Proceedings of the International Conference on
Nonlinear Analysis and Convex Analysis &
International Conference on Optimization:
Techniques and Applications -II-
(Hakodate, Japan, 2019), 147–160

A GLOBALLY CONVERGENT ACTIVE-SET MEMORYLESS QUASI-NEWTON METHOD BASED ON SPECTRAL-SCALING BROYDEN FAMILY FOR BOUND CONSTRAINED OPTIMIZATION

HIROAKI NISHIO, SHUMMIN NAKAYAMA, YASUSHI NARUSHIMA,
AND HIROSHI YABE

Abstract. In this paper, we consider a new algorithm for solving bound constrained optimization problems. To develop an efficient method for large-scale problems, we propose an active-set memoryless quasi-Newton method. This method combines an active-set strategy with the memoryless quasi-Newton method proposed by Nakayama et al. (2019), which is based on the Broyden family with the spectral-scaling secant condition. We incorporate a restart strategy into the active-set strategy and show the global convergence of our method within the framework of the Armijo line search. Some numerical experiments are given to investigate how the choice of parameters affects numerical performance.

1. Introduction

In this paper, we consider the following bound (box) constrained optimization problem:

$$(1.1)\qquad \begin{cases} \min\limits_{x\in K} f(x), \\ K = \{x \in R^n \mid l_i \le (x)_i \le u_i, \quad i = 1,\dots,n\}, \end{cases}$$

where $l_i \in R \cup \{-\infty\}, u_i \in R \cup \{\infty\}$ $(i = 1,\dots,n)$ and the objective function $f : R^n \to R$ is continuously differentiable and its gradient $\nabla f(x)$ is denoted by $g(x)$. In addition, for any vector v, we denote the i-th component of v by $(v)_i$. Without loss of generality, we assume that $l_i < u_i$ $(i = 1,\dots,n)$. If there exists an index i such that $l_i = u_i$, then we can set $(x)_i = l_i = u_i$ and delete $(x)_i$ from (1.1). Iterative methods are usually used for solving problem (1.1),

2010 *Mathematics Subject Classification.* 90C30, 90C53, 90C06.
Key words and phrases. Bound constrained optimization, active-set method, memoryless quasi-Newton method, Broyden family, global convergence.

and they are of the form

$$x_{k+1} = x_k + \eta_k d_k, \tag{1.2}$$

where $x_k \in R^n$ is the k-th approximation to a solution, $\eta_k > 0$ is a step size, and $d_k \in R^n$ is a search direction. We denote $g(x_k)$ by g_k.

The bound constrained optimization problem (1.1) appears in various kinds of applications including the control allocation problem [17], the image deblurring [19], the molecular conformation analysis [13] and the linear support vector machine [26]. Furthermore, problem (1.1) also appears as a subproblem in augmented Lagrangian methods and penalty methods [9, 11].

There are several numerical methods for solving problem (1.1), which include active-set type methods, projected gradient methods and affine scaling methods, for example [2, 3, 6, 7, 10, 15, 16]. In this paper, we focus on an active-set method. Recently, Yuan and Lu [27] proposed an active-set limited memory BFGS algorithm that used an active set identification technique [10] to estimate active variables, and determined the search direction for free variables by the limited memory quasi-Newton method.

Inspired by their algorithm, we combine this technique with memoryless quasi-Newton methods based on the spectral-scaling Broyden family. When we solve large-scale bound constrained optimization problems, the number of free variables becomes large. If we use usual quasi-Newton methods, we need too much memory requirement, because the approximate matrix of Hessian is dense. So, we focus on the memoryless quasi-Newton method proposed by Shanno [25], which was developed for solving unconstrained optimization problems. Memoryless quasi-Newton methods need less memory requirement. Recently, memoryless quasi-Newton methods have been studied by several researchers [18, 20, 22, 23]. Based on the Broyden family with the spectral-scaling secant condition [4, 5], Nakayama et al. [24] claimed that the preconvex class performs better than the BFGS update.

In this paper, we modify the active-set strategy of Yuan and Lu [27] which is based on Facchinei et al. [10] and combine the modified active-set strategy with the memoryless quasi-Newton method based on the spectral-scaling Broyden family. In our numerical experiments, we investigate how our modification and a choice of parameters of the Broyden family affect numerical performance.

This paper is organized as follows. In Section 2, we first modify the algorithm of Yuan and Lu [27]. Next, we introduce memolyless quasi-Newton methods and propose our method. In addition, we show the global convergence of the proposed method. In Section 3, we present some numerical experiments. Finally Section 4 gives conclusions.

2. Proposed method and its global convergence

In this section, we first introduce the active-set method of Yuan and Lu [27] and give its modification. Next, we introduce quasi-Newton methods and combine the modified active-set method with memoryless quasi-Newton methods. Finally, we present the global convergence results of the proposed method.

