Research Report

KSTS/RR-15/005
March 9, 2015

The L-BFGS method for nonlinear GMRES acceleration

by

Yuto Yokota
Takashi Nodera

Yuto Yokota
Graduate School of Science and Technology
Keio University

Takashi Nodera
Department of Mathematics
Keio University

Department of Mathematics
Faculty of Science and Technology
Keio University

©2015 KSTS
3-14-1 Hiyoshi, Kohoku-ku, Yokohama, 223-8522 Japan

KSTS/RR-15/005
March 9, 2015

The L-BFGS method for nonlinear GMRES acceleration

Yuto Yokota[1] Takashi Nodera[2]

March 9, 2015

Abstract

The limited-memory BFGS (L-BFGS) method is considered for the nonlinear generalized minimal residual (N-GMRES) for unconstrained nonlinear optimization. In this paper, a universal preconditioning approach for the N-GMRES that can be applied to a quasi-Newton procedure is explored. Numerical experiments on optimization problems suggest that the L-BFGS is able to speed up significantly, the convergence of the N-GMRES.

Keywords: nonlinear GMRES, quasi-newton method, L-BFGS, unconstrained optimization,
AMS(MOS) subject classifications. 65F10, 65K10, 65F08

Contents

1 Introduction

The nonlinear GMRES (N-GMRES) method with the steepest decent [8] is an efficient preconditioner for solving general unconstrained optimization problems. This method accelerates the convergence of the alternating least squares (ALS) optimization method for

the canonical tensor approximation problem. The N-GMRES method can be used as a simple wrapper around any other iterative optimization process to accelerate this process. Steepest descent optimization is particularly suited for solving optimization problems under appropriate conditions. However, this method converges a little bit slowly, because the steepest descent is characterized by a property of linear convergence. The conventional steepest descent algorithm may require an extremely large number of iterations, functions and gradient evaluations. To overcome this issue, the quasi-Newton method must be obtained in the framework of the unconstrained minimization of quadratic functions. The quasi-Newton method which is also known as the limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) method, is explored in this paper. A more universal preconditioning approach for the N-GMRES method used in combination with the new accelerator L-BFGS is proposed in this study.

Consider the following unconstrained nonlinear optimization problem with associated first-order optimal equations:

Optimization Problem:

$$\text{find } x^* \text{ that minimizes } f(x).$$

First-Order Optimal Equations:

$$\nabla f(x) \equiv \mathbf{g}(x) = 0. \tag{1}$$

The N-GMRES optimization algorithm, which is illustrated in Algorithm 1, consists of three steps that can be summarized as follows. In the first step, a preliminary new iteration $\bar{x}_{i+1}$ is generated from the last iteration x_i using a one-step iterative update process $M(.)$. This is a preconditioning process. In this paper, an L-BFGS preconditioning for $M(.)$ has been employed. In the second step, an accelerated iteration $\hat{x}_{i+1}$ is obtained through combining the previous iterations in a window size w, using a nonlinear GMRES approach. In the third step, a line search is performed that minimizes objective function $f(x)$ on a half line starting at the preliminary iteration $\bar{x}_{i+1}$ and it connects with the accelerated iteration $\hat{x}_{i+1}$ to obtain the new iteration x_{i+1}.

This paper is organized as follows: Section 2 is an outline of the N-GMRES optimization with a steepest descent preconditioning proposed by Sterck [9]. Section 3 details the L-BFGS preconditioner for the N-GMRES optimization algorithm. In Section 4, results of the numerical experiments obtained through running the codes of N-GMRES with the L-BFGS preconditioning, applied to two nonlinear optimization problems are documented. Section 5 is the conclusion.

