Optimization Methods and Software
Publication details, including instructions for authors and subscription information:
http://www.informaworld.com/smpp/title~content=t713645924

Concave programming for finding sparse solutions to problems with convex constraints

F. Rinaldi[a]
[a] Dipartimento di Informatica e Sistemistica, Sapienza Università di Roma, Roma, Italy

First published on: 26 August 2010

To cite this Article Rinaldi, F.(2010) 'Concave programming for finding sparse solutions to problems with convex constraints', Optimization Methods and Software,, First published on: 26 August 2010 (iFirst)
To link to this Article: DOI: 10.1080/10556788.2010.511668
URL: http://dx.doi.org/10.1080/10556788.2010.511668

Optimization Methods & Software
iFirst, 2010, 1–22

Taylor & Francis
Taylor & Francis Group

Concave programming for finding sparse solutions to problems with convex constraints

F. Rinaldi*

Dipartimento di Informatica e Sistemistica, Sapienza Università di Roma, Via Ariosto 25, 00185 Roma, Italy

(*Received 29 September 2009; final version received 26 July 2010*)

In this work, we consider a class of nonlinear optimization problems with convex constraints with the aim of computing sparse solutions. This is an important task arising in various fields such as machine learning, signal processing, and data analysis. We adopt a concave optimization-based approach, we define an effective version of the Frank–Wolfe algorithm, and we prove the global convergence of the method. Finally, we report numerical results on test problems showing both the effectiveness of the concave approach and the efficiency of the implemented algorithm.

Keywords: zero-norm; concave programming; Frank–Wolfe method

1. Introduction

We consider a class of constrained non-smooth optimization problems of the form:

$$\begin{aligned} &\min_{x\in R^n} g(x) + \lambda\|x\|_0 \\ &x \in C, \end{aligned} \tag{1}$$

where $\lambda > 0$, C is a compact convex set, g is a continuously differentiable function, and $\|x\|_0$ is the *zero-norm* of x defined as

$$\|x\|_0 = \text{card}\{x_i : x_i \neq 0\}.$$

The zero-norm is a non-convex discontinuous function; so dealing with it is a very hard task. The problem (1) is quite general and includes as special cases a wide variety of problems arising from different fields (e.g. machine learning, signal processing, and data analysis).

In machine learning, for instance, an interesting problem that can be formulated as in (1) is the sparse linear discriminant analysis (SLDA) [26]. Given a pair of symmetric matrices:

(i) *between-class* covariance matrix: A positive semi-definite;
(ii) *within-class* covariance matrix: B positive definite;

*Email: rinaldi@dis.uniroma1.it

ISSN 1055-6788 print/ISSN 1029-4937 online

DOI: 10.1080/10556788.2010.511668
http://www.informaworld.com

in SLDA, we want to find a sparse vector x which maximizes a class-separability criterion defined by the *generalized* Rayleigh quotient:

$$R(x; A, B) = \frac{x^{\mathrm{T}} A x}{x^{\mathrm{T}} B x}.$$

Namely, we want to solve the following optimization problem:

$$\begin{aligned} &\min_{x \in R^n} -\frac{1}{2} x^{\mathrm{T}} A x + \lambda \|x\|_0 \\ &x^{\mathrm{T}} B x \leq 1. \end{aligned} \tag{2}$$

Sparse principal component analysis (SPCA) is a well-known problem in data analysis [4,11,31]. In SPCA, given a (symmetric positive semi-definite) covariance matrix C, the goal is finding a sparse vector x which explains the maximum amount of variance. The zero-norm formulation related to this problem is

$$\begin{aligned} &\min_{x \in R^n} -\frac{1}{2} x^{\mathrm{T}} C x + \lambda \|x\|_0 \\ &x^{\mathrm{T}} x \leq 1. \end{aligned} \tag{3}$$

In signal analysis, a widely-studied problem is the sparse representation of signals [3,8]. Many media types (i.e. imagery, video and audio) can be sparsely represented using transform-domain methods, and in fact, various relevant problems dealing with such media can be easily viewed as the problem of finding sparse solutions to a linear system. Sparsity of representation is a key aspect in transform-based image compression [5,6,13], signal and image denoising [14–17], and image deblurring [18–20]. In practice, given a dictionary $A \in R^{m \times n}$ of elementary signals and a real noisy signal b, the goal is finding a sparse representation x of signal b in terms of the dictionary A. This problem can be formulated as follows:

$$\begin{aligned} &\min_{x \in R^n} \|x\|_0 \\ &\|Ax - b\|^2 \leq \delta, \end{aligned} \tag{4}$$

where δ is a fixed error tolerance.

In order to make problem (1) tractable, a simple approach can be that of replacing the zero-norm with the ℓ_1 norm [3,28,31], thus obtaining the problem

$$\begin{aligned} &\min_{x \in R^n} g(x) + \lambda \|x\|_1 \\ &x \in C, \end{aligned} \tag{5}$$

which can be efficiently solved even when the dimension of the problem is large. However, some experiments reported in [2,27] show that a concave optimization-based approach, for the special case of a polyhedral feasible set, performs better than the ℓ_1 norm-based one.

In this paper, inspired by the idea developed in [25,27,30], we propose a concave programming approach for solving problem (1). We replace the zero-norm with a separable concave function thus obtaining the following formulation:

$$\begin{aligned} &\min_{x \in R^n} g(x) + \lambda \sum_{j=1}^{n} h_j(x_j, u) \\ &x \in C, \end{aligned} \tag{6}$$

where $h_j : R \to R$, for $j = 1, \ldots, n$ are concave, continuously differentiable functions depending on a vector $u \in R^m$ of parameters.

In [25], Mangasarian first proposed a Frank–Wolfe (FW) type algorithm, the well-known successive linear approximation (SLA) algorithm, for minimizing a concave function over a polyhedral set. The SLA algorithm has also been used for constructing kernel classifiers that use a minimal number of data points in both generating and characterizing a classifier [22]. Two FW-based approaches for minimizing a concave function over a polyhedral set have also been proposed in [27,30]. We define a modified version of the FW algorithm to minimize a concave function over a compact convex set, namely to solve problems of the form (6), in which variables that are null at an iteration are eliminated for all the following ones, with significant savings in computational time. The algorithm involves a sequence of convex programs (that can be efficiently solved by existing solvers), and this makes possible its application to large dimensional problems.

The paper is organized as follows. In Section 2, we describe various smooth concave functions that can be used in place of the zero-norm when searching for sparse solutions to problems with convex constraints. In Section 3, we report an interesting result related to convex programming. In Section 4, after a brief review of the well-known FW method, we derive some new theoretical results, which have an important impact on the computational efficiency of the method. These results suggest the definition of a version of the method that eliminates the variables set to zero, thus allowing for a dimensionality reduction, which greatly increments the speed of the procedure. We formally prove, by means of the result reported in Section 3, the global convergence of this modified version of the FW method. In Section 5, we describe a version of the reduced FW algorithm with unitary stepsize that can be used when the problem we want to solve has a concave objective function. Finally, in Section 6, we report our numerical experience on various test problems. The results obtained show both the usefulness of the new concave formulations and the efficiency in terms of computational time of the implemented minimization algorithm.