The KKT conditions of (1.1) are equivalent to the following conditions:

$$\begin{cases} (g(\overline{x}))_i \geq 0 & \text{for all } i \in \overline{L} := \{i : l_i = (\overline{x})_i\}, \\ (g(\overline{x}))_i = 0 & \text{for all } i \in \overline{F} := \{i : l_i < (\overline{x})_i < u_i\}, \\ (g(\overline{x}))_i \leq 0 & \text{for all } i \in \overline{U} := \{i : u_i = (\overline{x})_i\}. \end{cases} \tag{2.1}$$

Note that $\overline{x}$ that satisfies (2.1) is called a stationary point. We define sets $L(x)$, $U(x)$, and $F(x)$ by

$$\begin{cases} L(x) := \{i : (x)_i \leq l_i + a_i(x)(g(x))_i\}, \\ U(x) := \{i : (x)_i \geq u_i + b_i(x)(g(x))_i\}, \\ F(x) := \{1, \ldots, n\} \backslash (L(x) \cup U(x)), \end{cases}$$

where $a_i(x)$ and $b_i(x)$ $(i = 1, \ldots, n)$ are nonnegative continuous functions bounded from above on K, such that if $(x)_i = l_i$ (respectively, $(x)_i = u_i$), then $a_i(x) > 0$ (respectively, $b_i(x) > 0$). Then there exist positive constants $\overline{a}$ and $\overline{b}$ such that

$$a_i(x) \leq \overline{a} \quad \text{and} \quad b_i(x) \leq \overline{b} \quad (i = 1, \ldots, n)$$

for any $x \in K$.

Let $Z_J \in R^{n \times |J|}$ be a matrix whose columns are $\{e_i | i \in J\}$ for any index set $J \subset \{1, \ldots, n\}$, where e_i is the i-th column of the identity matrix in $R^{n \times n}$. Yuan and Lu [27] defined the search direction d_k by

$$(d_k)_i = \begin{cases} l_i - (x_k)_i & \text{for all } i \in L(x_k), \\ u_i - (x_k)_i & \text{for all } i \in U(x_k), \\ -\alpha_*^k (Z_{F(x_k)} Z_{F(x_k)}^T H_k Z_{F(x_k)} Z_{F(x_k)}^T g_k)_i & \text{for all } i \in F(x_k), \end{cases} \tag{2.2}$$

where

$$\begin{aligned} \alpha_*^k = \max\{\alpha \mid\ & 0 \leq \alpha \leq 1, \\ & l_i \leq (x_k)_i - \alpha (Z_{F(x_k)} Z_{F(x_k)}^T H_k Z_{F(x_k)} Z_{F(x_k)}^T g_k)_i \leq u_i,\ i \in F(x_k)\} \end{aligned} \tag{2.3}$$

and H_k is an approximation to the matrix $\nabla^2 f(x_k)^{-1}$.

We note that there is a case where α_*^k in (2.3) becomes zero. For example, we assume that $F(x_k) = \{1, \ldots, n\}$ and that there exists i such that $(g_k)_i > 0$, $-(Z_{F(x_k)} Z_{F(x_k)}^T H_k Z_{F(x_k)} Z_{F(x_k)}^T g_k)_i > 0$ and $(x_k)_i = u_i$ hold. From the definition of α_*^k in (2.3), we have $\alpha_*^k = 0$ and $d_k = 0$. Thus x_k does not change, while x_k is not a stationary point, because x_k does not satisfy the third condition of (2.1). To avoid this phenomenon, we consider a modification of the active-set strategy of Yuan and Lu. Specifically, we define the index set $T(x_k)$ by

$$\begin{aligned} T(x_k) = \{ i \in F(x_k) \mid & \\ & \big((x_k)_i = l_i,\ -(Z_{F(x_k)}(Z_{F(x_k)}^T H_k Z_{F(x_k)}) Z_{F(x_k)}^T g_k)_i < 0,\ (g_k)_i < 0\big) \\ & \text{or } \big((x_k)_i = u_i,\ -(Z_{F(x_k)}(Z_{F(x_k)}^T H_k Z_{F(x_k)}) Z_{F(x_k)}^T g_k)_i > 0,\ (g_k)_i > 0\big) \} \end{aligned}$$

and use $-(H_k)_{ii}(g_k)_i$ as a search direction for $i \in T(x_k)$, where $(H_k)_{ii}$ is the $(i,\ i)$ element of H_k. Note that if H_k is positive definite, $(H_k)_{ii}$ is positive. Summarizing the above argument, we define the search direction d_k by

$$(d_k)_i = \begin{cases} l_i - (x_k)_i & \text{for all } i \in L(x_k), \\ u_i - (x_k)_i & \text{for all } i \in U(x_k), \\ -\alpha_{new}^k (Z_{F(x_k)\backslash T(x_k)} \bar{H}_k Z_{F(x_k)\backslash T(x_k)}^T g_k)_i & \text{for all } i \in F(x_k)\backslash T(x_k), \\ -\alpha_{new}^k (H_k)_{ii} (g_k)_i & \text{for all } i \in T(x_k), \end{cases} \tag{2.4}$$

