Pré-Publicações do Departamento de Matemática
Universidade de Coimbra
Preprint Number 19–47

MODELING HESSIAN-VECTOR PRODUCTS IN NONLINEAR OPTIMIZATION: NEW HESSIAN-FREE METHODS

L. SONG AND L. N. VICENTE

Abstract: In this paper, we suggest two ways of calculating interpolation models for unconstrained smooth nonlinear optimization when Hessian-vector products are available. The main idea is to interpolate the objective function using a quadratic on a set of points around the current one and concurrently using the curvature information from products of the Hessian times appropriate vectors, possibly defined by the interpolating points. These enriched interpolating conditions form then an affine space of model Hessians or model Newton directions, from which a particular one can be computed once an equilibrium or least secant principle is defined.

A first approach consists of recovering the Hessian matrix satisfying the enriched interpolating conditions, from which then a Newton direction model can be computed. In a second approach we pose the recovery problem directly in the Newton direction. These techniques can lead to a significant reduction in the overall number of Hessian-vector products when compared to the inexact or truncated Newton method, although simple implementations may pay a cost in linear algebra or number of function evaluations.

Keywords: Nonlinear/Nonconvex Optimization, Hessian-Vector Products, Quadratic Interpolation, Newton Direction, Hessian Recovery.

1. Introduction

Let us consider the minimization of a twice continuously differentiable function f,

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

in a context where the following information is available: Given $x \in \mathbb{R}^n$, one can compute $f(x)$, $\nabla f(x)$, and $\nabla^2 f(x) v$ for any vector $v \in \mathbb{R}^n$.

Received December 30, 2019.

Support for LS was partially provided by FCT/Portugal under grants PD/BI/128105/2016 and UID/MAT/00324/2019 (CMUC).

Support for LNV was partially provided by FCT/Portugal under grants UID/MAT/00324/2019 (CMUC) and P2020 SAICTPAC/0011/2015.

1.1. Literature review. Newton-based methods for unconstrained nonlinear optimization require the solution of a linear system at each iteration. The matrix of this system is the Hessian of f and its right-hand side is the negative of the gradient. There are instances where the Hessian is not available for factorization or where that is too expensive but where one can afford to do Hessian-vector products, in which cases the system should not be solved directly but by an iterative method. Then it is known that there is a residual error in the application of the iterative solver and that such a residual can be made smaller by asking more from the solver. This reasoning gave rise to the so-called inexact or truncated Newton methods which have formed an important numerical tool for many decades. It is well known since the contribution [6] what conditions one should impose on the norm of the residual of the linear system to obtain linear, superlinear, or quadratic local convergence in the iterates of the underlying method (see [15]). Global convergence of inexact Newton methods is also well studied [8, 13]. One knows well also how to deal with negative curvature while solving the linear system using Krylov-type methods (Conjugate Gradients or Lanczos), either using a trust-region technique [10, 18] or a line search [14].

When Hessian or Hessian-vector products are not available, estimating the Hessian can then play an important role, however the existing approaches are not entirely satisfactory. If the Hessian matrix is sparse and its sparsity pattern is known, the approach in [9] enforces multiple secant equations in a least squares sense, solving then a positive semi-definite system of equations in the nonzero Hessian components. Their approach does not show a significant improvement compared to the L-LBGS or Newton trust-region methods. In [17] the Hessian is estimated by finite differences in the gradient, but by dividing the Hessian columns first into groups. Using symmetry and the known sparsity of the Hessian, it is possible to find approximations to different Hessian columns at once. This method is cheap in computer arithmetic and provided better results when compared to [5]. A more recent approach [2] imposes the secant equations componentwise, leading to fewer equations when taking into account the available sparsity pattern. The numerical results show that the algorithm can find the Hessian approximation fast and accurately when the number of nonzero entries per row is relatively low.

1.2. The contribution of the paper. In this paper two techniques are proposed and analyzed for the Hessian-free scenario where only Hessian-vector products are available for use. Our goal is to use as few of these products as possible without losing the ability to converge to a solution or a stationary point of the original problem. Having this in mind we form a quadratic model around a point x, using function and gradient values at x and function values at the interpolating points y^ℓ, $\ell = 1, \ldots, p$. The matrix H of this model or some kind of Newton step has then to be recovered.

Our first approach enriches these interpolating conditions with the information coming from a single true Hessian-vector product $\nabla^2 f(x)(y - x)$, for a point y different from any of the y^ℓ's of those conditions. In fact, to avoid degeneracy in the enriched interpolating conditions (which are affine conditions on H), one has to choose y differently from those y^ℓ's and one cannot consider more than one of these products. The computation of the model Hessian is carried out by minimizing its norm or its distance to a previous model Hessian (say from a previous iteration of the optimization method) subject to the enriched interpolating conditions. Such a Hessian recovery can then lead to the computation of an approximate Newton step.

Our second approach allows us to consider more than one Hessian-vector product in the model formulation. The interpolating conditions are now enriched by the second-order information coming from the Hessian-vector products $\nabla^2 f(x)(y^\ell - x)$, $\ell = 1, \ldots, p$. Then, avoiding degeneracy and the inverse of the Hessian model, the recovery is done in the space of the Newton direction models, using a modified set of enriched interpolating conditions. Again, the computation of the Newton direction model is carried out by minimizing its norm or its distance to a previous Newton direction model subject to the modified enriched interpolating conditions.

In both cases we will provide some theoretical support for the recoveries by proving that the absolute error (in model Hessian or in model Newton direction) is decreasing in the case where f is quadratic and the enriched interpolating conditions are underdetermined. The recovery absolute error coming from the enriched interpolating conditions (in a determined situation) will be also analyzed for both cases. We report numerical results to confirm that both approaches are sound and can lead to a significant reduction in the number of Hessian-vector products. The dimension of the problems tested is rather small. The linear algebra is dense, and the number of functions evaluations used can be relatively high. It is left for future research the application to medium/large-scale problems. The second approach based on a Newton direction model can be easily parallelized (see Section 4).

The paper is organized as follows. In Section 2 we present our first approach, the one for the recovery of a model Hessian. In Section 3 we describe our second approach, the one for the recovery of a model Newton direction. In both cases we report illustrative numerical results for small problems. The paper is finished in Section 4 with some final remarks and prospects of future work. The notation $\mathcal{O}(A)$ will be used to represent the product of a constant times A whenever the multiplicative is independent of A. All vector and matrix norms are Euclidian unless otherwise specified.