2 N-GMRES Algorithm

A simple example of the N-GMRES algorithm with steepest descent preconditioning is documented in this section. Two variants of the steepest descent preconditioning for Step 1 of the N-GMRES have been proposed by [9] :

Steepest Descent Preconditioning:

$$\bar{x}_{i+1} = x_i - \beta \frac{\nabla f(x_i)}{\|\nabla f(x_i)\|},$$

Algorithm 1: N-GMRES Optimization Algorithm (window size w)

Input: w initial guess $x_0, x_1, \ldots, x_{w-1}$
Output: f^*, x^*

$i = w - 1$;
repeat
  Step 1 : generate preliminary iteration by one-step update process $M(.)$.;
    $\bar{x}_{i+1} = M(x_i)$;
  Step 2 : generate accelerated iteration by nonlinear GMRES step;
    $\hat{x}_{i+1} = \text{gmres}(x_{i-w+1}, \ldots, x_i; \bar{x}_{i+1})$;
  Step 3 : generate new iteration by line search process;
  if $\hat{x}_{i+1} - \bar{x}_{i+1}$ *is search direction* **then**
    $x_{i+1} = \text{linesearch}(x_{i+1} + \beta(\hat{x}_{i+1} - \bar{x}_{i+1}))$
  else
    $x_{i+1} = \bar{x}_{i+1}$;
  end
until *convergence criterion satisfied*;

where we can choose parameter β as follows:

$$\text{Option A} : \beta = \beta_{\text{sdls}}, \tag{2}$$

$$\text{Option B} : \beta = \beta_{\text{sd}} = \min(\delta, \|\nabla f(x_i)\|). \tag{3}$$

For Option A, β_{sdls} is the step size which is obtained through a line search procedure satisfying the Wolfe conditions [1, p. 307]. Next, for Option B, β_{sd} is the minimum of a small positive constant δ and the gradient norm. The preconditioning process A is usually considered to be a stand-alone optimization method, and the N-GMRES can be used to accelerate the convergence to find the optimal solution, rapidly. It has a strong convergence property, but it requires evaluating the number of functions and gradient f/g.

The preconditioning process B requires no additional f/g evaluations, but its convergence is not guaranteed. The role of the preconditioning process is to provide new directions for the nonlinear generalization to the Krylov space, and the iteration can be driven to convergence by N-GMRES minimization. In the initial step, the β_{sd} chosen must be no larger than a small constant because the linear case suggests that a small constant is sufficient to provide a new direction for the Krylov space, and the minimization of the residual norm is based on a linearization argument; small steps tend to lead to small linearization errors.

3 Limited-memory BFGS Algorithm

This section is a study of the Limited-memory BFGS method. A preconditioner of the N-GMRES using the L-BFGS method will be examined. This method accelerates the convergence for reaching the optimal solution, rapidly. The L-BFGS method is one of the quasi-Newton methods. The L-BFGS method resolves the issues of massive computer memory use Moré et al. [5]. The BFGS formula is as follows:

$$H_{k+1} = H_k - \frac{H_k s_k (B_k s_k)^T}{(s_k)^T H_k s_k} + \frac{y_k (y_k)^T}{(s_k)^T y_k}. \tag{4}$$

Rewriting equation (4), results in following equation:

$$H_{k+1} = \left(I - \frac{s_k y_k^T}{y_k^T s_k}\right) H_k \left(I - \frac{y_k s_k^T}{y_k^T s_k}\right) + \frac{s_k s_k^T}{y_k^T s_k}. \tag{5}$$

Set $V_k = I - \rho_k y_k s_k, \rho_k = 1/y_k^T s_k$ and this results in equation (6).

$$H_{k+1} = V_k^T H_k V_k + \rho_k s_k s_k^T. \tag{6}$$

Using formula (6), H_k can be expanded recursively, and this results in the following equation:

$$\begin{aligned} H_{k+1} = &(V_k^T \dots V_0^T) H_0 (V_0 \dots V_k) \\ &+ \rho_0 (V_k^T \dots V_1^T) s_0 s_0^T (V_1 \dots V_k) \\ &+ \rho_1 (V_k^T \dots V_2^T) s_1 s_1^T (V_2 \dots V_k) \\ &\quad \vdots \\ &+ \rho_k s_k s_k^T. \end{aligned} \tag{7}$$

