

## Research Article

# Gradient-Type Methods: A Unified Perspective in Computer Science and Numerical Analysis

**Stefano Fanelli**

*Dipartimento di Matematica, Università degli Studi di Roma Tor Vergata, Viale della Ricerca Scientifica, 00133 Roma, Italy*

Correspondence should be addressed to Stefano Fanelli, fanelli@mat.uniroma2.it

Received 4 April 2011; Accepted 11 May 2011

Academic Editors: G. Kyriacou and M. Sun

Copyright © 2011 Stefano Fanelli. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a general and comprehensive description of Optimization Methods, and Algorithms from a novel viewpoint. It is shown, in particular, that Direct Methods, Iterative Methods, and Computer Science Algorithms belong to a well-defined general class of both Finite and Infinite Procedures, characterized by suitable descent directions.

## 1. Introduction

The dichotomy between Computer Science and Numerical Analysis has been for many years the main obstacle to the development of *eclectic computational tools*. With the latter term the author indicates the capability of implementing algorithms properly adaptable to particular environmental requirements and, therefore, optimized for this aim.

Since the formulation of a problem requires the preliminary definition of the variables, and the functions involved in the model, the antithesis between finite and continuous applied mathematics is even stronger from a computational point of view.

In Computer Science, problems are typically defined on discrete sets (graphs, integer variables and so forth) and are characterized by procedures formalized in a *finite* number of steps.

Direct Methods, which are classical tools of Numerical Analysis, can be considered, in fact, *algorithms* according to the standard Computer Science definitions. However, the presence of ill-conditioned matrices can seriously affect the practical implementation of Direct Methods. On the other hand, Iterative Methods are based in the majority of cases on the convergence of a sequence approximating the optimal solution of a problem defined in a continuous range. Proper stopping rules on the truncation error reduce the latter

computational scheme to a finite process, but, unfortunately, in many cases the theoretical result is affected by a variety of numerical instability problems, thereby preventing a precise forecast of the *true* number of iterations, requested to achieve the desired approximation.

Furthermore, Linear Programming, Convex Quadratic Programming, and the unconstrained minimization of a symmetric positive definite bilinear form are continuous problems that can be exactly solved with a finite number of steps. This proves that the distinction between algorithms and infinite iterative procedures is not always characterized by the discrete or the continuous range of the variables involved in the problem.

Most of Numerical Analysis methods are based upon the application of the *Fixed Point theorem*, assuring the convergence of the iterative scheme by means of a contraction of the distance between successive terms of the sequence approximating the optimal solution.

Gradient methods are usually considered in the literature as particular procedures in the frame of optimization techniques, for classical unconstrained or constrained problems.

The main aim of the present paper is to show that *Gradient or Gradient-type* methods represent the fundamental computational tool to solve a wide set of continuous optimization problems, since they are based on a unitary principle, referred to both to *finite* and to *infinite* procedures.

Moreover, some classical discrete optimization algorithms can be also viewed in the framework of Gradient-type methods.

Hence, the *gradient approach* allows to deal with problems involving variables defined both in a continuous range and in a discrete one, by utilizing finite or infinite procedures in a quite general perspective.

It is essential to underline that ABS methods [1], which represent a remarkable class of algorithms for solving linear and nonlinear equations, are founded on a quite different approach. Roughly speaking, ABS-methods construct, in fact, a set of spanning matrices in  $R^n$ , by performing an adaptive optimization, associated to the dimension of the subspace and *parameter dependent*. In many ABS-methods the choice of the set of optimal parameters is, in fact, crucial in order to identify by a unified approach the structural features of the optimization algorithms. *Parameter dependence is not present in Gradient-type methods*.

It is important to emphasize that the typical finiteness of Computer Science algorithms is characterized by classes of Gradient-type methods converging to an isolated point of a suitable sequence, generated by the procedure.

Furthermore, the most recent algorithms for Local Optimization can be precisely described by Gradient-type methods in a general framework. As a matter of fact, Interior Points techniques [2, 3], Barrier Algorithms [4] represent a wide set of Gradient-type methods for NonLinear Programming.

Moreover, a fundamental role in this new approach is played by the properties of suitable *Structured matrices*, associated to the optimization procedures. Advanced Linear Algebra Techniques are, in fact, essential to construct low-complexity algorithms.

We point out, in particular, the techniques based on *Fast Transforms* and the corresponding approximations by algebras of matrices simultaneously diagonalized [5–10].

The utilization of Advanced Linear Algebra Techniques in NonLinear Programming opens a new research field, leading in many cases to a significant improvement both of the efficiency and in the practical application of Gradient-type methods for problems of operational interest [11, 12].

In Deterministic Global Optimization structured matrices allow remarkable results in the frame of the *Tunneling techniques*, by using the classical  *$\alpha BB$*  approach [9].

The novel results on *Tensor computation* [13] are a promising area of research to improve the efficiency of global optimization algorithms for large-scale problems and particularly for the effective construction of more general sets of *Repeller matrices in the Tunneling phases* [14, 15]. This approach can have important consequences also in Nonlinear Integer Optimization (see the pioneer work in [16]), taking into account the more recent results concerning the discretization of the problem by the *continuation methods* (see, e.g., [17]).

Therefore, this survey has also the aim of finding in-depth general relationships between Local Optimization techniques and Deterministic Global Optimization algorithms in the frame of Advanced Linear Algebra Techniques.

## 2. The Gradient and the Gradient-Type Approach

Let

$$\min_{\mathbf{x} \in A} f(\mathbf{x}), \quad A \subseteq R^n, \quad (2.1)$$

be an unconstrained problem to be solved.

By assuming  $f(\mathbf{x}) \in C^1(A)$ , the simplest heuristic procedure to deal with (2.1) is to determine the stationary points of  $f(\mathbf{x}) \in A$ , that is,  $\mathbf{x}^* \in A : \nabla f(\mathbf{x}^*) = 0$  by the recursive computational scheme:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \lambda_k \nabla f(\mathbf{x}^{(k)}), \quad k = 0, 1, \dots \quad (2.2)$$

with

- $\mathbf{x}^{(0)} \in A$  being the initial point of the procedure,
- $-\nabla f(\mathbf{x}^{(k)})$  the direction of maximum local decreasing,
- $\lambda_k$  the one-dimensional step-size,  $\lambda_k > 0$ ,

for all  $k$ ,  $\lambda_k$  is computed such that

$$\begin{aligned} \mathbf{x}^{(k+1)} &\in A, \\ f(\mathbf{x}^{(k+1)}) &\leq f(\mathbf{x}^{(k)}), \end{aligned} \quad (2.3)$$

and the sequence  $\{\mathbf{x}^{(k+1)}\}$  satisfies the condition

$$\lim_{k \rightarrow \infty} \nabla f(\mathbf{x}^{(k)}) = \mathbf{0}. \quad (2.4)$$

The iterative method (2.2) is a particular case of the following general Gradient-type method:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \lambda_k \mathbf{s}^{(k)}, \quad k = 0, 1, \dots, \quad (2.5)$$

(a) Given  $\{\gamma_k\}, \{\sigma_k\}, k = 0, 1 \dots$  s.t.  
 $\inf \{\gamma_k\} > 0, \inf \{\sigma_k\} > 0$   
 Let  $\mathbf{x}^{(0)} \in R^n$  be a starting point  
 For a given vector  $\mathbf{s}^{(k)} \in D(\gamma_k, \mathbf{x}^{(k)})$ , set  
 $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \lambda_k \mathbf{s}^{(k)}$ , with  $\lambda_k \in (0, \sigma_k \|\nabla f(\mathbf{x}^{(k)})\|)$ :  
 $f(\mathbf{x}^{(k+1)}) \approx \min_{\mu} \{f(\mathbf{x}^{(k)} - \mu \mathbf{s}^{(k)}), 0 < \mu \leq \sigma_k \|\nabla f(\mathbf{x}^{(k)})\|\}$

**Algorithm 1**

where

$\mathbf{s}^{(k)}$  is a *descent direction*, that is,  
 $\mathbf{s}^{(k)T} \nabla f(\mathbf{x}^{(k)}) > 0 \Leftrightarrow \cos(-\nabla f(\mathbf{x}^{(k)}), \widehat{\mathbf{s}^{(k)}}) \geq 0$ .

