Proximal-ACCPM: a versatile oracle based optimization method

F. Babonneau[*] C. Beltran[*] A. Haurie[*] C. Tadonki[*]
J.-P. Vial[*]

revised version, June 2005

Abstract

Oracle Based Optimization (OBO) conveniently designates an approach to handle a class of convex optimization problems in which the information pertaining to the function to be minimized and/or to the feasible set takes the form of a linear outer approximation revealed by an oracle. We show, through three representative examples, how difficult problems can be cast in this format, and solved. We present an efficient method, Proximal-ACCPM, to trigger the OBO approach and give a snapshot on numerical results. This paper summarizes several contributions with the OBO approach and aims to give, in a single report, enough information on the method and its implementation to facilitate new applications.

Keywords Non-differentiable optimization, cutting plane methods, interior-point methods, Proximal-ACCPM, multicommodity flow, p-median, integrated assessment models.

Acknowledgments The work was partially supported by the Swiss NSF (Fonds National Suisse de la Recherche Scientifique, grant # 12-57093.99).

1 Introduction

Oracle Based Optimization (OBO) conveniently designates an approach to handle a class of convex optimization problems in which the information pertaining to the function to be minimized and/or to the feasible set takes the form of a linear outer approximation revealed by an oracle. By oracle, we mean a black-box scheme that returns appropriate information on the problem at so-called query

[*] Logilab, HEC, Université de Genève, 40 Bd du Pont d'Arve, CH-1211 Geneva, Switzerland.

points. In convex unconstrained optimization, this information takes the form of a linear support for the epigraph set of the function to be minimized. This class of problems is known as "Nondifferentiable Convex Optimization". We use the terminology OBO to emphasize the principle of the method — a dialog between an optimizer and an oracle — and the fact that we can handle more general classes of problems.

The goal of this paper is two-fold. We first intend to present an efficient method, Proximal-ACCPM, that implements an OBO approach. We give a concise but accurate description of the analytic center cutting plane method (ACCPM), and more precisely of its recent enhancements that include a proximal term (Proximal-ACCPM) and a logarithmic barrier on the epigraph of the smooth component of the objective function. The main issue in a cutting plane method is to decide where to query the oracle in order to improve a current polyhedral approximation of the problem. Proximal-ACCPM selects the analytic center of this polyhedral set, that is, the point that minimizes the logarithmic barrier function on that set, augmented with a proximal term. This choice is efficient since it usually requires relatively few query points to achieve an accurate approximation of an optimal solution. Proximal-ACCPM relies on the interior-point methodology to compute the query points. This methodology is well suited to handle non-linear information and makes it easy to implement the extensions we discuss in the paper.

Our second goal is to provide a set of application problems that are very different in nature and thus illustrate the versatility of the method. This choice does not cover the full range of applications successfully handled with Proximal-ACCPM. Yet it gives a flavor of what can be done and hopefully it will convince readers to develop applications of their own.

In this paper we do not deal with the convergence issue. The pseudo-polynomial complexity of the method on the feasibility problem[1] has been proved in [16, 24]. The proofs are involved but the principles underlying the method are relatively simple. Neither will we review the literature on nondifferentiable convex optimization. The field is large and we content ourselves with referring to survey papers [23, 18]. In this presentation we concentrate on the description of the method with some recent extensions and we illustrate its implementation and performance on three large-scale applications recently reported in the literature.

The paper is organized as follows. In Section 2 we present the framework of Oracle Base Optimization. Section 3 provides a succinct description of Proximal-ACCPM. Two enhancements of the method are discussed. None of them is really new, but we believe that they crucially contribute to the overall efficiency of the implementation. We also discuss how to compute a lower bound and thus obtain a reliable stopping criterion. Section 4 deals with three examples. The first one, the well-known multicommodity flow problem, is representative of large-

[1] An optimality problem can be cast in the format of a pure feasibility problem.

scale continuous optimization. The method has been applied to the linear [1] and the nonlinear [2] cases. The nonlinear version of the multicommodity flow problems we present here is particularly interesting, because part of the problem structure need not be revealed by a first-order oracle. As it is presented in Section 3, Proximal-ACCPM directly incorporates the non-linear information and thus achieves a significant gain of efficiency.

The second application is the p-median problem, a combinatorial optimization problem that is solved by Lagrangian relaxation. This example illustrate how powerful is Lagrangian relaxation to generate lower bounds for the optimal value of this combinatorial problem. These bounds are further used in an enumerative scheme which computes an optimal integer solution. In the same subsection we present the new concept of semi-Lagrangian relaxation, recently introduced in [4]. There, it is shown that using semi-Lagrangian relaxation permits us to solve to optimality the original combinatorial problem without resorting to an enumerative scheme.

Our last application deals with air quality control in urban regions and the coupling of modules in Integrated Assessment Models (IAM). The economic activity creates pollutant emissions that are spatially distributed. Geographic and climate models translate those primary pollutant emissions into ozone concentrations which determine air quality. The objective of the study is to find an optimal adjustment of the economic activity that results in acceptable ozone concentrations. The modeling approach consists in coupling two models, a techno-economic model and a climate model, to properly handle the interaction between the economic activity and the air quality. From a methodological point of view, this approach is original as it allows the coupling of two models that have totally different natures.

2 Oracle based optimization

Oracle based optimization deals with the convex programming problem

$$\min\{f(u) = f_1(u) + f_2(u) \mid u \in U \subset \mathbb{R}^n\}, \tag{1}$$

where f_1 is a convex function, f_2 is a twice differentiable convex function and U is a convex set. We assume that $f_1(u)$ and U are revealed by a first order oracle while $f_2(u)$ is accessed through a second order oracle in an explicit way. By oracle, we mean a black-box procedure which at any *query point* u returns the information described in Definitions 1 and 2 below.

Definition 1 *A first-order oracle for problem (1) is a black box procedure with the following property. When queried at u, the oracle returns 1 or 2.*

1. *$u \notin U$ and (a, c) is such that $a^T u' - c \leq 0, \forall u' \in U$ (feasibility cut). In that case, we set $f_1(u) = +\infty$.*

2. $u \in U$ and (a, c) is such that $a^T u' - c \leq f_1(u'), \forall u' \in U$ (optimality cut)[2].

Definition 2 *A second-order oracle for problem (1) is a black-box procedure with the following property. When queried at u, the oracle returns the function value and the first and second derivatives of $f_2(u)$.*

In the traditional OBO approach, the function f_2 is handled in the same way as f_1, that is by means of a first-order oracle. This approach looses information. In this paper, we exploit the explicit knowledge of the function f_2 and its derivatives in the form of a barrier on the epigraph set.

Assumption 1 *The function f_2 is such that the logarithmic barrier $-\log(\zeta - f_2(u))$ on the epigraph set of f_2, $\{(u, \zeta) \mid \zeta \geq f_2(u),\ u \in U\}$, is self-concordant*[3].