2. Concave formulations for finding a sparse vector over a convex set

Consider the general problem of finding a vector belonging to a compact convex set C and having the minimum number of non-zero components, that is

$$\begin{aligned} &\min_{x \in R^n} \|x\|_0 \\ &x \in C. \end{aligned} \tag{7}$$

Since the objective function in (7) is discontinuous, we can use a continuously differentiable, concave function that somehow approximates the behaviour of the zero-norm function. A similar approach has already been proposed in [25,27,30] for finding sparse solutions to linear systems. In order to illustrate the idea underlying the concave approach, we observe that the objective function of problem (7) can be written as follows:

$$\|x\|_0 = \sum_{i=1}^{n} s(|x_i|),$$

where $s : R \to R^+$ is the *step function* such that $s(t) = 1$ for $t > 0$ and $s(t) = 0$ for $t \leq 0$. Following the approach described in [25], we replace the discontinuous step function by a continuously differentiable concave function $v(t) = 1 - e^{-\alpha t}$, with $t \geq 0$ and $\alpha > 0$, thus obtaining a problem

of the form

$$\begin{aligned} &\min_{x,y\in R^n} \sum_{i=1}^{n}(1-e^{-\alpha y_i}) \\ &x \in C \\ &-y_i \le x_i \le y_i \quad i=1,\ldots,n. \end{aligned} \tag{8}$$

The approach is well-motivated from a theoretical point of view. In fact, for $y \ge 0$, it is easy to see that,

$$\lim_{\alpha\to\infty} \sum_{i=1}^{n}(1-e^{-\alpha y_i}) = \|y\|_0,$$

so the objective function is a smooth approximation of the zero-norm. Another way to solve problem (7) can be that of using the logarithm function instead of the step function [30], and this leads to a concave smooth problem of the form

$$\begin{aligned} &\min_{x,y\in R^n} \sum_{i=1}^{n}\ln(\epsilon+y_i) \\ &x \in C \\ &-y_i \le x_i \le y_i \quad i=1,\ldots,n, \end{aligned} \tag{9}$$

with $0 < \epsilon \ll 1$. Formulation (9) is practically motivated by the fact that, due to the form of the logarithm function, it is better to increase one variable y_i while setting to zero another one rather than doing some compromise between both, and this should facilitate the computation of a sparse solution. The following two concave formulations, related to the ideas underlying (8) and (9) respectively, have been proposed in [27] for finding a sparse solution to a linear system:

$$\begin{aligned} &\min_{x\in R^n, y\in R^n} \sum_{i=1}^{n}(y_i+\epsilon)^p \\ &x \in C \\ &-y_i \le x_i \le y_i \quad i=1,\ldots,n \end{aligned} \tag{10}$$

with $0 < p < 1$, and $0 < \epsilon$;

$$\begin{aligned} &\min_{x\in R^n, y\in R^n} -\sum_{i=1}^{n}(y_i+\epsilon)^{-p} \\ &x \in C \\ &-y_i \le x_i \le y_i \quad i=1,\ldots,n \end{aligned} \tag{11}$$

with $1 \le p$, and $0 < \epsilon$.

3. Convex programming: a new result

In this section, we generalize some results about the big-M method [9,10] to the case of convex programming. These results will be used to derive the convergence of a modified FW method we will present.

PROPOSITION 1 *Consider the convex programming problems*

$$\begin{aligned} &\min f(x) \\ &s(x) \leq 0 \\ &Ax = b \end{aligned} \tag{12}$$

$$\begin{aligned} &\min f(x) + Me^{\mathrm{T}}z \\ &s(x) + w(z) \leq 0 \\ &Ax + Qz = b \\ &z \geq 0, \end{aligned} \tag{13}$$

where

(1) $e \in R^{n_z}$ *is a vector of ones*, $b \in R^p$, $A \in R^{p\times n}$, $Q \in R^{p\times n_z}$;
(2) $f : R^n \to R$ *is a convex, continuously differentiable function*;
(3) $s : R^n \to R^m$ *is a convex, continuously differentiable function*;
(4) $w : R^{n_z} \to R^m$ *is a convex, continuously differentiable function such that* $w(0) = 0$.

Assume that problem (12) *admits a solution* $x^\star$, *and that there exists a feasible vector* $\bar{x}$ *satisfying the following condition*:

$$s(\bar{x}) < 0. \tag{14}$$

Then there exists a value M^0 *such that for all* $M \geq M^0$, *we have that*

(i) *the vector* $(x^\star, 0)^{\mathrm{T}}$ *is a solution of* (13);
(ii) *if* $(\tilde{x}, \tilde{z})^{\mathrm{T}}$ *is a solution of* (13), *then* $\tilde{z} = 0$ *and* $\tilde{x}$ *is a solution of* (12).

Proof (i) Since $x^\star$ is a solution of problem (12), we have from Proposition A2 in the appendix that there exist Lagrange multipliers $\lambda^\star \in R^m$ and $\mu^\star \in R^p$ satisfying conditions (A2) in the appendix.

Now consider problem (13) and the Karush–Kuhn–Tucker system related to it:

$$\begin{aligned} \nabla f(x) + \nabla s(x)\lambda - A^{\mathrm{T}}\mu &= 0 \\ Me + \nabla w(z)\lambda - Q^{\mathrm{T}}\mu - \tau &= 0 \\ \lambda^{\mathrm{T}}[s(x) + w(z)] &= 0 \\ \tau^{\mathrm{T}}z &= 0 \\ z, \lambda, \tau &\geq 0. \end{aligned} \tag{15}$$

Since $w(0) = 0$, the vector $(x^\star, 0)^{\mathrm{T}}$ is a feasible point for (13), and, as for M sufficiently large we have

$$-\nabla w(0)\lambda^\star + Q^{\mathrm{T}}\mu^\star \leq Me,$$

it is possible to find a vector $\tau^\star \geq 0$ such that the vector $(x^\star, 0, \lambda^\star, \mu^\star, \tau^\star)^{\mathrm{T}}$ is a solution of (15). Thus, from Proposition A3 in the appendix, we have that $(x^\star, 0)$ is a global optimum of problem (13) and the assertion is proved.

(ii) By contradiction let us assume that there exists a sequence of positive scalars $\{M^k\}$, with $M^k \to \infty$ for $k \to \infty$, and a corresponding sequence of vectors $\{(x^k, z^k)^{\mathrm{T}}\}$ such that $z^k \neq 0$, and

$(x^k, z^k)^{\mathrm{T}}$ is solution of (13) when $M = M^k$. We can then define an infinite subset K such that, for all $k \in K$ we have $z_i^k > 0$ for some index $i \in \{1, \ldots, n_z\}$. For all k, we can write

$$f(x^k) + M^k e^{\mathrm{T}} z^k \geq f(x^k) + M^k e^{\mathrm{T}} z^k + \lambda^{\star\mathrm{T}}[s(x^k) + w(z^k)] + \mu^{\star\mathrm{T}}(b - Ax^k - Qz^k). \quad (16)$$

By convexity of f, s and w, we have

$$f(x^k) \geq f(x^\star) + \nabla f(x^\star)^{\mathrm{T}}(x^k - x^\star) \quad (17)$$

$$s(x^k) \geq s(x^\star) + \nabla s(x^\star)^{\mathrm{T}}(x^k - x^\star) \quad (18)$$

$$w(z^k) \geq w(0) + \nabla w(0)^{\mathrm{T}} z^k. \quad (19)$$

Using (17)–(19), we obtain

$$\begin{aligned} f(x^k) + M^k e^{\mathrm{T}} z^k &\geq f(x^k) + M^k e^{\mathrm{T}} z^k + \lambda^{\star\mathrm{T}}[s(x^k) + w(z^k)] + \mu^{\star\mathrm{T}}(b - Ax^k - Qz^k) \\ &\geq f(x^\star) + \nabla f(x^\star)^{\mathrm{T}}(x^k - x^\star) + \lambda^{\star\mathrm{T}}[s(x^\star) + \nabla s(x^\star)^{\mathrm{T}}(x^k - x^\star) + w(0) \\ &\quad + \nabla w(0)^{\mathrm{T}} z^k] + \mu^{\star\mathrm{T}}(b - Ax^k + Ax^\star - Ax^\star) - \mu^{\star\mathrm{T}} Q z^k + M^k e^{\mathrm{T}} z^k \\ &= f(x^\star) + [\nabla f(x^\star) + \nabla s(x^\star)\lambda^\star - A^{\mathrm{T}}\mu^\star]^{\mathrm{T}}(x^k - x^\star) \\ &\quad + \lambda^{\star\mathrm{T}}[s(x^\star) + w(0) + \nabla w(0)^{\mathrm{T}} z^k] + \mu^{\star\mathrm{T}}(b - Ax^\star) - \mu^{\star\mathrm{T}} Q z^k + M^k e^{\mathrm{T}} z^k. \end{aligned} \quad (20)$$

