(z-y)$.

2.4.2 Formulations as systems of equations

The variational inequality problem has several useful reformulations as a system of equations. Indeed, a large proportion of the recent research on algorithms for VI was devoted to such reformulations. Typically, the variational inequality is cast as a nonsmooth system of equations, where the word "nonsmooth" refers to Fréchet differentiability (F-differentiability). However,

a weaker type of differentiability, called Bouligand or B-differentiability, is often used in an adapted version of Newton's method. References on this formulation are given in Section 2.5.2.

Equations based on the fixed-point formulation

One obvious system of equations relies on the fixed-point formulation (2.10) above, i.e. it consists in solving

$$H(y) := y - M(y) = 0$$

In particular, the system of equations derived from the projection mapping $M(y) = \Pi_Y(y - F(y))$ is B-differentiable. See the section on algorithms below and the book of Harker [43, p. 132].

Equations based on Robinson's normal map

Another formulation in terms of systems of equations is based on the *normal map* introduced by Robinson [91].

Theorem 4 ([90],[91]) *Let Y be a closed, non-empty and convex subset of $\mathbb{R}^m$, and Q be a symmetric, positive definite $m \times m$ matrix. If y^* solves the problem $VI(F, Y)$ then*

$$\tilde{y} = y^* - Q^{-1}F(y^*)$$

solves

$$\tilde{H}(y) := F\big(\Pi_{Y,Q}(y)\big) + Q(y - \Pi_{Y,Q}(y)) = 0.$$

Conversely, if $\tilde{y}$ is a zero of the mapping $\tilde{H}(y)$, then $y^* = \Pi_{Y,Q}(\tilde{y})$ solves $VI(F, Y)$.

Proof: By Theorem 3, $\Pi_{Y,Q}(y^* - Q^{-1}F(y^*)) = y^*$, so that

$$\tilde{H}(\tilde{y}) = F(y^*) + Q(\tilde{y} - \Pi_{Y,Q}(\tilde{y})$$

$$= F(y^*) + Q(y^* - Q^{-1}F(y^*) - y^*)$$
$$= 0$$

On the other hand,

$$\tilde{H}(\tilde{y}) = F(\Pi_{Y,Q}(\tilde{y})) + Q(\tilde{y} - \Pi_{Y,Q}(\tilde{y})) = 0$$

implies

$$\tilde{y} = -Q^{-1}F(\Pi_{Y,Q}(\tilde{y})) + \Pi_{Y,Q}(\tilde{y})$$

and

$$\Pi_{Y,Q}(\tilde{y}) = \Pi_{Y,Q}\big(-Q^{-1}F(\Pi_{Y,Q}(\tilde{y})) + \Pi_{Y,Q}(\tilde{y})\big)$$

so that by Theorem 3, $y^* = \Pi_{Y,Q}(\tilde{y})$ solves $VI(F,Y)$. □

Unlike the projection map, the Robinson map has the advantage of being defined everywhere, even when F is defined only over Y.

2.4.3 Convex feasibility formulation

Under the assumptions that F is a pseudo-monotone and continuous mapping, and that Y is a closed, convex and nonempty set, $VI(F,Y)$ can be formulated as a *convex feasibility problem:*

$$\text{Find a point } y^* \in Y^*$$

where Y^* is a closed, convex and bounded set. This can be found from the following theorem, which was proved under monotonicity of F by Minty [74]. The extension to pseudo-monotone mappings is straightforward, but it is not clear to us who first proved it.

Theorem 5 *Let F be a pseudo-monotone and continuous mapping, and Y a closed, convex and nonempty set. Then $y^* \in Y$ solves the $VI(F,Y)$ if and only if $y^* \in Y$ and*

$$F(y)^t(y - y^*) \geq 0 \quad \forall y \in Y \tag{2.14}$$

This effectively means that the solution set Y^* of $VI(F,Y)$, which can eventually consist of a unique point, is defined as the intersection of all half-spaces defined by (2.14). In other words, there is a convex feasibility formulation of $VI(F,Y)$, with the feasibility set Y^* implicitly defined by the infinite family of cutting planes (2.14). Note that (2.14) ensures both the convexity and closedness of Y^*, while $Y^* \subset Y$ ensures its boundedness.

2.4.4 Optimization formulations

Whenever F is a gradient mapping, the variational inequality problem $VI(F,Y)$ corresponds to an optimality condition and can thus also be solved as an optimization problem. This was illustrated in Section 2.1. However, it is possible to reformulate any VI as an optimization problem; two of the best known such formulations are presented here. We now define the *primal gap function.*

Definition 8 *The* gap function *g associated with VI(F,Y) and $y \in Y$ is defined as*

$$g(y) = \inf_{z \in Y} F(y)^t(z - y)$$

Provided Y is compact, the "inf" can be replaced by a "min"; note also that for Y polyhedral, $g(y)$ can be evaluated by solving a linear optimization problem, and thus can be used as stopping criterion for iterative algorithms.

Clearly, the primal gap $g(y)$ is always non-positive (simply take $z = y$). However, the reader should be warned that several authors define the

primal gap as $-g(y)$, ensuring that it is always non-negative. We prefer our definition, as we find it closer to the variational inequality problem definition.

When Y is compact, y^* is a solution of $VI(F,Y)$ if and only if $g(y^*) = 0$, so that $VI(F,Y)$ is equivalent to the maxmin problem

$$\max_{y \in Y} g(y) \equiv \max_{y \in Y} \min_{z \in Y} F(y)^t(z-y) \tag{2.15}$$

This is almost always a difficult problem, with $g(y)$ usually non-concave and nondifferentiable.

Under the assumptions of Theorem 5, we define a ***dual gap function*** as:

$$g_d(y) = \inf_{z \in Y} F(z)^t(z-y)$$

Once again, $g_d(y) \leq 0 \quad \forall y \in Y$ and $g_d(y^*) = 0$ if and only if y^* solves $VI(F,Y)$, by Theorem 5. One can therefore consider the optimization problem reformulation of $VI(F,Y)$

$$\max_{y \in Y} g_d(y) \equiv \max_{y \in Y} \min_{z \in Y} F(z)^t(z-y). \tag{2.16}$$

Although this is a concave maximization problem, $g_d(y)$ being a minimum of affine functions, (2.16) is not necessarily easier than (2.15), as the evaluation of $g_d(y)$ involves the solution of a nonconvex optimization problem.

2.5 Algorithms for variational inequalities

We give in this section a broad overview of algorithms for variational inequalities. It is clearly impossible to cover here the dozens of existing algorithms and their variants; our goal is to describe the main classes of methods and their characteristics. VIP algorithms can be divided into four main groups, according to the formulation of the problem, as in Section 2.4:

1. **Fixed-point problems:** this class contains such classical approaches as the projection method and the Newton method for variational inequalities.

2. **Systems of equations:** has received much attention in the past years, especially the nonsmooth systems that use the B-differentiability.

3. **Optimization formulations:** constrained and unconstrained optimization equivalents of VIPs, to be solved by established nonlinear programming tools.

4. **Convex feasibility formulations:** also the subject of renewed interest. The algorithms presented in this thesis fall in this category.

One class of methods not included here is that of interior point methods; the area is very new, the papers are few, and although these methods are full of promises, little has been done in terms of numerical testing. These methods typically require the differentiability of the mapping. See the algorithms of Nesterov and Nemirovskii [80] and Tseng [104], as well as those of Sharifi-Mokhtarian and Goffin [96], Ralph and Wright [88], and Wu [107].

We also do not cover in this review the large literature on algorithms for specialized variational inequalities such as the LCP, the NCP and the box-constrained VI. Note however that several VIPs *can* be solved as such complementarity problems by use of the results of Section 2.3.

2.5.1 Fixed-point algorithms

We have described in Section 2.4.1 the fixed-point formulation:

$$\text{Find a point } y^* \text{ such that } y^* \in M(y^*)$$

where

$$M(y) = \{z^* \in Y \mid \tilde{F}(z^*, y)^t(z - z^*) \geq 0 \quad \forall z \in Y\}$$

Fixed-point algorithms are basically fixed-point iterations $y_{k+1} \in M(y_k)$. Finding a point in $M(y_k)$ for some y_k is called a **subproblem**. It remains to decide on $\tilde{F}$; clearly, interesting choices are ones for which $M(y_k)$ is single-valued and relatively easy to compute. Many options involve an affine mapping $\tilde{F}(z,y)$:

• $\tilde{F}(z,y) = F(y) + \nabla F(y)(z-y)$	Newton
• $\tilde{F}(z,y) = F(y) + \frac{1}{2}(\nabla F(y) + \nabla F(y)^t)(z-y)$	Symmetrized Newton
• $\tilde{F}(z,y) = F(y) + Q(y)(z-y)$	Quasi-Newton
• $\tilde{F}(z,y) = F(y) + M(z-y)$	Projection
• $\tilde{F}(z,y) = F(y) + diag(\nabla F(y))(z-y)$	Linearized Jacobi

where

- $\nabla F(y)$ is the Jacobian matrix
- $Q(y)$ is a symmetric, positive definite approximation of the Jacobian
- M is some constant symmetric, positive-definite matrix
- *diag* is an operator that keeps only the diagonal of a matrix

In some cases, the subproblems obtained with such linear mappings are easier to solve than the original problem: when $Y = \mathbb{R}^+$, the subproblem is an LCP; when $Y = \mathbb{R}$, the solution of the subproblem is simply the solution of $\tilde{F}(z,y) = 0$, which is a linear system of equations.

Newton's method is probably the most powerful of the above, given its quadratic local convergence rate, under appropriate assumptions. Unfortunately, the subproblems it generates are usually difficult to solve, unless $Y = \mathbb{R}$ or $Y = \mathbb{R}^+$. The last four suggestions above are *symmetric* maps, i.e. their Jacobian is symmetric; this entails that the subproblems can be formulated as nonlinear optimization problems, and solved with existing software. This is the justification behind the symmetrized Newton method. The

Quasi-Newton method replaces the Jacobian evaluation, which may be expensive or difficult, by rank-one or rank-two updates inspired from nonlinear programming. The linearized Jacobi method is a first-order approximation of the straightforward extension of the Jacobi method for systems of equations. The projection method was discussed in more detail above. Note that the Quasi-Newton and Projection methods will yield *convex* optimization subproblems, due to the positive definiteness of their matrices; if furthermore Y is polyhedral, then the subproblems are linearly-constrained, quadratic optimization problems.

Also recall that when $Y = \mathbb{R}^m$ or $Y = \mathbb{R}^m_+$, then the subproblems are respectively linear systems of equations and linear complementarity problems, thus much easier to solve than in the general case.

Nonlinear mappings $\tilde{F}(z, y)$ have also been suggested, for example:

- $\tilde{F}_i(z, y) = F(y_1, \ldots, y_{i-1}, z_i, y_{i+1}, \ldots, y_m)$ Jacobi method
- $\tilde{F}_i(z, y) = F(z_1, \ldots, z_{i-1}, z_i, y_{i+1}, \ldots, y_m)$ Gauss-Seidel method.

Being separable, the corresponding subproblems can be solved as nonlinear optimization problems. Note that the Jacobi method is also sometimes called diagonalization or relaxation method.

We finally outline two variations of the projection method, that are probably the most appropriate competitors of ACCPM-VI, and that will appear again in the Numerical Results section of Chapter 4. Although they do not fit the general iterative scheme of Dafermos (see section 2.4.1), they are in essence fixed-point methods.

The **extra-gradient algorithm** of Korpelevitch ([55]; see [8], [53] and [68]) is based on projections, but converges under weaker assumptions than the projection method, see the discussion of convergence below. It uses two

projections at each iteration; under the Euclidian norm, it is defined by:

$$\begin{aligned} \bar{y} &= \Pi_Y(y_k - \alpha F(y_k)) \\ y_{k+1} &= \Pi_Y\big(y_k - \alpha F(\bar{y})\big) \end{aligned}$$

where α is some positive stepsize. Unfortunately, it has a slow rate of convergence, linear at best. Recently, Solodov and Tseng [98] suggested a modified projection algorithm with the same convergence property as the extragradient method, but which uses only one projection per iteration. The iteration, with a symmetric positive definite matrix Q and a positive stepsize γ, is:

$$\begin{aligned} \bar{y} &= \Pi_Y\big(y_k - \alpha F(y_k)\big) \\ y_{k+1} &= y_k - \gamma Q^{-1}\big(y_k - \alpha F(y_k) - \bar{y} + \alpha F(\bar{y})\big) \end{aligned}$$

where $\alpha \in (0, \infty)$ is chosen so that the mapping $I - \alpha F$ is strongly monotone. The authors indicate that the matrix Q can be used as a scaling parameter to improve the rate of convergence.

We note that the above iterative methods for $VI(F, Y)$ could be used in conjunction with techniques of **approximation of the set** Y. *Inner approximations* of a polyhedral, compact Y have been suggested by Lawphongpanich and Hearn [57]. At each iteration, one deals with the convex hull of a (small) subset of the extreme points of Y; for large-scale problems, this can considerably simplify the solution of the subproblems. *Outer approximations* have been suggested by Fukushima [29]; while the convexity of Y is still required, this approach is applicable to sets that are not polyhedral. Again, the advantage is that of handling a set that approximates Y while being simpler.

Convergence The projection method is globally convergent, but requires the strong monotonicity of F (technically, one even needs to know or have a bound on the constant of strong monotonicity). The plain Newton's

method has a very desirable quadratic convergence rate, but is only locally convergent; furthermore, the condition of this convergence is basically the strict monotonicity of F. Quasi-Newton and other symmetric methods are attractive because of the relative ease of their subproblems, but the assumptions on F required for local convergence are even stronger than for the Newton method, and their convergence rates are slower. Details on the convergence of linear approximation methods are given in [41], [43], [65] and [66]. The Jacobi and Gauss-Seidel methods have similar local convergence results as the linear methods: some type of diagonal dominance of the Jacobian of F is required. Note also that with the exception of the projection method, convergence proofs rely on the differentiability of F.

A globally convergent method to improve the local character of the iterative methods described above was first devised by Marcotte and Dussault [70]. Akin to the techniques of nonlinear optimization, a linesearch guarantees convergence even when the initial iterate is far from the solution. The authors proved that the (primal) gap function can be used to guide the linesearch process in the Newton method for VIP. Their algorithm is globally convergent under the assumption of continuous differentiability and monotonicity of F, and compactness and convexity of Y. Under stronger hypotheses, their algorithm also achieves the quadratic rate of convergence of the standard Newton's method, at least locally. Clearly, their method still requires the solution of a (simpler) variational inequality problem, at each iteration.

The extra-gradient method and its variant by Solodov and Tseng converge under the monotonicity of the mapping F. Furthermore, its differentiability is not required.

2.5.2 Systems of equations

We consider in this section two classes of algorithms that are based on the solution of systems of equations. We first discuss the popular nonsmooth equations that use the projection operators discussed in Section 2.4.2. We then briefly introduce the so-called continuation methods, that use the KKT conditions of Section 2.3.

As shown in Section 2.4.2, variational inequality problems can be formulated as systems of nonsmooth equations. The goal of such formulations is to use the powerful damped Newton method for systems of equations; however, the lack of F(réchet)-differentiability demands that the usual techniques be adapted. References on the subject are numerous, but the seminal articles by Pang ([83], [84]), Pang and Qi [87] and Xiao and Harker ([109], [110]) provide a good starting point. See also the recent Facchinei, Fischer, Kanzow [23] and the references to PATH, NE/SQP and the likes, below.

It would be difficult to present nonsmooth, equation-based algorithms, without introducing first a fair amount of nonsmooth analysis theory. However, it is not our purpose here to discuss these algorithms in details, and we shall be content with a summary.

The Bouligand derivative is defined as follows:

Definition 9 ([83]) *A function H: $\mathbb{R}^n \to \mathbb{R}^m$ is said to be* B-differentiable *at a point y if there exists a function $BH(y)$: $\mathbb{R}^n \to \mathbb{R}^m$, called the* B-derivative *of H at y, which is positively homogeneous of degree 1, such that*

$$\lim_{z \to 0} \frac{H(y+z) - H(y) - BH(y)(z)}{\|z\|} = 0.$$

(In the notation $BH(y)(z)$ ([4]), z is the variable of the function and y is a parameter; we prefer it to $BH(y)z$, sometimes seen, which we find somewhat unclear). The main difference between Bouligand- and Fréchet-differentiability

is the fact that $BH(y)(z)$ needs not be a linear function of z in the former case, while it must be linear in the latter case.

Also, for a mapping $H : \mathbb{R}^m \to \mathbb{R}^m$, let us write $H'(y;d)$ to represent the vector of *directional derivatives* in the direction d:

$$H_i'(y;d) = \lim_{\lambda \to 0+} \frac{H_i(y+\lambda d) - H_i(y)}{\lambda}.$$

The directional derivative is very closely related to the B-derivative. In fact, Shapiro [94] proved that a Lipschitz continuous mapping $H : \mathbb{R}^m \to \mathbb{R}^m$ is B(ouligand)-differentiable if and only if its directional derivatives exist. Furthermore, in this case, the B-derivative of the mapping H at y is identical to the vector of directional derivatives corresponding to the m components H_i of H:

$$H'(y,d) = BH(y)(d)$$

As most authors do, we assume the Lipschitz continuity of H throughout this section.

The principle behind nonsmooth Newton methods is to solve, for the nonsmooth but B-differentiable mapping H, the equation

$$H(y) = 0$$

with an adapted Newton method that uses the B-derivative instead of the usual Jacobian. One solves, for $d_k \in \mathbb{R}^m$, the system

$$H(y_k) + BH(y_k)(d_k) = 0,$$

which is equivalent here to

$$H(y_k) + H'(y_k, d_k) = 0,$$

and sets $y_{k+1} = y_k + d_k$. Like the usual Newton method, this approach will only converge locally; global convergence is achieved by a linesearch along the

direction d_k. Appropriate differentiable merit functions have been developed for that purpose.

Newton-type, nonsmooth-equations methods form a powerful approach to the solution of variational inequalities $VI(F, Y)$. They have good rates of convergence, and are independent of any monotonicity property of F. Indeed, most of the recent algorithm implementations for complementarity problems have used this approach: PATH [20], NE/SQP [86], SEMISMOOTH [17], and their variants [9, 10]. However, these methods rely explicitly on the (F-)differentiability of both the mapping F and the (explicitly known) functions defining the set Y. In that sense, they do not belong in the same class of algorithms as the ones we introduce in this thesis.

Having discussed equation-based methods that use projection operators, let us now turn to continuation methods, which use the KKT system

$$\begin{aligned} F(y^*) - \nabla G(y^*)\pi - \nabla H(y^*)\mu &= 0 \\ H(y^*) &= 0 \\ \pi^t G(y^*) &= 0 \\ \pi &\geq 0 \\ G(y^*) &\geq 0 \end{aligned}$$

that was given in Theorem 2. Upon the introduction of positive parameters ϵ and ν, we obtain the perturbed system

$$\begin{aligned} F(y) + \epsilon y - \nabla G(y)\pi - \nabla H(y)\mu &= 0 \\ H(y) &= 0 \\ \pi^t G(y) &= \nu \\ \pi &\geq 0 \\ G(y) &\geq 0, \end{aligned}$$

which is solved for a sequence of decreasing values of ϵ and ν. (These equa-

tions are of course reminiscent of interior point methods in linear optimization, and continuation methods are often given this epithet). Depending on the exact algorithm, there may be more parameters than the two above; the systems, usually solved with Newton steps, may or may not preserve the nonnegativity constraints; and the constraint $\pi^t G(y)$ may be treated with a function like the Fischer function [26], for the sake of gaining some smoothness property. The Jacobians that occur through the Newton steps are nonsingular under a monotonicity condition on F. Again, this class of algorithms requires the differentiability of F and the functions that define Y in $VI(F, Y)$; as such, they are not direct competitors of our method.

Note that continuation methods are not only equation-based: for example, some continuation methods consist in replacing F by $F + \alpha I$, where I is the identity map, and $\alpha > 0$ the parameter. These methods have also been called *proximal* methods. See [43] for references and details.

2.5.3 Optimization problems

We gave in Section 2.4.4 two formulations of the VIP as optimization problems. From an algorithmic point of view, these formulations are not very attractive: in one case, a nonconvex, nondifferentiable optimization problem must be solved, and in the other case, the simple evaluation of the function is a nonconvex optimization problem.

For both theoretical and algorithmic purposes, the reformulation of VIPs as optimization problems has been a topic of research. Clearly, the equivalence between the general, asymmetric VIP and an "easy" (convex, differentiable, etc.) optimization problem is appealing.

Fukushima [30] first showed that the VIP is equivalent to a *differentiable*, constrained optimization problem. He used a *regularized gap function* (com-

pare with Definition 8)

$$g(y) = \inf_{z \in Y} (F(y)^t(z-y) + \alpha \|z-y\|^2)$$

where $\alpha > 0$. More precisely, the optimization problem is differentiable whenever F is. Unfortunately, this does not take care, in general, of the non-convexity of the optimization problem. However, under the differentiability of F and the positive definiteness of ∇F everywhere in Y, all stationary points of the optimization problem are solutions of the VIP. The author also suggests a descent direction for the optimization problem that does not use the Jacobian ∇F but rather a projection on Y. This is especially interesting when the Jacobian is difficult to evaluate. This algorithm is actually very closely related to the projection algorithm described in Section 2.5.1.

Recently, Yamashita and Fukushima [111] presented an *unconstrained* optimization formulation of the variational inequality problem. The conditions under which it can be considered for practical use are the same as those given in [30], which essentially boil down to differentiability and strong monotonicity of F.

Other articles dealing with the optimization formulation and the use of gap functions include [56], [108], [48] and [100].

