J Optim Theory Appl (2017) 172:206–221
DOI 10.1007/s10957-016-1013-z

An Approach for Analyzing the Global Rate of Convergence of Quasi-Newton and Truncated-Newton Methods

T. L. Jensen[1] · M. Diehl[2]

Received: 8 April 2016 / Accepted: 14 September 2016 / Published online: 23 September 2016

Abstract Quasi-Newton and truncated-Newton methods are popular methods in optimization and are traditionally seen as useful alternatives to the gradient and Newton methods. Throughout the literature, results are found that link quasi-Newton methods to certain first-order methods under various assumptions. We offer a simple proof to show that a range of quasi-Newton methods are first-order methods in the definition of Nesterov. Further, we define a class of generalized first-order methods and show that the truncated-Newton method is a generalized first-order method and that first-order methods and generalized first-order methods share the same worst-case convergence rates. Further, we extend the complexity analysis for smooth strongly convex problems to finite dimensions. An implication of these results is that in a worst-case scenario, the local superlinear or faster convergence rates of quasi-Newton and truncated-Newton methods cannot be effective unless the number of iterations exceeds half the size of the problem dimension.

Keywords Quasi/truncated-Newton methods · First-order methods · Complexity analysis

Mathematics Subject Classification 90C53 · 49M15 · 47N10

Communicated by Ilio Galligani.

✉ T. L. Jensen
tlj@es.aau.dk

M. Diehl
moritz.diehl@imtek.uni-freiburg.de

[1] Department of Electronic Systems, Aalborg University, Aalborg, Denmark

[2] Department of Microsystems Engineering (IMTEK) and Department of Mathematics, University of Freiburg, Freiburg, Germany

1 Introduction

Quasi-Newton and truncated-Newton methods have been widely used since their origin. Arguments for their use are that their convergence is faster than that of the classical gradient method, due to curvature information and simple implementations and that, in the limited-memory form and when using truncated-Newton, they are suitable for large-scale optimization. These features have made the methods a textbook must-have [1–3]. In this paper, we will analyze some of the more common quasi-Newton methods, specifically quasi-Newton methods of the Broyden [4] and Huang [5] families, which include the well-known variants Broyden–Fletcher–Goldfarb–Shanno (BFGS) [6–9], symmetric rank 1 (SR1) [4,10–13] and Davidon–Fletcher–Powell (DFP) [14,15] (see also [16]). We will also consider the limited-memory BFGS (L-BFGS) method [17,18]. Local convergence of these algorithms is well studied, see [2,3] for an overview.

Many connections exist between quasi-Newton methods and first-order methods. The DFP method using exact line search generates the same iterations as the conjugate gradient method for quadratic functions [19] (see also [20, pp. 57] and [21, pp. 200–222]). A more general result is that the nonlinear Fletcher–Reeves conjugate gradient method is identical to the quasi-Newton Broyden method (with exact line search and initialized with the identity matrix) when applied to quadratic functions [2, The. 3.4.2]. These connections only apply for quadratic functions, but stronger properties and connections are also known. A memoryless BFGS with exact line search is equivalent to the nonlinear Polak–Ribiére conjugate gradient methods with exact line search [22] (note that the Hestenes-Stiefel and Polak–Ribiére nonlinear conjugate gradient methods are equivalent when utilizing exact line search [3, §7.2]). The DFP method with exact line search and initialized with the identity matrix is a nonlinear conjugate gradient method with exact line search [21, pp. 216–222]. Further, all quasi-Newton methods of the Broyden family are equivalent when equipped with exact line search [23]. Consequently, all the above statements can be extended to all methods of the Broyden family when exact line search is utilized (see also [2, pp. 64]).

Even though it seems clear that quasi-Newton methods and first-order methods are similar, it is not uncommon to encounter thoughts on the subject along the lines: “Two of the most popular methods for smooth large-scale optimization, the inexact- (or truncated-) Newton method and limited-memory BFGS method, are typically implemented so that the rate of convergence is only linear. They are good examples of algorithms that fall between first- and second-order methods” [24]. This statement raises two questions (i) Why is a limited-memory BFGS method not considered a first-order method? (ii) Can we give a more informative classification of truncated-Newton methods?

We can already partly answer question one since this is addressed in the convex programming complexity theory of Nemirovsky and Yudin [25]. Since the limited-memory BFGS utilizes information from a first-order oracle, it is a first-order method which implies certain worst-case convergence rates. The first-order definition and related worst-case convergence rates can be simplified for instructional purposes, as done by Nesterov [26]. However, the inclusion of quasi-Newton methods in the simplified and more accessible analysis of first-order methods by Nesterov is not addressed.

The second question on the classification of truncated-Newton methods is open for discussion. The truncated-Newton method implicitly utilizes second-order information. Customary classification would then denote the truncate-Newton method a second-order method since it requires a second-order oracle. However, the standard Newton method is also a second-order method. The worst-case convergence rates of the Newton and truncated-Newton methods are not the same and grouping these two substantially different methods into the same classification is dissatisfactory.

Contributions This paper elaborates on previous results and offers a more straightforward and accessible proof to show that a range of quasi-Newton methods, including the limited-memory BFGS method, are first-order methods in the definition of Nesterov [26]. Further, by defining a so-called generalized first-order method, we extend the analysis to include truncated-Newton methods as well. For the sake of completeness, we also consider complexity analysis of a class of smooth and strongly convex problems in finite dimensions. For a worst-case scenario, quasi-Newton and truncated-Newton methods applied to this class of problems show linear convergence rate for as many iterations as up to half the size of the problem dimension $k \leq \frac{1}{2}n$. Hence, problems exist for which local superlinear or faster convergence of quasi-Newton and truncated-Newton methods will not be effective unless $k > \frac{1}{2}n$.

