

On limited-memory quasi-Newton methods for minimizing a quadratic function

David Ek* Anders Forsgren*

September 28, 2018

Abstract

The main focus in this paper is exact linesearch methods for minimizing a quadratic function whose Hessian is positive definite. We give two classes of limited-memory quasi-Newton Hessian approximations that generate search directions parallel to those of the method of preconditioned conjugate gradients, and hence give finite termination on quadratic optimization problems. The Hessian approximations are described by a novel compact representation which provides a dynamical framework. We also discuss possible extensions of these classes and show their behavior on randomly generated quadratic optimization problems. The methods behave numerically similar to L-BFGS. Inclusion of information from the first iteration in the limited-memory Hessian approximation and L-BFGS significantly reduces the effects of round-off errors on the considered problems.

In addition, we give our compact representation of the Hessian approximations in the full Broyden class for the general unconstrained optimization problem. This representation consists of explicit matrices and gradients only as vector components.

Keywords. method of conjugate gradients, quasi-Newton method, unconstrained quadratic program, limited-memory method, exact linesearch method

1. Introduction

In this work we mainly study the behavior of limited-memory quasi-Newton methods on unconstrained quadratic optimization problems on the form

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} x^T H x + c^T x, \quad (\text{QP})$$

where $H = H^T$ and $H \succ 0$. (Throughout, we use “ \succ ” to denote positive definite.) In particular, exact linesearch limited-memory quasi-Newton methods that generate search directions parallel to those of the method of preconditioned conjugate gradients (PCG) are considered. Under exact linesearch parallel search directions imply identical iterates. Limited-memory quasi-Newton methods have previously been studied by various authors, e.g., as memory-less quasi-Newton methods by Shanno [19], limited-memory BFGS (L-BFGS) by Nocedal [16] and more recently

*Optimization and Systems Theory, Department of Mathematics, KTH Royal Institute of Technology, SE-100 44 Stockholm, Sweden (daviek@kth.se, andersf@kth.se). Research supported by the Swedish Research Council (VR).

as limited-memory reduced-Hessian methods by Gill and Leonard [12]. In contrast, we specialize to exact linesearch methods for problems on the form (QP). The model method is PCG, which is interpreted as a particular quasi-Newton method as is done by e.g., Shanno [19] and Forsgren and Odland [10]. We start from a result by Forsgren and Odland [10], which provides necessary and sufficient conditions on the Hessian approximation for exact linesearch methods on (QP) to generate search directions that are parallel to those of PCG. The focus is henceforth directly on Hessian approximations with this property. The approximations are described by a novel compact representation which contains explicit matrices together with gradients and search directions as vector components. The framework for the compact representation is first given for the full Broyden class where we consider unconstrained optimization problems on the form

$$\min_{x \in \mathbb{R}^n} f(x), \tag{1.1}$$

where the function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is assumed to be smooth. Compact representations of quasi-Newton matrices have previously been used by various authors but were first introduced by Byrd, Nocedal and Schnabel [1]. They were thereafter extended to the convex Broyden class by Erway and Marcia [5, 6], and to the full Broyden class by DeGuchy, Erway and Marcia [3]. In contrast, we give an alternative compact representation of the Hessian approximations in the full Broyden class which only contains explicit matrices and gradients as vector components. In addition we discuss how exact linesearch is reflected in this representation.

Compact representations of limited-memory Hessian approximations in the Broyden class are also discussed by Byrd, Nocedal and Schnabel [1] and Erway and Marcia [6]. In contrast, our discussion is on limited-memory representations of Hessian approximations intended for exact linesearch methods for problems on the form (QP), and the approximations are not restricted to the Broyden class. In addition, our alternative representation provides a dynamical framework for the construction of limited-memory approximations for the mentioned purpose.

The motivation for this work originates from interior-point methods, which constitute some of the most widely used methods in numerical optimization. As the problems become larger the arising systems of linear equations typically become increasingly computationally expensive to solve and iterative methods may be considered. In exact arithmetic, our model method is the method of preconditioned conjugate gradients, but this method may be too inaccurate in finite precision. Quasi-Newton methods may be expected to be significantly more accurate, but the computational cost is typically too high. In consequence, we aim for less computationally expensive limited-memory versions of quasi-Newton methods that are more accurate than the method of preconditioned conjugate gradients. The goal is to provide better understanding of whether it is viable and/or efficient to use such methods to approximately solve the systems of linear equations that arise as interior-point methods converge.

In Section 2 we provide a brief background to quasi-Newton methods, unconstrained quadratic optimization problems (QP) and to the groundwork that provides the basis for this study. Section 3 contains the alternative compact representation

for the full Broyden class. In Section 4 we present results which include two limited-memory Hessian approximation classes together with a discussion of how to solve the systems of linear equations that arise using reduced-Hessian methods. Section 5 contains numerical results on randomly generated quadratic optimization problems. Finally in Section 6 we give some concluding remarks.

2. Background

In this section we give a short introduction to quasi-Newton methods for unconstrained optimization problems on the form (1.1). Thereafter, we give a background to unconstrained quadratic optimization problems (QP) and to the groundwork that provides the basis for this study.

2.1. Background on quasi-Newton methods

Quasi-Newton methods were first introduced as variable metric methods by Davidon [2] and later formalized by Fletcher and Powell [9]. For a thorough introduction to quasi-Newton methods see, e.g., [7, Chapter 3] and [17, Chapter 6]. In quasi-Newton methods the search direction, p_k , at iteration k is generated by

$$B_k p_k = -g_k, \quad (2.1)$$

where B_k is an approximation of the true Hessian $\nabla^2 f(x_k)$ and g_k is the gradient $\nabla f(x_k)$. It is throughout this work assumed that B_k is symmetric, i.e. $B_k = B_k^T$. However, there are classes that consider asymmetric Hessian approximations, e.g. the three-parameter Huang class [15]. The symmetric part of the Huang class is a two-parameter class that satisfy the scaled secant condition

$$B_k s_{k-1} = \sigma_k y_{k-1}, \quad (2.2)$$

where $s_{k-1} = x_k - x_{k-1}$, $y_{k-1} = g_k - g_{k-1}$ and σ_k is one of the free parameters. The most well-known quasi-Newton class is obtained if $\sigma_k = 1$ in (2.2), namely the one-parameter Broyden class. The Hessian approximations of the Broyden class can be written as

$$\begin{aligned} B_k = & B_{k-1} - \frac{1}{s_{k-1}^T B_{k-1} s_{k-1}} B_{k-1} s_{k-1} s_{k-1}^T B_{k-1} + \frac{1}{y_{k-1}^T s_{k-1}} y_{k-1} y_{k-1}^T \\ & + \phi_{k-1} \omega_{k-1} \omega_{k-1}^T, \end{aligned} \quad (2.3)$$

where

$$\omega_{k-1} = (s_{k-1}^T B_{k-1} s_{k-1})^{1/2} \left(\frac{1}{y_{k-1}^T s_{k-1}} y_{k-1} - \frac{1}{s_{k-1}^T B_{k-1} s_{k-1}} B_{k-1} s_{k-1} \right),$$

with ϕ_{k-1} as the free parameter [8]. The Broyden-Fletcher-Goldfarb-Shanno (BFGS) update scheme is obtained if $\phi_{k-1} = 0$ and Davidon-Fletcher-Powell (DFP) if $\phi_{k-1} = 1$. In this work we study Hessian approximations described by compact representations with gradients and search directions as vector components. We will

therefore throughout this work explicitly use the quantities g , p and the steplength α in all equations. In this notation, the Broyden class Hessian approximations in (2.3) may be written as

$$B_k = B_{k-1} + \frac{1}{g_{k-1}^T p_{k-1}} g_{k-1} g_{k-1}^T + \frac{1}{\alpha_{k-1} (g_k - g_{k-1})^T p_{k-1}} (g_k - g_{k-1}) (g_k - g_{k-1})^T + \phi_{k-1} \omega_{k-1} \omega_{k-1}^T, \quad (2.4)$$

where

$$\omega_{k-1} = (-g_{k-1}^T p_{k-1})^{1/2} \left(\frac{1}{(g_k - g_{k-1})^T p_{k-1}} (g_k - g_{k-1}) - \frac{1}{g_{k-1}^T p_{k-1}} g_{k-1} \right).$$

In (2.4) it may be observed that the previous Hessian approximation is in general updated by a rank-two matrix with range equal to the space spanned by the current and the previous gradient. Furthermore, it is well known that under exact linesearch all Broyden class updates generates identical iterates, as shown by Dixon [4].

The case $\phi_{k-1} = 0$ in (2.4), i.e., the BFGS update, will have a particular role in part of our analysis. We will refer to quantities B_k , p_k and α_k corresponding to this case as B_k^{BFGS} , p_k^{BFGS} and α_k^{BFGS} .

2.2. Background on quadratic problems

Solving (QP) is equivalent to solving the linear system

$$Hx + c = 0, \quad (2.5)$$

which has a unique solution if $H \succ 0$. The quadratic optimization problem in (QP), and hence the linear system (2.5), is in this work considered to be solved by an exact linesearch method on the following form. The steplength, iterate and gradient at iteration k is updated as

$$\alpha_k = -\frac{g_k^T p_k}{p_k^T H p_k}, \quad x_{k+1} = x_k + \alpha_k p_k, \quad g_{k+1} = g_k + \alpha_k H p_k,$$

which together with a specific formula for p_k constitute the particular exact linesearch method. The model exact linesearch method is summarized in the algorithm below.

Algorithm 2.1 An exact linesearch method for solving (QP).

$k \leftarrow 0$, $x_k \leftarrow$ Initial point, $g_k \leftarrow Hx_k + c$

While $\|g_k\| \neq 0$ **do**

$p_k \leftarrow$ search direction

$\alpha_k \leftarrow -\frac{g_k^T p_k}{p_k^T H p_k}$

$x_{k+1} \leftarrow x_k + \alpha_k p_k$

$g_{k+1} \leftarrow g_k + \alpha_k H p_k$

$k \leftarrow k + 1$

End

The search direction in Algorithm 2.1 may be calculated using PCG with a symmetric positive definite preconditioner M . The corresponding algorithm for solving (2.5) may be formulated using the Cholesky factor L defined by $M = LL^T$. This is equivalent to the application of the methods of conjugate gradients (CG) to the preconditioned linear system

$$L^{-1}HL^{-T}\hat{x} + L^{-1}c = 0, \quad (2.6)$$

with $\hat{x} = L^T x$, see, e.g., Saad [18, Chapter 9.2]. If all quantities generated by CG on (2.6) are denoted by " $\hat{\cdot}$ ", then these quantities will relate to those from CG on (2.5) as, $\hat{g} = L^{-1}g$ and $\hat{p} = L^T p$. The iteration space when $M = I$ or when M is an arbitrary symmetric positive definite matrix will thus be related through a linear transformation. In this work the following PCG update is considered,

$$p_k^{PCG} = \begin{cases} -M^{-1}g_0, & k = 0 \\ -M^{-1}g_k + \frac{g_k^T M^{-1}g_k}{g_{k-1}^T M^{-1}g_{k-1}} p_{k-1} & k \geq 1. \end{cases} \quad (2.7)$$

If no preconditioner is used, i.e. $M = I$, then (2.7) is the update referred to as Fletcher-Reeves, which together with the exact linesearch method of Algorithm 2.1 is equivalent to the method of conjugate gradients by Hestenes and Stiefel [14]. If the search direction (2.7) is used in Algorithm 2.1, the method terminates when $\|g_r\| = 0$ for some r where $r \leq n$ and x_r solves (QP). The search directions generated by the method are mutually conjugate with respect to H and satisfy $p_i \in \text{span}(\{M^{-1}g_0, \dots, M^{-1}g_i\})$, $i = 0, \dots, r$. In addition, it holds that the generated gradients are mutually conjugate with respect to M^{-1} , i.e. $g_i^T M^{-1}g_j = 0$, $i \neq j$. By expanding (2.7), the search direction of PCG may be expressed as

$$p_k^{PCG} = -g_k^T M^{-1}g_k \sum_{i=0}^k \frac{1}{g_i^T M^{-1}g_i} M^{-1}g_i. \quad (2.8)$$

The discussion in this work is mainly on Hessian approximations B_k that generate p_k that are parallel to p_k^{PCG} . We will therefore hereinafter only consider the preconditioner $M = B_0$ where B_0 is symmetric positive definite. Forsgren and Odland have provided necessary and sufficient conditions on B_k for an exact linesearch method to generate p_k that are parallel to p_k^{PCG} [10]. This result provides the basis of this work and for completeness it is reviewed below.