2.5.4 Convex feasibility problems

The idea of using the convex feasibility problem reformulation (Section 2.4.3)

$$\text{Find a point } y^* \in Y \text{ such that } \quad F(y)^t(y-y^*) \geq 0 \quad \forall y \in Y$$

to solve variational inequalities is at least as old as the papers of Zukhovitskii, Polyak and Primak [114] and Auslender [4, pp.155–157]. More recently, Lüthi ([63], 1985) also used this formulation. The author's approach, similar to the one described in this thesis, consists of improving an approximation of the

solution set of VIP by cutting planes generation: at current point y_k, the half-space

$$\{y \in \mathbb{R}^n \mid F(y_k)^t(y_k - y) \geq 0\}$$

is added to the current (outer) approximation of the solution set, and a next iteration point y_{k+1} picked within this new approximation. Lüthi picks this new point as the center of a circumscribing ellipsoid, whereas we use an analytic center approach. His method is closely related to the ellipsoid method of linear programming.

Recently, Lemaréchal, Nemirovskii, Nesterov [60] and Nesterov, Vial [81] have introduced algorithms that are also based on the convex feasibility formulation but that further use the dual gap function

$$g_d(y) = \inf_{z \in Y} F(z)^t(z - y).$$

The concavity of $g_d(y)$ allows a treatment by nondifferentiable optimization techniques. In [60], the authors adapt to the VIP a variant of the typical bundle method for nondifferentiable problems.

The authors of [81] describe a clever and sophisticated algorithm whose roots are in the Analytic Center Cutting Plane Method (ACCPM) of Goffin and Vial. The VIP is embedded in a homogeneous projective space, to take advantage of the theory of self-concordant functions [80] while a proximal term ensures that the analytic centers will not diverge to infinity. Also, a sequence of weighted sums of previous analytic centers is generated, in parallel to the sequence of analytic centers; while the analytic centers may not converge to a solution of $VIP(F, Y)$ for F pseudo-monotone, it is proved that the sequence of weighted sums does. This concept of double sequence has been used advantageously in our algorithms; see Section 3.3.4. However, none of the above papers report any numerical results for variational problems.

We also mention the article of Magnanti and Perakis [66], where the authors develop a general framework for the analysis of algorithms based

on the convex feasibility formulation. Under the assumption of strong-f-monotonicity (i.e. co-coercivity), the complexity of four algorithms is analyzed, using a volume-reduction argument.

Finally, let us note the article of Goffin, Marcotte and Zhu [35] which is the starting point of this thesis; we simply mention that it belongs to this class of convex feasibility algorithms, as the theory behind it will be covered in detail in the next chapters.

Before closing this section, the attention of the reader should be directed on two important points. First, the above algorithms, based on the convex feasibility reformulation, apply only to problems that are at least pseudo-monotone; under any weaker assumption on F, the equivalence between the VIP and its convex feasibility formulation does not hold. This, of course, restricts their domain of application, although a fair proportion of VI problems do exhibit a minimal amount of monotonicity (i.e. at least pseudo-monotonicity). Second, algorithms based on the convex feasibility reformulation typically do **not** rely on derivative information (the Jacobian ∇F), in opposition to the majority of the recent algorithms for variational inequalities. This can be a tremendous advantage when the Jacobian is difficult or impossible to evaluate. It also allows the definition (and solution) of variational inequality problems with point-to-set mappings, in which cases a Jacobian is at best difficult to define. This last topic is the subject of the next section.

2.6 Extensions to point-to-set mappings

The extension of "traditional" finite-dimensional variational inequalities to cases with a point-to-set mapping F has been discussed by Auslender [4] and Rockafellar [92]. Such problems must clearly be brought into play when

one considers the VIP corresponding to the optimization of a nonsmooth function (in which case $F(y)$, for some y, is a subdifferential), or cases where $F(y)$ is only known implicitly, for example as the (non unique) solution of a nonlinear programming problem. Let us then make the following definitions.

Definition 10 ([92]) *Let $\mathcal{F}$ be a point-to-set map from $\mathbb{R}^m$ into $2^{\mathbb{R}^m}$. Let Y be a nonempty convex subset of $\mathbb{R}^m$. Then a solution to the* variational inequality problem, *denoted as $VI(\mathcal{F}, Y)$, is a point $y^* \in Y$ such that*

$$F(y^*)^t(y - y^*) \geq 0 \text{ for some } F(y^*) \in \mathcal{F}(y^*) \text{ and } \forall y \in Y.$$

Definition 11 ([92]) *A point-to-set mapping $\mathcal{F} : \mathbb{R}^m \to 2^{\mathbb{R}^m}$ is* monotone *over a set Y if*

$$(F(y_1) - F(y_2))^t(y_1 - y_2) \geq 0$$
$$\forall F(y_1) \in \mathcal{F}(y_1),\ \forall F(y_2) \in \mathcal{F}(y_2),\ \forall y_1, y_2 \in Y.$$

It is maximal monotone *if the graph of $\mathcal{F}$, defined as*

$$G(\mathcal{F}) = \{(y, t) : y \in \text{dom } Y, t \in \mathcal{F}(y)\}$$

is maximal, i.e. for any monotone map $\mathcal{F}'$, $G(\mathcal{F}) \subset G(\mathcal{F}')$ implies $G(\mathcal{F}) = G(\mathcal{F}')$

A common example of a maximal monotone mapping is the subdifferential of a convex function. Lemaréchal, Nemirovskii and Nesterov [59] have extended the concept of convex feasibility problem formulation to such VIPs.

Definition 12 ([59]) *Let $\mathcal{F}$ be a point-to-set map from $\mathbb{R}^m$ into $2^{\mathbb{R}^m}$. Let Y be a nonempty convex subset of $\mathbb{R}^m$. Then a* weak solution *to the variational inequality problem is a point $y^* \in Y$ such that*

$$F(y)^t(y - y^*) \geq 0, \quad \forall y \in Y \quad \text{and} \quad \forall F(y) \in \mathcal{F}(y).$$

The following theorem, also from [59], relates weak solutions to solutions, in the monotone case.

Theorem 6 *Let Y be a nonempty, closed, convex subset of $\mathbb{R}^m$, with nonempty interior, and let $\mathcal{F}$ be a monotone mapping with domain $dom(\mathcal{F})$, $int(Y) \subset dom(\mathcal{F}) \subset Y$, then*

1. *Every solution of the variational inequality is also a weak solution;*
2. *Conversely, provided either:*
   - *$Y \subset dom(\mathcal{F})$ and $\mathcal{F}$ is single-valued continuous, or*
   - *$\mathcal{F}$ is maximal monotone.*

then every weak solution solves the variational inequality.

The theorem above justifies the formulation of the solution set Y^* as the intersection of an infinite number of half-spaces:

$$Y^* = \{\, y^* \in Y \mid F(y)^t(y - y^*) \geq 0, \quad \forall\, y \in Y, \quad \forall\, F(y^*) \in \mathcal{F}(y^*) \,\}$$

This formulation is the basis of our *cutting-plane* approach.

Finally, let us define a primal gap function for the variational inequality with multi-valued mapping.

Definition 13 *The* gap function *g associated with $VI(\mathcal{F}, Y)$ is defined as*

$$g(y) = \inf_{z \in Y, F(y) \in \mathcal{F}(y)} F(y)^t(z - y).$$

Chapter 3

A Linear Cut Algorithm

3.1 Introduction

We describe in this chapter an algorithm for the variational inequality problem $VI(\mathcal{F}, Y)$ over a convex set Y. The algorithm is defined for mappings $\mathcal{F}$ that are either pseudo-monotone single-valued or maximal monotone multi-valued (point-to-set mappings). No assumption is made on the differentiability of single-valued mappings, a useful feature in cases where the jacobian is expensive or impossible to evaluate; see e.g. the MMMR example below (for multi-valued mappings, differentiability is hardly even defined). The set Y may include linear equalities, which are treated explicitly by our method; it may alternatively be implicitly defined by a separating oracle.

Our method is based on the convex feasibility problem reformulation of the VI (see Section 2.4.3), extending the work of Goffin, Marcotte and Zhu [35]. It is fundamentally an Analytic Center Cutting Plane Method (ACCPM); in this respect a sequence of analytic centers is built as the iterations progress. However, there are VIs with simply monotone mappings for which the sequence does not converge to a solution: we introduce a second sequence of points, based on the analytic centers, for which we observe stronger convergence properties.

Cutting planes were used to solve variational inequalities by Lemaréchal, Nemirovskii, Nesterov [60] and Lüthi [63]; see also the integrative framework of Magnanti, Perakis [66]. Within this cutting plane approach, the Analytic Center Cutting Plane method, which includes the algorithm presented in this thesis, has been used by Goffin, Marcotte, Zhu [35] and Nesterov, Vial [81].

The chapter is divided as follows. We review in Section 3.2 the convex feasibility formulation and extend the definition of analytic center to cases with linear equality constraints. We describe in Section 3.3 the algorithm in detail. In Section 3.4 we present some technical points concerning our MATLAB implementation of the method, and conclude in Section 3.5 with several numerical examples that range from typical economical equilibria to a pollution permits market model and to financial options pricing.

3.2 Basic concepts

In this section we recall some fundamental results about the convex feasibility formulation of variational inequalities, we present a generic algorithm for solving such problems, and we also extend the concept of analytic center to sets that include equality constraints.

3.2.1 The Convex feasibility formulation

The basic definitions and results on variational inequalities were covered in the last chapter. Let us however recall here two theorems that are the cornerstones of the convex feasibility reformulation. First, from Section 2.4.3:

Theorem 7 *Let Y be a nonempty, closed, convex subset of $\mathbb{R}^m$ and let F be a continuous, pseudo-monotone mapping from Y into $\mathbb{R}^m$. Then y^* solves*

the $VI(F,Y)$ if and only if $y^* \in Y$ and

$$F(y)^t(y - y^*) \geq 0 \quad \forall y \in Y \tag{3.1}$$

Second, Theorem 6 of Section 2.6 implies:

Theorem 8 *Let Y be a nonempty, closed, convex subset of $\mathbb{R}^m$, with nonempty interior, and let $\mathcal{F}$ be a point-to-set, maximal monotone mapping with domain $dom(\mathcal{F})$, $int(Y) \subset dom(\mathcal{F}) \subset Y$. Then, for the variational inequality problem $VI(\mathcal{F},Y)$, any weak solution is a solution, and any solution is a weak solution.*

Therefore, under the respective assumptions of the theorems above, the solution set Y^* of the variational inequality problem can be formulated as the intersection of an infinite number of half-spaces. It is then possible to solve $VI(\mathcal{F},Y)$ or $VI(F,Y)$ as a convex feasibility problem, if we let "*y feasible*" correspond in our case to "$y \in Y^*$". A generic algorithm to find a point $y^* \in Y^*$ would be as follows:

Step 0: Set $k = 0$, $Y_0 = Y$
Step 1: Pick a point $y_k \in \text{int}(Y_k)$
Step 2: Check if $y_k \in Y^*$; if yes then stop.
Step 3: Pick some $F(y_k) \in \mathcal{F}(y_k)$
Set $Y_{k+1} := Y_k \cap \{y : F(y_k)^t y_k \geq F(y_k)^t y\}$
Set $k := k + 1$
Return to step 1

where the first operation of Step 3 is skipped for a variational inequality with a single-valued mapping. An iteration is depicted in Figure 3.1.

Note that the set Y could be defined implicitly by a separation oracle, such that a *feasibility cut* is returned instead of a "VI cut" whenever $y_k \notin Y$. Note also that in practice, algorithms are limited by the finite precision of

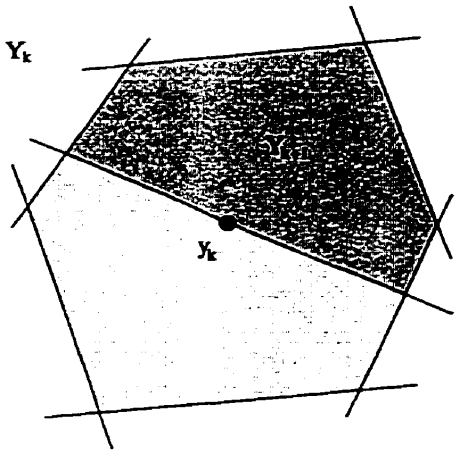


Figure 3.1: A linear cut through a point y_k of Y_k

the computers, so that the following definition of an approximate solution will be useful; the definition relies on the concept of concept of primal gap $g(y)$ (Section 2.4.4).

Definition 14 *For the variational inequality VI(F,Y), the point* y^* *is an* ϵ–approximate solution *if* $y^* \in Y$ *and*

$$g(y^*) \geq -\epsilon$$

3.2.2 Analytic centers

The analytic center, as originally introduced by G. Sonnevend [99] and discussed in the previous chapter, pertains to a polyhedron defined by a set of linear inequalities. However, the extension to sets with both linear inequalities and equalities is possible:

Definition 15 *Consider the set*

$$Y = \{\, y \mid A^t y \leq c,\ By = d \,\}$$

and the associated dual potential function

$$\varphi_D(y) = \sum_i \ln(c_i - A_i^t y)$$

where the index i is used to denote the components of c and the rows of A^t.

The analytic center y^c *of* Y *is defined as the point maximizing the dual potential function over the set* $\bar{Y} = \{ y \mid A^t y < c,\ By = d \}$*:*

$$y^c = \arg\max_{y \in \bar{Y}} \varphi_D(y)$$

It is well known that if Y is bounded, the center y^c is unique. Writing out the first-order optimality conditions for the equivalent mathematical program

$$\begin{array}{ll} \text{maximize}_{y,s} & \sum_i \ln s_i \\ \text{subject to} & A^t y + s = c \\ & By = d \\ & s > 0 \end{array}$$

we obtain

$$Ax + B^t \mu = 0 \tag{3.2}$$

$$A^t y + s = c \tag{3.3}$$

$$By = d \tag{3.4}$$

$$Xs = e \tag{3.5}$$

$$x, s > 0 \tag{3.6}$$

where x and μ are the variables associated with, respectively, the inequality and the equality constraints. We follow the convention to call equation (3.2) the primal feasibility conditions, (3.3) and (3.4) the dual feasibility conditions, and (3.5) the centrality conditions; according to this notation, the analytic center lies in the dual space.

3.3 An Analytic center cutting plane algorithm

We now describe our algorithm. For the sake of clarity, we assume that the mapping is single-valued, and we point out, whenever appropriate, the

modifications that a multi-valued mapping would require. (The basic theory of variational inequality with multi-valued mappings was discussed in Section 2.6). We also make the assumption of a *polyhedral* Y; as mentioned earlier, a general convex Y can be treated by outer approximation (feasibility cuts).

Consider the problem VI(F,Y):

$$\begin{array}{ll} \text{Find } y^* & \text{such that } F(y^*)^t(y-y^*) \geq 0 \quad \forall y \in Y \\ & \text{where } Y = \{\, y \mid A^t y \leq c,\ By = d \,\} \end{array}$$

where $F : \mathbb{R}^m \rightarrow \mathbb{R}^m$ is pseudo-monotone (maximal monotone for point-to-set mappings), A is a real $m \times n$ matrix and B is a real $p \times m$ matrix with $p \leq m$. We assume the existence of a bounded solution y^* and that box constraints $l \leq y \leq u$ can be added if Y is unbounded.

Then the following algorithm can be used to find an ϵ–approximate solution of VI(F, Y):

ALGORITHM ACCPM-VI (LINEAR CUTS)

Step 0: **Initialization**
Set $k = 0$, $Y_0 = Y$

Step 1: **Analytic Center**
Find an approximate analytic center y_k of Y_k.

Step 2: **Termination Criterion**
Compute the convex combination of centers $\bar{y}$ as per Section 3.3.4
Compute the primal gap $g(\bar{y}, Y)$
If $g(\bar{y}, Y) \geq -\epsilon$ then stop

Step 3: **New Cut**
Compute cut $F(y_k)^t y_k \geq F(y_k)^t y$
Set $Y_{k+1} := Y_k \cap \{y : F(y_k)^t y_k \geq F(y_k)^t y\}$
$k := k + 1$
Return to step 1

Step 1, computing an analytic center, is done with a primal-dual Newton method which uses the last center as a warm start. Although the gap computation in Step 2 involves solving a linear program, this can usually be done

very quickly, as the size of most VI's is small relative to the size of LPs that can be solved efficiently. Furthermore, it is not crucial to evaluate the gap at every iteration. Gap evaluation can be difficult in the point-to-set mapping case, since $\mathcal{F}(\bar{y})$ is often not explicitly available. For lack of a better concept, the gap is then evaluated at some point $F(y_k) \in \mathcal{F}(\bar{y})$.

Step 3 consists basically of a function evaluation; the relative time spent on this step varies with the application, as will become clear in the section on numerical experiments. For the multi-valued case, any point $F(y_k) \in \mathcal{F}(y_k)$ is picked.

The rest of this section is devoted to a method for finding the approximate analytic center y_{k+1} of Y_{k+1}, starting from y_k.

3.3.1 An infeasible primal-dual Newton step

At iteration k, the approximate analytic center $y_k \in \mathbb{R}^m$ of Y_k is a solution, together with $x_k \in \mathbb{R}^k_+$, $\mu_k \in \mathbb{R}^p$ and $s_k \in \mathbb{R}^k_+$, of the system of equations:

$$\begin{aligned} Ax + B^t\mu &= 0 \\ A^t y + s &= c \\ By &= d \end{aligned}$$

while

$$\delta := e - Xs \quad \neq \quad 0$$

Here A and c would typically contain both *initial* inequality constraints (including box constraints) and *generated* inequality constraints. Initial constraints are the ones that define Y, while generated ones are the cuts from the algorithm. The problem of interest $VI(F, Y)$ lies in the dual space. In what follows, we explicitly recognize the non-centrality of the point y_k in deriving the steps.

The center y_{k+1} of Y_{k+1} is the solution (with x_{k+1}, μ_{k+1} and s_{k+1}) of

$$\begin{aligned} Ax + a\xi + B^t\mu &= 0 && (3.7)\\ A^t y + s &= c && (3.8)\\ a^t y + \sigma &= a^t y_k && (3.9)\\ By &= d && (3.10)\\ Xs &= e && (3.11)\\ \xi\sigma &= 1 && (3.12)\\ x, s, \xi, \sigma &> 0 && (3.13) \end{aligned}$$

where $a := F(y_k)$, $x_{k+1} := [x;\ \xi] \in \mathbb{R}^{k+1}$ and $s_{k+1} := [s;\ \sigma] \in \mathbb{R}^{k+1}$. Here ξ and σ are "new" unidimensional variables corresponding to the new constraint (3.9). The system (3.7)–(3.13) is solved iteratively with Newton's method; the initial point we use is $y = y_k$, $x = x_k$, $s = s_k$ and ξ and σ are two positive numbers that we define later. This initial point will not be feasible in the sense of (3.7)– (3.10), although it will be feasible with respect to positivity.

The Newton steps dy, dx, ds, $d\xi$ and $d\sigma$ from the initial point are defined by the system

$$\begin{aligned} Adx + ad\xi + B^t d\mu &= -Ax_k - a\xi - B^t\mu = -a\xi && (3.14)\\ A^t dy + ds &= -Ay_k - s_k + a^t y_k = 0 && (3.15)\\ a^t dy + d\sigma &= -a^t y_k - \sigma + a^t y_k = -\sigma && (3.16)\\ Bdy &= -By_k + d = 0 && (3.17)\\ S_k dx + X_k ds &= e - X_k s_k =: \delta && (3.18)\\ \sigma d\xi + \xi d\sigma &= 1 - \xi\sigma = 1 - \xi\sigma && (3.19) \end{aligned}$$

The rightmost terms come from the feasibility of (y_k, x_k, s_k) with respect to Y_k. From (3.15) and (3.18),

$$ds = -A^t dy \qquad (3.20)$$

$$dx = S_k^{-1}(\delta - X_k ds) \tag{3.21}$$

$$= S_k^{-1}(\delta + X_k A^t dy) \tag{3.22}$$

Using (3.14) and the above we have

$$Adx = AS_k^{-1}(\delta + X_k A^t dy) = -B^t d\mu - a(\xi + d\xi)$$

from which, using $\Delta := AS_k^{-1}X_k A^t$,

$$dy = \Delta^{-1}(-AS_k^{-1}\delta - B^t d\mu - (\xi + d\xi)a). \tag{3.23}$$

Combining (3.17) and the definition $H := B\Delta^{-1}B^t$, we obtain

$$Bdy = -B\Delta^{-1}B^t d\mu - B\Delta^{-1}(AS_k^{-1}\delta + (\xi + d\xi)a) = 0$$

and

$$d\mu = -H^{-1}B\Delta^{-1}(AS_k^{-1}\delta + (\xi + d\xi)a) \tag{3.24}$$

If we define $\xi_N := \xi + d\xi$, $\sigma_N := \sigma + d\sigma$, $m^2 := a^t\Delta^{-1}B^t H^{-1}B\Delta^{-1}a$ and $r^2 := a^t\Delta^{-1}a$, (3.16) and the expressions for dy and $d\mu$ above yield

$$\begin{aligned} a^t dy &= -a^t\Delta^{-1}(AS_k^{-1}\delta + B^t d\mu + (\xi + d\xi)a) \\ &= a^t\Delta^{-1}B^t H^{-1}B\Delta^{-1}a\xi_N - a^t\Delta^{-1}a\xi_N - a^t\Delta^{-1}AS_k^{-1}\delta \\ &\quad + a^t\Delta^{-1}B^t H^{-1}B\Delta^{-1}AS_k^{-1}\delta \\ &= (m^2 - r^2)\xi_N - a^t\Delta^{-1}(I - B^t H^{-1}B\Delta^{-1})AS_k^{-1}\delta = -\sigma_N \end{aligned}$$

Finally we can write from the above and (3.19)

$$\sigma\xi_N + \xi\sigma_N = \sigma\xi_N + \xi(r^2 - m^2)\xi_N + \xi a^t\Delta^{-1}(I - B^t H^{-1}B\Delta^{-1})AS_k^{-1}\delta = 1 + \xi\sigma$$

and

$$\xi_N = \frac{1 + \xi\sigma - \xi a^t\Delta^{-1}(I - B^t H^{-1}B\Delta^{-1})AS_k^{-1}\delta}{\sigma + \xi(r^2 - m^2)} \tag{3.25}$$

$$\sigma_N = (r^2 - m^2)\xi_N + a^t\Delta^{-1}(I - B^t H^{-1}B\Delta^{-1})AS_k^{-1}\delta \tag{3.26}$$

Because linear equations are solved in one step by Newton's method, both primal and dual feasibilities are regained in just one step. Indeed,

$$A^t(y_k + dy) + s_k + ds = A^t(y_k + dy) + s_k - A^t dy = c$$

$$\begin{aligned} B(y_k + dy) &= By_k - B^t\Delta^{-1}(AS_k^{-1}\delta + B^t d\mu + (\xi + d\xi)a) \\ &= By_k = d \end{aligned}$$

$$\begin{aligned} & A(x_k + dx) + a\xi_N + B^t(\mu_k + d\mu) \\ &= Ax_k + AS_k^{-1}\delta + AS_k^{-1}X_k A^t dy + a\xi_N + B^t(\mu_k + d\mu) \\ &= Ax_k + AS_k^{-1}\delta - \Delta\Delta^{-1}(B^t d\mu + a\xi_N + AS_k^{-1}\delta) + a\xi_N + B^t(\mu_k + d\mu) \\ &= Ax_k + B^t\mu_k \\ &= 0 \end{aligned}$$

and

$$\begin{aligned} & a^t(y_k + dy) + \sigma_N \\ &= a^t y_k + a^t\Delta^{-1}B^t H^{-1}B\Delta^{-1}a(\xi_N + AS_k^{-1}\delta) - a^t\Delta^{-1}AS_k^{-1}\delta \\ &\quad -a^t\Delta^{-1}a\xi_N + (r^2 - m^2)\xi_N + a^t\Delta^{-1}(I - B^t H^{-1}B\Delta^{-1})AS_k^{-1}\delta \\ &= a^t y_k \end{aligned}$$

Each time a new cut is added to the model, it is necessary to give values to the new variables ξ and σ. We use the result of the following lemma for that purpose.