The following theorem generalizes a well-known result shown in [18].

**Theorem 2.1** (see [7]). *If  $\mathbf{s}^{(k)}$  is a descent direction in  $\mathbf{x}^{(k)}$  for a function  $f(\mathbf{x})$ , then*

$$\mathbf{s}^{(k)} = -A_k^{-1} \nabla f(\mathbf{x}^{(k)}) \quad (2.6)$$

with  $A_k$  being a symmetric positive definite (spd) matrix.

Moreover, the following property holds:

$$\cos(-\nabla f(\mathbf{x}^{(k)}), \widehat{\mathbf{s}^{(k)}}) \geq c > 0 \iff \text{cond}(A_k) \leq M, \quad \forall k. \quad (2.7)$$

*Remark 2.2.* Particular cases of descent directions can be obtained, by setting

$A_k = I$  (Steepest Descent method),  
 $A_k = \nabla^2 f(\mathbf{x}^{(k)})$  (Newton-Raphson method),  
 $A_k \approx \nabla^2 f(\mathbf{x}^{(k)})$  (general quasi-Newton and classical BFGS methods),  
 $A_k$  structured  $\approx \nabla^2 f(\mathbf{x}^{(k)})$  (low-complexity BFGS-type methods)

(see [5, 6, 18, 19]).

It is useful to underline that the general theory of admissible directions for unconstrained optimization [20] is also a special case of (2.5). By setting, in fact, for a given  $\gamma > 0$ :

$$D(\gamma, \mathbf{x}^{(k)}) = \left\{ \mathbf{s}^{(k)} \in R^n : \|\mathbf{s}^{(k)}\| = 1, \nabla f(\mathbf{x}^{(k)})^T \mathbf{s}^{(k)} \geq \gamma \|\nabla f(\mathbf{x}^{(k)})\| \right\}, \quad (2.8)$$

one can obtain other Gradient-type methods described by (2.5).

The iterative scheme described by Algorithm 1 contains several ingredients of a general Gradient-type method see [20].

The convergence of Algorithm 1 is guaranteed by the following result (see again [20]).

**Theorem 2.3.** Let  $f(\mathbf{x}) \in C^1(A)$ ,  $A$  open  $\subset R^n$ . Let  $K = \{\mathbf{x} \in R^n : f(\mathbf{x}) \leq f(\mathbf{x}^{(0)})\} \subset A \Rightarrow \mathbf{x}^{(0)} \in K$ . Then, for all  $\{\mathbf{x}^{(k)}\}$  evaluated by Algorithm 1:

- (i)  $\mathbf{x}^{(k)} \in K$ , for all  $k$ ;
- (ii) if  $\mathbf{x}^{(k+1)} \neq \mathbf{x}^{(k)}$ ,  $\{\mathbf{x}^{(k)}\}$  has at least an Extremal Point (EP)  $\mathbf{x}^*$ ;
- (iii) every EP  $\mathbf{x}^*$  of  $\{\mathbf{x}^{(k)}\}$  is a stationary point, that is,  $\nabla f(\mathbf{x}^*) = 0$ .

*Remark 2.4.* Notice that Theorem 2.3 can be also applied in the case of classical Computer Science algorithms. As a matter of fact, if condition (ii) is not verified, then, by definition,  $\exists \tilde{k} : \mathbf{x}^{(\tilde{k}+1)} = \mathbf{x}^{(\tilde{k})}$ , implying  $\nabla f(\mathbf{x}^{(\tilde{k})}) = 0$ , that is, the convergence to a stationary point in a finite number of steps  $\tilde{k}$ . Moreover, the convergence to an isolated point  $\hat{\mathbf{x}}$  of the sequence  $\{\mathbf{x}^{(k)}\}$  can be proven ab absurdo by showing that

$$\text{if } \hat{\mathbf{x}} \in \text{EP} \Rightarrow \forall k > k_0, \quad f(\mathbf{x}^{(k)}) - f(\mathbf{x}^{(k+1)}) > c_0, \quad c_0 > 0. \quad (2.9)$$

We will see in par. 3-4 that the convergence in a finite number of steps of a given iterative procedure can be verified in this way both for the unconstrained problems and for the constrained ones.

### 3. Local Unconstrained Optimization

Let  $C$  and  $\mathbf{b}$  be a spd matrix of order  $n$  and a  $n$ -dimensional vector, respectively.

It is well known that the problem

$$\begin{aligned} \min \quad & \frac{1}{2} \mathbf{x}^T C \mathbf{x} - \mathbf{b}^T \mathbf{x}, \\ & \mathbf{x} \in R^n \end{aligned} \quad (3.1)$$

can be exactly solved in at most  $n$  steps by the Conjugate Gradient (CG) method [21], which represents a direct method to solve (3.1). The quadratic form associated to a spd matrix  $C$  is, in fact, a convex function.

However, it can be also proved that the application of the procedure defined in (2.2), that is, the Steepest Descent method, always requires an *infinite* number of iterations, apart from the trivial case  $\mathbf{x}^{(0)} = \mathbf{x}^*$ . The latter result shows that the existence of a finite procedure to solve (3.1) *does not depend only by the role played by convexity* but it is also the consequence of a sort of *optimal matching between the problem and the corresponding algorithm*, which is in this case the (CG) method. On the other hand, the latter method can be also interpreted as an *iterative method* in the family of the following fixed point procedures:

$$\mathbf{x}^{(k+1)} = \frac{((rI - C)\mathbf{x}^{(k)} + \mathbf{b})}{r} \quad (3.2)$$

with  $r$  being a suitable scalar parameter. By setting, in fact,

$$H = I - \frac{C}{r},$$

$$D = \begin{pmatrix} \frac{1}{r} & 0 & \cdots & \cdots \\ 0 & \frac{1}{r} & 0 & \cdots \\ 0 & 0 & \frac{1}{r} & \cdots \\ 0 & \cdots & \cdots & \frac{1}{r} \end{pmatrix}, \quad (3.3)$$

one can obtain the classical iterative scheme

$$\mathbf{x}^{(k+1)} = H\mathbf{x}^{(k)} + D\mathbf{b}. \quad (3.4)$$

Since  $\|H\|_s = 1 - 1/\text{cond}(C)$ , (3.4) is convergent if the original matrix  $C$  is well conditioned. Moreover, if  $\hat{\mathbf{x}}$  is the optimal solution of (3.1), the truncation error of the method is:

$$\|\mathbf{x}^{(k)} - \hat{\mathbf{x}}\|_2 \leq \left(1 - \frac{1}{\text{cond}(C)}\right)^k \|\mathbf{x}^{(0)} - \hat{\mathbf{x}}\|_2. \quad (3.5)$$

In the case of (CG) method, one can prove the inequality

$$\|\mathbf{x}^{(k)} - \hat{\mathbf{x}}\|_2 \leq 2 \left( \frac{\sqrt{\text{cond}(C)} - 1}{\sqrt{\text{cond}(C)} + 1} \right)^k \|\mathbf{x}^{(0)} - \hat{\mathbf{x}}\|_2. \quad (3.6)$$

Equation (3.6) shows that, if the dimension  $n$  is huge and the matrix  $C$  is well conditioned, from a computational point of view it is more convenient to implement the (CG) method as a classical iterative procedure with a stopping rule based on the above inequality.

So, once again, the distinction between Numerical Analysis direct methods (or Computer Science algorithms) and infinite procedures cannot be considered as the fundamental classification rule in computational mathematics.

In the case of Steepest Descent method, the truncation error is

$$\|\mathbf{x}^{(k)} - \hat{\mathbf{x}}\|_2 \leq 2 \left( \frac{\text{cond}(C) - 1}{\text{cond}(C) + 1} \right)^k \|\mathbf{x}^{(0)} - \hat{\mathbf{x}}\|_2. \quad (3.7)$$

The difference between (3.6) and (3.7) clearly indicates the major efficiency of (CG) method.