Proposition 2.1. (Forsgren and Odland [10, Proposition 4]) *Consider iteration k of the exact linesearch method of Algorithm 2.1 where $1 \leq k < r$. Assume that $p_i = \delta_i p_i^{PCG}$ with $\delta_i \neq 0$ for $i = 0, \dots, k-1$, where p_i^{PCG} , $i = 0, \dots, k-1$, are the search directions of the method of preconditioned conjugate gradients using a positive definite symmetric preconditioning matrix B_0 , as stated in (2.7). Let C_k be defined as*

$$C_k = I - \frac{1}{g_{k-1}^T p_{k-1}} p_{k-1} g_k^T.$$

Then,

$$C_k^{-1} = I + \frac{1}{g_{k-1}^T p_{k-1}} p_{k-1} g_k^T.$$

and it holds that $B_0 C_k p_k^{PCG} = -g_k$. In addition, if p_k is given by $B_k p_k = -g_k$ with B_k nonsingular, then, for any nonzero scalar δ_k , it holds that $p_k = \delta_k p_k^{PCG}$ if and only if

$$B_k C_k^{-1} B_0^{-1} g_k = \frac{1}{\delta_k} g_k,$$

or equivalently if and only if

$$B_k = C_k^T W_k C_k, \text{ with } W_k B_0^{-1} g_k = \frac{1}{\delta_k} g_k, \text{ for } W_k \text{ nonsingular.}$$

Finally, it holds that $B_k \succ 0$ if and only if $W_k \succ 0$.

Proof. See [10, Proposition 4]. ■

With the exact linesearch method of Algorithm 2.1 for solving the unconstrained quadratic optimization problem (QP), parallel search directions imply identical iterates, and therefore search directions parallel to those of PCG imply finite termination. Huang has shown that the quasi-Newton Huang class, the Broyden class and PCG generate parallel search directions [15].

Finally we review a result which is related to the conjugacy of the search directions. The result will have a central part the analysis to come.

Lemma 2.1. Consider iteration k , $1 \leq k < r$, of the exact linesearch method of Algorithm 2.1 for solving the unconstrained quadratic optimization problem (QP). Let B_0 be a symmetric positive definite $n \times n$ matrix. If $p_i = \delta_i p_i^{PCG}$, $\delta_i \neq 0$, $i = 0, \dots, k-1$, then $p_k = \delta_k p_k^{PCG}$, $\delta_k \neq 0$ if and only if $p_k \in \text{span}(\{B_0^{-1} g_0, \dots, B_0^{-1} g_k\})$ and

$$g_i^T p_k = c_k \neq 0, \quad i = 0, \dots, k, \quad (2.9)$$

with $c_k = -\delta_k g_k B_0^{-1} g_k$.

Proof. Note that by the assumptions, g_i , $i = 0, \dots, k$, are identical to those generated by PCG. We first show the only-if direction. Premultiplication of p_k^{PCG} of (2.8) by g_i^T while taking into account the conjugacy of the g_j 's with respect to B_0^{-1} gives $g_i^T p_k^{PCG} = -g_k^T B_0^{-1} g_k$, so that $g_i^T (\delta_k p_k^{PCG}) = c_k$ for $c_k = -\delta_k g_k B_0^{-1} g_k$. In addition, (2.8) shows that $p_k \in \text{span}(\{B_0^{-1} g_0, \dots, B_0^{-1} g_k\})$.

To show the other direction, let

$$p_k = \sum_{j=0}^k \gamma_j B_0^{-1} g_j, \quad (2.10)$$

Premultiplication of (2.10) by g_i^T while taking into account the conjugacy of the g_j 's with respect to B_0^{-1} gives

$$g_i^T p_k = \gamma_i g_i^T B_0^{-1} g_i, \quad i = 0, \dots, k,$$

hence if p_k satisfies (2.9) with $c_k = -\delta_k g_k B_0^{-1} g_k$, then it follows that

$$\gamma_i = -\delta_k \frac{g_k B_0^{-1} g_k}{g_i^T B_0^{-1} g_i}, \quad i = 0, \dots, k. \quad (2.11)$$

Insertion of (2.11) into (2.10) gives $p_k = \delta_k p_k^{PCG}$, with p_k^{PCG} given by (2.8). ■

3. A compact representation of the Broyden class Hessian approximations

In this section we consider unconstrained optimization problems on the form (1.1) and give a compact representation of the Hessian approximations in the full Broyden class. The representation contains only explicit matrices and gradients as vector components. Thereafter we also show how exact linesearch is reflected in the representation.

Lemma 3.1. *Consider iteration k of solving the unconstrained optimization problem (1.1) by a quasi-Newton method where, for a given B_0 , at each iteration i , $i = 0, \dots, k-1$, the search direction p_i has been given by $B_i p_i = -g_i$, where B_i is any nonsingular Broyden class Hessian approximation of the form (2.4). Any Hessian approximation in the Broyden class can then be written as*

$$B_k = B_0 + \sum_{i=0}^{k-1} \left[\frac{1}{g_i^T p_i} g_i g_i^T + \frac{1}{\alpha_i (g_{i+1} - g_i)^T p_i} (g_{i+1} - g_i)(g_{i+1} - g_i)^T + \phi_i \omega_i \omega_i^T \right],$$

where

$$\omega_i = (-g_i^T p_i)^{1/2} \left(\frac{1}{(g_{i+1} - g_i)^T p_i} (g_{i+1} - g_i) - \frac{1}{g_i^T p_i} g_i \right),$$

or equivalently

$$B_k = B_0 + G_k T_k G_k^T,$$

where

$$G_k = [g_0 \quad g_1 \quad \dots \quad g_{k-1} \quad g_k] \in \mathbb{R}^{n \times (k+1)},$$

and $T_k \in \mathbb{R}^{(k+1) \times (k+1)}$ is a symmetric tridiagonal matrix on the form

$$T_k = T_k^C + T_k^\phi, \quad (3.1)$$

with

$$e_1^T T_k^C e_1 = \frac{1}{g_0^T p_0} + \frac{1}{\alpha_0 (g_1 - g_0)^T p_0}, \quad (3.2a)$$

$$e_{i+1}^T T_k^C e_{i+1} = \frac{1}{g_i^T p_i} + \frac{1}{\alpha_{i-1} (g_i - g_{i-1})^T p_{i-1}} + \frac{1}{\alpha_i (g_{i+1} - g_i)^T p_i}, \quad i = 1, \dots, k-1, \quad (3.2b)$$

$$e_{i+1}^T T_k^C e_i = e_i^T T_k^C e_{i+1} = -\frac{1}{\alpha_{i-1} (g_i - g_{i-1})^T p_{i-1}}, \quad i = 1, \dots, k, \quad (3.2c)$$

$$e_{k+1}^T T_k^C e_{k+1} = \frac{1}{\alpha_{k-1} (g_k - g_{k-1})^T p_{k-1}}, \quad (3.2d)$$

$$e_1^T T_k^\phi e_1 = -\phi_0 g_0^T p_0 \left(\frac{1}{(g_1 - g_0)^T p_0} + \frac{1}{g_0^T p_0} \right)^2, \quad (3.2e)$$

$$e_{i+1}^T T_k^\phi e_{i+1} = -\phi_{i-1} g_{i-1}^T p_{i-1} \left(\frac{1}{(g_i - g_{i-1})^T p_{i-1}} \right)^2 - \phi_i g_i^T p_i \left(\frac{1}{(g_{i+1} - g_i)^T p_i} + \frac{1}{g_i^T p_i} \right)^2, \quad i = 1, \dots, k-1, \quad (3.2f)$$

$$e_{i+1}^T T_k^\phi e_i = e_i^T T_k^\phi e_{i+1} = \phi_{i-1} \frac{g_{i-1}^T p_{i-1}}{[(g_i - g_{i-1})^T p_{i-1}]^2} + \phi_{i-1} \frac{1}{(g_i - g_{i-1})^T p_{i-1}}, \quad i = 1, \dots, k, \quad (3.2g)$$

$$e_{k+1}^T T_k^\phi e_{k+1} = -\phi_{k-1} g_{k-1}^T p_{k-1} \left(\frac{1}{(g_k - g_{k-1})^T p_{k-1}} \right)^2. \quad (3.2h)$$

Proof. The result follows directly from telescoping (2.4) and writing it on outer product form. ■

The compact representation in Lemma 3.1 requires storage of $(k+1)$ gradient vectors and an explicit component matrix, T_k , of size $(k+1) \times (k+1)$. In comparison to compact representations given in [1], [5] and [6] that require storage of $2k$ vector-pairs $(B_0 s_i, y_i)$, $i = 0, \dots, k-1$, and an implicit $2k \times 2k$ component matrix. However when considering the inverse Hessian approximation, although the amount of storage is preserved in the suggested representation, it does not provide an explicit component matrix. For a discussion on the corresponding representation of the inverse Hessian approximation, see Appendix C.

One of the most commonly used quasi-Newton update schemes is the BFGS update, i.e., the update where B_k takes the form (2.4) for $\phi_{k-1} = 0$. We will put a particular focus on this update in the remainder of this section and refer to quantities B_k , p_k and α_k corresponding to this case as B_k^{BFGS} , p_k^{BFGS} and α_k^{BFGS} .

The compact representation for the corresponding Hessian approximations B_k^{BFGS} are given in the following corollary.

Corollary 3.1. Consider iteration k of solving the unconstrained optimization problem (1.1) by a quasi-Newton method where, for a given B_0 , at each iteration i , $i = 0, \dots, k-1$, the search direction p_i has been given by $B_i^{BFGS} p_i = -g_i$. The BFGS Hessian approximation B_k^{BFGS} can then be written as

$$B_k^{BFGS} = B_0 + \sum_{i=0}^{k-1} \left[\frac{1}{g_i^T p_i} g_i g_i^T + \frac{1}{\alpha_i (g_{i+1} - g_i)^T p_i} (g_{i+1} - g_i)(g_{i+1} - g_i)^T \right],$$

or equivalently

$$B_k^{BFGS} = B_0 + G_k T_k^{BFGS} G_k^T,$$

where

$$G_k = [g_0 \ g_1 \ \dots \ g_{k-1} \ g_k] \in \mathbb{R}^{n \times (k+1)},$$

and $T_k^{BFGS} \in \mathbb{R}^{(k+1) \times (k+1)}$ is a symmetric tridiagonal matrix with elements given in (3.2a)-(3.2d).