Lemma 3 *Let* $q := a^t\Delta^{-1}(I - B^t H^{-1}B\Delta^{-1})AS_k^{-1}\delta$. *Then the values of* ξ *and* σ

$$\xi = \frac{\sqrt{q^2 + 4(r^2 - m^2)} - q}{2(r^2 - m^2)} \qquad \sigma = \frac{1}{\xi}$$

are the ones that maximize both ξ_N *and* σ_N.

Proof: The maximum of (3.25)

$$\xi_N(\xi,\sigma) = \frac{1+\xi\sigma-\xi q}{\sigma+\xi(r^2-m^2)}$$

is attained when

$$\nabla\xi_N = \begin{bmatrix} \frac{(\sigma-q)(\sigma+\xi(r^2-m^2))-(r^2-m^2)(1+\xi\sigma-\xi q)}{(\sigma+\xi(r^2-m^2))^2} \\ \frac{\xi(\sigma+\xi(r^2-m^2))-(1+\xi\sigma-\xi q)}{(\sigma+\xi(r^2-m^2))^2} \end{bmatrix} = 0$$

which implies the result above. Since σ_N is a non-negative multiple of ξ_N, it is also maximized by the given values of ξ and σ. □

Level of centering For the convergence of the algorithm, it is important that this first Newton step does not yield a point that is too much off center. This concept was precisely quantified in [36] and is adapted here for completeness. For the sake of brevity, we give the result under the assumption that the steps are taken with $\delta = 0$ (this is *not* an at-large assumption of exact centering of the last center: only when defining the steps do we assume $\delta = 0$); as the above authors showed in Section 6 of their article, handling the case $\delta \neq 0$ is a straightforward matter. Let us rewrite, using (3.20–3.25), the steps dx and ds as

$$\begin{aligned} ds &= \xi_N S_k^{1/2} X_k^{-1/2} p \\ dx &= \xi_N S_k^{-1/2} X_k^{1/2} p \end{aligned}$$

where

$$p = S_k^{-1/2} X_k^{1/2} A^t \Delta^{-1/2} (I - \Delta^{-1/2} B^t H^{-1} B \Delta^{-1/2}) \Delta^{-1/2} a$$

Using the fact that $I - \Delta^{-1/2} B^t H^{-1} B \Delta^{-1/2}$ is a projection matrix, one can easily compute that $p^t p = r^2 - m^2$. Let us define $\eta := \|e - X_k s_k\|$. Then, with a proof almost identical to that of Theorem 3.1 of [36], we can see that if the values of ξ and σ are chosen as

$$\xi = \frac{\beta}{\sqrt{r^2-m^2}(1+\sqrt{1-\beta^2})}$$

$$\sigma = \frac{1}{\xi},$$

where $\beta < \sqrt{1-\eta}$, we have

$$\|X_N s_N - e\| \leq \eta + \sqrt{\beta^4 + (1-\beta^2)^2}$$

The minimum bound is attained for $\beta = 1/\sqrt{2}$ and yields

$$\|X_N s_N - e\| \leq \eta + 1/\sqrt{2} < \eta + .708$$

In practice, we prefer to use the values of ξ and σ given in Lemma 3 which, compared to the values just above, favor positivity over centrality; this because the non-positivity of ξ and σ, for example through computer roundoff, often has disastrous consequences, while the centering is a quite robust procedure.

3.3.2 A centering Newton step

Once a feasible point of Y_{k+1} has been attained, it is necessary to improve its centrality, i.e. to work on equations (3.11) and (3.12). The presentation will be clearer if we write $A := [A\ a]$, $c := [c;\ a^t y_k]$ and redefine the feasible point $(y_N, x_N, \xi_N, s_N, \sigma_N) := (y_k + dy, x_k + dx, \xi + d\xi, s_k + ds, \sigma + d\sigma)$ as (y_N, x_N, s_N) by setting $x_N := [x_N; \xi_N] \in \mathbb{R}^{k+1}$ and $s_N := [s_N; \sigma_N] \in \mathbb{R}^{k+1}$. From the results of the previous section, the point (y_N, x_N, s_N) satisfies the system

$$\begin{aligned} Ax + B^t\mu &= 0 \\ A^t y + s &= c \\ By &= d \\ x, s &> 0 \end{aligned}$$

but not the centrality equations $Xs = e$. With this fact we can write the Newton equations

$$\begin{aligned} Adx + B^t d\mu &= -Ax_N - B^t \mu_N = 0 \\ A^t dy + ds &= -Ay_N - s_N + c = 0 \\ Bdy &= -By_N + d = 0 \\ S_N dx + X_N ds &= e - X_N s_N =: \delta \end{aligned}$$

and the Newton steps

$$\begin{aligned} d\mu &= -H^{-1} B \Delta^{-1} A S_N^{-1} \delta \\ dy &= -\Delta^{-1} B^t d\mu - \Delta^{-1} A S_N^{-1} \delta \\ ds &= -A^t dy \\ dx &= -S_N^{-1} X_N ds + S_N^{-1} \delta \end{aligned}$$

where $\Delta := AS_N^{-1} X_N A^t$ and $H := B\Delta^{-1} B^t$.

Convergence to the center Following the results of [36], we can prove that the centering steps as described above converge quadratically to the center. We first rewrite ds and dx as

$$\begin{aligned} ds &= S_N^{\frac{1}{2}} X_N^{-\frac{1}{2}} p_s \\ dx &= S_N^{-\frac{1}{2}} X_N^{\frac{1}{2}} p_x \end{aligned}$$

where

$$\begin{aligned} p_s &:= P S_N^{-\frac{1}{2}} X_N^{-\frac{1}{2}} \delta, \\ p_x &:= (I - P) S_N^{-\frac{1}{2}} X_N^{-\frac{1}{2}} \delta, \end{aligned}$$

and

$$P := S_N^{-\frac{1}{2}} X_N^{-\frac{1}{2}} A^t \Delta^{-\frac{1}{2}} (I - \Delta^{-\frac{1}{2}} B^t H^{-1} B \Delta^{-\frac{1}{2}}) \Delta^{-\frac{1}{2}} A X_N^{-\frac{1}{2}} S_N^{-\frac{1}{2}}$$

is a projection matrix.

Defining $x_+ := x_N + dx$ and $s_+ := s_N + ds$ we obtain

$$X_+ s_+ = X_N s_N + X_N ds + S_N dx + dx \circ ds$$

where $\circ$ denotes the component-wise product. Define also $\eta := \|X_N s_N - e\|$. From the above and $d_x \circ d_s = p_x \circ p_s$, $\|X_+ s_+ - e\| = \|p_x \circ p_s\|$. Finally, since $p_x + p_s = S_N^{-\frac{1}{2}} X_N^{-\frac{1}{2}} \delta$ and $p_x^t p_s = 0$, Mizuno's [75] Lemma 1 applies and:

$$\begin{aligned}
\|X_+ s_+ - e\| &\leq \frac{\sqrt{2}}{4} \delta^t X_N^{-1} S_N^{-1} \delta \\
&= \frac{\sqrt{2}}{4} \sum_{i=1}^{\text{length}(\delta)} \frac{\delta_i^2}{x_i s_i} \\
&\leq \frac{\sqrt{2}}{4} \sum_{i=1}^{\text{length}(\delta)} \frac{\delta_i^2}{\min_i x_i s_i} \\
&= \frac{\sqrt{2}}{4} \sum_{i=1}^{\text{length}(\delta)} \frac{\delta_i^2}{1-\eta} \\
&\leq \frac{\sqrt{2}}{4} \frac{\eta^2}{1-\eta}
\end{aligned}$$

This shows that the recentering steps will improve $\|X_+ s_+ - e\|$ quadratically as long as $\eta < \frac{4}{4+\sqrt{2}} < 0.739$, since this implies $\frac{\sqrt{2}}{4} \frac{\eta^2}{1-\eta} < \eta$. If $0.739 \leq \eta < 1$, a dual recentering could be used, as suggested in [36].

3.3.3 Convergence of the sequence of analytic centers

Under some hypotheses, Goffin, Marcotte and Zhu [35] proved the convergence of the sequence of analytic centers generated by ACCPM-VI (linear cuts) to a solution of $VI(F, Y)$; they also gave a complexity result. The hypotheses are that the mapping F is pseudo-co-coercive, and that Y is the unit box $0 \leq y \leq e$.

The authors' result extends naturally to the case where equality constraints of the type $y_i = k_i$ are present since such constraints simply fix the

values of the corresponding variables. The problem reduces from m dimensions to $m - p$, where p is the number of equality constraints. In fact, the authors of [35] write that analogous convergence and complexity results for the general $Y = \{ y \mid A^t y \leq c,\ 0 \leq y \leq e \}$ would hold after a *"... complete and easy but tedious rewriting..."* of [33] and again, the extension to an equality-constrained $VI(F, Y)$ is rather straightforward. Indeed, if we consider $Y = \{ y \in \mathbb{R}^m \mid A^t y \leq c,\ 0 \leq y \leq e, By = d \}$ with B a real, full-rank $p \times m$ matrix with $p \leq m$, we can write without loss of generality

$$By = [B_B\ B_R][y^B y^R]^t = B_B y^B + B_R y^R = d$$

where B_B is square $p \times p$ and invertible. Then y^B is uniquely defined in terms of y^R,

$$y^B = B_B^{-1}(d - B_R y^R) \tag{3.27}$$

and a reduction of Y to a set $Y^R \subset \mathbb{R}^{m-p}$ (defined with linear inequalities only), together with an appropriate elliptic norm, leads to convergence and complexity results similar to those of [35].

It is very important to make clear that despite the convergence results above, we do not consider the sequence of analytic centers when looking for a solution to the variational inequality. Details are given in the next section.

3.3.4 Construction of the solution point

Nesterov and Vial [81] show a simple, two-dimensional and monotone variational inequality, for which the sequence of analytic centers converges to a point that is *not* a solution (Note: the mapping is not pseudo-co-coercive). They also show, for a slightly different algorithm, that a sequence of certain convex combinations of previous centers will converge to a solution for any monotone VI problem.

Drawing an analogy between the Nesterov-Vial algorithm and ours, we construct at each iteration k a point $\bar{y}_k$ that is a convex combination of the previous analytic centers. The weights of the combination are the last $k-1$ elements of the primal variable x, or approximately the inverses of the slacks for the cuts generated at each of the previous centers. The equivalence is only approximate because of the inexact centering. These weights are normalized by their sum; $\bar{y}_k$ is then:

$$\bar{y}_k = \sum_{i=1}^{k-1} \frac{x_{(n+i)}}{\sum_{j=1}^{k-1} x_{(n+j)}} y_i$$

Here $x_{(n+i)}$ refers to the $(n+i)^{\text{th}}$ component of x; also, recall that n is the number of initial linear inequality constraints.

The sequence formed by the $\bar{y}_k$ points converges to the solution for the Nesterov-Vial problem. Also, the speed of convergence improves significantly for all problems tested. A full explanation of the behaviour of this sequence is not yet available.

Note that if the gap is computed at each iteration, two function evaluations are necessary per iteration: $F(y_k)$ to generate the cut and $F(\bar{y}_k)$ to obtain the gap value.

3.4 Implementation of the algorithm

The algorithm was implemented using the MATLAB language and environment, version 5.1. MATLAB is a convivial setting for matrix computations and algorithm development. With the computing intensive tasks devoted to compiled subroutines, it can also be competitive with C or Fortran codes, see for example the LIPSOL solver of Zhang [112].

Factorization The update Newton step described in Section 3.3.1 implies the solution of a linear system, as well as the solution of a second smaller

system when equality constraints are present. The same thing is true for the recentering Newton step of Section 3.3.2.

The solution of linear systems is performed by dense Cholesky factorizations. Because of the density of the introduced cuts, the matrices to be factored are dense; however, they have the dimension $m \times m$ of the problem's size, independent of the potentially large number of cuts generated.

When equality constraints are present, one further $p \times p$ matrix is factored for each step. This matrix can become positive semidefinite, but not positive definite with respect to machine-precision, causing the failure of the Cholesky factorization. We apply, when necessary, the inelegant but efficient technique of over-weighting the diagonal elements.

Sparsity and Products The sparsity of the initial constraints, box and others, is exploited whenever possible in the matrix products. This is made easy by MATLAB's sparsity handling capability.

Initial center An initial analytic center is needed to start the algorithm. When only box constraints are involved, this is a trivial task. When more complex constraints are present, the recentering subroutine is applied to a point that is feasible by construction.

Stopping criterion The primal gap function $g(y)$ is used as a stopping criterion. Its evaluation at y_k corresponds to solving the linear optimization problem

$$\min_{z \in Y} F(y_k)^t (z - y_k).$$

We use the CPLEX software to solve this linear problem; given the dimension of the Y's under consideration —up to a few hundred variables—, this step is very rapid.

Cuts removal After the addition of several dozens or hundreds of cuts, it is theoretically possible to remove many of them without hindering the

convergence of the algorithm while improving its speed. However, in practice, these cuts are very difficult to identify. The harsh approach of removing cuts with the greatest slacks can slightly improve the speed of the algorithm although convergence cannot be guaranteed. As a rule, we keep all cuts.

3.5 Numerical Results

We report here on our computational experience with the MATLAB implementation of the algorithm. We include examples both with and without equality constraints. Most problems, except the larger ones, are solved to the precision level $g(y_k) > -10^{-4}$; this is a stringent criterion which yields very good accuracies $\|y_k - y^*\|$. In most examples, initial box constraints must be set before the first iterations. The variables of all the problems happen to be bounded below by 0; the upper bounds in the problems without equalities were set at 1×10^3, and at 1×10^2 for those with equality constraints. This choice has to do with the magnitudes of the solutions' components, not with the presence of equality constraints. Some problems are inherently bounded upwards.

Results are reported as follows: *Center precision* refers to the value of $\|Xs - e\|$ at which a point is considered centered; a value of 0.1 ensures a relatively good centering, while 0.9 is much more "approximate". The *# iterations* is the number of iterations required to meet the primal gap stopping criterion; it is therefore also the number of analytic centers, and of cutting planes, generated for the problem. The *# of centering steps* is the total number of centering steps that were required; this excludes the first step at each iteration, i.e. those steps that we called update steps. As a rule, either one or no centering step at all is required at each iteration. The *cpu* column indicates the seconds or minutes of cpu time that were required for the completion of the algorithm. All runs except one were performed on an

UltraSparc 1 workstation, model 200E, 128 Mb of memory. The CO_2 environmental model was run on the multiprocessor HP machine of the E.T.H. in Zürich.

3.5.1 An environmental model for pricing CO_2 emission permits

This recent extension of the well-known MARKAL model was developed by a group of researchers at the ETH and the Paul Scherrer Institute in Zürich (see [5] and [11]). Their model, called MMMR for Markal Macro Multi-Regional, integrates the energy models of three countries to allow the possibility of the trade of pollution emission permits; a country can thus buy a right to pollute, while the selling country must reduce its pollution by the same amount.

At the international level, the model has a $VI(F, Y)$ in ten variables and one equality constraint; the experience of [5] show that the mapping F is "rather" monotone on Y, and "almost" pseudo-monotone (in the sense that pseudo-monotonicity holds between most pairs of points and monotonicity between many of them). At the national level, a large nonlinear programming MARKAL-MACRO model is solved for each country, yielding a value called *excess demand*. The sum of the excess demands of the countries is nothing but the evaluation of F at a certain point y_k. When the international-level problem is solved solely on the basis of the (national-level) excess demands, the authors of [5], [11] call the approach "aggregated". Should more information (such as derivatives) be available from the national-level problems, the approach would be "disaggregated". This latter approach can, at least theoretically, allow faster solution of the international-level equilibrium problem. On the other hand, the aggregated view requires only a minimal amount of coordination and homogeneity of the national models, and this is a decisive advantage in a practical context (so that the thesis [11] considers in detail

aggregated methods only). Furthermore, in the general case, one cannot expect the NLP's to have unique solutions, and therefore F may actually be a point-to-set mapping $\mathcal{F}$. Clearly, the existence of the jacobian of such a mapping cannot be guaranteed.

Within the aggregated approach, the author of [11] compared a Negishi approach and an ACCPM-based method roughly equivalent to the algorithm presented above. (Note that when using the Negishi method, the international-level problem is not a VI). He concludes that the ACCPM approach is both much easier to implement and manage, and that even without using the accelerating technique of Section 3.3.4, the ACCPM approach is two or three times faster than the Negishi approach, on the instances considered. The ACCPM method was also much faster than a center-of-gravity, cutting-plane method that was tested.

Very few other variational inequality algorithms could be used in this setting: no derivative information, low monotonicity. Two of them are the extra-gradient method, and the projection-based approach of Solodov and Tseng, that were introduced in Section 2.5.1. Some comparison points between the latter method and ACCPM-VI are given in Section 4.6.3.

Also note that a further challenge in solving the international-level equilibrium comes from the fact that simply evaluating F at a certain point can take several minutes on a fast computer.

We do not report cpu times for this example: 100 iterations can take 10 hours and more, but spent almost exclusively on the national level NLP's. The 10-dimensional VI itself, excluding the function evaluation, is solved in 0.01 second or less per iteration.

Results are shown in Table 3.1.

Center precision	# iterations	# of centering steps	gap
0.9	106	0	-1.0e-2
0.1	100	101	-1.2e-2

Table 3.1: Example `mmmr` 10 variables

3.5.2 A Walrasian equilibrium problem: scarf

A *Walrasian* or *General Equilibrium* is attained when there are no goods whose demand exceeds the supply. Scarf ([93]) has described an economy with a number of goods, consumers and producers, that are related through prices variables and production activities variables.

With the definitions

p	prices of the goods (m by 1)
y	activities levels (n by 1)
$d(p)$	consumers demand functions (m by 1)
A	constant technology or activity matrix; translates activity levels into outputs (m by n)
b	initial endowment of goods (m by 1)

the equilibrium is determined by the Nonlinear Complementarity Problem (see [93] and [71] for example)

$$\begin{array}{ccc} A^t p \leq 0 & y \geq 0 & y^t A^t p = 0 \\ b - d(p) + Ay \geq 0 & p \geq 0 & p^t(b - d(p) + Ay) = 0 \end{array}$$

This formulation, which includes both primal and dual conditions on top of the complementarity, is necessary to apply NCP algorithms. It also hides the equivalent $VI(F, Y)$ with

$$F(p) = b - d(p), \quad Y = \{p \mid Ap \leq 0,\ p \geq 0\}.$$

This primal-only formulation involves m variables instead of the NCP's $m+n$, and can be found as a direct consequence of Theorem 2.

Center precision	# iterations	# of centering steps	cpu secs	gap
0.9	127	0	3.3	-0.90e-04
0.1	123	122	4.1	-0.86e-04

Table 3.2: Example `scarf` 14 variables

Center precision	# iterations	# of centering steps	cpu secs	gap
0.9	104	2	2.9	-0.91e-04
0.1	102	101	3.7	-0.87e-04

Table 3.3: Example `scarf, normalised` 14 variables

iteration #	gap	y_1	y_2	y_{12}	y_{13}	y_{14}
1	—	0.0217	0.0239	0.5665	0.0363	0.0304
50	-0.1429	0.0617	0.0583	0.0629	0.0361	0.0930
102	-1.2847e-04	0.062144	0.058334	0.062012	0.036515	0.092786
103	-1.0792e-04	0.062144	0.058334	0.062012	0.036515	0.092785
104	-9.0878e-05	0.062145	0.058334	0.062012	0.036515	0.092785

Table 3.4: Progression in `normalised scarf`

In [93], two Walrasian equilibrium examples are given, the largest one with 14 goods and 26 production activities. The resulting NCP has 40 variables, but the equivalent VI only 14.

Notice a further advantage of our algorithm over most NCP methods. By assumption of the Walrasian model, the demand function d is homogeneous of degree 0. As shown in [71], the jacobian matrix $\nabla d(p)$ is then singular; typical "fixes" are to fix one price, called *numéraire*, or to normalise the prices with an equality constraint. Our approach requires no such fixing; prices can simply be normalised after completion of the algorithm. Of course, it is also possible to use a normalising equation if desired; results of runs with and without normalisation are given in Tables 3.2, 3.3 and 3.4. Figure 3.2 shows, for two levels of recentering precision, the evolution of the gap with respect to the iterations. We display both the gap associated to the analytic centers, and the gap at the weighted average of past centers.

