Solutions of large-scale nonlinear systems VIA using quasi-Newton methods of order 1 and 2

By

Nouf Abdullah Hassan Alshehri

Department of Mathematics College of Science King Khalid University

Doi: 10.21608/ajsr.2024.390105

Receiving the search 2 – 9 - 2024
Acceptance of publication 30 – 9 - 2024

Nouf Abdullah Hassan Alshehri (2024). Solutions of large-scale nonlinear systems VIA using quasi-Newton methods of order1 and2**.*The Arab Journal of Scientific Research*,** AIESA, Egypt, 8(9), 1-22.

https://ajsr.journals.ekb.eg

Solutions of large-scale nonlinear systems VIA using quasi-Newton methods of order 1 and 2

Abstract:

This thesis studies numerical solutions of large-scale nonlinear systems using unconstrained optimization techniques. We focus on Quasi-Newton methods of order 1 and 2. We describe the methods, the corresponding algorithms, and their costs and convergence rates in order to allow a motivated methods choice. We limit ourselves to nonlinear linear systems resulting from finite elements discretization of boundary value problems.

Acknowledgements

Foremost, I thank ALLAH for his assistance to achieve my master degree in Mathematics. I would like to give my heartfelt appreciation to my parents for their love and encouraging me to achieve advanced degrees and who never hesitate to support me morally and financially.

Also, I would like to thank my husband who has encouraged me all the way and who helped me to finish that which I have started. I would like to thank my brothers and sisters for their support, love, and consistent praise.

I would like to express my deep gratitude to my supervisor Dr. Younes Chahlaoui for his constant support and guidance, who has been always generous during all phases of the research.I am also grateful to both examiners Dr. Awad AlMohy and Dr. Tamer Nabil for their careful reading of my manuscript and their many insightful comments and suggestions.

Finally, I want to express my appreciation to all the friends who have stood with me all along these years.

Chapter 1 Introduction

Nonlinear equations arise in essentially every branch of modern science, engineering, and math- ematics. However, in only a very few special cases it is possible to obtain useful solutions to nonlinear equations via analytical calculations [7]. As a result, many scientists resort to computational methods [3, 7, 17, 19, 22, 24].

Partial Differential Equations (PDEs) constitute by far the biggest source of large nonlinear systems (Example.3.2 and Example.3.3 in § 3). The typical way to solve such equations is to discretize them, i.e., to approximate them by equations that involve a finite number of unknowns. The nonlinear systems that arise from these discretizations are generally large and sparse, i.e., they have very few nonzero entries. Once the numerical approximation is made, the problem come to either find $\overline{x}$ such that $f(\overline{x}) = 0$, where f is a mapping from R^{n} to R^{n}, or to minimize an energy functional $\mathbf{J}$, from R^{n} to R.

Unconstrained optimization problems, § 2, consider the problem of minimizing an objective smooth function that depends on real variables with no restrictions on their values [7, 17].

$$\min_{x\in\mathrm{R}^{n}} f(x).$$

Unconstrained optimization problems arise directly in some applications, but they also arise indirectly from reformulations of constrained optimization problems. Often, it is practical toreplace the constraints of an optimization problem with penalized terms (Lagrange multipliers) in the objective function and to solve the problem as an unconstrained problem.

An important aspect of continuous optimization (constrained and unconstrained) is whether the functions are smooth, by which we mean that the second derivatives exist and are contin- uous. There has been extensive study and development of algorithms for the unconstrained optimization of smooth functions [3, 7, 17, 19, 22, 24]. At a high level, algorithms for uncon- strained minimization follow the following general structure:

1. Choose a starting point x_0.
2. Beginning from x_0, generate a sequence of iterates $(x_k)_{k=0}^{\infty}$ with non-increasing function f value until a solution point with sufficient accuracy is found or until no further progress can be made.

To generate the next iterate x_{k+1}, the algorithm uses information about the function at x_k and possibly earlier iterates (i.e., x_i, $i < k$). One of the most known method is the Newton method (§ 3.1).

Newton's method gives rise to a wide and important class of algorithms that require com- putation of the gradient vector

T

$\partial_1 f(x) \ \ldots \ \partial_n f(x)$ ı h

and the Hessian matrix $\nabla f(x) =$

$\nabla \ f(x) = [\partial_i \partial_j f(x)]_{\boldsymbol{i,j}}$. 2

Although the computation or approximation of the Hessian can be a time-consuming operation, this computation is justified for many problems [3, 7, 17, 19, 22, 24].