Proof. The BFGS updates are obtained by setting $\phi_i = 0$ for $i = 0, \dots, k-1$ in (2.4). The result then follow directly from Lemma 3.1 by setting $T_k^{BFGS} = T_k^C$. ■

3.1. Exact linesearch

In this section we consider the case when the linesearch steplength is chosen such that

$$g_k^T p_{k-1} = 0,$$

i.e. α_{k-1} is chosen as the steplength to a stationary point along p_{k-1} . Under exact linesearch the rank-one matrix $\phi_{k-1} \omega_{k-1} \omega_{k-1}^T$ in (2.4) reduces to

$$\phi_{k-1} \omega_{k-1} \omega_{k-1}^T = -\frac{\phi_{k-1}}{g_{k-1}^T p_{k-1}} g_k g_k^T. \quad (3.3)$$

Consequently, the choice of Broyden member is only reflected in the diagonal of T_k in Lemma 3.1. This can be observed directly in (3.2e) - (3.2h) by making use of the exact linesearch condition $g_i^T p_{i-1} = 0$, $i = 1, \dots, k$. All non-diagonal terms of T_k^ϕ become zero and the diagonal terms may be simplified to

$$e_{i+1}^T T_k^\phi e_{i+1} = \begin{cases} 0 & i = 0, \\ -\frac{\phi_{i-1}}{g_{i-1}^T p_{i-1}} & i = 1, \dots, k. \end{cases} \quad (3.4)$$

Any Hessian approximation in the Broyden class may in fact be written as $B_k = B_k^{BFGS} - (\phi_{k-1}/g_{k-1}^T p_{k-1}) g_k g_k^T$, thus B_k is independent of ϕ_i for $i = 0, \dots, k-2$ and the choice of Broyden member only affects the scaling of the search direction. This result is not new, however an addition to this and an alternative proof using the proposed compact representation is given in Lemma 3.2. The result is given to emphasize the properties that follow solely from exact linesearch. In comparison to the properties that stem from exact linesearch on quadratic optimization problems (QP), which are discussed in Section 4.

Lemma 3.2. Consider iteration k of solving the unconstrained optimization problem (1.1) by a quasi-Newton method where, for a given B_0 , at each iteration i , $i = 0, \dots, k-1$, the search direction p_i has been given by $B_i p_i = -g_i$, where B_i is any nonsingular Broyden class Hessian approximation of the form (2.4). In addition, assume that α_i and α_i^{BFGS} correspond to the same stationary point of $f(x_i + \alpha p_i)$, i.e., $\alpha_i p_i = \alpha_i^{BFGS} p_i^{BFGS}$, $i = 0, \dots, k-1$. Any Hessian approximation in the Broyden class can then be written as

$$B_k = B_k^{BFGS} - \frac{\phi_{k-1}}{g_{k-1}^T p_{k-1}} g_k g_k^T,$$

and the search direction satisfies

$$p_k = \frac{1}{1 - \frac{\phi_{k-1}}{g_{k-1}^T p_{k-1}} g_k^T p_k^{BFGS}} p_k^{BFGS},$$

if B_k^{BFGS} is nonsingular and $1 - \frac{\phi_{k-1}}{g_{k-1}^T p_{k-1}} g_k^T p_k^{BFGS} \neq 0$. In addition, it holds that

$$\frac{1}{g_i^T p_i} - \frac{\phi_{i-1}}{g_{i-1}^T p_{i-1}} = \frac{1}{g_i^T p_i^{BFGS}}, \quad i = 1, \dots, k.$$

Proof. Recall that all the Broyden class updates generate identical iterates under exact linesearch, hence the generated gradients are independent of the member. The proof will be by induction. As base step, consider $k = 1$. B_0 is independent of ϕ and thus

$$B_1 = B_1^{BFGS} - \frac{\phi_0}{g_0^T p_0} g_1 g_1^T.$$

Note that $B_1^{BFGS} p_1^{BFGS} = -g_1$, furthermore if B_1^{BFGS} is nonsingular and $1 - \frac{\phi_0}{g_0^T p_0} g_1^T p_1^{BFGS} \neq 0$ then the requirements of Lemma A.1 are satisfied. It then follows that

$$p_1 = \frac{1}{1 - \frac{\phi_0}{g_0^T p_0} g_1^T p_1^{BFGS}} p_1^{BFGS},$$

and

$$\frac{1}{g_1^T p_1} - \frac{\phi_0}{g_0^T p_0} = \frac{1}{g_1^T p_1^{BFGS}}.$$

For the induction step, assume that the result holds for $k = 0, \dots, r-1$ and consider $k = r$. Any Broyden class Hessian approximation can by Lemma 3.1 and (3.3) be written as

$$B_r = B_0 + \sum_{i=0}^{r-1} \left[\left(\frac{1}{g_i^T p_i} - \frac{\phi_{i-1}}{g_{i-1}^T p_{i-1}} \right) g_i g_i^T + \frac{1}{\alpha_i (g_{i+1} - g_i)^T p_i} (g_{i+1} - g_i)(g_{i+1} - g_i)^T \right] - \frac{\phi_{r-1}}{g_{r-1}^T p_{r-1}} g_r g_r^T. \quad (3.5)$$

To simplify the notation the quantity $\phi_{-1}/g_{-1}^T p_{-1}$ is used and set to zero. By the assumptions $\alpha_i p_i = \alpha_i^{BFGS} p_i^{BFGS}$ for $i = 0, \dots, r-1$ and hence the second term

in the sum is independent of the Broyden member. Furthermore, by the induction hypothesis it holds that

$$\frac{1}{g_i^T p_i} - \frac{\phi_{i-1}}{g_{i-1}^T p_{i-1}} = \frac{1}{g_i^T p_i^{BFGS}}, \quad i = 1, \dots, r-1. \quad (3.6)$$

Insertion of (3.6) into (3.5) and using Corollary 3.1 gives

$$B_r = B_r^{BFGS} - \frac{\phi_{r-1}}{g_{r-1}^T p_{r-1}} g_r g_r^T.$$

It holds that $B_r^{BFGS} p_r^{BFGS} = -g_r$ and if B_r^{BFGS} is nonsingular and $1 - \frac{\phi_{r-1}}{g_{r-1}^T p_{r-1}} g_r^T p_r^{BFGS} \neq 0$ then the requirements of Lemma A.1 are satisfied. Thus it follows that

$$p_r = \frac{1}{1 - \frac{\phi_{r-1}}{g_{r-1}^T p_{r-1}} g_r^T p_r^{BFGS}} p_r^{BFGS},$$

and

$$\frac{1}{g_r^T p_r} - \frac{\phi_{r-1}}{g_{r-1}^T p_{r-1}} = \frac{1}{g_r^T p_r^{BFGS}}.$$

This completes the induction. \blacksquare

The result of Lemma 3.2 directly shows that all members of the Broyden class generate parallel search directions under exact linesearch, and explicitly how the choice of member affects the scaling.

4. Quadratic problems

In this section we consider quadratic problems on the form (QP) and start from the requirement that p_k generated by the exact linesearch method of Algorithm 2.1 shall be parallel to p_k^{PCG} . Motivated by the performance of the Broyden class, we start by considering Hessian approximations $B_k = B_{k-1} + U_k$ where U_k is a symmetric rank-two matrix with $\mathcal{R}(U_k) = \text{span}(\{g_{k-1}, g_k\})$ and thereafter look at generalizations. A characterization of all such update matrices U_k is provided as well as a multi-parameter Hessian approximation that generates $p_k = \delta_k p_k^{PCG}$ for scalar a δ_k . Thereafter, we consider limited-memory Hessian approximations with this property, discuss potential extensions and how to solve the arising systems with a reduced-Hessian method.

Proposition 4.1. *Consider iteration k , $1 \leq k < r$, of the exact linesearch method of Algorithm 2.1 for solving the unconstrained quadratic optimization problem (QP). Assume that $p_i = \delta_i p_i^{PCG}$ with $\delta_i \neq 0$ for $i = 0, \dots, k-1$, where p_i^{PCG} , $i = 0, \dots, k-1$, are the search directions of the method of preconditioned conjugate gradients using a positive definite symmetric preconditioning matrix B_0 , as stated in (2.7). Let B_{k-1} be a nonsingular matrix such that $B_{k-1} p_{k-1} = -g_{k-1}$ and $B_{k-1} B_0^{-1} g_k = g_k$. Let $U_k = B_k - B_{k-1}$ and assume that B_k and p_k satisfy $B_k p_k = -g_k$ with B_k*

nonsingular. Then, if U_k is symmetric, rank-two with $\mathcal{R}(U_k) = \text{span}(\{g_{k-1}, g_k\})$ it holds that $p_k = \delta_k p_k^{PCG}$, $\delta_k \neq 0$, if and only if

$$U_k = \frac{1}{g_{k-1}^T p_{k-1}} g_{k-1} g_{k-1}^T + \rho_{k-1} (g_k - g_{k-1}) (g_k - g_{k-1})^T + \left(\frac{1}{\delta_k} - 1\right) \frac{1}{g_k^T B_0^{-1} g_k} g_k g_k^T,$$

where ρ_{k-1} is a free parameter.

Proof. The assumptions in the proposition together with Proposition 2.1 and $B_k = B_{k-1} + U_k$ give the following necessary and sufficient condition on U_k such that $p_k = \delta_k p_k^{PCG}$ for a scalar $\delta_k \neq 0$.

$$U_k \left(B_0^{-1} g_k + \frac{g_k^T B_0^{-1} g_k}{p_{k-1}^T g_{k-1}} p_{k-1} \right) = \left(\frac{1}{\delta_k} - 1 \right) g_k + \frac{g_k^T B_0^{-1} g_k}{p_{k-1}^T g_{k-1}} g_{k-1}. \quad (4.1)$$

Any symmetric rank-two matrix, U_k , with $\mathcal{R}(U_k) = \text{span}(\{g_{k-1}, g_k\})$ can be written as

$$U_k = \eta_{k-1} g_{k-1} g_{k-1}^T + \rho_{k-1} (g_k - g_{k-1}) (g_k - g_{k-1})^T + \varphi_k g_k g_k^T. \quad (4.2)$$

Insertion of (4.2) into (4.1), taking into account $g_k^T B_0^{-1} g_{k-1} = 0$ and $g_k^T p_{k-1} = 0$ gives

$$\varphi_k g_k^T B_0^{-1} g_k g_k + \eta_{k-1} g_k^T B_0^{-1} g_k g_{k-1} = \left(\frac{1}{\delta_k} - 1 \right) g_k + \frac{g_k^T B_0^{-1} g_k}{p_{k-1}^T g_{k-1}} g_{k-1},$$

which is independent of ρ_{k-1} . Identification of terms gives

$$\begin{aligned} \varphi_k &= \left(\frac{1}{\delta_k} - 1 \right) \frac{1}{g_k^T B_0^{-1} g_k}, \\ \eta_{k-1} &= \frac{1}{g_{k-1}^T p_{k-1}}. \end{aligned}$$