In [22] the finite version of (CG)-method was extended to a family of nonquadratic functions, including the following important sets:

$$F(\mathbf{x}) = \frac{\mathbf{x}^T C \mathbf{x}}{(\mathbf{c}^T \mathbf{x})^2}, \quad \mathbf{x} \in X, \quad (3.8)$$

$$G(\mathbf{x}) = \mathbf{x}^T C \mathbf{x} * (\mathbf{c}^T \mathbf{x})^k, \quad k \text{ integer}, \mathbf{x} \in X, \quad (3.9)$$

where  $X = \{\mathbf{x} \in R^n : \mathbf{c}^T \mathbf{x} > 0\}$ .

According to the classical definition, the function  $F$  indicated in (3.8) is called *conic*. If  $k = -2$ , then  $G(\mathbf{x}) \equiv F(\mathbf{x})$ .

Hence,  $G$  represent a class of nonquadratic functions for which the optimal solution can be found with a finite number of steps if the matrix  $C$  is *spd*.

As a matter of fact, the following result holds:

**Theorem 3.1** (see [22] Theorem 3.1, Lemmas 3.2 and 5.1). *Let  $G(\mathbf{x})$  be defined as in (3.9). Then the minimum problem*

$$\begin{aligned} \min G(\mathbf{x}), \\ \mathbf{x} \in X, \end{aligned} \quad (3.10)$$

*can be solved in at most  $n$  steps.*

Let us now consider some generalizations of the convexity, which play an important role in global optimization see [7].

Let  $\mathbf{s}^{(k)}$  be a descent direction in  $\mathbf{x}^{(k)}$  for a function  $f(\mathbf{x})$ . The importance of the following definitions will be shown in the next results of this paragraph.

*Definition 3.2.* A function  $f(\mathbf{x}) \in C^1(R^n)$  is called *algorithmically convex* if for all  $\mathbf{x}^{(k)}, \mathbf{x}^{(k+1)}$  evaluated by an algorithm of type (2.5), one has

$$\left(\mathbf{s}^{(k+1)} - \mathbf{s}^{(k)}\right)^T \left(\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\right) \geq 0. \quad (3.11)$$

*Definition 3.3.* A function  $f(\mathbf{x}) \in C^1(R^n)$  is called *weakly convex* if for all  $\mathbf{x}^{(k)}, \mathbf{x}^{(k+1)}$  evaluated by an algorithm (2.5), the following inequality holds:

$$\frac{\|\nabla f(\mathbf{x}^{(k+1)}) - \nabla f(\mathbf{x}^{(k)})\|^2}{(\nabla f(\mathbf{x}^{(k+1)}) - \nabla f(\mathbf{x}^{(k)}))^T (\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)})} \leq M. \quad (3.12)$$

*Definition 3.4.* Let  $\mathbf{s}^{(k)} = -A_k^{-1} \nabla f(\mathbf{x}^{(k)})$ , for all  $k$  be descent directions of an algorithm of type (2.5) applied to problem (2.1). Then the method is called *secant* if the matrix  $A_k$  solves the *secant equation*:

$$A_k \left(\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\right) = \nabla f(\mathbf{x}^{(k)}) - \nabla f(\mathbf{x}^{(k-1)}). \quad (3.13)$$

Definition 3.2 is clearly a generalization of convexity. As a matter of fact, if  $\mathbf{s}^{(k)} = \nabla f(\mathbf{x}^{(k)})$  then  $f(\mathbf{x}) \in C^1(A)$ ,  $A \subseteq R^n$ , is convex if and only if (3.11) is verified for all  $\mathbf{x}^{(k)}$ ,  $\mathbf{x}^{(k+1)} \in A$  (see [23]).

Definition 3.3 is also a generalization of convexity. In [24], in fact, it is proved that if  $f(\mathbf{x}) \in C^1(A)$ ,  $A \subseteq R^n$ , is convex, then (3.12) is satisfied for all  $\mathbf{x}^{(k)}$ ,  $\mathbf{x}^{(k+1)} \in A$ . So (3.12) is a *necessary, but not sufficient*, condition for a function  $f$  to be convex.

Definition 3.4 is an  $n$ -dimensional generalization of the classical secant iterative formula to compute the zeroes of the derivative of a function  $f(x) \in C^1(R^1)$ , that is,

$$x_{k+1} = \frac{f'(x_k)x_{k-1} - f'(x_{k-1})x_k}{f'(x_k) - f'(x_{k-1})}. \quad (3.14)$$

Observe, in fact, that (3.14) can be rewritten as

$$\begin{aligned} x_{k+1} &= x_k - \frac{f'(x_k)}{a_k}, \\ a_k(x_k - x_{k-1}) &= f'(x_k) - f'(x_{k-1}). \end{aligned} \quad (3.15)$$

Hence, the expression of  $a_k$  is the 1-dimensional version of (3.13).

In [7] it is proved the following result.

**Theorem 3.5** (see also [18]). *Let  $\mathbf{s}^{(k)} = -A_k^{-1}\nabla f(\mathbf{x}^{(k)})$  be descent directions of a secant method, that is, satisfying (3.13), applied to problem (2.1). Moreover, let conditions (2.7) and (3.12) be verified. Then,  $\exists\{\nabla f(\mathbf{x}^{(k_i)})\}$ , such that*

$$\lim_{i \rightarrow \infty} \nabla f(\mathbf{x}^{(k_i)}) = \mathbf{0}. \quad (3.16)$$

*Remark 3.6.* Theorem 3.5 shows that a *global convergence* for a quasi-Newton secant method applied to problem (2.1) can be obtained if the function  $f(\mathbf{x})$  is weakly convex and the matrices  $A_k$  approximating  $\nabla^2 f(\mathbf{x}^{(k)})$  are well conditioned.

*Remark 3.7.* By utilizing Armijo-Goldstein-Wolfe's method [18] and setting  $\mathbf{s}^{(k)} = \nabla f(\mathbf{x}^{(k)})$ , the step  $\lambda_k$  in (2.5) is such that for all  $k$

$$\begin{aligned} (\nabla f(\mathbf{x}^{(k+1)}) - \nabla f(\mathbf{x}^{(k)}))^T (\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}) &> 0, \\ f(\mathbf{x}^{(k+1)}) &< f(\mathbf{x}^{(k)}). \end{aligned} \quad (3.17)$$

Hence, by Definition 3.2, in this case the function  $f(\mathbf{x})$  is also algorithmically convex. For general descent directions  $\mathbf{s}^{(k)}$ , evaluated by a quasi-Newton secant method, inequality (3.11) is not always satisfied.

#### 4. Local Constrained Optimization

Quadratic Programming (QP) is defined in the following way:

$$\begin{aligned} \min \mathbf{x}^T \mathbf{C} \mathbf{x} + \mathbf{c}^T \mathbf{x}, \\ \mathbf{A} \mathbf{x} = \mathbf{b}, \\ \mathbf{x} \geq \mathbf{0} \end{aligned} \tag{4.1}$$

with  $\mathbf{C}$  being a symmetric semidefinite positive (ssdp) matrix of order  $n$  and  $\mathbf{A}$  a matrix with  $m$  rows and  $n$  columns.

*Remark 4.1.* Let  $P = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{A} \mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}$ . The optimal solution of (4.1) can be located in any point of  $P$ . Hence, (4.1) is a *continuous problem which cannot be immediately reduced to a finite problem* as in the case  $F(\mathbf{x}) = \mathbf{c}^T \mathbf{x}$ , that is, Linear Programming (LP).

