Numerical Algorithms **30:** 241–258, 2002.

Preconditioned spectral gradient method

F. Luengo [a,*], M. Raydan [b,**], W. Glunt [c,***] and T.L. Hayden [d,***]

[a] *Dpto. de Matemáticas y Computación, Facultad de Ciencias, Universidad del Zulia, Ap. 527, Maracaibo, Venezuela*
E-mail: fluengo@cantv.net
[b] *Dpto. de Computación, Universidad Central de Venezuela, Ap. 47002, Caracas 1041-A, Venezuela*
E-mail: mraydan@reacciun.ve
[c] *Department of Mathematics and Computer Science, Austin Peay State University, Clarksville, TN 37044, USA*
E-mail: bud@ms.uky.edu
[d] *Department of Mathematics, University of Kentucky, Lexington, KY 40506, USA*
E-mail: hayden@ms.uky.edu

Received 4 July 2001; revised 14 March 2002
Communicated by C. Breziski

The spectral gradient method is a nonmonotone gradient method for large-scale unconstrained minimization. We strengthen the algorithm by modifications which globalize the method and present strategies to apply preconditioning techniques. The modified algorithm replaces a condition of uniform positive definitness of the preconditioning matrices, with mild conditions on the search directions. The result is a robust algorithm which is effective on very large problems. Encouraging numerical experiments are presented for a variety of standard test problems, for solving nonlinear Poisson-type equations, an also for finding molecular conformations by distance geometry.

Keywords: Spectral gradient method, preconditioning techniques, multidimensional scaling, nonmonotone line search, Poisson-type equations

AMS subject classification: 49M07, 49M10, 90C06, 65K

1. Introduction

We consider the unconstrained minimization problem:

$$\min_{x\in\Re^n} f(x), \tag{1}$$

where $f:\Re^n \to \Re$ is continuously differentiable and its gradient is available. We are

* Author was partially supported by CONICIT and UCV-PROJECT 97-003769.
** Author was partially supported by UCV-PROJECT 97-003769.
*** Authors were partially supported by NSF grant CHE-9301120.

interested in the large-scale case for which the Hessian of f is either not available or requires a prohibitive amount of storage.

The spectral gradient method, originally proposed by Barzilai and Borwein [6], has a number of interesting features that make it attractive for the numerical solution of (1). The iterations are defined by

$$x_{k+1} = x_k - \left(\frac{s_{k-1}^t s_{k-1}}{s_{k-1}^t y_{k-1}}\right) g_k,$$

where $s_{k-1} = x_k - x_{k-1}$, $y_{k-1} = g_k - g_{k-1}$ and g_k is the gradient of f at x_k. Notice that the inverse of the *steplength* (the scalar factor multiplying the search direction $-g_k$) is a Rayleigh quotient corresponding to the average Hessian matrix $\int_0^1 \nabla^2 f(x_{k-1}+ts_{k-1})\,\mathrm{d}t$. Obtaining the steplength in this way requires less computational work than a line search, and incorporates second order information into the search direction. In the quadratic case, the relationship to the eigenvalues and eigenvectors of the Hessian, the global convergence, and the efficiency of the method depending on the condition number of the Hessian matrix may be found in [14,25]. It has been applied succesfully to find local minimizers of large scale problems [7–10,12,13,23,29].

In this work we present new algorithms that combine and extend the following results: the globalization of the spectral gradient method by Raydan in [26] that is based on some ideas from [16], and an addition of a preconditioner by Glunt et al. [15] and Molina and Raydan [22].

The spectral gradient method uses the negative gradient vector as a search direction. Since preconditioning the gradient method can be viewed as a technique for calculating approximations to the Newton direction (especially if a good preconditioner is used), we will use preconditioning in the spectral gradient method to obtain a method that uses less storage than Newton's method, and obtains more accurate solutions with fewer iterations than the non-preconditioned version. We are assuming that the user is capable of supplying an inexpensive and efficient preconditioning matrix $G(x)$ that approximates the Hessian $H(x)$ of f near local minimizers.

Preconditioning for nonlinear problems is usually associated with outer-inner iterations (e.g., inexact Newton method with conjugate gradient inner iterations). In this work, it is used in a different way. We only consider one iterative process and our preconditioning matrix changes at every iteration. This approach is similar to the one used for preconditioning conjugate gradient methods in the nonquadratic case (see, e.g., [24,27]). In that sense we will discuss preconditioning for several standard test problems, a nonlinear Poisson-type equation and the Stress function associated with finding molecular conformations.

The paper is divided into sections as follows. In section 2 we combine the main ingredients of the algorithm, in a direct fashion, to obtain what we call the basic version. We discuss the convergence properties of this version under restrictive assumptions on the preconditioning matrices used during the process. The drawbacks of the basic algorithm lead us to present, in section 3, what we call the robust version. We replace a condition of uniform positive definitness of the preconditioning matrices with mild con-

ditions on the search directions. These conditions and some additional control steps in the robust algorithm will be transparent to the user, allowing more freedom in the choice of preconditioning schemes. Finally, in section 4, we present numerical results on three different type of problems to illustrate the behavior of the robust version of the algorithm.

2. Basic algorithm

The iterates of the Preconditioned Spectral Gradient (PSG) method presented by Glunt et al. [15] are defined by

$$x_{k+1} = x_k - \alpha_k^{-1} z_k,$$

where $z_k = G_k^{-1} g_k$, G_k is a nonsingular approximation to the Hessian of f at x_k and the scalar α_k is given by

$$\alpha_k = (-\alpha_{k-1}) \frac{z_{k-1}^t y_{k-1}}{z_{k-1}^t g_{k-1}},$$

where x_0 and α_0 are given initial data (see also [22]).

The PSG method requires no line search during the process but does not guarantee monotonic descent in the objective function. Recently, Raydan [26] proposed a globalization scheme for the spectral gradient algorithm that fits nicely with the nonmonotone behavior of this family of methods. Roughly speaking, the algorithm forces the step to satisfy this weak condition:

$$f(x_{k+1}) \leqslant \max_{0 \leqslant j \leqslant M} f(x_{k-j}) + \gamma g_k^t (x_{k+1} - x_k),$$

where M is a nonnegative integer and γ is a small positive number. When $M > 0$ this condition allows the objective function to increase at some iterations and still guarantees global convergence, as we discuss later. This globalization strategy is based on the nonmonotone line search technique of Grippo et al. [16].