■

The result in Proposition 4.1 provides a two-parameter update matrix, U_k . If the conditions of Proposition 4.1 apply then it follows directly from U_k that the iterates satisfy the scaled secant condition (2.2). This can be seen by considering $B_k \alpha_{k-1} p_{k-1}$ with $B_k = B_{k-1} + U_k$

$$(B_{k-1} + U_k) \alpha_{k-1} p_{k-1} = -\rho_{k-1} \alpha_{k-1} g_{k-1}^T p_{k-1} (g_k - g_{k-1}).$$

Consequently the characterization in Proposition 4.1 provides a class which under exact linesearch on quadratic optimization problems (QP) is equivalent to the symmetric Huang class. The scaling in the secant condition does neither affect the search direction nor the scaling of it. Utilizing the secant condition sets the parameter

$$\rho_{k-1} = -1 / (\alpha_{k-1} g_{k-1}^T p_{k-1})$$

that together with the change of variable

$$\varphi_k = \left(\frac{1}{\delta_k} - 1 \right) \frac{1}{g_k^T B_0^{-1} g_k} = -\frac{\phi_{k-1}}{g_{k-1}^T p_{k-1}},$$

gives the exact linesearch form of the Broyden class matrices in (2.4). Hence, as expected, utilizing the secant condition fixates one of the parameters and gives the Broyden class. It can also be observed in the update matrix U_k of Proposition 4.1 why the Broyden symmetric rank-one update breaks down for the unit steplength. At iteration k when $\alpha_{k-1} = 1$, then $\rho_{k-1} = -1/g_{k-1}^T p_{k-1}$ causes the $g_{k-1} g_{k-1}^T$ component to become zero and hence δ_k , or ϕ_{k-1} if preferred, can not be chosen such that U_k reduces to a rank-one matrix. Moreover, if U_k of Proposition 4.1 is required to be a symmetric rank-one matrix, ρ_{k-1} may be eliminated from the update formula to give the one-parameter class suggested by Forsgren and Odland [10].

The result of Proposition 4.1 motivates further study of the structure in the corresponding Hessian approximations.

Lemma 4.1. *Consider iteration k , $1 \leq k < r$, of the exact linesearch method of Algorithm 2.1 for solving the unconstrained quadratic optimization problem (QP). Assume that $B_i p_i = -g_i$, $i = 0, \dots, k-1$, where B_0 is a symmetric positive definite $n \times n$ matrix and*

$$\begin{aligned} B_i &= B_{i-1} + \frac{1}{g_{i-1}^T p_{i-1}} g_{i-1} g_{i-1}^T \\ &\quad + \rho_{i-1} (g_i - g_{i-1})(g_i - g_{i-1})^T + \varphi_i g_i g_i^T, \end{aligned} \quad (4.3)$$

for $i = 1, \dots, k$ with ρ_{i-1} and φ_i chosen such that B_i is nonsingular. Then B_k takes the form

$$B_k = B_0 + \sum_{i=0}^{k-1} \left(-\frac{1}{g_i^T B_0^{-1} g_i} g_i g_i^T + \rho_i (g_{i+1} - g_i)(g_{i+1} - g_i)^T \right) + \varphi_k g_k g_k^T. \quad (4.4)$$

Proof. With the assumptions in the proposition, the update of (4.3) satisfies the requirements of Proposition 4.1 and hence for each i , $i = 0, \dots, k-1$, it follows that $p_i = \delta_i p_i^{PCG}$ where $\delta_i = 1/(1 + \varphi_i g_i^T B_0^{-1} g_i)$ and $\varphi_0 = 0$. Premultiplication of $p_i = \delta_i p_i^{PCG}$ by g_i^T gives

$$g_i^T p_i = \frac{1}{1 + \varphi_i g_i^T B_0^{-1} g_i} g_i^T p_i^{PCG}, \quad i = 0, \dots, k-1, \quad (4.5)$$

Inverting (4.5) and taking into account that $g_i^T p_i^{PCG} = -g_i^T B_0^{-1} g_i$, $i = 0, \dots, k-1$, gives

$$\frac{1}{g_i^T p_i} + \varphi_i = -\frac{1}{g_i^T B_0^{-1} g_i}, \quad i = 0, \dots, k-1. \quad (4.6)$$

By telescoping (4.3) at iteration k we obtain

$$B_k = B_0 + \sum_{i=0}^{k-1} \left[\left(\frac{1}{g_i^T p_i} + \varphi_i \right) g_i g_i^T + \rho_i (g_{i+1} - g_i)(g_{i+1} - g_i)^T \right] + \varphi_k g_k g_k^T. \quad (4.7)$$

Insertion of (4.6) into (4.7) gives (4.4). ■

Lemma 4.1 and (4.4) show that if the search direction at iteration k , $1 \leq k < r$, is given by $B_k p_k = -g_k$ with B_k given by (4.4), then p_k is independent of all ρ_i , $i = 0, \dots, k-1$, at every iteration k as long as B_k is nonsingular. This result is formalized in the following proposition.

Proposition 4.2. *Consider iteration k , $1 \leq k < r$, of the exact linesearch method of Algorithm 2.1 for solving the unconstrained quadratic optimization problem (QP). Assume that $p_i = \delta_i p_i^{PCG}$ with $\delta_i \neq 0$ for $i = 0, \dots, k-1$, where p_i^{PCG} , $i = 0, \dots, k-1$, are the search directions of the method of preconditioned conjugate gradients using a positive definite symmetric preconditioning matrix B_0 , as stated in (2.7). Let p_k satisfy $B_k p_k = -g_k$ where*

$$B_k = B_0 + \sum_{i=0}^{k-1} \left(-\frac{1}{g_i^T B_0^{-1} g_i} g_i g_i^T + \rho_i^{(k)} (g_{i+1} - g_i)(g_{i+1} - g_i)^T \right) + \varphi_k g_k g_k^T, \quad (4.8)$$

with $\rho_i^{(k)} > 0$, $i = 0, \dots, k-1$, and φ_k chosen such that B_k is nonsingular. Then,

$$p_k = \frac{1}{1 + \varphi_k g_k^T B_0^{-1} g_k} p_k^{PCG}. \quad (4.9)$$

In particular, if $\varphi_k > -1/(g_k^T B_0^{-1} g_k)$, then $B_k \succ 0$.

Proof. From Proposition 4.1 and Lemma 4.1 it follows that B_k given by (4.4) generates $p_k = \delta_k p_k^{PCG}$ where $\delta_k = 1/(1 + \varphi_k g_k^T B_0^{-1} g_k)$ and hence satisfies

$$(g_{i+1} - g_i)^T p_k = 0, \quad i = 0, \dots, k-1, \quad (4.10)$$

by Lemma 2.1. If $\rho_i > 0$, $i = 0, \dots, k-1$, and φ_k chosen such that B_k is nonsingular then the solution is unique and independent of ρ_i , $i = 0, \dots, k-1$, and thus $\rho_i = \rho_i^{(k)}$, $i = 0, \dots, k-1$. Moreover, if $\varphi_k = 0$ then the matrix of (4.8) is positive definite by Lemma A.2 and it then follows from Lemma A.1 that $B_k \succ 0$ for $\varphi_k > -1/(g_k^T B_0^{-1} g_k)$. ■

The result in Proposition 4.2 together with exact linesearch method of Algorithm 2.1 provide a multiple-parameter class that generates parallel search directions to those of PCG. In the framework of updates on the form $B_k = B_{k-1} + U_k$ this class allows update matrices with $\mathcal{R}(U_k) = \text{span}(\{g_0, \dots, g_k\})$ and reduces to the symmetric Huang class if $\mathcal{R}(U_k) = \text{span}(\{g_{k-1}, g_k\})$ is required. In (4.8) of Proposition 4.2 it can be observed that the direction is determined by the components in the first term of the sum, compare with (2.8). The parameter φ_k only scales the direction and it is independent of $\rho_i^{(k)}$, $i = 0, \dots, k-1$. Certain choices of these parameters merely guarantee nonsingularity of the Hessian approximation and may provide numerical stability. We will therefore refer to the terms corresponding to the parameters $\rho_i^{(k)}$, $i = 0, \dots, k-1$, as stabilizers.

4.1. Limited-memory Hessian approximations

In this section we extend the above discussion to limited-memory Hessian approximations. Note that the approximation given in (4.4) can be written on outer product form, similarly as in Lemma 3.1, using gradients as vector components. This gives a form, $B_k = B_0 + M_k$ where $\mathcal{R}(M_k) = \text{span}(\{g_0, \dots, g_k\})$. From (2.8) it directly follows that $p_k^{PCG} \in \text{span}(\{B_0^{-1}g_0, \dots, B_0^{-1}g_k\})$ and that p_k^{PCG} has a nonzero component in every direction $B_0^{-1}g_i$, $i = 0, \dots, k$. Hessian approximations on the form of (4.4) will consequently not be able to generate p_k parallel to p_k^{PCG} with (2.1) if gradient information is discarded. However, this can be done, as e.g. in [1], by at each iteration recalculating the basis vectors from the vector pairs (s_i, y_i) , $i = k - m, \dots, k - 1$, or, as shown in Theorem 4.1 below, by adding a correction term to the right hand side of the quasi-Newton equation (2.1).

Theorem 4.1. *Consider iteration k , $1 \leq k < r$, of the exact-linesearch method of Algorithm 2.1 for solving the unconstrained quadratic optimization problem (QP). Assume that $p_i = \delta_i p_i^{PCG}$ with $\delta_i \neq 0$ for $i = 0, \dots, k - 1$, where p_i^{PCG} , $i = 0, \dots, k - 1$, are the search directions of the method of preconditioned conjugate gradients using a positive definite symmetric preconditioning matrix B_0 , as stated in (2.7). Let $\mathcal{A}_k = \{j_1, \dots, j_{m_k}\} \subseteq \{0, \dots, k\}$ with $j_1 < j_2 < \dots < j_{m_k}$ such that $k \in \mathcal{A}_k$ and let $\mathcal{I}_k = \{0, \dots, k\} \setminus \mathcal{A}_k$. Furthermore, let p_k satisfy $B_k p_k = -N_k g_k$ where*

$$B_k = B_0 + \sum_{i=1}^{m_k-1} \left[\left(\frac{1}{g_{j_i}^T p_{j_i}} + \varphi_{j_i} \right) g_{j_i} g_{j_i}^T + \rho_{j_i}^{(k)} (g_{j_{i+1}} - g_{j_i})(g_{j_{i+1}} - g_{j_i})^T \right] + \varphi_k g_k g_k^T, \quad (4.11a)$$

and

$$N_k = \left(I - \frac{1}{1 + \varphi_k g_k^T B_0^{-1} g_k} \sum_{i \in \mathcal{I}_k} \left(\frac{1}{g_i^T p_i} + \varphi_i \right) g_i g_i^T B_0^{-1} \right) g_k, \quad (4.11b)$$

with $\varphi_0 = 0$, φ_i , $i = 1, \dots, k - 1$, and $\rho_{j_i}^{(k)}$, $i = 1, \dots, m_k - 1$, chosen such that B_k is nonsingular. Then,

$$B_k = B_0 + \sum_{i=1}^{m_k-1} \left(-\frac{1}{g_{j_i}^T B_0^{-1} g_{j_i}} g_{j_i} g_{j_i}^T + \rho_{j_i}^{(k)} (g_{j_{i+1}} - g_{j_i})(g_{j_{i+1}} - g_{j_i})^T \right) + \varphi_k g_k g_k^T,$$

and

$$p_k = \frac{1}{1 + \varphi_k g_k^T B_0^{-1} g_k} p_k^{PCG}.$$

In particular, if $\rho_{j_i}^{(k)} > 0$, $i = 1, \dots, m_k - 1$, and $\varphi_k > -1/(g_k^T B_0^{-1} g_k)$, then $B_k \succ 0$.