The recursive formula of the Hessian matrix consists of an initial matrix H_0 and x and ∇f at each iteration. If the problem size is n, we require only $\mathcal{O}(kn)$ computer memory. Since the L-BFGS method iterates only m steps, the required computational memory of x and ∇f are needed $\mathcal{O}(mn)$. Then, using the information of only m steps, equation (7) is rewritten as follows:

$$\begin{aligned} H_{k+1} = &(V_k^T \dots V_{k-m}^T) H_0 (V_{k-m} \dots V_k) \\ &+ \rho_0 (V_k^T \dots V_{k-m+1}^T) s_0 s_0^T (V_{k-m+1} \dots V_k) \\ &+ \rho_1 (V_k^T \dots V_{k-m+2}^T) s_1 s_1^T (V_{k-m+2} \dots V_k) \\ &\quad \vdots \\ &+ \rho_k s_k s_k^T. \end{aligned} \tag{8}$$

An algorithmic description of the L-BFGS process is given below:

1. Choose an initial guess x_0, window size w, parameter $0 < \beta' < 1/2, \beta' < \beta < 1$, SPD matrix H_0, and set $k := 0$.
2. Compute $d_k = -H_k \nabla f(x_k)$, $x_{k+1} = x_k + \alpha_k d_k$. α_k satisfying the Wolfe conditions [1, p. 307] here.
3. Set $\hat{w} = \min(k, w-1)$ and compute H_0 by using y_k, s_k $\hat{w}+1$ times.
4. Set $k := k+1$ and go to Step 2.

3.1 L-BFGS Preconditioning

As mentioned in Section 1, the N-GMRES method with the steepest descent preconditioner has only a linear convergence property and for this reason it converges slowly. The proposal is to use the L-BFGS as a precondtioner of the N-GMRES to accelerate convergence of

the objective function using a quadratic convergence property of a Newton-like method. In Step 1, the L-BFGS algorithm generates a sequence of iterates $\{x_k\}$ by

$$\bar{x}_{i+1} = x_i - \beta \frac{d_i}{\|d_i\|_2^2}. \tag{9}$$

Here vector d_i is a search direction vector computed in each iteration, and parameter β is a step size which is obtained through line search. As can be seen from the figures of the numerical experiments in Section 4, the convergence of the stand-alone L-BFGS method is faster than the steepest descent method. Through choosing the L-BFGS method as a preconditioner of the N-GMRES, a significant improvement of convergence can be expected. In addition, as mentioned above, the L-BFGS preconditioner has a few memory requirements. Therefore, using this method as a preconditioner for the N-GMRES, a better convergence property is possible, minimizing extra memory cost.

4 Numerical Experiments

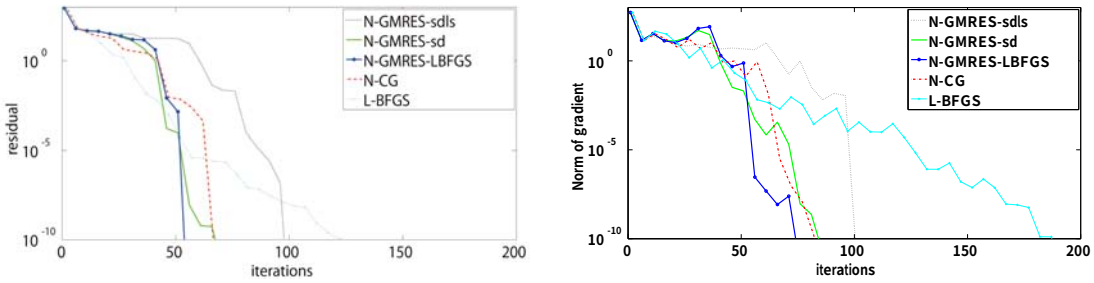
Some numerical experiments for the N-GMRES algorithm with a L-BFGS preconditioner have been compared with the N-CG, L-BFGS, and N-GMRES algorithms with steepest descent preconditioners.