A direct combination of the PSG method and the nonmonotone globalization strategy described above produces the following algorithm:

Global Preconditioned Spectral Gradient algorithm ("Basic" version)**.**
Given $x_0 \in \Re^n$, an integer $M \geqslant 0$, real numbers $\gamma \in (0, 1)$, $\delta > 0$, $0 < \sigma_1 < \sigma_2 < 1$, $0 < \varepsilon < 1$, and $\alpha_0 \in [\varepsilon, 1/\varepsilon]$. Set $k = 0$.

Step 1: If $\|g_k\|$ is sufficiently small then stop.

Step 2: Choose a preconditioning matrix $G_k \approx H(x_k)$, and solve $G_k z_k = -g_k$ for z_k.

Step 3: Set $\lambda = 1/\alpha_k$.

Step 4: (Nonmonotone line search)

$$\text{If } f(x_k + \lambda z_k) \leqslant \max_{0 \leqslant j \leqslant \min(k, M)} \{ f(x_{k-j}) \} + \gamma \lambda z_k^t g_k,$$

then $\lambda_k = \lambda$, $x_{k+1} = x_k + \lambda_k z_k$, and go to step 6.

Step 5: Choose $\sigma \in [\sigma_1, \sigma_2]$, set $\lambda = \sigma\lambda$, and go to step 4.

Step 6: Set $\alpha_{k+1} = -z_k^t(g_{k+1} - g_k)/\lambda_k(z_k^t g_k)$.

Step 7: If $((\alpha_{k+1} \geqslant 1/\varepsilon)$ or $(\alpha_{k+1} \leqslant \varepsilon))$ then $\alpha_{k+1} = \delta$.

Step 8: Set $k = k + 1$, and go to step 1.

Notice that the global PSG algorithm reduces to the global minimization method discussed by Raydan [26] if $G_k = I$ for every k, i.e., if the search direction is always chosen to be the negative gradient direction. Notice also that the global PSG algorithm reduces to the preconditioned algorithm presented by Glunt et al. [15] if the globalization steps 4, 5 and 7 are ignored.

We now present some convergence properties of the global PSG algorithm when applied to problem (1). These results are obtained under a restrictive assumption on the set of matrices G_k that are chosen during the process. This restrictive assumption will be removed in section 3. We discuss the convergence properties, and also weak points, of the "basic" PSG algorithm for two reasons. First, the results will be used later in section 3. Second, they will add significant understanding to the behavior of preconditioning techniques to find local minimizers of nonquadratic functions, and hence guide us to produce a robust version of the PSG algorithm.

Our next result establishes some properties of the search directions, in the global PSG algorithm, that are needed to prove global convergence in theorem 2.1.

Lemma 2.1. Suppose the sequence of symmetric $n \times n$ matrices $\{G_k\}$, in the global PSG algorithm, are chosen so that the eigenvalues satisfy

$$0 < \eta_{\min} \leqslant \eta_i(G_k) \leqslant \eta_{\max},$$

for all $i = 1, \ldots, n$ and all k. If $g_k \neq 0$ for each k, then

(i) $g_k^t G_k^{-1} g_k \geqslant (1/\eta_{\max})\|g_k\|_2^2$, and

(ii) $\|G_k^{-1} g_k\|_2 \leqslant (1/\eta_{\min})\|g_k\|_2$.

Proof. Straightforward. □

Theorem 2.1. Assume that $\Omega = \{x\colon\ f(x) \leqslant f(x_0)\}$ is a bounded set. Let $f : \Re^n \to \Re$ be continuously differentiable in some neighborhood N of Ω. Let $\{x_k\}$ be the sequence generated by the global PSG algorithm, and let the sequence of matrices $\{G_k\}$ satisfy the hypothesis in lemma 2.1. Then, either $g(x_j) = 0$ for some finite j, or the following properties hold:

(i) the sequence $\{x_k\}$ remains in Ω and every limit point is a stationary point;

(ii) no limit point of $\{x_k\}$ is a local maximum of f;

(iii) if the number of stationary points of f in Ω is finite, then the sequence $\{x_k\}$ converges.

Proof. In order to establish (i), we make use of the first part of the proof of the convergence theorem in [16, p. 709].

Let us define $m(k) = \min(k, M)$. Clearly, $m(0) = 0$ and

$$0 \leqslant m(k) \leqslant \min\big(m(k-1) + 1, M\big) \quad \text{for } k \geqslant 1.$$

Moreover, $0 < \lambda_k \leqslant \max\{\varepsilon^{-1}, \delta^{-1}\}$ for all k. Using lemma 2.1, we can obtain positive numbers c_1 and c_2 such that the search direction z_k satisfies $g_k^t z_k \leqslant -c_1 \|g_k\|_2^2$, and $\|z_k\|_2 \leqslant c_2 \|g_k\|_2$. In fact, in the global PSG algorithm, $c_1 = (\eta_{\max})^{-1}$ and $c_2 = (\eta_{\min})^{-1}$. Finally, in [16] the trial steps are all constant ($a > 0$). In our algorithm, all trial steps are in the positive closed and bounded interval $[\min\{\varepsilon, \delta^{-1}\},\ \max\{\varepsilon^{-1}, \delta^{-1}\}]$. Therefore, repeating the same arguments in [16, p. 710–711] we obtain equation (14), that in our case reduces to

$$\lim_{k\to\infty} \lambda_k \|g_k\|_2 = 0. \tag{2}$$

Let $\bar{x}$ be a limit point of $\{x_k\}$ and relabel $\{x_k\}$ a subsequence converging to $\bar{x}$. By (2) either $\lim_{k\to\infty} \|g_k\|_2 = 0$, which implies by continuity $g(\bar{x}) = 0$, or $\inf \lambda_k = 0$. If $\inf \lambda_k = 0$, there exists a subsequence $\{x_k\}_K$ such that

$$\lim_{k\in K} \lambda_k = 0.$$

In that case, from the way λ_k is chosen in step 4, there exists an index $\bar{k}$ sufficiently large such that for all $k \geqslant \bar{k},\ k \in K$, there exists $\rho_k,\ (0 < \sigma_1 \leqslant \rho_k \leqslant \sigma_2)$, for which $\lambda_k/\rho_k > 0$ fails to satisfy the nonmonotone line search, i.e.,

$$f\left(x_k + \frac{\lambda_k}{\rho_k} z_k\right) > \max_{0\leqslant j\leqslant M} f(x_{k-j}) + \gamma \frac{\lambda_k}{\rho_k} g(x_k)^t z_k \geqslant f(x_k) + \gamma \frac{\lambda_k}{\rho_k} g(x_k)^t z_k.$$