Proof. The assumptions in the theorem satisfy the requirements of Proposition 4.2 and Lemma 4.1. Consider B_k given by (4.8) of Proposition 4.2, the search direction

generated by $B_k p_k = -g_k$ then satisfies (4.9). The corresponding matrix B_k can by Lemma 4.1 equivalently be written as the telescoped form of (4.3)

$$B_0 + \sum_{i=0}^{k-1} \left[\left(\frac{1}{g_i^T p_i} + \varphi_i \right) g_i g_i^T + \rho_i^{(k)} (g_{i+1} - g_i)(g_{i+1} - g_i)^T \right] + \varphi_k g_k g_k^T, \quad (4.12)$$

where $\varphi_0 = 0$. Identifying terms in (4.12) and (4.8) of Proposition 4.2 gives

$$\frac{1}{g_i^T p_i} + \varphi_i = -\frac{1}{g_i^T B_0^{-1} g_i}, \quad i = 0, \dots, k-1. \quad (4.13)$$

Insertion of (4.12) into the quasi-Newton equation (2.1) gives

$$\begin{aligned} & \left(B_0 + \sum_{i=0}^{k-1} \left[\left(\frac{1}{g_i^T p_i} + \varphi_i \right) g_i g_i^T + \rho_i^{(k)} (g_{i+1} - g_i)(g_{i+1} - g_i)^T \right] + \varphi_k g_k g_k^T \right) p_k \\ &= -g_k. \end{aligned} \quad (4.14)$$

The direction satisfies (4.9) and hence it follows from Lemma 2.1 that $p_k^T g_i = c_k$, $i = 0, \dots, k$. Premultiplication of (4.9) by g_k^T gives

$$c_k = -\frac{g_k^T B_0^{-1} g_k}{1 + \varphi_k g_k^T B_0^{-1} g_k}. \quad (4.15)$$

Consider (4.14), multiplication of p_k with the terms in the sum corresponding to indices in \mathcal{I}_k , application of Lemma 2.1 and insertion of (4.15) gives

$$\begin{aligned} & \left(B_0 + \sum_{i \in \mathcal{A}_k} \left[\left(\frac{1}{g_i^T p_i} + \varphi_i \right) g_i g_i^T + \rho_i (g_{i+1} - g_i)(g_{i+1} - g_i)^T \right] + \varphi_k g_k g_k^T \right) p_k \\ &= -g_k + \frac{g_k^T B_0^{-1} g_k}{1 + \varphi_k g_k^T B_0^{-1} g_k} \sum_{i \in \mathcal{I}_k} \left(\frac{1}{g_i^T p_i} + \varphi_i \right) g_i, \end{aligned} \quad (4.16)$$

which gives (4.11a). Insertion of (4.13) into the matrix of (4.16) gives

$$\begin{aligned} & \left(B_0 + \sum_{i \in \mathcal{A}_k} \left[-\frac{1}{g_i^T B_0^{-1} g_i} g_i g_i^T + \rho_i (g_{i+1} - g_i)(g_{i+1} - g_i)^T \right] + \varphi_k g_k g_k^T \right) p_k \\ &= - \left(I - \frac{1}{1 + \varphi_k g_k^T B_0^{-1} g_k} \sum_{i \in \mathcal{I}_k} \left(\frac{1}{g_i^T p_i} + \varphi_i \right) g_i g_k^T B_0^{-1} \right) g_k. \end{aligned} \quad (4.17)$$

Denote the matrix of the left hand side of (4.16) as B_k and the matrix of right hand side as N_k . By Lemma A.2 $B_k \succ 0$ if $\varphi_k = 0$ and hence $p_k = p_k^{PCG}$ is the unique solution to (4.17) with $\varphi_k = 0$. It then follows from Lemma A.1 that $B_k \succ 0$ if $\varphi_k > -1/(g_k^T B_0^{-1} g_k)$ and that (4.9) is the unique solution to (4.16). \blacksquare

The result in Theorem 4.1 together with the exact linesearch method in Algorithm 2.1 provides a limited-memory method that generates p_k parallel to p_k^{PCG} . If all indices are chosen to be active then the Hessian approximation in (4.11a) is equivalent to (4.8) of Proposition 4.2. Conversely, if the indices corresponding to all previous gradients, i.e. $i = 0, \dots, k-1$ are inactive, and $\varphi_k = 0$ for all k then (4.11) with $B_k p_k = -N_k g_k$ is equivalent to the PCG update (2.7) with preconditioner B_0 . The update scheme of Theorem 4.1 contains only gradients as vector components and with the exact linesearch method of Algorithm 2.1 the finite termination property is maintained. However, this is at the expense of adding a correction term on the right hand side.

Note that the if part of Proposition 4.1 may be extended to updates on the form

$$U_k = \frac{1}{g_{k-1}^T p_{k-1}} g_{k-1} g_{k-1}^T + \rho_{k-1} \left(\sum_{j=0}^k m_j^{(k)} g_j \right) \left(\sum_{j=0}^k m_j^{(k)} g_j \right)^T + \left(\frac{1}{\delta_k} - 1 \right) \frac{1}{g_k^T B_0^{-1} g_k} g_k g_k^T,$$

where $\sum_{j=0}^k m_j^{(k)} = 0$. Lemma 4.1, Proposition 4.2 and hence also Theorem 4.1 may then with Lemma A.8 similarly be extended to hold for updates on this form. However, in our opinion this would not provide a significant increase in understanding but instead make the analysis more tedious and difficult to follow. We therefore chose to give the result in Proposition 4.1 for rank-two update matrices with $\mathcal{R}(U_k) = \text{span}(\{g_{k-1}, g_k\})$. Moreover, the update in Theorem 4.1 relies heavily on the result in Lemma 2.1 which is exact on quadratic problems. For non-quadratic problems other more accurate approximations and modifications may be considered to improve the method.

The discussion has so far been on Hessian approximations on the form $B_k = B_0 + M_k$ where M_k has gradients as basis components. The discussion will now be extended to also consider M_k that in addition includes information from the search directions in the basis. By (2.7) it is sufficient to have $\mathcal{R}(M_k) = \text{span}(\{B_0 p_{k-1}, g_k\})$. A Hessian approximation that fulfills this and gives $p_k = p_k^{PCG}$ is given in Proposition 2.1 as $B_k = C_k^T B_0 C_k$. The idea is to combine this approximation with the stabilizers of Proposition 4.2 and allow for scaling. The resulting Hessian approximation together with some of its properties is given in the theorem below.

Theorem 4.2. *Consider iteration k , $1 \leq k < r$, of the exact linesearch method of Algorithm 2.1 for solving the unconstrained quadratic optimization problem (QP). Assume that $p_i = \delta_i p_i^{PCG}$ with $\delta_i \neq 0$ for $i = 0, \dots, k-1$, where p_i^{PCG} , $i = 0, \dots, k-1$, are the search directions of the method of preconditioned conjugate gradients using a positive definite symmetric preconditioning matrix B_0 , as stated in (2.7). Let $\mathcal{A}_k \subseteq \{0, \dots, k-1\}$ and let p_k satisfy $B_k p_k = -g_k$ with*

$$B_k = C_k^T B_0 C_k + \sum_{i \in \mathcal{A}_k} \rho_i^{(k)} (g_{i+1} - g_i) (g_{i+1} - g_i)^T + \varphi_k g_k g_k^T, \quad (4.18)$$

where

$$C_k = \left(I - \frac{1}{g_{k-1}^T p_{k-1}} p_{k-1} g_k^T \right),$$

with $\rho_i^{(k)}$, $i \in \mathcal{A}_k$, and φ_k chosen such that B_k is nonsingular. Then

$$p_k = \left(\frac{1}{1 + \varphi_k g_k^T B_0^{-1} g_k} \right) p_k^{PCG}.$$

In particular, if $\rho_i^{(k)} > 0$, $i \in \mathcal{A}_k$, and $\varphi_k > -1/(g_k^T B_0^{-1} g_k)$, then $B_k \succ 0$.

Proof. Consider the case $\varphi_k = 0$. The equation $C_k^T B_0 C_k p_k = -g_k$ has by Proposition 2.1 the unique solution $p_k = p_k^{PCG}$. If the matrix remains nonsingular adding terms orthogonal to p_k have no effect on the direction. By Lemma 2.1

$$(g_{i+1} - g_i)^T p_k = 0, \quad i \in \mathcal{A}_k. \quad (4.19)$$

Let $u \in \mathbb{R}^n$ be a nonzero vector, pre- and postmultiplication of (4.18) with $\varphi_k = 0$ by u^T respectively u gives

$$u^T B_k u = u^T C_k^T B_0 C_k u + \sum_{i=1}^{m_k-1} \rho_{j_i}^{(k)} \left((g_{j_{i+1}} - g_{j_i})^T u \right)^2. \quad (4.20)$$

It follows that B_k is positive definite if

$$u^T C_k^T B_0 C_k u > 0, \quad (4.21a)$$

$$\sum_{i \in \mathcal{A}_k} \rho_i^{(k)} \left((g_{i+1} - g_i)^T u \right)^2 \geq 0, \quad (4.21b)$$

The positive definiteness of B_0 and the existence of C_k^{-1} gives (4.21a) and (4.21b) is satisfied if $\rho_i^{(k)} > 0$, $i \in \mathcal{A}_k$. Proving that B_k with $\varphi_k = 0$ is positive definite if $\rho_i^{(k)} > 0$, $i \in \mathcal{A}_k$. It then follows from Lemma A.1 that (4.18) is positive definite if, in addition, $\varphi_k > -1/(g_k^T B_0^{-1} g_k)$ and that p_k obtained from $B_k p_k = -g_k$ with B_k given in (4.18) satisfies

$$p_k = \left(\frac{1}{1 + \varphi_k g_k^T B_0^{-1} g_k} \right) p_k^{PCG}.$$

■

The results in Theorem 4.1 and Theorem 4.2 provide multi-parameter limited-memory Hessian approximations where the memory usage can be changed between the iterations. The information in the Hessian approximation may be chosen as the method progresses and there is no restriction to only include information from the m_k -latest iterations. All information may also be expressed in terms of search directions and the current gradient g_k . This provides the ability to reduce the amount of storage when the arising systems are solved by reduced-Hessian methods, described in Section 4.2, with search directions in the basis.