Since $w(0) = 0$, the vector $(x^\star, 0)^{\mathrm{T}}$ is a feasible point for (13), and, as for M^k sufficiently large we can find a vector $\tau^{\star k} \geq 0$ such that the vector $(x^\star, 0, \lambda^\star, \mu^\star, \tau^{\star k})^{\mathrm{T}}$ is a solution of (15), we rewrite (16) as follows:

$$f(x^k) + M^k e^{\mathrm{T}} z^k \geq f(x^\star) + [M^k e + \nabla w(0)\lambda^\star - Q^{\mathrm{T}}\mu^\star]^{\mathrm{T}} z^k = f(x^\star) + \tau^{\star k\mathrm{T}} z^k. \quad (21)$$

Furthermore, there exists a value $\tilde{M}$ such that $\forall\ M^k \geq \tilde{M}$, we have ${\tau_i}^{\star k} > 0$, and, as $z_i^k > 0$, we have

$$f(x^k) + M^k e^{\mathrm{T}} z^k \geq f(x^\star) + \tau^{\star k\mathrm{T}} z^k > f(x^\star) \quad (22)$$

but this contradicts the fact that $(x^k, z^k)^{\mathrm{T}}$ is optimum for problem (13).

Now we need to show that, for a given value $M \geq \tilde{M}$, if $(\tilde{x}, \tilde{z})^{\mathrm{T}}$, with $\tilde{z} = 0$, is a solution of (13), then $\tilde{x}$ is a solution of (12). As $(\tilde{x}, \tilde{z})^{\mathrm{T}}$ is a solution of (13), we can write

$$f(\tilde{x}) = f(\tilde{x}) + Me^{\mathrm{T}}\tilde{z} \leq f(x) + Me^{\mathrm{T}} z$$

for each $(x, z)^{\mathrm{T}}$ feasible point of problem (13). Furthermore, it is easy to see that a vector x is a feasible point of (12) iff $(x, 0)$ is a feasible point of (13). Then, we have

$$f(\tilde{x}) \leq f(x)$$

for each x feasible point of problem (12). ■

4. The FW – reduced dimension algorithm

The FW algorithm is a well-known algorithm in operations research. It was originally proposed by Marguerite Frank and Phil Wolfe in 1956 as a procedure for solving quadratic programming

problems with linear constraints [21]. Extensive discussion of its application to more general problems are given in [1,12,24].

In this section, we first describe the algorithm and give some results about its convergence to a stationary point. Then we propose a new efficient version of the FW algorithm for solving problems of the following form:

$$\begin{aligned} &\min f(x) = g(x) + h(x) = g(x) + \sum_{j=1}^{n} h_j(x_j) \\ &x \in C \\ &x_i \geq 0, \quad i \in I \subseteq \{1, \ldots, n\}, \end{aligned} \tag{23}$$

where:

(i) C is a compact set having the following form:

$$C = \left\{ x \in R^n : s_l(x_{\bar{I}}) + \sum_{i \in I} w_{li}(x_i) \leq 0, \quad l = 1, \ldots, m; \quad Ax = b \right\}, \tag{24}$$

where $A \in R^{p \times n}$, $x_{\bar{I}} = \{x_i : i \notin I\}$, $s_l : R^{n-|I|} \to R$, and $w_{li} : R \to R$, for $l = 1, \ldots, m$ and $i \in I$, are convex, continuously differentiable functions;

(ii) $g : R^n \to R$ is a continuously differentiable function;

(iii) $h_j : R \to R$, for $j = 1, \ldots, n$ are concave, continuously differentiable functions.

We further assume that $w_{li}(0) = 0$ for $l = 1, \ldots, m$ and $i \in I$.

In order to give a better explanation of problem (23), we link the notation used above to one of the real problems described in the first section. For instance, we can consider problem (4) and replace the zero norm by the concave exponential approximation as in (8), thus obtaining a concave problem of the following form:

$$\begin{aligned} &\min_{x,y \in R^n} \sum_{i=1}^{n} (1 - e^{-\alpha y_i}) \\ &\|Ax - b\|^2 \leq \delta \\ &-y_i \leq x_i \leq y_i \quad i = 1, \ldots, n. \end{aligned} \tag{25}$$

Then, if we view problem (25) as problem (23), we can write:

- $g(x) = 0$;
- $h_i(y_i) = 1 - e^{-\alpha y_i}$, $i = 1, \ldots, n$;
- $s_1(x) = \|Ax - b\|^2 - \delta$, $w_{1i}(y_i) = 0$, $i = 1, \ldots, n$;
- $s_{1+j}(x) = x_j$, $w_{1+ji}(y_i) = \begin{cases} 0 & i \neq j \\ -y_i & i = j \end{cases}$ $i, j = 1, \ldots, n$;
- $s_{1+n+j}(x) = -x_j$, $w_{1+n+ji}(y_i) = \begin{cases} 0 & i \neq j \\ -y_i & i = j \end{cases}$ $i, j = 1, \ldots, n$.

Herein, we report the FW algorithm for minimizing a continuously differentiable function over a compact convex set:

4.1 *FW algorithm*

(1) Let $x^0 \in C$ be the starting point.
(2) For $k = 0, 1, \ldots$ obtain solution $\overline{x}^k$ by solving the following problem:

$$\overline{x}^k = \arg\min_{x \in C} \nabla f(x^k)^{\mathrm{T}}(x - x^k). \tag{26}$$

(3) If $\nabla f(x^k)^{\mathrm{T}}(\overline{x}^k - x^k) = 0$ then STOP.
(4) Otherwise, define a feasible descent direction

$$d^k = \overline{x}^k - x^k$$

and generate a new feasible vector

$$x^{k+1} = x^k + \alpha^k d^k$$

with $\alpha^k \in (0, 1]$ determined by means of an Armijo-like rule.

We want to remark that the optimization problem to be solved at Step 2 has two important features:

(1) linear objective function;
(2) compact convex set.

So that, when our problem has linear and quadratic constraints, it can be efficiently solved by available solvers (e.g. CPLEX) even when the dimensions are very large.

The following result, proved in [1], provides an analysis of convergence behavior of the FW algorithm.

PROPOSITION 2 *Let $\{x^k\}$ be a sequence generated by the FW algorithm*

$$x^{k+1} = x^k + \alpha^k d^k.$$

Assume that the method used for choosing stepsize α^k satisfies the following conditions:

(i) $f(x^{k+1}) < f(x^k)$, with $\nabla f(x^k) \neq 0$;
(ii) if $\nabla f(x^k) \neq 0 \; \forall \, k$, then we have

$$\lim_{k \to \infty} \nabla f(x^k)^{\mathrm{T}} d^k = 0.$$

Then every limit point $\bar{x}$ of $\{x^k\}$ is a stationary point.

The next proposition shows that, under suitable conditions on the concave functions h_j, the FW algorithm does not change a non-negative variable once that it has been fixed to zero.