In many applications, the objective function f_1 is a strictly positively weighted sum of p nonsmooth convex functions

$$f_1(u) = \sum_{i=1}^{p} \pi_i f_{1i}(u).$$

In that expression, we can consider that $f_1(u)$ is revealed by p independent first-order oracles. The epigraph of the function f is the set defined by $\{(u, z, \zeta) \mid \pi^T z \geq f_1(u),\ \zeta \geq f_2(u)\}$. Using this property, problem (1) can also be written in as

$$\begin{array}{ll} \min & \pi^T z + \zeta \\ \text{s.t.} & f_{1j}(u) - z_j \leq 0, \quad j = 1, \ldots, p, \\ & f_2(u) - \zeta \leq 0, \\ & u \in U. \end{array} \tag{2}$$

This formulation is conveniently named the *disaggregate mode*.

The first order oracle is used to build a polyhedral approximation of the epigraph of f_1. Suppose the oracle has been queried at u^k, $k = 1, \ldots, \kappa$, and has returned feasibility and/or optimality cuts associated with those points. The corresponding inequalities are collected in

$$A^T u - E^T z \leq c.$$

[2] In general, $a \in \partial f_1(u)$, $c = a^T u - f_1(u)$, but this is not necessarily so. The cut may have no intersection with the epigraph set (i.e., may be situated strictly below that set).

[3] The concept of self-concordant function has been introduced by Nesterov and Nemirovski [26] to extend the theory of interior-point methods for linear programming to a more general class of functions. The condition links the second and third derivatives of the function. For a thorough but more readable presentation of the theory of self-concordant functions we refer to [25].

In that definition, the subgradients a of the function f_1 form the matrix A while E is a binary matrix that is constructed as follows. If the objective f_1 is treated in an aggregate mode ($p = 1$), then E is a binary row vector. An entry one in E indicates that the z variable is present in the cut, implying that the cut is an *optimality cut*. In contrast, a zero indicates that the cut is a *feasibility cut*. If the objective f_1 is disaggregated into p components, row j of E corresponds to a variable z_j and each column corresponds to a cut. An entry one in row j and column k indicates that the cut k is an optimality cut for $f_{1j}(u)$. If column k is a null vector, then cut k is a feasibility cut.

Let $\bar{\theta}$ be the best recorded value such that $\bar{\theta} = \min_{k \leq \kappa}\{f_1(u^k) + f_2(u^k)\}$. In view of the above definitions, we can define the localization set $\mathcal{L}_\kappa$ as

$$\mathcal{L}_\kappa = \left\{(u, z, \zeta) \mid A^T u - E^T z \leq c,\ f_2(u) \leq \zeta,\ \pi^T z + \zeta \leq \bar{\theta}\right\},$$

which is a subset of an outer approximation of the epigraph of f that contains all optimal pairs $(u^*, f(u^*))$. Thus, the search for a new query point should be confined to the localization set. Among possible solution methods for (1), we briefly sketch cutting plane schemes which work as follows:

1. Select a query point in the localization set.
2. Send the query point to the first order oracle and get back the optimality/feasible cuts.
3. Send the query point to the second order oracle to compute the objective function f_2.
4. Update the lower and upper bounds and the localization set.
5. Test termination.

The main issue in the design of a cutting plane scheme is step 1. Different choices lead to different results. In that paper, we propose a particular method, named *Proximal-ACCPM*, that selects the analytic center of the localization set as the new query point.

3 Proximal-ACCPM

It is well-known that efficient methods for non differentiable convex optimization rely on some regularization scheme to select the query point. We discuss here such a scheme; it is based on the concept of proximal analytic center which corresponds to the minimum of the standard logarithmic barrier augmented with a proximal term.

3.1 Proximal analytic center

We associate with the localization set a standard (weighted) logarithmic barrier

$$F(s_0, s, \sigma) = -w_0 \log s_0 - \sum_{i=1}^{\kappa} w_i \log s_i - \omega \log \sigma, \tag{3}$$

with $(s_0, s, \sigma) > 0$ defined by

$$\begin{array}{rcl} s_0 & = & \bar{\theta} - \pi^T z - \zeta, \\ s_i & = & c_i - (A^T u - E^T z)_i, \quad i \in K = \{1, \ldots, \kappa\}, \\ \sigma & = & \zeta - f_2(u). \end{array}$$

The barrier function is augmented with a proximal term to yield the augmented barrier

$$\Psi(u, s_0, s, \sigma) = \frac{\rho}{2} ||u - \bar{u}||^2 + F(s_0, s, \sigma), \tag{4}$$

where $\bar{u} \in \mathbb{R}^n$ is the query point that has achieved the best objective value $\bar{\theta}$. We name it the proximal reference point. The proximal analytic center is defined as the solution of

$$\begin{array}{rll} \min\limits_{u,z,\zeta,s_0,s,\sigma} & \Psi(u, s_0, s, \sigma) & \\ \text{s.t.} & s_0 + \pi^T z + \zeta = \bar{\theta}, & \\ & s_i + (A^T u - E^T z)_i = c_i, & i \in K = \{1, \ldots, \kappa\}, \\ & \sigma + (f_2(u) - \zeta) = 0, & \\ & s_0 > 0, \ s > 0, \ \sigma > 0. & \end{array} \tag{5}$$

If $(u, z, \zeta, s_0, s, \sigma)$ is feasible to (5), then (5) is equivalent to minimizing $\Phi(u, z, \zeta) = \Psi(u, s_0, s, \sigma)$, in which s_0, s and σ are replaced by their value in u, z and ζ. Note that the localization set is not necessarily compact, but it is easy to show that, thanks to the proximal term, the generalized analytic center exists and is unique.

In the next paragraphs, we shall use the following notation. Given a vector $s > 0$, S is the diagonal matrix whose main diagonal is s. We also use $s^{-1} = S^{-1}e$ to denote the vector whose coordinates are the inverse of the coordinates of s. Similarly, $s^{-2} = S^{-2}e$. Finally, given two vectors x and y of same dimension, xy denotes their component-wise product. With this notation, the first order optimality conditions for (5) are

$$\rho(u - \bar{u}) + Aws^{-1} + \omega f_2'(u)\sigma^{-1} = 0, \tag{6a}$$

$$\pi w_0 s_0^{-1} - Ews^{-1} = 0, \tag{6b}$$

$$w_0 s_0^{-1} - \omega \sigma^{-1} = 0, \tag{6c}$$

$$s_0 + \pi^T z + \zeta - \bar{\theta} = 0, \tag{6d}$$

$$s_i + (A^T u - E^T z)_i - c_i = 0, \quad i \in K = \{1, \ldots, \kappa\}, \tag{6e}$$

$$\sigma + f_2(u) - \zeta = 0. \tag{6f}$$

The algorithm that computes the analytic center is essentially a Newton method applied to (6a)-(6f). We shall see later how the vector $\xi = ws^{-1}$ is used to derive a lower bound for the optimal solution.