There are two fundamental strategies for moving from x_k to x_{k+1}: line search and trust region. Most algorithms follow one of these two strategies. The line-search method modifies the search direction to obtain another downhill, or descent, direction for f . It then tries different step lengths along this direction until it finds a step that not only decreases f but also achieves at least a small fraction of this direction's potential. The trust-region methods use the original quadratic model function, but they constrain the new iterate to stay in a local neighbourhood of the current iterate. To find the step, it is necessary to minimize the quadratic function subject to staying in this neighbourhood, which is generally ellipsoidal in shape. Line-search and trust-region techniques are suitable if the number of variables *n* is not too large, because the cost per iteration is of order n^3. Codes for problems with a large number of variables tend to use truncated Newton methods, which usually settle for an approximate minimizer of the quadratic model.

If computing the exact Hessian matrix is not practical, the same algorithms can be used with a reasonable approximation of the Hessian matrix [6, 10]. Two types of methods use approximations to the Hessian in place of the exact Hessian. One approach is to use difference approximations to the exact Hessian. Difference approximations exploit the fact that each column of the Hessian can be approximated by taking the difference between two instances of the gradient vector evaluated at two nearby points. For sparse Hessians, it is often possible to approximate many columns of the Hessian with a single gradient evaluation by choosing the evaluation points judiciously. Quasi-Newton Methods (§ 4 and § 5) build up an approximation to the Hessian by keeping track of the gradient differences along each step taken by the algorithm. Various conditions are imposed on the approximate Hessian. For example, its behaviour along the step just taken is forced to mimic the behaviour of the exact Hessian, and it is usually keptpositive definite.

The first chapter is dedicated to a general introduction and preliminaries on the subject. The second chapter introduces finite element systems and benchmark model. Sparse systems and Gauss method are the subjects of the third chapter. We study conjugate gradient method in the fourth chapter, then preconditioned conjugate gradient methods in the fifth chapter. We compare numerically all presented methods in the sixth chapter. Then, we summarize and present some conclusions and future work in the last chapter. All MATLAB codes are presentedin the last pages of the thesis.

Chapter 2
Optimization problem review

We refer to [3, 4, 5, 7, 9, 17, 18] for the relevant background on optimization.

2.1 Unconstrained optimization - link with solutions ofequations

Let $J : R^n \longrightarrow R$. We consider
$J\,(u) = \inf\{J\,(v)|\ v \in R^n\}$,
without constraint conditions on v. A necessary condition is given by

Theorem 2.1 *If* J *is differentiable at the point u (its minimum), then* $\nabla J\,(u) = 0$

Proof. Let $v \in R^n$ and $\theta > 0$. We have $\mathbf{J}\,(u + \theta v) \geq \mathbf{J}\,(u)$, so $\underline{\mathbf{J}\,(u + \theta v) - \mathbf{J}\,(u)}$ ≥ 0. θ
Then taking the limit at $\theta \longrightarrow 0^+$, we have $\langle \nabla J\,(u), v\rangle \geq 0$. If we take instead of v, $-v$ then
$\langle \nabla J\,(u), v\rangle \leq 0$ so $\nabla J\,(u) = 0$.
Sometimes, it is better to replace $\nabla J\,(u) = 0$ by $\langle \nabla J\,(u), v\rangle = 0,\ \forall v \in R^n$.
In general, $\nabla J\,(u) = 0$ is not sufficient to decide if u is an optimum. Take as an example:
2 2
$\mathbf{J}\,(u_1, u_2) = u_1 - u_2$, at (0, 0).
For a necessary and sufficient condition we have the following theorem.

Definition 2.2 A function $f : U \rightarrow\subset R^n \rightarrow R$ is called **convex** if:
$\forall x_1, x_2 \in U, \forall t \in [0, 1] : \quad f(tx_1 + (1 - t)x_2) \leq tf(x_1) + (1 - t)f(x_2)$.