The rest of the paper is organized as follows. Section 2 below describes the used definition of first-order methods. Section 3 describes quasi-Newton methods, and it is shown that a range of quasi-Newton methods are first-order methods in the definition of Nesterov. Section 4 describes a generalized first-order method and shows that the truncated-Newton method belongs to this class and that the worst-case convergence rate is the same as that of first-order methods. Section 5 contains the conclusions.

2 First-Order Methods

We denote $\mathbb{N} := \{0, 1, 2, \ldots\}$ the natural numbers including 0. A vector is denoted $x = [x^1, \ldots, x^n]^T \in \mathbb{R}^n$. The vector $e_i \in \mathbb{R}^n$ is the ith standard basis vector for $\mathbb{R}^n$. We will define the span of a set $X \subseteq \mathbb{R}^n$ as span $X := \{\sum_{i=1}^{|X|} c_i x_i \,:\, x_1, \ldots, x_{|X|} \in X; c_1, \ldots, c_{|X|} \in \mathbb{R}\}$. Note that this means that span $\{x_1, \ldots, x_m\} = \{\sum_{i=1}^{m} c_i x_i \,:\, c_1, \ldots, c_m \in \mathbb{R}\}$. Let X and Y be two sets with $X \subseteq \mathbb{R}^n$ and $Y \subseteq \mathbb{R}^n$, and then the (Minkowski) sum of sets is

$$X + Y := \{x + y \,:\, x \in X,\ y \in Y\}. \tag{1}$$

We consider the convex, unconstrained optimization problem

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad f(x) \tag{2}$$

with optimal objective $f(x^\star) = f^\star$. We assume that f is a convex function, continuously differentiable with Lipschitz continuous gradient constant L:

$$f(x) \leq f(y) + \nabla f(y)^T(x - y) + \tfrac{1}{2}L\|x - y\|_2^2, \qquad \forall x, y \in \mathbb{R}^n. \tag{3}$$

From time to time, we will strengthen our assumption on f and assume that f is also strongly convex with strong convexity parameter $\mu > 0$, such that

$$f(x) \geq f(y) + \nabla f(y)^T(x-y) + \tfrac{1}{2}\mu\|x-y\|_2^2, \qquad \forall x, y \in \mathbb{R}^n. \tag{4}$$

For twice differentiable functions, the requirements on μ and L are equivalent to the matrix inequality $\mu I \preceq \nabla^2 f(x) \preceq LI$. The condition number of a function is given by $Q = \frac{L}{\mu}$. Following Nesterov [26], we will define a first-order (black-box) method for the problem (2) as follows.

Definition 2.1 A first-order method for differentiable objectives is any iterative method that from an initial point x_0 generates $(x_i)_{i=1,\ldots,k+1}$ such that

$$\begin{aligned} &x_{k+1} \in x_0 + F_{k+1}, \text{ where} \\ &F_{k+1} = F_k + \operatorname{span}\{\nabla f(x_k)\} \end{aligned}$$

with $F_0 = \emptyset$ and $k \in \mathbb{N}$.

Strong results exist for such first-order methods [26]. These results date back to [25], but methods for achieving these bounds were first given in [27]. In Theorem 2.1, we reproduce an important result related to the first-order methods in Definition 2.1 as provided in [26].

Theorem 2.1 [26, Th 2.1.4 and Th 2.1.13] *For any $k \in \mathbb{N}$, $1 \leq k \leq \frac{1}{2}(n-1)$ and x_0, there exists a (quadratic) function $f : \mathbb{R}^n \mapsto \mathbb{R}$, which has a Lipschitz continuous gradient with constant L, such that any first-order method satisfies*

$$\frac{f(x_k) - f^\star}{\|x_0 - x^\star\|_2^2} \geq \frac{3L}{32(k+1)^2}.$$

There exists a function $f : \mathbb{R}^\infty \mapsto \mathbb{R}$ with a Lipschitz continuous gradient which is strongly convex with condition number Q, such that any first-order method satisfies

$$\frac{\|x_k - x^\star\|_2^2}{\|x_0 - x^\star\|_2^2} \geq \left(\frac{\sqrt{Q}-1}{\sqrt{Q}+1}\right)^{2k}.$$

We extend the above result for smooth and strongly convex problems from infinite-dimensional to finite-dimensional problems.

Theorem 2.2 *For any $k \in \mathbb{N}$, $1 \leq k \leq \frac{1}{2}n$ and $x_0 \in \mathbb{R}^n$, there exists a function $f : \mathbb{R}^n \mapsto \mathbb{R}$ with a Lipschitz continuous gradient which is strongly convex with condition number $Q \geq 8$, such that any first-order method satisfies*

$$\frac{\|x_k - x^\star\|_2^2}{\|x_0 - x^\star\|_2^2} \geq \frac{1}{2}\left(\frac{\sqrt{Q}/\beta-1}{\sqrt{Q}/\beta+1}\right)^{2k}$$

with constant $\beta = 1.1$.

Proof See "Appendix 1". □

This means that for certain problems, the convergence rate of any first-order method cannot be faster than linear. Since some methods achieve these bounds, it is known the latter are tight (up to a constant) [26]. Several other variants have been published [28–32], see also the overview [33].

3 Quasi-Newton Methods

For the (line search) quasi-Newton methods, we select the iterations as follows

$$x_{k+1} = x_k - t_k H_k \nabla f(x_k), \qquad k = 0, 1, \dots \tag{5}$$

where x_0 and H_0 are provided initializations. Consequently, several methods for selecting the step sizes t_k and the approximations of the inverse Hessian H_k exist. Often H_{k+1} is built as a function of H_k and the differences

$$y_k = \nabla f(x_{k+1}) - \nabla f(x_k), \quad s_k = x_{k+1} - x_k. \tag{6}$$

The following lemma connects the line search quasi-Newton methods to first-order methods in a straightforward manner.