In view of Assumption (1), Φ is self-concordant; Newton's method is thus polynomially convergent [25]. For the sake of simplicity, let us define $v = (u, z, \zeta)$. In the case when v is feasible to (5) the Newton direction is

$$dv = -[\Phi''(v)]^{-1}\Phi'(v).$$

The damped Newton's method for computing the proximal analytic center consists in taking damped steps to preserve feasibility of v. The aim is to achieve a sufficient decrease of Φ, until the domain of quadratic convergence is reached. Let

$$\lambda(v) = ([\Phi''(v)]^{-1}\Phi'(v))^T\Phi'(v) = -dv^T\Phi'(v). \tag{7}$$

As long as $\lambda(v) > \frac{3-\sqrt{5}}{2}$ a step of length $(1 + \lambda(v))^{-1}$ preserves feasibility and induces a decrease of Φ by an absolute constant. When $\lambda(v) \leq \frac{3-\sqrt{5}}{2}$ a full step is feasible and the method converges quadratically. The method has polynomial complexity.

The stopping criterion is triggered by the proximity measure. When $\lambda(v)$ falls below the threshold value $\eta < \frac{3-\sqrt{5}}{2}$, the search for the proximal analytic center stops. In practice, the much looser criterion $\eta = 0.99$ suffices.

3.2 Infeasible Newton's method

Unfortunately we don't have easy access to feasible solution for problem (5). In cutting plane schemes, new constraints cut off the current iterate from the new localization set and there is no direct way to retrieve feasibility if the cuts are deep. Since we can't anymore eliminate the variables (s_0, s, σ), we can't apply a feasible Newton method to minimize Φ. Thus, we propose an infeasible start Newton method for (5), which aims to achieve feasibility and optimality simultaneously in the extended space $(u, z, \zeta, s_0, s, \sigma)$.

In the course of the optimization process, the first order conditions (6a)-(6f) are never satisfied. However, we can assume that $(s_0, s, \sigma) > 0$. We introduce the residuals $r = (r_u, r_z, r_\zeta, r_{s_0}, r_s, r_\sigma)$ and write

$$\begin{aligned}
\rho(u - \bar{u}) + Aws^{-1} + \omega f_2'(u)\sigma^{-1} &= -r_u, && \text{(8a)}\\
w_0\pi s_0^{-1} - Ews^{-1} &= -r_z, && \text{(8b)}\\
w_0 s_0^{-1} - \omega\sigma^{-1} &= -r_\zeta, && \text{(8c)}\\
s_0 + \pi^T z + \zeta - \bar{\theta} &= -r_{s_0}, && \text{(8d)}\\
s_i + (A^T u - E^T z)_i - c_i &= -r_{s_i}, \quad i \in K = \{1, \ldots, \kappa\}, && \text{(8e)}\\
\sigma + f_2(u) - \zeta &= -r_\sigma. && \text{(8f)}
\end{aligned}$$

The Newton direction associated to (8a)-(8f) is given by

$$P \begin{pmatrix} du \\ dz \\ d\zeta \\ ds_0 \\ ds \\ d\sigma \end{pmatrix} = \begin{pmatrix} r_u \\ r_z \\ r_\zeta \\ r_{s_0} \\ r_s \\ r_\sigma \end{pmatrix}, \tag{9}$$

where

$$P = \begin{pmatrix} \rho I + \omega f_2(u)''\sigma^{-1} & 0 & 0 & 0 & -AS^{-2} & \omega f_2(u)'\sigma^{-2} \\ 0 & 0 & 0 & -w_0\pi s_0^{-2} & E^T S^{-2} & 0 \\ 0 & 0 & 0 & -w_0 s_0^{-2} & 0 & \omega\sigma^{-2} \\ 0 & \pi^T & 1 & 1 & 0 & 0 \\ A^T & -E^T & 0 & 0 & I & 0 \\ f_2'(u) & 0 & -1 & 0 & 0 & 1 \end{pmatrix}.$$

Since (6d) and (6e) are linear, a full Newton step, i.e., a step of length 1, yields a point that is feasible with respect to these equations. However, the same step does not yield a feasible point with respect to the nonlinear equation (6f). Thus, the method remains essentially infeasible and we cannot use the proximity measure λ to determine the steplength α_{step}. Instead, we use the following empirical rule. Let

$$\alpha_{max} = \max(\alpha \mid s + \alpha ds > 0,\ s_0 + \alpha ds_0 > 0,\ \sigma + \alpha d\sigma > 0),$$

the selected step is

$$\alpha_{step} = \min(1,\ \gamma\alpha_{max}),$$

where the parameter γ is a safeguard to stay away from the boundary of the domain. In practice, we take $\gamma = 0.95$.

When $f_2(u)$ is linear (or constant), it may be the case that (6d) and (6e) become satisfied. Instead of using the default step length $(1+\lambda(v))^{-1}$, as prescribed by the theory, we perform the one-dimensional linesearch

$$\alpha_{step} = \arg\min \Psi(v + \alpha dv).$$

As mentioned earlier, the query point is not feasible for the new cuts returned by the first order oracle. Finding a good starting value for $s_{\kappa+1}$ and/or s_0 after a cut has been added is an issue. Though [17] proposes a scheme that preserves the polynomial complexity of the method, in our practical implementation we use a simple heuristic that turn out to be very efficient.

To summarize, a basic step of the Newton iteration is

1. Send the current point u to the second order oracle to compute the objectif function $f_2(u)$ and its first and second derivatives.

2. Compute the Newton step $(du, dz, d\zeta, ds_0, ds, d\sigma)$ by (9).

3. Compute a step length α_{step} to update $(u, z, \zeta, s_0, s, \sigma)$.

4. Test termination.

3.3 Lower bound

A lower bound for (1) permits a measure of progress to optimality. We now explain a way to generate such a bound. The first step in the derivation of the lower bound consists in introducing the perturbed function $f(u) - r^T u$, where r is a vector to be specified later. The second step is to replace the non-smooth function $f_1(u)$ by its current polyhedral approximation. This is done by replacing $f_1(u)$ by $\pi^T z$ under the constraints $A^T u - E^T z \leq c$. We thus have the bounding inequality

$$f(u) - r^T u \geq \min_{u,z}\{\pi^T z + f_2(u) - r^T u \mid A^T u - E^T z \leq c\}.$$

In view of the convexity of f_2, we may write

$$\begin{aligned} f(u) - r^T u \quad &\geq \quad f_2(u^c) - f_2'(u^c)^T u^c + \\ & \qquad \min_{u,z}\{\pi^T z + f_2'(u^c)u - r^T u \mid A^T u - E^T z \leq c\}, \end{aligned}$$

where u^c is a point of choice (e.g., approximate analytic center). By duality we obtain

$$\begin{aligned} f(u) - r^T u \quad &\geq \quad f_2(u^c) - f_2'(u^c)^T u^c + \\ & \qquad \min_{u,z} \max_{\xi \geq 0}\{(f_2'(u^c) + A\xi)^T u + (\pi - E)^T z - c^T \xi - r^T u\}, \\ &= \quad f_2(u^c) - f_2'(u^c)^T u^c + \\ & \qquad \max_{\xi \geq 0}\left\{-c^T \xi + \min_{u,z}\left[(f_2'(u^c) + A\xi - r)^T u + (\pi - E\xi)^T z\right]\right\} \end{aligned} \tag{10}$$