where

$$\bar{H}_k = Z_{F(x_k)\backslash T(x_k)}^T H_k Z_{F(x_k)\backslash T(x_k)} \tag{2.5}$$

and

$$\begin{aligned} \alpha_{new}^k = \max\{ \alpha \mid & \ 0 \le \alpha \le 1, \\ & l_i \le (x_k)_i - \alpha (Z_{F(x_k)\backslash T(x_k)} \bar{H}_k Z_{F(x_k)\backslash T(x_k)}^T g_k)_i \le u_i,\ i \in F(x_k)\backslash T(x_k), \\ & l_j \le (x_k)_j - \alpha (H_k)_{jj} (g_k)_j \le u_j,\ j \in T(x_k) \}. \end{aligned} \tag{2.6}$$

It is significant that we suitably choose H_k in (2.5) for large-scale problems. Accordingly, we apply memoryless quasi-Newton methods to the active-set method (2.4)–(2.6). In the following, we briefly introduce quasi-Newton methods and memoryless quasi-Newton methods. Let B_k be an approximation to $\nabla^2 f(x_k)$ and set $H_k = B_k^{-1}$. The matrix B_k or H_k is updated at each iteration such that the secant condition

$$B_k s_{k-1} = y_{k-1} \qquad \text{or} \qquad H_k y_{k-1} = s_{k-1} \tag{2.7}$$

holds, where $s_{k-1} = x_k - x_{k-1}$ and $y_{k-1} = g_k - g_{k-1}$. As updating formulas that satisfy (2.7), the DFP update and the BFGS update are well-known. They are respectively given by

$$B_k = B_{k-1} - \frac{B_{k-1}s_{k-1}y_{k-1}^T + y_{k-1}(B_{k-1}s_{k-1})^T}{s_{k-1}^T y_{k-1}} + \left(1 + \frac{s_{k-1}^T B_{k-1} s_{k-1}}{s_{k-1}^T y_{k-1}}\right) \frac{y_{k-1}y_{k-1}^T}{s_{k-1}^T y_{k-1}} \tag{2.8}$$

and

$$B_k = B_{k-1} - \frac{B_{k-1}s_{k-1}(B_{k-1}s_{k-1})^T}{s_{k-1}^T B_{k-1} s_{k-1}} + \frac{y_{k-1}y_{k-1}^T}{s_{k-1}^T y_{k-1}}. \tag{2.9}$$

In this paper, we focus on the Broyden family:

$$\begin{cases} B_k = B_{k-1} - \dfrac{B_{k-1}s_{k-1}s_{k-1}^T B_{k-1}}{s_{k-1}^T B_{k-1} s_{k-1}} + \dfrac{y_{k-1}y_{k-1}^T}{s_{k-1}^T y_{k-1}} + \phi_{k-1} v_{k-1} v_{k-1}^T, \\ v_{k-1} = \sqrt{s_{k-1}^T B_{k-1} s_{k-1}} \left(\dfrac{y_{k-1}}{s_{k-1}^T y_{k-1}} - \dfrac{B_{k-1}s_{k-1}}{s_{k-1}^T B_{k-1} s_{k-1}} \right), \end{cases} \tag{2.10}$$

where $\phi_{k-1} \in R$ is a parameter. We can consider the Broyden family of the inverse matrix version:

(2.11)

$$\begin{cases} H_k = H_{k-1} - \dfrac{H_{k-1}y_{k-1}y_{k-1}^T H_{k-1}}{y_{k-1}^T H_{k-1} y_{k-1}} + \dfrac{s_{k-1}s_{k-1}^T}{s_{k-1}^T y_{k-1}} + \phi_{k-1}^H w_{k-1} w_{k-1}^T. \\ w_{k-1} = \sqrt{y_{k-1}^T H_{k-1} y_{k-1}} \left(\dfrac{s_{k-1}}{s_{k-1}^T y_{k-1}} - \dfrac{H_{k-1}y_{k-1}}{y_{k-1}^T H_{k-1} y_{k-1}} \right). \end{cases}$$

The Broyden family includes the DFP update (2.8) and the BFGS update (2.9), i.e., $\phi_{k-1} = 0$ (or $\phi_{k-1}^H = 1$ in (2.11)) corresponds to the BFGS update and $\phi_{k-1} = 1$ (or $\phi_{k-1}^H = 0$ in (2.11)) corresponds to the DFP update. The Broyden family with $\phi_{k-1} \in [0, 1]$ is called the convex class. If $\phi_{k-1} \in [0, 1]$, the approximate matrix by the Broyden family is positive definite. In the convex class, it is known that $\phi_k = 0$ (namely, the BFGS formula) is suggested as the best choice. On the other hand, Zhang and Tewarson [28] studied the preconvex class, which means the Broyden family with $\phi_{k-1} \in (\phi_{k-1}^*, 0)$, where ϕ_{k-1}^* is a threshold. As same as the convex class, the preconvex class generates positive definite approximate matrices. If $\phi_{k-1} = \phi_{k-1}^*$, then the approximate matrix degenerates. Zhang and Tewarson claimed that better choices than the BFGS formula could be found in the preconvex class. Note that (2.10) with $\phi_{k-1} \in [0, 1]$ corresponds to (2.11) with $\phi_{k-1}^H \in [0, 1]$ and (2.10) with $\phi_{k-1} \in (\phi_{k-1}^*, 0)$ corresponds to (2.11) with $\phi_{k-1}^H \in (1, \infty)$. Thus, (2.11) with

$\phi^H_{k-1} \in [0,1]$ and $\phi^H_{k-1} \in (1,\infty)$ is also called the convex class and the preconvex class, respectively.