Theorem 2.3 *Let* $J : R^n \longrightarrow R$ *be differentiable. We have the equivalence:*

$\mathbf{J}$ *is convex* $\Leftrightarrow$ $\mathbf{J}(v) \geq \mathbf{J}(u) + \langle \nabla \mathrm{J}(u), v - u \rangle, \quad \forall u, v \in \mathrm{R}^n.$

□

Proof. (=⇒) If J is convex, then for $\theta \in (0, 1)$, we have

$\mathbf{J}(u + \theta(v - u)) \leq (1 - \theta)\mathbf{J}(u) + \theta\mathbf{J}(v),$

$$\frac{\mathrm{J}(u + \theta(v - u)) - \mathrm{J}(u)}{\theta} \leq \mathrm{J}(v) - \mathrm{J}(u).$$

Take $\theta \longrightarrow 0^+$, we get

so

(⇐=)

$\langle \nabla \mathrm{J}(u), v - u \rangle \leq \mathrm{J}(v) - \mathrm{J}(u),$

$\mathrm{J}(v) \geq \mathrm{J}(u) + \langle \nabla \mathrm{J}(u), v - u \rangle.$

If $\mathrm{J}(v) \geq \mathrm{J}(u) + \langle \nabla \mathrm{J}(u), v - u \rangle$, take v and $u = v + \theta(u - v) = \theta u + (1 - \theta)v$. We have

$\mathrm{J}(v) \geq \mathrm{J}(v + \theta(u - v)) - \theta\langle \nabla \mathrm{J}(v + \theta(u - v)), u - v \rangle,$

$\mathrm{J}(u) \geq \mathrm{J}(v + \theta(u - v)) + (1 - \theta)\langle \nabla \mathrm{J}(v + \theta(u - v)), u - v \rangle.$

Multiply (2.1) by $(1 - \theta)$ and (2.2) by θ, then taking the sum we get that

$\theta\mathbf{J}(u) + (1 - \theta)\mathbf{J}(v) \geq \mathbf{J}(\theta u + (1 - \theta)v) = \mathbf{J}(v + \theta(u - v)).$

Chapter 3

Problems and basic methodspresentation

Partial Differential Equations (PDEs) constitute by far the biggest source of large nonlinear systems [3, 25]. The typical way to solve such equations is to discretize them, i.e., to approximate them by equations that involve a finite number of unknowns. The nonlinear systems that arise from these discretizations are generally large and sparse, i.e., they have very few nonzero entries. Once the numerical approximation is made, the problem come to either find $\overline{x}$ such that $f(\overline{x}) = 0$ where f is a mapping from R^n to R^n, or to minimize an energy functional J , from R^n to R.

We have the following theorem that link solving PDEs to optimization [1, 2].

Theorem 3.1 *[1, 2] Let a be a bilinear symmetric and positive form and L a linear form on a vectorial space* V_0*. Then the following statements are equivalent:*

- $a(u, v) = L(v), \quad \forall v \in V_0.$
- $\mathbf{J}\ (u) \leq \mathbf{J}\ (v),\ \forall v \in V$ *, where* $\mathbf{J}\ (u) = \frac{1}{\ } a(u,\ u) -$ $L(u).$

0 $\quad$ 2

□

Proof. Let $\lambda \in$ R and $v \in V_0$. We have

$a(u + \lambda v, u + \lambda v) = a(u, u) + 2\lambda a(u, v) + \lambda^2 a(v, v),$

$\lambda 2$

$\mathbf{J}\ (u + \lambda v) = \mathbf{J}\ (u) + \lambda\, [a(u, v) - L(v)] +$

If $a(u, v) = L(v),\ \forall v \in V_0$, then $a(v, v).$ —

$2\mathrm{J}\ (u + \lambda v) = \mathrm{J}\ (u) +_{\lambda} 2$

$_2\ a(v, v) \geq \mathrm{J}\ (u)$ —

as a is a positive form.

For the reciprocal, we have

$0 \leq (\mathbf{J}\ (u + \lambda v) - \mathbf{J}\ (u))\, \lambda^{-1} = [a(u, v) - L(v)] + \overset{\lambda}{-} a(v, v),$

then changing v to $-v$ we have 2

$0 \leq (\mathbf{J}\ (u - \lambda v) - \mathbf{J}\ (u))\, \lambda^{-1} = [a(u, -v) - L(-v)] + \overset{\lambda}{-} a(v, v).$

Taking the limit in both formulas for $\lambda \longrightarrow 0^+$ we have 2

$a(u, v) - L(v) \geq 0$ and $a(u, v) - L(v) \leq 0,$

which gives us the result.

We show two nonlinear examples.

Chapter 4

Broyden methods and rank-1 updating

It is known that the classical Newton method needs (n^2 + n) scalar functions evaluations and solving a linear system of

$O(n^3)$ elementary operations per iteration [3]. We will show that Broyden methods will diminish the convergence order from quadratic to superlinear. The idea of Broyden is to approximate the Jacobian matrices $\nabla f\,(x_k)$ by operators B_k, such that there
is no need to compute explicitly B^{-1} (we compute B^{-1} from B^{-1}), and such that B^{-1} is k k $k{-}1$
$k{+}1$
computed from B^{-1} using $O(n^2)$ elementary operations per iteration with ^{k}evaluation of f at x_k
and x_{k+1} only.

