Numerical Algorithms
https://doi.org/10.1007/s11075-023-01626-6

ORIGINAL PAPER

An adaptive projection BFGS method for nonconvex unconstrained optimization problems

Gonglin Yuan[1] · Xiong Zhao[1] · Kejun Liu[1] · Xiaoxuan Chen[1]

Received: 7 April 2023 / Accepted: 13 July 2023

Abstract
The BFGS method is a common and effective method for solving unconstrained optimization problems in quasi-Newton algorithm. However, many scholars have proven that the algorithm may fail in some cases for nonconvex problems under Wolfe conditions. In this paper, an adaptive projection BFGS algorithm is proposed naturally which can solve nonconvex problems, and the following properties are shown in this algorithm: ① The generation of the step size α_j satisfies the popular Wolfe conditions; ② a specific condition is proposed which has sufficient descent property, and if the current point satisfies this condition, the ordinary BFGS iteration process proceeds as usual; ③ otherwise, the next iteration point x_{j+1} is generated by the proposed adaptive projection method. This algorithm is globally convergent for nonconvex problems under the weak-Wolfe-Powell (WWP) conditions and has a superlinear convergence rate, which can be regarded as an extension of projection BFGS method proposed by Yuan et al. (J. Comput. Appl. Math. 327:274-294, 2018). Furthermore, the final numerical results and the application of the algorithm to the Muskingum model demonstrate the feasibility and competitiveness of the algorithm.

Keywords BFGS method · Nonconvex functions · Adaptive projection technique · Global convergence · Muskingum model

✉ Xiong Zhao
chao18874188217@163.com

Gonglin Yuan
glyuan@gxu.edu.cn

Kejun Liu
kejun_l@163.com

Xiaoxuan Chen
xxchen1195@163.com

1 School of Mathematics and Information Science, Center for Applied Mathematics of Guangxi, Guangxi University, Nanning, Guangxi 530004, China

Published online: 05 August 2023

1 Introduction

Consider problem

$$\min\{\varphi(x) \ | \ x \in \mathscr{R}^n\}, \tag{1.1}$$

where $\varphi : \mathscr{R}^n \rightarrow \mathscr{R}$ and $\varphi(x)$ is continuously differentiable. This model of optimization problems is used in many areas, such as engineering and finance (see [13, 14, 20, 27, 28, 36, 37], etc.). There are many methods to solve (1.1) such as conjugate gradient method [19, 34] and quasi-Newton method [2, 39]. For the sake of illustration, the following notes are made: $||.||$ denotes the Euclidean norm, φ_j denotes the function value $\varphi(x_j)$, and the gradient functuon value $g(x_j)$ is replaced by g_j. The BFGS algorithm for solving (1.1) are based on the following iterations:

$$x_{j+1} = x_j + \alpha_j d_j, \, j \in \mathscr{N}, \tag{1.2}$$

where step size α_j is generated by inexact line search mechanisms, and d_j represents the search direction of the current point x_j which can be gotten by solving this equation $B_j d_j = -g_j$, where B_j is an approximation to the Hessian matrix $\nabla^2\varphi(x_j)$ at x_j. The most widely used quasi-Newton update formula, the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method [3, 11, 16, 31], updates B_j via

$$B_{j+1} = B_j - \frac{B_j s_j s_j^T B_j}{s_j^T B_j s_j} + \frac{y_j y_j^T}{s_j^T y_j}, \tag{1.3}$$

where $s_j = x_{j+1} - x_j$ and $y_j = g_{j+1} - g_j$. Widely used weak Wolfe-Powell (WWP) condition in inexact line searches is as follows:

$$\varphi(x_j + \alpha_j d_j) - \varphi(x_j) \leq \zeta_1 \alpha_j g(x_j)^T d_j, \tag{1.4}$$

$$g(x_j + \alpha_j d_j)^T d_j \geq \zeta_2 g(x_j)^T d_j, \tag{1.5}$$

where $0 < \zeta_1 < \zeta_2 < 1$.

Powell [30] firstly proved that the BFGS algorithm converges globally for convex problems under Wolfe conditions. Subsequently, some scholars [6] extended these to Broyden family algorithms. In addition, there were many global convergence results [4, 5, 17, 32] and local convergence results [8, 9, 18] for BFGS method on convex functions. Many modified quasi-Newton update formula [7, 21, 22, 29, 35, 41, 44] had been presented to obtain better results.