In the memoryless quasi-Newton methods, we focus on the updating formula of H_k in (2.11). By replacing H_{k-1} with the identity matrix I, (2.11) becomes

$$\begin{cases} H_k = I - \dfrac{y_{k-1}y_{k-1}^T}{y_{k-1}^T y_{k-1}} + \dfrac{s_{k-1}s_{k-1}^T}{s_{k-1}^T y_{k-1}} + \phi^H_{k-1} w_{k-1} w_{k-1}^T, \\ w_{k-1} = \sqrt{y_{k-1}^T y_{k-1}} \left(\dfrac{s_{k-1}}{s_{k-1}^T y_{k-1}} - \dfrac{y_{k-1}}{y_{k-1}^T y_{k-1}} \right). \end{cases} \tag{2.12}$$

The quasi-Newton method based on (2.12) is called the memoryless quasi-Newton method based on the Broyden family. We note that we can compute the product of H_k and a vector v by using inner products of vectors and that it is written by

$$H_k v = v - \frac{y_{k-1}^T v}{y_{k-1}^T y_{k-1}} y_{k-1} + \frac{s_{k-1}^T v}{s_{k-1}^T y_{k-1}} s_{k-1} + \phi^H_{k-1}(w_{k-1}^T v) w_{k-1}.$$

Therefore, we can directly apply this technique to large-scale optimization problems.

To establish the global convergence of the method, we now modify the above memoryless quasi-Newton method by combining two types of secant conditions. The first one is the spectral-scaling secant condition by Cheng and Li [5], and the second one is given by Li and Fukushima [21]. The usual secant condition (2.7) is based on the first order approximation of g, namely $\nabla^2 f(x_k)s_{k-1} \approx y_{k-1}$. Cheng and Li considered the previous relation multiplied by $\gamma_k > 0$, namely, $\gamma_{k-1}\nabla^2 f(x_k)s_{k-1} \approx \gamma_{k-1}y_{k-1}$ and gave the spectral-scaling secant condition:

$$B_k s_{k-1} = \gamma_{k-1} y_{k-1} \quad \text{or} \quad H_k y_{k-1} = \frac{1}{\gamma_{k-1}} s_{k-1}, \tag{2.13}$$

where $\gamma_k > 0$ is a spectral-scaling parameter. Note that B_k approximates $\gamma_{k-1}\nabla^2 f(x_k)$ instead of $\nabla^2 f(x_k)$ and that γ_{k-1} is used for numerical stability. In this paper, we choose γ_{k-1} satisfying

$$\underline{\gamma} \le \gamma_{k-1} \le \overline{\gamma} \tag{2.14}$$

with positive constants $\underline{\gamma}$ and $\overline{\gamma}$. Next, we incorporate Li-Fukushima's modification into the spectral-scaling secant condition to preserve the positive definiteness of B_k (or H_k). Specifically, we use $z_{k-1} := y_{k-1} + \zeta_{k-1}s_{k-1}$ instead of y_{k-1} in (2.13), and then the modified spectral-scaling secant condition we consider is given by

$$B_k s_{k-1} = \gamma_{k-1} z_{k-1} \quad \text{and} \quad H_k z_{k-1} = \frac{1}{\gamma_{k-1}} s_{k-1}.$$

The additional term $\zeta_{k-1}s_{k-1}$ can be regarded as a regularizer, and the parameter ζ_{k-1} is chosen such that there exist positive constants $\underline{\zeta}$ and $\overline{\zeta}$ satisfying the relations

$$s_{k-1}^T z_{k-1} = s_{k-1}^T(y_{k-1} + \zeta_{k-1}s_{k-1}) \geq \underline{\zeta}\|s_{k-1}\|^2 \tag{2.15}$$

and

$$0 \leq \zeta_{k-1} \leq \overline{\zeta}, \tag{2.16}$$

where $\|\cdot\|$ denotes the l_2 norm. Replacing y_{k-1} by $\gamma_{k-1}z_{k-1}$ in (2.12), we have a memoryless Broyden family based on the modified spectral-scaling secant condition, as follows:

$$\begin{cases} H_k = I - \dfrac{z_{k-1}z_{k-1}^T}{z_{k-1}^T z_{k-1}} + \dfrac{1}{\gamma_{k-1}} \dfrac{s_{k-1}s_{k-1}^T}{s_{k-1}^T z_{k-1}} + \phi_{k-1}^H w_{k-1}w_{k-1}^T, \\ w_{k-1} = \sqrt{z_{k-1}^T z_{k-1}} \left(\dfrac{s_{k-1}}{s_{k-1}^T z_{k-1}} - \dfrac{z_{k-1}}{z_{k-1}^T z_{k-1}} \right). \end{cases} \tag{2.17}$$