If $\xi \geq 0$ is such that $f_2'(u^c) + A\xi = r$ and $E\xi = \pi$, then

$$f(u) \geq f_2(u^c) - f_2'(u^c)^T u^c + r^T u - c^T \xi.$$

We now show how one can get such a vector ξ at the end of the iterations that compute the proximal analytic center. In view of (8b), we let $\xi = \xi^c = w(s^c)^{-1} > 0$ and we scale ξ^c by using the special structure of the matrix E to have $\pi - E\xi^c = 0$ and we define $r = f_2'(u^c) + A\xi^c$. In view of the optimality conditions (6a) and (6b) one may expect r to be small. We obtain the bound for the optimal objective function value by

$$\begin{aligned} f(u^*) &\geq f_2(u^c) - f_2'(u^c)^T u^c - c^T \xi^c + r^T u^*, \\ &\geq f_2(u^c) - f_2'(u^c)^T u^c - c^T \xi^c + r^T (u^* - u^c) + r^T u^c, \\ &\geq f_2(u^c) - f_2'(u^c)^T u^c + r^T u^c - c^T \xi^c - ||r||\delta. \end{aligned} \tag{11}$$

The last inequality follows from Cauchy-Schwartz and $\delta \geq ||u^* - u^c||$ is an upper bound on the distance of the current point u^c to the optimal set. Finding a good value for δ cannot be done on theoretical grounds. It is essentially problem dependent. In practice, we obtained good results by taking the "empirical" value $\delta = 5 \times ||u^c - \bar{u}||$.

If the variable u is constrained to be nonnegative in (1), we can further improve the computation of the lower bound by taking $r = -\min\{0, f_2'(u^c) + A\xi^c\}$, where the min operator is taken component-wise. In that case, the coefficient of u in the inner minimization is always nonnegative and $(f_2'(u^c) + A\xi - r)^T u = 0$ at the solution of (10). This remark is particularly useful when $r = 0$. Then we obtain the exact lower bound $f_2(u^c) - f_2'(u^c)^T u^c - c^T \xi^c$.

3.4 Implementation

Since the oracle is entirely user-defined, we do not include it in the description. The code has two main blocks: the first one computes query points; the second one organizes the dialog between the oracle and the query point generator. The code also includes an important initialization block.

Initialization This module initializes the instance and the various parameters.

Query point generator This modules includes two submodules: the first one creates the localization set based on the information sent by the cut manager; the second one computes approximate proximal analytic centers.

Manager This module keeps track of the cuts generated by the oracle and of the current primal and dual coordinates of the analytic center. It also controls the parameters that are dynamically adjusted and computes criteria values that can be used by the user to stop the algorithm. Finally, it acts as a filter between the oracle and the query point generator.

Two parameters of Proximal-ACCPM are often critical in the applications: the weight w_0 on the epigraph cut in (3) and the coefficient ρ of the proximal term in (4). The general strategy is to assign to w_0 a value equal to the number of generated cuts [18]. The management of the proximal term is more problem dependent. This point will be briefly commented in the next section. When the problem to be solved has no box constraints on the variables (e.g., when relaxing equality constraints in Lagrangian relaxation) the computation of the Newton direction in Proximal-ACCPM can be made more efficient than in plain ACCPM [11].

The code is written in Matlab; it has around 700 lines of code in the query point generator and 400 in the manager. Matlab is particularly efficient in dealing with linear algebra. Not much gain can be expected by translating the code into C++. However, a C version would make it easier to link Proximal-ACCPM with oracles written in C or FORTRAN or to do an embedding of Proximal-ACCPM within a larger optimization scheme (e.g., a branch and bound scheme). The code is the result of a continuing development efforts by teams at Logilab partly supported by Swiss NSF.

4 Applications

We have seen that oracle based optimization is relevant when it is possible to approximate the epigraph set of the function to be minimized, and the feasible set, by polyhedral sets. Let us list a few techniques that lead to this situation: Lagrangian relaxation [15], Lagrangian decomposition [19], column generation [3], Benders' decomposition [5], dual gap function in variational inequalities [27], etc. In this section we present three representative applications, one in large-scale nonlinear continuous optimization, one in combinatorial optimization and one dealing with the coupling of economic and environmental models. Those problems have been fully treated in [1, 2, 4, 8].

In each case, we give a brief presentation of the problem and report a sample of numerical results. This will give the reader an idea of the type of problems that can be solved with Proximal-ACCPM. When the numerical results are displayed in a table, we give the following information: problem identification, denoted 'Problem ID', number of outer iterations (equivalently, the number of oracle calls), denoted 'Outer', number of inner iterations (Newton iterations to compute an analytic center), denoted 'Inner', total CPU time in second, denoted 'CPU' and the fraction of the CPU time spent in the oracle, denoted '%Oracle'.

4.1 Multicommodity flow problems

Given a network represented by the directed graph $\mathcal{G}(\mathcal{N}, \mathcal{A})$, with node set $\mathcal{N}$ and arc set $\mathcal{A}$, the multicommodity flow problem consists in shipping some commodity flows from sources to sinks such that the demands for each commodities are satisfied, the arc flow constraints are met and the total cost flow is minimum. The arc-flow formulation of the multicommodity flow problem is

$$\min \quad \sum_{a \in \mathcal{A}} f_a(y_a) \tag{12a}$$

$$\text{s.t.} \quad \sum_{k\in\mathcal{K}} x_a^k = y_a, \qquad \forall a \in \mathcal{A}, \tag{12b}$$

$$Nx^k = d_k\delta^k, \qquad \forall k \in \mathcal{K}, \tag{12c}$$

$$x_a^k \geq 0, \qquad \forall a \in \mathcal{A},\ \forall k \in \mathcal{K}. \tag{12d}$$

Here, N is the network matrix; $\mathcal{K}$ is the set of commodities; d_k is the demand for commodity k; and δ^k is vector with only two non-zeros components: a 1 at the supply node and a -1 at the demand node. The variable x_a^k is the flow of commodity k on the arcs a of the network and x^k is the vector of x_a^k. The objective function f is a congestion function on the arcs.
For the sake of simpler notation we write problem (12) in the more compact formulation

$$\min\{f(y) \mid Bx = y,\, x \in X\}, \tag{13}$$

where X represents the set of feasible flows that meet the demands with respect to the network constraints. Bx defines the load flow.