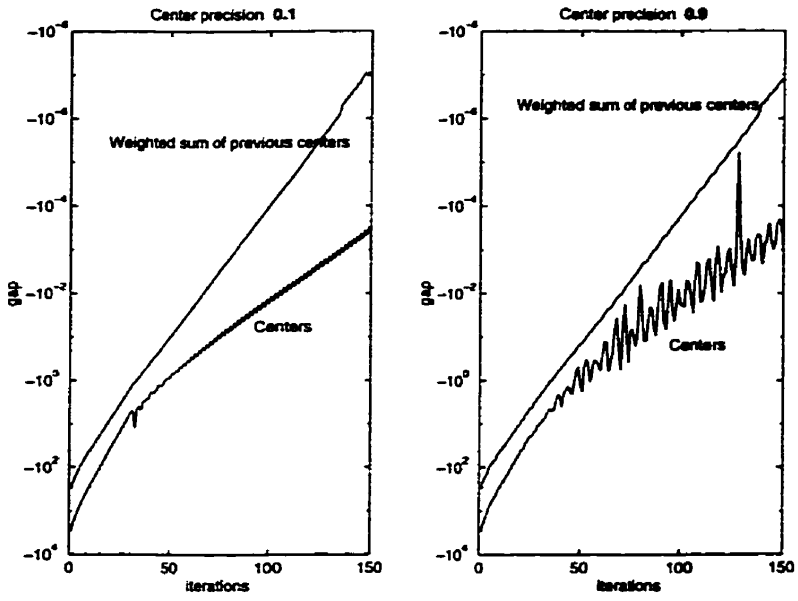


Figure 3.2: Example `scarf`: Gap vs iterations for different centering precisions

3.5.3 An option pricing model in finance

In the area of finance, the pricing of stock options and other derivative products has become of paramount importance in the last two decades.

In [47], the authors reformulated the Black-Scholes model for American type options as an infinite-dimensional VI which, through discretization by finite-differences, is approximated by a finite-dimensional VI. For a simple American option with one underlying asset, the discretization takes place along two axes, the current price of the asset and the time to expire. A time-stepping approach allows the problem to be solved as a sequence of VIs, one for each time step. We solved a problem with the data:

- Strike price: $K = \$10$
- Time span: $T = 6$ intervals of 2 weeks
- Price span: $[\$0, \$20]$ in intervals of $\$0.05$.
- Risk-free interest rate: $r = 3\%$
- Diffusion term: $\sigma = 0.2$

Center precision	# iterations	# of centering steps	cpu minutes	gap
0.9	1378	0	190	0.99e-3
0.1	1399	1398	382	-0.98e-3

Table 3.5: Example `option` 400 variables

We were here only interested in solving a "real-world" large VI, so that we report only on the first of the six 400-variable problems. The reader can see Chapter 4 where similar problems were solved entirely. See also Appendix A for a description of the model.

The resulting $VI(F, Y)$ has 400 variables, and has no other initial constraints than boxes. The mapping F is linear with a non-symmetric jacobian. This problem is highly structured and sparse while its size grows very rapidly with finer discretizations; in a sense, it is very simple and very large. Our purpose here was not to compete with more specifically-adapted algorithms, but to show that our algorithm can solve problems with a few hundred variables. Numerical results are shown in Tables 3.5.

3.5.4 Other problems without linear equality constraints

The problems in this section do not use the capability of the algorithm to handle equality constraints.

The first example is a Nash equilibrium, the second one a spatial price equilibrium problem, the third is a traffic assignment problem and the fourth is an energy model. They are available from MCPLIB [19].

Example `choi`: Choi, DeSarbo and Harker [13] have applied a Stackelberg-Nash equilibrium approach to the pricing of analgesics. The model has 13 variables, and the utility function used is quadratic and is not a gradient mapping, i.e. its jacobian is not symmetric. The results are shown in Table 3.6.

Center precision	# iterations	# of centering steps	cpu secs	gap
0.9	67	0	13	-0.95e-04
0.1	64	66	13	-0.95e-04

Table 3.6: Example `choi` 13 variables

Example `tobin`: The second problem is a well-known spatial price equilibrium model from Tobin [102]. It is a so-called *price* formulation: supply and demand at the different markets are functions of the prices. In opposition to the quantity formulation, this one involves no equality constraints. This model has two interacting commodities on five markets or regions. The transportation cost, demand and supply functions are nonlinear, and the problem is asymmetric, as well as non-monotone at the solution. See Tables 3.7 and 3.8 below. Figure 3.3 shows, for two levels of recentering precision, the evolution of the gap with respect to the iterations. We display both the gap associated to the analytic centers, and the gap at the weighted average of past centers.

Center precision	# iterations	# of centering steps	cpu secs	gap
0.9	608	0	40	-0.2e-04
0.1	592	594	68	-0.2e-04

Table 3.7: Example `tobin` 42 variables

iteration #	gap	y_1	y_9	y_{17}	y_{40}	y_{41}	y_{42}
1	—	500	500	500	500	500	500
250	-113.1288	0.0689	3.9642	12.7263	22.7554	18.7534	29.1753
500	-0.0994	3.7882e-05	3.8827	12.7678	22.7644	18.7604	29.1615
606	-0.0011	1.4303e-06	3.8827	12.7679	22.7644	18.7604	29.1615
607	-0.0010	1.3816e-06	3.8827	12.7679	22.7644	18.7604	29.1615
608	-2.1252e-05	1.3241e-06	3.8827	12.7679	22.7644	18.7604	29.1615

Table 3.8: Progression in `tobin`

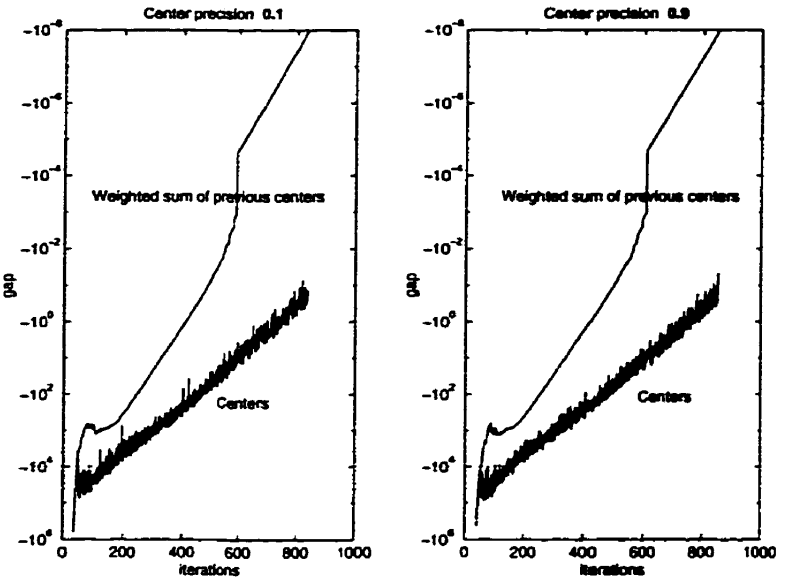


Figure 3.3: Example `tobin`: Gap vs iterations

Examples `gafni` and `bertsekas`: This interesting application to traffic assignment was described in [7]. There are two strictly equivalent formulations of the same problem (see the results in Tables 3.9 and 3.10). The first formulation is a nonlinear complementarity problem (NCP) in 15 variables while the second is a more general mixed complementarity problem (MCP) in 5 variables. The delay functions are quadratic, and the nonsymmetry of the problem is controlled through a parameter γ. While both formulations are just specialized variational inequality problems and can thus be solved as VI's, the lower dimension of the MCP is very favorable to the speed of the algorithm.

Center precision	# iterations	# of centering steps	cpu secs	gap
0.9	197	0	4.9	-0.95e-04
0.1	194	193	6.2	-0.99e-04

Table 3.9: Example `bertsekas` 15 variables

Center precision	# iterations	# of centering steps	cpu secs	gap
0.9	43	0	0.95	-0.89e-4
0.1	41	41	1.0	-0.90e-04

Table 3.10: Example `gafni` 5 variables

Example etamge: This is an MPSGE formulation of the ETA-MACRO model of A. Manne, [67]. The model represents the interactions between the energy sector and the economy of the United States and comprises 114 variables. See the numerical results in Table 3.11 below.

Center precision	# iterations	# of centering steps	cpu minutes	gap
0.9	2354	7	49	-0.99e-04
0.1	2492	1804	85	-0.67e-04

Table 3.11: Example **etamge** 114 variables

3.5.5 Other problems with linear equality constraints

We present computational experience with two spatial price equilibrium problems that are readily available in the literature. The so-called *quantity formulation* of such problems uses inverse demand and supply functions, i.e. price functions respectively in terms of the demand and supply quantities. With this formulation the equilibrium is defined as a variational inequality problem with (linear) conservation of flow constraints (see e.g. [77]). Our algorithm allows us to work directly with this original formulation.

Example nagurney: Our first example is a small problem from A. Nagurney's book [77]. It has 2 supply markets and 2 demand markets; with the shipment quantities on the 4 arcs, it has dimension 8. The transportation costs, supply prices and demand prices are all linear functions. The jacobian of F in the $VI(F, Y)$ formulation is asymmetric, so that no equivalent convex optimization formulation exists. The results are given in Table 3.12.

Center precision	# iterations	# of centering steps	cpu secs	gap
0.9	61	0	1.2	-0.76e-04
0.1	57	57	1.5	-0.78e-04

Table 3.12: Example **nagurney** 8 variables

Example harker: The second equality-constrained example (see [42]) has 5 markets all acting as demand and supply markets for the traded commodity. There are 16 connecting arcs for a total of 26 variables. All functions are polynomial and nonlinear. This problem includes a parameter γ which controls the level of asymmetry of the jacobian of F, with $\gamma = 0$ implying a symmetric matrix. Results are shown for two values of γ, 1 and 15; see Tables 3.13 and 3.14. It is especially interesting to note the influence of asymmetry. Symmetrized methods such as the ones discussed in [42] are, quite naturally, sensitive to deviations from symmetry: there the $\gamma = 15$ case takes twice the time and the iterations of the $\gamma = 1$ case (4 minutes vs 2 minutes, on a pre-1988 computer). Our algorithm seems on the other hand to be unaffected by asymmetry.

Center precision	# iterations	# of centering steps	cpu secs	gap
0.9	262	0	9.2	-0.82e-03
0.1	252	251	13	-0.70e-03

Table 3.13: Example harker 26 variables $\gamma = 1$

Center precision	# iterations	# of centering steps	cpu secs	gap
0.9	236	0	8.2	-0.97e-03
0.1	233	232	11	-0.93e-03

Table 3.14: Example harker 26 variables $\gamma = 15$

3.6 Conclusion

We have presented an algorithm for pseudo-monotone or monotone variational inequalities $VI(\mathcal{F}, Y)$ over convex sets. At each iteration, it requires one function evaluation to generate a cut, and a second one if the gap at the weighted average point is to be computed. Beyond that, each iteration usually consists of solving from one to four linear systems, depending on the

desired recentering precision (which controls the number of centering steps) and the presence or not of equality constraints. The mapping $\mathcal{F}$ needs not be differentiable if it is single-valued, or it may be point-to-set and (maximal) monotone. The set Y can be defined by linear equalities and inequalities, or implicitly by a separation oracle.

Our approach may not be competitive with specialized methods for problems with special features: LCP's, differentiable NCP's and MCP's, strongly monotone and differentiable VI's, etc. However, when the jacobian is difficult to obtain, or simply not defined (e.g. in multi-valued cases), it is our opinion that there are few other efficient alternatives. Furthermore, the low level of monotonicity required in practice (pseudo-monotonicity or plain monotonicity) makes the ACCPM-VI method with linear cuts attractive both in terms of robustness and speed. This is well illustrated in the MMMR application above.

Finally, let us point out that faced with a VI with a strongly monotone mapping, it is possible to accelerate considerably the ACCPM-VI (linear cuts) by using *quadratic cuts*, and that this is true whether or not jacobians are available. This "ACCPM-VI (quadratic cuts)" is the next chapter's topic.

Chapter 4

A Quadratic Cut Algorithm

In the previous chapter, we introduced an algorithm for pseudo-monotone variational inequalities, that uses linear cuts. For a variational inequality problem whose mapping is both differentiable and strongly monotone, we suggest an algorithm that uses *quadratic cuts* based on the Jacobian matrix. Furthermore, our numerical experience indicates that approximations of the Jacobian matrix are also efficient, when the Jacobian's evaluation is difficult or impossible.

The chapter is divided as follows. We introduce in Section 4.1 the Dikin ellipsoid and quadratic cut concepts. We present in Section 4.2 the quadratic cut algorithm. In Section 4.3, we discuss a leaner variation of the previous section's algorithm, which, is faster in practice, and no less robust. Jacobian approximations are treated in Section 4.4, while Sections 4.5 and 4.6 respectively cover our MATLAB implementation of the algorithm and our numerical results.

4.1 Ellipsoids and quadratic cuts

4.1.1 Cuts

Let us first recall Theorem 5, which allows the generation of **linear cuts** for any VIP that is at least pseudo-monotone.

Theorem 9 *Let Y be a nonempty, closed, convex subset of $\mathbb{R}^m$ and let F be a continuous, pseudomonotone mapping from Y into $\mathbb{R}^m$. Then y^* solves the $VI(F,Y)$ if and only if $y^* \in Y$ and*

$$F(y)^t(y - y^*) \geq 0 \quad \forall y \in Y \tag{4.1}$$

Therefore, at any point $y_k \in Y$, the cut $F(y_k)^t(y_k - y) \geq 0$ defines a half-space

$$\{y \in Y \mid F(y_k)^t(y - y_k) \leq 0\}$$

which contains the solution set of $VI(F,Y)$ and on whose border y_k lies.

When F is differentiable, it is also possible to define **quadratic cuts** by using the first order approximation of $F(y)$ at some point $y_k \in Y$

$$F(y) \cong F(y_k) + \nabla F(y_k)(y - y_k).$$

Using this approximation with the definition of $VI(F,Y)$, we have

$$Q(y_k) := \{y \in Y \mid (y - y_k)^t \nabla_k (y - y_k) + F(y_k)^t (y - y_k) \leq 0\} \tag{4.2}$$

where we define $\nabla_k := \frac{(\nabla F(y_k) + \nabla F(y_k)^t)}{2}$. We also use the notation $a_k := F(y_k)$ whenever necessary to avoid any confusion when multiplying ∇_k by the vector $F(y_k)$. In fact, using a parameter α, we can define a family of cuts whose members are

$$Q_\alpha(y_k) := \{y \in Y \mid \alpha(y - y_k)^t \nabla_k (y - y_k) + F(y_k)^t (y - y_k) \leq 0\} \tag{4.3}$$

for some $\alpha \geq 0$. Clearly, we have the quadratic cut (4.2) above for $\alpha = 1$ and the linear cut (4.1) for $\alpha = 0$. If we assume ∇_k to be a positive definite matrix, which will be the case when, as we shall suppose, F is strongly monotone, the cuts $Q_\alpha(y_k)$ $(\alpha > 0)$ have the following properties.

- $Q_\alpha(y_k)$ is an ellipsoid.
- y_k lies on the border of the ellipsoid; in fact, y_k is the tangency point of the ellipsoid to the line $\{y \mid F(y_k)^t(y - y_k) = 0\}$.
- the center of the ellipsoid is at $y_k - 0.5\alpha^{-1}\nabla_k^{-1}a_k$.

For a general mapping F, there is no guarantee that a solution point y^* will not be left out by the quadratic cut (we call such cuts *unsafe*). However, under the assumption of $\bar{\alpha}$-strong monotonicity, one can build quadratic cuts that are independent of ∇_k and that never leave the solution out (*safe* cuts), as explained in the following lemma, suggested by Lüthi [64].

Lemma 4 *The point y^* is a solution of the problem $VI(F, Y)$ with F $\bar{\alpha}$-strongly monotone if and only if $y^* \in Y$ and*

$$\bar{\alpha}(y^* - y)^t(y^* - y) + F(y)^t(y^* - y) \leq 0 \quad \forall y \in Y$$

Proof: Suppose that y^* is a solution of the problem $VI(F, Y)$; by definition,

$$F(y^*)^t(y - y^*) \geq 0 \quad \forall y \in Y.$$

By adding, for any y, the above inequality to the strong monotonicity inequality

$$(F(y) - F(y^*))^t(y - y^*) \geq \bar{\alpha}(y - y^*)^t(y - y^*)$$

we obtain

$$F(y)^t(y - y^*) \geq \bar{\alpha}(y - y^*)^t(y - y^*) = \bar{\alpha}(y^* - y)^t(y^* - y)$$

Reversely,

$$\bar{\alpha}(y^* - y)^t(y^* - y) + F(y)^t(y^* - y) \leq 0 \quad \forall y \in Y$$

implies

$$F(y)^t(y - y^*) \geq 0 \quad \forall y \in Y$$

which, by theorem 5 and the monotonicity of F, implies that y^* solves $VI(F, Y)$. □

Also, under the same assumption of $\bar{\alpha}$-strong monotonicity, some members of the family of cuts $Q_\alpha(y_k)$ are safe cuts: any ellipsoid $Q_\alpha(y_k)$ with α such that $\alpha\nabla_k \preceq \bar{\alpha}I$, will not cut off the solution point y^*. The proof is similar to that of the lemma above. (For a reminder of the relation between strong monotonicity and the jacobian of the mapping, the reader is refered back to Lemma 1)

Despite the theoretical availability of safe cuts, our suggestion, described in detail in the following sections, is to use *unsafe* cuts on a temporary basis, rather than keeping *safe* ones until completion of the algorithm. The motivation for this is simple: if a cut must be "opened up" (i.e. defined with a very small positive parameter α) because only a rough lower bound on the strong monotonicity parameter is known, then this safe cut brings little more to the model than the corresponding linear cut.

4.1.2 Analytic centers and Dikin ellipsoids

The analytic center was introduced by G. Sonnevend [99] for a bounded polyhedron (with an interior) defined by a set of linear inequalities. (Analytic centers were already discussed in Section 1.3.2, but are reviewed here for the sake of clarity.)

Definition 16 *Consider the set*

$$Y = \{ y \mid A^t y \leq c \}$$

and the associated dual potential function

$$\sum_i \ln(c_i - a_i^t y)$$

where the index i is used to denote the components of c and the rows a_i^t of A^t.

The analytic center y^c *of Y is defined as the point maximizing the dual potential function over the interior of Y:*

$$y^c = \arg \max_{y \in \text{int} Y} \left(\sum_i \ln(c_i - a_i^t y) \right)$$

Alternatively, one can define the analytic center with the primal potential function or the primal-dual one, obtaining in all cases the same point. It is well known that if Y is bounded, the center y^c is unique. Writing out the first-order optimality conditions for the equivalent mathematical program

$$\begin{array}{ll} \text{maximize}_{y,s} & \sum_i \ln s_i \\ \text{subject to} & A^t y + s = c \\ & s > 0 \end{array}$$

we obtain the now-classical equations

$$Ax = 0 \tag{4.4}$$

$$A^t y + s = c \tag{4.5}$$

$$Xs = e \tag{4.6}$$

$$x, s > 0 \tag{4.7}$$

In the next section, the definition of analytic center will be extended to sets defined by linear and quadratic inequalities.

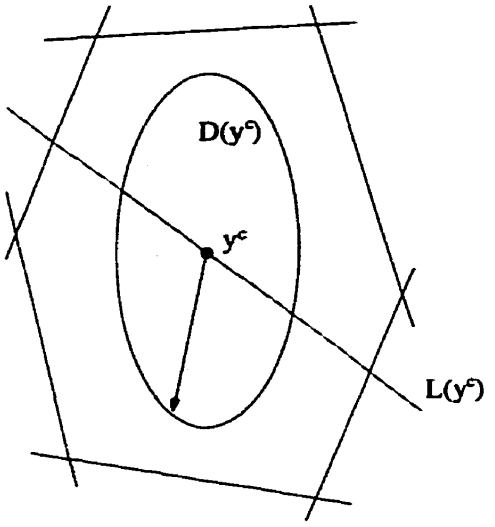


Figure 4.1: Dikin ellipse and Dikin direction of some polyhedral set

Let us also define the *ellipsoid of Dikin*

$$D(\bar{y}) = \{y \mid (y-\bar{y})^t(AS^{-2}A^t)(y-\bar{y}) \leq 1\}$$

where $s = c - A^t\bar{y}$ as in the above equations. See [2] or [89] for reference. This ellipsoid can be defined at *any* point $\bar{y} \in \text{int}(Y)$, and it is always inscribed in Y, i.e. $D(\bar{y}) \subset Y$; a proof is given in [89].

For the mapping $F : \mathbb{R}^m \to \mathbb{R}^m$ and the hyperplane $L(y^c) = \{y \mid F(y^c)^t y = F(y^c)^t y^c\}$, the point in $D(y^c)$ that is the farthest from $L(y^c)$ is

$$y^c - \frac{1}{\sqrt{F(y^c)^t(AS^{-2}A^t)^{-1}F(y^c)}}(AS^{-2}A^t)^{-1}F(y^c).$$

The second term is called *Dikin direction* (see a graphical representation in Figure 4.1.2). Note that in the *linear cut* framework, i.e. that of Chapter 3, where a linear cut is introduced at y_k, the Dikin direction is the Newton direction towards the next analytic center y_{k+1}.

Finally, Anstreicher [2] recently showed that the size of $D(y)$ can be augmented while remaining inscribed.

4.2 A quadratic cut algorithm

A variational inequality algorithm using linear cutting planes and analytic centers was introduced in [35]; in [18] the method was extended to the $VI(F, Y)$ with a point-to-set mapping F and linear equality contraints in Y. Fundamentally, a cut is introduced at the center of a *set of localisation*, known to contain the solution(s). This cut separates the set of localisation in two parts; under some qualification of F, one of the two parts is known to contain the solution set. The size of the set of localization can then be decreased; the analytic center of the reduced set is found, and the process repeated.