Lemma 3.1 *Any method of the form*

$$x_{k+1} = x_k - t_k H_k \nabla f(x_k), \qquad k = 0, 1, \dots$$

where $t_k \in \mathbb{R}$, $H_0 = \alpha I$, $\alpha \in \mathbb{R}, \alpha \neq 0$ *and*

$$H_{k+1} z \in \operatorname{span}\{H_k z, x_{k+1} - x_k, H_k(\nabla f(x_{k+1}) - \nabla f(x_k))\} \, \forall z \in \mathbb{R}^n$$

is a first-order method.

Proof The first iteration $k = 0$ is a special case; this is verified by

$$x_1 = x_0 - t_0 H_0 \nabla f(x_0) = x_0 - t_0 \alpha \nabla f(x_0) \in x_0 + \operatorname{span}\{\nabla f(x_0)\}.$$

The iterations $k = 1, 2, \dots$ will be shown by induction of the statement:

$$\forall\, k = 1, 2, \dots, X(k) : \begin{cases} H_k z \in \operatorname{span}\{z, \nabla f(x_0), \nabla f(x_1), \dots, \nabla f(x_k)\} \, \forall z \in \mathbb{R}^n \\ x_{k+1} \in x_0 + \operatorname{span}\{\nabla f(x_0), \nabla f(x_1), \dots, \nabla f(x_k)\} \end{cases}$$

Induction start for $k = 1$ we have

$$\begin{aligned} H_1 z &\in \operatorname{span}\{H_0 z, x_1 - x_0, H_0(\nabla f(x_1) - \nabla f(x_0))\} \\ &\in \operatorname{span}\{\alpha z, -t_0 \alpha \nabla f(x_0), \alpha(\nabla f(x_1) - \nabla f(x_0))\} \\ &\in \operatorname{span}\{z, \nabla f(x_0), \nabla f(x_1)\} \, \forall z \in \mathbb{R}^n \end{aligned}$$

and

$$
\begin{aligned}
x_2 = x_1 - t_1 H_1 \nabla f(x_1) &= x_0 - t_0 H_0 \nabla f(x_0) - t_1 H_1 \nabla f(x_1) \\
&\in x_0 + \operatorname{span}\{\nabla f(x_0), \nabla f(x_1)\}
\end{aligned}
$$

so $X(1)$ holds. *Induction step.* Assume $X(k)$ holds. We have

$$
\begin{aligned}
H_{k+1} z &\in \operatorname{span}\{H_k z, x_{k+1} - x_k, H_k(\nabla f(x_{k+1}) - \nabla f(x_k))\} \\
&\in \operatorname{span}\{H_k z, H_k \nabla f(x_k), H_k \nabla f(x_{k+1}) - H_k \nabla f(x_k))\} \\
&\in \operatorname{span}\{H_k z, H_k \nabla f(x_k), H_k \nabla f(x_{k+1})\} \\
&\in \operatorname{span}\{z, \nabla f(x_0), \ldots, \nabla f(x_k), \nabla f(x_{k+1})\} \ \forall z \in \mathbb{R}^n,
\end{aligned} \tag{7}
$$

where we have assumed that $X(k)$ holds. Equation (7) is the first part of $X(k+1)$. For the iterators, we then have

$$
\begin{aligned}
x_{k+2} &= x_{k+1} - t_{k+1} H_{k+1} \nabla f(x_{k+1}) \\
&\in x_0 + \operatorname{span}\{\nabla f(x_0), \ldots, \nabla f(x_k)\} + \operatorname{span}\{\nabla f(x_0), \ldots, \nabla f(x_{k+1})\} \\
&\in x_0 + \operatorname{span}\{\nabla f(x_0), \ldots, \nabla f(x_{k+1})\},
\end{aligned} \tag{8}
$$

where we have used $X(k)$ and (7). Equation (8) is the second part of $X(k+1)$; consequently, $X(k+1)$ holds. The iterations then satisfy $x_{k+1} \in x_0 + F_{k+1}$ and the method is a first-order method. □

We will now show that the quasi-Newton methods in the Broyden and Huang family, and the L-BFGS methods are first-order methods. This implies that they share the worst-case complexity bounds of any first-order method given in Theorems 2.1 and 2.2. Note that these methods are not necessarily optimal, in the sense that (up to a constant) they achieve the bounds in Theorems 2.1 and 2.2. As described in the introduction, these are known results [25]. Our reason for conducting this analysis is twofold: we believe that (i) the provided analysis is more intuitive and insightful (ii) it provides a logical background for analyzing truncated-Newton, taking a similar approach.

3.1 The Broyden Family

The (one parameter) Broyden family includes updates on the form

$$
H_{k+1} = H_k + \rho_k s_k s_k^T - \varrho_k H_k y_k y_k^T H_k + \frac{\eta_k}{\varrho_k} v_k v_k^T \tag{9}
$$

where η_k is the Broyden parameter and

$$
\rho_k = \frac{1}{y_k^T s_k}, \qquad \varrho_k = \frac{1}{y_k^T H_k y_k}, \qquad v_k = \rho_k s_k - \varrho_k H_k y_k.
$$

The Broyden family includes the well-known BFGS, DFP, and SR1 quasi-Newton methods as special cases with certain settings of η_k, specifically:

$$\begin{aligned} \text{BFGS}: &\quad \eta_k = 1, && (10)\\ \text{DFP}: &\quad \eta_k = 0 && (11)\\ \text{SR1}: &\quad \eta_k = \frac{1}{\rho_k y_k^T (s_k - H_k y_k)}. && (12) \end{aligned}$$

Corollary 3.1 *Any quasi-Newton method of the Broyden family with $\eta_k \in \mathbb{R}$, any step size rule and $H_0 = \alpha I$, $\alpha \in \mathbb{R}$, $\alpha \neq 0$ is a first-order method.*