Hence,

$$\frac{f(x_k + (\lambda_k/\rho_k) z_k) - f(x_k)}{\lambda_k/\rho_k} > \gamma g(x_k)^t z_k.$$

By the mean value theorem, this relation can be written as

$$g(x_k + t_k z_k)^t z_k > \gamma g(x_k)^t z_k, \quad \text{for all } k \in K,\ k \geqslant \bar{k}, \tag{3}$$

where t_k is a scalar in the interval $[0, \lambda_k/\rho_k]$ that goes to zero as $k \in K$ goes to infinity.

Taking a convenient subsequence such that $z_k/\|z_k\|$ is convergent to $\bar{z}$, noticing that $(x_k + (\lambda_k/\rho_k) z_k)$ goes to $\bar{x}$, dividing both members in (3) by $\|z_k\|$, and taking limits we deduce that $(1 - \gamma) g(\bar{x})^t \bar{z} \geqslant 0$. Since $(1 - \gamma) > 0$ and $g(x_k)^t z_k < 0$ for all k, then $g(\bar{x})^t \bar{z} = 0$, which implies by lemma 2.1 that $g(\bar{x}) = 0$, and (i) is established.

Assertions (ii) and (iii) follow directly from the convergence theorem in [16]. □

Theorem 2.1 states that if the matrices G_k are chosen to be uniformly positive definite, then all the limit points of the sequence of iterates $\{x_k\}$ are stationary points. Moreover, if the matrices G_k are good approximations to the Hessian of f at x_k, then the number of iterations and the computational work will be reduced when compared

to the spectral gradient method without any preconditioning strategy, see [15,18]. Unfortunately, unless the function f is convex or the initial guess is close to a strict local minimizer, at some iterations x_k the Hessian matrix could be indefinite. Therefore, new features need to be added to the "basic" algorithm to allow indefinite approximations G_k of the Hessian, and still guarantee global convergence.

On the other hand, choosing a positive definite G_k, at every iteration, guarantees that the search directions are descent directions, and that implies that the global PSG algorithm cannot cycle indefinitely between steps 4 and 5 (see [26] for details). Hence, if the method is allowed to use indefinite approximations to the Hessian, it should be able to produce descent directions. Finally, for practical purposes, the preconditioning strategy should be activated only "near" local minimizers to accelerate the process (see [15]), otherwise G_k should be chosen to be the identity matrix. Indeed, since the fast convergence of Newton's method is a local property, accurate approximations to the Newton direction are only needed to accelerate the convergence of the iterations in the neighborhood of a limit point and crude improvements upon the negative gradient direction suffice elsewhere.

Summing up, the method should allow indefinite matrices G_k, it should always produce descent search directions, and should be able to detect neighborhoods of local minimizers. These three issues are the subject of our next section.

3. Robust version

In theorem 2.1 we showed that the existence of a sequence of positive definite preconditioners with uniform bounds on the maximal and minimal eigenvalues of the sequence is a sufficient condition to guarantee global convergence in the "basic" algorithm. However, uniform boundness of the eigenvalues is a very strong condition that fortunately is not a necessary condition. Roughly speaking, necessary conditions to guarantee global convergence (see [16]) are that $z_k^t g_k$ is always negative and approaches zero at the same speed of the norm of the gradient, and also that the size of the steps z_k are never "too large" when compared to the size of the gradients (see (i) and (ii) in lemma 2.1). We will accomplish both requirements by simply imposing conditions directly on the vectors z_k and not on the matrices G_k. These conditions and the corresponding control steps in the algorithm will be transparent to the user, allowing more freedom in the choice of preconditioning strategies.

Although there are some basic principles that apply to all preconditioning schemes, the user must take advantage of the specific structure of the problem at hand to find a good strategy. In a similar way, the user must also take advantage of the problem at hand to design suitable criteria to activate the preconditioner. However, our robust version of the algorithm will automatically deactivate the preconditioner, searching the negative gradient direction, if it causes deterioration of convergence. So, at this point and for the clarity of the presentation, the preconditioner is activated whenever a "local test" supplied by the user is satisfied. Later, in section 4, we will discuss specific criteria to activate the preconditioner for several problems.

Adding these control steps to the basic version, we obtain what we call the robust version of the global PSG algorithm.

Global Preconditioned Spectral Gradient algorithm ("Robust" version)**.**
Given $x_0 \in \Re^n$, an integer $M \geqslant 0$, real numbers $\gamma \in (0, 1)$, $\delta > 0$, $0 < \sigma_1 < \sigma_2 < 1$, $0 < \varepsilon < 1$, and $\alpha_0 \in [\varepsilon, 1/\varepsilon]$. Set $k = 0$, $z_0 = -g_0$, and *precond* = *OFF*.

Step 1: If $\|g_k\|$ is sufficiently small then stop.

Step 2: Set $\lambda = 1/\alpha_k$.

Step 3: (Nonmonotone line search)

$$\text{If } f(x_k + \lambda z_k) \leqslant \max_{0 \leqslant j \leqslant \min(k, M)} \{f(x_{k-j})\} + \gamma \lambda z_k^t g_k,$$

then $\lambda_k = \lambda$, $x_{k+1} = x_k + \lambda_k z_k$, and go to step 5.

Step 4: Choose $\sigma \in [\sigma_1, \sigma_2]$, set $\lambda = \sigma\lambda$, and go to step 3.

Step 5: Set $\alpha_{k+1} = -z_k^t(g_{k+1} - g_k)/\lambda_k(z_k^t g_k)$.

Step 6: If (($\alpha_{k+1} \geqslant 1/\varepsilon$) or ($\alpha_{k+1} \leqslant \varepsilon$)) then $\alpha_{k+1} = \delta$.