We introduce in this section a method which shares the same cutting planes and analytic center foundation, but that uses quadratic cuts to improve the rate of convergence. It is crucial to realize that each quadratic cut is used only temporarily, as described below.

This section describes our quadratic cut algorithm. After establishing the basis of the algorithm, we define the concept of analytic center for a linear-quadratic set. We then present the *updating step*, i.e. the first step out of the current analytic center, and the *centering steps*, used after the update to obtain the next analytic center.

4.2.1 A quadratic cut algorithm

As exposed in the previous section, the Jacobian-based quadratic cut

$$Q_\alpha(y_k) := \{\, y \in Y \mid \alpha(y - y_k)^t \nabla_k (y - y_k) + F(y_k)^t (y - y_k) \leq 0 \,\}$$

can suffer the major drawback of possibly cutting off solution points, unless a sufficiently small α is used. This "sufficiently" is difficult to quantify, and we prefer to use a constant value of α. Our choice of α was guided by the simple

variational inequality problem equivalent to the minimization of a convex, quadratic function without constraints:

$$\min f(y) = (1/2)\, y^t M y + m^t y + k \tag{4.8}$$

and the equivalent problem $VI(F,Y)$, $F(y) = My + m$, $Y = \mathbb{R}^m$.

The classical Newton's method for unconstrained minimization would solve problem (4.8) in one step by setting, from any current point y_k, the next point y_{k+1} as:

$$y_{k+1} = y_k - (\nabla^2 f(y_k))^{-1} \nabla f(y_k) = y_k - \nabla_k^{-1} a_k$$

The step $-\nabla_k^{-1} a_k$ is optimal, and to a large extent, the success of Newton's method comes from this property.

For the corresponding $VI(F,Y)$, our algorithm (which is defined below) would step, from a current point y_k, to a next point y_{k+1} that is the center of the ellipsoid $Q_\alpha(y_k)$:

$$y_{k+1} = y_k - 0.5\alpha^{-1} \nabla_k^{-1} a_k$$

Clearly, the optimal step $-\nabla_k^{-1} a_k$ is taken by our algorithm when $\alpha = \frac{1}{2}$ is used. From now on, we will then refer to $Q_{\alpha=1/2}(y_k)$ as the quadratic cut $Q(y_k)$.

To avoid leaving a solution point irremediably out of reach, quadratic cuts are introduced on a *temporary* basis only: a cut $Q(y_k)$ is kept only until the next analytic center is found. At that point, the quadratic cut is abandonned and replaced by its corresponding, safer, linear cut. A basic description of the algorithm is then as follows:

ALGORITHM ACCPM-VI (QUADRATIC CUTS)

Step 0: **Initialization**

Set $k = 0$, $Y_0 = Y$, y_0 analytic center of Y_0

Step 1: **Termination Criterion**

If the primal gap $g(y_k) \geq -\epsilon$ then stop

Step 2: **New Cut**

Compute cut $(1/2)(y - y_k)^t \nabla_k (y - y_k) + F(y_k)^t (y - y_k) \leq 0$

Step 3: **Analytic Center**

Find an approximate analytic center y_{k+1} of

$Y_k \cap \{y : (1/2)(y - y_k)^t \nabla_k (y - y_k) + F(y_k)^t (y - y_k) \leq 0\}$

Step 4: **Localisation Set Update**

Set $Y_{k+1} := Y_k \cap \{y : F(y_k)^t y_k \geq F(y_k)^t y\}$

$k := k + 1$

Return to Step 1

Step 1 consists of evaluating the primal gap defined earlier; this is the same criterion that was used in ACCPM-VI (linear cuts). Step 2 introduces the quadratic cut, which basically corresponds to a function evaluation and a Jacobian evaluation. Step 3 will be discussed in the next sections; it is necessary to both define the linear-quadratic analytic center and give a method to find it. It is very important to realise that in Step 4, the cut used in the update is not the quadratic cut but its corresponding linear cut. Note that although it is perfectly possible to use here the convex combination of centers approach of Section 3.3.4, our numerical experience indicates that it brings no advantage, and we usually dispense with this step.

4.2.2 Analytic center of a linear-quadratic set

The analytic center was defined in section 4.1.2 for a polyhedron; when a quadratic cut is added, the definition must be adapted as follows.

To keep the notation simple, let us first define $Y_k = \{ y \mid A^t y \leq c \}$, but without loosing sight of the fact that A and c grow at each iteration, since a linear cut is added, and are therefore dependent on k.

If we consider the set

$$\begin{aligned} Y_k \cap Q(y_k) &= \{ y \in \mathbb{R}^m \mid A^t y \leq c \} \cap \\ &\quad \{ y \in \mathbb{R}^m \mid (1/2)(y - y_k)^t \nabla_k (y - y_k) + F(y_k)^t (y - y_k) \leq 0 \} \end{aligned}$$

and we define corresponding slack variables

$$\begin{aligned} s &= c - A^t y \\ s_q &= -F(y_k)^t (y - y_k) - (1/2)(y - y_k)^t \nabla_k (y - y_k) \end{aligned}$$

then the analytic center of $Y_k \cap Q(y_k)$ can be defined as the point maximizing the appropriate potential function over the interior of $Y_k \cap Q(y_k)$:

$$y^c = \arg \max_{y \in \text{int}(Y_k \cap Q(y_k))} \left(\ln s_q + \sum_i \ln(c_i - a_i^t y) \right)$$

where as before i denotes the components of c and the rows a_i^t of the $n \times m$ matrix A^t. The optimality conditions for this maximization problem can be found to be

$$\begin{array}{lcl} Ax + x_q \Big(\nabla_k (y - y_k) + F(y_k) \Big) & = & 0 \\ A^t y + s & = & c \\ (1/2)(y - y_k)^t \nabla_k (y - y_k) + F(y_k)^t (y - y_k) + s_q & = & 0 \\ Xs & = & e \\ x_q s_q & = & 1 \\ x, s, x_q, s_q & > & 0 \end{array} \qquad (4.9)$$

The variables $x \in \mathbb{R}^n_+$ and $x_q \in \mathbb{R}_+$ are respectively associated with the linear cuts and the quadratic cut. It is usual to refer to the first equation as the the primal feasibility constraint, to the second and third equations as the dual feasibility constraints, to the fourth and fifth equations as the centrality constraints.

4.2.3 An updating step

We describe in this section and the next one a method for computing the next analytic center y_{k+1} when a (quadratic) cut is introduced at the current point y_k. Fundamentally, a Newton method is applied to the system of equations defining the analytic center. More than one Newton step may be necessary to reach y_{k+1}, and we distinguish the first step out of y_k, called the updating step, from the following ones, called centering steps.

The difficulty with the updating step lies within the latest cutting plane; this is true whether the cut be linear, as in [35] and [18], or quadratic as here. Indeed, because y_k lies *on* the cut, the corresponding slack value is necessarily 0, thus infeasible with respect to the positivity constraints. The linear cut approach of [35] and [18] deal with this difficulty by introducing a positive slack value for the new cut, even if this destroys the primal and dual feasibility; this positive value can be chosen analytically to ensure that the (full) Newton step will preserve all the positivity constaints. In a quadratic cut, it is more difficult to do so for lack of provably good values for the variables s_q and x_q associated to the new cut.

We propose an alternative route: from a current point y_k we step to a point on a certain path between y_k and the center of the quadratic cut. This path possesses the very interesting property of passing through the *analytic center of the intersection of Dikin's ellipsoid* $E(y_k, 1)$ *and the quadratic cut ellipsoid*; in Figure 4.2.3, this center is denoted y_{DQ}. Provided Dikin's ellipsoid represents reasonably well the localisation polyhedron Y_k, this analytic center can be expected to be close to the analytic center of the intersection of Y_k and the quadratic cut ellipsoid.

Let

$$\Delta := AS^{-2}A^t = LL^t$$

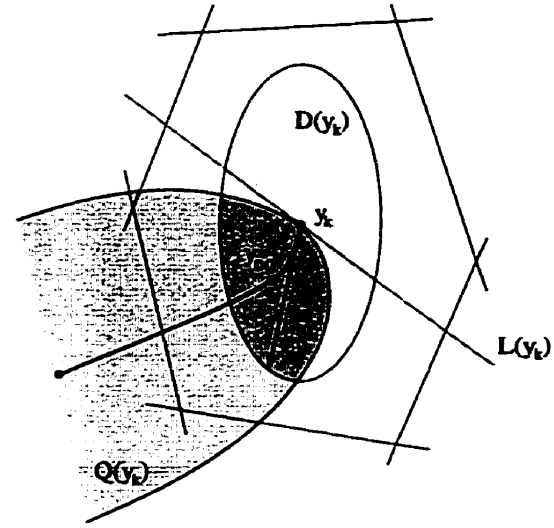


Figure 4.2: Path from y_k to the center of $Q(y_k)$

be the Cholesky factorisation of Δ and let $L^{-1}\nabla_k L^{-t}$ have the eigenvalue-eigenvector factorisation

$$L^{-1}\nabla_k L^{-t} = Q\Lambda Q^t.$$

The matrix Q is orthogonal ($QQ^t = I$) and Λ is diagonal. Then using the changes of variables

$$L^t(y - y_k) := z \text{ and } w := Q^t z$$

we rewrite the quadratic cut

$$(1/2)(y - y_k)^t \nabla_k (y - y_k) + F(y_k)^t (y - y^t) \leq 0$$

as

$$\frac{1}{2} w^t \Lambda w + F(y_k)^t L^{-t} Q w \leq 0$$

and the Dikin cut

$$(y - y_k)^t \Delta (y - y_k) \leq 1$$

as

$$w^t w \leq 1$$

The problem that defines the analytic center of the Dikin and quadratic cut ellipsoids is then

$$\text{maximize}_w \quad \log(1 - w^t w) + \log\left(-\tfrac{1}{2} w^t \Lambda w - F(y_k)^t L^{-t} Q\right)$$

over the interior of the intersection of the two ellipsoids. If we define $s_{\text{dkn}}(w) := 1 - w^t w$ and $s_{\text{cut}}(w) := -\frac{1}{2} w^t \Lambda w - F(y_k)^t L^{-t} Q$, the first order optimality condition is

$$\frac{-2w}{s_{\text{dkn}}(w)} + \frac{-\Lambda w - Q^t L^{-1} F(y_k)}{s_{\text{cut}}(w)} = 0$$

or

$$w = -\frac{1}{2}\left(\frac{s_{\text{cut}}(w)}{s_{\text{dkn}}(w)} I + \frac{1}{2}\Lambda\right)^{-1} Q^t L^{-1} F(y_k)$$

This equation traces, for values of a parameter $\theta := \frac{s_{\text{dkn}}(w)}{s_{\text{cut}}(w)}$ from 0 to ∞, a curve from y_k to the center of the quadratic cut ellipsoid; see Figure 4.2.3. Furthermore, the curve passes through the analytic center of the intersection of the two ellipsoids (y_{DQ}), and the center is easily computed once θ is known, because of the diagonality of $\frac{s_{\text{cut}}(w)}{s_{\text{dkn}}(w)} I + (1/2)\Lambda$. Therefore, a curvilinear search can be performed to find an appropriate, approximate value of θ, and the step easily recovered through

$$dy := y - y_k = L^{-t} Q w$$

In the original space, the curve is

$$y(\theta) = y_k - \frac{1}{2} L^{-t} Q (\theta^{-1} I + \frac{1}{2}\Lambda)^{-1} Q^t L^{-1} F(y_k)$$

Clearly, $y(0) = y_k$ and $\lim_{\theta \to \infty} y(\theta) = y_k - \nabla_k^{-1} a_k$; these points are respectively the current point and the center of quadratic cut ellipsoid. Also, the derivative of the multi-valued function y with respect to θ is

$$\begin{aligned} \frac{dy(\theta)}{d\theta} &= -0.5 L^{-t} Q (I + \frac{\theta}{2}\Lambda)^{-2} Q^t L^{-1} F(y_k) \\ &\overset{\theta=0}{=} -0.5 \Delta^{-1} F(y_k) \end{aligned}$$

so that for small θ, the tangent to $y(\theta)$ is the Dikin direction.

The linesearch along the curve $w(0) \rightarrow w(\infty)$ would naturally try to maximise the Dikin-Quadratic cut potential

$$\varphi_{DQ}(w(\theta)) = \log(1 - w^t w) + \log(-\frac{1}{2} w^t \Lambda w - w^t Q^t L^{-1} F(y_k))$$

This would lead to the analytic center of the intersection of Dikin's ellipsoid with the quadratic cut ellipsoid. However, since our ultimate goal is the analytic center of the intersection of the polyhedral localisation set and the quadratic cut ellipsoid, it would be more appropriate to guide the linesearch with the Polyhedral set-Quadratic cut potential

$$\varphi_{PQ}(w(\theta)) = \sum_i \log(c_i - a_i^t y_k - a_i^t L^{-t} Q w) + \log(-\frac{1}{2} w^t \Lambda w - w^t Q^t L^{-1} F(y_k))$$

4.2.4 A dual centering step

Following the updating step, which led us from y_k to y^+, it may be necessary to move closer to the analytic center of the Polyhedron-Quadratic cut intersection. This can be done with pure dual steps; recall that the updating step does not yield a full set of primal-dual values, in opposition to the updating step of Chapter 3, so that dual centering is best. The reference problem is

$$\begin{array}{ll} \text{maximize}_{y \in \text{int} Y_k} & \log\Big(-(1/2)(y - y_k)^t \nabla_k (y - y_k) - F(y_k)^t (y - y_k)\Big) \\ & + \sum_i \log(c_i - a_i^t y) \end{array}$$

and its first-order condition is

$$(1/s_q(y))(\nabla_k (y - y_k) + F(y_k)) + AS(y)^{-1} e = 0.$$

where

$$\begin{aligned} s_q(y) &:= -(1/2)(y - y_k)^t \nabla_k (y - y_k) - F(y_k)^t (y - y_k) \\ s(y) &:= c - A^t y \end{aligned}$$

Applying Newton's method to these equations from the point y^+ gives

$$\begin{aligned} M\, y_{step} &= -(1/s_q(y^+))v - AS(y^+)^{-1}e \\ \text{where } M &:= (1/s_q(y^+))\nabla_k + AS(y^+)^{-2}A^t - (1/s_q(y^+))^2 vv^t \\ \text{and } v &:= \nabla_k (y^+ - y_k) + F(y_k) \end{aligned}$$

The criterion used to evaluate the centrality of a point y^+ is an elliptic norm of y_{step} scaled by the Hessian M of the potential function (see [95]):

$$\delta(y^+) := \|y_{step}\|_M = \sqrt{y_{step}^t M y_{step}}$$

More than one centering step may have to be taken, depending on the precision of centering that is desired.

4.3 A practical quadratic cut algorithm

In practice, the algorithm described in the last section is more complex and heavier than needed. In fact, it is possible to avoid the eigenvalue-eigenvector decomposition and the nonlinear linesearch without significant effects on the number of iterations, while accelerating the time per iteration. The improvement of the performance is especially visible with larger problems, where the eigenvalue-eigenvector decomposition becomes time-expensive.

In this section we describe this alternative algorithm.

4.3.1 Two linesearches for an initial slack

We briefly mentioned in section 4.2.3 the main difficulty with the approach of [18] when a quadratic cut is involved: namely, to find provably good initial values for the new variables s_q and x_q associated to the quadratic cut.

Recall that s_q is the slack for the quadratic cut: $s_q = -((1/2)(y - y_k)^t \nabla_k (y - y_k) + F(y_k)^t (y - y_k))$. As much as possible, our initial value

for s_q should be close to its final value, i.e. the slack at the next analytic center. Also, we know that at the analytic center, $x_q = 1/s_q$.

We suggest the following approach to set the initial value of s_q. Two linesearches are performed from y_k, with the potential

$$\ln s_q + \sum_i \ln(c_i - A^t y)$$

as criterion. The two directions in question are the straight line from y_k to the center of the quadratic cut ellipsoid $Q(y_k)$, and the Dikin direction. Whichever linesearch is the most successful in finding an approximate center will have its corresponding slack as the initial slack value.

The reasoning behind this idea is the following. On one hand, if the quadratic cut ellipsoid is small relative to the Dikin ellipsoid —for example if $Q(y_k) \subset D(y_k)$— then the next analytic center y_{k+1} is likely to be close to the center of $Q(y_k)$, and the first linesearch will detect this. If on the other hand the quadratic cut ellipsoid is large compared to the Dikin ellipsoid, the effect of the quadratic cut is little more than the effect of the corresponding linear cut $F(y_k)^t(y - y_k) \leq 0$. It is known in this case that there is a point on the Dikin direction that is in the radius of quadratic convergence for the Newton method underlying the centering steps of the next section (see [34]). See Figures 4.3 and 4.4 for a graphical representation.

Therefore, it can be expected that the best of the linesearches will yield a reasonable approximation of the next analytic center, and so a reasonable initial value for s_q. The initial value of x_q is simply taken to be $1/s_q$.

4.3.2 A primal-dual infeasible update step

We call *updating step* the first (Newton) step that is taken out of the current point y_k, towards the next analytic center y_{k+1} of the intersection $Y_k \cap Q(y_k)$, which is defined by the equations (4.9).

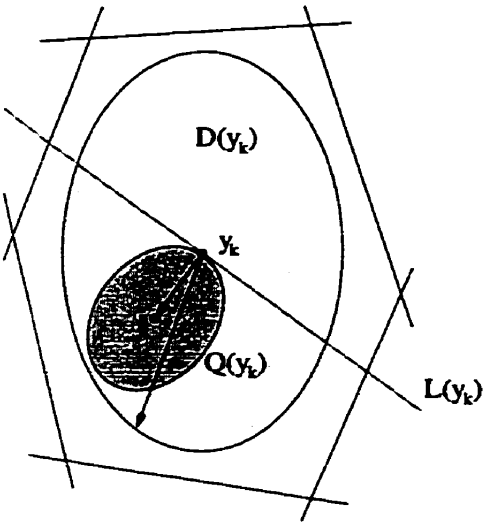


Figure 4.3: The two search directions; case 1

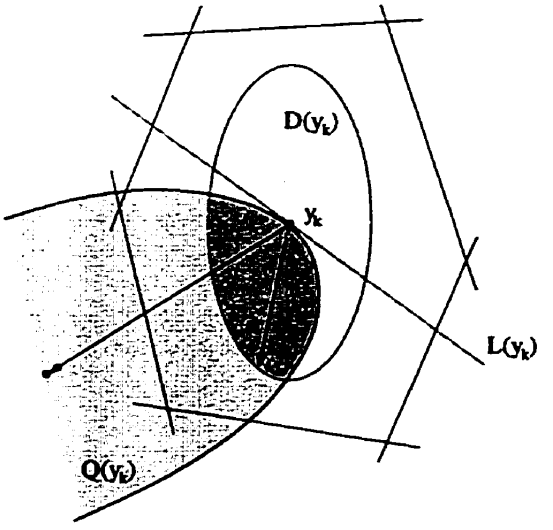


Figure 4.4: The two search directions; case 2

Applying Newton's method to this set of equations we obtain a first step out of y_k. Our approach is primal-dual so that in fact we derive steps for all variables: y, s, x, s_q and x_q. Some remarks are in order, concerning the starting point $(y_s, s_s, x_s, s_{q_s}, x_{q_s})$ for the Newton step. First, notice that, after the computation of the point $(y_k, s_k, x_k, s_{q_k}, x_{q_k})$, (i.e. the previous analytic center as well as the values of its corresponding variables), one linear inequality (a cut) has been added to Y_{k-1} to define Y_k. This was the linear cut replacing the abandoned quadratic cut. As a consequence, s_s and x_s have one more component each than s_k and x_k. The starting values s_s and x_s are then set, with their "extra" components, so that on one hand, the linear dual feasibility condition holds, and on the other hand, the centrality is not disturbed:

$$s_s = \begin{bmatrix} s_k \\ F(y_k)^t(y_{k-1} - y_k) \end{bmatrix} \qquad x_s = \begin{bmatrix} x_k \\ \frac{1}{F(y_k)^t(y_{k-1}-y_k)} \end{bmatrix}$$

The starting values s_{q_s} and x_{q_s} are determined by the procedure described in section 4.3.1. Finally, the initial value y_s is simply the current point, y_k.

Using this starting point, we obtain the following equations defining the step (dy, ds, dx, ds_q, dx_q):

$$\begin{aligned}
Adx + F(y_k)dx_q + 2x_{q_s}\nabla_k\, dy &= -Ax_s - x_{q_s}F(y_k) && (4.10)\\
A^t dy + ds &= -Ay_s - s_s + c = 0 && (4.11)\\
F(y_k)^t dy + ds_q &= -s_{q_s} && (4.12)\\
S_s dx + X_s ds &= \delta && (4.13)\\
s_{q_s} dx_q + x_{q_s} ds_q &= 0 && (4.14)
\end{aligned}$$

where $\delta := e - X_s s_s$. The right handside term in (4.11) is a consequence of dual feasibility of (y_s, s_s). In (4.13), δ is *not* zero, which explicitely accounts for the inaccuracy in the centering of the previous center, y_k.