The standard Lagrangian relaxation of (13) assigns the dual variables u to the coupling constraints $Bx = y$ and relaxes them. The Lagrangian problem is

$$\max_{u\geq 0} \mathcal{L}(u), \tag{14}$$

where

$$\begin{aligned}\mathcal{L}(u) &= \min_{x\in X, y} f(y) + u^T(Bx - y),\\ &= \min_y (f(y) - u^T y) + \min_{x\in X} u^T Bx,\\ &= -f_*(u) + \min_{x\in X} u^T Bx.\end{aligned}$$

The function $f_*(u)$ is the Fenchel conjugate of f; it is convex. In the multicommodity case, the second part of the Lagrangian is a sum of $|\mathcal{K}|$ shortest path problems. We denote

$$\mathrm{SP}(\bar{u}) = \min_{x\in X} (B^T\bar{u})^T x. \tag{15}$$

We recall that in Proximal-ACCPM, we treat the negative of the objective function (14). Let $\bar{x}$ be an optimal solution returned by the oracle (15) at a given point $\bar{u}$. Since $\mathrm{SP}(u)$ results from a minimization problem, the inequality $\mathrm{SP}(u) \leq (B\bar{x})^T u$ provides a linear upper estimate of the concave function $\mathrm{SP}(u)$. The solution computed by the oracle $-f_*(\bar{u}) + (B\bar{x})^T\bar{u}$ produces a lower bound for the original problems. Instead of using (11) to compute an upper bound, we use the variable ξ to compute a feasible solution to (12) (It can be shown).

For the nonlinear multicommodity flow problem, we use the most widely used function in telecommunications, the so-called Kleinrock congestion function:

$$f(y) = \frac{y}{c - y},$$

where c is the vector of capacities on the arcs. The conjugate function is

$$f_*(u) = 2\sqrt{c^T u} - c^T u - 1, \quad \forall u \geq \frac{1}{c}.$$

For the linear case, the objective function is

$$f(y) = \begin{cases} t^T y, & 0 \leq y \leq c, \\ +\infty, & \text{otherwise}, \end{cases}$$

where c is the vector of capacities and t the vector of unit shipping cost on the arcs. The conjugate function is

$$f_*(u) = c^T u, \quad \forall u \geq 0.$$

To get a feel for the numerical performance, we pick few examples that have been solved in [1, 2]. We select 3 types of problems. `Planar` and `Grid` instances are telecommunications networks while `Winnipeg`, `Barcelona` and `Chicago` are transportation problems. Table 1 gives for each problem the number of nodes, the number of arcs, and the number of commodities. The oracle is a shortest path problem solved with Dijkstra algorithm. The code is written in C. The tests were performed on a PC (Pentium IV, 2.8 GHz, 2 Gb of RAM) under Linux operating system.

Table 2 shows the numerical results to solve the linear and the nonlinear case with a relative otpimality gap less than 10^{-5}. We followed different strategies in the management of the proximal term, depending on whether the problem is linear or not. In the linear case, a constant value for the proximal parameter, say $\rho = 10^{-2}$ is suitable. In the nonlinear case, the proximal parameter is dynamically adjusted, according to success or failure in improving the value of the Lagrangian dual objective (lower bound). We start with $\rho = 1$ and multiply the current ρ by 10 in case of a 3 consecutive failures, up to the limit value $\rho = 10^{10}$.

Problem ID	# nodes	# arcs	# commodities
`planar500`	500	2842	3525
`planar800`	800	4388	12756
`planar1000`	1000	5200	20026
`grid12`	900	3480	6000
`grid13`	900	3480	12000
`grid14`	1225	4760	16000
`grid15`	1225	4760	32000
`Winnipeg`	1067	2975	4345
`Barcelona`	1020	2522	7922
`Chicago`	933	2950	93513

Table 1: Test problems.

	Linear case				Nonlinear case			
Problem ID	Outer	Inner	CPU	%Oracle	Outer	Inner	CPU	%Oracle
planar500	229	744	88.7	21	127	324	32.2	37
planar800	415	1182	557.2	16	182	429	110.5	40
planar1000	1303	2817	7846.7	12	381	869	568.1	26
grid12	509	1341	658.5	18	201	409	106.7	41
grid13	673	1629	1226.8	12	222	454	128.7	39
grid14	462	1363	843.6	22	204	414	173.2	48
grid15	520	1450	1055.1	20	203	414	172.8	48
Winnipeg	224	592	81.2	18	338	988	215.0	14
Barcelona	157	421	35.9	23	253	678	101.1	15
Chicago	180	493	79.2	47	145	370	48.6	41

Table 2: Numerical results.

The results in Table 2 have been further improved by means of column elimination and an active set strategy. With these enhancements, the method could solve huge instances with up to 40,000 arcs and 2,000,000 commodities. It has also been compared to other state-of-the-art methods. It appears to be very competitive, especially in the linear case, where it turns out to be from 4 to 30 times faster than the best known results. (For more details, see [1, 2].)

4.2 Lagrangian relaxations of the p-median problem

In the p-median problem the objective is to open p 'facilities' from a set of m candidate facilities relative to a set of n 'customers', and to assign each customer to a single facility. The cost of an assignment is the sum of the shortest distances c_{ij} from a customer to a facility. The distance is sometimes weighted by an appropriate factor, e.g., the demand at a customer node. The objective is to minimize this sum. Applications of the p-median problem can be found in cluster analysis, facility location, optimal diversity management problem, etc. [7]. The p-median problem is NP-hard [22].

The p-median problem can be formulated as follows

$$\min_{x,y} \quad \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \tag{16a}$$

$$\text{s.t.} \quad \sum_{i=1}^{m} x_{ij} = 1, \quad \forall j, \tag{16b}$$

$$\sum_{i=1}^{m} y_i = p, \tag{16c}$$

$$x_{ij} \le y_i, \quad \forall i, j, \tag{16d}$$

$$x_{ij}, y_i \in \{0,1\}, \tag{16e}$$

where $x_{ij} = 1$ if facility i serves the customer j, otherwise $x_{ij} = 0$ and $y_i = 1$ if we open facility i, otherwise $y_i = 0$.

In the following two sections we formulate the (standard) Lagrangian relaxation of the p-median problem, and the semi-Lagrangian relaxation.

4.2.1 Standard Lagrangian relaxation of the p-median problem

In this section we focus in the resolution of the (standard) Lagrangian relaxation (LR) of the p-median problems by means of Proximal-ACCPM. To this end, we relax constraints (16b) and (16c) in (16), to yield the dual problem

$$\max_{u,v} \mathcal{L}_1(u, v), \tag{17}$$

and the oracle

$$\begin{aligned} \mathcal{L}_1(u, v) \quad &= \min_{x,y} \quad \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij}x_{ij} + \sum_{j=1}^{n} u_j(1 - \sum_{i=1}^{m} x_{ij}) + v(p - \sum_{i=1}^{m} y_i) && \text{(18a)} \\ &\text{s.t.} \quad x_{ij} \le y_i, \quad \forall i, j, && \text{(18b)} \\ &\qquad\ \ x_{ij}, y_i \in \{0, 1\}, && \text{(18c)} \end{aligned}$$

where $u \in \mathbb{R}^n$ is associated to the constraints $\sum_{i=1}^{m} x_{ij} = 1, \quad j = 1, \ldots, n$, and $v \in \mathbb{R}$ to the constraint $\sum_{i=1}^{m} y_i = p$.