Note that the result in Theorem 4.2 can be generalized by instead introducing stabilizers on the form

$$\sum_{i \in \mathcal{A}_k} \left[\rho_i^{(k)} \left(\sum_{j=0}^k m_{ij}^{(k)} g_j \right) \left(\sum_{j=0}^k m_{ij}^{(k)} g_j \right)^T \right],$$

where $\mathcal{A}_k \subseteq \{0, \dots, k-1\}$ and $\sum_{j=0}^k m_{ij}^{(k)} = 0$ for all $i \in \mathcal{A}_k$. Or alternatively to Hessian approximations on the form $B_k = C_k^T B_0 C_k + F_k$ as long as B_k remains nonsingular and $F_k p_k = 0$. However, for the same reasons as above and due to the numerical properties, shown in the Section 5, we chose to give the result for the formulation in Theorem 4.2.

4.2. Solving the systems

In this section we discuss solving systems of linear equations using reduced-Hessian methods. These methods provide an alternative procedure for solving systems arising in quasi-Newton methods. We follow Gill and Leonard [11, 12] and refer to their work for a thorough introduction.

Assume that the Hessian approximation given by (4.18) of Theorem 4.2 is used together with the exact linesearch method of Algorithm 2.1 for solving the unconstrained quadratic optimization problem (QP). The search direction at iteration k then satisfies $p_k = \delta_k p_k^{PCG}$ for a scalar δ_k and hence by (2.7) $p_k \in \text{span}(\{p_{k-1}, B_0^{-1} g_k\})$. Define $\mathcal{S}_k^m = \{p_{k-1}, B_0^{-1} g_k\}$ and let \mathcal{S}_k be a subspace such that $\mathcal{S}_k^m \subseteq \mathcal{S}_k$. Furthermore let S_k be a matrix whose columns span \mathcal{S}_k and Z_k be the matrix defined by the QR-factorization $S_k = Z_k R_k$ where R_k is a nonsingular upper triangular matrix. It then follows that the search direction can be written as $p_k = Z_k u_k$ for some vector u_k . Premultiplication of the quasi-Newton equation (2.1) by Z_k^T together with $p_k = Z_k u_k$ gives

$$Z_k^T B_k Z_k u_k = -Z_k^T g_k, \quad (4.22)$$

which has a unique solution if B_k is positive definite. Hence $p_k = Z_k u_k$ where u_k satisfies (4.22). Note that the analogous procedure is also applicable for the result of Theorem 4.1 where the Hessian approximation is given by (4.11a) and p_k is generated by $B_k p_k = -N_k g_k$.

The minimal space required is $\mathcal{S}_k = \mathcal{S}_k^m$ but other feasible choices are for example $\mathcal{S}_k = \{B_0^{-1} g_0, \dots, B_0^{-1} g_k\}$, by (2.8), or $\mathcal{S}_k = \{p_{t-1}, B_0^{-1} g_t, \dots, B_0^{-1} g_k\}$ where $0 < t < k$.

5. Numerical results

In this section we give numerical results for solving randomly generated quadratic optimization problems on the form (QP). Our framework is a MATLAB implementation where the arising systems of linear equations were solved by MATLAB's built in solver. We refer to Gill and Leonard [11, 12] for a more detailed update and solve

of the reduced systems. The Hessians were symmetric positive definite and the condition number corresponding to the problems with $n = 40, 200-300$ and 1000 variables were in the order of 10^2-10^3 respectively 10^4 and 10^5 . All figures correspond to representative results from approximately 100 simulations. Convergence for a member in the proposed class of quasi-Newton methods (4.8) in Proposition 4.2, here denoted by MuP, is shown in Figure 1. The figure also contains the convergence of the BFGS method and PCG in both finite and exact arithmetic, all with $B_0 = I$. In this study we consider exact arithmetic PCG as the original but with 512 digits precision. The parameters of (4.8) were chosen as follows, $\varphi_k = 0$ for all k and

$$\rho_i^{(k)} = \xi_i^{(k)} \rho_i^B, \quad i = 0, \dots, k-1,$$

where $\xi_i^{(k)}$, $i = 0, \dots, k-1$, are normally distributed random variables and ρ_i^B , $i = 0, \dots, k-1$, are the quantities corresponding to the secant condition. Note that the scaling of $\rho_i^{(k)}$, $i = 0, \dots, k-1$, are randomly changed for every k .

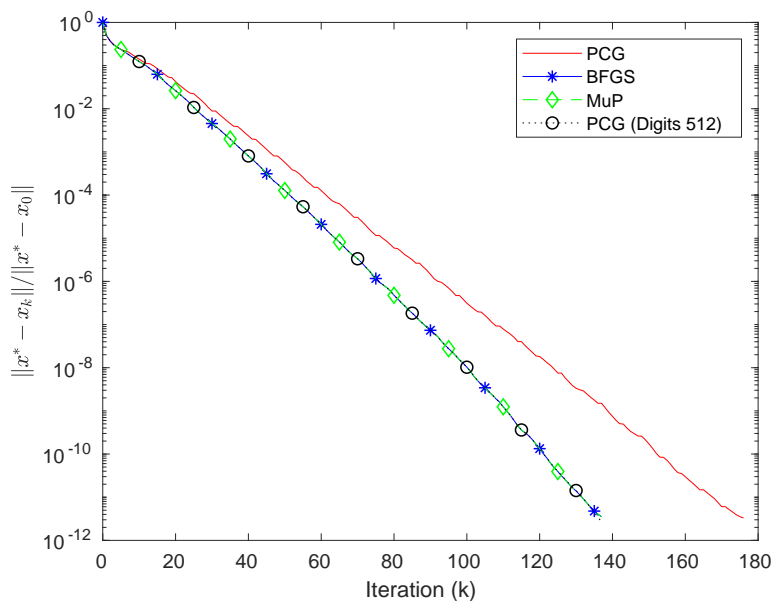


Figure 1: Convergence for solving randomly generated quadratic problems with $n = 300$ variables and $\xi_i^{(k)} \in [10^{-1}, 10^8]$, $i = 0, \dots, k-1$.

All the methods compared in Figure 1 are with the exact linesearch method of Algorithm 2.1 equivalent in exact arithmetic on unconstrained quadratic optimization problems (QP). However, in finite arithmetic this is not the case. As can be seen in the figure, PCG suffers from round-off errors while BFGS behaves like the exact arithmetic PCG. The maximum error from all simulations between the iterates of BFGS and exact PCG was $5.1 \cdot 10^{-14}$, i.e.

$$\max_i \|x_i^{BFGS} - x_i^{PCG}\| = 5.1 \cdot 10^{-14}.$$

Consequently, the BFGS method does not suffer from round-off errors on these quadratic optimization problems. By the result of Proposition 4.2 it is not required to fix the parameters $\rho_i^{(k)}$, $i = 0, \dots, k-1$, and as Figure 1 shows there is an interval where this result also holds in finite arithmetic. The secant condition is expected to provide an appropriate scaling of the quantities since it gives the true Hessian in n iterations. Our results indicate that there is no particular benefit for the quadratic case to choose the values given by the secant condition. This freedom may be useful, since values of $\rho_i^{(k)}$ close to zero for some i may make the Hessian approximation close to singular, and such values could potentially be avoided.

Convergence for the method with limited-memory scheme (4.11) of Theorem 4.1, here denoted by LC, with $\rho_i^{(k)} = \rho_i^B$, $i = 0, \dots, k-1$, and $\varphi_k = 0$ for all k is shown in Figure 2 and 3. The figures also contain the convergence of the BFGS method, L-BFGS as proposed by Nocedal in [16] and PCG, all with $B_0 = I$. In Figure 2 and 3 we also show that the limited-memory methods are able to maintain the exact arithmetic behavior for an increased number of iterations if information from the first iteration is included in the Hessian approximation. To make a fair comparison, we also modify the standard L-BFGS Hessian approximation to also include information from the first iteration. A comparison between the LC and L-BFGS versions is shown in the Table 1. Note that LC uses information from $(m+1)$ gradients to match the gradient information of m vector pairs (y, s) in L-BFGS.

Table 1: A comparison between the considered LC and L-BFGS versions.

	Vector information	Version 1 [Standard]	Version 2 [-0]
L-BFGS	(y, s)	m -latest	(y_0, s_0) and $(m-1)$ -latest
LC	g	$(m+1)$ -latest	g_0 and m -latest

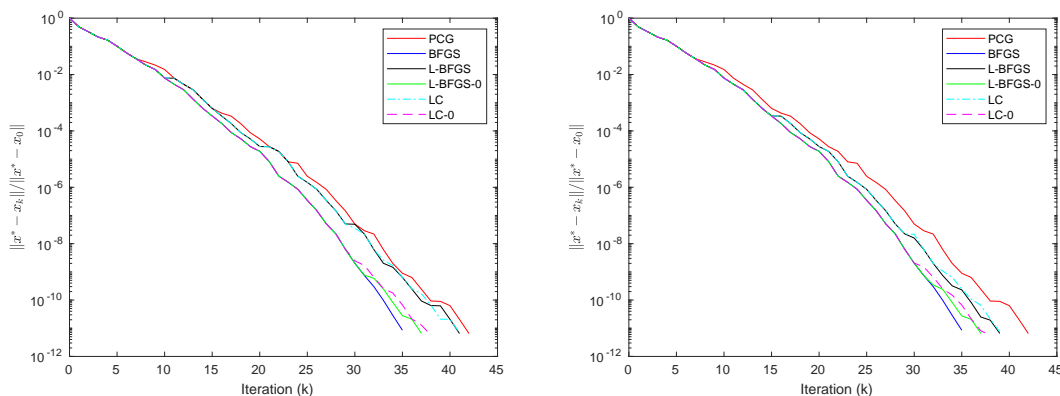


Figure 2: Convergence for solving randomly generated quadratic problems with $n = 40$ variables. The left figure corresponds to $m = 3$ and the right to $m = 8$.

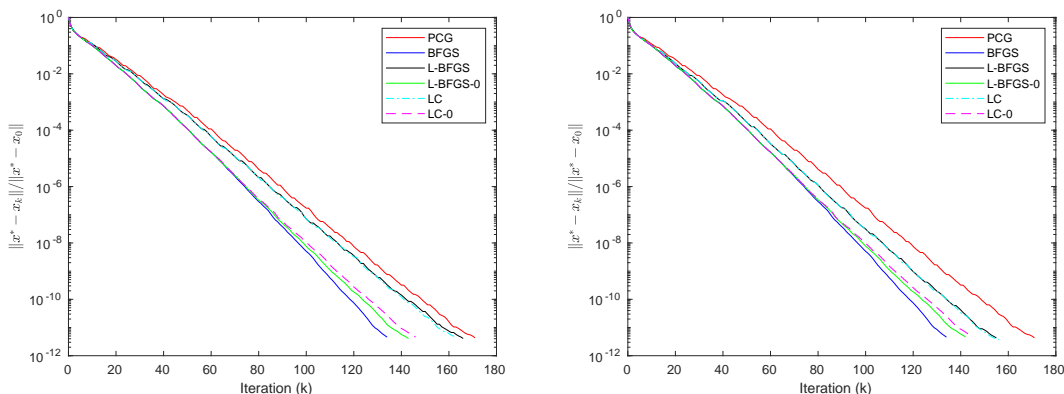


Figure 3: Convergence for solving randomly generated quadratic problems with $n = 200$ variables. The left figure corresponds to $m = 3$ and the right to $m = 8$.