Step 7: (local test)

If (*precond* = *OFF*) and user local test holds, then *precond* = *ON*.

Step 8: If (*precond* = *ON*) then choose a preconditioning matrix G_{k+1}, and solve $G_{k+1}z_{k+1} = -g_{k+1}$ for z_{k+1}. Else go to step 11.

Step 9: If ($z_{k+1}^t g_{k+1} \leqslant -\varepsilon \max\{\|g_{k+1}\|^2, \|z_{k+1}\|^2\}$) then go to step 12.

Step 10: If ($z_{k+1}^t g_{k+1} \geqslant \varepsilon \max\{\|g_{k+1}\|^2, \|z_{k+1}\|^2\}$) then *precond* = *OFF*, set $z_{k+1} = -z_{k+1}$ and go to step 12.

Step 11: Set $z_{k+1} = -g_{k+1}$, and *precond* = *OFF*.

Step 12: Set $k = k + 1$, and go to step 1.

Remarks.

(i) The object of step 6 is to avoid *negative* step lengths and to keep the sequence $\{\lambda_k\}$ uniformly bounded. Indeed, for all k, $0 < \lambda_k \leqslant \max(\varepsilon^{-1}, \delta^{-1})$.

(ii) Steps 9–11 guarantee that the descent directions z_k are sufficiently steep to ensure that the global PSG algorithm cannot cycle indefinitely between steps 3 and 4 (see the proof of theorem 3.1).

(iii) We do not force the matrices G_k to be positive definite. Nevertheless, If solving the preconditioned linear system at step 8 produces an uphill direction (step 10), then we take the negative direction. In this case, the preconditioning strategy failed to produce a descent direction, which implies that the iterates are not in a convex neighborhood of the solution and we turn the preconditioner off.

(iv) If the direction z_{k+1} obtained at step 8 is *numerically* orthogonal to the vector g_{k+1}, then we take the conservative choice $z_{k+1} = -g_{k+1}$ and turn the preconditioner off.

(v) In order to save computations involved by invoking the preconditioner too early and often, we suggest that when the preconditioner is turned off by the above tests, then the strictness of the local test be increased, for example by lowering a tolerance factor adaptively in the code.

(vi) The calculation of $z_{k+1}^t g_{k+1}$ at steps 9 and 10 does not represent an additional cost since it will be used at step 3 to perform the nonmonotone line search. We need the additional calculation of the 2-norm of the vector z_{k+1} at steps 9 and 10, but only when the preconditioner is on.

The convergence properties of the global PSG algorithm, with no assumptions on the matrices G_k and no assumptions on the "user local test", are summarized in the following theorem.

Theorem 3.1. Assume that $\Omega = \{x\colon\ f(x) \leqslant f(x_0)\}$ is a bounded set. Let $f : \Re^n \to \Re$ be continuously differentiable in some neighborhood N of Ω. Let $\{x_k\}$ be the sequence generated by the global PSG algorithm ("robust version"). Then, either $g(x_j) = 0$ for some finite j, or the following properties hold:

(i) the sequence $\{x_k\}$ remains in Ω and every limit point is a stationary point;

(ii) no limit point of $\{x_k\}$ is a local maximum of f;

(iii) if the number of stationary points of f in Ω is finite, then the sequence $\{x_k\}$ converges.

Proof. Once again, to establish (i), we make use of the first part of the proof of the convergence theorem in [16, p. 709].

Let us define $m(k) = \min(k, M)$. Clearly, $m(0) = 0$ and

$$0 \leqslant m(k) \leqslant \min\bigl(m(k-1) + 1, M\bigr) \quad \text{for } k \geqslant 1.$$

Moreover, $0 < \lambda_k \leqslant \max\{\varepsilon^{-1}, \delta^{-1}\}$ for all k. Using the fact that all search directions z_k pass through the control steps 9, 10 or 11, we can obtain positive numbers c_1 and c_2 such that they satisfy $g_k^t z_k \leqslant -c_1 \|g_k\|_2^2$, and $\|z_k\|_2 \leqslant c_2 \|g_k\|_2$ for all k. In fact, whether the search direction comes out of step 9, 10 or 11,

$$z_{k+1}^t g_{k+1} \leqslant -\varepsilon \max\bigl\{\|g_{k+1}\|^2, \|z_{k+1}\|^2\bigr\}, \tag{4}$$

and so $z_k^t g_k \leqslant -\varepsilon \|g_k\|^2$ for all k, and $c_1 = \varepsilon$. On the other hand, using (4) to obtain $z_k^t g_k \leqslant -\varepsilon \|z_k\|^2$, and the Cauchy–Schwarz inequality it follows that

$$\varepsilon \|z_{k+1}\|_2^2 \leqslant \bigl|z_{k+1}^t g_{k+1}\bigr| \leqslant \|z_{k+1}\|\, \|g_{k+1}\|.$$

Since we are assuming that no finite termination will occur, then $\|z_{k+1}\| \neq 0$, and so

$$\|z_{k+1}\| \leqslant \frac{1}{\varepsilon}\|g_{k+1}\|, \tag{5}$$

and $c_2 = 1/\varepsilon$. Finally, in [16] the trial steps are all constant ($a > 0$). In our robust algorithm, all trial steps are in the positive closed and bounded interval $[\min\{\varepsilon, \delta^{-1}\}, \max\{\varepsilon^{-1}, \delta^{-1}\}]$. Therefore, repeating the same arguments in [16, p. 710–711] we obtain equation (14), that in our case reduces to:

$$\lim_{k\to\infty} \lambda_k \|g_k\|_2 = 0.$$

Let $\bar{x}$ be a limit point of $\{x_k\}$ and relabel $\{x_k\}$ a subsequence converging to $\bar{x}$. Either $\lim_{k\to\infty} \|g_k\|_2 = 0$, which implies by continuity $g(\bar{x}) = 0$, or $\inf \lambda_k = 0$. If $\inf \lambda_k = 0$, then by the same arguments used in the proof of theorem 2.1 we can build a convenient subsequence such that $z_k/\|z_k\|$ is convergent to $\bar{z}$, and $g(\bar{x})^t \bar{z} = 0$, which implies by (4) and (5) that $g(\bar{x}) = 0$, and (i) is established.