PROPOSITION 3 *Let $\{x^k\}$ be any sequence generated by the FW algorithm. There exists a value M such that, if $i \in I$ and*

$$h_i^{'}(0) \geq M$$

then we have that

$$x_i^k = 0 \quad \text{implies} \quad x_i^{k+1} = 0.$$

Proof At each iteration k of the FW algorithm, the problem to be solved is

$$\begin{aligned}
&\min \sum_{j=1}^{n} \nabla g_j(x^k) x_j + \sum_{j:x_j^k \neq 0} h_j'(x_j^k) x_j + \sum_{j \notin I: x_j^k = 0} h_j'(0) x_j + \sum_{j \in I: x_j^k = 0} h_j'(0) x_j \\
&x \in C \\
&x_i \geq 0, \quad i \in I \subseteq \{1, \ldots, n\}.
\end{aligned} \tag{27}$$

Let $\bar{x}^k$ be a solution of (27). As g is continuously differentiable and C is compact, there exists a value $L < \infty$ such that

$$\|\nabla g(x)\|_\infty \leq L \quad \forall x \in C. \tag{28}$$

For any $i \in I$ such that $x_i^k = 0$, by (ii) of Proposition 1 it follows that there exists a value S such that if $\nabla g_i(x^k) + h_i'(0) \geq S$ then we have $\bar{x}_i^k = 0$. Thus, if $i \in I, x_i^k = 0$ and $h_i'(0) \geq M = S + L$, then we obtain

$$x_i^{k+1} = x_i^k + \alpha^k(\bar{x}_i^k - x_i^k) = 0.$$

■

On the basis of Proposition 3, we can define the following version of the FW algorithm, where the convex problems to be solved are of reduced dimension. We denote by Ω the feasible set of problem (23), i.e.

$$\Omega = \{x \in R^n : x \in C, x_i \geq 0, i \in I\}.$$

4.2 *FW – reduced dimension (FW-RD) algorithm*

(1) Let $x^0 \in C$ be the starting point.
(2) For $k = 0, 1, \ldots$, let $I^{x^k} = \{i \in I : x_i^k = 0\}$ and $C^{x^k} = \{x \in \Omega : x_i = 0, \forall i \in I^{x^k}\}$ obtain solution $\overline{x}^k$ by solving the following problem:

$$\overline{x}^k = \arg \min_{x \in C^{x^k}} \nabla f(x^k)^{\mathrm{T}}(x - x^k). \tag{29}$$

(3) If $\nabla f(x^k)^{\mathrm{T}}(\overline{x}^k - x^k) = 0$ then STOP.
(4) Otherwise, define a feasible descent direction

$$d^k = \overline{x}^k - x^k$$

and generate a new feasible vector

$$x^{k+1} = x^k + \alpha^k d^k$$

with $\alpha^k \in (0, 1]$ determined by means of an Armijo-like rule.

Note that the convex programming problem (29) is equivalent to a convex problem of dimension $n - |I^{x^k}|$, and that $I^{x^k} \subseteq I^{x^{k+1}}$, so that the problems to be solved are of non-increasing dimensions. This yields obvious advantages in terms of computational time.

The following technical result will be used in the proof of the convergence of the modified FW method:

PROPOSITION 4 *Let $x^k \to x$ be a sequence of points in Ω such that*

$$C^{x^{k+1}} \subseteq C^{x^k} \tag{30}$$

and

$$C^x \subseteq C^{x^k}. \tag{31}$$

Then, for any $y \in C^x$, there exists a sequence $\{y^k\}$ converging to y, with $y^k \in C^{x^k}$.

Proof By using (30) and (31), we obtain

$$y \in C^{x^k}. \tag{32}$$

Since $x^k \in \Omega$, it follows that $x^k \in C^{x^k}$. Let us now consider the following sequence:

$$y^k = \lambda^k y + (1 - \lambda^k)x^k,$$

where $\{\lambda^k\}$ is a sequence converging to 1, with $\lambda^k \leq 1$. As C^{x^k} is a compact convex set, we have

$$y^k \in C^{x^k}$$

and

$$y^k \longrightarrow y.$$

■

Now, we can formally prove the convergence of the proposed algorithm to a stationary point.

PROPOSITION 5 *Let $\{x^k\}$ be a sequence generated by the FW-RD Algorithm*

$$x^{k+1} = x^k + \alpha^k d^k.$$

Assume that method used for choosing stepsize α^k satisfies the following conditions:

(i) $f(x^{k+1}) < f(x^k)$, with $\nabla f(x^k) \neq 0$;
(ii) *if* $\nabla f(x^k) \neq 0 \; \forall \, k$, then we have

$$\lim_{k \to \infty} \nabla f(x^k)^{\mathrm{T}} d^k = 0.$$

Suppose there exists a value S such that $h_i'(0) \geq S \; \forall x_i = 0$ with $i \in I$, then every limit point $\bar{x}$ of $\{x^k\}$ is a stationary point.

Proof As we assumed compactness of C, a limit point $\bar{x} \in C$ exists and the norm of vector d^k is bounded above

$$\|d^k\| = \|\bar{x}^k - x^k\| \leq \|\bar{x}^k\| + \|x^k\|.$$

We can now define a subsequence $\{x^k\}_K$ such that

$$\lim_{k\to\infty,k\in K} x^k = \bar{x}, \qquad \lim_{k\to\infty,k\in K} d^k = \bar{d}.$$

By using hypothesis (ii), we obtain

$$\nabla f(\bar{x})^{\mathrm{T}}\bar{d} = 0.$$

Let d^k be a direction generated by the FW method; we have

$$\nabla f(x^k)^{\mathrm{T}} d^k \leq \nabla f(x^k)^{\mathrm{T}}(x - x^k), \quad \forall\, x \in C^{x^k}. \tag{33}$$

We want to show that, by taking the limit as $k \in K$, $k \to \infty$, we obtain

$$0 = \nabla f(\bar{x})^{\mathrm{T}}\bar{d} \leq \nabla f(\bar{x})^{\mathrm{T}}(x - \bar{x}), \quad \forall\, x \in C^{\bar{x}}.$$

By contradiction, let us assume that there exists a point $\tilde{s} \in C^{\bar{x}}$ satisfying the following inequality:

$$\nabla f(\bar{x})^{\mathrm{T}}\bar{d} > \nabla f(\bar{x})^{\mathrm{T}}(\tilde{s} - \bar{x}). \tag{34}$$

By using Proposition 4, as $\tilde{s} \in C^{\bar{x}}$, there exists a subsequence $\{s^k\}_K$ converging to $\tilde{s}$, with $s^k \in C^{x^k}$. For k sufficiently large we have from inequality (34)

$$\nabla f(x^k)^{\mathrm{T}} d^k > \nabla f(x^k)^{\mathrm{T}}(s^k - x^k),$$

but this contradicts (33).

Now we prove that $\bar{x}$ is a stationary point. Indeed, $\bar{x}$ is a solution of

$$\begin{aligned} &\min \nabla f(\bar{x})^{\mathrm{T}} x = \min \sum_{j:\bar{x}_j \neq 0} \left(\nabla g_j(\bar{x}) + h'_j(\bar{x}_j)\right) x_j + \sum_{j \notin I^{\bar{x}}:\bar{x}_j = 0} \left(\nabla g_j(\bar{x}) + h'_j(0)\right) x_j \\ &x \in \Omega \\ &x_i = 0, \quad i \in I^{\bar{x}}. \end{aligned} \tag{35}$$

As g is continuously differentiable and C is compact, there exists a value $L < \infty$ such that

$$\|\nabla g(\bar{x})\|_\infty \leq L \tag{36}$$

and by (i) of Proposition 1 it follows that there exists a value M such that, if $\nabla g_j(\bar{x}) + h'_j(0) \geq M$, with $j \in I^{\bar{x}}$, then $\bar{x}$ is a solution of

$$\min \sum_{j:\bar{x}_j \neq 0} \left(\nabla g_j(\bar{x}) + h'_j(\bar{x}_j)\right) x_j + \sum_{j \notin I^{\bar{x}}:\bar{x}_j = 0} \left(\nabla g_j(\bar{x}) + h'_j(0)\right) x_j + \sum_{j \in I^{\bar{x}}:\bar{x}_j = 0} \left(\nabla g_j(\bar{x}) + h'_j(0)\right) x_j \tag{37}$$

$x \in \Omega$.