Rearranging these equations, we obtain the following step:

$$
\begin{aligned}
dy &= -\Delta^{-1}(AS_s^{-1}e + \frac{2}{s_{q_s}}F(y_k)) && (4.15)\\
ds &= -A^t dy && (4.16)\\
ds_q &= -s_{q_s} - F(y_k)^t dy && (4.17)\\
dx &= S_s^{-1}(\delta - X_s ds) && (4.18)\\
dx_q &= -2x_{q_s} + \frac{2 - x_{q_s} ds_q}{s_{q_s}} && (4.19)
\end{aligned}
$$

where Δ is defined as

$$\Delta = AS_s^{-1}X_s A^t + \frac{x_{q_s}}{s_{q_s}}F(y_k)F(y_k)^t + x_{q_s}\nabla_k$$

Notice that this step preserves dual feasibility since

$$A^t(y + dy) + s + ds \;=\; A^t y + s + A^t dy - A^t dy \;=\; 0.$$

If after the step, the centering level

$$\left\| \begin{bmatrix} \mathrm{diag}(x_s + dx)\cdot(s_s + ds) - e \\ (x_{q_s} + dx_q)\cdot(s_{q_s} ds_q) - 1 \end{bmatrix} \right\| \qquad (4.20)$$

of the new point is considered sufficient, then $y + dy$ is taken as the new approximate center y_{k+1}. Otherwise, one or more *centering steps* are required; these are described in the following section.

4.3.3 Primal-dual centering steps

Let us denote by $y^+, s^+, x^+, s_q^+, x_q^+$ the result of the updating step, i.e.

$$
\begin{aligned}
y^+ &= y_s + dy\\
s^+ &= s_s + ds\\
x^+ &= x_s + dx\\
s_q^+ &= s_{q_s} + ds_q\\
x_q^+ &= x_{q_s} + dx_q
\end{aligned}
$$

We again apply Newton's method to the system of equations (4.9), using this time $(y^+, s^+, x^+, s_q^+, x_q^+)$ as starting point. We obtain

$$\begin{aligned} Adx + (\nabla_k(y^+ - y_k) + a_k)dx_q + x_q^+\nabla_k\, dy &= -Ax^+ - x_q^+(\nabla_k(y^+ - y_k) + a_k) \\ A^t dy + ds &= -Ay^+ - s^+ + c = 0 \\ (\nabla_k(y^+ - y_k) + a_k)^t dy + ds_q &= -q^2 - a_k^t(y^+ - y_k) - s_q^+ \\ S^+dx + X^+ds &= \delta \\ s_q^+ dx_q + x_q^+ ds_q &= 1 - s_q^+ x_q^+ \end{aligned}$$

where we define $a_k = F(y_k)$, $\delta := e - X^+s^+$ and $q^2 := (1/2)(y^+ - y_k)^t\nabla_k(y^+ - y_k)$. These equations yield the *centering step*

$$\begin{aligned} dy &= -\Delta^{-1}\left(AS_s^{-1}e + \left(\frac{1}{s_q^+} + x_q^+ + q^2 + a_k^t(y^+ - y_k)\right)\left(2\nabla_k(y^+ - y_k) + a_k\right)\right) \\ ds &= -A^t dy \\ ds_q &= -s_q^+ - q^2 - F(y_k)^t((y^+ - y_k) - (2\nabla_k(y^+ - y_k) + F(y_k))dy \\ dx &= (S^+)^{-1}(\delta - X^+ds) \\ dx_q &= \frac{1 - x_q^+ ds_q}{s_q^+} \end{aligned}$$

where Δ is defined as

$$\Delta = A(S^+)^{-1}X^+A^t + \frac{x_q^+}{s_q^+}\left(\nabla_k(y^+ - y_k) + F(y_k)\right)\left(2\nabla_k(y^+ - y_k) + F(y_k)\right)^t + 2x_q^+\nabla_k$$

Again, this step preserves the dual feasibility. The centering measure is again (4.20) and centering steps can be taken until the desired level of centering is reached.

4.4 Jacobian matrix approximations

Faced with an algorithm that uses derivative information (the Jacobian ∇F), one asks if it is not possible to keep the spirit of the method while avoiding the derivatives evaluations. The answer, often, is yes; in optimization, the

idea led to, among others, the quasi-Newton methods. In our case, the use of Jacobian approximations based on mapping evaluations can also be fruitful. This is the topic of this section.

Drawing from well-established optimization theory, we use a Broyden-Fletcher-Goldfarb-Shanno (BFGS) matrix for the Jacobian approximation. The BFGS-type approximation is a symmetric matrix which is updated by a rank-two correction, at each iteration, i.e. at each evaluation of F. Importantly, the BFGS scheme is built to preserve positive definiteness of the approximation matrix, under a condition discussed below.

Defining the notation $y_\Delta := y_{k+1} - y_k$ and $F_\Delta := F(y_{k+1}) - F(y_k)$, the BFGS Jacobian approximation for $\nabla F(y_{k+1})$ is

$$J_{k+1} = J_k + \frac{F_\Delta F_\Delta^t}{F_\Delta^t y_\Delta} - \frac{J_k y_\Delta y_\Delta^t J_k}{y_\Delta^t J_k y_\Delta}.$$

The updating process is usually initiated with the identity matrix. Under the strict monotonicity of F, positive definiteness is preserved: if J_k is positive definite, then so is J_{k+1}. Indeed, for any $z \neq 0$,

$$\begin{aligned} z^t J_{k+1} z &= z^t J_k z + \frac{z^t F_\Delta F_\Delta^t z}{F_\Delta^t y_\Delta} - \frac{z^t J_k y_\Delta y_\Delta^t J_k z}{y_\Delta^t J_k y_\Delta} \\ &= \frac{(z^t J_k z)(y_\Delta^t J_k y_\Delta) - (y_\Delta^t J_k z)^2}{y_\Delta^t J_k y_\Delta} + \frac{(F_\Delta^t z)^2}{F_\Delta^t y_\Delta} \end{aligned}$$

The first term of the right-hand side is non-negative by the Cauchy-Schwarz inequality and the positive-definiteness of J_k; the second term is non-negative because of the monotonicity assumption. Also, the two terms cannot vanish at the same time: the first vanishes only if $z = \lambda y_\Delta$ for some $\lambda \in \mathbb{R}$ ($\lambda \neq 0$), and in that case $F_\Delta^t z = \lambda F_\Delta^t y_\Delta$, which is nonzero by strict monotonicity.

Quasi-Newton updates such as the BFGS are built upon the concept of *finite-step convergence:* if F is a symmetric linear mapping, then the updates converge to the exact Jacobian in a finite number of steps. Unfortunately, this finite convergence is lost for nonlinear mappings. Furthermore, this property

relies on $F(y_{k+1})(y_{k+1} - y_k) = 0$, which will not hold in our algorithm. Note also that as a sum of symmetric matrices, J_{k+1} is symmetric, and thus only approximates the symmetric part of the Jacobian. Nothing is lost here for our algorithm, since the quadratic cut only uses the symmetric part of the Jacobian in any case.

The scaling of the updates (λJ_k for some $\lambda \in \mathbb{R}^+$) is a common improvement for quasi-Newton methods, and this is the last point we discuss in this section. In optimization, scaling is motivated by an argument emphasizing an improvement of the solution at each iteration, as opposed to some global convergence over several iterations (see, for example, Luenberger [62, sections 9.1,9.5-9.6]) . The argument applies to the idealized case of strictly convex, unconstrained quadratic minimization, with exact optimal steplengths available; scaling is however also applied to the more general cases (nonlinear but non quadratic functions, inexact linesearches), usually with great benefit. The same is true with our algorithm, namely, the argument from optimization applies to an idealized VI problem, but scaling can bring benefits in the general VI case. Indeed, consider a problem $VI(F, Y)$ where F is a linear mapping: $F(y) = My + b$ with M positive definite, and $Y = \mathbb{R}^m$ (unconstrained). Also, let our algorithm be slightly modified as follows. First, we keep no cut whatsoever from one iteration to the next, thus being guided only by the single quadratic cut generated each time from the current Jacobian approximation. This implies that the analytic center is the center of the quadratic cut. Second, we make no centering steps ever, only an updating step at each iteration; if we did not prohibit them, centering steps could happen, because of the linesearch that is our third and last modification: an exact linesearch is performed on each (updating) step, with respect to the goal $F(y^*) = 0$. Then,

1. our VIP algorithm generates the same iteration points as the corresponding quasi-Newton (optimization) method, i.e. the method whose

Hessian approximations are our Jacobian approximations, and

2. the classical argument [62, pp.261–262] in favor of spreading the eigenvalues of $J_k M^{-1}$ above and below 1, applies here as well.

We are therefore interested in a scaling of the updates

$$J_{k+1} = J_k + \lambda\Big(\frac{F_\Delta F_\Delta^t}{F_\Delta^t y_\Delta} - \frac{J_k y_\Delta y_\Delta^t J_k}{y_\Delta^t J_k y_\Delta}\Big), \quad \lambda \in \mathbb{R}^+,$$

that will ensure that the eigenvalues of $J_{k+1}M^{-1}$ include 1 in the interval they span. Let us introduce the notation

$$T_k = M^{-1/2} J_k M^{-1/2}, \qquad t_k = M^{1/2} y_\Delta$$

Then, since $F_\Delta = My_{k+1} + b - My_k - b = My_\Delta$, we have $F_\Delta = M^{1/2}t_k$ and

$$\frac{y_\Delta^t J_k y_\Delta}{y_\Delta^t F_\Delta} = \frac{t_k^t T_k t_k}{t_k^t t_k}.$$

Note also that T_k and $J_k M^{-1}$ are similar, because $M^{1/2}T_k M^{-1/2} = J_k M^{-1}$, and thus have the same eigenvalues $e_1 \leq e_2 \leq \ldots \leq e_m$. Finally, Rayleigh's principle ensures that

$$e_1 \leq \frac{t_k^t T_k t_k}{t_k^t t_k} \leq e_m$$

so that setting

$$\lambda = \frac{t_k^t t_k}{t_k^t T_k t_k} = \frac{y_\Delta^t F_\Delta}{y_\Delta^t J_k y_\Delta}$$

gives the desired scaling of the eigenvalues.

Note that for the highly specialized problem and algorithm described above, it can be proved that the condition number e_m/e_1 of $J_k M^{-1}$ is no better than that of $J_{k+1}M^{-1}$, so that in theory only a scaling of the initial approximation would be needed. In practice, it is better to use the scaling factor at each iteration.

4.5 Some notes on a MATLAB implementation

We implemented the "curve-searching", dual algorithm of Section 4.2 and the "line-searching", primal-dual algorithm of Section 4.3 with the MATLAB environment and language. The integrated functions of MATLAB, e.g. Cholesky decomposition, and the straightforward handling of sparsity, allow the rapid development of efficient codes. Some remarks on our particular implementations follow.

Linesearches At the outset of each iteration, two linesearches are performed (Section 4.3.1) for the algorithm of Section 4.3. We use a Newton method linesearch, with backtracking to remain within the current set Y_k as well as within the quadratic cut. For the algorithm of Section 4.2, the curve is searched by bisection, with appropriate safeguards.

Sparsity We called A the matrix in which the vectors of the (linear) cuts are stored; within A, it is important to differentiate between the *initial constraints (columns)*, i.e. those that define the Y of $VI(F, Y)$, from the *generated constraints (columns)*, i.e. those produced by the algorithm. The latter are of the form $F(y)$ for some y, almost always dense, and stored as such. The former, however, are usually sparse, e.g. when Y is simply a box. This sparse part of A is therefore treated differently: its sparsity is used both in its storage and in the computations. Note that this can be an important part of A: for a box constrained problem in m variables ($2m$ initial columns) requiring $2m$ iterations to solve, at the worst (in the last iteration) only one half of A is dense.

Linear systems Each step, whether update or recentering, requires the solution of a square, m-dimensional linear system. This is done with a Cholesky factorization. Note that sparse techniques cannot be used, as the matrix $AS_s^{-1}X_sA^t$, and thus the matrix Δ, are dense.

Positive Definite matrices Positive definiteness of the Jacobian matrix is important for the quadratic cut; if, because of round-off error, the matrix fails to be positive definite, an over-weighting of the diagonal is used to recover the property.

Gap evaluation The evaluation of the primal gap is done by the software CPLEX, which is linked to MATLAB by a MEX-file.

Initial point An initial point (y, x, s) is easy to find in many cases with simple sets Y. Otherwise, a feasible point is built and the centering step used to find the initial analytic center.

4.6 Numerical results

The implementations of the quadratic cut algorithms were run on a variety of examples. There are few genuine variational inequality problems available in the literature, but we nevertheless found, or built, enough problems to highlight the main features of the algorithm. For details on any given example, see the given references.

The following remarks apply to all the sections below. Under the *Method* heading, *Curve-searching* refers to the algorithm of Section 4.2; *Line-searching* refers to the algorithm of Section 4.3; *BFGS* refers to the algorithm of Section 4.3 but using a BFGS approximation of the Jacobian, with scaling; and *Linear* is the linear cut method that was described in Chapter 3. The four methods use, respectively and per iteration:

1. one evaluation of F and one of ∇F;
2. one evaluation of F and one of ∇F;
3. one evaluation of F;
4. two evaluations of F.

The iteration count is given under *# iterations*. The *# centering steps* column indicates the total number of centering steps that were necessary, for all the iterations; typically, an iteration involves the (unavoidable) updating step and one centering step. The level of asymmetry of the Jacobian, evaluated at the solution point, is given under the heading *Asymmetry*; the measure, for a Jacobian J, is

$$\frac{\|J - J^t\|}{\|J - J^t\| + \|J + J^t\|}$$

where we use the Frobenius norm of matrices. With this measure, a symmetric matrix rates 0 and a skew-symmetric one, 1 (the numerator is the skew-symmetric part of J, while $\|J + J^t\|$ is the symmetric part of J). The above measure is invariant with respect to scalings $J := kJ$, as it should be. Furthermore, for matrices with uniformly or normally distributed elements, the asymmetry measure depends very little on the size of the matrix, beyond $m = 25$ or 50. Note that for the examples considered below, the asymmetry level varies little over the feasible area, so that the solution-point evaluation is a good indicator of the asymmetry at the other points in Y.

The *Cpu time* column gives the time spent to solve the problem, excluding data input; this is evaluated by MATLAB's built-in function `cputime`. We indicate under *Gap* the level of our stopping criterion, the primal gap.

We do not report on our tests with Jacobian approximations of other types, such as BFGS without scaling and memoryless BFGS. These approximations invariably led to slower run times than the scaled BFGS.

All tests were done with a centering level of 0.1, i.e. a point is considered to be a good enough analytic center when this centrality measure is below 0.1. The level of centering that yields the fastest run time depends on the algorithm and the problem at hand. Our experiments with the quadratic algorithm seem to show that for the problems we considered, 0.1 was a good

all-round value. In comparison, one often does best with a centering level of 0.9 with ACCPM-VI (linear cuts).

All problems but one were run on an UltraSparc 1 SUN station, model 200E, with 128 Mb of memory. The Markal-Macro Multi Regional (MMMR) example was run on the parallel HP machine of the E.T.H. in Zürich.

4.6.1 A small problem: choi

We present, to begin, results for a small problem that is well known in the literature. choi is a 13-variable Nash equilibrium model that was written by Choi, DeSarbo and Harker [13]. It is available as a box-constrained variational inequality, —or mixed complementarity problem—, from the web site MCPLIB [21]. The mapping F is nonlinear, nonquadratic, and nonsymmetric. This problem is a good candidate for an acceleration of the basic linear cut method by quadratic cuts. Indeed, mapping evaluations are long, by comparison with linear mappings of the same dimension; therefore, the slightly longer iteration time of the quadratic cut approach is vastly compensated by the drop in the number of iterations, when compared with the linear cut approach. Results are given in Table 4.1. Figure 4.5 shows the improvement of the primal gap with the number of iterations; here the line "Jacobian" refers to the line-searching, primal-dual algorithm of Section 4.3.

Method	# iterations	# centering steps	Cpu time	Gap
Curve-searching	7	12	1.51 sec	$1 \cdot 10^{-4}$
Line-searching	8	6	1.06 sec	$1 \cdot 10^{-4}$
BFGS	12	13	1.49 sec	$1 \cdot 10^{-4}$
Linear	64	66	12.11 sec	$1 \cdot 10^{-4}$

Table 4.1: Example choi, 13 variables, Asymmetry=0.07

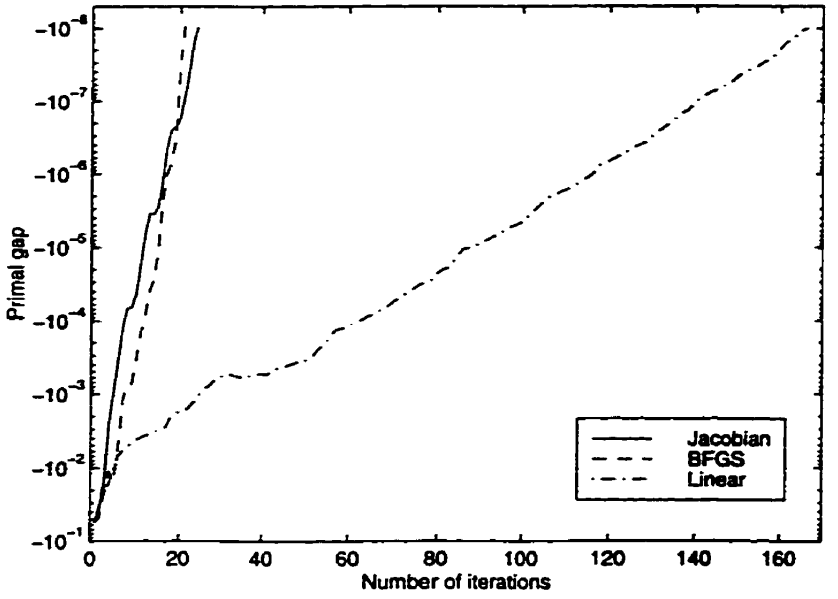


Figure 4.5: Example `choi`: Gap vs iterations for three methods

4.6.2 A CO_2 market: The MMMR model

While the improvement between quadratic cuts and linear cuts in `choi` is sizeable, the problem is one that can also solved by other efficient methods that require an explicit knowledge of the Jacobian. The MMMR model is different in that the Jacobian is not computable; very few methods exist for this kind of problem. The quadratic cut approach was used with success, despite the low monotonicity level of the problem.

The Markal-Macro Multi-Regional (MMMR) model is a large-scale representation of the energy activities of many countries, as well as a macroeconomic model of the economies of these countries. It was developped by a group a researchers at the E.T.H. and the Paul-Scherrer Institute (Zürich); see [5] and the recent thesis [11]. The model was used for the pricing of CO_2 emission permits: within the context of a "pollution trade" market, countries are allowed to buy or sell their rights to emit certain amounts of polluting gases.

Method	# iterations	# centering steps	Gap
BFGS	35	56	$-1.3 \cdot 10^{-2}$
Linear	75	75	$-1.3 \cdot 10^{-2}$

Table 4.2: Example **MMMR**, 10 variables

At the base of the model are national-level sub-models, that, in mathematical programming terms, are large nonlinear optimization problems. At the international level, these NLPs are linked by an equilibrium problem in the form of a complementarity problem $F(y) \geq 0 \quad y \geq 0 \quad F(y)^t y = 0$, or equivalently a variational problem $VI(F, \mathbb{R}^m_+)$. The evaluation of the function F at point y corresponds here to solving each of the national level NLPs, where y is used as a parameter by the nonlinear programs. Notice then that: 1) function evaluations are costly, and need to be kept to a minimum (the NLPs considered in [5] require several minutes of cpu on a mini-supercomputer); 2) the Jacobian ∇F is not defined, let alone easily computable.

We report on our results with a three-country, five-time-period, two-product problem described in [5]. The derived VIP has ten variables, and the mapping F in question was proved **not** to be monotone, although it is almost pseudo-monotone (such statements are difficult to quantify).

Figure 4.6 shows the improvement that can be made by using a quadratic cut approach, over the linear cut method.

4.6.3 Problems from the projection-method literature

We present results on two problems found in the recent article of Solodov and Tseng [98] on a variant of the projection method. As the authors argue, their method and the extra-gradient method of Korpelevitch [55] are among the most widely applicable methods for the monotone $VI(F, Y)$ since they

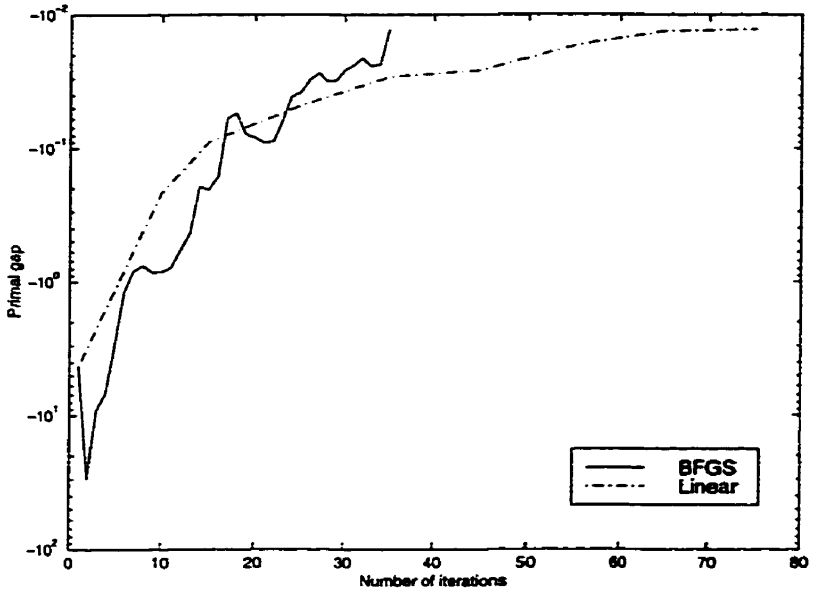


Figure 4.6: Example mmmr: Gap vs iterations for two methods

neither require the strong monotonicity of F nor its differentiability. Furthermore, the two methods apply, in theory at least, to problems with Y polyhedral or not (in practice, the projection operation may be difficult).

Strictly speaking, ACCPM-VI (quadratic cuts) cannot be used when F is simply monotone; however, it has been used advantageously even on problems that are not everywhere monotone (e.g. the MMMR example), so that we feel that there is some justification in a comparison with projection-based methods.