Assertions (ii) and (iii) follow directly from the convergence theorem in [16]. □

4. Numerical results

We will present numerical results on three different type of problems to illustrate the behavior of the global PSG method ("robust version"). First, we will consider standard test problems that can be found in the literature and that have been extensively used to test unconstrained optimization codes. Second, we will discuss the behavior of the method when applied to the solution of nonlinear system of equations that appear frequently in the numerical solution of nonlinear Poisson-type equations. Finally, we will apply the global PSG method to solve Multidimensional Scaling (MDS) problems. In particular, we will apply our scheme to find local minimizers of the classical stress function that plays an important role in psychometrics and also in chemistry to obtain molecular conformations from distance geometry.

We will compare the global PSG method with the global SG method (no preconditioning strategy). The SG method has been already compared favorably against the conjugate gradient (CG) method, which is the natural competitor for very large-scale problems (see [10,26]). Preconditioning variants of the CG method for nonquadratic functions have been considered and, unfortunately, so far the results have been negative (see, for example, [24,27]). As a consequence, there is no established preconditioned CG algorithm in the nonquadratic case to be considered for numerical comparisons.

For the global SG method we keep *precond* = *OFF* since it ignores the steps 7–10 in the global PSG algorithm. In all the experiments described in this section, we used $\gamma = 10^{-4}$, $\varepsilon = 10^{-10}$, $\sigma_1 = 0.1$, and $\sigma_2 = 0.5$. We use $M = 10$, and we have chosen the parameter ε to be a very small number to accept the spectral gradient step as many

times as possible. However, if the condition in step 6 was satisfied at iteration k, then the parameter δ was chosen in the following way:

$$\delta = \begin{cases} 1 & \text{if } \|g_k\|_2 > 1, \\ \|g_k\|_2^{-1} & \text{if } 10^{-5} \leqslant \|g_k\|_2 \leqslant 1, \\ 10^5 & \text{if } \|g_k\|_2 < 10^{-5}. \end{cases}$$

Notice that, with this choice of δ, the sequence $\{\lambda_k\}$ remains uniformly bounded (see remark (i)). In step 4, σ is chosen by means of a quadratic interpolation.

4.1. Standard test problems

We test our codes on eight well-known test functions. Table 1 lists the functions and the structure of their Hessians. A description of the functions and the starting points can be found in [26, and references therein].

All the experiments were run on a PC Pentium III 550 MHz clock and 128 MBytes RAM memory in double precision Fortran 90 (Fortran PowerStation 4.0). For the global PSG we used the tridiagonal part of the Hessian as a preconditioner. In this case, solving the preconditioned linear system requires O(n) flops and the storage for the matrix G_k is only 2 n-dimensional vectors. The preconditioner is activated ("local test") whenever $\|g_k\|_2 \leqslant cf$, where cf is a tolerance factor ($cf = \inf$ means that the preconditioner is activated from the first iteration). If the preconditioning strategy is deactivated ($precond = OFF$), as described in remark (iii), then we set $cf = cf \cdot 10^{-2}$. All runs were stopped when

$$\|g_k\|_2 \leqslant tol\big(1 + \big|f(x_k)\big|\big),$$

where $tol = 10^{-6}$ for most cases, except for function 3 (Oren's power) for which $tol = 10^{-5}$. In all the experiments we verified that the two methods converged to the same point.

The numerical results are shown in table 2. We report the function and the dimension of the matrix ($func(n)$), the number of iterations required for convergence ($iter$), the CPU time until convergence ($time$), the tolerance factor for the "local test" (cf), the number of line search (ls), and the iteration at which $precond$ is activated ($p - on$).

Table 1
Standard test functions.

Function	Name	Hessian
1	Brown almost linear	dense
2	Broyden tridiagonal	tridiagonal
3	Oren's power	dense
4	Penalty 1	dense
5	Extended Powell singular	pentadiagonal
6	Extended Rosenbrock	tridiagonal
7	Variably dimensioned	dense
8	Strictly convex 2	diagonal

Table 2
Results for SG and PSG on standard test functions.

		Global SG			Global PSG			
func(n)	*cf*	*iter*	*ls*	*time*	*iter*	*ls*	*time*	$p-on$
1 (1000)	1	4	0	0.01	6	0	0.1	3
1 (10000)	1	53	3	0.46	20	1	0.36	3
1 (50000)	1	57	7	3.33	16	0	1.73	4
2 (1000)	inf	40	0	0.08	16	0	0.4	1
2 (10000)	inf	93	6	1.67	16	0	0.38	1
2 (50000)	inf	132	17	12.36	16	0	2.31	1
3 (1000)	inf	264	46	0.23	45	0	0.07	1
3 (10000)	inf	992	189	7.37	85	9	1.43	+/82
3 (50000)	inf	2706	599	143.3	146	12	12.71	+/142
4 (1000)	10^{-2}	57	0	0.14	113	12	0.31	56
4 (10000)	10^{-2}	70	0	1.16	86	0	1.48	54/73
5 (1000)	inf	731	146	2.18	30	1	0.13	1
5 (10000)	inf	1656	335	47.02	30	1	1.15	1
5 (50000)	inf	1452	259	229.5	30	1	6.62	1
6 (1000)	inf	103	25	0.27	19	1	0.05	1
6 (10000)	inf	67	10	1.21	19	1	0.43	1
6 (50000)	inf	73	12	7.56	19	1	2.61	1
7 (1000)	1	54	0	0.1	56	0	0.11	53
7 (10000)	1	*	*	*	95	1	1.84	70
8 (1000)	inf	82	7	0.18	7	0	0.03	1
8 (10000)	inf	59	7	1.17	7	0	0.19	1
8 (50000)	inf	47	3	5.21	7	0	1.06	1

For both methods the number of gradient evaluations is always equal to the number of iterations. The asterisk ($*$), that appears under the multicolumns "Global SG" means that the method could not find a local solution after 10000 iterations. The plus ($+$) symbol that appears under the multicolumn "Global PSG" (function 3), means that the preconditioner was activated and deactivated more than twice, and we only report the last time it was activated. For function 4 and $n = 10\,000$, the preconditioner in the global PSG method was always deactivated inmediately after being activated, i.e., it was deactivated at iterations 55 and 74. For all other functions, once the preconditioner was activated for the first time, it was never deactivated.