Proof Using Eq. (9), we have:

$$\begin{aligned} H_{k+1}z &= H_k z + \rho_k s_k s_k^T z - \varrho_k H_k y_k y_k^T H_k z + \frac{\eta_k}{\varrho_k} v_k v_k^T z \\ &\in \operatorname{span}\{H_k z, s_k, H_k y_k, v_k\} \forall z \in \mathbb{R}^n \\ &\in \operatorname{span}\{H_k z, s_k, H_k y_k\} \forall z \in \mathbb{R}^n \\ &\in \operatorname{span}\{H_k z, x_{k+1} - x_k, H_k(\nabla f(x_{k+1}) - \nabla f(x_k))\} \forall z \in \mathbb{R}^n \end{aligned}$$

and the results then follow directly from Lemma 3.1. □

3.2 The Huang Family

The Huang family includes updates on the form [5]:

$$K_{k+1} = H_k + \psi_k \sigma_k s_k (\phi_k s_k + \varphi_k H_k^T y_k)^T - \varsigma_k H_k y_k (\theta_k s_k + \vartheta_k H_k^T y_k)^T \tag{13}$$

where $\psi_k, \phi_k, \varphi_k, \theta_k, \vartheta_k \in \mathbb{R}$ and

$$\sigma_k = \frac{1}{(\phi_k s_k + \varphi_k H_k^T y_k)^T y_k}, \qquad \varsigma_k = \frac{1}{(\theta_k s_k + \vartheta_k H_k^T y_k)^T y_k}$$

Note that the Huang family includes the DFP methods with the selection $\psi_k = 1, \phi_k = 1, \varphi_k = 0, \theta_k = 0, \vartheta_k = 1$, but also a range of other methods, see e.g. [5], including non-symmetric forms.

Corollary 3.2 *Any quasi-Newton method of the Huang family, any step size rule and $H_0 = \alpha I$, $\alpha \in \mathbb{R}$, $\alpha \neq 0$ is a first-order method.*

Proof Using Eq. (13), we have:

$$\begin{aligned} H_{k+1}z &= H_k z + \psi_k \sigma_k s_k (\phi_k s_k + \varphi H_k^T y_k)^T z - \varsigma_k H_k y_k (\theta_k s_k + \vartheta_k H_k^T y_k)^T z \\ &\in \operatorname{span}\{H_k z, s_k, H_k y_k\} \forall z \in \mathbb{R}^n \\ &\in \operatorname{span}\{H_k z, x_{k+1} - x_k, H_k(\nabla f(x_{k+1}) - \nabla f(x_k))\} \forall z \in \mathbb{R}^n \end{aligned}$$

and the results then follow directly from Lemma 3.1. □

3.3 Limited-Memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS)

For L-BFGS, we only build a Hessian approximation $\bar{H}_k$ out of the m most recent gradients [3]

$$\begin{aligned}\bar{H}_k &= (V_{k-1}^T \dots V_{k-m}^T)\bar{H}_k^0(V_{k-m} \dots V_{k-1})\\ &\quad + \rho_{k-m}(V_{k-1}^T \dots V_{k-m+1}^T)s_{k-m}s_{k-m}^T(V_{k-m+1} \dots V_{k-1})\\ &\quad + \quad \vdots\\ &\quad + \rho_{k-1}s_{k-1}s_{k-1}^T\end{aligned}$$

where $V_k = I - \rho_k y_k s_k^T$. From this, it is clear that for $k \leq m$ and $\bar{H}_k^0 = H_0$ the approximations of the Hessian for the BFGS and L-BFGS methods are identical $\bar{H}_k = H_k$. This motivates that L-BFGS is also a first-order method.

Corollary 3.3 *The L-BFGS method with any step size rule and* $\bar{H}_k^0 = \gamma_k I$, $\gamma_k \in \mathbb{R}$, $\gamma_k \neq 0$ *is a first-order method*

Proof See “Appendix 2”. □

4 Generalized First-Order Methods and the Truncated-Newton Method

In this section, we will show that the truncated-Newton method is very similar as regards behavior as a first-order method. However, the truncated-Newton method is not a first-order method following Definition 2.1. Instead, when comparing a first-order method with a truncated-Newton method, we must assume some properties of the underlying iterative solver of the Newton equation

$$\nabla f(x_k) + \nabla^2 f(x_k)(x - x_k) = 0$$

which is usually achieved by an underlying lower level algorithm that is often a first-order methods itself, e.g. the conjugate gradient method. For a fixed higher level index k, these iterations start with $x_{k,0} := x_k$ (warm-start) and $F'_{k,0} = \emptyset$, and subsequently iterate for $i \in \mathbb{N}$

$$\begin{aligned}&x_{k,i+1} \in x_k + F'_{k,i+1}, \quad \text{with}\\ &F'_{k,i+1} = F'_{k,i} + \operatorname{span}\{\nabla f(x_k) + \nabla^2 f(x_k)(x_{k,i} - x_k)\}.\end{aligned}$$

After the lower level iterations have been completed, e.g. at iterate $i^\star + 1$, the new higher level iterate is set to

$$x_{k+1} = x_k + t_k(x_{k,i^\star+1} - x_k) \in x_k + F'_{k,i^\star+1}$$

where t_k is obtained by a line search. By renumbering all iterates so that the lower and higher level iterations become part of the same sequence, we might express the last equation as

$$x_{k+i^\star+1} \in x_k + F'_{k+i^\star+1}$$

where the iterates and subspaces after x_k and $F'_k = \emptyset$ are generated following

$$\begin{aligned} &x_{k+i+1} \in x_k + F'_{k+i+1}, \quad \text{with} \\ &F'_{k+i+1} = F'_{k+i} + \operatorname{span}\{\nabla f(x_k) + \nabla^2 f(x_k)(x_{k+i} - x_k)\}. \end{aligned}$$

We see that the main differences of a truncated-Newton method compared to a standard first-order method are (i) that the subspace is reset at the beginning of the lower level iterations (here, at index k) and (ii) that the exact gradient evaluations are replaced by a first-order Taylor series (here evaluated at x_k). This motivates the question of how to define a generalization of first-order methods that is also applicable to the class of truncated-Newton methods. In this definition, we will introduce a higher degree of freedom, but make sure that the truncated-Newton methods are contained. One way to formulate the generalized first-order subspaces could be

$$\bar{F}'_{k+1} = \bar{F}'_k + \operatorname{span}\{\nabla f(x_i) + \nabla^2 f(x_i)\,(x_k - x_i) \;:\; 0 \le i \le k\}.$$

This definition would include all truncated-Newton methods that use first-order methods in their lower level iterations, together with standard first-order methods, and many more methods as well. Note that the dimensions of the subspaces do not only increase by one in each iteration as in the standard first-order method, but possibly by $k+1$. Here, we can disregard the fact that in practice, we will not be able to generate these large dimensional subspaces. In this paper, we decide to generalize the subspace generation further, as follows.