We tested the two largest "true" (i.e. general) variational inequality problems, —the authors also reported on LPs and LCPs—, one in dimension 10 and another in dimension 20. In [98], the problems are respectively called `Nash10` and `qHPHard`, and are taken from [42] and [45]. The authors modified the two problems slightly by adding a constraint $\sum_1^n y_k = m$ where m is the dimension of the problem. Since in its actual form our quadratic cut algorithm does not handle equality constraints, we solved the fully equiva-

Method	# iterations	# centering steps	Elapsed time	Gap
Curve-searching	56	78	6.50 sec	$5 \cdot 10^{-3}$
Line-searching	56	84	4.46 sec	$5 \cdot 10^{-3}$
BFGS	57	57	4.03 sec	$5 \cdot 10^{-3}$
Linear	204	204	11.4 sec	$1 \cdot 10^{-3}$

Table 4.3: Example **qHPHard**, 19 variables, Asymmetry= 0.034

Method	# iterations	# centering steps	Elapsed time	Gap
Curve-searching	5	5	0.54 sec	$1 \cdot 10^{-2}$
Line-searching	11	5	0.75 sec	$1 \cdot 10^{-3}$
BFGS	14	16	0.97 sec	$1 \cdot 10^{-3}$
Linear	68	73	3.56 sec	$1 \cdot 10^{-4}$

Table 4.4: Example **Nash10**, 9 variables, Asymmetry = 0

lent versions of the problems in 9 and 19 variables, using the constraint to eliminate one variable in each case. We report our results in Tables 4.3 and 4.3. It must be stressed that the two algorithms that use the Jacobian itself rely on a richer set of information than the projection-type algorithms; this gives them an unfair advantage. On the other hand, the BFGS and Linear approaches use the same information as the projection-type algorithms, i.e. only the mapping evaluations.

Note that, for the sake of comparison, we report not the cpu time but the elapsed time (MATLAB function **etime**), and we set our stopping criterion to match the different criterion used in [98]. The results obtained by Solodov and Tseng with projection-type algorithms were 251.6 seconds and 555 function evaluations for **qHPHard**, and 10.6 seconds and 192 function evalutations for **Nash10**. Improvements of one order of magnitude seem possible with the quadratic cut ACCPM-VI, while the linear cut ACCPM-VI performs quite adequately.

4.6.4 Tests of the behaviour of the algorithm

To better evaluate the behaviour of the quadratic cut ACCPM-VI, we used a problem model allowing enough flexibility in terms of size, asymmetry, nonlinearity, etc. The model is inspired from [70] but has also been used with modifications by several authors. The mapping F for the variational problem $VI(F, Y)$ is defined as

$$F(y) = \alpha(A - A^t)y + \beta B^t B y + \gamma \text{diag}(\arctan(y)) + b$$

where A and B are randomly generated $m \times m$ matrices whose elements are taken from a uniform distribution on $[0, 1]$, and the third term is a diagonal matrix with i^{th} element $\arctan(y_i)$. The parameters α, β and γ can be set to any nonnegative number, and the vector b is used to control the position of the solution (interior points, vertices, points in a face). The first term, with $A - A^t$, brings asymetry to the problem without changing its monotonicity. The second matrix $B^t B$ is symmetric and positive definite, thus bringing strict monotonicity; when the second term dominates the others, the problem is typically easier to solve. The third term is used to bring nonlinearity into the scene. We defined the set Y as the simplex

$$Y = \{y \in \mathbb{R}^m \mid y \geq 0, \ \sum_{i=1}^{m} y_i \leq m\}$$

We were first interested in observing the behaviour of the algorithm with respect to the dimension of the problem. We generated several problems of dimension 10 to 200 variables. All problems were run with the same set of parameters, $\alpha = 1$, $\beta = 3$, $\gamma = 2$; in all cases, the vector b was set to yield the desired interior solution: $[0.3, 0.3, \ldots, 0.6, 0.6, \ldots, 0.9, 0.9, \ldots, 0.9]$ with each of the three blocks being about a third of the full vector. The primal gap stopping criterion was slightly relaxed for the higher dimensions, so as to keep the number of correct digits relatively constant. It seemed more

relevant to us to require a constant precision on each component, rather than keeping the same gap value. The stopping criterion was set to -10^{-2} for the dimensions 150 and 200, to -10^{-3} for dimension 100, and to -10^{-4} for all lower dimensions. In Figure 4.7, we plot the numbers of iterations and cpu seconds required to solve each problem. Although the iteration count is relatively linear with respect to dimension, the time needed to solve the problems increases much more rapidly than the dimension. This is of course to be expected, given the order $O(nm^2)$ of the matrix products involved at each iteration (m and n being as usual the size of the problem and the cumulative number of cuts)

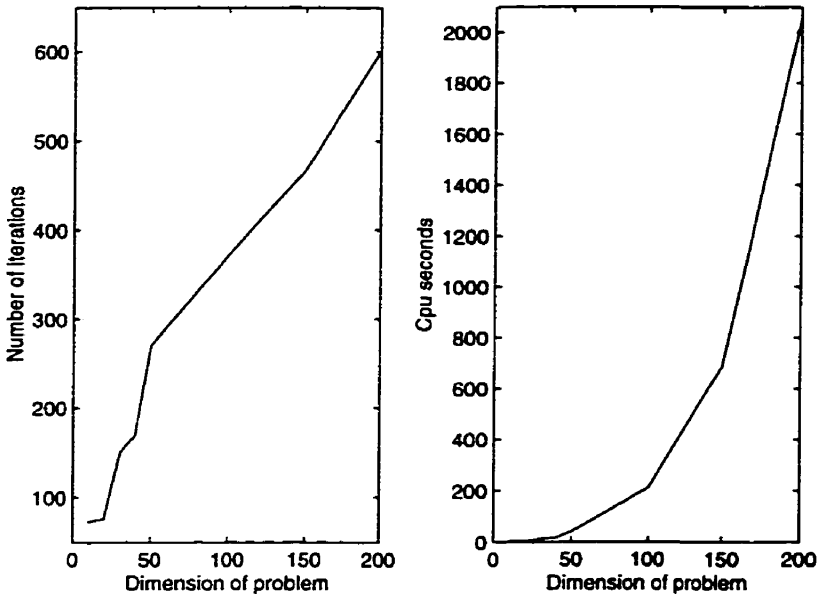


Figure 4.7: Effect of the dimension of the problem

We were also interested by the role played by asymmetry in the efficiency of the quadratic cut algorithm. Because of the implied Jacobian symmetry of the quadratic cut, one could expect that departure from symmetry would be less favorable to the use of the quadratic cut.

We ran a 25-variable problem, successively increasing the parameter α from 0 to 100. The other parameters were held at $\beta = 10$ and $\gamma = 10$. Note

that changing α does not influence the monotonicity of the problem. The results are plotted in Figure 4.8, where we give results for both the linear cut ACCPM-VI and the quadratic cut ACCPM-VI. Clearly, asymmetry is detrimental to the performance of the quadratic cut: from $\alpha = 8$ (asymmetry level ≈ 0.05), the linear cut method is superior to the quadratic cut approach.

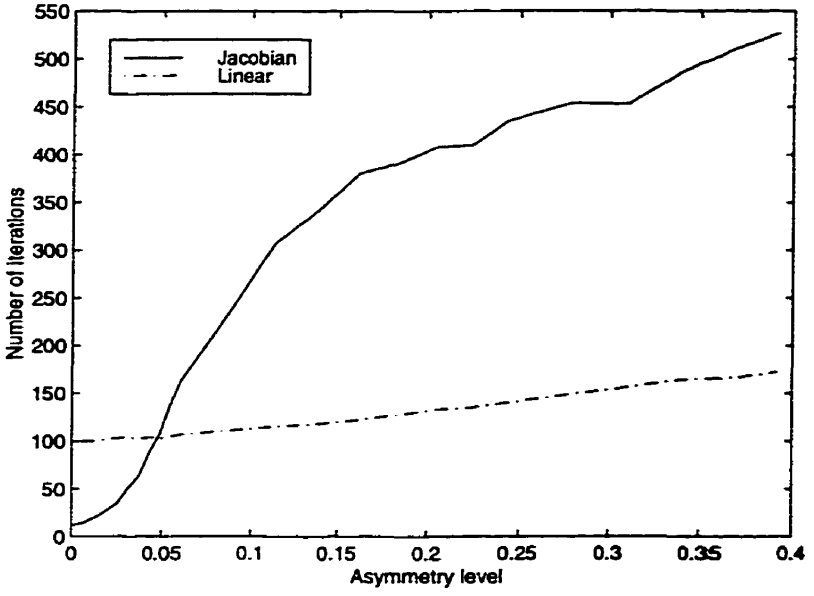


Figure 4.8: Effect of the asymmetry of the problem

4.6.5 Valuation of options in finance

In finance, derivative products such as put and call options have seen their importance grow considerably over the past decades. The valuation of these products is a topic of intense research; several approaches compete for speed and accuracy. One such approach is to formulate the problem as a set of LCPs (or VIPs), and is described in Appendix A. Our goal here is to test a medium size, real-life application of variational inequalities, with our algorithm. Given that the problem is highly structured and sparse, we cannot dispute that specialized LCP algorithms would be more appropriate in this case.

Method	# iterations	# centering steps	Cpu time	Gap
Curve-searching	251	252	302.4 sec	$1 \cdot 10^{-4}$
Line-searching	251	253	89.5 sec	$1 \cdot 10^{-4}$
BFGS	257	265	111 sec	$1 \cdot 10^{-4}$
Linear	748	748	368 sec	$1 \cdot 10^{-4}$

Table 4.5: Example `options`, 100 variables, Asymmetry=0.002

We solved a typical 3-month American put option, with the following parameters:

- Strike price: $K = \$25$
- Time span: $T = 3$ months, divided in 24 equal periods.
- Price span: $[\$0, \$50]$ in intervals of $\$0.5$
- Risk-free interest rate: $r = 10\%$
- Diffusion term: $\sigma = 0.4$

The time-stepping algorithm means that 24 VIPs in 100 variables each are solved successively. Table 4.5 shows the *average* number of iterations and cpu time for the 100-variable VIPs.

Conclusion and contributions to original knowledge

The finite-dimensional variational inequality is a mathematical tool whose power and versatility have only begun to be recognized. It is our opinion that with the development of reliable, flexible, and efficient algorithms to solve it, the variational inequality will take the place it deserves in the engineer's, the economist's, and the operations researcher's toolbox. Unfortunately, the generality of the VI makes unlikely the possibility of an algorithm that would be best for all instances. Rather, one can expect that several algorithms will each best solve a certain type of VIs. In this thesis, it was precisely our goal to develop, implement, and test two such algorithms.

The first one, ACCPM-VI (linear cuts) is indicated for problems $VI(F, Y)$ that lack such features as differentiability and strong monotonicity of F. It can be applied to problems with multi-valued mappings. For example, if the evaluation of F at a point is implicitly defined by a nonlinear program, it may well be that both Jacobian information is unavailable, and F is multi-valued (when the solution of the program is not unique). The algorithm can treat explicitly linear equality constraints Y, and through outer linear approximations, any convex set Y.

The second algorithm, ACCPM-VI (quadratic cuts), applies to the same classes of problems as the linear version, except that F needs be strongly monotone for the quadratic cuts to be well-defined; the relative importance

of the differentiability of F is not clear at the moment, given that in our tests, Jacobian evaluations can be avoided and replaced by approximations, without much damage to the performance of the algorithm. When both the linear and the quadratic cut versions of ACCPM-VI apply to a problem, the level of asymmetry of the Jacobian (whether used explicitly or not) will usually determine which version performs best.

There are few variational inequality algorigthms that can treat problems lacking strict monotonicity and differentiability. Of those few, projection-based methods (e.g. [55], [98]) and convex feasibility reformulation methods ([60], [66], [81]) are the main two groups. A full-scale comparison of such methods is difficult for two reasons. First, a good set of test problems remains to be built; this is true for general VIs but even more so for not differentiable VIs. Second, most methods from the above two groups were either never implemented, or only very superficially. However, we believe that we have shown in this thesis that ACCPM-VI (linear cuts), and ACCPM-VI (quadratic cuts) when it applies, are both robust and efficient methods for solving a large variety of variational inequality problems; furthermore, our numerical tests seem to indicate that ACCPM-VI, in either version, would have the upper hand over current projection-based methods, in many instances.

Of course, there remains plenty of room for future research; we give some possibilities here. First and foremost, the convergences and complexities of both the linear and quadratic cut approaches remain to be analyzed, which may not be an easy task given the difficulties that the analysis of other ACCPM-type methods has posed so far. Second, the handling of equality constraints remains to be extended to the quadratic method. More generally on the topic of equality constraints, a comparison of reduction techniques, as was alluded to in Sections 3.3.3 and 4.6.3, and direct handling of equalities, would be interesting. Third, it may be possible to find heuristic or theoretical

rules to adjust two parameters: the level of centering, and the α parameter of the quadratic cut (Section 4.1); the former directly influences the number of centering steps, while the latter changes the quadratic cut. Fourth, it seems attractive to keep quadratic cuts, if they can be proved to be safe; this is closely related to bounds on the strong monotonicity parameter. Finally, the idea of abandonning less useful cuts to speed up the linear algebra keeps a certain appeal for analytic center cutting plane methods, although in practice it has been rather disappointing (see the thesis [22]).

Contributions to original knowledge By and large, the first two chapters of this thesis report on existing work, while the last two contain new results. The only notable exception to this is Section 1.4.4 which presents new ideas that were not pursued further in this thesis. In Chapter 3, we extend the work of [35] in many directions. First, linear equality constraints are allowed and treated explicitly. Second, a sequence of convex combinations of analytic centers is described, and this sequence is observed to have a performance vastly superior to that of the sequence of analytic centers. Third, the possibility of maximal monotone, multi-valued mappings is considered. Finally, the implementation of the method demonstrated the ability of the algorithm to solve efficiently a large variety of problems. In Chapter 4, all the material is original, including the definition of analytic centers for non-polyhedral sets, the infeasible dual and primal-dual Newton steps in the presence of a quadratic cut, the Jacobian approximations, and of course the MATLAB implementations of the two versions of the quadratic cut algorithm.

Appendix A

The pricing of options as complementarity problems

Given that finance is not a typical or well-known area of application of variational inequalities and complementarity problems, we present here the concepts that underlie the option valuation examples of Chapters 3 and 4. Our account does not provide all the details, and the interested reader is referred to the texts by Wilmott, Howison and Dewynne [105] and [106] for a more complete description and references to articles on the subject.

Options are derivative products: they are, by nature, dependent on, or derived from, an underlying asset or group of assets (bonds, equity stocks, commodities, etc.) The most common options are called *puts* and *calls*. A put gives its owner the right to sell to a party the underlying asset at a specified price. This price is the *strike* or *exercise* price. Reversely, a call is a right to buy the underlying asset at the strike price, from the party that issued, i.e. sold, the call contract. Clearly, a right is *not* an obligation: the owner of the call or put has the option to exercise her right or not. When the date at which the option can be used is fixed, e.g. 3 months after the date of purchase, the option is of European type. When the owner has the choice to exercise her right at anytime within some period, the option is

called American. The choice of name is due to historical reasons; European and American options are traded everywhere, not only on one continent or another. When there is only one underlying asset, the options described above are called *vanilla* calls and puts, to underline their commonplace and simple nature.

We will discuss here the American vanilla put option which depends on the following parameters:

- Exercise price E
- Time span T
- Risk-free interest rate r
- Volatility σ.

The time span specifies the period during which the option may be exercised; this typically ranges from three months to a year. The annual risk-free rate is the best rate of return that can be obtained with near-absolute certainty, in treasury bills for example. Finally, the volatility is a measure of the amplitude of the underlying asset value variations. We denote by S the value of the underlying asset and use t as time variable, $0 \leq t \leq T$. The value of the put, which is a function of both S and t, is written $V(S,t)$.

At this point, we introduce the Black-Scholes differential equation, the cornerstone of a large part of modern derivative pricing theory. A justification of the Black-Scholes model can be found in any finance text on derivative products. With the above notation, it is

$$\frac{\partial V(S,t)}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V(S,t)}{\partial S^2} + rS\frac{\partial V(S,t)}{\partial S} - rV = 0 \qquad \text{(A.1)}$$

With the addition of some boundary conditions, this equation is used to derive *explicit* formulæ for the European options. The case of American options is more complicated, and there are no known explicit formulæ, so

the function $V(S,t)$ must be approximated numerically. The American put model is also based on the Black-Scholes formula; whenever an early exercise is not advisable, the conditions

$$V > E - S \qquad \frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0$$

hold. When, on the other hand, the value S of the underlying asset warrants an early exercise, the conditions

$$V = E - S \qquad \frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV < 0$$

then hold. The value of S which marks the separation between the "early exercise" area and the "do not exercise" area, is called the free boundary. The difficulty with the American option is that the free boundary is unknown and hard to find. Fortunately, the so-called complementarity formulation

$$V - (E - S) \geq 0 \qquad -\frac{\partial V}{\partial t} - \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} - rS\frac{\partial V}{\partial S} + rV \geq 0 \qquad \text{(A.2)}$$

$$\Big(V - (E - S)\Big) \cdot \Big(-\frac{\partial V}{\partial t} - \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} - rS\frac{\partial V}{\partial S} + rV\Big) = 0 \qquad \text{(A.3)}$$

avoids this difficulty as it makes no explicit reference to the free boundary. The linear complementarity problem above requires a transformation before being amenable to finite dimensional, numerical methods: discretization. Short of a formula, i.e. an explicit knowledge, of the function $V(S,t)$ over the continuous variables S and t, the asset value space and the time space are discretized, for example with

$$0 \leq n\,\Delta S \leq N\,\Delta S \text{ for } n = 0, 1, \ldots, N$$

$$0 \leq m\,\Delta t \leq M\,\Delta t \text{ for } n = 0, 1, \ldots, M$$

where ΔS and Δt are "small" enough intervals in the value space and time space, respectively. The constant M is taken so that $M\,\Delta t = T$, and N is large enough that S, the value of the stock, could reach $N\,\Delta S$ only with a

very small probability. This discretization forms a grid in the $S \times t$ space, and $V(S,t)$ is approximated at the grid points; we will use the notation

$$V_n^m := V(n\,\Delta S, m\,\Delta t) \qquad \text{and} \qquad V^m = [V_0^m\ V_1^m\ \cdots\ V_N^m]^t.$$

The derivatives in (A.2)–(A.3) are then approximated with finite differences. For example, one can use forward differences

$$\frac{\partial V(S,t)}{\partial t} = \frac{V(S,t+\Delta t) - V(S,t)}{\Delta t} + O(\Delta t)$$

for the time derivatives and central differences

$$\frac{\partial V(S,t)}{\partial S} = \frac{V(S+\Delta S,t) - V(S+\Delta S,t)}{2\Delta S} + O((\Delta S)^2)$$

$$\frac{\partial^2 V(S,t)}{\partial S^2} = \frac{V(S+\Delta S,t) - 2V(S,t) + V(S-\Delta S,t)}{(\Delta S)^2} + O((\Delta S)^2)$$

for the asset value derivatives. If the terms $O(\Delta t)$ and $O((\Delta S)^2)$ are abandoned, the Black-Scholes formula takes the form

$$\frac{V_n^{m+1} - V_n^m}{\Delta t} + \frac{1}{2}\sigma^2 S^2 \frac{V_{n+1}^m - 2V_n^m + V_{n-1}^m}{(\Delta S)^2} + rS\frac{V_{n+1}^m - V_{n-1}^m}{2\Delta S} - rV_n^m.$$

After some rearrangement, the complementarity problem (A.2)–(A.3) is then approximated by the following set of linear complementarity problems (LCPs):

$$\begin{array}{c} MV^m \geq V^{m+1}, \qquad V^m \geq \Lambda, \\ \left(MV^m - V^{m+1}\right) \cdot \left(V^m - \Lambda\right) \geq 0 \end{array} \qquad m = M-1, \ldots, 1, 0$$

where M is a tridiagonal matrix

$$M = \begin{bmatrix} B_0 & C_0 & 0 & \cdots & 0 \\ A_1 & B_1 & C_1 & & \vdots \\ 0 & A_2 & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & C_{N-1} \\ 0 & \cdots & 0 & A_N & B_N \end{bmatrix}$$

with

$$\begin{aligned} A_n &= -\frac{1}{2}(\sigma^2 n^2 - rn)\Delta t \\ B_n &= 1 + (\sigma^2 n^2 + r)\Delta t \\ C_n &= -\frac{1}{2}(\sigma^2 n^2 + rn)\Delta t. \end{aligned}$$

The vector Λ is constant and equal to

$$\Lambda = [E, \max(E - \Delta S, 0), \max(E - 2\Delta S, 0), \ldots, \max(E - N\Delta S, 0)]^t$$

To this set of complementarity problems must be added some boundary conditions, which reflect financial common sense; for example, at expiry date T, an option is worth its payoff, no more, no less, under the no-transaction-cost hypothesis. Also, the values of the option for certain extreme values of the underlying asset, are known with certainty. The boundary conditions are

$$\begin{array}{rcll} V^M & = & \Lambda & \text{(option value at expiry)} \\ V_0^m & = & E & m = 0, 1, \ldots, M \\ V_N^m & = & 0, & m = 0, 1, \ldots, M \end{array}$$

The reason for which the model is set as a set of LCPs, instead of one larger one with $(N+1) \times (M+1)$ variables, is to allow *time-stepping*. Indeed, notice that if V^{m+1} is known, then V^m can be computed as the solution of an LCP in just $N + 1$ variables. (The boundary condition takes care of the initial V^M vector) This approach is much more efficient than the solution of a large LCP treating all variables simultaneously.

The method presented above is called the *implicit finite differences method*; it is implicit because the components of the vector V^m cannot be found independently one of the other. A minimally different approach, the Crank-Nicolson method, was used in the numerical tests of Chapters 3 and 4. The only difference with the implicit method is in the finite difference scheme used to approximate the differential terms. Although the Crank-Nicolson method involves a small number of supplementary operations, it gives an $O((\Delta t)^2)$ approximation of the time derivative, which improves the overall efficiency of the algorithm. A description of the Crank-Nicolson method is given in [105].