Precisely, let $f : \mathrm{R}^n \longrightarrow \mathrm{R}^n$ be differentiable (of class C^1) on an open convex D. Let $x \in$ D and s /= 0 such that $x^+ = x + s \in$ D. We will associate x to x_k and x^+ to x_{k+1} to obtain a good approximation of $\nabla f(x)$.
As ∇f is continues at x^+, for all $\epsilon > 0$, there exists $\delta > 0$ such that
$\|f(x) - f(x\ ) - \nabla f(x\ )(x - x\ )\| \le \epsilon\|x - x\ \|$,
which means $+$ $+$ $+$
$f(x)$ ' $f(x^+) + \nabla f(x^+)(x - x^+)$,
so, if $\bar{B}$ is an approximation of $\nabla f(x)$, it is logic to ask that $\bar{B}$ satisfies
$f(x) = f(x^+) + \bar{B}(x - x^+)$.
As a result:
$\bar{B}s = y = f(x^+) - f(x)$ where $s = x_{k+1} - x_k = x^+ - x$. (4.1)

If $n = 1$, the equation (4.1) completely determines $\bar{B}$ and the method will be:
$x \quad = x$
$— B^{-1}f(x) = x$
$—\frac{xk - xk{-}1}{\quad}f(x\ )$,

k

k

$k+1$ k k k

k $f(x$

$) - f(x\ k-1)$

then, $x = \underline{x_{k-1} f(x_k) - x_k f(x_{k-1})}$,

which is the ***Secant method***.

$k+1$

$f(x_k) - f(x)$

$(k-1)$

If $n > 1$, we can still say that the only new information on f is given by s. Broyden supposed that $\bar{B}$ will not differ of B by much on the orthogonal complement of s. This is equivalent to say that

$\bar{B} z = Bz$ if $\langle z, s\rangle = z^T s = s^T z = 0,$

$\bar{B} = B +$

$-$ $(y - Bs)s^T z$

$\underline{\qquad\qquad}(y - Bs)s^T$ $\underline{\qquad\qquad}$

$\langle s, s\rangle.$ (4.3)

Effectively, we have $Bz = Bz + \langle s, s\rangle = Bz$, because $s^T z = 0$ which is (4.2). Then,

we have $\bar{B} s = Bs \pm$ uniqueness.$(y \quad Bs)s^T s$

$\langle s, s\rangle$ $= Bs + (y$ ~~$- Bs) = y,$~~ which is (4.1). And consequently the

Moreover, we would like that from all matrices satisfying (4.1), $\bar{B}$ will be the closest to B.

We have:**Theorem 4.1** *Let* $B \in L(\mathbb{R}^n)$, $y \in \mathbb{R}^n$ *and* $s \in \mathbb{R}^n$, $s /= 0$. *The* $\bar{B}$*solution of given by* (4.3) *is the unique* $\min\{\|\hat{B} - B\|_F : \hat{B}s = y\}$.

□

Proof. First, we will show that $\bar{B}$ is a solution of the problem. Let $y = \hat{B}s$, we have

$$\|\bar{B} - B\|_F = \left\| \frac{(\hat{B} - B)ss^T}{\langle s, s\rangle} \right\|_F \le \frac{1}{\langle s, s\rangle} \|\hat{B} - B\|_F \|ss^T\|_F = \|\hat{B} - B\|_F ,$$

because of the exclusive property of the Frobenius norm

$$\|ss^T\|_F = \sum_i |s_i|^2 = s^T s = \langle s, s\rangle.$$

So $\bar{B}$ is the best matrix solution of
$\min\{\|\hat{B} - B\|_F : \hat{B}s = y\}$.
We need to show that $\bar{B}$ is unique. The mapping $f : L(R^n) \longrightarrow R$ defined by $f(A) = \|B - A\|_F$
is strictly convex on $L(R^n)$. Indeed, if $\theta \in (0, 1)$ and the matrices A_1, A_2 are not proportional
one another, we have

$f(\theta A_1 + (1 - \theta)A_2) = \|B - (\theta A_1 + (1 - \theta)A_2)\|_F$
$= \|\theta(B - A_1) + (1 - \theta)(B - A_2)\|_F$
$\le \theta\|B - A_1\|_F + (1 - \theta)\|B - A_2\|_F$

(we will have equality only if $A_1 = \alpha A_2$). So the set $\{\hat{B} \in L(R^n) : \hat{B}s = y\}$ is strictly convex.
If B_1 and B_2 are such that $B_1 s = y$, $B_2 s = y$ for all $\theta \in (0, 1)$, we have
$(\theta B_1 + (1 - \theta)B_2)s = \theta B_1 s + (1 - \theta) B_2 s = \theta y + (1 - \theta)y = y$.
\`‘y’x \`‘y’x

From the theory of functions, any function that is strictly convex defined on a convex set has at most one minimum [4], then we have the uniqueness.