We name *Oracle 1* this oracle; it is trivially solvable. Its optimal solution is also optimal for its linear relaxation. Consequently, the optimum of $\mathcal{L}_1$ coincides with the optimum of the linear relaxation of (16).

To show Proximal-ACCPM performance when solving the standard Lagrangian relaxation (18), we take a few examples reported in [11]. In this technical report, several p-median problems based on data from the *traveling salesman problem* (TSP) library [28] are solved. Instances of the grid problem, where the customers are regularly spaced points on square, are also solved. In Table 3 we show the results for ten representative instances (Proximal-ACCPM stopping criterion set equal to 10^{-6}). In this case, the proximal parameter is set to $\rho = 1$ initially and is dynamically adjusted by multiplicative factors 2 and 0.5 depending on the success or failure in improving the objective of the Lagrangian dual objective. The updating is limited by the bounds 10^{-6} and 10^4. Programs have been written in MATLAB and run in a PC (Pentium-III PC, 800 MHz, with 256 Mb of RAM) under the Linux operating system.

4.2.2 Semi-Lagrangian relaxation of the p-median problem

The standard Lagrangian relaxation is commonly used in combinatorial optimization to generate lower bounds for a minimization problem. An optimal integer solution is obtained by a branch and bound scheme. The semi-Lagrangian relaxation (SLR) is a more powerful scheme, introduced in [4], that generates an

Problem ID	n	p	Outer	Inner	CPU	%Oracle
Grid1521	1521	10	348	902	132	33
Grid1849	1849	10	417	1042	241	32
Grid2025	2025	10	382	961	229	37
Grid2304	2304	10	448	1111	370	34
Grid2500	2500	10	440	1095	428	34
TSP1817	1817	10	1070	2303	1861	10
TSP2103	2103	10	316	701	156	48
TSP2152	2152	10	196	430	98	51
TSP2319	2319	10	369	775	237	46
TSP3038	3038	10	127	292	102	62

Table 3: Numerical results.

optimal integer solution for (linear) combinatorial problems with equality constraints.

To strengthen $\mathcal{L}_1$, the SLR introduces in (16) the redundant constraints $\sum_i x_{ij} \leq 1$, $j = 1, \ldots, n$, and $\sum_i y_i \leq p$. After relaxing (16b-16c), we obtain the SLR dual problem

$$\max \mathcal{L}_3(u, v), \tag{19}$$

and the new oracle

$$\begin{aligned}
\mathcal{L}_3(u,v) \quad = \min_{x,y} \quad & \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} + \sum_{j=1}^{n} u_j(1 - \sum_{i=1}^{m} x_{ij}) + v(p - \sum_{i=1}^{m} y_i) && \text{(20a)}\\
\text{s.t.} \quad & \sum_{i=1}^{m} x_{ij} \leq 1, \quad \forall j, && \text{(20b)}\\
& \sum_{i=1}^{m} y_i \leq p, && \text{(20c)}\\
& x_{ij} \leq y_i, \quad \forall i, j, && \text{(20d)}\\
& x_{ij}, y_i \in \{0, 1\}. && \text{(20e)}
\end{aligned}$$

This oracle, which we name *Oracle 3*, is much more difficult than Oracle 1 (in fact, Oracle3 is NP-hard). To cope with this difficulty one can use an intermediate oracle (*Oracle 2*) defined as the Oracle 3 but without constraint (20c). We denote $\mathcal{L}_2$ the associated dual function. In general, Oracle 2 is easier to solve than Oracle 3, especially in cases where the p-median underlying graph associated to Oracle 2 decomposes into independent subgraphs. In such situation, we solve an integer problem per subgraph (see [4] for more details).

It can be seen that solving the SLR dual problem (20) completely solves the p-median problem. Based on this result, we design a branch-and-bound free procedure to completely solve the p-median problem. This procedure successively maximizes the dual functions $\mathcal{L}_i(u, v), i = 1, 2, 3$. In this succession of three

Instance			Lower bound			Upper bound		%Opt.
Problem ID	n	p	Or. 1	Or. 2	Or. 3	Value	Method	
rl1304	1304	10	2131787.5	2133534	-	2134295	VNDS	99.96
rl1304	1304	500	97008.9	97024	-	97024	SLR	100
vm1748	1748	10	2982731.0	2983645	-	2983645	SLR	100
vm1748	1748	500	176976.2	176986	176986	176986	SLR	100
d2103	2103	10	687263.3	687321	-	687321	SLR	100
d2103	2103	500	63938.4	64006	64006	64006	SLR	100
pcb3038	3038	5	1777657.0	1777677	-	1777835	VNDS	99.99
pcb3038	3038	500	134771.8	134798	134798	136179	VNDS	98.98
fl3795	3795	150	65837.6	65868	-	65868	SLR	100
fl3795	3795	500	25972.0	25976	25976	25976	SPR	100

Table 4: Solution quality

dual problems, the optimal solution of one dual problem is used as the starting point for the next dual problem. After solving the last dual problem ($\mathcal{L}_3(u, v)$) we obtain, as a by-product, an optimal integer solution for (16). These dual problems are solved by means of Proximal-ACCPM. Oracle 2 and 3 are solved by means of CPLEX 8.1. Note that, although our procedure is branch-and-bound free, CPLEX is, of course, based on a sophisticated branch-and-bound procedure.

If we are not able to solve the three dual problems we will only have a lower bound of the p-median optimal value. In this case, we will compute an integer solution for the p-median problem by means of an heuristic as for example the 'Variable Neighborhood Decomposition Search' (VNDS) [20]. The quality or the integer solution will be determined by the dual lower bound.

In Tables 4 and 5 we show the results (solution quality and performance) for 10 representative examples of the 44 instances tested in [4]. These instances can be found in the TSPLIB [28] and range from 1304 to 3795 customers, which implies 2 to 14 million binary variables. The proximal parameter is set to the constant value $\rho = 10^{-2}$ for problems with Oracle 2 and Oracle 3. In these tables 'Or.' stands for Oracle, 'VNDS' for variable neighborhood decomposition search, 'SLR' for semi-Lagrangian relaxation and 'ANIS' for averaged number of independent subgraphs. '%Opt.' gives the quality of the solution and is computed as

$$100 \times \left(1 - \frac{\text{`Upper bound'} - \text{`Lower bound'}}{\text{`Lower bound'}}\right).$$

Programs have been written in MATLAB and run on a PC (Pentium-IV Xeon PC, 2.4 GHz, with 6 Gb of RAM) under the Linux operating system. Note that in some cases the Oracle 3 is not called. The reason is either because the problem has been completely solved by the second dual problem or the CPU time limit has been reached when solving the second dual problem.