2. Hessian recovery from Hessian-vector products

Let x be a given point. Suppose also that we have calculated f and ∇f at x as well as f at a number of points $y^1, \ldots, y^p$. We can then use quadratic interpolation to fit the data by determining a symmetric matrix H such that

$$f(x) + \nabla f(x)^\top (y^\ell - x) + \frac{1}{2}(y^\ell - x)^\top H (y^\ell - x) \;=\; f(y^\ell), \quad \ell = 1, \ldots, p. \tag{1}$$

Furthermore, given a set of vectors $v^1, \ldots, v^m$, with m possibly much smaller than n, suppose that we have calculated $w^j = \nabla^2 f(x) v^j$, $j = 1, \ldots, q$. Hence we could then ask our Hessian model H to satisfy $Hv^j = w^j$, $j = 1, \ldots, q$. However it is important to notice two immediate facts, reported in Remark 2.1.

Remark 2.1. *First we cannot have $q > 1$. Any use of a pair v^1, v^2 would make the conditions $Hv^1 = w^1$ and $Hv^2 = w^2$ degenerate in H, in the sense that the matrix multiplying the component variables of H would be rank deficient. This fact can be easily confirmed from multiplying each by the other vector, i.e., by looking at $(v^2)^\top H v^1 = (v^2)^\top w^1$ and $(v^1)^\top H v^2 = (v^1)^\top w^2$. Secondly, even when taking $q = 1$, one cannot consider $v^1 = y^\ell - x$, for any ℓ, for the exact same reason. In fact, multiplying $H(y^\ell - x) = w^1$ on the left by $(1/2)(y^\ell - x)^\top$ would lead us to the same term in H as of the corresponding interpolating one in (1).*

2.1. Hessian recovery. From Remark 2.1, we know that we can only consider one vector v for the Hessian multiplication $w = \nabla^2 f(x)\, v$, and that this vector cannot be any of the interpolation vectors $y^\ell - x$. Then, in the same vein as it was done in [4] for derivative-free optimization, a model Hessian H

could then be calculated from the solution of the recovery problem

$$\min_{H} \quad \mathrm{norm}(H) \quad \text{s.t.} \quad (1) \ \text{ and } \ Hv = w. \tag{2}$$

The norm(H) could be taken in a certain ℓ_1 sense, leading to a linear program (see [1]). It could also be set as the Frobenius norm, $\mathrm{norm}(H) = \|H\|_F$, leading to a quadratic program. Alternatively, one can recover a model Hessian in a least secant fashion (as done in [16] for derivative-free optimization using the Frobenius norm)

$$\min_{H} \quad \mathrm{norm}(H - H^{prev}) \quad \text{s.t.} \quad (1) \ \text{ and } \ Hv = w, \tag{3}$$

where H^{prev} is a previously computed model Hessian (say, from a previous iteration of an optimization scheme).

2.2. Theoretical motivation. We will now see that when f is quadratic the error in the difference between the optimal solution H^* of (3) and the true Hessian decreases relatively to the previous estimate H^{prev}. To prove such a result it is convenient to use the Frobenius norm in (3) and consider:

$$\min_{H} \quad \frac{1}{2}\|H - H^{prev}\|_F^2 \quad \text{s.t.} \quad (1) \ \text{ and } \ Hv = w, \tag{4}$$

Let us first write the quadratic f centered at x

$$f(y) \ = \ a + b^\top(y-x) + \frac{1}{2}(y-x)^\top C(y-x), \tag{5}$$

where $a = f(x)$, $b = \nabla f(x)$, and C is a symmetric matrix.

Theorem 2.1. *Let f be given by (5) and assume that the system of linear equations defined by (1) and $Hv = w$ is feasible and underdetermined in H. Let H^* be the optimal solution of problem (4). Then*

$$\|H^* - C\|_F^2 \ \leq \ \|H^{prev} - C\|_F^2.$$

Proof: The proof follows the argument in [16]. From (1), we have $(y^\ell - x)^\top(C - H^*)(y^\ell - x) = 0$, $\ell = 1, \ldots, p$. We also have $(C - H^*)v = 0$. Hence, $C - H^*$ is a feasible direction for the affine space in H defined by (1) and $Hv = w$. It then turns out that the function

$$m(\theta) \ = \ \frac{1}{2}\|(H^* - H^{prev}) + \theta(C - H^*)\|_F^2$$

has a minimum at $\theta = 0$. From the trace definition of the Frobenius norm

$$m'(\theta) \ = \ \left[(H^* - H^{prev}) + \theta(C - H^*)\right]^\top (C - H^*).$$

Hence,

$$(H^* - H^{prev})^\top (C - H^*) \;=\; 0,$$

which then implies (given the symmetry of the matrices and considering only the diagonal entries of the above matrix product)

$$\sum_{i=1}^{n}\sum_{j=1}^{n}(H^*_{ij} - H^{prev}_{ij})(C_{ij} - H^*_{ij}) \;=\; 0.$$

The rest of the proof requires the following calculations:

$$\begin{aligned}
&\|H^{prev} - C\|_F^2 - \|H^* - H^{prev}\|_F^2 - \|H^* - C\|_F^2\\
&= \sum_{i=1}^{n}\sum_{j=1}^{n}[(H^{prev}_{ij} - C_{ij})^2 - (H^*_{ij} - H^{prev}_{ij})^2 - (H^*_{ij} - C_{ij})^2]\\
&= \sum_{i=1}^{n}\sum_{j=1}^{n}[(H^{prev}_{ij} - C_{ij} + H^*_{ij} - H^{prev}_{ij})(H^{prev}_{ij} - C_{ij} - H^*_{ij} + H^{prev}_{ij}) - (H^*_{ij} - C_{ij})^2]\\
&= \sum_{i=1}^{n}\sum_{j=1}^{n}[(H^*_{ij} - C_{ij})(2H^{prev}_{ij} - C_{ij} - H^*_{ij} - H^*_{ij} + C_{ij})]\\
&= 2\sum_{i=1}^{n}\sum_{j=1}^{n}[(H^*_{ij} - C_{ij})(H^{prev}_{ij} - H^*_{ij})] \;=\; 0.
\end{aligned}$$

Hence we have established that

$$\begin{aligned}
\|H^* - C\|_F^2 &= \|H^{prev} - C\|_F^2 - \|H^* - H^{prev}\|_F^2\\
&\le \|H^{prev} - C\|_F^2.
\end{aligned}$$