Nowadays, there are still two unanswered questions about the quasi-Newton method under the WWP conditions: ① The first is whether the Davidon-Fletcher-Powell (DFP) algorithm can converge globally for convex problems [12], ② and the other is whether the BFGS algorithm can converge globally for non-convex problems. In the paper, we wish to construct a modified BFGS algorithm which will converge globally for the non-convex problems with the WWP condition and perform well in numerical experiments. At present, projection technology is used in many fields [24, 46]. It also has many applications in optimization problems which can obtain better theoretical properties in some poor cases [23, 33, 39, 40, 43, 45]. A novel projection technique will be proposed in this modified quasi-Newton formula.

The motivation and algorithm of this paper are shown in the next section. In Section 3, the proposed algorithm will be shown to be globally convergent. Experimental results of the proposed algorithm and its applications in Muskingum model will be present in the last two sections respectively.

2 Motivation and algorithm

The following conclusion can easily be deduced from strongly convex function

$$s_j^T y_j \geq \rho \|s_j\|^2 > 0, \tag{2.1}$$

where $\rho > 0$ is a coefficient related to the strong convexity of the function. If the curvature condition (2.1) holds, the BFGS algorithm can converge globally, while for general convex functions, we can only have the following condition: $s_j^T y_j > 0$, which can insure the next iteration matrix B_{j+1} of BFGS algorithm inherits the positive definiteness of the previous matrix B_j, and it is one of the conditions for the BFGS algorithm to converge globally. However, inequality (2.1) is difficult to satisfy for nonconvex problems. Therefore, in order to obtain it, many scholars have proposed modified BFGS formulas. For example, the BFGS update rule modified by Li and Fukushima [22] is shown below:

$$B_{j+1} = \begin{cases} B_j - \frac{B_j s_j s_j^T B_j}{s_j^T B_j s_j} + \frac{y_j y_j^T}{y_j^T s_j} & \text{if } \frac{y_j^T s_j}{\|s_j\|^2} \geq C_1 \|g_j\|^{C_2} \\ B_j & \text{otherwise ,} \end{cases} \tag{2.2}$$

where $C_1 > 0$ and $C_2 > 0$ are constants.

Besides that, under certain conditions, the sufficient descent property

$$d_j^T g_j \leq m < 0 \tag{2.3}$$

can also get (2.1), where m is a constant. Based on this, Yuan et al. [39] proposed a projection technique under a modified weak-Wolfe-Powell conditions [42]. For the special quadratic function $\varphi(x) = b^T x + \frac{1}{2} x^T H x + c$, where H is an $n \times n$ symmetric matrix, $b \in \mathscr{R}^n$ and $c \in \mathscr{R}$, the function is a uniformly convex function, and BFGS method has good convergence for solving it since the function has many good features. Therefore, they use some good properties of the uniformly convex function in the BFGS method to obtain some ideal convergence. Their idea is to project the point that does not satisfy the sufficient descent condition onto a paraboloid by projection method, so as to establish the convergence of the algorithm. Such a paraboloid is defined in the article:

$$\{x \in \mathscr{R}^n | \lambda \|\mathscr{W}_j - x\|^2 + (\mathscr{W}_j - x)^T g(\mathscr{W}_j) = 0\}, \tag{2.4}$$

where $\lambda > 0, \mathscr{W}_j = x_j + \alpha_j d_j$. $\lambda\|\mathscr{W}_j - x\|^2 + (\mathscr{W}_j - x)^T g(\mathscr{W}_j)$ can be regarded as the combination of the first-order and second-order expansion terms of a strong convex function at $\mathscr{W}_j$, where 2λ can be an eigenvalue of its Hessian matrix. The next point created by the projection technique is shown below:

$$x_{j+1} = x_j + \frac{\lambda\|\mathscr{W}_j - x_j\|^2 + (\mathscr{W}_j - x_j)^T g(\mathscr{W}_j)}{\|g(\mathscr{W}_j) - g(x_j)\|^2}[g(\mathscr{W}_j) - g(x_j)]. \tag{2.5}$$

With this modified WWP line search technique, the Polak-Ribière-Polyak (PRP) conjugate gradient method for nonconvex functions can also converge globally when using the above paraboloid definition and the following projection update formulas [43]:

$$x_{j+1} = x_j + \frac{g(\mathscr{W}_j)^T(\mathscr{W}_j - x_j) + \lambda\|\mathscr{W}_j - x_j\|^2}{\|g(x_j)\|}g(x_j). \tag{2.6}$$

However, the assumptions and proofs about these projection-like methods [39, 40, 43] are actually incomplete, and we need to complete them. Besides that, is it possible to prove the global convergence of the projection BFGS algorithm using a general WWP line search? And it is important to note that these projection-like methods do not involve a discussion of the convergence rate.