- **OS** : Windows 7 Professional (64-bit)
- **CPU** : Intel(R) Core-i7 2700K(3.5GHz,TB3.9GHz)
- **Memory** : 16GB
- **Precision** : double precision
- **Program Language** : MATLAB R2013a

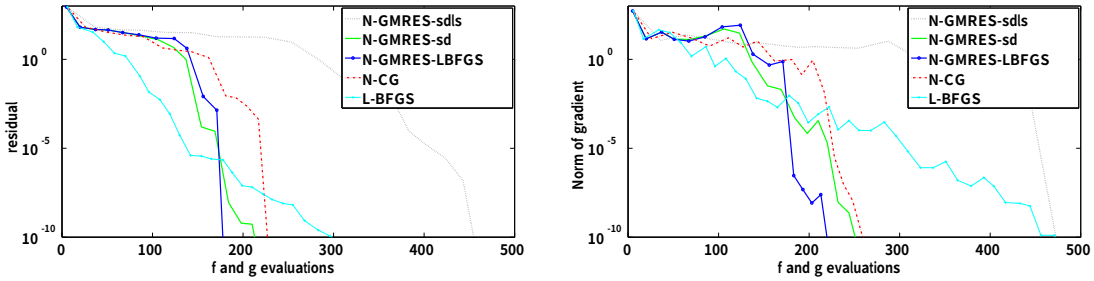
The "SD" and the "SDLS" were proposed by Sterck [9] and these use the steepest descent method as preconditioners, but are different in terms of the choice of parameter β. In the former $\beta = \min(\delta, \|\nabla f(x_i)\|)$ is used, and in the latter the step size that is obtained from the line search is used. In both tests, the Moré-Thuente line search method [5] were used as well as the N-CG and L-BFGS optimization methods as implemented in the Poblano toolbox for MATLAB [2]. For all experiments, the Moré-Thuente line search parameters were as follows:

- Function value tolerance: $c_1 = 10^{-4}$
- Gradient norm tolerance: $c_2 = 10^{-2}$
- Starting search step length: $\beta = 1$
- Maximum number of f/g evaluation: 20

All initial guesses are determined uniformly and randomly with components in interval [0,1].



(a) Convergence to f^* (b) Convergence behavior of the gradient norm



(c) Convergence to f^* (d) Convergence behavior of the gradient norm

Figure 1: Example A ($n = 1000$) - Comparison of convergence. (a) convergence tof^*, (b) convergence behavior of the gradient norm, (c) convergence to f^* with f and g evaluations, and (d) convergence behavior of the gradient norm with f and g evaluations.

Table 1: Example A - Average number of f/g evaluations needed to reach $|f(x_i) - f^*| < 10^{-10}$

size(n)	SDLS	SD	N-GMRES-LBFGS	N-CG	L-BFGS
5000	409	246	199	237	293
10000	470	231	208	245	290

4.1 Example A

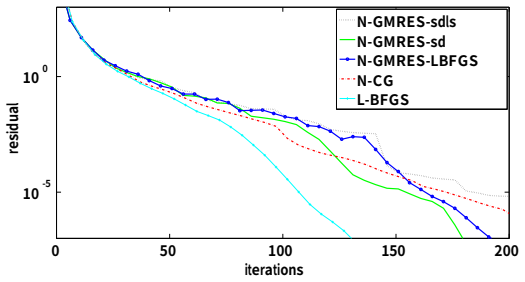
Extended Rosenbrock Function:

$$f(x) = \sum_{i=1}^{n/2} [100(x_{2i-1}^2 - x_{2i})^2 + (x_{2i-1} - 1)^2] \tag{10}$$

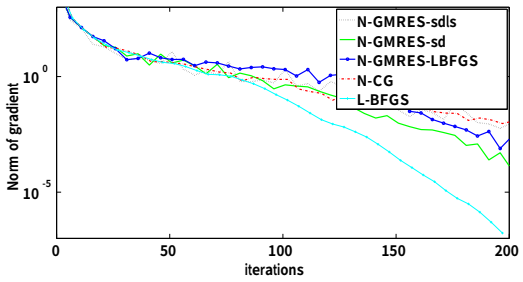
- Minimum $f(x^*) = 0$ at $x^* = (1, 1, \ldots, 1)^T$

In Figure 1, the numerical results for example A are shown. It can be seen that the N-GMRES method using the L-BFGS preconditioning is significantly faster than the stand-alone L-BFGS method in terms of iterations and f/g evaluations. In addition, the N-GMRES with a L-BFGS preconditioning was faster compared to classical procedures.