Definition 4.1 A generalized first-order method for a problem with a twice continuously differentiable objective is any iterative algorithm that selects

$$\begin{aligned} &x_{k+1} \in x_0 + S_{k+1}, \quad \text{where} \\ &S_{k+1} = S_k + \operatorname{span}\{\nabla f(p) + \nabla^2 f(p)\, d \;:\; p \in x_0 + S_k, d \in S_k\} \end{aligned} \tag{14}$$

with $S_0 = \{0\}$ and $k \in \mathbb{N}$.

Remark 4.1 Note that here we use $S_0 = \{0\}$ and not $S_0 = \emptyset$ to ensure that the set-builder notation in (14) is well defined for the first iteration.

At this point, a short discussion of the term “generalized first-order methods” seems appropriate. The model in Definition 4.1 requires second-order information and using standard terminology for such methods, see for instance [25], these should be denoted second-order methods. However, one motivation for using truncated-Newton methods is the implicit usage of the second-order information in which the Hessian matrix $\nabla^2 f(p) \in \mathbb{R}^{n\times n}$ is never formed but only touched upon via the matrix-vector product $\nabla^2 f(p)d \in \mathbb{R}^n$. Further, whether or not the Hessian matrix is used explicitly or implicitly, a truncated-Newton method is more similar to first-order methods than to second-order methods from a global convergence perspective, as we will show in the following section. Hence, we believe that the term “generalized first-order methods” provides a meaningful description of these methods.

This also motivates the following definition of a *generalized first-order oracle* (deterministic, black box):

$$\text{input: } (x, z) \quad \text{return: } (f(x), \nabla f(x), \nabla^2 f(x)z)$$

This means that with a two-tuple input (x, z), the oracle returns the three-tuple output $(f(x), \nabla f(x), \nabla^2 f(x)z)$. In this case, the second-order information is not directly available as in a second-order oracle but only $\nabla^2 f(x)z$. This results in a stronger and more precise description of truncated-Newton methods than saying that they fall between first- and second-order methods as suggested in [24] and discussed in the introduction.

4.1 Equivalence of Generalized and Standard First-Order Methods for Quadratic Functions

For a quadratic function, $f(x)$, the spaces of the usual first-order method and the generalized first-order method coincide, $F_k = S_k$.

Lemma 4.1 *For quadratic functions f, $S_k = F_k$, $k \geq 1$.*

Proof As f is quadratic, ∇f is linear, and thus coincides with its first-order Taylor series, i.e. $\nabla f(p) + \nabla^2 f(p)\, d = \nabla f(p + d)$. To simplify the notation, denote the constant Hessian $\nabla f^2(x) = P$ for all $x \in \mathbb{R}^n$. For $F_1 = \text{span}\{\nabla f(x_0)\}$, and for quadratic problems $S_1 = 0 + \text{span}\{\nabla f(x_0) + P \cdot 0\} = \text{span}\{\nabla f(x_0)\}$. For the standard first-order method $k \geq 1$, the following holds

$$\begin{aligned} F_{k+1} &= F_k + \text{span}\{\nabla f(x_k)\} \\ &= F_k + \text{span}\{\nabla f(x_0) + P(x_k - x_0)\} \\ &= F_k + \text{span}\{P(x_k - x_0)\} \\ &= F_k + \text{span}\{Px \ : \ x \in F_k\}. \end{aligned} \tag{15}$$

Conversely, for the generalized first-order method applied to a quadratic function for $k \geq 1$ it holds that

$$\begin{aligned} S_{k+1} &= S_k + \text{span}\{\nabla f(p) + \nabla^2 f(p)\, d \ : \ p \in x_0 + S_k, d \in S_k\} \\ &= S_k + \text{span}\{\nabla f(p + d) \ : \ p \in x_0 + S_k, d \in S_k\} \\ &= S_k + \text{span}\{\nabla f(x_0 + d) \ : \ d \in S_k\} \\ &= S_k + \text{span}\{\nabla f(x_0) + Pd \ : \ d \in S_k\} \\ &= S_k + \text{span}\{Pd \ : \ d \in S_k\}\,. \end{aligned} \tag{16}$$

Since $S_1 = F_1$ and (15)–(16) are the same recursions, $S_k = F_k,\ \forall k \geq 1$. □

With the Lemma 4.1 at hand, we can now present an important result which motivates the name “generalized first-order methods”.

Theorem 4.1 *Theorems 2.1 and 2.2 holds for generalized first-order methods.*

Proof Generalized first-order methods and first-order methods are equivalent for quadratic problems following Lemma 4.1. Hence, the results in Theorems 2.1 and 2.2 follow directly (see also [26, The. 2.1.7 and The. 2.1.13]). □

This means that the convergence for generalized first-order methods is the same as for first-order methods, i.e., in a worst-case scenario, they are sub-linear for smooth problems and linear for smooth and strongly convex problems, including the finite-dimensional case. This implies that from a global convergence perspective, a generalized first-order method has more in common with first-order methods compared to second-order methods.