■

Let α represent the coefficients of H in $(1/2)w^\top Hw$ in terms of the monomial basis. The quadratic components of this basis are of the form $(1/2)w_i^2$, $i = 1, \ldots, n$ and w_iw_j, $1 \le i < j \le n$. So, we have

$$(1/2)h_{11}w_1^2 = \alpha_1[(1/2)w_1^2], \ldots, (1/2)h_{nn}w_n^2 = \alpha_n[(1/2)w_n^2],$$

$$h_{12}w_1w_2 = \alpha_{n+1}[w_1w_2]$$

and so on. The recovery problem (4) can then be formulated approximately[1] as

$$\min_{\alpha} \quad \frac{1}{2}\|\alpha - \alpha^{prev}\|^2 \quad \text{s.t.} \quad M\alpha \;=\; \delta, \tag{6}$$

where

$$M = \begin{bmatrix} M^1 \\ M^2 \end{bmatrix}, \quad \delta = \begin{bmatrix} \delta^1 \\ \delta^2 \end{bmatrix},$$

$$M^1\alpha = \begin{bmatrix} \frac{1}{2}(y^1 - x)^\top H(y^1 - x) \\ \vdots \\ \frac{1}{2}(y^p - x)^\top H(y^p - x) \end{bmatrix}, \quad M^2\alpha = Hv,$$

$$\delta^1 = \begin{bmatrix} f(y^1) - f(x) - \nabla f(x)^\top (y^1 - x) \\ \vdots \\ f(y^p) - f(x) - \nabla f(x)^\top (y^p - x) \end{bmatrix}, \quad \delta^2 = w.$$

Another piece of motivation for this approach comes from the fact that the enriched interpolating conditions defined by (1) and $Hv = w$, once determined (i.e. with as many equations as variables), may produce a model Hessian H that used together with $\nabla f(x)$ can give rise to a fully quadratic model. Such a model has the same orders of accuracy as a Taylor-based model [3] (see also [4]).

Theorem 2.2. *If p is chosen such that $p+n = \frac{n^2+n}{2}$ and if M is nonsingular, then the model Hessian H resulting from $M\alpha = \delta$ in (6) can give rise to a fully quadratic model, in other words, one has*

$$\|H - \nabla^2 f(x)\| \;=\; \mathcal{O}(\Delta_y),$$

where $\Delta_y = \max_{1 \le \ell \le p} \left\| y^\ell - x \right\|$ and the constant multiplying Δ_y depends on the inverse of an appropriate scaled version of M.

Proof: First we follow the argument in [3][Theorem 4.2] and consider that x is at the origin, without any lost of generality. One can start by making a Taylor expansion of f around x along all the displacements $y^\ell - x$, $\ell = 1, \ldots, p$, leading to

$$M^1(\alpha - \alpha^x) \;=\; \mathcal{O}(\Delta_y^3), \tag{7}$$

[1]The norm used in (6) for α is a minor variation of the Frobenius norm of H.

where α^x stores the components of $\nabla^2 f(x)$ and each component of the right-hand side is bounded by $(1/6)L_{\nabla^2 f}\|y^\ell - x\|^3$, with $L_{\nabla^2 f}$ the Lipschitz constant of $\nabla^2 f$. One also has

$$M^2(\alpha - \alpha^x) \; = \; 0.$$

Now we divide each row of (7) by Δ_y^2. The proof is concluded by considering $[M^1/\Delta_y^2; M^2]$ as the scaled version of M alluded in the statement of the result. ■

2.3. Numerical results for the determined case. As we have discussed in Theorem 2.2, if p is chosen such that $p + n = \frac{n^2+n}{2}$ and if the matrix M is nonsingular and well conditioned, the model Hessian H resulting from $M\alpha = \delta$ in (6) becomes fully quadratic. The error between the Hessian model H and $\nabla^2 f(x)$ is then of the $\mathcal{O}(\Delta_y)$, where

$$\Delta_y = \max_{1 \leq \ell \leq p} \|y^\ell - x\|.$$

In this section we will report some illustrative numerical results to confirm that an approach built on such an Hessian model can lead to an economy of Hessian-vector products. Our term of comparison will be the inexact Newton method (as described in [15, Section 7.1]), where the system

$$\nabla^2 f(x) d^{IN} = -\nabla f(x)$$

is solved by applying a truncated linear conjugate (CG) method (stopping once a direction of negative curvature is found or a relative error criterion is met). In our case, after computing H from solving $M\alpha = \delta$ in (6), to compute our search direction d^{MH}, we apply the exact same truncated CG method to

$$H d^{MH} = -\nabla f(x)$$

as in the inexact Newton method. The computed directions d^{IN} or d^{MH} are necessarily descent in the sense of making an acute angle with $-\nabla f(x)$.

For both the inexact Newton method and our model Hessian approach, a new iterate is of the form $x + \alpha d$, where d is given by d^{IN} or d^{MH} respectively. The same cubic interpolation line search [19, Section 2.4.2] is used to compute the stepsizes α^{IN} and α^{MH}. In this line search, the objective function is approximated by a cubic polynomial with function values at three points and a derivative value at one point. The line search starts with a unit stepsize and terminates either successfully with a value α satisfying a sufficient decrease condition for the function (of the form $f(x+\alpha d) \leq f(x)+c_1\alpha\nabla f(x)^\top d$, with $c_1 = 10^{-4}$) or unsuccessfully with a stepsize smaller than 10^{-10}.

To form the model described in (2) one needs p interpolation points $y^1, \cdots, y^p$ and one vector v for Hessian multiplication. We have used the following scheme: Before the initial iteration we have randomly generated a set of p points, $\{y^1, \cdots, y^p\}$, and a vector v, in the unit ball $B(0;1)$ centered at the origin. Then, at each iteration x_k, the interpolation points used were of the form $x_k + r_k y_k^\ell$, $\ell = 1, \ldots, p$, and the vector v_k of the form $r_k\, v$, where

$$r_k = \min\{10^{-2}, \max\{10^{-4}, \|x_k - x_{k-1}\|\}\},\ k = 1, 2, \ldots.$$

For the purpose of this numerical illustration, we selected 48 unconstrained (smooth and nonlinear) very small problems from the CUTEst collection (see Appendix B), also used in the papers [2, 12]. Both methods were stopped when an iterate x_k was found such that $\|\nabla f(x_k)\| < 10^{-5}$. We built performance profiles (see Appendix A) using as performance metric the numbers of Hessian-vector products and iterations (Figure 1) and the number of function evaluations (Figure 2). One can see that our approach can effectively lead to a significant reduction on the number of Hessian-vector products. We estimate that this reduction is approximately 50% as both approaches take on average 2 CG inner iterations to compute a direction, and the number of main iterations is comparable. Of course, one has to pay a significant cost in number of function evaluations which is of the order of n^2 per main iteration.