Let us consider, for instance, the following problems:

$$\begin{aligned} \min x_1^2 - 3x_2, \\ 2x_1 - x_2 \geq 4, \\ -x_1 - 2x_2 \geq -16, \\ -2x_1 + 4x_2 \geq -8, \\ -7x_1 + 8x_2 \geq -35, \\ x_1 \geq 0, \quad x_2 \geq 0, \end{aligned} \tag{4.2}$$

$$\begin{aligned} \min x_1^2 + 4x_2^2 - 4x_1 - 24x_2 + 40, \\ -7x_1 - 6x_2 + 42 \geq 0, \\ -5x_1 + 3x_2 + 10 \geq 0, \\ x_1 \geq 0, \quad x_2 \geq 0. \end{aligned} \tag{4.3}$$

The optimal solution of (4.2) is the point  $(3, 2)$  which is in the boundary of  $P$  but *is not a vertex*. On the other hand, problem (4.3) has the optimal solution in the inner point  $(2, 3)$ . However, QP can be solved in general in a *finite number of steps* by means of Frank-Wolfe's algorithm [25]. So, QP can be considered as a *finite continuous constrained optimization problem*.

The following question arises: does QP characterize *the boundary, separating finite continuous constrained optimization problems from infinite ones*? In other words, there exist more general nonlinear constrained optimization problems that can be solved in a finite number of iterations? Since in the unconstrained case we have shown in the previous paragraph that there exist nonquadratic problems that can be exactly solved in a finite number of iterations by utilizing the (CG)-method, the answer is expected to be positive.

Given a convex function  $f(\mathbf{x}) \in C^1(S)$ ,  $S$  convex  $\subseteq R^n$ , Convex Programming with Linear Constraints (CPLC) is defined as

$$\begin{aligned} \min f(\mathbf{x}), \\ A\mathbf{x} = \mathbf{b}, \\ \mathbf{x} \geq \mathbf{0}. \end{aligned} \tag{4.4}$$

Problem (4.4) can be solved by the Reduced Gradient (RG) algorithm or by the Gradient Projection (GP) method [23, 26, 27].

Assuming  $A$  with maximum rank and taking into account Remark 2.4, one can introduce the following.

*Definition 4.2.* Let  $\hat{f}(\mathbf{x}) \in C^1(S)$ ,  $S$  convex  $\subseteq R^n$ , be a convex function.

Let  $\hat{P} = \{\mathbf{x} \in R^n : \hat{A}\mathbf{x} = \hat{\mathbf{b}}, \mathbf{x} \geq \mathbf{0}\}$  be a nonempty polyhedron. The corresponding CPLC problem (4.4) is a finite continuous constrained optimization problem, if and only if there exists a convergent Gradient-type method (2.5) and a positive real number  $c_0$ , such that if (2.5) would require an infinite number of steps, then

$$\begin{aligned} \mathbf{x}^{(k)} \in \hat{P}, \quad \forall k, \\ \inf_k \left\{ \hat{f}(\mathbf{x}^{(k)}) - \hat{f}(\mathbf{x}^{(k+1)}) \right\} \geq c_0, \quad \forall k > k_0, \quad c_0 > 0. \end{aligned} \tag{4.5}$$

Equation (4.5) clearly implies that  $\exists k_* : \hat{f}(\mathbf{x}^{(k_*)}) = \min_{\mathbf{x} \in \hat{P}} \hat{f}(\mathbf{x})$ .

The importance of Definition 4.2 can be pointed out by the next result, showing the relationship between (4.4) and a particular linear optimization problem.

**Theorem 4.3.** Let  $\mathbf{x}^{(k)}$  be an admissible solution of (4.4). Let  $\mathbf{s}^{(k)}$  be a descent direction in  $\mathbf{x}^{(k)}$  for the function  $f(\mathbf{x})$ . Then  $\mathbf{s}^{(k)}$  is an admissible descent direction for (4.4) if  $A\mathbf{s}^{(k)} = \mathbf{0}$ .

Moreover, for any fixed  $\hat{\mathbf{x}}^{(k)}$  the optimal solution  $\mathbf{s}^*$  of the problem

$$\begin{aligned} \min \nabla f(\hat{\mathbf{x}}^{(k)})^T \mathbf{s}, \\ A\mathbf{s} = \mathbf{0}, \\ \|\mathbf{s}\| = 1 \end{aligned} \tag{4.6}$$

is given by

$$\mathbf{s}^* = \frac{(I - A^T(AA^T)^{-1}A)\nabla f(\hat{\mathbf{x}}^{(k)})}{\|(I - A^T(AA^T)^{-1}A)\nabla f(\hat{\mathbf{x}}^{(k)})\|}. \tag{4.7}$$

By setting  $\mathbf{c} = f(\widehat{\mathbf{x}}^{(k)})$  and  $\mathbf{x} = \mathbf{s}$  it was proven in [28] that (4.6) is equivalent to a general LP problem, that is,

$$\begin{aligned} \min \mathbf{c}^T \mathbf{x}, \\ A\mathbf{x} = \mathbf{b}, \\ \mathbf{x} \geq \mathbf{0}. \end{aligned} \quad (4.8)$$

Furthermore, if  $T = \{\mathbf{x} \in \mathbb{R}_+^n, A\mathbf{x} = \mathbf{0}, \|\mathbf{x}\| = 1\}$ , the following result holds (see [29]).

**Theorem 4.4.** *Given a suitable integer  $L$  and the function.*

$$g(\mathbf{x}) = \sum_{j=1}^n \ln \frac{\mathbf{c}^T \mathbf{x}}{x_j} = n \ln \mathbf{c}^T \mathbf{x} - \sum_{j=1}^n \ln x_j, \quad (4.9)$$

then, (4.6) and hence (4.8) are equivalent to find a point  $\mathbf{x}^*$ :

$$\begin{aligned} \mathbf{x}^* \in T, \\ g(\mathbf{x}^*) < -2nL. \end{aligned} \quad (4.10)$$

Moreover, it is possible to determine a real number  $c_0$  and a sequence  $\{\mathbf{x}^{(k)}\} \in T$  by a GP algorithm with a suitable scaling procedure (see again [29]) such that

$$g(\mathbf{x}^{(k+1)}) < g(\mathbf{x}^{(k)}) - c_0. \quad (4.11)$$

By Theorem 4.4 and Definition 4.2 it follows that there exists a Gradient-type method (2.5) solving LP in a *finite* number of steps. Hence LP is a finite continuous constrained optimization problem. It is important to underline that the latter result is not a consequence of the *intrinsic finiteness* of the set of the possible optimal solutions (the vertices of a polyhedron) as in the classical simplex algorithm.

Given the convex functions,  $f(\mathbf{x}), h_1(\mathbf{x}), h_2(\mathbf{x}) \cdots h_m(\mathbf{x}) \in C^1(S)$ ,  $S$  convex  $\subseteq \mathbb{R}^n$ , let us now consider the general Convex Programming (CP) problem:

$$\begin{aligned} \min f(\mathbf{x}), \\ h_i(\mathbf{x}) \leq 0, \quad i = 1, 2, \dots, m, \\ \mathbf{x} \geq \mathbf{0}. \end{aligned} \quad (4.12)$$

The following property is well known [23, 26].

*Definition 4.5.* Letting  $\hat{\mathbf{x}} \geq 0$  and  $I = \{i : h_i(\hat{\mathbf{x}}) = 0\}$ , then the constraints of (4.12) are qualified if one of the following conditions is satisfied:

$$\exists \mathbf{x}^* \geq \mathbf{0} : h_i(\mathbf{x}^*) < 0, \quad i = 1, 2, \dots, m, \quad (4.13)$$

$$h_i(\hat{\mathbf{x}}) \text{ is locally concave in } \hat{\mathbf{x}}, \quad \forall i \in I, \forall \hat{\mathbf{x}}. \quad (4.14)$$

If  $h_i(\mathbf{x}) = \mathbf{c}_i^T \mathbf{x}$ , then (4.14) is trivially satisfied for all  $\hat{\mathbf{x}}$ , for all  $i \in I$ .

So, from Definition 4.5 we deduce that the constraints of CPLC problem (4.4) are always qualified. Assuming in (4.4)  $A$  with maximum rank, we clearly obtain a condition equivalent to (4.13).