The convergence of LC in Figure 2 is similar to L-BFGS and lies between the convergence of BFGS and PCG. The figure verifies the theoretical result for which the method behaves more similar to BFGS the more information that is included in the limited-memory Hessian approximation. When information is discarded both LC and L-BFGS lose the exact arithmetic behavior and convergence is slowed down. Figure 3 shows that these characteristics are preserved as the dimension of the system increases. The figures also show that the methods are able to maintain the exact arithmetic behavior for an increased number of iterations, and hence reduce round-off error effects and the total number of iterations, by including information from the first iteration in the Hessian approximations. In this case L-BFGS slightly outperforms LC but one should bear in mind the difference in information. Partly information from search directions but also that L-BFGS-0 includes y_0 that also contains information from g_1 which LC-0 does not have any information from. In addition, both LC-0 and L-BFGS-0 are less sensitive to changes in m compared to their respective standard version.

Next we show the convergence for the quasi-Newton method that uses the Hessian approximation (4.18) of Theorem 4.2, here denoted by symPCGs, with $\rho_i^{(k)} = \rho_i^B$, $i = 0, k - m + 1, \dots, k - 1$, and $\varphi_k = 0$ for all k . Figure 4 also contains the convergence of the BFGS method, L-BFGS-0, LC-0 and PCG, all with $B_0 = I$. Both symPCGs and LC-0 were solved with the technique described in Section 4.2 using $S_k = \begin{pmatrix} p_0 & p_1 & p_{k-2} & p_{k-1} & B_0^{-1}g_k \end{pmatrix}$ for $k > 4$. Hence systems of size at most 5×5 were solved at every iteration.

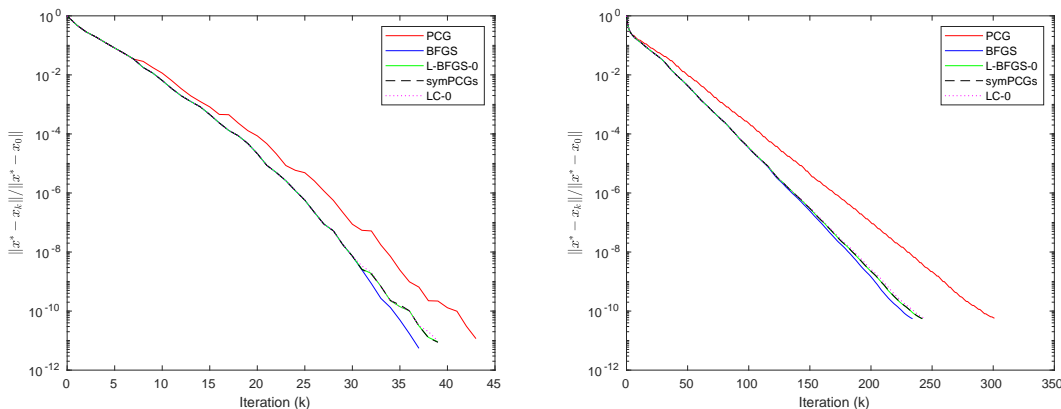


Figure 4: Convergence for solving randomly generated quadratic problems with $m = 5$. The left figure corresponds to $n = 40$ variables and the right to $n = 1000$.

The convergence of symPCGs in Figure 4 is comparable with the convergence of L-BFGS-0. Note that the gradient information in the corresponding limited-memory Hessian approximations are identical whereas the L-BFGS-0 approximation consist of information from additional search directions. The performance of LC-0 is improved in Figure 4 compared to in Figure 2 and 3 due to better numerical properties of the reduced solve. Furthermore, the right part of Figure 4 shows that the round-off error effects can be significantly reduced by including information from the first iteration.

6. Conclusion

In this work we have given one multi-parameter and two limited-memory quasi-Newton Hessian approximation classes which on quadratic optimization problems (QP) with the exact linesearch method of Algorithm 2.1 generate p_k parallel to p_k^{PCG} . In addition, we characterized all symmetric rank-two update matrices, U_k with $\mathcal{R}(U_k) = span(\{g_{k-1}, g_k\})$ which has this property. The Hessian approximations were described by a novel compact representation which framework was first presented in Section 3 for the full Broyden class on unconstrained optimization problems (1.1). The representation of the full Broyden class consist only of explicit matrices and gradients as vector components.

We emphasize that our way of stating the equivalence to PCG together with our alternative representation illustrate the freedom that exists and provide a dynamical framework for the construction of limited-memory Hessian approximations.

Numerical simulations on randomly generated unconstrained quadratic optimization problems have shown that for these problems the multi-parameter class, with parameters within a certain range, is equivalent to the BFGS method in finite arithmetic. It was also shown that finite arithmetic BFGS behaves as exact PCG on the considered problems. The characteristics of the convergence of the proposed limited-memory methods were evaluated and it was shown that they are numerically comparable with L-BFGS. It was also shown that on these problems, including in-

formation from the first iteration in the Hessian approximation significantly reduces round-off error effects.

The results of this work are meant to contribute to the theoretical and numerical understanding of limited-memory quasi-Newton methods for minimizing a quadratic function. We hope that they can lead to further research on limited-memory methods for unconstrained optimization problems. In particular, limited-memory methods for minimizing a near-quadratic function and for systems arising as interior-point methods converge.

A. Appendix

Lemma A.1. *If $Ax = b$, with A nonsingular then*

$$(A + \gamma bb^T)y = b, \quad \text{for } y = \frac{1}{1 + \gamma b^T x}x,$$

if $1 + \gamma b^T x \neq 0$. If, in addition, $b^T x \neq 0$, it holds that

$$\frac{1}{b^T y} = \frac{1}{b^T x} + \gamma. \quad (\text{A.1})$$

Finally, if $A = A^T \succ 0$, then $b^T x > 0$ and $A + \gamma bb^T \succ 0$ if and only if

$$\gamma > -\frac{1}{b^T x}.$$

Proof. Assume that $Ax = b$ where A is nonsingular. Premultiplication of $(A + \gamma bb^T)y = b$ by A^{-1} gives

$$(I + \gamma A^{-1}bb^T)y = A^{-1}b. \quad (\text{A.2})$$

Insertion of $x = A^{-1}b$ into (A.2) and rearranging gives

$$y = (1 - \gamma b^T y)x. \quad (\text{A.3})$$

Insertion of $y = \alpha x$ into (A.3) and solving for α yields

$$\alpha = \frac{1}{1 + \gamma b^T x}, \quad 1 + \gamma b^T x \neq 0.$$

The result in (A.1) follows by premultiplication of $y = \frac{1}{1 + \gamma b^T x}x$ by b^T and rearranging. For the final result, note that $b^T x = x^T Ax > 0$ since $A \succ 0$ and that

$$(A + \gamma bb^T) = A^{1/2} \left(I + \gamma A^{-1/2}bb^T A^{-1/2} \right) A^{1/2},$$

which is a congruent transformation and hence $I + \gamma A^{-1/2}bb^T A^{-1/2} \succ 0$ if and only if $A + \gamma bb^T \succ 0$. Then consider the similarity transformation

$$A^{-1/2} \left(I + \gamma A^{-1/2}bb^T A^{-1/2} \right) A^{1/2} = I + \gamma x b^T,$$

where the only eigenvalue not equal to unity is $1 + \gamma b^T x$, which is positive only if $\gamma > -\frac{1}{b^T x}$, $b^T x \neq 0$. ■

Lemma A.2. Let B_0 be a symmetric positive definite $n \times n$ matrix and let g_i , $i = 0, \dots, k$, be nonzero vectors that are conjugate with respect to B_0^{-1} . Define B_k as

$$B_k = B_0 + \sum_{i=0}^{k-1} \left(-\frac{1}{g_i^T B_0^{-1} g_i} g_i g_i^T + \rho_i (g_{i+1} - g_i)(g_{i+1} - g_i)^T \right), \quad (\text{A.4})$$

where $\rho_i \in \mathbb{R}$, $i = 0, \dots, k-1$. Then $B_k \succ 0$ if $\rho_i > 0$, $i = 1, \dots, k-1$.

Proof. Any vector p in \mathbb{R}^n can be written as

$$p = \sum_{i=0}^k \alpha_i B_0^{-1} g_i + B_0^{-1} u, \quad \text{with } g_i^T B_0^{-1} u = 0, \quad i = 0, \dots, k. \quad (\text{A.5})$$

Insertion of (A.5) into $p^T B_k p$ gives

$$\begin{aligned} p^T B_k p &= p^T \left(B_0 + \sum_{i=0}^{k-1} \left[-\frac{1}{g_i^T B_0^{-1} g_i} g_i g_i^T + \rho_i (g_{i+1} - g_i)(g_{i+1} - g_i)^T \right] \right) p \\ &= p^T B_0 p - \sum_{i=0}^{k-1} \frac{(g_i^T p)^2}{g_i^T B_0^{-1} g_i} + \sum_{i=0}^{k-1} \rho_i ((g_{i+1} - g_i)^T p)^2 \\ &= \sum_{i=0}^k \alpha_i^2 g_i^T B_0^{-1} g_i + u^T B_0^{-1} u - \sum_{i=0}^{k-1} \frac{(\alpha_i g_i^T B_0^{-1} g_i)^2}{g_i^T B_0^{-1} g_i} + \sum_{i=0}^{k-1} \rho_i ((g_{i+1} - g_i)^T p)^2 \\ &= \alpha_k^2 g_k^T B_0^{-1} g_k + u^T B_0^{-1} u + \sum_{i=0}^{k-1} \rho_i (\alpha_{i+1} g_{i+1} B_0^{-1} g_{i+1} - \alpha_i g_i B_0^{-1} g_i)^2. \quad (\text{A.6}) \end{aligned}$$

For the remainder of the proof, let $\rho_i > 0$, $i = 0, \dots, k-1$. It follows from (A.6) that B_k is positive semidefinite with $p^T B_k p = 0$ only if

$$\alpha_k^2 g_k^T B_0^{-1} g_k = 0, \quad (\text{A.7a})$$

$$u^T B_0^{-1} u = 0, \quad (\text{A.7b})$$

$$\alpha_{i+1} g_{i+1} B_0^{-1} g_{i+1} - \alpha_i g_i B_0^{-1} g_i = 0, \quad i = 0, \dots, k-1. \quad (\text{A.7c})$$

From the positive definiteness of B_0 , (A.7a) gives $\alpha_k = 0$, which in combination with (A.7c) gives $\alpha_i = 0$, $i = 0, \dots, k$. In addition, (A.7b) gives $u = 0$. Therefore, $p^T B_k p = 0$ only if $p = 0$, proving that B_k is positive definite. ■

Lemma A.3. Let B_0 be a symmetric positive definite $n \times n$ matrix and let g_i , $i = 0, \dots, k$, be nonzero vectors that are conjugate with respect to B_0^{-1} . Define B_k as

$$B_k = B_0 + \sum_{i=0}^{k-1} \left(-\frac{1}{g_i^T B_0^{-1} g_i} g_i g_i^T + \rho_i^{(k)} \left(\sum_{j=0}^k m_{ij}^{(k)} g_j \right) \left(\sum_{j=0}^k m_{ij}^{(k)} g_j \right)^T \right), \quad (\text{A.8})$$

where $\rho_i^{(k)} \in \mathbb{R}$, $i = 0, \dots, k-1$ and $\sum_{j=0}^k m_{ij}^{(k)} = 0$, $i = 0, \dots, k-1$. Then $B_k \succ 0$ if $\rho_i^{(k)} > 0$, $i = 0, \dots, k-1$ and $M^{(k)}$ has full row rank, where $M^{(k)}$ denotes the $k \times (k+1)$ matrix with elements $m_{ij}^{(k)}$, $i = 0, \dots, k-1$, $j = 0, \dots, k$.