As in (2.11), (2.17) with $\phi_{k-1}^H = 1$ corresponds to the memoryless BFGS formula, and (2.17) with $\phi_{k-1}^H = 0$ corresponds to the memoryless DFP formula. Also we call (2.17) with $\phi_{k-1}^H \in [0, 1]$ and $\phi_{k-1}^H \in (1, \infty)$ the convex class and the preconvex class, respectively. We emphasize that if $\phi_{k-1}^H \in [0, \infty)$, H_k in (2.17) is positive definite. In this paper, ϕ_{k-1}^H is chosen such that

$$0 \leq \phi_{k-1}^H \leq \overline{\phi} \tag{2.18}$$

holds, where $\overline{\phi}$ is a fixed positive constant.

Summarizing the above arguments, we propose a new active-set method for solving problem (1.1). Specifically, we adopt the search direction (2.4)–(2.6) in which we use (2.17) for H_k in (2.5). In (1.2), we choose a step size η_k satisfying the Armijo condition by a backtracking technique. That is, for a constant $\beta \in (0, 1)$, find the smallest integer $i = 0, 1, \ldots$ such that

$$f(x_k + \beta^i d_k) \leq f(x_k) + \sigma\beta^i g_k^T d_k \tag{2.19}$$

holds and set $\eta_k = \beta^i$. We present our algorithm as follows.

Algorithm 1. (Active-set memoryless quasi-Newton method based on spectral-scaling Broyden family)

Step 0: Given a starting point $x^0 \in K$, and constants $\sigma \in (0, 1)$ and $\beta \in (0, 1)$. Set $k = 0$.
Step 1: Compute the search direction d_k in (2.4)–(2.6) by using H_k in (2.17).
Step 2: If a stopping criterion is satisfied, we stop the algorithm.
Step 3: Find a step size η_k by (2.19).

Step 4: Set $x_{k+1} = x_k + \eta_k d_k$.
Step 5: Set $k \coloneqq k + 1$ and go to step 1.

Now, we give the global convergence property of the proposed method. For this purpose, we make the following assumptions.

Assumption 1. The level set $L^0 = \{x \in R^n : f(x) \leq f(x^0)\} \cap K$ is compact.

Assumption 2. The gradient g of f is Lipschitz continuous on L^0, namely, there exists a positive constant L such that

$$\|g(u) - g(v)\| \leq L\|u - v\| \quad \forall u, \forall v \in L^0. \tag{2.20}$$

Assumption 3. For all $k \geq 1$, s_{k-1} and z_{k-1} are linearly independent.

We can obtain the largest and smallest eigenvalues of (2.17) from the result in Al-Baali [1]. Estimating the eigenvalues, we have the following proposition, which implies that H_k in (2.17) is uniformly positive definite and bounded above.

Proposition 2.1. *Suppose that Assumptions* 1–3 *hold and* (2.14), (2.15) *and* (2.18) *are satisfied. Then, there exist positive constants* m *and* M *such that* H_k *in* (2.17) *satisfies*

$$m\|x\|^2 \leq x^T H_k x \leq M\|x\|^2 \ \forall x \in R^n.$$

The next proposition gives a necessary and sufficient condition for x_k to be a stationary point of (1.1).

Proposition 2.2. *Let the sequence* $\{x_k\}$ *be generated by Algorithm* 1. *Suppose that all assumptions of Proposition* 2.1 *hold. Then* x_k *is a stationary point of* (1.1) *if and only if* $d_k = 0$.

The following lemma is given by Facchinei et al. [10] for a general iterative method (1.2) and plays an important role in showing the global convergence.

Lemma 2.3. *Let the sequence* $\{x_k\}$ *be generated by* (1.2), *where* η_k *is computed by* (2.19). *Suppose that Assumption 1 holds and that there exist scalars* $\mu > 0$ *and* $p > 1$ *such that, for every* $k = 0, 1, \ldots,$ *the search direction* $d_k \in R^n$ *satisfies the following conditions:*

a. $x_k + d_k \in K$,
b. $g_k^T d_k \leq -\mu\|d_k\|^p$,
c. $d_k = 0$ *if and only if* x_k *is a stationary point of* (1.1),
d. *if* $x_k \to \bar{x}$ *and* $d_k \to 0$, *then* $\overline{x}$ *is a stationary point of* (1.1).

Then the sequence $\{x_k\}$ *has at least an accumulation point and every accumulation point of this sequence is a stationary point for* (1.1).

Using Lemma 2.3, we have the global convergence theorem of our proposed method.