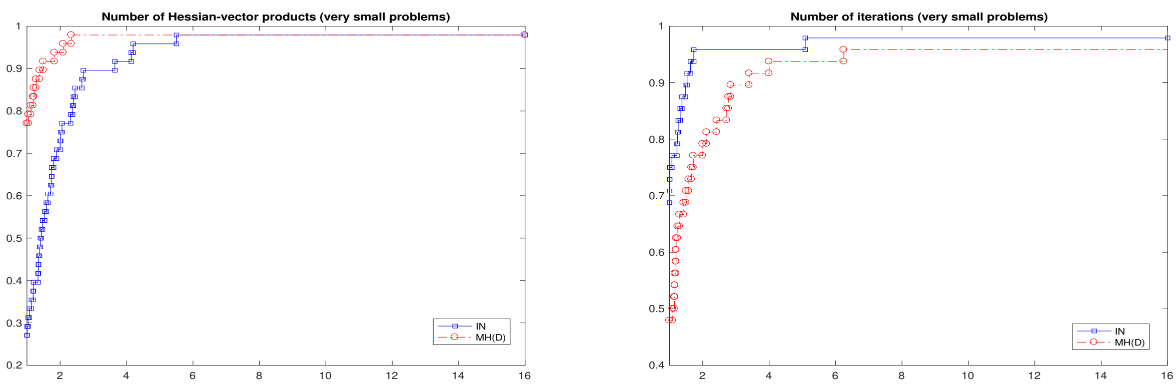

FIGURE 1. Testing the Hessian recovery within a line-search algorithm. Performance profiles for the numbers of Hessian-vector products and iterations, for the set of very small problems of Appendix B. The value of p was set to $\frac{n^2+n}{2} - n$.

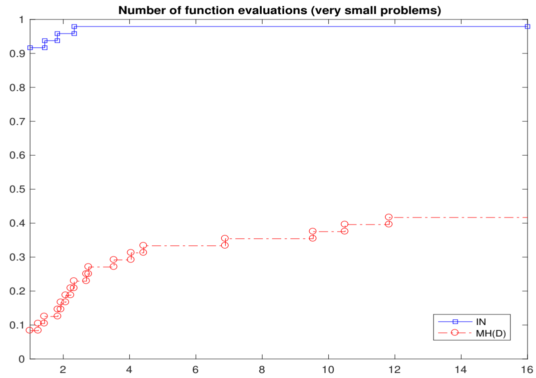

FIGURE 2. Testing the Hessian recovery within a line-search algorithm. Performance profiles for number of function evaluations, for the set of very small problems of Appendix B. The value of p was set to $\frac{n^2+n}{2} - n$.

2.4. Numerical results for the determined case when the Hessian sparsity is known. In many optimization problems, the Hessian matrix of the objective function is sparse and the corresponding sparsity pattern is known. Let

$$\Omega(\nabla^2 f) = \{(i,j) : i \le j, \nabla^2 f_{ij}(x) = 0 \text{ for all } x\}$$

be the sparsity pattern of $\nabla^2 f$. When $|\Omega(\nabla^2 f)| \ll n(n+1)/2$, it is then beneficial and often necessary to use specialized algorithms and data structures that take advantage of the known sparsity pattern. One can tailor our model Hessian approach to problems with sparse Hessian matrices when the sparsity patterns are known. We require the Hessian model to share the same sparsity pattern of the true Hessian, recovering only the nonzero elements. In fact, instead of solving problem (6) with respect to the whole Hessian matrix, we solve problem

$$\min_{\alpha_\Omega} \quad \frac{1}{2}\|\alpha_\Omega - \alpha_\Omega^{prev}\|^2 \quad \text{s.t.} \quad M_\Omega \alpha_\Omega \;=\; \delta, \tag{8}$$

where the elements in the rows of M_Ω and in the vector α_Ω correspond now only to nonzero entries.

We have tested our sparse Hessian recovery approach using the same algorithmic environment of Subsection 2.3, the only difference being in the usage of the model equation $M_\Omega \alpha_\Omega = \delta$ in (8) and a smaller value of p (now given by the difference between the number of nonzeros of the Hessian and n, so that the matrix M_Ω is squared). The sparse problems used are listed in Appendix C. The experiments are reported in Figures 3 and 4 in the form of performance profiles. The conclusions are similar to those in Subsection 2.3.

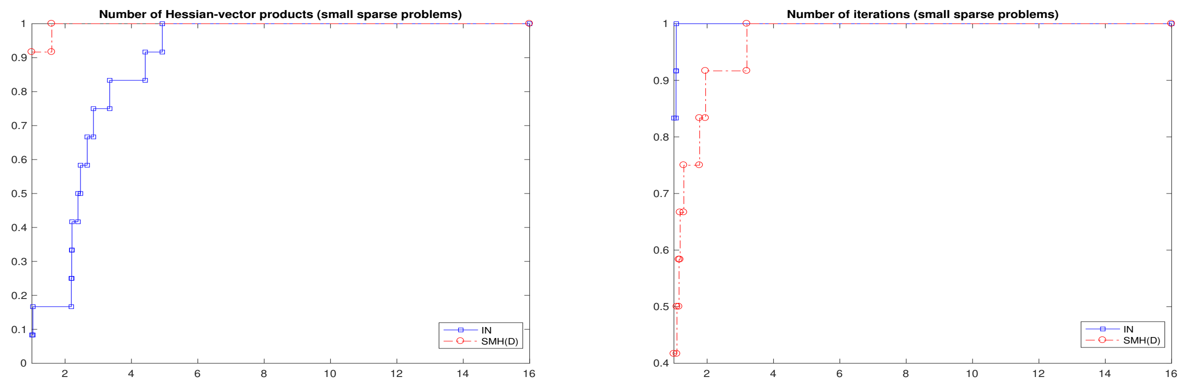

FIGURE 3. Testing the Hessian recovery within a line-search algorithm. Performance profiles for the numbers of Hessian-vector products and iterations, for the set of small sparse problems of Appendix C. The value of p was set to number of nonzeros minus n.

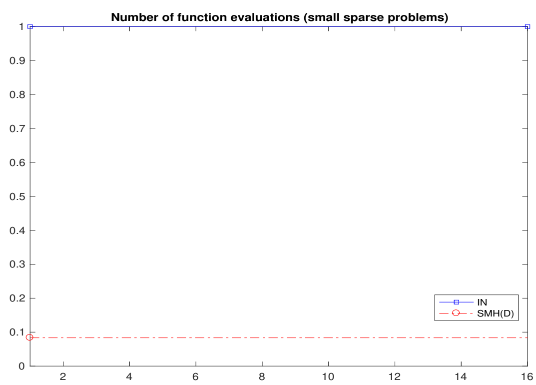

FIGURE 4. Testing the Hessian recovery within a line-search algorithm. Performance profiles for number of function evaluations, for the set of small sparse problems of Appendix C. The value of p was set to number of nonzeros minus n.