Therefore, if $h'_j(0) \geq S = M + L$, we have

$$\nabla f(\bar{x})^{\mathrm{T}}\bar{x} \leq \nabla f(\bar{x})^{\mathrm{T}} x \quad \forall x \in \Omega,$$

and this proves that $\bar{x}$ is a stationary point of problem (23). ■

Concerning the separable concave functions used in problems (8)–(11), we have for $j = 1, \ldots, n$

- $h_j(y_j; \alpha) = 1 - e^{-\alpha y_j}$ and $h_j^{'}(0) = \alpha$;
- $h_j(y_j; \epsilon) = \ln(y_j + \epsilon)$ and $h_j^{'}(0) = 1/\epsilon$;
- $h_j(y_j; \epsilon, p) = (y_j + \epsilon)^p$ and $h_j^{'}(0) = p(\epsilon)^{p-1}$ with $0 < p < 1$;
- $h_j(y_j; \epsilon, p) = -(y_j + \epsilon)^{-p}$ and $h_j^{'}(0) = p(\epsilon)^{-p-1}$ with $1 \leq p$.

Therefore, the assumption of Proposition 5 holds for suitable values of the parameters of the above concave functions, so that Algorithm FW-RD can be applied.

5. The FW-RD algorithm with unitary stepsize

When the function f of problem (23) is concave, we can use a constant stepsize $\alpha = 1$ and still be sure the algorithm converges to a stationary point. The following proposition shows convergence of the FW algorithm with stepsize $\alpha^k = s$ and $s \in (0, 1]$ when a concave function is minimized over a compact convex set:

PROPOSITION 6 *Let f be a continuously differentiable, concave function. Let $\{x^k\}$ be a sequence generated by the FW algorithm*

$$x^{k+1} = x^k + \alpha^k d^k,$$

where a constant stepsize is chosen

$$\alpha^k = s, \quad k = 0, 1, \ldots$$

with $s \in (0, 1]$. Then we have

$$\nabla f(x^k)^{\mathrm{T}} d^k \longrightarrow 0,$$

and every limit point $\bar{x}$ of $\{x^k\}$ is a stationary point.

Proof We have from concavity of f:

$$f(x^{k+1}) \leq f(x^k) + \nabla f(x^k)^{\mathrm{T}}(x^{k+1} - x^k) < f(x^k).$$

Note that since $\{f(x^k)\}$ is monotonically decreasing, $\{f(x^k)\}$ either converges to a finite value or diverges to $-\infty$.

Let $\bar{x}$ be a limit point of $\{x^k\}$; since f is continuous $f(\bar{x})$ is a limit point of $\{f(x^k)\}$, and so it follows that the entire sequence converges to $f(\bar{x})$. Therefore, we obtain

$$f(x^k) - f(x^{k+1}) \longrightarrow 0.$$

From concavity of f:

$$f(x^k) - f(x^{k+1}) \geq -\alpha^k \nabla f(x^k)^{\mathrm{T}} d^k.$$

Since α^k is a constant stepsize, we have that

$$\nabla f(x^k)^{\mathrm{T}} d^k \longrightarrow 0.$$

By Proposition 2 it follows that every limit point $\bar{x}$ of $\{x^k\}$ is a stationary point. ■

Here is a version of the modified FW algorithm, with unitary stepsize, for concave functions.

5.1 *FW-RD algorithm with unitary stepsize* (*FW-RDUS*)

(1) Let $x^0 \in C$ be the starting point.
(2) For $k = 0, 1, \ldots$, let $I^{x^k} = \{i \in I : x_i^k = 0\}$ and $C^{x^k} = \{x \in \Omega : x_i = 0, \forall\ i \in I^{x^k}\}$ obtain solution $\overline{x}^k$ by solving the following problem:

$$\overline{x}^k = \arg\min_{x \in C^{x^k}} \nabla f(x^k)^{\mathrm{T}}(x - x^k). \tag{38}$$

(3) If $\nabla f(x^k)^{\mathrm{T}}(\overline{x}^k - x^k) = 0$ then STOP.
(4) Otherwise

$$x^{k+1} = \bar{x}^k.$$

The following result about the convergence of the FW-RDUS algorithm is an immediate consequence of Proposition 5.

COROLLARY 1 *Let $\{x^k\}$ be a sequence generated by the FW-RDUS Algorithm. Suppose there exists a value S such that $h_i'(0) \geq S\ \forall\ x_i = 0$ with $i \in I$, then every limit point $\bar{x}$ of $\{x^k\}$ is a stationary point.*

The assumption of Corollary 1 holds for suitable values of the parameters of the concave functions presented in Section 2, so that algorithm FW-RDUS can be applied when using those functions. The results obtained on computational experiments will be presented in the next section.

6. Computational experiments

In our computational experiments, we have considered the problem of finding a sparse representation of a signal (see (4), Section 1). We remark that the aim of the experiments has been that of evaluating the effectiveness of the various formulations in finding sparse vectors (possibly the sparsest vectors) belonging to a convex set.

For each class of problems, we performed experiments using:

- formulation (8), denoted by *exp*, with $\alpha = 5$;
- formulation (9), denoted by *log*, with $\epsilon = 10^{-5}$;
- formulation (10), denoted by *Formulation I*, with $\epsilon = 10^{-7}$ and $p = 0.1$;
- formulation (11), denoted by *Formulation II*, with $\epsilon = 10^{-5}$ and $p = 1$.

6.1 *Implementation details*

Algorithms FW and FW-RDUS were implemented in C using CPLEX (10.0) as solver of the quadratic programming problems. The experiments were carried out on an Intel Pentium 4, 3.2 GHz, 1.0 GB RAM.

6.2 *Random test problems*

For several values of n and m, we randomly generated the matrix A, the vector b, and a value of the tolerance δ_1. Then we obtained two more values of the tolerance as follows: $\delta_2 = 2\delta_1$;

Table 1. Comparison on random test problems (average zero-norm value/best zero-norm value/percentage of best values attained).

Problem	n	m	δ	ℓ_1	exp	log	Form. I	Form. II
1	100	20	0.93	8	7.6/4/12	4.4/4/58	4.4/4/58	5.2/4/30
2	100	20	1.86	5	5.0/2/25	2.1/2/92	2.0/2/97	3.0/2/52
3	100	20	3.71	3	3.0/1/43	1.1/1/99	1.0/1/100	1.6/1/80
4	200	40	3.00	19	14.6/6/18	6.7/6/39	6.7/6/42	8.1/6/18
5	200	40	6.00	10	10.0/3/10	3.8/3/18	3.3/3/17	4.7/3/18
6	200	40	12.01	4	6.2/2/26	2.0/2/100	2.0/2/100	3.0/2/72
7	400	80	13.24	36	29.5/16/1	13.7/12/4	13.6/12/5	17.0/13/10
8	400	80	26.49	30	20.7/7/1	6.1/6/94	6.1/6/95	8.9/6/34
9	400	80	52.99	5	11.7/4/10	4.0/4/100	4.0/4/100	5.5/4/52
10	800	160	58.77	80	57.1/43/1	26.6/23/2	26.0/23/3	38.3/24/1
11	800	160	117.54	42	39.7/22/1	11.9/11/21	11.8/11/24	20.8/11/1
12	800	160	235.08	16	21.9/7/1	7.0/7/100	7.0/7/100	10.0/7/46
13	1600	320	263.96	147	109.5/75/1	48.4/45/5	48.1/44/1	92.0/48/1
14	1600	320	527.92	82	73.1/29/1	20.2/20/77	20.2/20/81	56.3/21/1
15	1600	320	1055.80	22	37.7/14/2	12.2/12/75	12.6/12/37	19.7/12/15

$\delta_3 = 4\delta_1$. The results obtained on these problems are shown in Table 1, where we report:

(1) The number n of variables, the number m of constraints.
(2) For the ℓ_1 norm formulation,

$$\min_{x \in R^n} \|x\|_1$$
$$\|Ax - b\|^2 \leq \delta,$$

denoted by ℓ_1, the zero-norm of the optimal solution attained.
(3) For each nonlinear concave formulation:
   (i) the average of the zero-norm value of the stationary points determined;
   (ii) the best zero-norm value of those stationary points;
   (iii) percentage of runs where the best zero-norm value was attained.