We observe that the global PSG method is a very robust method to find local minimizers of nonquadratic functions. It outperforms the global SG method in number of iterations and CPU time, except for problem 4. However, even for that problem the global PSG method is competitive in number of iterations and CPU time. It is worth noticing that for problems 2, 6 and 8, the preconditioning matrix is the exact Hessian (diagonal or tridiagonal) and so the global PSG method shows superlinear convergence in the last few iterations.

For problems 2, 3, and 5, the global PSG method shows a significant reduction in number of iterations and CPU time when compared to the global SG method. For these

problems, the Hessian at the solution is either singular or extremely ill-conditioned, which is the ideal scenario to apply preconditioning techniques. It is worth mentioning that only in the singular (or ill-conditioned) case did the global SG method fail to compete with recent implementations of the conjugate gradient method (see the numerical results in [26]). We conclude that the spectral choice of steplength combined with a nonmonotone line search strategy plays an important role in the good behavior of the PSG method.

4.2. Nonlinear Poisson-type equations

We test our method on the discretized nonlinear Poisson-type equation. We follow the ideas for the discretization as presented in [1,2] for the problem

$$\frac{\partial}{\partial x}\left(k(u)\frac{\partial u}{\partial x}\right)+\frac{\partial}{\partial y}\left(k(u)\frac{\partial u}{\partial y}\right)=F(x,y) \tag{6}$$

on

$$\Omega=[0,1]\times[0,1],$$

and Dirichlet boundary conditions ($u = 0$) on the boundary of Ω. Note that if $k(u) = 1$ we recover the standard linear heat equation. We investigate the case where $k(u) = 3.33 + 0.091u$, which is an approximation to a problem of interest that occurs in certain fluid problems and was considered by Averick and Ortega [1]. A second case investigated is for $k(u) = 1+u^2$ which arises in the more general framework of Navier–Stokes equations. See, for instance, [5]. In both cases, the function $F(x, y)$ was chosen so that the exact solution is given by $u(x, y) = xy(1 - x)(1 - y)$.

Applying classical discretization techniques (see [1]) for the two-dimensional case, the numerical solution of (6) can be obtained by solving the following nonlinear system of equations

$$M(x)x-b(x)=0,$$

where

$$M(x):\Re^{m^2\times m^2}\to\Re^{m^2\times m^2}$$

is a tridiagonal block symmetric and positive definite matrix for every x, m is a positive integer such that $1/(m + 1)$ is the mesh size on Ω, and $x, b(x) \in \Re^{m^2}$. We can view $M(x)x - b(x)$ as the gradient of some function Φ (which we do not need to know explicitly) and apply our algorithms for the spectral gradient methods to find zeroes of the gradient of Φ. We apply the SG and the PSG algorithms. We do not apply globalization, although it is easy to implement, since we observed in our experiments that there is no backtracking required for convergence in this example. For instance, if we minimize the following function

$$f(x)=\left\|M(x)x-b(x)\right\|_2^2, \tag{7}$$

then the nonmonotone line search condition (step 3) is always satisfied.

Table 3
Results for SG and PSG when $\omega = 2/(1 + 2.5/m)$ and $k(u) = 1 + u^2$.

m	Method	Iterations	Disc error	CPU time
50	SG	343	1.12e−7	6.58
50	PSG	38	1.28e−7	1.05
100	SG	528	3.18e−8	42.9
100	PSG	48	3.28e−8	5.9
150	SG	1252	1.05e−7	238
150	PSG	53	1.44e−8	15.7
200	SG	1636	1.4e−6	578
200	PSG	62	8.31e−9	36.4

Table 4
Results for SG and PSG when $\omega = 2/(1 + 2.5/m)$ and $k(u) = 3.33 + 0.91u$.

m	Method	Iterations	Disc error	CPU time
50	SG	217	3.94e−7	4.19
50	PSG	38	7.18e−7	1.06
100	SG	440	5.06e−7	36.14
100	PSG	51	1.81e−7	6.3
150	SG	664	8.04e−7	127.7
150	PSG	62	8.19e−7	21.4
200	SG	1236	1.38e−6	445.1
200	PSG	81	4.37e−8	47.4

In the preconditioned versions we use $M(x)$ as an approximation to the Hessian of Φ as was done in [1], i.e., $G_k = M(x_k)$. The preconditioning strategy is activated from the beginning ($k = 0$), and it is never deactivated. To solve the preconditioned linear system at step 8, we use the well-known SSOR preconditioning strategy and choose an appropriate parameter ω (see [3] for details).

We show that the method is sensitive to the choice of ω, as is known for the linear case. The optimal ω for the nonlinear case is similar to the known linear behavior. In particular, we report the number of iterations, discretization error and CPU time in tables 3–6, for $tol = 10^{-8}$ and two different values of ω. We use $\omega = 1$ and also

$$\omega = \frac{2}{1 + 2.5/m},$$

that has been analyzed by Axelsson and Barker [4].

All the experiments were run on an HP 9000/777 workstation in *C* (gcc version 2.95.2). The starting point u_0 was chosen as $0.8 \times u^\star$, where $u^\star$ is the analytic solution of the nonlinear Poisson equation. This choice seems to be close to the solution. However, extensive numerical experimentation indicates that, for this application, the behavior of the global PSG method is not sensitive to the initial guess.