*Definition 4.6.* A set  $C \subseteq \mathbb{R}^n$  is called a *convex cone* if

$$\begin{aligned} \mathbf{x} \in C &\implies \lambda \mathbf{x} \in C, \quad \forall \lambda > 0, \\ \forall \mathbf{x}^{(1)}, \mathbf{x}^{(2)} \in C, \quad \forall 0 \leq \lambda_1, \lambda_2 \leq 1, \quad \lambda_1 \mathbf{x}^{(1)} + \lambda_2 \mathbf{x}^{(2)} &\in C. \end{aligned} \quad (4.15)$$

The following theorem was proved in [30] in a general Hilbert space (see Theorem 2.3).

**Theorem 4.7.** *Let  $S_1, S_2 \in \mathbb{R}^n$  be closed convex cones, and let  $S_1^\circ$  denote the interior of  $S_1$ . Assume that  $S_1^\circ \neq \emptyset$ .*

*Then the corresponding conic feasibility problem*

$$\text{find } \mathbf{x} \in S_1^\circ \cap S_2 \quad (4.16)$$

*can be solved in a finite number of steps.*

The technique utilized to prove Theorem 4.7 is based upon the so-called *Method of Alternative Projections* (MAP) (see [31]).

Theorem 4.7 was extended in [30] (see Proposition 2.1) by assuming  $S_1$  and  $S_2$  be closed convex sets, thereby proving that a *convex feasibility problem* is equivalent to a conic feasibility problem. However, the open question remains how to express explicit formulas for the projection operators to convert the algorithm from  $S_1$  and  $S_2$  to the *conified* closed sets  $\overline{\text{con}}(S_1)$  and  $\overline{\text{con}}(S_2)$  in the case of nonlinear and nonquadratic problems. The Linear Matrix Inequality (LMI) feasibility problem was, in fact, efficiently solved in the literature (see [32]).

*Remark 4.8.* Theorem 4.7 can be applied to CPLC problem (4.4), by assuming

$$S_1 = \{\mathbf{x} \in \mathbb{R}^n : A\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}, \quad (4.17)$$

$$S_2^{(k)} = \left\{ \mathbf{x} \in \mathbb{R}^n : f(\mathbf{x}) \leq t^{(k)}, t^{(k)} \in \mathbb{R} \right\}, \quad k = 1, 2, \dots, k_0.$$

Hence, explicit formulas for the projection operators for suitable classes of nonlinear convex feasibility problems in terms of the corresponding conified sets might allow to solve CPLC problem (4.4) in the nonquadratic case with a *finite number of steps*. By utilizing Theorem 3.1, we can prove, in fact, the following important theorem.

**Theorem 4.9** (see [33]). *Consider the particular CPLC problem*

$$\begin{aligned} \min \mathbf{x}^T C \mathbf{x} * (\mathbf{c}^T \mathbf{x})^{-2}, \quad C \text{ spd}, \\ A \mathbf{x} = \mathbf{b}, \\ -\mathbf{c}^T \mathbf{x} \leq 0, \\ \mathbf{x} \geq \mathbf{0}. \end{aligned} \tag{4.18}$$

Assume that the optimal solution  $\mathbf{x}_*$  of problem (4.18) be such that  $-\mathbf{c}^T \mathbf{x}_* < 0$ . Then, (4.18) can be converted into a convex feasibility problem by utilizing a proper modification of the Alternative Projection method, and the latter algorithm converges to the optimal solution with a finite number of steps.

*Remark 4.10.* Given the convex set of feasible solutions

$$S_1 = \left\{ \mathbf{x} \in \mathbb{R}^n : A \mathbf{x} = \mathbf{b}, -\mathbf{c}^T \mathbf{x} \leq 0, \mathbf{x} \geq \mathbf{0} \right\}, \tag{4.19}$$

the proof of Theorem 4.9 is essentially based upon the following computational ingredients:

- (a) by Theorem 4.7, one can convert the closed convex set defined in (4.19) into a closed convex cone;
- (b) by Theorem 3.1, the extended version of (CG)-method and a suitable projection algorithm can be applied to problem (4.18) thereby obtaining a convergence with a finite number of steps.

## 5. Global Optimization

One can prove the following global convergence theorem [7].

**Theorem 5.1.** *Consider Problem (2.1), where  $f(\mathbf{x}) \in C^2(\mathbb{R}^n)$ .*

*Let  $f_{\min}$  be the value of the optimal solution. Assume that*

$$\begin{aligned} \forall \epsilon_a \in \mathfrak{R}^+, \exists \epsilon_s \in \mathfrak{R}^+ : \|\nabla f(\mathbf{x}^{(k)})\| > \epsilon_s \text{ except for } k : \\ f(\mathbf{x}^{(k)}) - f_{\min} < \epsilon_a. \end{aligned} \tag{5.1}$$

*If in an iterative scheme of BFGS-type,*

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \lambda_k B^{(k)-1} \nabla f(\mathbf{x}^{(k)}), \quad (B^{(k)} = \varphi(\tilde{B}^{(k-1)}, \dots), \forall k), \tag{5.2}$$

the following conditions are satisfied:

$$\frac{\|\nabla f(\mathbf{x}^{(k+1)}) - \nabla f(\mathbf{x}^{(k)})\|^2}{(\nabla f(\mathbf{x}^{(k+1)}) - \nabla f(\mathbf{x}^{(k)}))^T \lambda_k \mathbf{d}^{(k)}} = \frac{\|\mathbf{y}_k\|^2}{\mathbf{y}_k^T \mathbf{s}_k} \leq M, \quad (5.3)$$

$$\text{cond}(B^{(k)}) \leq N. \quad (5.4)$$

Then

$$\forall \epsilon_a \in \mathfrak{R}^+, \exists k^{**} : \forall k > k^{**} \quad f(\mathbf{x}^{(k)}) - f_{\min} < \epsilon_a. \quad (5.5)$$

Theorem 5.1 points out as follows three conditions for a global optimization BFGS-type method.

Condition (5.1) assumes an *optimal matching* between the BFGS-type algorithm and the function  $F$  [34]. (5.3) is equivalent to (3.12), that is,  $f(\mathbf{x})$  is weakly convex (see [24]).

Condition (5.4) can be easily satisfied, by modifying the matrices  $B^{(k)}$  by a restarting procedure, because *every descent direction is associated to an spd matrix* (see Theorem 2.1).

Let us now consider the classical “box-constrained” problem:

$$\min f(\mathbf{x}), \quad \mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U. \quad (5.6)$$

Let  $\mathbf{x}_{c(m)}^L \leq \mathbf{x}_{c(m)} \leq \mathbf{x}_{c(m)}^U$  denote the *current box* at iteration  $m$ .

Set

$$\alpha_{\mathbf{x}_{c(m)}} = \max \left\{ 0, -\frac{1}{2} \min_i \lambda_i \left\{ \nabla^2 f(\mathbf{x}_{c(m)}) \right\} \right\}, \quad (5.7)$$

$$L_{c(m)}(\mathbf{x}_{c(m)}) = f(\mathbf{x}_{c(m)}) + \alpha_{\mathbf{x}_{c(m)}} \left( \mathbf{x}_{c(m)}^L - \mathbf{x}_{c(m)} \right) \left( \mathbf{x}_{c(m)}^U - \mathbf{x}_{c(m)} \right). \quad (5.8)$$

The following global convergence theorem holds (see [35, 36]).

