

# How *grossone* can be helpful to iteratively compute negative curvature directions

Renato De Leone<sup>1</sup>[0000-0001-9783-608X], Giovanni Fasano<sup>2</sup>[0000-0003-4721-8114],  
Massimo Roma<sup>3</sup>[0000-0002-9858-3616], and Yaroslav D.  
Sergeyev<sup>4</sup>[0000-0002-1429-069X]

<sup>1</sup> Scuola di Scienze e Tecnologie, Università di Camerino, Camerino, Italy,  
`renato.deleone@unicam.it`

<sup>2</sup> Dipartimento di Management, Università Ca' Foscari Venezia, Venice, Italy,  
`fasano@unive.it`

<sup>3</sup> Dipartimento di Ingegneria Informatica, Automatica e Gestionale 'A. Ruberti',  
SAPIENZA, Università di Roma, Rome, Italy,  
`roma@dis.uniroma1.it`

<sup>4</sup> Dipartimento di Ingegneria Informatica, Modellistica, Elettronica e Sistemistica,  
Università della Calabria, Rende, Italy,  
`yaro@dimes.unical.it`

**Abstract.** We consider an iterative computation of negative curvature directions, in large scale optimization frameworks. We show that to the latter purpose, borrowing the ideas in [1], [3] and [4], we can fruitfully pair the Conjugate Gradient (CG) method with a recently introduced numerical approach involving the use of *grossone* [5]. In particular, though in principle the CG method is well-posed only on positive definite linear systems, the use of *grossone* can enhance the performance of the CG, allowing the computation of negative curvature directions, too. The overall method in our proposal significantly generalizes the theory proposed for [1] and [3], and straightforwardly allows the use of a CG-based method on indefinite Newton's equations.

**Keywords:** negative curvature directions, second order necessary optimality conditions, *grossone*, conjugate gradient method

## 1 Introduction

When considering the solution of the unconstrained optimization problem

$$\min_{x \in \mathbb{R}^n} f(x), \quad (1)$$

where  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  is a nonlinear smooth function and  $n$  is large, specific methods should be applied in case stationary points satisfying second order necessary optimality conditions are sought. Moreover, additional cares on the algorithms adopted are definitely mandatory, since standard stationarity conditions in general do not ensure the convexity of the objective function in a neighborhood of

critical points. In this regard, the computation of negative curvature directions for the objective function is an essential tool, to guarantee convergence to second order points.

Observe that convergence towards second order stationary points, where the Hessian matrix is positive semidefinite, requires to efficiently explore the eigenspaces of the Hessian matrix at the solution. Thus, the optimization method adopted to solve (1) should be able to cope also with nonconvexities of the objective function. In particular, as showed in [3], this can be accomplished by suitable Krylov-subspace methods, such that at each iteration  $j$ , a pair of directions  $(s_j, d_j)$  is computed satisfying specific properties. The vector  $s_j$  must be a direction which approximately solves Newton's equation  $\nabla^2 f(x_j) s = -\nabla f(x_j)$  at  $x_j$ . Its purpose is essentially to ensure convergence to stationary points. On the other hand, the direction  $d_j$  is a so called *negative curvature direction*, which is used to possibly force convergence to stationary points which satisfy second order necessary optimality conditions [9]. This implies that the sequence  $\{d_j\}$  is expected to satisfy the next assumption (which might be possibly weakened).

**Assumption 1** *The vectors in the sequence  $\{d_j\}$  are bounded and*

$$d_j^T \nabla^2 f(x_j) d_j < 0$$

$$d_j^T \nabla^2 f(x_j) d_j \rightarrow 0 \quad \implies \quad \min [0, \lambda^{\min}(\nabla^2 f(x_j))] \rightarrow 0,$$

*being  $\lambda^{\min}(\nabla^2 f(x_j))$  the smallest eigenvalue of the Hessian matrix  $\nabla^2 f(x_j)$ .*

Roughly speaking, the conditions in Assumption 1 imply that the negative curvature directions  $\{d_j\}$  need to eventually approximate an eigenvector associated with the smallest negative eigenvalue of the Hessian matrix. In [3] indications on the computation of the pair  $(s_j, d_j)$  were given, though the computation of  $d_j$  involved the use of Planar-CG methods, which impose an heavy computation in case some (so called) planar iterations are performed. This approach proved to be effective but required a complex analysis involving different articulated subcases. Here, we aim at describing a strong simplification in the computation the directions  $\{d_j\}$ , by using a novel approach which extends some ideas in [4].

## 2 Our proposal

To this purpose, we use the Krylov-subspace method in [1], for indefinite Newton's equations. This method alternates standard CG iterations and *planar* iterations, each of them being equivalent to a double CG iteration. This method is used to satisfy the next lemma (the proof follows from Theorem 3.2 in [3]).

**Lemma 1.** *Given problem (1), suppose at iteration  $n$  (being  $n$  the space dimension) of the Krylov-subspace method used to solve Newton's equation  $\nabla^2 f(x_j) s = \nabla f(x_j)$ , the decompositions*

$$\nabla^2 f(x_j) R_j = R_j T_j, \quad T_j = L_j B_j L_j^T$$