We used 100 random starting points for all the problems. From Table 1, we can see that *Formulation I* gives the best results among all the formulations. We further note that the results obtained by means of the concave formulations are clearly better than those corresponding to the ℓ_1 formulation.

Summarizing, the computational experiments confirm the effectiveness of the concave-based approach for finding sparse solutions to problems with convex constraints, and show that the concave formulations here proposed represent good alternatives to the ℓ_1 formulation. We remark that a wider availability of efficient formulations is important as it can make easier the search of sparse solutions for different classes of problems.

Finally, in order to assess the differences in terms of computational time between the standard FW algorithm and a new version of the algorithm presented in the preceding section and denoted by Algorithm FW-RDUS, we report in Table 2 the results obtained by the two algorithms using *log formulation*. As we might expect, the differences are noticeable and show the usefulness of Algorithm FW-RDUS. Further experiments not here reported and performed using the other concave formulations point out the same differences between the two algorithms in terms of computational time. In all the tests, we detected no difference between the two algorithms in terms of computed solution.

Finding a global optimal solution to a concave function over a compact convex set is an NP-Hard problem in general. Then, we decided to try some random examples where the matrix A

Table 2. Comparison using log formulation between the two versions of the FW algorithm in terms of CPU-time (s).

Problem	FW	FW-RDUS
1	0.453	0.094
2	0.141	0.047
3	0.140	0.032
4	1.000	0.188
5	0.890	0.141
6	0.625	0.109
7	8.219	1.015
8	7.515	1.563
9	6.579	1.359
10	73.015	6.391
11	81.656	7.437
12	76.391	4.141
13	767.657	93.094
14	866.719	51.89
15	812.609	46.32

Table 3. Comparison on random global optimization problems (average zero-norm value/percentage of best values attained).

Problem	n	m	$\| x^\star \|_0$	exp	log	Form. I	Form. II
G1	100	20	2	81.6/18	6.7/72	6.6/74	11.3/46
G2	200	40	4	101.2/49	4.0/100	4.0/100	8.5/87
G3	400	80	8	128.1/69	8.0/100	8.0/100	13.3/92
G4	800	160	16	147.47/79	16.0/100	16.0/100	24.0/94
G5	1600	320	32	415.72/69	32.0/100	32.0/100	47.8/94

includes many linear dependent columns, and check if the various formulations can filter out those columns. For several values of n and m, we randomly generated a matrix A with many linear dependent columns and the vector b as a linear combination of the linear independent columns. The results obtained on these problems are shown in Table 3, where we report:

(1) The number n of variables, the number m of constraints.
(2) The number of non-zero components of the sparsest solution $x^\star$.
(3) For each nonlinear concave formulation:
    (i) the average of the zero-norm value of the stationary points determined;
    (ii) percentage of runs where the sparsest solution was attained.

We set the tolerance $\delta = 10^{-6}$ and used 100 random starting points for all the problems. From Table 3 we can see that *Formulation I* gives the best results among all the formulations. Anyway, all formulations show a good ability in finding a global solution. So, the use of these formulations, combined with a simple global approach (e.g. random multistart), seems to ensure that a good sparse solution can be easily found.

6.3 *Comparison with other existing methods*

In order to better assess the performance of our algorithm, we have compared it with two well-known and deeply used methods for sparse reconstruction of signals, namely l_1–l_s [23] and l_1-magic [7]. l_1–l_s is an implementation of the interior-point method for ℓ_1-regularized least

squares. l_1–l_s solves an optimization problem of the form

$$\min_{x \in R^n} \|Ax - b\|^2 + \lambda \|x\|_1,$$

with $A \in R^{m \times n}$, $b \in R^m$, $x \in R^n$ and $\lambda \in R^+$ a suitably chosen parameter.

l_1-magic implements an interior-point method for finding a sparse reconstruction of a given signal. The problem solved by l_1-magic has the following form:

$$\min_{x \in R^n} \|x\|_1$$
$$\|Ax - b\|^2 \leq \epsilon,$$

where $\epsilon \in R^+$ is a user-specified parameter.

We considered two different classes of problems:

(1) Signal recovery problems similar to those described in [7]: we have a dictionary $A \in R^{m \times n}$ of elementary signals and a real noisy signal b (noise level σ). The goal is finding a sparse representation x of signal b in terms of the dictionary A.
(2) Pathological problems similar to those described in [29]: these small problems ($m = 128$ and $n = 512$) are pathological because the magnitudes of the non-zero entries of the exact solutions x lie in a large range (i.e. the largest magnitudes are significantly larger than the smallest magnitudes).

The parameters used for the algorithms are:

- l_1–l_s: $\lambda = 0.001\|A^{\mathrm{T}}b\|_\infty$;
- l_1-magic: $\epsilon = \sigma^2(n + 2\sqrt{2})$;
- FW-RDUS: $\delta = \epsilon$.

In Tables 4–8, we report the results for five different problems (dimensions of the problems in Table 9) belonging to the first class. In Figures 1–2, we, respectively, show the original solution for Problem S5 and the solution obtained by the FW-RDUS using Formulation I. In Tables 10–13, we report the results for four different problems belonging to the second class.

Table 4. Problem S1: comparison with other existing methods.

Algorithm	Error	Time
FW-RDUS-exp	9.60602E−003	7.800E−002
FW-RDUS-log	1.00381E−002	6.200E−002
FW-RDUS-Form. I	9.96317E−003	7.800E−002
FW-RDUS-Form. II	1.04946E−002	7.800E−002
l_1–l_s	1.48100E−002	1.234
l_1-magic	1.11700E−002	0.266

Table 5. Problem S2: comparison with other existing methods.

Algorithm	Error	Time
FW-RDUS-exp	4.17507E−003	0.360
FW-RDUS-log	3.63983E−003	0.281
FW-RDUS-Form. I	3.64687E−003	0.219
FW-RDUS-Form. II	3.76712E−003	0.219
l_1–l_s	8.13100E−003	2.141
l_1-magic	1.44500E−002	1.109

Table 6. Problem S3: comparison with other existing methods.

Algorithm	Error	Time
FW-RDUS-exp	7.98488E−003	2.843
FW-RDUS-log	6.27492E−003	3.046
FW-RDUS-Form. I	6.40922E−003	2.968
FW-RDUS-Form. II	6.15745E−003	3.109
l_1–l_s	1.54800E−003	6.438
l_1-magic	1.54800E−002	6.031

Table 7. Problem S4: comparison with other existing methods.

Algorithm	Error	Time
FW-RDUS-exp	4.30954E−003	17.625
FW-RDUS-log	4.12071E−003	17.578
FW-RDUS-Form. I	4.11681E−003	17.641
FW-RDUS-Form. II	4.16755E−003	18.546
l_1–l_s	1.09100E−002	29.030
l_1-magic	2.31800E−002	43.157

Table 8. Problem S5: comparison with other existing methods.