5 Conclusions

In this paper, we have tried to answer two important questions. First, why are limited-memory BFGS methods not considered first-order methods? We believe that this is connected to the accessibility of the analysis provided in the work [25]. To this end, we have given a more straightforward analysis, using the definition of first-order methods as provided in [26]. The second question is whether it is possible to give a better description of the classification of truncated-Newton methods? We have described a class of methods named generalized first-order methods to which the truncated-Newton methods belong and have shown that the worst-case global convergence rate for generalized first-order methods is the same as that applying to for first-order methods. Thus, in a worst-case scenario, quasi-Newton and truncated-Newton methods are lower-bounded by a linear rate of convergence for $k \leq \frac{1}{2}n$ according to Theorem 2.2 (see also the comment [26, pp. 41]). In a worst-case scenario, a better convergence, such as superlinear, can only be effective for $k > \frac{1}{2}n$. Since the number of iterations k for this bound depends on the dimensionality of the problem n, this bound has the strongest implication for large-scale optimization.

Acknowledgements The authors would like to thank the anonymous reviewers of an earlier submission of this manuscript. The first author thanks Lieven Vandenberghe for interesting discussions on the subject. The second author thanks Michel Baes and Mike Powell for an inspirational and controversial discussion on the merits of global iteration complexity versus locally fast convergence rates in October 2007 at OPTEC in Leuven. *Grants* The work of T. L. Jensen was supported by The Danish Council for Strategic Research under Grant No. 09-067056 and The Danish Council for Independent Research under Grant No. 4005-00122. The work of M. Diehl was supported by the EU via FP7-TEMPO (MCITN-607957), ERC HIGHWIND (259166), and H2020-ITN AWESCO (642682).

Appendix 1: Proof of Theorem 2.2

Proof We will follow the approach [26, The. 2.1.13] but for finite-dimensional problems (this is more complicated as indicated in [26, p. 66]). Consider the problem instance

$$\begin{aligned}&\underset{x\in\mathbb{R}^n}{\text{minimize}} \quad f(x) = \tfrac{1}{2}x^T P x - c^T x\\ &\text{with} \quad P = \begin{bmatrix} A & 0\\ 0 & I\end{bmatrix} \in \mathbb{R}^{n\times n}, \quad A = \begin{bmatrix} p & 1 & 0 & 0 & \cdots & 0\\ 1 & p & 1 & 0 & \cdots & 0\\ 0 & 1 & p & 1 & & \vdots\\ \vdots & & \ddots & \ddots & \ddots & 0\\ 0 & \cdots & 0 & 1 & p & 1\\ 0 & \cdots & 0 & 0 & 1 & q\end{bmatrix} \in \mathbb{R}^{\tilde{n}\times\tilde{n}},\\ &\quad c = e_1 \in \mathbb{R}^{n\times n}\end{aligned}$$

where $\tilde{n} \le n$ and the initialization $x_0 = 0$. We could also have generated the problem $\bar{f}(\bar{x}) = f(\bar{x} + \bar{x}_0)$ and initialized it at $\bar{x}_0$ since this is just a shift of sequences of any first-order method. To see this, let $k \ge 1$ and $\bar{x} \in \bar{x}_0 + \bar{F}_k$ be an allowed point for a first-order method applied to a problem with objective $\bar{f}$ using $\bar{x}_0$ as initialization. Let $x = \bar{x} + \bar{x}_0$, then

$$\begin{aligned}\nabla\bar{f}(\bar{x}) = \nabla\bar{f}(x - \bar{x}_0) = \nabla f(x - \bar{x}_0 + \bar{x}_0) = \nabla f(x),\\ \bar{f}(\bar{x}) = \bar{f}(x - \bar{x}_0) = f(x - \bar{x}_0 + \bar{x}_0) = f(x).\end{aligned}$$

Consequently $x \in F_k = \bar{F}_k$ and we can simply assume $x_0 = 0$ in the following.

We select $p = 2 + \mu$ and $q = (p + \sqrt{p^2 - 4})/2$ with the bounds $1 \le q \le p$. Then:

$$A = \tilde{A} + (q - p + 1)e_{\tilde{n}}e_{\tilde{n}}^T, \quad \tilde{A} = \begin{bmatrix} p & 1 & 0 & 0 & \cdots & 0\\ 1 & p & 1 & 0 & \cdots & 0\\ 0 & 1 & p & 1 & & \vdots\\ \vdots & & \ddots & \ddots & \ddots & 0\\ 0 & \cdots & 0 & 1 & p & 1\\ 0 & \cdots & 0 & 0 & 1 & p-1\end{bmatrix}$$

The value $q - p + 1 \ge 0$ for $\mu \ge 0$. The eigenvalues of the matrix $\tilde{A}$ are given as [34]

$$\lambda_i = p + 2\cos\left(\frac{2i\pi}{2\tilde{n}+1}\right), \quad i = 1, \ldots, \tilde{n}.$$

The smallest and largest eigenvalues of A are then bounded as

$$\begin{aligned}\lambda_{\min}(A) &\ge p + 2\cos\left(\frac{2\tilde{n}\pi}{2\tilde{n}+1}\right) \ge p - 2 = \mu,\\ \lambda_{\max}(A) &\le p + 2\cos\left(\frac{2\pi}{2\tilde{n}+1}\right) + q - p + 1 \le q + 3 \le p + 3 = L.\end{aligned}$$

With $\mu \le 1$, we have $\lambda_{\min}(P) = \lambda_{\min}(A)$ and $\lambda_{\max}(P) = \lambda_{\max}(A)$. The condition number is given by $Q = \frac{L}{\mu} = \frac{p+3}{p-2} = \frac{5+\mu}{\mu} \ge 6$, and the solution is given as