**Theorem 5.2.** Consider Problem (5.6) and assume  $f(\mathbf{x}) \in C^2$ . These hypotheses imply

$$\text{cond}(\nabla^2 f(\mathbf{x})) \leq c, \quad \forall m \quad \exists \alpha_m^* = \max_{\mathbf{x}_{c(m)}} \alpha_{\mathbf{x}_{c(m)}}. \quad (5.9)$$

Set

$$f_{c(m)}^L = \inf_{\mathbf{x}_{c(m)}} L_{c(m)}(\mathbf{x}_{c(m)}), \quad f_{c(m)}^U = f\left(\frac{\mathbf{x}_{c(m)}^L + \mathbf{x}_{c(m)}^U}{2}\right). \quad (5.10)$$

Then, it follows for all  $m$ :

$$f_{c(m)}^L \leq f_{c(m+1)}^L \leq \min_{\mathbf{x}_{c(m+1)}} f(\mathbf{x}_{c(m+1)}) \equiv \min_{\mathbf{x}} f(\mathbf{x}), \quad (5.11)$$

$$f_{c(m)}^U \geq f_{c(m+1)}^U \geq \min_{\mathbf{x}} f(\mathbf{x}) \geq f_{c(m)}^L. \quad (5.12)$$

Moreover, for all  $\epsilon_a > 0$ ,  $\exists m^* : \text{for all } m \geq m^*$ :

$$\begin{aligned} f_{c(m)}^U - f_{c(m)}^L &< \epsilon_a, \\ \left\| \mathbf{x}_{c(m)}^U - \mathbf{x}_{c(m)}^L \right\|_2 &\leq \sqrt{\frac{4\epsilon_a}{c}}, \quad c \text{ constant} \end{aligned} \quad (5.13)$$

Theorem 5.2 can be immediately extended to Problem (2.1), by assuming a growth condition on the function  $f(\mathbf{x})$ .

In fact, we have the following.

**Corollary 5.3.** Given  $f(\mathbf{x}) \in C^2(R^n)$ :

$$\lim_{\|\mathbf{x}\| \rightarrow \infty} f(\mathbf{x}) = +\infty. \quad (5.14)$$

Equation (5.14) implies

$$\begin{aligned} \exists K_0 : \min_{\mathbf{x} \in R^n} f(\mathbf{x}) &\equiv \min_{\|\mathbf{x}\| \leq K_0} f(\mathbf{x}), \\ \left\| \nabla^2 f(\mathbf{x}) \right\| &\leq c_1, \quad \|\mathbf{x}\| \leq K_0. \end{aligned} \quad (5.15)$$

Assume

$$\left\| \nabla^2 f(\mathbf{x})^{-1} \right\| \leq c_2 \text{ and hence } \text{cond}(\nabla^2 f(\mathbf{x})) \leq c_1 c_2. \quad (5.16)$$

Then, the convergence results proved for (5.6) can be applied to (2.1).

We can fruitfully combine the results of Theorems 5.1 and 5.2, by proving the following.

**Theorem 5.4.** Consider Problem (5.6) and assume  $f(\mathbf{x}) \in C^2(R^n)$ .  
If in a BFGS-type iterative scheme

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \mu_k \mathcal{B}^{(k)-1} \nabla f(\mathbf{x}^{(k)}), \quad (5.17)$$

the following conditions are satisfied:

$$\mathbf{x}^L \leq \mathbf{x}^{(k)} \leq \mathbf{x}^U, \quad \forall k, \quad (5.18)$$

$$\text{cond}(\mathcal{B}^{(k)}) \leq N, \quad (5.19)$$

$$\frac{\|\nabla f(\mathbf{x}^{(k+1)}) - \nabla f(\mathbf{x}^{(k)})\|^2}{(\nabla f(\mathbf{x}^{(k+1)}) - \nabla f(\mathbf{x}^{(k)}))^T \lambda_k \mathbf{d}^{(k)}} = \frac{\|\mathbf{y}_k\|^2}{\mathbf{y}_k^T \mathbf{s}_k} \leq M. \quad (5.20)$$

Then (5.17) is convergent to the optimal solution of (5.6).

*Proof.* By the assumptions it follows:  $\text{cond}(\nabla^2 F(\mathbf{x})) \leq c$ .

Hence, by (5.7) we have for all  $m$ :

$$\exists \alpha_m^* = \max_{\mathbf{x}_{c(m)}} \alpha_{\mathbf{x}_{c(m)}}. \quad (5.21)$$

Set

$$\tilde{L}_{c(m)}(\mathbf{x}_{c(m)}) = f(\mathbf{x}_{c(m)}) + \alpha_m^* (\mathbf{x}_{c(m)}^L - \mathbf{x}_{c(m)}) (\mathbf{x}_{c(m)}^U - \mathbf{x}_{c(m)}). \quad (5.22)$$

Therefore, by (5.21) and (5.22) for all  $m$ :

$$\begin{aligned} L_{c(m)}(\mathbf{x}_{c(m)}) &\geq \tilde{L}_{c(m)}(\mathbf{x}_{c(m)}), \quad \forall \mathbf{x}_{c(m)}. \\ \tilde{L}_{c(m)}(\mathbf{x}_{c(m)}) &\text{convex } \forall \mathbf{x}_{c(m)}. \end{aligned} \quad (5.23)$$

So,

$$f_{c(m)}^L = \inf_{\mathbf{x}_{c(m)}} L_{c(m)}(\mathbf{x}_{c(m)}) = \min_{\mathbf{x}_{c(m)}} \tilde{L}_{c(m)}(\mathbf{x}_{c(m)}), \quad \forall m. \quad (5.24)$$

Let  $\mathbf{x}_{c(m)}^{(\tilde{k}_m)}$  be a local minimum in the box  $c(m)$ .

If  $\mathbf{x}_{c(m+1)}^L \leq \mathbf{x}_{c(m)}^{(\tilde{k}_m)} \leq \mathbf{x}_{c(m+1)}^U$  and  $f((\mathbf{x}_{c(m+1)}^L + \mathbf{x}_{c(m+1)}^U)/2) \geq f(\mathbf{x}_{c(m)}^{(\tilde{k}_m)})$ , then define

$$f_{c(m+1)}^U = f(\mathbf{x}_{c(m)}^{(\tilde{k}_m)}). \quad (5.25)$$

Else, set

$$\begin{aligned} \mathbf{x}_{c(m+1)}^{(0)} &= \frac{\left(\mathbf{x}_{c(m+1)}^L + \mathbf{x}_{c(m+1)}^U\right)}{2}, \\ f_{c(m+1)}^U &= f\left(\tilde{\mathbf{x}}_{c(m+1)}^{(k_{m+1})}\right) \end{aligned} \quad (5.26)$$

with  $\tilde{\mathbf{x}}_{c(m+1)}^{(k_{m+1})}$  being a local minimum evaluated by the starting point  $\mathbf{x}_{c(m+1)}^{(0)}$  and contained in the box  $c(m+1)$ . Since the assumptions of Theorem 5.2 are satisfied, by the results of [7] (see Theorem 2 and Corollary 2), it follows that (5.19), (5.20) imply that for all  $\epsilon_b > 0$ ,  $\exists\{\tilde{\mathbf{x}}_{c(m_i)}^{(k_{m_i})}\}$ :

$$\begin{aligned} f_{c(m_i)}^U &\geq f_{c(m_{i+1})}^U, \\ \left\| \nabla f\left(\tilde{\mathbf{x}}_{c(m_i)}^{(k_{m_i})}\right) \right\| &< \epsilon_b. \end{aligned} \quad (5.27)$$

Applying Theorem 5.2, by inequalities (5.11) and (5.27) and by setting  $\epsilon = \max\{\epsilon_a, \epsilon_b\}$ , we have that for all  $\epsilon > 0$ ,  $\exists\{\tilde{\mathbf{x}}_{c(m_i)}^{(k_{m_i})}\}$ ,

$$\begin{aligned} \left\| \nabla f\left(\tilde{\mathbf{x}}_{c(m_i)}^{(k_{m_i})}\right) \right\|_2 &< \epsilon, \\ \left\| \mathbf{x}_{c(m_i)}^U - \mathbf{x}_{c(m_i)}^L \right\|_2 &\leq \sqrt{\frac{4\epsilon}{c}}, \\ f\left(\tilde{\mathbf{x}}_{c(m_i)}^{(k_{m_i})}\right) - f_{\min} &\leq f_{c(m_i)}^U - f_{c(m_i)}^L < \epsilon. \end{aligned} \quad (5.28)$$

This completes the proof.  $\square$

Although the local minimization phases are performed effectively by the iterative scheme (5.17), the convergence of the method to the global minimum is usually very slow by the very nature of the  $\alpha BB$  approach. In particular, the number of the upper bounds  $f_{c(m_i)}^U$  and the corresponding boxes  $m_i$ , requested to obtain a satisfactory approximation can be unacceptable from a computational point of view. In order to overcome this problem, a fast determination of “good” local minima is essential.