Theorem 2.4. *Let the sequence* $\{x_k\}$ *be generated by Algorithm* 1*. Suppose that if* $x_k \to \bar{x}$*, then* $F(x_k) = \{i \mid l_i < (\bar{x})_i < u_i\}$ *holds for* k *sufficiently large. Suppose that all assumptions of Proposition* 2.1 *hold. Then the sequence* $\{x_k\}$ *has at least an accumulation point and every accumulation point of this sequence is a stationary point for* (1.1)*.*

3. Numerical experiments

In this section, we report numerical results to investigate numerical performance of our method. In CUTEst [14], there are 65 bound constrained optimization problems. However, the initial points of four problems are not in the feasible regions. Therefore, we removed these problems and tested 61 problems. All codes were written in Python 3.7 with PyCUTEst [12]. PyCUTEst is a Python interface to CUTEst. They were run on a PC with 3.5GHz Intel Core i5, 32.0 GB RAM memory and Linux OS Ubuntu 16. We stopped the algorithm if $\|P_K(g_k)\|_\infty < 10^{-5}$ held or if CPU time exceeded 600 seconds or if a numerical overflow occurred. Here, $P_K(g_k)$ means

$$P_K(g_k) = \begin{cases} \min(0, (g_k)_i) & (x_k)_i = l_i, \\ (g_k)_i & l_i < (x_k)_i < u_i, \\ \max(0, (g_k)_i) & (x_k)_i = u_i. \end{cases}$$

By taking into account (2.1), $P_K(g_k) = 0$ implies that x_k is a stationary point.

We set parameters to be $a_i(x) = b_i(x) = 10^{-6}$, $\gamma_{k-1} = \dfrac{s_{k-1}^T z_{k-1}}{z_{k-1}^T z_{k-1}}$, $\sigma = 10^{-4}$, $\beta = 0.5$ and

$$\zeta_{k-1} = \begin{cases} 0, & \text{if } s_{k-1}^T y_{k-1} \geq \underline{\zeta}\|s_{k-1}\|^2 \\ \underline{\zeta} - \dfrac{s_{k-1}^T y_{k-1}}{\|s_{k-1}\|^2}, & \text{otherwise,} \end{cases} \tag{3.1}$$

where we used $\underline{\zeta} = 0.01$. When $s_{k-1}^T y_{k-1} \geq \underline{\zeta}\|s_{k-1}\|^2$ holds, ζ_{k-1} in (3.1) obviously satisfies (2.15) and (2.16). Otherwise, we have

$$\begin{aligned} s_{k-1}^T z_{k-1} = s_{k-1}^T (y_{k-1} + \zeta_{k-1} s_{k-1}) &= s_{k-1}^T y_{k-1} + \underline{\zeta}\|s_{k-1}\|^2 - s_{k-1}^T y_{k-1} \\ &= \underline{\zeta}\|s_{k-1}\|^2 \end{aligned}$$

and

$$\zeta_{k-1} = \underline{\zeta} - \frac{s_{k-1}^T y_{k-1}}{\|s_{k-1}\|^2} > 0.$$

Also, from (2.20), we get

$$\zeta_{k-1} = \underline{\zeta} - \frac{s_{k-1}^T y_{k-1}}{\|s_{k-1}\|^2} \le \underline{\zeta} + \frac{L\|s_{k-1}\|^2}{\|s_{k-1}\|^2} = \underline{\zeta} + L.$$

Therefore, ζ_{k-1} in (3.1) satisfies (2.15) and (2.16).

To compare numerical performance between the tested methods, we adopted the performance profiles based on the CPU time by Dolan and Moré [8]. In [8], the performance profile is explained as follows.

> For n_s solvers and n_p problems, the performance profiles $P : \mathbb{R} \to [0,1]$ is defined as follows: Let $\mathcal{P}$ and $\mathcal{S}$ be the set of problems and the set of solvers, respectively. For each problem $p \in \mathcal{P}$ and for each solver $s \in \mathcal{S}$, we define $t_{p,s}$ = CPU time required to solve problem p by solver s. The performance ratio is given by $r_{p,s} = t_{p,s}/\min_s t_{p,s}$. Then, the performance profile is defined by $P(\tau) = \frac{1}{n_p}\mathrm{size}\{p \in \mathcal{P} | r_{p,s} \le \tau\}$, for all $\tau \ge 1$, where size A, for any set A, stands for number of the elements in that set. Note that $P(\tau)$ is the probability for solver $s \in \mathcal{S}$ such that a performance ratio $r_{p,s}$ is within a factor $\tau \ge 1$ of the best result. The left side value of the performance profile, namely $P(1)$, gives the percentage of the test problems for which a method is the best result. The right side value, namely $P(\tau)$ with sufficient large τ, gives the percentage of the test problems that are successfully solved by each of the methods. The top curve is the method that solves the most problems in a result that is within a factor τ of the best result.

To investigate the difference of Yuan-Lu's active-set strategy (2.2)–(2.3) and the modified strategy (2.4)–(2.6), we tested the methods given in Table 1.