The equation (4.1) is the key relation in developing Quasi-Newton methods (also called ***method with variable metric***) (these are all methods proposing Gradient approximation for zeros looking or Hessien approximation for optimum looking), which is why it is called ***Quasi-Newton equation*** [6, 10]. Moreover, it is used to develop a second class Broyden

methods called ***rank-2 methods*** (§ 5), where we impose to all matrices, to be used for $\overline{B}$, to

satisfy (4.1).

We define rank-1 updating method called ***Broyden method*** by

$x_{k+1} = x_k - B^{-1} f(x_k),$ $\quad_k \quad$ $k = 0, 1, 2, . . .$ (4.4)

where we recall that $B_k \in L(R^n)$ are generated via (4.3) like formula

$(y_k - B_k s_k)s^T$

$B_{k+1} = B_k + k$

$\langle s_k, s_k \rangle$

with $y_k = f(x_{k+1}) - f(x_k)$ and $s_k = x_{k+1} - x_k$. Naturally, we suppose that $s_k 0$ at each

iteration. Notice that x_0 and B_0 been given, Broyden method needs only n evaluations of scalar functions, $f(x_k)$, instead of n^2 for the classical Newton method.

From (4.4), a priori, we will need to compute the solution of

$B_k s_k = -f(x_k),$

for a cost of $O(n^3)$. But we can use the following result of Shermnan and Morrisson [26] (Lemma. 4.3). But, we need the following lemma.

Chapter 5

Construction of Quasi-Newton methods of rank-2 for unconstrainedoptimization problem

We mean by ***updating formula***, all formulas approximating the Jacobian (for zero finding) or the Hessian (for minimization problem) at iteration *k*, to another approximation at iteration

(*k* + 1), without explicitly computing the inverses (we $_{k+}$ $_{k}$ compute B^{-1} from B^{-1}).

In the following, we will suppose $f : \mathrm{R}^n \longrightarrow \mathrm{R}$ is twice differentiable on a convex D, and we have an approximation B of the Hessian $\nabla^2 f(x)$ for x in D and a direction s such that $x + s$ is still in D. The goal is to obtain an approximation $\bar{B}$ of $\nabla^2 f(x^+)$ at $x^+ = x + s$.

5.1 Symmetry and Quasi-Newton equation

From all what we did see, as the Hessian is symmetric, we desire that the updating formula preserve the symmetry. We would like:

We should have: B symmetric $\Longrightarrow$

$\bar{B}$ symmetric. (5.1) $\bar{B}s = y = \nabla f(x^+) - \nabla f(x)$ where $s = x^+ - x,$ (5.2)

is the Quasi-Newton equation associated to $F = \nabla f$. It is natural to ask if we can satisfy (5.1)and (5.2) by an updating formula of rank 1. To check that, we reconsider the formula:

$\bar{B} = B +$

$(y - Bs)c^T$

$\langle c, s\rangle,$ (5.3)

with $c \in \mathrm{R}^n$ such that $\langle c, s\rangle =$

0. If $\bar{B}$ satisfies (5.1) then $\bar{B} = B +$

$(y - Bs)(y - Bs)^T$

$\langle y - Bs, s\rangle$

, (5.4) is the unique possible solution provided that $\langle y - Bs, s\rangle = 0$.

Indeed, if we would like $\bar{B} = (\bar{B})^T$, B being symmetric, it is necessary and sufficient that:

$(y - Bs)c^T = c(y - Bs)^T$,

which implies $c = y - Bs$ provided that $\langle y - Bs, s\rangle = 0$, which gives (5.4). Moreover, if $y = Bs$

then $\bar{B} = B$ is the solution from (5.3) (it is natural to not change B if it is convenient). While

y Bs, but $\langle y - Bs, s\rangle = 0$ then there is no solution (because $\bar{B}$ is necessary of the form (5.4)

which has no sense).