Algorithm	Error	Time
FW-RDUS-exp	9.08758E−004	89.172
FW-RDUS-log	1.07722E−003	89.515
FW-RDUS-Form. I	1.05751E−003	91.140
FW-RDUS-Form. II	1.35252E−003	88.485
l_1–l_s	5.98705E−003	136.400
l_1-magic	1.39400E−002	196.760

Table 9. Dimensions of the signal recovery problems used in the experiments.

Problem	n	m
S1	100	20
S2	200	40
S3	400	80
S4	800	160
S5	1600	320

For each algorithm, we report:

- the relative error between the recovered solution x and the sparsest solution $x^\star$, namely $\|x - x^\star\|/\|x^\star\|$;
- the CPU time in seconds.

From the results of Tables 4–8, 10–13, we get that our algorithm outperforms l_1–l_s and l_1-magic in terms of CPU time. Indeed, the accuracy, in terms of the relative error, of our algorithm is better than that of the other two algorithms in the vast majority of the problems. This confirms the robustness and efficiency of the algorithm proposed in this work.

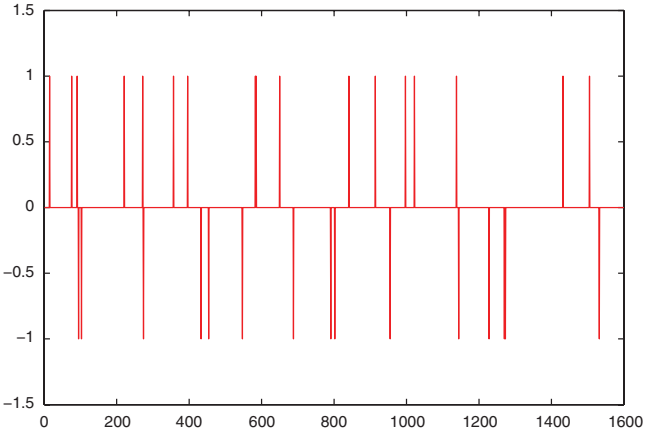

Figure 1. Problem S5: original sparse solution.

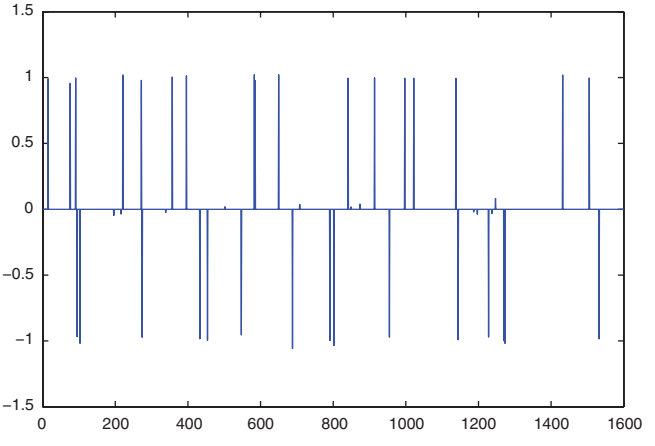

Figure 2. Problem S5: approximated solution (FW-RDUS-Form. I).

Table 10. Problem CT1: comparison with other existing methods.

Algorithm	Error	Time
FW-RDUS-exp	0.52245	48.984
FW-RDUS-log	0.52210	20.656
FW-RDUS-Form. I	0.51896	16.172
FW-RDUS-Form. II	0.56945	10.140
l_1–l_s	0.51970	36.880
l_1-magic	0.52060	18.750

Table 11. Problem CT2: comparison with other existing methods.

Algorithm	Error	Time
FW-RDUS-exp	3.66650E−005	16.485
FW-RDUS-log	4.11663E−006	25.750
FW-RDUS-Form. I	3.72668E−005	26.156
FW-RDUS-Form. II	1.78544E−003	19.219
l_1–l_s	4.52100E−004	22.170
l_1-magic	8.42300E−004	55.406

Table 12. Problem CT3: comparison with other existing methods.

Algorithm	Error	Time
FW-RDUS-exp	5.48016E−003	47.125
FW-RDUS-log	1.17253E−003	21.500
FW-RDUS-Form. I	7.81051E−002	16.250
FW-RDUS-Form. II	5.40180E−003	18.828
l_1–l_s	3.96100E−002	29.720
l_1-magic	6.13900E−002	68.078

Table 13. Problem CT4: comparison with other existing methods.

Algorithm	Error	Time
FW-RDUS-exp	9.91784E−005	64.953
FW-RDUS-log	5.27335E−005	21.969
FW-RDUS-Form. I	6.74718E−005	25.734
FW-RDUS-Form. II	3.41117E−004	21.907
l_1–l_s	1.58600E−004	22.000
l_1-magic	2.33000E−004	51.531

6.4 *Comments to the algorithm*

The new algorithm we described has obviously some pros and cons. We first describe the pros:

- The algorithm solves a more general class of problems than that solved by the other FW-based sparse techniques (e.g. SLA algorithm).
- The concave approximations of the ℓ_0 norm guarantee better results in terms of sparsity compared with the ℓ_1 norm.
- The fact that we solve at each step of the algorithm a problem of reduced dimension helps in decreasing the CPU-time.

Now we report the cons:

- The algorithm basically effects a greedy search. It starts with the full set of features and progressively eliminates the 'less promising' ones. So, once a component is eliminated, it cannot be reconsidered again.
- As we already said, we need to minimize a concave function over a compact convex set. As this is an NP-Hard problem, there is no guarantee that the solution attained by our algorithm is a global optimum.

Anyway, our numerical experience showed that both the cons are not a big deal in practice.

7. Conclusions and future work

In this work, we have considered the problem of finding a sparse solution to a problem with convex constraints, which arises in different important fields, such as signal processing and data analysis. We have proposed a concave optimization-based approach for dealing with this issue. Furthermore, we described a new efficient version of the FW algorithm and we proved its convergence to a stationary point.

The computational experiments evidenced that the concave formulations can be valid alternatives to the ℓ_1 formulation, as – in most cases – they get sparser solutions. The results we report

also show a considerable speed-up when using the variable fixing variant of the FW method in place of the traditional one. This speed-up might be extremely beneficial when multiple runs of the algorithm are performed, e.g. in a Multistart method.

Future work will be devoted to the development of global optimization algorithms for finding sparse solutions to problems having convex constraints and to the definition of suitable techniques for SLDA and SPCA.

Acknowledgements

The author is grateful to the anonymous reviewers for their useful comments and suggestions.