$x^\star = P^{-1}e_1$. The inverse A^{-1} can be found in [34], and the ith entry of the solution is then

$$(x^\star)^i = \begin{cases} \frac{(-1)^{i+1}s_{\tilde{n}-i}}{qr_{\tilde{n}-1}-r_{\tilde{n}-2}} & , \; i = 1, 2, \ldots, \tilde{n} \\ 0 & , \; i = \tilde{n}+1, \ldots n \end{cases}$$

where

$$r_0 = 1, \; r_1 = p, \; r_i = pr_{i-1} - r_{i-2}, \quad s_0 = 1, \; s_1 = q, \; s_i = ps_{i-1} - s_{i-2}, \quad i = 2, \ldots, \tilde{n}-1$$

Since q is a root of the second-order polynomial $y^2 - py + 1$, we have $s_i = q^i$, $\forall i \geq 0$. Using $Q = \frac{p+3}{p-2} \Leftrightarrow p = 2\frac{Q+\frac{3}{2}}{Q-1}$ and then

$$q = \frac{p + \sqrt{p^2-4}}{2} = \frac{Q + \frac{3}{2} + \sqrt{\frac{1}{2} + 5Q}}{Q-1} \leq \frac{Q + \beta\sqrt{Q}}{Q - \beta\sqrt{Q}} \tag{17}$$

A simple calculation of β in (17) can for instance be $\beta = \left.\frac{\frac{3}{2}+\sqrt{\frac{1}{2}+5Q}}{\sqrt{Q}}\right|_{Q=8} \simeq 2.78$, which is sufficient for any $Q \geq 8$. However, solving the nonlinear equation yields that $\beta = 1.1$ is also sufficient for any $Q \geq 8$. Since $\nabla f(x) = Px - c$, the set F_k expands as:

$$F_{k+1} = F_k + \text{span}\{\nabla f(x_k)\} = F_k + \text{span}\{Px_k - c\}.$$

Since P is tridiagonal and $x_0 = 0$, we have

$$\begin{aligned} F_1 &= \emptyset + \text{span}\{Px_k - c \; : \; x_k \in 0 + \emptyset\} = \text{span}\{e_1\} \\ F_2 &= \text{span}\{e_1\} + \text{span}\{Px_1 - c \; : \; x_1 \in F_1\} = \text{span}\{e_1, e_2\} \\ &\vdots \\ F_k &= \text{span}\{e_1, e_2, \ldots, e_k\}. \end{aligned}$$

Considering the relative convergence

$$\frac{\|x_k - x^\star\|_2^2}{\|x_0 - x^\star\|_2^2} = \frac{\|x_k - x^\star\|_2^2}{\|x^\star\|_2^2} \geq \frac{\sum_{i=k+1}^{\tilde{n}} s_{\tilde{n}-i}^2}{\sum_{i=1}^{\tilde{n}} s_{\tilde{n}-i}^2} = \frac{\sum_{i=0}^{\tilde{n}-k-1} q^{2i}}{\sum_{i=0}^{\tilde{n}-1} q^{2i}} = \frac{1 - q^{2(\tilde{n}-k)}}{1 - q^{2\tilde{n}}} = \frac{q^{2\tilde{n}}q^{-2k} - 1}{q^{2\tilde{n}} - 1}. \tag{18}$$

Fixing $\tilde{n} = 2k$, we have for $k = \frac{1}{2}\tilde{n} \leq \frac{1}{2}n$

$$\frac{\|x_k - x^\star\|_2^2}{\|x_0 - x^\star\|_2^2} \geq \frac{q^{2k}-1}{q^{4k}-1} = \frac{q^{2k}(q^{2k}-1)}{(q^{2k}-1)(q^{2k}+1)}q^{-2k} = \frac{q^{2k}}{(q^{2k}+1)}q^{-2k} \geq \tfrac{1}{2}q^{-2k}$$

and inserting (17) yields

$$\frac{\|x_k - x^\star\|_2^2}{\|x_0 - x^\star\|_2^2} \geq \frac{1}{2}\left(\frac{Q - \beta\sqrt{Q}}{Q + \beta\sqrt{Q}}\right)^{2k} = \frac{1}{2}\left(\frac{\sqrt{Q}/\beta - 1}{\sqrt{Q}/\beta + 1}\right)^{2k}.$$

□

Remark We note that it is possible to explicitly state a smaller β and hence a tighter bound, but we prefer to keep the explanation of β simple.

Algorithm 1: Evaluation of $r \leftarrow \bar{H}_k \nabla f(x_k)$.

```
input : $\{s_i, y_i, \rho_i\}$, $\bar{H}_k^0$, $\nabla f(x_k)$
output: $r$
$q \leftarrow \nabla f(x_k)$
for $i = k-1, k-2, \ldots, k-m$ do
    $\alpha_i \leftarrow \rho_i s_i^T q$
    $q \leftarrow q - \alpha_i y_i$
$r \leftarrow \bar{H}_k^0 q$
for $i = k-m, k-2, \ldots, k-1$ do
    $\beta \leftarrow \rho_i y_i^T r$
    $r \leftarrow r + s_i(\alpha_i - \beta)$
```

Appendix 2: Proof of Corollary 3.3

For this proof, we note that the multiplication $\bar{H}_k \nabla f(x_k)$ can be calculated efficiently via Algorithm 1 [3, Alg. 7.4]. From Algorithm 1, we obtain that with $\bar{H}_k^0 = \gamma_k I$ and using (6),

$$\begin{aligned}\bar{H}_k \nabla f(x_k) &\in \operatorname{span}\{\nabla f(x_k), y_{k-1}, \ldots, y_{k-m}, s_{k-1}, \ldots, s_{k-m}\}\\ &\in \operatorname{span}\{\nabla f(x_k), \nabla f(x_{k-1}), \ldots, \nabla f(x_{k-m}), \bar{H}_{k-1}\nabla f(x_{k-1}),\\ &\qquad \ldots, \bar{H}_{k-m}\nabla f(x_{k-m})\}\end{aligned}$$

and then recursively inserting

$$\begin{aligned}\bar{H}_k \nabla f(x_k) &\in \operatorname{span}\{\nabla f(x_k), \nabla f(x_{k-1}), \ldots, \nabla f(x_{k-m}), \nabla f(x_{k-m-1}),\\ &\qquad \bar{H}_{k-2}\nabla f(x_{k-2}), \ldots, \bar{H}_{k-m-1}\nabla f(x_{k-m-1})\}\\ &\in \operatorname{span}\{\nabla f(x_k), \ldots, \nabla f(x_0)\}.\end{aligned}$$