2.5. Recovery cost in the general case. The necessary and sufficient optimality conditions for the convex QP (6) can be stated as

$$\begin{aligned} \alpha - \alpha^{prev} - M^\top \lambda = 0 \\ M\alpha = \delta, \end{aligned} \tag{9}$$

where λ denotes the Lagrange multipliers. Such multipliers can then be recovered by solving

$$MM^\top \lambda \;=\; \delta - M\alpha^{prev}. \tag{10}$$

The system (10) can either be solved directly or iteratively. If solved directly the cost is of the order of $(p+n)^2n^2$ to form $MM^\top$ and of $(p+n)^3$ to factorize it, and the overall storage of the order of $(p+n)^2$. If the Conjugate Gradient (CG) method is applied, the overall cost is of the order of $c_g(p+n)n^2$, where c_g is the number of CG iterations. In fact, each matrix vector multiplication with either $M^\top$ or M costs $\mathcal{O}((p+n)n^2)$. Solving the KKT system (9) using an indefinite factorization is even less viable given that the storage space would be of the order of $(n^2+p)^2$.

3. Newton direction recovery from Hessian-vector products

In this section, we introduce a new approach to recover the Newton direction from Hessian-vector products that does not require an explicit recovery of the Hessian matrix.

3.1. Newton direction recovery. Let us first consider a quadratic Taylor expansion of the form

$$f(x) + \nabla f(x)^\top (y^\ell - x) + \frac{1}{2}(y^\ell - x)^\top \nabla^2 f(x)(y^\ell - x) \;\simeq\; f(y^\ell), \quad \ell = 1, \ldots, p, \tag{11}$$

made using a sample set $\{y^1, \ldots, y^p\}$. We will synchronize expansion (11) with Hessian-vector products along $y^\ell - x$, $\ell = 1, \ldots, p$. In fact, we require the calculation of

$$z^\ell \;=\; \nabla^2 f(x)(y^\ell - x), \quad \ell = 1, \ldots, p. \tag{12}$$

Since our interest relies specifically on the calculation of the Newton direction, assuming that the model Hessian $\nabla^2 f(x)$ is nonsingular, we obtain from (11) and (12)

$$f(x)+(\nabla^2 f(x)^{-1}\nabla f(x))^\top \nabla^2 f(x)(y^\ell - x)+\frac{1}{2}(y^\ell - x)^\top z^\ell \;\simeq\; f(y^\ell), \quad \ell = 1,\ldots,p.$$

Then, introducing the model vector $d \simeq -\nabla^2 f(x)^{-1}\nabla f(x)$, one arrives at a new set of enriched interpolating conditions

$$(z^\ell)^\top d \;=\; -f(y^\ell) + f(x) + \frac{1}{2}(y^\ell - x)^\top z^\ell, \quad \ell = 1,\ldots,p. \tag{13}$$

Equations (13) lead then to a new recovery problem

$$\min_{d} \quad \mathrm{norm}(d - d^{prev}) \quad \text{s.t.} \quad (13). \tag{14}$$

When d^{prev} is the previously recovered Newton direction, we are following the spirit of a quasi-Newton least secant approach. One could also consider the case $d^{prev} = 0$ as it was done in some derivative-free approaches for Hessian recovery. Let us now give two arguments to motivate this approach.

3.2. Theoretical motivation. First, as in the previous section, we can provide motivation for this approach when f is assumed quadratic (5), this time with a nonsingular Hessian C. Here we need to consider the square of the ℓ_2-norm in (14)

$$\min_{d} \quad \frac{1}{2}\|d - d^{prev}\|^2 \quad \text{s.t.} \quad (13). \tag{15}$$

We will show that in the quadratic case the error in the approximation of the Newton direction is monotonically non increasing.

Theorem 3.1. *Let f be given by (5) with C nonsingular and assume that the system of linear equations (13) is feasible and underdetermined in d. Let d^* be the optimal solution of problem (15). Then*

$$\|d^* - (-C^{-1}b)\|^2 \;\leq\; \|d^{prev} - (-C^{-1}b)\|^2. \tag{16}$$

Proof: From the expression (5) for f, one has

$$f(y^\ell) \;=\; a + (C^{-1}b)^\top C(y^\ell - x) + \frac{1}{2}(y^\ell - x)^\top C(y^\ell - x), \quad \ell = 1,\ldots,p.$$

and hence, using $z^\ell = C(y^\ell - x)$, $\ell = 1,\ldots,p$, and (13), one arrives at $(z^\ell)^\top(d^* - (-C^{-1}b)) = 0$. The conclusion is that $d^* - (-C^{-1}b)$ is a feasible direction for the affine space in d defined by (13).

The rest of the proof follows the same lines as in the proof of Theorem 2.1. The function

$$m(\theta) \;=\; \frac{1}{2}\|(d^* - d^{prev}) + \theta(-C^{-1}b - d^*)\|^2$$

has a minimum at $\theta = 0$, from which we conclude that $(d^* - d^{prev})^\top(-C^{-1}b - d^*) = 0$. From here we obtain

$$\begin{aligned} \|d^* - (-C^{-1}b)\|^2 &= \|d^{prev} - (-C^{-1}b)\|^2 - \|d^* - d^{prev}\|^2 \\ &\le \|d^{prev} - (-C^{-1}b)\|^2. \end{aligned}$$

■

The second argument establishes the accuracy of the recovery under the assumption that $p \ge n$ (see the end of this subsection for a discussion about this assumption and how to circumvent it practice). We will establish a bound on the norm of the absolute error of the recovered Newton direction d^N based on $\Delta_y = \max_{1\le\ell\le p}\|y^\ell - x\|$, $\Delta_z = \max_{1\le\ell\le p}\|z^\ell\|$, and the conditioning of the matrix M_L^z, whose rows are $(1/\Delta_z)(z^\ell)^\top$, $\ell = 1,\ldots,p$.

Theorem 3.2. *Suppose that $p \ge n$, the matrix M_L^z is full column rank, and $\nabla^2 f(x)$ is invertible. Then, if d^N satisfies (13), in a least squares sense when $p > n$, one has*

$$\left\|-\nabla^2 f(x)^{-1}\nabla f(x) - d^N\right\| \;\le\; \Lambda_z \mathcal{O}\left(\frac{\Delta_y^3}{\Delta_z}\right),$$

where Λ_z is a bound on the norm of the left inverse of M_L^z and the multiplicative constant in $\mathcal{O}$ depends on the Lipschitz constant of $\nabla^2 f$.