Table 1. Tested methods with or without the modification

method	
Method 1	Algorithm 1 ($\phi_{k-1}^H = 1.0$)
Method 2	Algorithm 1 ($\phi_{k-1}^H = 1.0$) with (2.2)–(2.3) instead of (2.4)–(2.6)

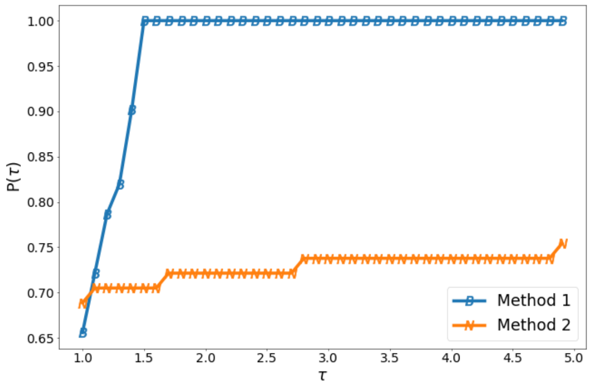

FIGURE 1. Performance profiles of Table 1

In Figure 1, Method 1 is clearly superior to Method 2. From this fact, we can see that the modification of the direction for $i \in T(x_k)$ works well. Actually, Method 1 could successfully solve 18 problems which could not be solved by Method 2. For these problems, Method 2 stopped even if x_k was not a stationary point.

Next, we tested our method with the parameter $\phi^H_{k-1} = 0$ (DFP), 0.25, 0.5, 0.75, 1 (BFGS), 1.25, 1.5 and 1.75. Note that the method with $\phi^H_{k-1} = 0$, 0.25, 0.5, 0.75 and 1.0 is the convex class and that with 1.25, 1.5 and 1.75 is the preconvex class (see Table 2).

TABLE 2. Tested methods with convex and preconvex classes

method	active-set strategy	class
$\phi^H_{k-1} = 0$ (DFP)	(2.4)–(2.6)	convex
$\phi^H_{k-1} = 0.25$	(2.4)–(2.6)	convex
$\phi^H_{k-1} = 0.5$	(2.4)–(2.6)	convex
$\phi^H_{k-1} = 0.75$	(2.4)–(2.6)	convex
$\phi^H_{k-1} = 1.0$ (BFGS)	(2.4)–(2.6)	convex
$\phi^H_{k-1} = 1.25$	(2.4)–(2.6)	preconvex
$\phi^H_{k-1} = 1.5$	(2.4)–(2.6)	preconvex
$\phi^H_{k-1} = 1.75$	(2.4)–(2.6)	preconvex

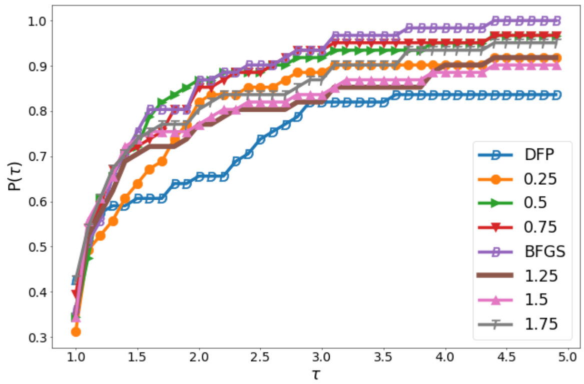

FIGURE 2. Performance profiles of Table 2

In Figure 2, $\phi_{k-1}^H = 1.0$ (BFGS) is better than the other methods. On the other hand, $\phi_{k-1}^H = 0$ (DFP) does not perform well. In the convex class, the numerical performance becomes better as ϕ_{k-1}^H is close to 1.0. In the preconvex class, we can see that numerical performance becomes better as ϕ_{k-1}^H becomes larger. The convex class is better than the preconvex class.

4. Conclusion

We have proposed an active-set memoryless quasi-Newton method based on the spectral-scaling Broyden family for bound constrained optimization problems and have shown its global convergence. The modification of the search direction for $i \in T(x_k)$ gives a better effect on numerical performance.

Acknowledgements. This research was supported in part by JSPS KAKENHI (grant number 20K11698) and the Research Institute for Mathematical Sciences in Kyoto University.

References

[1] M. Al-Baali, *On measure functions for the self-scaling updating formulae for quasi-Newton methods*, Optimization **32** (1995), 59–69.
[2] D. P. Bertsekas, *Projected Newton methods for optimization problems with simple constraints*, SIAM Journal on Control and Optimization **20** (1982), 221–246.
[3] R. H. Byrd, P. Lu, J. Nocedal and C. Zhu, *A limited memory algorithm for bound constrained optimization*, SIAM Journal on Scientific Computing **16** (1995), 1190–1208.

[4] Z. Chen and W. Cheng, *Spectral-scaling quasi-Newton methods with updates from the one parameter of the Broyden family*, Journal of Computational and Applied Mathematics **248** (2013), 88–98.