References

[1] D.P. Bertsekas, *Nonlinear Programming*, 2nd ed., Athena Scientific, Belmont, MA, 1999.
[2] P.S. Bradley and O.L. Mangasarian, *Feature selection via concave minimization and support vector machines*, Proceedings of the 15th International Conference on Machine Learning (ICML '98), J. Shavlik, ed., Morgan Kaufmann, San Francisco, California, 1998, pp. 82–90.
[3] A.M. Bruckstein, D.L. Donoho, and M. Elad, *From sparse solutions of systems of equations to sparse modeling of signals and images*, SIAM Rev. 51(1) (2009), pp. 34–81.
[4] J. Cadima and I.T. Jolliffe, *Loading and correlations in the interpretation of principal components*, J. Appl. Stat. 22 (1995), pp. 203–214.
[5] E. Candès and J. Romberg, *Quantitative robust uncertainty principles and optimally sparse decompositions*, Found. Comput. Math. 6(2) (2006), pp. 227–254.
[6] E. Candès, J. Romberg, and T. Tao, *Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information*, IEEE Trans. Inform. Theory 52(2) (2006), pp. 489–509.
[7] E. Candès, J. Romberg, and T. Tao, *Stable signal recovery from incomplete and inaccurate measurements*, Comm. Pure Appl. Math. 59(8) (2006), pp. 1207–1223.
[8] S.S. Chen, D.L. Donoho, and M.A. Saunders, *Atomic decomposition basis pursuit*, SIAM Rev. 43 (2001), pp. 129–159.
[9] G.B. Dantzig and M.N. Thapa, *Linear Programming 1: Introduction*, Springer Series in Operations Research and Financial Engineering, Springer Verlag, New York, 1997.
[10] G.B. Dantzig and M.N. Thapa, *Linear Programming 2: Theory and Extensions*, Springer Series in Operations Research and Financial Engineering, Springer Verlag, New York, 2003.
[11] A. D'Aspremont, L. El Ghaoui, M.I. Jordan, and G.R.G. Lanckriet, *A direct formulation for sparse PCA using semidefinite programming*, SIAM Rev. 49(3) (2007), pp. 434–448.
[12] V.F. Demjanov and A.M. Rubinov, *Approximate Methods in Optimization Problems*, American Elsevier, NY, 1970.
[13] D. Donoho, *Compressed sensing*, IEEE Trans. Inform. Theory, 52(4) (2006), pp. 1289–1306.
[14] D.L. Donoho and I.M. Johnstone, *Ideal denoising in an orthonormal basis chosen from a library of bases*, C. R. Acad. Sci. Paris Ser. I Math. 319 (1994), pp. 1317–1322.
[15] D.L. Donoho and I.M. Johnstone, *Ideal spatial adaptation by wavelet shrinkage*, Biometrika 81 (1994), pp. 425–455.
[16] D.L. Donoho and I.M. Johnstone, *Minimax estimation via wavelet shrinkage*, Ann. Statist. 26 (1998), pp. 879–921.
[17] D.L. Donoho, I.M. Johnstone, G. Kerkyacharian, and D. Picard, *Wavelet shrinkage— asymptopia*, J. R. Stat. Soc. Ser. B 57 (1995), pp. 301–337.
[18] M.A. Figueiredo and R.D. Nowak, *An EM algorithm for wavelet-based image restoration*, IEEE Trans. Image Process. 12(8) (2003), pp. 906–916.
[19] M.A. Figueiredo and R.D. Nowak, *Bound optimization approach to wavelet-based image deconvolution*, Conference on Image Processing – ICIP 2005, Genoa, Italy, Vol. 2, 2005, pp. 782–785.
[20] M.A. Figueiredo, J.M. Bioucas-Dias, and R.D. Nowak, *Majorization-minimization algorithms for wavelet-based image restoration*, IEEE Trans. Image Process. 16(12) (2007), pp. 2980–2991.
[21] M. Frank and P. Wolfe, *An algorithm for quadratic programming*, Naval Res. Logist. Q. 3 (1956), pp. 95–110.
[22] G.M. Fung, O.L. Mangasarian, and A.J. Smola, *Minimal kernel classifiers*, J. Mach. Learn. Res. 3 (2002), pp. 303–321.
[23] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, *An interior-point method for large-scale l1-regularized least squares*, IEEE J. Sel. Top. Signal Process. 1(4) (2007), pp. 606–617.
[24] E.S. Levitin and B.T. Poljak, *Constrained minimization methods*, Z. Vycisl. Mat. i Mat. Fiz. 6 (1965), pp. 787–823.
[25] O.L. Mangasarian, *Machine learning via polyhedral concave minimization*, in *Applied Mathematics and Parallel Computing – Festschrift for Klaus Ritter*, H. Fischer, B. Riedmueller, and S. Schaeffler, eds., Physica-Verlag, Germany, 1996, pp. 175–188.
[26] B. Moghaddam, Y. Weiss, and S. Avidan, *Generalized spectral bounds for sparse LDA*, in Proceedings of the 23rd International Conference on Machine Learning, Pittsburgh, PA, 2006.

[27] F. Rinaldi, F. Schoen, and M. Sciandrone, *Concave programming for minimizing the zero-norm over polyhedral sets*, Comput. Opt. Appl. 46(3) (2010), pp. 467–486.
[28] R. Tibshirani, *Regression shrinkage and selection via the Lasso*, J. R. Stat. Soc. Ser. B 58(1) (1996), pp. 267–288.
[29] Z. Wen, W. Yin, D. Goldfarb, and Y. Zhang, *A fast algorithm for sparse reconstruction based on shrinkage, subspace optimization and continuation*, Submitted to SIAM Journal on Scientific Computing, Rice University CAAM Technical Report TR09-01, 2009.
[30] J. Weston, A. Elisseef, and B. Schölkopf, *Use of the zero-norm with linear models and kernel model*, J. Mach. Learn. Res. 3 (2003), pp. 1439–1461.
[31] H. Zou, T. Hastie, and R. Tibshirani, *Sparse principal component analysis*, J. Comput. Graph. Stat. 15 (2006), pp. 256–286.

Appendix: Optimality conditions for constrained problems based on Lagrange multipliers

We now recall some well-known optimality conditions for constrained problems based on Lagrange multipliers, namely Karush–Kuhn–Tuker conditions (see [1] for further details).

We consider the problem:

$$\begin{aligned} &\min f(x) \\ &\varphi(x) \leq 0 \\ &\psi(x) = 0, \end{aligned} \tag{A1}$$

where $f : R^n \to R$, $\varphi : R^n \to R^m$, and $\psi : R^n \to R^p$ are continuously differentiable functions.

DEFINITION A1 *A feasible vector x is said to be regular if the equality constraints gradients $\nabla\psi_i(x)$, $i = 1, \ldots, p$, and the active inequality constraint gradients $\nabla\varphi_i(x)$, $i \in A(x) = \{i : \varphi_i(x) = 0\}$, are linearly independent.*

We now state necessary conditions for optimality.

PROPOSITION A1 *Let $x^\star$ be a local minimum of the problem* (A1). *Assume that $x^\star$ is regular. Then there exists Lagrange multipliers $\lambda^\star \in R^m$ and $\mu^\star \in R^p$ satisfying the following conditions:*

$$\begin{aligned} &\nabla f(x^\star) + \nabla\varphi(x^\star)\lambda^\star - \nabla\psi(x^\star)\mu^\star = 0 \\ &\lambda^{\star\mathrm{T}}\varphi(x^\star) = 0 \\ &\lambda^\star \geq 0. \end{aligned} \tag{A2}$$

There are a number of conditions (so-called constraint qualifications) that guarantee the existence of Lagrange multipliers. The following proposition is due to Slater.

PROPOSITION A2 *Let $x^\star$ be a local minimum of the problem* (A1) *Assume that ψ_i are affine functions, that φ_j are convex functions and that there exists a feasible vector $\bar{x}$ satisfying the following condition:*

$$\varphi_j(\bar{x}) < 0 \quad \forall\, j \in A(x^\star). \tag{A3}$$

Then $x^\star$ satisfies the necessary conditions of Proposition A1.

Under some suitable convexity assumptions, we can state sufficient conditions for optimality.

PROPOSITION A3 *Let f and φ_i $i = 1, \ldots, m$ be convex continuously differentiable functions, and let equality constraints $\psi_i(x)$ $i = 1, \ldots, p$ be affine functions. If there exists Lagrange multipliers $\lambda^\star \in R^m$ and $\mu^\star \in R^p$ satisfying the following conditions:*

$$\begin{aligned} \nabla f(x^\star) + \nabla\varphi(x^\star)\lambda^\star - \nabla\psi(x^\star)\mu^\star &= 0 \\ \varphi(x^\star) \leq 0,\ \psi(x^\star) &= 0 \\ {\lambda^\star}^{\mathrm{T}}\varphi(x^\star) &= 0 \\ \lambda^\star \geq 0, & \end{aligned} \tag{A4}$$

then $x^\star$ is a global minimum of the problem (A1).