The iterations are then given as

$$\begin{aligned}x_{k+1} = x_k - t_k \bar{H}_k \nabla f(x_k) &= x_0 - \sum_{i=0}^{k} t_i \bar{H}_i \nabla f(x_i)\\ &\in x_0 + \operatorname{span}\{\nabla f(x_0), \nabla f(x_1), \ldots, \nabla f(x_k)\}\end{aligned}$$

and L-BFGS is a first-order method. □

References

1. Conn, A.R., Gould, N.I.M., Toint, P.L.: Trust-Region Methods. SIAM, Philadelphia (2000)
2. Fletcher, R.: Practical Methods of Optimization, 2nd edn. Wiley, Hoboken (2000)

3. Nocedal, J., Wright, S.J.: Numerical Optimization, 2nd edn. Springer Series in Operations Research, Berlin (2006)
4. Broyden, C.G.: Quasi-Newton methods and their application to function minimization. Math. Comput. **21**, 368–381 (1967)
5. Huang, H.Y.: Unified approach to quadratically convergent algorithms for function minimization. J. Optim. Theory Appl. **5**(6), 405–423 (1970)
6. Broyden, C.G.: The convergence of a class of double-rank minimization algorithms: 2. The new algorithm. IMA J. Appl. Math. **6**(3), 222–231 (1970)
7. Fletcher, R.: A new approach to variable metric algorithms. Comput. J. **13**(3), 317–322 (1970)
8. Goldfarb, D.: A family of variable-metric methods derived by variational means. Math. Comput. **24**(109), 23–26 (1970)
9. Shanno, D.F.: Conditioning of quasi-Newton methods for function minimization. Math. Comput. **24**(111), 647–656 (1970)
10. Davidon, W.C.: Variance algorithm for minimization. Comput. J. **10**(4), 406–410 (1968)
11. Fiacco, A.V., McCormick, G.P.: Nonlinear Programming. Wiley, New York (1968)
12. Murtagh, B.A., Sargent, R.W.H.: A constrained minimization method with quadratic convergence. In: Optimization. Academic Press, London (1969)
13. Wolfe, P.: Another variable metric method. Working paper (1968)
14. Davidon, W.C.: Variable metric method for minimization. Technical report. AEC Research and Development Report, ANL-5990 (revised) (1959)
15. Fletcher, R., Powell, M.J.D.: A rapidly convergent descent method for minimization. Comput. J. **6**(2), 163–168 (1963)
16. Hull, D.: On the huang class of variable metric methods. J. Optim. Theory Appl. **113**(1), 1–4 (2002)
17. Nocedal, J.: Updating quasi-Newton matrices with limited storage. Math. Comput. **5**, 773–782 (1980)
18. Liu, D.C., Nocedal, J.: On the limited-memory BFGS method for large scale optimization. Math. Program. **45**, 503–528 (1989)
19. Meyer, G.E.: Properties of the conjugate gradient and Davidon methods. J. Optim. Theory Appl. **2**(4), 209–219 (1968)
20. Polak, E.: Computational Method in Optimization. Academic Press, Cambridge (1971)
21. Ben-Tal, A., Nemirovski, A.: Lecture notes: optimization iii: Convex analysis, nonlinear programming theory, nonlinear programming algorithms. Georgia Institute of Technology, H. Milton Stewart School of Industrial and System Engineering. http://www2.isye.gatech.edu/~nemirovs/OPTIII_LectureNotes.pdf (2013)
22. Shanno, D.F.: Conjugate gradient methods with inexact searches. Math. Oper. Res. **3**(3), 244–256 (1978)
23. Dixon, L.C.W.: Quasi-newton algorithms generate identical points. Math. Program. **2**(1), 383–387 (1972)
24. Nocedal, J.: Finding the middle ground between first and second-order methods. OPTIMA 79— Mathematical Programming Society Newsletter, discussion column (2009)
25. Nemirovskii, A.S., Yudin, D.B.: Problem Complexity and Method Efficiency in Optimization. Wiley, New York (1983). First published in Russian (1979)
26. Nesterov, Y.: Introductory Lectures on Convex Optimization, A Basic Course. Kluwer Academic Publishers, Berlin (2004)
27. Nesterov, Y.: A method for unconstrained convex minimization problem with the rate of convergence $O(1/k^2)$. Dokl. AN SSSR (translated as Soviet Math. Docl.) **269**, 543–547 (1983)
28. Nesterov, Y.: On an approach to the construction of optimal methods of minimization of smooth convex functions. Ekonom. i. Mat. Mettody **24**, 509–517 (1988)
29. Nesterov, Y.: Smooth minimization of nonsmooth functions. Math. Program. Ser. A **103**, 127–152 (2005)
30. Nesterov, Y.: Gradient methods for minimizing composite objective function. Université Catholique de Louvain, Center for Operations Research and Econometrics (CORE). No 2007076, CORE discussion papers (2007)
31. Bioucas-Dias, J.M., Figueiredo, M.A.T.: A new TwIST: two-step iterative shrinkage/thresholding algorithms for image restoration. IEEE Trans. Image Process. **16**(12), 2992–3004 (2007)
32. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. Imag. Sci. **2**, 183–202 (2009)

33. Tseng, P.: On accelerated proximal gradient methods for convex-concave optimization. unpublished manuscript (2008)
34. Gregory, R.T., Karney, D.L.: A Collection of Matrices for Testing Computational Algorithms. Wiley, New York (1969)