More precisely, by the utilization of *terminal repellers* and *tunneling techniques* [14], one can build algorithms based on a sequence of cycles, where each cycle has two phases, that is, a local optimization phase and a tunneling one. The main aim of these procedures is to build a favourable sequence of local minima (maxima), thereby determining a set of possible candidates for the global minimum (maximum) more efficiently.

By injecting in the method suitable “tunneling phases,” one can avoid the unfair entrapment in a “bad” local minimum, that is, when the condition

$$f_{c(m+1)}^U = f\left(\tilde{\mathbf{x}}_{c(m+1)}^{(k_{m+1})}\right) = f\left(\tilde{\mathbf{x}}_{c(m+1)}^{(k_m)}\right) = f_{c(m)}^U \gg f_{\min} \quad (5.29)$$

is verified for several iterations. For this purpose, the *power of the repellers*, utilized in the tunneling phases, plays a crucial role. The classical and well-known use of scalar repellers [14, 34] is often unsuitable, when the dimension  $n$  of the problem assumes values of operational interest. A repeller structured matrix, based on the sum of a diagonal matrix and a low-rank one [15], can be constructed to overcome the latter difficulty.

Let  $\mathbf{x}^{(\tilde{k})}$  be an approximation of a local minimizer for  $f(\mathbf{x}) \in C^1$ .

A matrix  $\mathcal{A}^{(\tilde{k})}$  is called a *repeller matrix* for  $\mathbf{x}^{(\tilde{k})}$  if  $\exists \hat{\mathbf{x}}$ ,

$$\begin{aligned}\hat{\mathbf{x}} &= \mathbf{x}^{(\tilde{k})} - \mathcal{A}^{(\tilde{k})} \nabla f(\mathbf{x}^{(\tilde{k})}), \\ f(\hat{\mathbf{x}}) &< f(\mathbf{x}^{(\tilde{k})}).\end{aligned}\tag{5.30}$$

The repeller matrix  $\mathcal{A}^{(\tilde{k})}$  for any given computed local minimizer  $\mathbf{x}^{(\tilde{k})}$  can be approximated in the following way (see [15]):

$$\mathcal{A}^{(\tilde{k})} \approx \lambda^{(\tilde{k})} I + \left( \frac{I}{\mu} + R \right)^{-1}, \quad 2 \leq \text{rank}(R) \leq 4 \tag{5.31}$$

with  $\lambda^{(\tilde{k})}$  being the *maximal scalar repeller* [34] that is,

$$\lambda^{(\tilde{k})} = \frac{\epsilon_a}{\left\| \nabla f(\mathbf{x}^{(\tilde{k})}) \right\|^2}, \quad \left\| \nabla f(\mathbf{x}^{(\tilde{k})}) \right\| \ll \sqrt{\epsilon_a}, \quad \epsilon_a \text{ desired precision}, \tag{5.32}$$

with  $R$  being of the following structure:

$$\begin{aligned}R &= \mu_1 \mathbf{p}\mathbf{p}^T + \mu_2 \mathbf{q}\mathbf{q}^T + \mu_3 \mathbf{p}\mathbf{r}^T + \mu_4 \mathbf{r}\mathbf{q}^T \\ \mathbf{p}, \mathbf{q}, \mathbf{r} &\text{ suitable vectors } \mu_1, \mu_2, \mu_3, \mu_4 \text{ scalars.}\end{aligned}\tag{5.33}$$

In this way, the application of a BFGS-type method can be effectively extended to the tunneling phases and hence to the whole global optimization scheme (see [9, 33]).

The structure in (5.33) can be generalized by using the recent Tensor-Train (TT)-cross approximation theory [13].

It is well known, in fact, that a rank- $p$  matrix can be recovered from a cross of  $p$  linearly independent columns (or rows). Therefore, an arbitrary matrix can be interpolated by a *pseudoskeleton approximation* (see [15] and again [13]). In particular, since a repeller matrix is not arbitrary and possesses some hidden structure, it is fundamental to discover a *low-parametric representation*, which can be useful in the tunneling phases.

An operational cross approximation method, evaluating large close-to-rank- $p$  matrices in  $\mathcal{O}(np^2)$  time complexity and by computing  $\mathcal{O}(np)$  elements, was shown in [37].

## 6. Discrete Optimization

A well-known family of Computer Science methods is represented by the so-called *Greedy algorithms*. The simplest application of this type of procedures is in the standard Knapsack Problem (KP), that is,

$$\begin{aligned} \max \mathbf{c}^T \mathbf{x}, \\ \mathbf{a}^T \mathbf{x} \leq b, \\ \mathbf{x} \geq \mathbf{0}, \text{ integer.} \end{aligned} \quad (6.1)$$

Greedy approach is essentially a generalization of the classical *Dynamical Programming* (DP) methods, which are based on the *Bellman Principle*. By utilizing the DP computational scheme and assuming  $y$  integer, problem (6.1) can be reduced to the recursive solution of the following family of problems:

$$\begin{aligned} \max \mathbf{c}^{(k)T} \mathbf{x}^{(k)}, \\ \mathbf{a}^{(k)T} \mathbf{x}^{(k)} \leq y, \\ \mathbf{x}^{(k)} \geq \mathbf{0}, \text{ integer,} \\ 1 \leq k \leq n, \quad 1 \leq y \leq b, \text{ integer,} \end{aligned} \quad (6.2)$$

where  $\mathbf{c}^{(k)}$ ,  $\mathbf{a}^{(k)}$ ,  $\mathbf{x}^{(k)}$  indicate the vectors associated to the first  $k$  components of  $\mathbf{c}, \mathbf{a}, \mathbf{x}$ , respectively.

Given  $k$  and  $y$ , let  $\psi_k(y)$  be the value of the objective function corresponding to the optimal solution of problem (6.2).

The algorithm computes  $\psi_k(y)$  by the recursive formula

$$\psi_k(y) = \max \left\{ \psi_{k-1}(y), \psi_k(y - a_k^{(k)}) + c_k^{(k)} \right\}. \quad (6.3)$$

By (6.3), the optimal value of (6.2) is determined by a generalized discrete Steepest Descent algorithm, since  $c_k^{(k)}$  is the  $k$ .th component of the gradient of the objective function and represents, in fact, *the increase* associated to the choice of the  $k$ .th object.

Therefore, formula (6.3) is based on a discrete Steepest Descent approach, and the value  $\psi_k(y - a_k^{(k)}) + c_k^{(k)}$  assures that the corresponding solution is admissible.

Integer Nonlinear Programming with Linear Constraints problems (INPLCs) can be transformed into continuous GO problems over the unit hypercube [17]. In order to reduce the difficulties caused by the introduction of undesirable local minimizers, a special class of continuation methods, called smoothing methods, can be introduced [38]. These methods deform the original objective function into a function whose smoothness is controlled by a parameter. Of course, the success of the latter approach depends on the existence of a suitable smoothing function.

Hence, the Gradient-type methods for Global Optimization of Section 4 can be also applied to INPLC.

## 7. Conclusions

In this paper we have tried to demonstrate that Gradient or Gradient-type methods lead both to a general approach to optimization problems and to the construction of efficient algorithms.

In particular, we have shown that the class of problems for which the optimal solution can be obtained in a finite number of steps is larger than canonical unconstrained Convex Quadratic problems or Convex Quadratic Programming. Moreover, we have pointed out that the classical distinction between Direct Methods and Iterative Methods cannot be considered as a fundamental classification of techniques in Numerical Analysis. Many optimization problems can be, in fact, solved in a finite number of steps by suitable hybrid efficient algorithms (see [33]).

Furthermore, if the matrices involved in the computation are well conditioned, the superiority of Iterative Methods with respect to Direct ones, which is a typical feature of (CG) algorithm, can be proved in a more general context (see again [33]).