The updating formula (5.3) is of the type

$\bar{B} = B + \alpha^{-1}vc^T$with $v = y - Bs$ and $\alpha = \langle c, s\rangle = 0$.

We say that this is a ***rank-1 updating*** formula because $\bar{B}$ is different from B by the matrix

$\alpha^{-1}vc^T$ that is rank 1 (we remark that $vc^T u = 0$ for all vectors u orthogonal to c).

The updating formula (5.4) is known as the ***rank-1 symmetric updating formula***. If

B is symmetric and nonsingular, let $H = B^{-1}$, then $\bar{B}$ is nonsingular and its inverse $\bar{H}$ by is given $\bar{H} = H +$

$(s - Hy)(s - Hy)^T$

$\langle s - Hy, y\rangle$

(5.5)

provided that $\langle s - Hy, y\rangle =$

0. This is a simple consequence of Lemma. 4.3. Indeed, let

$u = \dfrac{(y - Bs)}{}$

$\langle y - Bs, s\rangle$

and $v = y - Bs$. (5.4) becomes, using the lemma,

$\bar{H} = H - \frac{1}{\sigma} Huv^T H,$ if $\sigma = 1 + \langle v,$
$Hu\rangle =$
0. (5.6)

Let us come back to the definition of u and v, by definition of H, we have:

$\sigma = \underline{\langle y - Bs, H(y - Bs)\rangle} = \underline{\langle y - Bs, Hy\rangle}$.
$\langle y - Bs, s\rangle$

B being symmetric, its inverse is also, and

$\sigma = \underline{\langle Hy - s, y\rangle}$.
$\langle y - Bs, s\rangle\langle y - Bs, s\rangle$

Thenceforth, σ /= 0 if and only if $\langle Hy - s, y\rangle =$
0, then from (5.6)

$\bar{H} = H - \underline{\langle y - Bs, s\rangle} H \underline{(y - Bs)} (y - Bs)^T H,$
$\langle Hy - s, y\rangle$ $\langle y - Bs, s\rangle$that is
which shows (5.5).

$\bar{H} = H -$
$(Hy - s)(Hy - s)^T$

$\overline{\qquad\qquad\qquad}$,
$\langle Hy - s, y\rangle$

The following theorem, due to Fiacco and McCormick [8], shows that the above updating formula has an interesting feature when applied to a quadratic case (which justifies the method).

Theorem 5.1 *Let* $A \in \mathrm{L}(\mathrm{R}^n)$ *be a symmetric nonsingular matrix and* $y_k = As_k$, $0 \le k \le m$ *where* $\{s_0, s_1, . . . , s_m\}$ *is spanning* R^n. *Let* H_0 *be a symmetric matrix, and for* $k = 0, 1, . . . , m$ *the sequence*
in which we suppose that
$H_{k+1} = H_k +$
$(s_k - H_k y_k)(s_k - H_k y_k)^T$

$\langle s_k - H_k y_k, y_k \rangle$ (5.7)

$\langle s_k - H_k y_k, y_k \rangle =$

0. (5.8)

Then $H_{m+1} = A^{-1}$.

Proof. The technique of the proof is purely algebraic. First, we will prove by induction that

$H_k y_j = s_j$, for $0 \leq j < k$ and $k = 1, 2, \ldots, m + 1$.

- For $k = 1$, we have $H_1 y_0 = s_0$, since $s = \overline{x} - x$ and $y = \nabla f(\overline{x}) - \nabla f(x)$. This $H_1 y_0 = s_0$ is the secant condition of the update.
- Suppose that the statement is true for k, this means that $H_k y_j = s_j$, for $0 \leq j < k$.
- Now, we will prove that it is true for $k + 1$. From the updating formula we have

$(s_k - H_k y_k)(s_k - H_k y_k)^T y_j$

$Hk+1yj = Hkyj +$

$\langle s_k - H_k y_k, y_k \rangle$

We have

$= H_k y_j +$

$(s_k - H_k y_k)[(s_k - H_k y_k)^T y_j]$

$\langle s_k - H_k y_k, y_k \rangle$

.

$(s_k - H_k y_k)^T y_j = s^T y_j - y^T H^T y_j,$ k k k

$= s^T y_j - y^T H_k y_j$ (because H_k is symetric),

k k

$= s^T y_j - y^T s_j$ (because $H_k y_j = s_j$),

k k

$= s^T A s_j - s^T A s_j$ (because $y_j = A s_j$),

k k

$= 0.$

Thus $H_k y_j = s_j,$ where $0 \leq j < k$ for $k = 1, 2, \ldots, m + 1$.