It should be noted that our description of the problem and the algorithm was done in the actual space of the variables, i.e. asset value and time. This is in opposition with the transformation of variables that several authors

use; this allows them to treat *dimensionless* variables, and to reduce the vanilla option model to a **diffusion equation**, frequent in many areas of applied mathematics. For our approach, we saw no specific advantage to this transformation, which of course requires a reversed transformation for interpretation of the solution, and used the original variables themselves.

Finally, the LCPs above can be easily proved equivalent to the following set of variational inequalities, which are used in our numerical examples:

$$\left.\begin{array}{l} \text{Find } V_*^m \geq \Lambda \text{ such that} \\ (MV_*^m - V^{m+1})^t (V^m - V_*^m) \geq 0 \quad \forall\, V^m \geq \Lambda \end{array}\right\} m = M-1, \ldots, 1, 0$$

The set of VIs is solved sequentially, and V^{m+1} is thus a constant vector in each of them.

Bibliography

[1] A. ALTMAN AND K. C. KIWIEL, "A note on some analytic center cutting plane methods for convex feasibility and minimization problems", Technical Report (1994), Systems Research Institute, Warsaw, Poland.

[2] K. ANSTREICHER, "Ellipsoidal Approximations of Convex Sets Based on the Volumetric Barrier", *Unpublished* (1997).

[3] D.S. ATKINSON AND P.M. VAIDYA, "A cutting plane algorithm for convex programming that uses analytic centers", *Mathematical Programming* 69 (1995), 1–43.

[4] A. AUSLENDER, "Optimisation. Méthodes numériques", Masson (1976), Paris.

[5] O. BAHN, B. BÜELER, S. KYPREOS, H.-J. LUETHI, "Modeling an international market of CO2 emission permits", submitted to: A. V. Gheorghe (Ed.), *International Journal Global Energy Issues*, Special issue on Knowledge and Decision Support Systems for Integrated Modelling in Energy Policies and Management, 1997.

[6] O. BAHN, O. DU MERLE, J.-L. GOFFIN AND J.P. VIAL, "A Cutting Plane Method from Analytic Centers for Stochastic Programming", *Mathematical Programming* 69 (1995), 45–73.

[7] D. BERTSEKAS, E. GAFNI, "Projection Methods for variational inequalities with applications to the traffic assignment problem", *Mathematical Programming* 17 (1982), 139–159.

[8] D. BERTSEKAS, J. TSITSIKLIS, "Parallel and distributed computation. Numerical methods", Prentice-Hall (1989), Englewood Cliffs.

[9] S.C. BILLUPS, "Algorithms for Complementarity Problems and Generalized Equations", Ph.D. thesis, University of Wisconsin-Madison, Madison, Wisconsin, 1995.

[10] S.C. BILLUPS, M. C. FERRIS, "QPCOMP: a quadratic program based solver for mixed complementarity problems", *Mathematical Programming* 76-3 (1997), series B, 533–562.

[11] B. BÜELER, "Computing Economic Equilibria and its Application to International Trade of CO_2 Permits: An Agent-Based Approach", Ph.D. thesis, Swiss Federal Institute of Technology, Zürich, Switzerland, December 1997.

[12] E.W. CHENEY AND A.A. GOLDSTEIN, "Newton's method for convex programming and Tchebycheff approximation", *Numerische Mathematik* 1 (1959), 253–268.

[13] S. CHOI, W. DESARBO, P. HARKER, "Product positioning under price competitition", *Management Science* 36-2 (1990), 175-199.

[14] R. COTTLE, J.-S. PANG, R. STONE, "The Linear Complementarity Problem", Academic Press (1992), Boston.

[15] S.C. DAFERMOS, "An iterative scheme for variational inequalities", *Mathematical Programming* 26 (1983), 40–47.

[16] G.D. DE GHELLINCK AND J.-PH. VIAL, "A polynomial Newton method for linear programming", *Algorithmica* 1 (1986), 425–453.

[17] T. De Luca, F. Facchinei, C. Kanzow, "A semismooth equation approach to the solution of nonlinear complementarity problems", *Mathematical Programming* 75-3 (1996), 407–439.

[18] M. Denault, J.-L. Goffin, "On a primal-dual analytic center cutting plane method for variational inequalities", *Cahiers du GERAD* Numéro G-97-56, GERAD, Montreal.

[19] S. Dirkse, M. Ferris, "MCPLIB: A collection of nonlinear mixed complementarity problems", *Optimization Methods and Software* 5 (1995), 319–345.

[20] S. Dirkse, M. Ferris, "The PATH solver: A non-monotone stabilization scheme for mixed complementarity problems ", *Optimization Methods and Software* 5 (1995), 123–156.

[21] S. Dirkse, M. Ferris, MCPLIB archive, *World Wide Web* 1995, ftp://ftp.cs.wisc.edu/math-prog/mcplib/.

[22] O. du Merle, "Points intérieurs et plans coupants: mise en œuvre et développement d'une méthode pour l'optimisation convexe et la programmation linéaire structurée de grande taille", Doctoral Thesis (1995), Faculté des Sciences Économiques et Sociales, Université de Genève, Suisse.

[23] F. Facchinei, A. Fischer, C. Kanzow, "A semismooth newton method for variational inequalities: the case of box constraints", in M. Ferris and J.-S. Pang (eds), *Complementarity and variational problems: State of the art* (1997), Society for Industrial and Applied Mathematics (SIAM), Philadelphia.

[24] M. Ferris and J.-S. Pang (eds), "Complementarity and variational problems: State of the art" (1997), Society for Industrial and Applied Mathematics (SIAM), Philadelphia.

[25] A.V. FIACCO AND G.P. MCCORMICK, "Nonlinear Programming: Sequential Unconstrained Minimization Techniques" (1968), John Wiley & Sons, New York.

[26] A. FISCHER, "A special Newton-type optimization method" *Optimization* 24 (1992), 269–284.

[27] K.R. FRISCH, "The Logarithmic Potential Method for Convex Programming", Instittute of Economics, University of Oslo, Norway (unpublished manuscript (1955))

[28] M. FUKUSHIMA, "An outer approximation algorithm for solving general convex programs", *Operations Research* 31 (1982), 101–113.

[29] M. FUKUSHIMA, "A relaxed projection method for variational inequalities", *Mathematical Programming* 35 (1986), 58–70.

[30] M. FUKUSHIMA, "Equivalent differentiable optimization problems and descent methods for asymmetric variational inequality problems", *Mathematical Programming* 53 (1992), 99–110.

[31] J.-L. GOFFIN, A. HAURIE, J.-PH. VIAL, "Decomposition and nondifferentiable optimization with the projective algorithm", *Management Science* 38-2 (1992), 284–302.

[32] J.-L. GOFFIN, A. HAURIE, J.-PH. VIAL AND D.L. ZHU, "Using central prices in the decomposition of linear programs", *European Journal of Operational Research* 64 (1993), 393–409.

[33] J.-L. GOFFIN, Z.-Q. LUO, AND Y. YE, "On the complexity of a column generation algorithm for convex or quasiconvex feasibility problems", in *Large Scale Optimization: State of the Art* (1993), W.W. Hager, D.W. Hearn and P.M. Pardalos (eds), Kluwer Academic Publishers.

[34] J.-L. GOFFIN, Z.-Q. LUO, Y. YE "Complexity analysis of an interior cutting plane method for convex feasibility problems", *SIAM Journal on Optimization* 6-3 (1996), 638–652.

[35] J.-L. GOFFIN, P. MARCOTTE, D. ZHU, "An analytic center cutting plane method for pseudo-monotone variational inequalities", *Operations Research Letters* 20 (1997), 1–6.

[36] J.-L. GOFFIN, F. SHARIFI-MOKHTARIAN, "Using the primal-dual infeasible Newton method in the analytic center method for problems defined by deep cutting planes", *Cahiers du GERAD* G-94-41 (1994; revised 1997), GERAD, Montreal.

[37] J.-L. GOFFIN AND J.-PH. VIAL, "Cutting planes and column generation techniques with the projective algorithm", *Journal of Optimization Theory and Applications* 65-3 (1989), 409–429.

[38] J.-L. GOFFIN AND J.-PH. VIAL, "On the computation of weighted analytic centers and dual ellipsoids with the projective algorithm", *Mathematical Programming*, **60** (1993), 81–92.

[39] J. GONDZIO AND J.-PH. VIAL, "Warm start and ϵ-subgradients in cutting plane scheme for block-angular linear programs", *Logilab Technical Report* 1997.1, June 1997.

[40] O. GÜLER, C. ROOS, T. TERLAKY AND J.-PH. VIAL, "Interior point approach to the theory of linear programming", Technical Report 1992.3, Logilab/HEC Management Studies, University of Geneva, Switzerland, February 1992, to appear in Management Science.

[41] J.H. HAMMOND, T.L. MAGNANTI, "Generalized descent methods for asymmetric systems of equations", *Mathematics of Operations Research* 12 (1987), 678-699.

[42] P. HARKER "Accelerating the convergence of the diagonalization and projection algorithms for finite-dimensional variational inequalities", *Mathematical Programming* 41 (1988), 29–59.

[43] P. HARKER "Lectures on Computation of Equilibria with Equation-Based Methods", (1993), CORE Lecture Series, CORE Foundation, Louvain-la-Neuve.

[44] P. HARKER, J.-S. PANG, "Finite-dimensional variational inequality and nonlinear complementarity problems: a survey of theory, algorithms and applications", *Mathematical Programming* 48 (1990), 161–220.

[45] P. HARKER, J.-S. PANG, "A damped Newton method for the linear complementarity problem", in G. Allgower, K. Georg (eds), *Computational Solution of Nonlinear Systems of Equations*, Lectures in Applied Mathematics 26 (1990), American Mathematical Society, Providence, RI.

[46] J.-B. HIRIART-URRUTY, C. LEMARÉCHAL, "Convex Analysis and Minimization Algorithms I and II", A Series of Comprehensive Studies in Mathematics 305 (1993), Springer Verlag.

[47] P. JAILLET, D. LAMBERTON, B. LAPEYRE "Variation Inequalities and the Pricing of American Options", *Acta Applicandae Mathematicae* 21 (1990), 263–289.

[48] C. KANZOW, H.-D. QI, "A QP-free constrained Newton-type method for variational inequality problems", *Preprint 121* (1997) Institute of Applied Mathematics, University of Hamburg, Hamburg, Germany.

[49] S. KARAMARDIAN, "Complementarity problems over cones with monotone and pseudo-monotone maps", *Journal of Optimization Theory and Applications* 18-4 (1976), 445–454.

[50] N.K. KARMARKAR, "A New Polynomial Time Algorithm for Linear Programming", *Combinatorica* 4 (1984), 373–395.

[51] J.E. KELLEY, "The cutting plane method for solving convex programs", *Journal of the SIAM* 8 (1960), 703–712.

[52] K.C. KIWIEL, "Complexity of some cutting plane methods that use analytic centers", Technical Report (1994), Systems Research Institute, Warsaw, Poland.

[53] E.N. KHOBOTOV, "A modification of the extra gradient method for the solution of variational inequalities and some optimization problems", *Zh. Vychisl. Mat. i Mat. Fiz.*, 27 (1987), pp. 1462–1473.

[54] D. KINDERLEHRER, G. STAMPACCHIA, "An Introduction to Variational Inequalities and their applications", (1980), Academic Press, New York.

[55] G.M. KORPELEVICH, "The extragradient method for finding saddle points and other problems", *Matecon* 12 (1976), pp. 747–756

[56] T. LARSSON, M. PATRIKSSON, "A class of gap function for variational inequalities", *Mathematical Programming* 64 (1994), 53–79.

[57] S. LAWPHONGPANICH, D.W. HEARN, "Simplicial decomposition of asymmetric traffic asssignment problem", *Transportation Research* 18B (1984), 123–133.

[58] C. LEMARÉCHAL, "Nondifferentiable Optimization", in: G.L. Nemhauser et al. (eds.), *Handbooks in OR & MS* 1 (1989), 529–572.

[59] C. LEMARÉCHAL, A. NEMIROVSKII, Y. NESTEROV, "New Variants of Bundle Methods", *Rapports de Recherche INRIA* No 1508 (1991), INRIA-Rocquencourt.

[60] C. LEMARÉCHAL, A. NEMIROVSKII, Y. NESTEROV, "New Variants of Bundle Methods", *Mathematical Programming* 69 (1995), 111–147.

[61] A. Y. LEVIN, "On an algorithm for minimizing a convex function", *Soviet Mathematics Doklady* 6(1) (1965), 286–290.

[62] D. G. LUENBERGER, "Linear and Nonlinear Programming", second edition (1984), Addison-Wesley, Reading, Massachusetts.

[63] H.-J. LÜTHI, "On the solution of variational inequalities by the ellipsoid method", *Mathematics of Operations Research* 10-3 (1985), 515–522.

[64] H.-J. LÜTHI, "The analytic center quadratic cut algorithm for strongly monotone variational inequality problems", manuscript, December 1996.

[65] T. MAGNANTI, G. PERAKIS, "On the convergence of classical variational inequality problems", *Working Paper OR 280-93* (1993), Operations Research Center, MIT, Cambridge.

[66] T. MAGNANTI, G. PERAKIS, "A unifying geometric solution framework and complexity analysis for variational inequalities", *Mathematical Programming* 71 (1995), 327-351.

[67] A.S. MANNE "ETA-MACRO: A Model of Energy- Economy Interactions", in C.J. Hitch (ed.), *Modeling Energy-Economy Interactions*, (1977), Resources for the Future, Washington, D.C.

[68] P. MARCOTTE "Application of Khobotov's algorithm to variational inequalities and network equilibrium problems", *Inform. Systems Oper. Res.* 29 (1991), pp. 258–270.

[69] P. MARCOTTE "Inéquations variationnelles: Motivation, algorithmes de résolution et quelques applications", Centre de Recherche sur les Transports, Publication CRT-97-02, (1997). In French.

[70] P. MARCOTTE, J.-P. DUSSAULT, "A note on a globally convergent Newton method for solving monotone variationalinequality", *Operations Research Letters* 6-1 (1987), 35–42.

[71] L. MATHIESEN "An algorithm based on a sequence of linear complementarity problems applied to a wlarasian equilibrium model: an example", *Mathematical Programming* 37 (1997), 1–18.

[72] N. MEGIDDO, "Pathways to the optimal set in linear programming", 131–158 in: N. Megiddo (ed.), *Progress in Mathematical Programming— Interior Point and Related Methods* (1989), Springer Verlag, New York.

[73] M. MINOUX, "Mathematical Programming, Theory and Algorithms", (1986), Wiley, Chichester .

[74] G. MINTY, "Monotone (nonlinear) operators in Hilber space", *Duke Mathematics Journal* 29 (1962), 341–346.

[75] S. MIZUNO, "A new polynomial time method for a linear complementarity problem", *Mathematical Programming* 56 (1992), 31–43.

[76] K.G. MURTY, "Linear complementarity, Linear and Nonlinear Programming" (1988), Helderman, Berlin.

[77] A. NAGURNEY, "Network Economics: a variational inequality approach" (1993), Kluwer Academic Publishers.

[78] A.S. NEMIROVSKII AND D.B. YUDIN , "Estimation of the informational complexity of mathematical programming problems", *Maketon* 13 (1976), 2–45.

[79] YU. NESTEROV, "Complexity estimates for some cutting plane methods based on the analytic barrier", *Mathematical Programming* 69 (1995), 149–176.

[80] YU. NESTEROV. A. NEMIROVSKII, "Interior Point Polynomial Algorithms in Convex Programming: Theory and Applications", (1994), SIAM, Philadelphia.

[81] Y. NESTEROV, J.-PH. VIAL, "Homogeneous analytic center cutting plane methods for convex problems and variational inequalities", *Logilab Technical Report* 1997.4 (1997).

[82] J.M. ORTEGA, W.C. RHEINBOLDT, "Iterative Solution of Nonlinear Equations in Several Variables", (1970), Academic Press, New York.

[83] J.S. PANG, "Newton's method for B-differentiable equations", *Mathematics of Operations Research* 15-2 (1990), 311–341.

[84] J.S. PANG, "A B-differentiable equation-based, globally and locallly quadratically convergent algorithm for nonlinear programs, complementarity and variational inequality problems", *Mathematical Programming* 51 (1991), 101–131.

[85] J.S. PANG, D. CHAN, "Iterative methods for variational and complementarity problems", *Mathematical Programming* 24 (1982), 284–313.

[86] J.S. PANG, S.A. GABRIEL, "NE/SQP: A robust algorithm for the nonlinear complementarity problem", *Mathematical Programming* 60 (1993), 295–338.

[87] J.S. PANG, L. QI, "Nonsmooth equations: motivation and algorithms", *SIAM Journal of Optimization* 3-3 (1993), 443–465.

[88] D. RALPH, S. WRIGHT, "Superlinear convergence of an interior-point method for monotone variational inequalities", in M. Ferris and J.-S. Pang (eds), *Complementarity and variational problems: State of the art* (1997), Society for Industrial and Applied Mathematics (SIAM), Philadelphia.

[89] J. RENEGAR, "A polynomial-time algorithm based on Newton's method for linear programming", *Mathematical Programming* 40 (1988), 59–93.

[90] S. M. ROBINSON, "Generalized Equations", in A. Bachem, M. Grötschel, B. Korte (eds), *Mathematical Programming. The State of the Art. Bonn 1982*, Springer-Verlag, Berlin.

[91] S. M. ROBINSON, "Normal maps induced by linear transformation", *Mathematics of Operations Research* 17-3 (1992), 691–714.

[92] R.T. ROCKAFELLAR, "Lagrange multipliers and variational inequalities", Chapter 20 in R.W. Cottle, F. Giannessi, J.L. Lions (eds), *Variational inequalities and Complementarity problems: Theory and Applications* (1980), Wiley, New York.

[93] H. SCARF, WITH THE COLLABORATION OF T. HANSEN, "The Computation of Economic Equilibria" (1973), Yale University Press.

[94] A. SHAPIRO, "On concepts of directional differentiability", *Journal of Optimization Theory and Applications* 66 (1990), 447–487.

[95] F. SHARIFI-MOKHTARIAN, J.-L. GOFFIN, "A Nonlinear Analytic Center Cutting Plane Method for a Class of Convex Programming Problems", *Cahiers du GERAD* Numéro G-96-53 (1996), GERAD, Montreal.

[96] F. SHARIFI-MOKHTARIAN, J.-L. GOFFIN, "Long-step interior-point algorithms for a class of variational inequalities with monotone operators", *Journal of Optimization Theory and Applications*, 97–1, (1998).

[97] N.Z. SHOR, "Utilization of the operation of space dilatation in the minimization", *Cybernetics* 6(1), 7–15.

[98] M. SOLODOV, P. TSENG, "Modified projection-type methods for monotone variational inequalities", *SIAM Journal on Control and Optimization* 34-5 (1996), 1814–1830.

[99] GY. SONNEVEND, "New algorithms in convex programming based on a notion of "centre" (for systems of analytic inequalities) and on rational extrapolation", 311–326, in: Hoffman, Hiriart-Urruty, Lemarechal, Zowe (eds.), *Trends in Mathematical Optimization* (1988), Birkhäuser Verlag.

[100] D. SUN, M. FUKUSHIMA, L. QI, "A computable generalized hessian of the D-gap function and Newton-type methods for variational inequality problems", in M. Ferris and J.-S. Pang (eds), *Complementarity and variational problems: State of the art* (1997), Society for Industrial and Applied Mathematics (SIAM), Philadelphia.

[101] R. TOBIN, "Sensitivity analysis for variational inequalitites", *Journal of Optimization Theory and Appliations*, 48 (1986), 191–204.

[102] R. TOBIN, "A variable dimension solution approach for the general spatial price equilibrium problem", *Mathematical Programming* 40 (1988), 33–51.

[103] M.J. TODD AND B.P. BURRELL, "An extension of Karmarkar's Algorithm for Linear Programming using Dual Variables", *Algorithmica* 1(4) (1986), 409–424.

[104] P. TSENG, "Global linear convergence of a path-following algorithm for some monotone variational inequality problems", *Journal of Optimization Theory and Applications* 75 (1992), 265–279.

[105] P. WILMOTT, J.N. DEWYNNE, S.D. HOWISON, "Option Pricing: Mathematical Models and Computation" (1993), Oxford Financial Press.

[106] P. WILMOTT, S. HOWISON, J. DEWYNNE, "The Mathematics of Financial Derivatives. A Student Introduction." (1995), Cambridge University Press.

[107] J.H. WU, "A modified primal path-following scheme for the monotone variational inequality problem", Research Report CRT-985 (1994), Centre de recherche sur les transports, Université de Montréal; to appear in *Journal of Optimization Theory and Applications.*

[108] J.H. WU, M. FLORIAN, P. MARCOTTE, "A general descent framework for the monotone variational inequality problem", *Mathematical Programming* 61 (1993), 281–300.

[109] B. XIAO, P. HARKER, "A nonsmooth Newton method for variational inequalities, I: theory", *Mathematical Programming* 65 (1994), 151–194.

[110] B. XIAO, P. HARKER, "A nonsmooth Newton method for variational inequalities, II: numerical results", *Mathematical Programming* 65 (1994), 195–216.

[111] N. YAMASHITA, M. FUKUSHIMA, "Equivalent unconstrained minimization and global error bounds for variational inequality problems", *SIAM Journal on Control and Optimization* 35-1 (1997), 273–284.

[112] Y. ZHANG, "Solving Large-Scale Linear Programs by Interior-Point Methods Under the MATLAB Environment", Technical Report TR96-01 (1996), University of Maryland Baltimore County.

[113] D. ZHU, P. MARCOTTE, "New classes of generalized monotonicity", Research Report CRT-94-59 (1994), Centre de recherche sur les transports, Université de Montréal.

[114] S.I. ZUKHOVITSKII, R.A. POLYAK, M.E. PRIMAK, "Two methods of search of equilibrium of n person concave games", *Soviet Math. Dokl.* 10-2 (1969).