Instance			Outer			ANIS	CPU			
Problem ID	n	p	Or. 1	Or. 2	Or. 3		Or. 1	Or. 2	Or. 3	Total
rl1304	1304	10	390	35	0	1	95	17241	0	17336
rl1304	1304	500	133	15	0	143	8	40	0	48
vm1748	1748	10	500	21	0	1	174	3771	0	3945
vm1748	1748	500	146	15	2	131	14	61	22	97
d2103	2103	10	241	7	0	2	41	504	0	545
d2103	2103	500	500	26	2	39	143	10086	4309	14538
pcb3038	3038	5	341	5	0	1	111	1988	0	2099
pcb3038	3038	500	211	17	2	38	56	3269	3900	7225
fl3795	3795	150	1000	27	0	17	1100	39199	0	40299
fl3795	3795	500	500	38	1	25	259	2531	218	3008

Table 5: Performance

4.3 Coupling economic and environmental models

Integrated assessment of environmental (IAM) policies is becoming an important priority due to the social need for local air pollution control or global climate change mitigation. Typically an IAM will combine an economic model and an environmental model to yield an evaluation of the costs and benefits associated with some environmental goals, given the technological and economic choices that are available. In this section we present a successful implementation using Proximal-ACCPM in this context.

In [21], it has been proposed to use an oracle-based method to couple an Eulerian air quality model and a techno-economic model of energy choices in an urban region. The implementation of the approach has been further developed and tested in [8]. Ozone (O_3) pollution is usually modelled in so-called Eulerian models that represent the transport of primary pollutants (typically NO_x and VOCs) and the air photochemistry under various weather conditions and for the specific topography of the region considered. These models take the form of large scale distributed parameter systems that are run over specific “weather episodes”[4]. These simulations serve to build air-quality indicators like, e.g. the *ozone concentration peak* or *the average over a threshold* (AOT) during an episode. On the other side techno-economic models are dynamic capacity expansion and production models, also called activity analysis models. A typical example is MARKAL, initially developed to represent energy-technology choices at a country level (see [12], [6]) and also adapted to the description of these choices at a city level in [13] and [14]. In a MARKAL model the planning horizon is in general defined as 9 periods of 5 years. The model finds, for specified demands in energy services, world prices of imported energy and given a gamut of technology choices, an investment plan and a production program that minimize a system-wide total

[4] For example a two-day summer sunny period which may amplify the probability of ozone peaks in green areas.

discounted cost while satisfying some pollutant emissions limits.

From this brief description of the two categories of models, the reader may realize that they belong to very different worlds. The interaction of the models in a coupling procedure can be schematized as follows. The economic model produces a vector of pollutants emissions per sector of activity. These emissions are then distributed over time and space using patterns that depend on the type of activity. For instance, global urban heating emissions are easily dispatched in space using the geographical distribution of buildings. They are also distributed in time to follow a yearly seasonal pattern. The other important cause of emission is the volume of traffic. The economic activity analysis outputs a distribution of technologies used in the traffic different modes, each mode resulting in its own level of emissions. To obtain the spatio-temporal distribution of these emissions due to traffic one resorts to a complex congestion model of traffic (EM2), that essentially computes traffic equilibria. These different sources of pollutant emissions are then combined into a spatio-temporal distribution map of emissions. The last step in the analysis consists in simulations performed with the Eulerian model to compute air quality indices on a set of critical episodes. The combination of models that eventually produces the air quality indices is complex, but at the end one can effectively compute air quality indices as a function of the global emissions of pollutants by sector of economic activity. Clearly, one cannot expect this function to be linear. Even worse, the computation may be very time consuming.

We have described a one-way interaction of the models, starting from the economic model and ending with air quality indices. Let us now describe the feedback from the air quality assessment. Indeed, one may want to limit peaks of pollution. This can be translated into upper limits on the air quality indices. We now study this reverse mechanism and show how the complete problem can be recast in the format of problem (1). Let us first schematize the economic activity analysis as the linear program

$$\min\{c^T x \mid Ax = a,\ x \geq 0\}. \tag{21}$$

We shall refer to it as the E^3 model. The economic activity x induces a vector y of pollutants emissions. This vector is indexed by sector of activity. In the paradigm of linear activity analysis, the total emission vector is assumed to be a linear function of the economic activity level, say

$$y = Bx.$$

The complex transformation of the vector y of sectorial emissions into air quality indices is represented by a vector function $\Pi(y)$. In [8] it is shown that one can compute the function value and estimate its gradient at any point y. If $\bar{\Pi}$ is the bound imposed on the air quality indices (higher indices imply lower air quality), we can represent our complex problem as the mathematical programming problem

$$\min\{c^T x \mid Ax = a,\ Bx - y = 0,\ \Pi(y) \leq \bar{\Pi},\ x \geq 0\}. \tag{22}$$

This large-scale highly nonlinear model is intractable by standard optimization tool. However, it is quite easily amenable to an Oracle Based Optimization approach. To this end, we introduce the function

$$f(y) = \min\{c^T x \mid Ax = a,\ Bx = y,\ x \geq 0\}, \tag{23}$$

and the set

$$Y = \{y \mid \Pi(y) \leq \bar{\Pi}\}. \tag{24}$$

Our original problem can now be written as

$$\min\{f(y) \mid y \in Y\}.$$

It remains to show that the above problem is of the same type as (1). It is a well-known fact of convex analysis that the function $f(y)$ is convex (this is easily seen by considering the dual of the linear program that defines f) and that one can compute a subgradient at each point of the domain of the function. Unfortunately, one cannot make a similar statement on Y. Being the result of such a complex transformation process, $\Pi(y)$ is likely to be nonconvex. However, one can hope that in the range of values that are of interest the nonconvexity is mild. This is supported by empirical evidence. A gradient is also estimated by a finite difference scheme.

Even in presence of mild nonconvexity, one cannot exclude pathology in running Proximal-ACCPM. A separating hyperplane for the set Y may turn out to cut off part of the set, and exclude a point that was proved to be feasible earlier. To cope with this difficulty, the authors of [8] simply shifted the plane to maintain feasibility. They also made problem (23) easier by assuming monotonicity that made it possible to replace the equality constraint $Bx = y$ by $Bx \leq y$.

As the air chemistry description actually involves nonlinear functions, we have implemented a technique of successive local linearization of the air pollution dynamic equations. The details of the implementation are given in [8]. In a particular simulation based on data describing the Geneva (Switzerland) region, a solution to the reduced order optimization problem is obtained through Proximal-ACCPM, with 30 calls to the oracles (24 feasibility cuts and 6 optimality cuts were peformed). A feasibility cut (call to the air quality oracle) takes 30 minutes computing time (SUN Ultra-80, Ultrasparc driver) whereas an optimality cut (call to the techno-economic model) takes 10 seconds.

This application demonstrates the possibilities offered by an OBO method to tackle Integrated Assessment Models where part of the modeling is a large-scale simulator of complex physics and chemistry processes. Since Proximal-ACCPM

keeps the number of oracle calls to a small or moderate size it permits the use of these simulators in the design of some oracles and therefore it realizes the coupling that is the essence of IAMs.
Remark A similar implementation has been realized recently for an IAM of climate change policies. It is reported in [9, 10]. In that case the coupling is realized between an economic growth model and an intermediate complexity climate model. This second successful experience that we will not further described here confirms the potential of OBO techniques for the exploitation of complex and large-scale IAMs.