Now, as $AH_{m+1}y_j = As_j = y_j$ for $0 \leq j < k$ and $k = 1, 2, \ldots, m + 1$, and $\{s_j\}^m$ spans R^n, we have $H_{m+1}A = I$. So $H_{m+1} = A^{-1}$. 0

The interest in the previous theorem is that we have an iterative scheme of the form $x_{k+1} =$

$x_k + s_k$ such that (5.8) is true, then

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

where the matrix H_k is updated by (5.7), gives us a tool to find the minimum of a quadratic form in a finite number of steps. We have, indeed:

$x_{k+1} = x_k - H_k \nabla f(x_k)$, with $s_k = -H_k \nabla f(x_k)$, and $y_k = As_k$.

From the theorem above we have in $(m + 1)$ steps

$x_{m+2} = x_{m+1} - A^{-1} \nabla f(x_{m+1})$.

If we have

$$f(x) = \frac{1}{\;}\langle Ax, x\rangle + b^T x + c, \qquad 2$$

where $A \in \mathrm{L}(\mathrm{R}^n)$ is supposed to be nonsingular symmetric, the minimum x^* of f is characterized by $\nabla f\,(x^*) = Ax^* + b = 0$ that is $x^* = -A^{-1}b$. Relation. (5.9), by definition of the gradient, implies that

$x_{m+2} = x_{m+1} - A^{-1}(Ax_{m+1} + b) = A^{-1}b,$

which gives that x_{m+2} is the minimum x^* and the algorithm converges in $(m + 2)$ steps. unfortunately, there is no guarantee that $\langle s_k - H_k y_k, y_k\rangle =$

0 even if Goldfarb [11] has shown that if $(A^{-1} - H_k)$ is symmetric semi-definite (positive or negative) and if H_k are generated via (5.7) when (5.8) is true, with $H_{k+1} = H_k$ when (5.8) is not true then $H_{m+1} = A^{-1}$ is still true. And we can show the following lemma.

Lemma 5.2 *Let* $f(x) = a + b^T x + \frac{1}{2}\langle Ax, x\rangle$ *where* A *is symmetric nonsingular matrix and*
$y_k = As_k,\ \ 0 \le k \le m,$ *where* $\mathrm{R}^n = span\{s_0, s_1, \ldots, s_m\}.$
Let H_0 *be a symmetric matrix such that* $H_0 - A^{-1} \ge 0$ *(resp* ≤ 0*), if* H_k *is constructed such that*
$H_{k+1} = H_k +$
$(s_k - H_k y_k)(s_k - H_k y_k)^T$

$\langle s_k - H_k y_k, y_k\rangle$,

in which we suppose that $\langle s_k - H_k y_k, y_k\rangle =$
0*, if not we take* $H_{k+1} = H_k$*. We have*
1. $H_k - A^{-1}$ *is positive semidefinite.*
2. $H_{m+1} = A^{-1}$.
□
Proof.
1. By induction, we have

- for $k = 1$, we have from assumptions $H_0 - A^{-1}$ is positive semidefinite,
- assume that $H_k - A^{-1}$ is positive semidefinite for k,
- we need to show that $H_{k+1} - A^{-1}$ is also positive semidefinite.

We have for all j
$s^T (H_k - A^{-1}) y_k y^T (H_k - A^{-1})^T s_j$

k

$s^T (H$
$k+1$
$- A^{-1}) s_j$
$= s^T (H$ $= s^T$

$A^{-1}) s + {}^{j}$

X
≥ 0

$\langle (H_k$

k

$— A^{-1})y_k \quad j \qquad j$

$, y_k \rangle$

$s^T (H \ - A^{-1})y \ \ 2 \qquad j \ — A^{-1})s_k \, , \overline{\ \ k \qquad\qquad k\ \ } — + — j$

k

j

$\geq 0.$

x

≥ 0

$y^T (H_k - A^{-1})y_k$

The second term is positive as the denominator is positive and the numerator is eitherpositive or equal 0.

2. $H_{m+1} = A^{-1}$. Similarly as the proof in the previous theorem.

Conclusions

Quasi-Newton methods are useful tools in solving unconstrained large-scale nonlinear optimiza- tion problems. They are proposing very nice formulas to update the Hessian approximations to be used in recursive formulas to converge toward solutions quickly. The BFGS method is very useful strategy for this task. A good method requires fast convergence, simplicity of the algorithm, stability, little storage memory, and lastly, a good estimate of the solution. The BFGS method satisfies all these requirements and it is therefore an effective iterative method. But, we have to choose the starting points x_0 and H_0 reasonably good and use a convenient λ.