Proof. Any vector p in \mathbb{R}^n can be written as

$$p = \sum_{i=0}^k \alpha_i B_0^{-1} g_i + B_0^{-1} u, \quad \text{with } g_i^T B_0^{-1} u = 0, \quad i = 0, \dots, k. \quad (\text{A.9})$$

Insertion of (A.9) into $p^T B_k p$ gives

$$\begin{aligned} p^T B_k p &= p^T \left(B_0 + \sum_{i=0}^{k-1} \left[-\frac{1}{g_i^T B_0^{-1} g_i} g_i g_i^T + \rho_i \left(\sum_{j=0}^k m_{ij}^{(k)} g_j \right) \left(\sum_{j=0}^k m_{ij}^{(k)} g_j \right)^T \right] \right) p \\ &= p^T B_0 p - \sum_{i=0}^{k-1} \frac{(g_i^T p)^2}{g_i^T B_0^{-1} g_i} + \sum_{i=0}^{k-1} \rho_i \left(\sum_{j=0}^k m_{ij}^{(k)} g_j^T p \right)^2 \\ &= \sum_{i=0}^k \alpha_i^2 g_i^T B_0^{-1} g_i + u^T B_0^{-1} u - \sum_{i=0}^{k-1} \frac{(\alpha_i g_i^T B_0^{-1} g_i)^2}{g_i^T B_0^{-1} g_i} + \sum_{i=0}^{k-1} \rho_i \left(\sum_{j=0}^k m_{ij}^{(k)} g_j^T p \right)^2 \\ &= \alpha_k^2 g_k^T B_0^{-1} g_k + u^T B_0^{-1} u + \sum_{i=0}^{k-1} \rho_i \left(\sum_{j=0}^k m_{ij}^{(k)} \alpha_j g_j^T B_0^{-1} g_j \right)^2. \end{aligned} \quad (\text{A.10})$$

For the remainder of the proof, let $\rho_i > 0$, $i = 0, \dots, k-1$. It follows from (A.10) that B_k is positive semidefinite with $p^T B_k p = 0$ only if

$$\alpha_k^2 g_k^T B_0^{-1} g_k = 0, \quad (\text{A.11a})$$

$$u^T B_0^{-1} u = 0, \quad (\text{A.11b})$$

$$\sum_{j=0}^k m_{ij}^{(k)} \alpha_j g_j^T B_0^{-1} g_j = 0, \quad i = 0, \dots, k-1. \quad (\text{A.11c})$$

From the given properties of $M^{(k)}$, it follows that $\sum_{j=0}^k m_{ij}^{(k)} v_j = 0$, $i = 0, \dots, k-1$, for some vector $v \in \mathbb{R}^{k+1}$, implies that $v_j = c$, $j = 0, \dots, k$, for some constant c . Therefore, (A.11c) implies that $\alpha_j g_j^T B_0^{-1} g_j = c$, $j = 0, \dots, k$, for some c . But (A.11a) gives $\alpha_k = 0$, so that $c = 0$. Therefore, $\alpha_j = 0$, $j = 0, \dots, k$. In addition, (A.11b) gives $u = 0$. Consequently, $p^T B_k p = 0$ only if $p = 0$, proving that B_k is positive definite. ■

B. An alternative compact representation of BFGS

The Hessian approximations corresponding to the BFGS method can with the same technique as in Section 3 be described by a compact representation where the component matrix is diagonal.

Lemma B.1. Consider iteration k of solving the unconstrained optimization problem (1.1) by a quasi-Newton method where, for a given B_0 , at each iteration i , $i = 0, \dots, k-1$, the search direction p_i has been given by $B_i^{BFGS} p_i = -g_i$. The BFGS Hessian approximation B_k^{BFGS} can then be written as

$$B_k^{BFGS} = B_0 + \Upsilon_k D_k \Upsilon_k^T, \tag{B.1}$$

where

$$\Upsilon_k = [g_0 \ y_0 \ \dots \ g_{k-1} \ y_{k-1}] \in \mathbb{R}^{n \times 2k}, \tag{B.2}$$

and $D_k \in \mathbb{R}^{2k \times 2k}$ is a diagonal matrix on the form

$$D_k = \begin{bmatrix} \frac{1}{g_0^T p_0} & & & & & & & \\ & \frac{1}{y_0^T s_0} & & & & & & \\ & & \ddots & & & & & \\ & & & \frac{1}{g_{k-1}^T p_{k-1}} & & & & \\ & & & & & & \frac{1}{y_{k-1}^T s_{k-1}} & \\ & & & & & & & \end{bmatrix}. \tag{B.3}$$

Proof. The BFGS updates are obtained by setting $\phi_{k-1} = 0$ in (2.4). The result follows directly by telescoping and rewriting on outer-product form. \blacksquare

The general Broyden class has, with the representation in Lemma B.1, also a tridiagonal component matrix. The BFGS Hessian approximation can thus be seen as a special case when the component matrix reduces to a diagonal matrix. This is at the expense of a component matrix D_k of size $(2k)^2 / (k+1)^2$ times the size compared to T_k^{BFGS} of Corollary 3.1 which is tridiagonal. If the current gradient is not orthogonal to the previous search direction, i.e., the linesearch is not exact, then the tridiagonal matrix T_k of Lemma 3.1 can also be reduced to a diagonal matrix with a particular choice of ϕ_{k-1} for each k .

The transformation $G_k E_k = \Upsilon_k$ provides a relation between the two compact representations in Corollary 3.1 and Lemma B.1. The transformation matrix E_k is given by

$$E_k = \begin{bmatrix} 1 & -1 & & & & & & \\ & 1 & 1 & -1 & & & & \\ & & & 1 & & & & \\ & & & & \ddots & & & \\ & & & & & 1 & -1 & \\ & & & & & & & 1 \end{bmatrix} \in \mathbb{R}^{(k+1) \times 2k}.$$

At every iteration k the matrix expands by one row and two columns where the lower right corner is the (2×2) -block $\begin{bmatrix} 1 & -1 \\ & 1 \end{bmatrix}$.

Limited-memory Hessian approximations similar to those in Section 4 can be derived for the compact representation in Lemma B.1 using the same techniques.

C. Inverses

The inverse of the compact representation in Lemma 3.1 and Lemma B.1 can be computed with the Sherman-Morrison-Woodbury formula [13]. Assume that the component matrix T_k of Lemma 3.1 is nonsingular. The inverse corresponding to the representation in Lemma 3.1 is then given by

$$B_k^{-1} = B_0^{-1} - B_0^{-1}G_k (T_k^{-1} + G_k^T B_0^{-1}G_k)^{-1} G_k^T B_0^{-1},$$

or

$$B_k^{-1} = B_0^{-1} \left(B_0 - G_k (T_k^{-1} + G_k^T B_0^{-1}G_k)^{-1} G_k^T \right) B_0^{-1}.$$

The representation of the BFGS scheme in Lemma B.1 allows for a more explicit expression for the inverse. Namely

$$B_k^{-1} = B_0^{-1} (B_0 - \Upsilon_k M_k^{-1} \Upsilon_k^T) B_0^{-1},$$

where $M_k = D_k^{-1} + \Upsilon_k^T B_0^{-1} \Upsilon_k$ with elements

$$m_{ii} = \begin{cases} g_i^T B_0^{-1} g_i + g_i^T p_i & \text{if } i \text{ even,} \\ y_i^T B_0^{-1} y_i + y_i^T s_i & \text{if } i \text{ odd,} \end{cases}$$

and

$$m_{ij} = m_{ji} = g_i^T B_0^{-1} y_j, \quad i \neq j,$$

where $i = 0, \dots, k-1$ and $j = 0, \dots, k-1$.

Acknowledgement

We thank Elias Jarlebring for valuable discussions on finite precision arithmetic.

References

- [1] R. H. Byrd, J. Nocedal, and R. B. Schnabel. Representations of quasi-Newton matrices and their use in limited memory methods. *Math. Programming*, 63(2, Ser. A):129–156, 1994.
- [2] W. C. Davidon. Variable metric method for minimization. *SIAM J. Optim.*, 1(1):1–17, 1991.
- [3] O. DeGuchy, J. B. Erway, and R. F. Marcia. Compact representation of the full Broyden class of quasi-Newton updates. *ArXiv e-prints*, (arXiv:1705.08306), May 2017.
- [4] L. C. W. Dixon. Quasi-Newton algorithms generate identical points. *Math. Programming*, 2:383–387, 1972.
- [5] J. B. Erway and R. F. Marcia. On efficiently computing the eigenvalues of limited-memory quasi-Newton matrices. *SIAM J. Matrix Anal. Appl.*, 36(3):1338–1359, 2015.
- [6] J. B. Erway and R. F. Marcia. On solving large-scale limited-memory quasi-Newton equations. *Linear Algebra Appl.*, 515:196–225, 2017.
- [7] R. Fletcher. *Practical methods of optimization. Vol. 2*. John Wiley & Sons, Ltd., Chichester, 1981. Constrained optimization, A Wiley-Interscience Publication.
- [8] R. Fletcher. An overview of unconstrained optimization. In *Algorithms for continuous optimization (Il Ciocco, 1993)*, volume 434 of *NATO Adv. Sci. Inst. Ser. C Math. Phys. Sci.*, pages 109–143. Kluwer Acad. Publ., Dordrecht, 1994.

-
- [9] R. Fletcher and M. J. D. Powell. A rapidly convergent descent method for minimization. *Comput. J.*, 6:163–168, 1963/1964.
 - [10] A. Forsgren and T. Odland. On exact linesearch quasi-Newton methods for minimizing a quadratic function. *Comput. Optim. Appl.*, 69(1):225–241, 2018.
 - [11] P. E. Gill and M. W. Leonard. Reduced-Hessian quasi-Newton methods for unconstrained optimization. *SIAM J. Optim.*, 12(1):209–237, 2001.
 - [12] P. E. Gill and M. W. Leonard. Limited-memory reduced-Hessian methods for large-scale unconstrained optimization. *SIAM J. Optim.*, 14(2):380–401, 2003.
 - [13] G. H. Golub and C. F. Van Loan. *Matrix computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, fourth edition, 2013.
 - [14] M. R. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *J. Research Nat. Bur. Standards*, 49:409–436 (1953), 1952.
 - [15] H. Y. Huang. Unified approach to quadratically convergent algorithms for function minimization. *J. Optimization Theory Appl.*, 5:405–423, 1970.
 - [16] J. Nocedal. Updating quasi-Newton matrices with limited storage. *Math. Comp.*, 35(151):773–782, 1980.
 - [17] J. Nocedal and S. J. Wright. *Numerical optimization*. Springer Series in Operations Research and Financial Engineering. Springer, New York, second edition, 2006.
 - [18] Y. Saad. *Iterative methods for sparse linear systems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, second edition, 2003.
 - [19] D. F. Shanno. Conjugate gradient methods with inexact searches. *Math. Oper. Res.*, 3(3):244–256, 1978.