Proof: Expanding f at y^ℓ around x in (13) yields

$$(-\nabla^2 f(x)^{-1}\nabla f(x) - d^N)^\top z^\ell \;=\; \mathcal{O}(\Delta_y^3), \quad \ell = 1,\ldots,p,$$

where the constant in $\mathcal{O}(\Delta_y^3)$ depends on the Lipschitz constant of $\nabla^2 f$. The result follows by dividing both terms by Δ_z and multiplying by the left inverse of M_L^z. ■

One can derive an estimate solely dependent on Δ_y and on the conditioning of the matrix M_L^y formed by the rows $(1/\Delta_y)(y^\ell - x)^\top$, $\ell = 1,\ldots,p$. In fact, from

$$\Delta_y M_L^y \nabla^2 f(x) \;=\; \Delta_z M_L^z$$

one has

$$\left\|(M_L^z)^\dagger\right\| \;=\; \frac{\Delta_z}{\Delta_y}\|R_y\|,$$

with

$$R_y \;=\; \left(\nabla^2 f(x)(M_L^y)^\top(M_L^y)\nabla^2 f(x)\right)^{-1}\nabla^2 f(x)(M_L^y)^\top. \tag{17}$$

Corollary 3.1. *Suppose that $p \geq n$, the matrix M_L^z is full column rank, and $\nabla^2 f(x)$ is invertible. Then, if d^N satisfies (13), one has*

$$\left\|-\nabla^2 f(x)^{-1}\nabla f(x) - d^N\right\| \;\leq\; \|R_y\|\mathcal{O}(\Delta_y^2),$$

where the multiplicative constant in $\mathcal{O}$ depends on the Lipschitz constant of $\nabla^2 f$.

Hence by controlling the geometry of the points y^ℓ, $\ell = 1, \ldots, p$, around x one can provide an accurate bound when the Hessian of f is invertible and $p \geq n$. In general, we can attempt to control the conditioning of M_L^z, replacing some of the points y^ℓ if necessary. Such a conditioning must eventually become adequate if the vectors $y^\ell - x$ are sufficiently linearly independent and lie in eigenspaces of $\nabla^2 f(x)$ corresponding to eigenvalues not too close to zero.

Using $p = n$ Hessian-vector products at each iteration is certainly not a desirable strategy as that would be equivalent to access the entire Hessian matrix. It is however possible to use $p \ll n$ and still obtain an accurate Newton direction model. The possibility we have in mind is to build upon a previously computed Newton direction model calculated using $p = n$. Let x_{prev} be such an iterate, $y_{prev}^1, \ldots, y_{prev}^n$ be the corresponding sample points, and $z_{prev}^1, \ldots, z_{prev}^n$ be the corresponding Hessian-vector products. Suppose we are now at a new iterate x and we would like to reuse $f(y_{prev}^1), \ldots, f(y_{prev}^n)$ and

$$z_{prev}^1 = \nabla^2 f(x_{prev})(y_{prev}^1 - x_{prev}), \ldots, z_{prev}^n = \nabla^2 f(x_{prev})(y_{prev}^n - x_{prev}).$$

In such a case what we will have in (13) is

$$z_{prev}^\ell \;=\; \nabla^2 f(x_{prev})(y_{prev}^\ell - x_{prev}) \;\simeq\; \nabla f(y_{prev}^\ell) - \nabla f(x_{prev}), \quad \ell = 1, \ldots, p,$$

but what we wish we would have is

$$z^\ell \;=\; \nabla^2 f(x)(y_{prev}^\ell - x) \;\simeq\; \nabla f(y_{prev}^\ell) - \nabla f(x), \quad \ell = 1, \ldots, p,$$

So, one can obtain an approximation to z^ℓ from

$$z_{prev}^\ell + \nabla f(x_{prev}) - \nabla f(x), \quad \ell = 1, \ldots, p. \tag{18}$$

The error in such an approximation is of the

$$\mathcal{O}(\max\{\|y^{\ell}_{prev} - x_{prev}\|^2, \|y^{\ell}_{prev} - x\|^2\}),$$

which would then has to be divided by Δ_z in the context of Theorem 3.2. Of course, if we then keep applying this strategy the error will accumulate over the iterations, but there are certainly remedies such as bringing a few new, fresh z's at each iteration and applying restarts with $p = n$ whenever the conditioning of M^z_L becomes large.

3.3. Numerical results for the determined case using a correction. To use as few Hessian-vector products as possible, we start by using $p = n$ products at iteration zero, to then replace only one interpolation point at each iteration. We choose to replace the point farthest away from the current iterate x. (A perhaps more sound approach would have been to choose the z^{ℓ} that has contributed the most to the conditioning of M^z_L.) A new point is then added, generated in the ball $\mathcal{B}(x, r)$, where

$$r = \min\{10^{-2}, \max\{10^{-4}, \|x - x_{prev}\|\}\}.$$

Therefore, only one more Hessian-vector product and one more function evaluation is required at each iteration. We then replace all other z^{ℓ}_{prev}'s by (18). We monitor the condition number of M^z_L, and apply a restart (with $p = n$ as in iteration 0) whenever $\text{cond}(M^z_L) \geq 10^8$.

A Newton direction model d^N is then calculated by solving (13) directly. To guarantee that we have a descent direction d, meaning that $-\nabla f(x)^\top d > 0$, we modify the d^N from (13) so that $d = d^N - \beta \nabla f(x)$ where β is such that $\cos(d, -\nabla f(x)) = \eta$, and η was set to 0.95.

The modified Newton direction model was then used in a line-search algorithm using the same cubic line search procedure of Subsection 2.3. The comparison is again against the inexact Newton method (as described in [15, Section 7.1]). First we tested the very small problems of Appendix B. Again, we plot performance profiles (see Appendix A) using as performance metric the numbers of Hessian-vector products and iterations (Figure 5) and the number of function evaluations (Figure 6). The results are quite encouraging. We then selected a benchmark of 26 unconstrained nonlinear small problems from the CUTEst collection [11], listed in Appendix D. The experiments are reported in Figures 7 and 8 in the form of the same performance profiles. The results are similar and again promising.