[5] W. Y. Cheng and D. H. Li, *Spectral scaling BFGS method*, Journal of Optimization Theory and Applications **146** (2010), 305–319.

[6] T. F. Coleman and Y. Li, *An interior trust region approach for nonlinear minimization subject to bounds*, SIAM Journal on Optimization **6** (1996), 418–445.

[7] Y. H. Dai and R. Fletcher, *Projected Barzilai-Borwein methods for large-scale box-constrained quadratic programming*, Numerical Mathematik **100** (2005) 21–47.

[8] E. D. Dolan and J. J. Moré, *Benchmarking optimization software with performance profiles*, Mathematical Programming **91** (2002), 201–213.

[9] Z. Dostál, A. Friedlander and S. A. Santos, *Augmented Lagrangians with adaptive precision control for quadratic programming with simple bounds and equality constraints*, SIAM Journal on Optimization **13** (2003), 1120–1140.

[10] F. Facchinei, J. Júdice and J. Soares, *An active set Newton algorithm for large-scale nonlinear programs with box constraints*, SIAM Journal on Optimization **8** (1998), 158–186.

[11] F. Facchinei and S. Lucidi, *A class of penalty functions for optimization problems with bound constraints*, Optimization **26** (1992), 239–259.

[12] J. Fowkes and L. Roberts, *PyCUTEst: Python interface to the CUTEst optimization test environment*, `https://jfowkes.github.io/pycutest/_build/html/index.html`, (last access 2020/3/12).

[13] W. Glunt, T. L. Hayden and M. Raydan, *Molecular conformations from distance matrices*, Journal of Computational Chemistry **14** (1993), 114–120.

[14] N. I. M. Gould, D. Orban and Ph. L. Toint, *CUTEst: a Constrained and Unconstrained Testing Environment with safe threads for mathematical optimization*, Computational Optimization and Applications **60** (2015), 545–557.

[15] W. W. Hager, B. A. Mair and H. Zhang, *An affine-scaling interior-point CBB method for box-constrained optimization*, Mathematical Programming **119** (2009), 1–32.

[16] W. W. Hager and H. Zhang, *A new active set algorithm for box constrained optimization*, SIAM Journal on Optimization **17** (2006), 526–557.

[17] O. Härkegård, *Efficient active set algorithms for solving constrained least squares problems in aircraft control allocation*, in: Proceedings of the 41st IEEE conference on decision and control, Las Vegas, NV. December, 2002, pp. 1295–1300.

[18] S. Babaie-Kafaki, *On optimality of the parameters of self-scaling memoryless quasi-Newton updating formula*, Journal of Optimization Theory and Application **167** (2015), 91–101.

[19] D. Kim, S. Sra and I. S. Dhillon, *Tackling box-constrained optimization via a new projected quasi-Newton approach*, SIAM Journal on Scientific Computing **32** (2010), 3548–3563.

[20] C. X. Kou and Y. H. Dai, *A modified self-scaling memoryless Broyden-Fletcher-Goldfarb-Shanno method for unconstrained optimization*, Journal of Optimization Theory and Applications **165** (2015), 209–224.

[21] D. H. Li and M. Fukushima, *A modified BFGS method and its global convergence in nonconvex minimization*, Journal of Computational and Applied Mathematics **129** (2001), 15–35.

[22] A. U. Moyi and W. J. Leong, *A sufficient descent three-term conjugate gradient method via symmetric rank-one update for large-scale optimization*, Optimization **65** (2016), 121–143.

[23] S. Nakayama, Y. Narushima and H. Yabe, *A memoryless symmetric rank-one method with sufficient descent property for unconstrained optimization*, Journal of the Operations Research Society of Japan **61** (2018), 53–70.

[24] S. Nakayama, Y. Narushima and H. Yabe, *Memoryless quasi-Newton methods based on spectral-scaling Broyden family for unconstrained optimization*, Journal of Industrial and Management Optimization **15** (2019), 1773–1793.

[25] D. F. Shanno, *Conjugate gradient method with inexact searches*, Mathematics of Operations Research **3** (1978), 244–256.

[26] S. Sra, S. Nowozin and S. J. Wright (eds.), *Optimization for Machine Learning*, The MIT Press, Cambridge, 2012.

[27] G. Yuan and X. Lu, *An active set limited memory BFGS algorithm for bound constrained optimization*, Applied Mathmatical Modelling **35** (2011), 3561–3573.

[28] Y. Zhang and R. P. Tewarson, *Quasi-Newton algorithms with updates from the preconvex part of Broyden's family*, IMA Journal of Numerical Analysis **8** (1988), 487–509.

H. Nishio
Tokyo University of Science, Japan

S. Nakayama
Chuo University, Japan
E-mail address: shummin@kc.chuo-u.ac.jp

Y. Narushima
Keio University, Japan
E-mail address: narushima@ae.keio.ac.jp

H. Yabe
Tokyo University of Science, Japan
E-mail address: yabe@rs.tus.ac.jp