5 Conclusion

In that paper we have presented Proximal-ACCPM, an efficient method for convex nondifferentiable optimization, and discussed three large-scale applications that are representative of an oracle based optimization approach. Our presentation of Proximal-ACCPM focuses on the necessary information for an efficient implementation. It also includes recent extensions, in particular an explicit treatment of second-order information when this information is available. The three examples we selected have recently been reported in the literature. They are genuinely very large-scale problems. The first two are solved using a classical transformation known as Lagrangian relaxation. The transformed problem has much smaller dimension, thousands of variables instead of millions, but one can only collect information about it via a first-order oracle. It is shown that Proximal-ACCPM is powerful enough to solve huge instances of these problems. The third application fully exploits the concept of oracle based optimization to organize a dialog between two large-scale models that have totally different natures, a techno-economic model and a large-scale simulator of complex physics and chemistry processes. The exchanges between the two models are performed through few variables and each model is treated as a first-order oracle vis-à-vis these variables. These oracles, and especially the simulator, are computationally costly. To make the OBO approach successful, one needs a method that keeps the number of calls to the oracles as low as possible. Proximal-ACCPM does the job.

References

[1] F. Babonneau, O. du Merle, and J.-P. Vial. Solving large scale linear multicommodity flow problems with an active set strategy and Proximal-ACCPM. Technical report, Logilab, Hec, University of Geneva. To appear in *Operations Research*, 2005.

[2] F. Babonneau and J.-P. Vial. ACCPM with a nonlinear constraint and an active set strategy to solve nonlinear multicommodity flow problems. Technical report, Logilab, Hec, University of Geneva, June 2005.

[3] C. Barnhart, E.L. Johnson, G.L. Nemhauser, M.W.P. Savelsbergh, and P.H. Vance. Branch-and-price: Column generation for solving huge integer programs. *Operations Research*, 46(3):316–329, 1998.

[4] C. Beltran, C. Tadonki, and J.-P. Vial. Semi-lagrangian relaxation. Technical report, Logilab, HEC, University of Geneva, 2004.

[5] J.F. Benders. Partitioning procedures for solving mixed-variables programming problems. *Computational Management Science*, 2:3–19, 2005. Initially appeared in *Numerische Mathematik*, 4: 238-252, 1962.

[6] C. Berger, R. Dubois, A. Haurie, E. Lessard, R. Loulou, and J.-P. Waaub. Canadian MARKAL: An advanced linear programming system for energy and environmental modelling. *INFOR*, 30(3):222–239, 1992.

[7] O. Briant and D. Naddef. The optimal diversity management problem. *Operations research*, 52(4), 2004.

[8] D. Carlson, A. Haurie, J.-P. Vial, and D.S. Zachary. Large scale convex optimization methods for air quality policy assessment. *Automatica*, 40:385–395, 2004.

[9] L. Drouet, C. Beltran, N.R. Edwards, A. Haurie, J.-P. Vial, and D.S. Zachary. An oracle method to couple climate and economic dynamics. In A. Haurie and L. Viguier, editors, *Coupling climate and economic dynamics*. Kluwer, to appear in 2005.

[10] L. Drouet, N.R. Edwards, and A. Haurie. Coupling climate and economic models in a cost-benefit framework: A convex optimization approach. *Environmental Modeling and Assessment*, to appear.

[11] O. du Merle and J. P. Vial. Proximal ACCPM, a cutting plane method for column generation and Lagrangian relaxation: application to the p-median problem. Technical report, Logilab, University of Geneva, 40 Bd du Pont d'Arve, CH-1211 Geneva, Switzerland, 2002.

[12] L.G. Fishbone and H. Abilock. MARKAL, a linear programming model for energy systems analysis: Technical description of the BNL version. *International Journal of Energy Research*, 5:353–375, 1981.

[13] E. Fragnière and A. Haurie. MARKAL-Geneva: A model to assess energy-environment choices for a Swiss Canton. In C. Carraro and A. Haurie,

editors, *Operations Research and Environmental Management*, volume 5 of *The FEEM/KLUWER International Series on Economics, Energy and Environment*. Kluwer Academic Publishers, 1996.

[14] E. Fragnière and A. Haurie. A stochastic programming model for energy/environment choices under uncertainty. *International Journal of Environment and Pollution*, 6(4-6):587–603, 1996.

[15] A.M. Geoffrion. Lagrangean relaxation for integer programming. *Mathematical Programming Study*, 2:82–114, 1974.

[16] J.-L. Goffin, Z. Q. Luo, and Y. Ye. Complexity analysis of an interior point cutting plane method for convex feasibility problems. *SIAM Journal on Optimization*, 69:638–652, 1996.

[17] J.-L. Goffin and J.-P. Vial. Shallow, deep and very deep cuts in the analytic center cutting plane method. *Mathematical Programming*, 84:89–103, 1999.

[18] J.-L. Goffin and J.-P. Vial. Convex nondifferentiable optimization: A survey focussed on the analytic center cutting plane method. *Optimization Methods and Software*, 174:805–867, 2002.

[19] M. Guignard and S. Kim. Lagrangean decomposition: a model yielding stronger Lagrangean bounds. *Mathematical Programming*, 39:215–228, 1987.

[20] P. Hansen, N. Mladenovic, and D. Perez-Brito. Variable neighborhood decomposition search. *Journal of Heuristics*, 7:335–350, 2001.

[21] A. Haurie, J. Kübler, A. Clappier, and H. van den Bergh. A metamodeling approach for integrated assessment of air quality policies. *Environmental Modeling and Assessment*, 9:1–122, 2004.

[22] O. Kariv and L Hakimi. An algorithmic approach to network location problems. ii: the p-medians. *SIAM Journal of Applied Mathematics*, 37(3):539–560, 1979.

[23] C. Lemaréchal. Nondifferentiable optimization. In G.L. Nemhauser, A.H.G Rinnooy Kan, and M.J. Todd, editors, *Handbooks in Operations Research and Management Science*, volume 1, pages 529–572. North-Holland, 1989.

[24] Y. Nesterov. Complexity estimates of some cutting plane methods based on the analytic center. *Mathematical Programming*, 69:149–176, 1995.

[25] Y. Nesterov. *Introductory Lectures on Convex Optimization, a Basic Course*, volume 87 of *Applied Optimization*. Kluwer Academic Publishers, 2004.

[26] Y. Nesterov and A. Nemirovsky. *Interior Point Polynomial Algorithms in Convex Programming: Theory and Applications*. SIAM, Philadelphia, Penn., 1994.

[27] Y. Nesterov and J.-P. Vial. Homogeneous analytic center cutting plane methods for convex problems and variational inequalities. *SIAM Journal on Optimization*, 9:707–728, 1999.

[28] G. Reinelt. Tsplib, 2001. http://www.iwr.uni-heidelberg.de / groups / comopt / software / TSPLIB95.