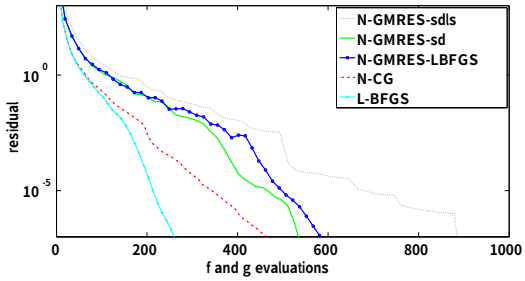
In Table 1, the average number of f/g evaluations that were required to reach $|f(x_i) - f^*| < 10^{-10}$ for random instances of example A with different sizes, were tabulated. The



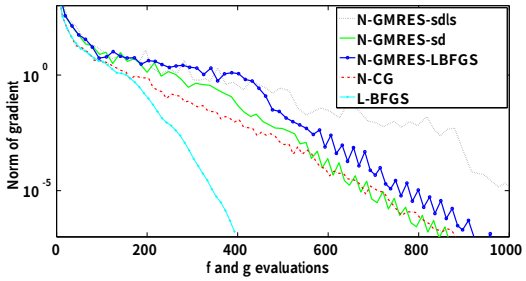
(a) Convergence to f^*



(b) Convergence Behavior of Gradient Norm



(c) Convergence to f^*



(d) Convergence Behavior of Gradient Norm

Figure 2: Example B ($n = 1000$), Comparison of Convergence. (a) convergence tof^*, (b) convergence behavior of the gradient norm, (c) convergence to f^* with f and g evaluations, and (d) convergence behavior of the gradient norm with f and g evaluations.

Table 2: Example B - Average number of f/g evaluations needed to reach $|f(x_i) - f^*| < 10^{-10}$

size(n)	SDLS	SD	N-GMRES-LBFGS	N-CG	L-BFGS
5000	2474	1361	1594	1738	692
10000	3582	2089	2286	2554(2)	979

figures in this table suggest that the N-GMRES with a L-BFGS preconditioning requires less f/g evaluations than the other four methods.

4.2 Example B

Sphere Function:

$$f(x) = \sum_{i=1}^{n} x_i^2 \tag{11}$$

- Simple convex function
- Minimum $f(x^*) = 0$ at $x^* = (0, 0, \ldots, 0)^T$

In Figure 2, the numerical results for example B are tabulated. From this figure, we can see that the stand-alone L-BFGS method is faster than the N-GMRES with a L-BFGS

preconditioning. The N-GMRES method is unable to accelerate the classical L-BFGS method for a simple convex function.

In Table 2, the average number of f/g evaluations that were required to reach $|f(x_i) - f^*| < 10^{-10}$ for random instances of example B with different sizes were tabulated. However, it should be noted that B and the N-GMRES with LBFGS preconditioning required approximately double the number of iterations. This was the result of a failure to determine parameter β through a line search. Consequently, in the neighbor of the optimal solution, the number of f/g evaluations by line search was significantly increased.

5 Conclusion

In this paper, a hybrid method combining the N-GMRES and the L-BFGS preconditioning was proposed. Numerical experiments showed that the proposed method accelerates the stand-alone method very significantly for a certain objective function. The most important property of the N-GMRES is that its convergence speed is not dependent on the initial guess, and that it guarantees global convergence. Depending on the choice of the preconditioner, this method can be applied to various optimization problems.