Bibliography

[1] H. Attouch, G. Buttazzo, and G. Michaille. *Variational Analysis in Sobolev and BV Spaces: Applications to PDEs and Optimization*, volume 17 of *Series on Optimization*. SIAM, 2014.

[2] O. Axelsson and V. A. Barker. *Finite Element Solution of Boundary Value Problems: Theory and Computation*, volume 35. SIAM, 1984.

[3] A. Beck. *Introduction to Nonlinear Optimization: Theory, Algorithms, and Applications with MATLAB*, volume 19 of *Series on Optimization*. SIAM, 2014.

[4] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge university press, 2004.

[5] E. K. Chong and S. H. Zak. *An Introduction to Optimization*. John Wiley & Sons, 2013.

[6] J. E. Dennis, Jr and J. J. Moré. Quasi-Newton Methods, Motivation and Theory. *SIAM Review*, 19(1):46–89, 1977.

[7] J. E. Dennis Jr and R. B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, volume 16 of *Classics in Applied Mathematics*. SIAM, 1996.

[8] A. V. Fiacco and G. P. McCormick. *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, volume 4 of *Classics in Applied Mathematics*. SIAM, 1990.

[9] R. Fletcher. *Practical Methods of Optimization*. John Wiley & Sons, 2013.

[10] P. E. Gill and W. Murray. Quasi-Newton Methods for Unconstrained Optimization. *IMA Journal of Applied Mathematics*, 9(1):91–108, 1972.

[11] D. Goldfarb. Sufficient Conditions for the Convergence of a Variable Metric Algorithm. In *Optimization*, pages 273–281. Academic Press New York, 1969.

[12] D. Goldfarb. Factorized Variable Metric Methods for Unconstrained Optimization. *Math- ematics of Computation*, 30(136):796–811, 1976.

[13] S. M. Goldfeld, R. E. Quandt, and H. F. Trotter. Maximization by Quadratic Hill-Climbing.

Econometrica: Journal of the Econometric Society, pages 541–551, 1966.

[14] G. H. Golub and C. F. Van Loan. *Matrix Computations*, volume 4th edition. JHU Press,2013.

[15] W. Hackbusch. *Iterative Solution of Large Sparse Systems of Equations*, volume 95 of

Applied Mathematical Sciences. Springer-Verlag New York, 2012.

[16] N. J. Higham. *Functions of Matrices: Theory and Computation*. SIAM, 2008.

[17] C. T. Kelley. *Iterative Methods for Optimization*, volume 18 of *Frontiers in Applied Math- ematics*. SIAM, 1999.

[18] A. B. Levy. *The Basics of Practical Pptimization*. SIAM, 2009.

[19] D. G. Luenberger, Y. Ye, et al. *Linear and Nonlinear Programming*, volume 116 of *Inter- national Series in Operations Research & Management Science*. Springer-Verlag, 1984.

[20] G. P. McCormick and K. Ritter. Methods of Conjugate Directions versus Quasi-Newton Methods. *Mathematical Programming*, 3(1):101–116, 1972.

[21] L. Nazareth. A relationship between the BFGS and Conjugate Gradient algorithms and its implications for new algorithms. *SIAM Journal on Numerical Analysis*, 16(5):794–800, 1979.

[22] J. M. Ortega and W. C. Rheinboldt. *Iterative Solution of Nonlinear Equations in SeveralVariables*. SIAM, 1970.

[23] M. J. Powell. A New Algorithm for Unconstrained Optimization. *Nonlinear Programming*, pages 31–65, 1970.

[24] A. Quarteroni, R. Sacco, and F. Saleri. *Numerical Mathematics*. Springer-Verlag, 2010.

[25] Y. Saad. *Iterative Methods for Sparse Linear Systems*. SIAM, 2003.

[26] J. Sherman and W. J. Morrison. Adjustment of an inverse matrix corresponding to changes in the elements of a given column or a given row of the original matrix. In *Annals of Mathematical Statistics*, volume 20, pages 621–621, 1949.

[27] G. Strang and H. Matthies. Numerical Computations in Nonlinear Mechanics. In *Interna- tional Conference on Scientific and Technical Calculation Methods, 4th, Versailles, France,December 10-14, 1979*, volume 1, 1980.

[28] J. H. Wilkinson. *The Algebraic Eigenvalue Problem*. Clarendon Press Oxford, 1965.