The value of α_k computed in the spectral gradient algorithm is known to be a Rayleigh quotient approximation to the eigenvalues of the Hessian (see [14]). Hence

Table 5
Effect of changing ω to 1.0 for PSG when $k(u) = 1 + u^2$.

m	Method	Iterations	Disc error	CPU time
50	PSG	122	1.22e−7	3.3
100	PSG	181	3.14e−8	21.5
150	PSG	492	4.8e−8	141.2
200	PSG	338	6.64e−8	191.2

Table 6
Effect of changing ω to 1.0 for PSG when $k(u) = 3.33 + 0.91u$.

m	Method	Iterations	Disc error	CPU time
50	PSG	112	7.06e−7	3.07
100	PSG	268	1.75e−7	32.1
150	PSG	391	1.87e−8	112.8
200	PSG	513	9.75e−8	292.1

Table 7
Estimate of the effect of preconditioning on the condition number with $\omega = 2/(1 + 2.5/m)$ and $k(u) = 1 + u^2$.

m	Precondition	Condition estimate
50	no	837
50	yes	12
100	no	4002
100	yes	25
150	no	8422
150	yes	37
200	no	13107
200	yes	50

by examining the maximum and minimum values obtained in the algorithm for α_k we can find an estimate for the condition number of the preconditioned Hessian near the solution. In particular, when using SG we are estimating the condition number of the original Hessian. We report in table 7, for the case $k(u) = 1 + u^2$ and $tol = 10^{-8}$, the estimate given by $(\max \alpha_k)/(\min \alpha_k)$. One can observe a significant reduction in the condition number estimate when the preconditioner is activated. This in turn indicates the appropriateness of our preconditioning strategy when combined with the PSG algorithm for solving the nonlinear heat equation, as our data shows (see tables 3–6).

4.3. MDS problems

We apply the global PSG method to solve Multidimensional Scaling (MDS) problems. In particular, we consider the metric MDS problem with fixed dissimilarities, that

can be formalized as the unconstrained minimization of the stress function, as follows

$$\min_{X \in \Re^{n \times p}} \frac{1}{2} \sum_{i,j} w_{ij} \big(\delta_{ij} - d_{ij}(X)\big)^2, \tag{8}$$

where w_{ij} are given nonnegative weights, δ_{ij} are the fixed dissimilarities, and $d_{ij}(X)$ are the interpoint distances of n points in $\Re^p$ represented by the $n \times p$ unknown coordinate matrix X. For additional information on MDS problems and stress, see [11,21], and [28].

Hence, we are concerned with finding local minimizers of (8) with fixed dissimilarities. We present numerical results to compare the behavior of the global PSG method with the non-preconditioned version (global SG). In all the experiments we verify that all methods converge to the same local minimum. At this point we would like to mention that our ideas can be later incorporated into a different algorithm to find the global minima of the stress function, but here we are not concerned with this issue. For a complete discussion on how to find global minima of the stress function and some related topics, see [17,19,20].

All the experiments were run on a PC Pentium III 550 MHz clock and 128 MBytes RAM memory in double precision Fortran 90 (Fortran PowerStation 4.0). To test the codes we used 3 distinct random dissimilarity matrices of dimensions 20, 50 and 100, respectively, and the dissimilarity matrix (*MOL*) of the peptide E. coli STh enterotoxin with 14 residues. This molecule has 160 atoms and so the matrix *MOL* is 160×160, (see [14]). For these matrices, we are trying to find configurations in $\Re^3$ which minimize the weighted distance to the given dissimilarity. The starting points are obtained by means of the inexpensive procedure described in [21]. For the PSG method we used the preconditioner proposed by Glunt et al. [15], which takes advantage of the specific block structure of the problem. In this case, solving the preconditioned linear systems requires $12n$ flops and the storage for the matrix G_k is only 6 n-dimensional vectors. The preconditioner is activated ("local test") whenever $\|g_k\|_2 \leqslant cf$, where cf is a tolerance factor ($cf = \inf$ means that the preconditioner is activated from the first iteration). If the preconditioning strategy fails, then we set $cf = cf \cdot 10^{-2}$. All runs were stopped whenever $\|g_k\|_2$ was less than or equal to a given tolerance (*tol*), or when the number of iterations reached 1000.

The numerical results are shown in table 8 for different values of *tol*. In particular, we report the dimension of the matrix (n), the number of iterations required for convergence (*iter*), the CPU time until convergence (*time*), the tolerance factor for the "local test" (*cf*), and the iteration at which *precond* is activated ($p - on$). For both methods the number of gradient evaluations is always equal to the number of iterations.

One can observe that for large problems the global PSG method is a very attractive option for MDS problems. In fact, the PSG method attains a solution point in all the experiments, requiring less CPU time and fewer iterations than the global SG method. Moreover, once the preconditioner is activated, the global PSG method needs half the number of iterations, on average, to obtain a more accurate solution. These remarks are consistent with the results obtained by Kearsley et al. [21] for the minimization of stress using Newton's method. In their approach, the use of full second order information to

Table 8
Results for SG and PSG on MDS problems.

			Global SG		Global PSG		
tol	*n*	*cf*	*iter*	*time*	*iter*	*time*	$p-on$
10^{-4}	20	10^{-1}	27	0.02	27	0.03	20
10^{-6}	20	10^{-1}	35	0.02	33	0.03	20
10^{-8}	20	10^{-1}	42	0.03	38	0.04	20
10^{-10}	20	10^{-1}	52	0.04	42	0.05	20
10^{-4}	50	1	49	0.13	49	0.13	–
10^{-6}	50	1	95	0.27	77	0.31	53
10^{-8}	50	1	161	0.46	101	0.41	53
10^{-10}	50	1	216	0.62	124	0.58	53
10^{-4}	100	inf	65	0.68	42	0.79	1
10^{-8}	100	10^{-1}	157	1.58	154	1.98	122
10^{-8}	100	inf	157	1.58	82	1.27	1
10^{-10}	100	10^{-1}	191	1.9	169	1.76	122
10^{-10}	100	inf	191	1.9	101	1.37	1
10^{-12}	100	10^{-1}	229	2.4	177	1.98	122
10^{-12}	100	inf	229	2.4	117	1.51	1
10^{-14}	100	inf	+1000	$*$	141	1.81	1
10^{-6}	160	inf	17	0.51	22	0.93	1
10^{-8}	160	inf	216	6.64	68	3.53	1
10^{-10}	160	inf	334	10.35	115	6.24	1
10^{-12}	160	inf	+1000	$*$	214	19.67	1

compute the search directions represents a significant amount of computational work ($O(n^3)$ flops per iteration because the Hessian of the stress function is dense). However, it also accounts for the good quality of the obtained solutions when compared with the ones obtained by the popular *majorization* gradient method, see [17,18]. In that sense, our preconditioning strategy can be viewed as a compromise between computational cost and quality of the solution.

We would like to comment that the behavior of the PSG method is sensitive to the parameter *cf*, i.e., to the iteration at which we activate the preconditioner. For our particular application and our set of random matrices the previously discussed "local test" seems to be a reasonable choice. However, in general, this is a delicate issue that deserves special attention.