References

[1] Cheng, W., Y. and Li, D. H, "Spectral Scaling BFGS Method," J. Optim Theory Appl., 146, pp. 305–319, 2010. http://dx.doi.org/10.1007/s10957-010-9652-y

[2] Dunlavy, M., Kolda, T. G. and Acar, E., "Poblano v1.0: A Matlab Toolbox for Gradient-Based Optimization," Technical Report SAND2010-1422, Sandia National Laboratories, Albuquerque, NM and Livermore, CA, 2010. https://software.sandia.gov/trac/poblano

[3] Fang, H. and Saad, Y., "Two Classes of Multisecant Methods for Nonlinear Acceleration," Numerical Linear Algebra with Appl. 16, pp. 197–221, 2009. http://dx.doi.org/10.1002/nla.617

[4] Liu, D. C. and Nocedal, J., "On the Limited Memory BFGS Method For Large Scale Optimization," Mathematical Programming 45, pp. 503–528, 1989. http://dx.doi.org/10.1007/BF01589116

[5] Moré, J. J. and Garbow, B. S., and Hillstrom, K. E. "Testing Unconstrained Optimization Software", ACM Transactions on Mathematical Software (TOMS) 7, pp. 17–41, 1981. http://dx.doi.org/10.1145/355934.355936

[6] Nocedal, J. and Wright, S, J., "Numerical Optimization," Second Edition, Springer, Berlin, 2006. http://dx.doi.org/10.1007/978-0-387-40065-5

[7] Hager, W. W. and Zhang, H., "A Survey of Nonlinear Conjugate Gradient Methods," Pacific Journal of Optimization, 2, pp. 35–58, 2006. http://www.ybook.co.jp/online2/oppjo/vol2/p35.html

[8] De Sterck, De., "A Nonlinear GMRES Optimization Algorithm for Canonical Tensor Decomposition," SIAM J. Sci. Comput. 34, pp. A1351–A1379, 2013. http://dx.doi.org/10.1137/110835530

[9] ———, "Steepest Descent Preconditioning for Nonlinear GMRES Optimization," Numerical Linear Algebra with Appl. 20,pp. 453–471, 2013. http://dx.doi.org/10.1002/nla.1837

[10] Walker, H. and Ni, P., "Anderson Acceleration for Fixed-point Iterations," SIAM J. Numer. Anal. 49, pp. 1715–1735, 2011. http://dx.doi.org/10.1137/10078356X

[11] Washio, T. and Oosterlee, C. W., "Krylov Subspace Acceleration for Nonlinear Multigrid Schemes," Electronic Transactions on Numerical Analysis 6, pp. 271–290, 1997. http://dx.doi.org/10.1.1.40.4741

Author addresses

1. **Yuto Yokota**, School of Fundamental Science and Technology,Graduate School of Science and Technology, Keio University, JAPAN.
mailto:yuto_yokota@keio.jp

2. **Takashi Nodera**, Department of Mathematics,Faculty of Science and Technology, Keio University, JAPAN.
mailto:nodera@math.keio.ac.jp

Department of Mathematics
Faculty of Science and Technology
Keio University

Research Report

2014

[14/001] A. Larraín-Hubach, Y. Maeda, S. Rosenberg, F. Torres-Ardila,
Equivariant, strong and leading order characteristic classes associated to fibrations,
KSTS/RR-14/001, January 6, 2014

[14/002] Ryoichi Suzuki,
A Clark-Ocone type formula under change of measure for canonical Lévy processes,
KSTS/RR-14/002, March 12, 2014

2015

[15/001] Shiro Ishikawa,
Linguistic interpretation of quantum mechanics: Quantum Language,
KSTS/RR-15/001, January 22, 2015

[15/002] Takuji Arai, Ryoichi Suzuki
Local risk-minimization for Lévy markets,
KSTS/RR-15/002, February 2, 2015

[15/003] Kazuma Teramoto, Takashi Nodera
Lanczos type method for computing PageRank,
KSTS/RR-15/003, March 9, 2015

[15/004] Yoichi Matsuo, Takashi Nodera
Block symplectic Gram-Schmidt method,
KSTS/RR-15/004, March 9, 2015

[15/005] Yuto Yokota, Takashi Nodera
The L-BFGS method for nonlinear GMRES acceleration,
KSTS/RR-15/005, March 9, 2015