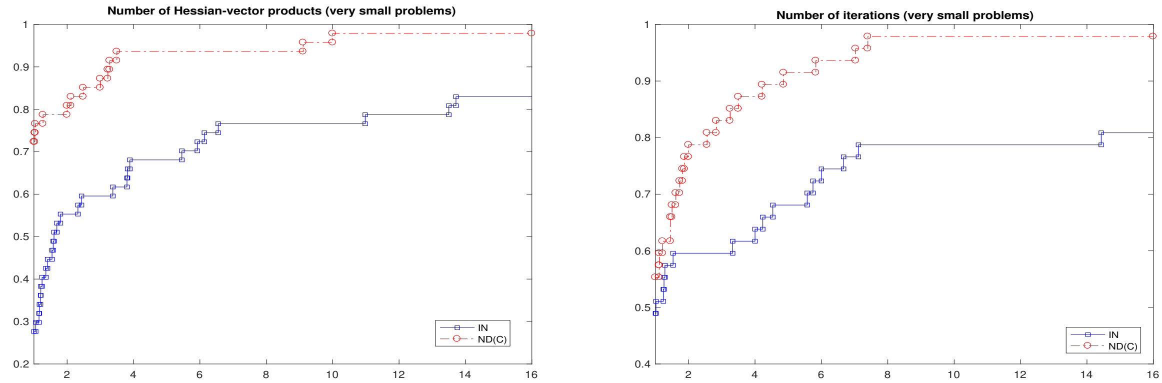

FIGURE 5. Testing the Newton direction recovery within a line-search algorithm. Performance profiles for the numbers of Hessian-vector products and iterations, for the set of very small problems of Appendix B.

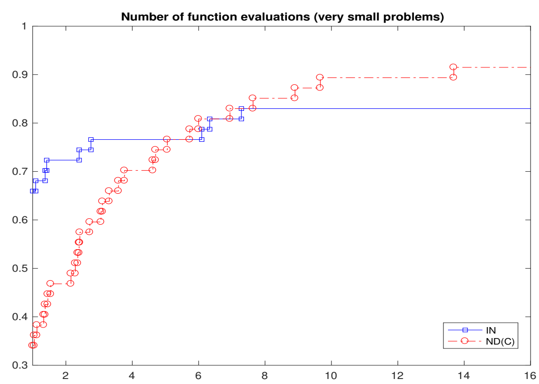

FIGURE 6. Testing the Newton direction recovery within a line-search algorithm. Performance profiles for the numbers of function evaluations for the set of very small problems of Appendix B.

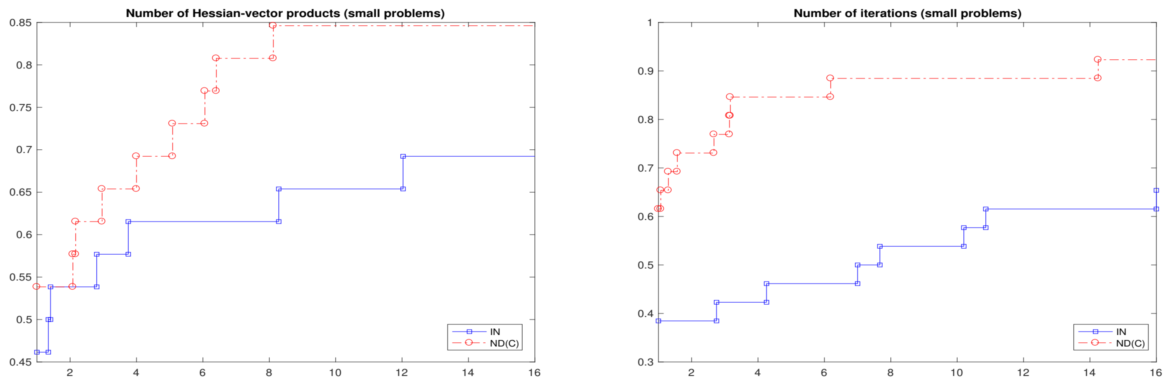

FIGURE 7. Testing the Newton direction recovery within a line-search algorithm. Performance profiles for the numbers of Hessian-vector products and iterations, for the set of small problems of Appendix D.

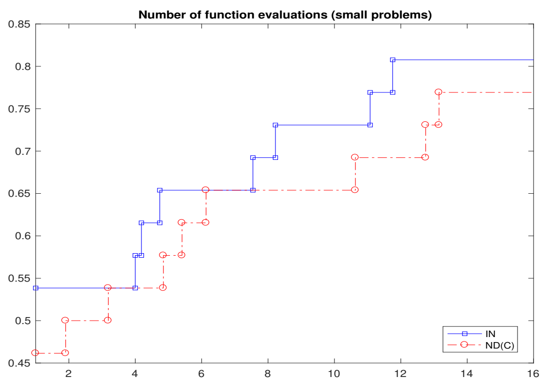

FIGURE 8. Testing the Newton direction recovery within a line-search algorithm. Performance profiles for the numbers of function evaluations for the set of small problems of Appendix D.

4. Final remarks

In this paper we showed how to use interpolation techniques from Derivative-Free Optimization to model Hessian-vector products. We aimed at presenting new, refreshing ideas, laying down the theoretical groundwork for future more elaborated algorithmic developments. Two approaches were presented and analyzed. In the first one, one aims at recovering a model the Hessian matrix, possibly sparse if the true Hessian sparsity pattern is known. A drawback of this approach is that at most one Hessian-vector product can be used in the recovery. The second approach aims at directly recovering the Newton direction itself, and it may incorporate several Hessian-vector products at the same time. However, a dense system of linear equations needs to be solved.

It is left for future work the development of competitive versions of these two approaches for medium/large scale problems. In the particular case of the second approach based on the calculation of a Newton direction model, one can consider solving the linear system (13) using an iterative solver. In such a case, one can easily envision a parallel procedure for the storage of the matrix M_L^z (storing row-wise the vectors z^ℓ's) and the calculation of the products M_L^z times a vector required when applying an iterative solver.

A third recovery approach can also be derived, where the Newton direction and the inverse of the Hessian are recovered at once (possibly never storing the whole inverse, rather forming its product times the gradient). This approach has performed the worse, and we have decided to leave the details for a future PhD thesis of the first author.

A. Performance profiles

Performance profiles [7] are used to compare the performance of several solvers on a set of problems. Let $\mathcal{S}$ be a set of solvers and $\mathcal{P}$ a set of problems. Let $t_{p,s}$ be the performance metric of the solver $s \in \mathcal{S}$ on the problem $p \in \mathcal{P}$. Then the performance profile of solver $s \in \mathcal{S}$ is defined as the fraction of problems where the performance ratio is at most τ,

$$\rho_s(\tau) = \frac{1}{|\mathcal{P}|} \left| \left\{ p \in \mathcal{P} : \frac{t_{p,s}}{\min\{t_{p,s'} : s' \in S\}} \leq \tau \right\} \right|,$$

where $|\mathcal{P}|$ denotes the cardinality of $\mathcal{P}$. The value of $\rho_s(1)$ expresses the percentage of problems on which solver s performed the best. The values of $\rho_s(\tau)$ for large τ indicate the percentage of problems successfully solved by solver s. Hence, $\rho_s(1)$ and $\rho_s(\tau)$ for large τ are, respectively, measures of the efficiency and robustness of a given solver s. Solvers with profiles above others are naturally preferred.