Several *heuristic* and *ad hoc* algorithms in operational environments can be considered, in fact, as particular cases of a general Gradient-type approach to the problem. In some cases, surprisingly enough, the convergence of Iterative Methods can be guaranteed only by utilizing a special Line-Search Minimization algorithm (see f.i. Fletcher-Reeves method in conjunction with Armijo-Goldstein-Wolfe's procedure, [18], Theorem 5.8).

It is also important to underline that many combinatorial problems, representing a remarkable benchmark set in Computer Science, can be translated in terms of Gradient-type methods in a general framework.

Once again, we stress that the Fixed Point theorem, which is considered a milestone in Numerical Analysis and guarantees the convergence of most of classical Iterative Methods, represents the background for only a subset of Gradient-type methods.

## Acknowledgment

This paper was partially supported by PRIN 2008 N. 20083KLJEZ.

## References

- [1] J. Abaffy and E. Spedicato, *ABS Projection Algorithms*, Ellis Horwood Series: Mathematics and Its Applications, Ellis Horwood, Chichester, UK, 1989.
- [2] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, UK, 2004.
- [3] A. Forsgren, P. E. Gill, and M. H. Wright, "Interior methods for nonlinear optimization," *SIAM Review*, vol. 44, no. 4, pp. 525–597, 2002.
- [4] S. G. Nash and A. Sofer, *Linear and NonLinear Programming*, McGraw-Hill, New York, NY, USA, 1995.
- [5] C. Di Fiore, S. Fanelli, F. Lepore, and P. Zellini, "Matrix algebras in quasi-Newton methods for unconstrained minimization," *Numerische Mathematik*, vol. 94, no. 3, pp. 479–500, 2003.
- [6] C. Di Fiore, S. Fanelli, and P. Zellini, "Low-complexity minimization algorithms," *Numerical Linear Algebra with Applications*, vol. 12, no. 8, pp. 755–768, 2005.
- [7] C. Di Fiore, S. Fanelli, and P. Zellini, "Low complexity secant quasi-Newton minimization algorithms for nonconvex functions," *Journal of Computational and Applied Mathematics*, vol. 210, no. 1-2, pp. 167–174, 2007.
- [8] C. Di Fiore, F. Lepore, and P. Zellini, "Hartley-type algebras in displacement and optimization strategies," *Linear Algebra and Its Applications*, vol. 366, pp. 215–232, 2003.

- [9] S. Fanelli, "A new algorithm for box-constrained global optimization," *Journal of Optimization Theory and Applications*, vol. 149, no. 1, pp. 175–196, 2011.
- [10] C. Di Fiore, "Structured matrices in unconstrained minimization methods," in *Fast algorithms for Structured Matrices: Theory and Applications (South Hadley, MA, 2001)*, vol. 323 of *Contemporary Mathematics*, pp. 205–219, American Mathematical Society, Providence, RI, USA, 2003.
- [11] A. Bortoletti, C. Di Fiore, S. Fanelli, and P. Zellini, "A new class of quasi-Newtonian methods for optimal learning in MLP-networks," *IEEE Transactions on Neural Networks*, vol. 14, no. 2, pp. 263–273, 2003.
- [12] J.-F. Cai, R. H. Chan, and C. Di Fiore, "Minimization of a detail-preserving regularization functional for impulse noise removal," *Journal of Mathematical Imaging and Vision*, vol. 29, no. 1, pp. 79–91, 2007.
- [13] I. Oseledets and E. Tyrtyshnikov, "TT-cross approximation for multidimensional arrays," *Linear Algebra and Its Applications*, vol. 432, no. 1, pp. 70–88, 2010.
- [14] B. C. Cetin, J. Barhen, and J. W. Burdick, "Terminal repeller unconstrained subenergy tunneling (TRUST) for fast global optimization," *Journal of Optimization Theory and Applications*, vol. 77, no. 1, pp. 97–126, 1993.
- [15] I. Oseledets and E. Tyrtyshnikov, "A unifying approach to the construction of circulant preconditioners," *Linear Algebra and Its Applications*, vol. 418, no. 2-3, pp. 435–449, 2006.
- [16] R. P. Ge and C. B. Huang, "A continuous approach to nonlinear integer programming," *Applied Mathematics and Computation*, vol. 34, no. 1, part I, pp. 39–60, 1989.
- [17] K. M. Ng, *A continuation approach for solving nonlinear optimization problems with discrete variables*, Ph.D. Dissertation, Stanford University, Palo Alto, Calif, USA, 2002.
- [18] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer Series in Operations Research, Springer, New York, NY, USA, 1999.
- [19] C. Di Fiore, S. Fanelli, and P. Zellini, "An efficient generalization of Battiti-Shanno's Quasi-Newton Algorithm for learning in MLP-networks," in *Proceedings of the International Conference on Neural Information Processing (ICONIP '04)*, pp. 483–488, Springer, Calcutta, India, 2004.
- [20] J. Stoer, *Einführung in die Numerische Mathematik. I*, Springer, Berlin, Germany, 1972.
- [21] J. E. Dennis, Jr. and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice Hall Series in Computational Mathematics, Prentice Hall, Englewood Cliffs, NJ, USA, 1983.
- [22] L. Lukšan, "Conjugate gradient algorithms for conic functions," *Československá Akademie Věd*, vol. 31, no. 6, pp. 427–440, 1986.
- [23] D. G. Luemberger, *Linear and Nonlinear Programming*, Addison-Wesley, Reading, Mass, USA, 1984.
- [24] M. J. D. Powell, "Some global convergence properties of a variable metric algorithm for minimization without exact line searches," in *Nonlinear Programming (Proc. Sympos., New York, 1975)*, pp. 53–72, American Mathematical Society, Providence, RI, USA, 1976.
- [25] M. Frank and P. Wolfe, "An algorithm for quadratic programming," *Naval Research Logistics Quarterly*, vol. 3, pp. 95–110, 1956.
- [26] M. Minoux, *Mathematical Programming: Theory and Algorithms*, A Wiley-Interscience Publication, John Wiley & Sons, Chichester, UK, 1986.
- [27] J. B. Rosen, "The gradient projection method for nonlinear programming. I. Linear constraints," vol. 8, pp. 181–217, 1960.
- [28] N. Karmarkar, "A new polynomial-time algorithm for linear programming," *Combinatorica*, vol. 4, no. 4, pp. 373–395, 1984.
- [29] C. C. Gonzaga, "Conical projection algorithms for linear programming," *Mathematical Programming*, vol. 43, no. 2, pp. 151–173, 1989.
- [30] M. A. Rami, U. Helmke, and J. B. Moore, "A finite steps algorithm for solving convex feasibility problems," *Journal of Global Optimization*, vol. 38, no. 1, pp. 143–160, 2007.
- [31] H. H. Bauschke and J. M. Borwein, "On projection algorithms for solving convex feasibility problems," *SIAM Review*, vol. 38, no. 3, pp. 367–426, 1996.
- [32] L. Vandenberghe and S. Boyd, "Semidefinite programming," *SIAM Review*, vol. 38, no. 1, pp. 49–95, 1996.
- [33] S. Fanelli, Finite iterative procedures in optimization, in preparation.
- [34] C. Di Fiore, S. Fanelli, and P. Zellini, "Computational experiences of a novel global algorithm for optimal learning in MLP-networks," in *Proceedings of the International Conference on Neural Information Processing (ICONIP '02)*, vol. 1, pp. 317–321, 2002.

- [35] C. A. Floudas, *Deterministic Global Optimization*, vol. 37 of *Nonconvex Optimization and Its Applications*, Kluwer Academic, Dordrecht, The Netherlands, 2000.
- [36] C. A. Floudas and V. Visweswaran, "Primal-relaxed dual global optimization approach," *Journal of Optimization Theory and Applications*, vol. 78, no. 2, pp. 187–225, 1993.
- [37] E. E. Tyrtshnikov, "Incomplete cross approximation in the mosaic-skeleton method," *Computing*, vol. 64, no. 4, pp. 367–380, 2000.
- [38] J. J. Moré and Z. Wu, "Global continuation for distance geometry problems," *SIAM Journal on Optimization*, vol. 7, no. 3, pp. 814–836, 1997.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