5. Final remarks

Our numerical results, on three different type of problems, indicate that the PSG algorithm combines in a suitable way spectral step lengths, preconditioning techniques,

and nonmonotone globalization strategies, to produce a promising idea that accelerates the convergence of the gradient method for the minimization of nonquadratic functions.

We have discussed preconditioners that take advantage of the specific structure of the three considered type of problems. Taking advantage, whenever possible, of the structure of the Hessian is highly recommendable. However, if the Hessian has no distinctive structure then we can apply, as a default option, standard preconditioning techniques on the linear system $G_k z_k = -g_k$ (e.g., SSOR, incomplete Cholesky, etc.)

Acknowledgements

We are indebted to two anonymous referees for valuable comments and suggestions. We are also indebted to Abraham Salazar of the University of Kentucky for directing us to references [1,2,5].

References

[1] B.M. Averick and J.M. Ortega, Solution of nonlinear Poisson-type equations, Appl. Numer. Math. 8 (1991) 443–455.
[2] B.M. Averick and J.M. Ortega, Fast solution of nonlinear Poisson-type equations, SIAM J. Sci. Comput. 14 (1993) 44–48.
[3] O. Axelsson, A survey of preconditioned iterative methods for linear systems of algebraic equations, Bit 25 (1985) 166–187.
[4] O. Axelsson and V.A. Barker, *Finite Element Solution of Boundary Value Problems, Theory and Computation* (Academic Press, New York, 1984).
[5] H.E. Bailey and R.M. Beam, Newton's method applied to finite difference approximations for the steady-state compressible Navier–Stokes equations, J. Comput. Phys. 93 (1991) 108–127.
[6] J. Barzilai and J.M. Borwein, Two point step size gradient methods, IMA J. Numer. Anal. 8 (1988) 141–148.
[7] E.G. Birgin, R. Biloti, M. Tygel and L.T. Santos, Restricted optimization: A clue to fast and accurate implementation of the common reflection surface method, J. Appl. Geophys. 42 (1999) 143–155.
[8] E.G. Birgin, I. Chambouleyron and J.M. Martínez, Estimation of the optical constants and the thickness of thin films using unconstrained optimization, J. Comput. Phys. 151 (1999) 862–880.
[9] E.G. Birgin and Y.G. Evtushenko, Automatic differentiation and spectral projected gradient methods for optimal control problems, Optim. Methods Software 10 (1998) 125–146.
[10] E.G. Birgin and J.M. Martínez, A spectral conjugate gradient method for unconstrained optimization, Appl. Math. Optim. 43 (2001) 117–128.
[11] I. Borg and P.J.F. Groenen, *Modern Multidimensional Scaling: Theory and Applications* (Springer, Berlin, 1997).
[12] Z. Castillo, D. Cores and M. Raydan, Low cost optimization techniques for solving the nonlinear seismic reflection tomography problem, Optim. Engrg. 1 (2000) 155–169.
[13] D. Cores, G. Fung and R. Michelena, A fast and global two point low storage optimization technique for tracing rays in 2D and 3D isotropic media, J. Appl. Geophys. 45 (2000) 273–287.
[14] W. Glunt, T.L. Hayden and M. Raydan, Molecular conformations from distance matrices, J. Comput. Chem. 14 (1993) 114–120.
[15] W. Glunt, T.L. Hayden and M. Raydan, Preconditioners for distance matrix algorithms, J. Comput. Chem. 15 (1994) 227–232.

[16] L. Grippo, F. Lampariello and S. Lucidi, A nonmonotone line search technique for Newton's method, SIAM J. Numer. Anal. 23 (19) 707–716, 1986.
[17] P.J.F. Groenen, *The Majorization Approach to Multidimensional Scaling: Some Problems and Extensions* (DSWO Press, Univ. of Leiden, 1993).
[18] P.J.F. Groenen, W. Glunt and T.L. Hayden, Fast algorithms for multidimensional scaling: A comparison of majorization and spectral gradient methods, Technical Report, Data Theory Department, University of Leiden, Leiden, Netherlands (1996).
[19] P.J.F. Groenen and W.J. Heiser, The tunneling method for global optimization in multidimensional scaling, Psychometrika 61 (1996) 529–550.
[20] P.J.F. Groenen, W.J. Heiser and J.J. Meulman, Global optimization in least-squares multidimensional scaling by distance smoothing, J. Classification 16 (1999) 225–254.
[21] A.J. Kearsley, R.A. Tapia and M.W. Trosset, The solution of the metric stress and stress problems in multidimensional scaling using Newton's method, Comput. Statist. 13(3) (1998) 369–396.
[22] B. Molina and M. Raydan, The preconditioned Barzilai–Borwein method for the numerical solution of partial differential equations, Numer. Algorithms 13 (1996) 45–60.
[23] M. Mulato, I. Chambouleyron, E.G. Birgin and J.M. Martínez, Determination of thickness and optical constants of a-Si:H films from transmittance data, Appl. Phys. Lett. 77 (2000) 2133–2135.
[24] J. Nocedal, Conjugate gradient methods and nonlinear optimization, in: *Linear and Nonlinear Conjugate Gradient Related Methods*, eds. L. Adams and J.L. Nazareth, *AMS-IMS-SIAM Joint Summer Research Conference* (SIAM, Philadelphia, PA, 1996) pp. 9–23.
[25] M. Raydan, On the Barzilai and Borwein choice of steplength for the gradient method, IMA J. Numer. Anal. 13 (1993) 321–326.
[26] M. Raydan, The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem, SIAM J. Optim. 7 (1997) 26–33.
[27] D.F. Shanno, On preconditioned conjugate gradient methods, in: *Nonlinear Programming*, Vol. 4, eds. R.R. Meyer, O.L. Mangasarian and S.M. Robinson (Academic Press, New York, 1981) pp. 201–221.
[28] M.W. Trosset, Numerical algorithms for MDS, in: *Classification and Knowledge Organization*, eds. R. Klar and P. Opitz (Springer, Berlin, 1997) pp. 80–92.
[29] C. Wells, W. Glunt and T.L. Hayden, Searching conformational space with the spectral distance geometry algorithm, J. Molecular Structure (Theochem) 308 (1994) 263–271.