B. Very small test problems

Table 1. List of 48 very small CUTEst test problems.

Name	Dimension	Name	Dimension	Name	Dimension
ALLINITU	4	ARGLINA	10	ARWHEAD	10
BEALE	2	BIGGS6	6	BOX3	3
BROWNAL	10	BRYBND	10	CHNROSNB	10
COSINE	10	CUBE	2	DIXMAANA	15
DIXMAANB	15	DIXMAAND	15	DIXMAANE	15
DIXMAANF	15	DIXMAANG	15	DIXMAANH	15
DIXMAANI	15	DIXMAANJ	15	DIXMAANK	15
DIXMAANL	15	DIXON3DQ	10	DQDRTIC	10
EDENSCH10	10	ENGVAL2	3	EXPFIT	2
FMINSURF	15	GROWTHLS	3	HAIRY	2
HATFLDD	3	HATFLDE	3	HEART8LS	8
HELIX	3	HILBERTA	10	HILBERTB	10
HIMMELBG	2	HUMPS	2	KOWOSB	4
MANCINO	30	MSQRTALS	4	MSQRTBLS	9
POWER	10	SINEVAL	2	SNAIL	2
SPARSINE	10	SPMSRTLS	28	TRIDIA	10

C. Small sparse test problems

Table 2. List of 12 sparse small CUTEst test problems.

Name	Dimension	Name	Dimension	Name	Dimension
BDQRTIC	10	BROYDN7D	50	COSINE	200
DQRTIC	10	EDENSCH	200	ENGVAL1	200
LIARWHD	100	NONSCOMP	50	PENTDI	100
SROSENBR	50	TOINTGSS	50	TRIDIA	200

D. Small test problems

Table 3. List of 26 small CUTEst test problems.

Name	Dimension	Name	Dimension	Name	Dimension
BOX	200	BOXPOWER	200	BRYBND	100
CHNROSNB	50	DIXON3DQ	200	DQDRTIC	100
EDENSCH	200	ENGVAL1	200	EXTROSNB	100
GENHUMPS	100	HILBERTA	200	HILBERTB	200
INTEQNELS	100	LIARWHD	200	MOREBV	200
PENTDI	100	PENALTY1	100	POWELLSG	36
SPARSINE	100	SROSENBR	50	SROSENBR	100
TESTQUAD	100	TOINTGSS	50	TQUARTIC	100
TRIDIA	200	VAREIGVL	100		

References

[1] A. S. Bandeira, K. Scheinberg, and L. N. Vicente. Computation of sparse low degree interpolating polynomials and their application to derivative-free optimization. *Math. Program.*, 134:223–257, 2012.

[2] C. Cartis, N. I. M. Gould, J. Tanner, and Ph. L. Toint. *Personal communication*, June, 2018.

[3] A. R. Conn, K. Scheinberg, and L. N. Vicente. Geometry of interpolation sets in derivative free optimization. *Math. Program.*, 111:141–172, 2008.

[4] A. R. Conn, K. Scheinberg, and L. N. Vicente. *Introduction to Derivative-Free Optimization*. MPS-SIAM Series on Optimization. SIAM, Philadelphia, 2009.

[5] A. R. Curtis, M. J. D. Powell, and J. K. Reid. On the estimation of sparse Jacobian matrices. *J. Inst. Math. Appl*, 13:117–119, 1974.

[6] R. S. Dembo, S. C. Eisenstat, and T. Steihaug. Inexact Newton methods. *SIAM J. Numer. Anal.*, 19:400–408, 1982.

[7] E. D. Dolan and J. J. Moré. Benchmarking optimization software with performance profiles. *Math. Program.*, 91:201–213, 2002.

[8] S. C. Eisenstat and H. F. Walker. Globally convergent inexact Newton methods. *SIAM J. Optim.*, 4:393–422, 1994.

[9] R. Fletcher, A. Grothey, and S. Leyffer. Computing sparse Hessian and Jacobian approximations with optimal hereditary properties. In L. T. Biegler, T. F. Coleman, A. R. Conn, and F. N. Santosa, editors, *Large-Scale Optimization with Applications*, volume 93 of *The IMA Volumes in Mathematics and its Applications*, pages 37–52. Springer, New York, 1997.

[10] N. I. M. Gould, S. Lucidi, M Roma, and Ph. L. Toint. Solving the trust-region subproblem using the Lanczos method. *SIAM J. Optim.*, 9:504–525, 1999.

[11] N. I. M. Gould, D. Orban, and Ph. L. Toint. Cutest: a constrained and unconstrained testing environment with safe threads for mathematical optimization. *Comput. Optim. Appl.*, 60:545–557, 2015.

[12] S. Gratton, C. W. Royer, and L. N. Vicente. A decoupled first/second-order steps technique for nonconvex nonlinear unconstrained optimization with improved complexity bounds. *Math. Program.*, 2019 (to appear).

[13] L. Grippo, F. Lampariello, and S. Lucidi. A nonmonotone line search technique for Newton's method. *SIAM J. Numer. Anal.*, 34:707–716, 1986.

[14] S. G. Nash. Newton-type minimization via the Lanczos method. *SIAM J. Numer. Anal.*, 21:770–788, 1984.

[15] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer-Verlag, Berlin, second edition, 2006.

[16] M. J. D. Powell. Least Frobenius norm updating of quadratic models that satisfy interpolation conditions. *Math. Program.*, 100:183–215, 2004.

[17] M. J. D. Powell and Ph. L. Toint. On the estimation of sparse Hessian matrices. *SIAM J. Numer. Anal.*, 16:1060–1074, 1979.

[18] T. Steihaug. The conjugate gradient method and trust regions in large scale optimization. *SIAM J. Numer. Anal.*, 20:626–637, 1983.

[19] W. Sun and Y. Yuan. *Optimization Theory and Methods: Nonlinear Programming*. Springer, Berlin, 2006.

L. Song
CMUC, Department of Mathematics, University of Coimbra, 3001-501 Coimbra, Portugal
E-mail address: `lili.song@mat.uc.pt`

L. N. Vicente
Department of Industrial and Systems Engineering, Lehigh University, 200 West Packer Avenue, Bethlehem, PA 18015-1582, USA and Centre for Mathematics of the University of Coimbra (CMUC), Portugal
E-mail address: `lnv@mat.uc.pt`