These observations motivate us to invent an adaptive projection technique to overcome situations that do not satisfy the condition we specified, so that the modified BFGS method (or the PRP conjugate gradient method) for nonconvex functions can converge globally under the general WWP line search condition, which can also update different projection formulas adaptively according to different problems. Correspondingly, we define an index set that satisfy the special sufficient descent condition

$$SD := \{j | d_j^T g_j \leq -\rho\alpha_j \|d_j\|^2 \|g_j\|^\alpha, j \geq 0\}, \tag{2.7}$$

where ρ is a positive constant, $\alpha \in (\infty, +\infty)$ is a tuning parameter, and $\|g_j\|^\alpha$ can be viewed as an adaptive term. This definition also helps to explore the convergence rate of the algorithm. For points that do not satisfy the sufficient descent condition, we will use the projection method to get a new direction, so the definition of the adaptive projection surface needs to be given first:

$$\{x \in \mathscr{R}^n | \mu\|\mathscr{V}_j - x\|^2 \|g(x)\|^\alpha + (\mathscr{V}_j - x)^T g(\mathscr{V}_j) = 0\}, \tag{2.8}$$

where $\mathscr{V}_j = x_j + \alpha_j d_j$ and parameter $\mu > 0$. We can get the paraboloid that we mentioned above when $\alpha = 0$. Next, iterative process will be briefly described. For the current point x_j, the generation of the next point x_{j+1} will be divided into two cases:

Case (a) $: j \in SD$. Iteration proceeds as usual $x_{j+1} = \mathscr{V}_j$.

Case (b) $: j \notin SD$. First, we project the current point x_j onto the surface (2.8), then the next iteration x_{j+1} is defined below:

$$x_{j+1} = x_j + \frac{P_j}{\|g(\mathscr{V}_j) - g(x_j)\|^2}[g(\mathscr{V}_j) - g(x_j)], \tag{2.9}$$

where

$$P_j = \mu\|\mathscr{V}_j - x_j\|^2\|g(x_j)\|^\alpha + (\mathscr{V}_j - x_j)^T g(\mathscr{V}_j). \tag{2.10}$$

An adaptive projection method is presented for solving (1.1) as Algorithm 1.

Algorithm 1 Adaptive projection technique-BFGS algorithm for nonconvex unconstrained optimization problems (APT-BFGS)

step 1 : Choose an initial point $x_0 \in \mathscr{R}^n$, an initial symmetric and positive definite matrix $B_0 \in \mathscr{R}^{n\times n}$
step 2 : Given the necessary parameters $\rho > 0$, $\zeta_1 \in (0, \frac{1}{2})$, $\epsilon \in (0, \frac{1}{2})$, $\zeta_2 \in (\zeta_1, 1)$, $\mu \in (\zeta_2\rho, +\infty)$, $\alpha \in (-\infty, +\infty)$, and let $j := 0$.
step 3 : Calculate $\|g_j\|$, if $\|g_j\| \le \epsilon$ is true, the algorithm stops
step 4 : Get the direction d_j by solving the linear equation

$$B_j d_j = -g_j \tag{2.11}$$

step 5 : Determine a step size $\alpha_j > 0$ that satisfies the WWP condition (1.4) and (1.5).
step 6 : Let $\mathscr{V}_j := x_j + \alpha_j d_j$.
step 7 : If $j \in SD$. Let $x_{j+1} := \mathscr{V}_j$, $s_j := x_{j+1} - x_j$ and $y_j := g(x_{j+1}) - g(x_j)$, then skip to Step 9.
step 8 : Else $j \notin SD$. Define the next iteration x_{j+1} as shown in (2.9), $s_j := x_{j+1} - x_j$ and $y_j := g(\mathscr{V}_j) - g(x_j)$.
step 9 : Determine $B_{j+1}$by the ordinary BFGS formula

$$B_{j+1} = B_j - \frac{B_j s_j s_j^T B_j}{s_j^T B_j s_j} + \frac{y_j y_j^T}{s_j^T y_j}. \tag{2.12}$$

.
step 10 : Let $j := j + 1$ and, return to Step 3

---

3 Global convergence of APT-BFGS

This section aims to analyze the global convergence and convergence rate of Algorithm 1. First, the required assumptions and lemmas are given as follows:

Assumption 1 The level set $S_0 = \{x \in \mathscr{R}^n | \varphi(x) \le \varphi(x_0)\}$ is bounded, and $\varphi(x)$ is lower bounded by $\varphi(x^*) \in \mathscr{R}$. And the same order between its gradient and the variables holds, i.e.,

$$\|g(\dot{x}) - g(\ddot{x})\| = O(\|\dot{x} - \ddot{x}\|), \quad \forall \dot{x}, \ddot{x} \in \mathscr{R}^n. \tag{3.1}$$

Assumption 2 When $j \notin SD$, there will always be at least one step-size α_j that satisfies the following condition:

$$\varphi(x_{j+1}) - \varphi(x_j) \le \zeta_1 \alpha_j g(x_j)^T d_j. \tag{3.2}$$

Remark 1 It can be seen from the Assumption 1 that $g(x)$ is bounded on the level set S_0, that is, there is a positive constant $G_b > 0$ satisfying

$$\|g(\dot{x})\| \le G_b, \quad \forall \dot{x} \in S_0. \tag{3.3}$$

From Assumption 2 and (1.4) we can infer that

$$\varphi(x_{j+1}) - \varphi(x_j) \le \zeta_1 \alpha_j g(x_j)^T d_j. \tag{3.4}$$

is true for both **Cases** (**a**) and (**b**).

Lemma 1 **Case** (**a**) $j \in SD$. *Suppose that Assumptions 1 hold. If* $\{x_j\}$, $\{d_j\}$, $\{\alpha_j\}$ *and* $\{g_j\}$ *are generated by Algorithm 1, then the following inequality holds:*

$$s_j^T y_j \ge \eta_1 \|\alpha_j d_j\|^2 \|g_j\|^\alpha, \quad \forall j \ge 0, \tag{3.5}$$

where $\eta_1 > 0$ *is a constant.*

Proof Define $\eta_1 := (1 - \zeta_2)\rho$. From the step 7 of Algorithm 1, we can easily obtain

$$\begin{aligned} s_j^T y_j &= s_j^T [g(\mathscr{V}_j) - g(x_j)] && (3.6)\\ &= s_j^T [g(x_{j+1}) - g_j] && (3.7)\\ &\ge -(1 - \zeta_2) g(x_j)^T s_j && (3.8)\\ &= \alpha_j (1 - \zeta_2)(-g_j^T d_j) && (3.9)\\ &\ge (1 - \zeta_2)\rho \|\alpha_j d_j\|^2 \|g_j\|^\alpha, && (3.10) \end{aligned}$$

where the first and the last inequalities follow the second WWP line search condition (1.5) and the definition of the sufficient descent set SD, respectively. Then, the proof is completed. □

Lemma 2 **Case** (**b**) $j \notin SD$. *Suppose that Assumptions 1 hold. If* $\{x_j\}$, $\{d_j\}$, $\{\alpha_j\}$ *and* $\{g_j\}$ *are generated by Algorithm 1, then the following inequality holds:*

$$s_j^T y_j \ge \eta_2 \|\alpha_j d_j\|^2 \|g_j\|^\alpha, \quad \forall j \ge 0, \tag{3.11}$$

where $\eta_2 > 0$ *is a constant.*

Proof Define $\eta_2 := (\mu - \zeta_2\rho) > 0$. According to the step 8 of Algorithm 1 and the definition of P_j, we can derive that

$$\begin{aligned}
P_j &= g(\mathscr{V}_j)^T(\mathscr{V}_j - x_j) + \mu\|\mathscr{V}_j - x_j\|^2\|g_j\|^\alpha \\
&= \alpha_j g(x_j + \alpha_j d_j)^T d_j + \mu\|\alpha_j d_j\|^2\|g_j\|^\alpha \\
&\geq \alpha_j \zeta_2 g(x_j)^T d_j + \mu\|\alpha_j d_j\|^2\|g_j\|^\alpha \\
&> \alpha_j \zeta_2(-\rho\alpha_j\|d_j\|^2\|g_j\|^\alpha) + \mu\|\alpha_j d_j\|^2\|g_j\|^\alpha \\
&= (\mu - \zeta_2\rho)\|\alpha_j d_j\|^2\|g_j\|^\alpha,
\end{aligned}$$

where the second inequality can be deduced from the condition $j \notin SD$. And

$$\begin{aligned}
s_j^T y_j &= \frac{P_j}{\|g(\mathscr{V}_j) - g(x_j)\|^2}\|g(\mathscr{V}_j) - g(x_j)\|^2 \\
&= P_j \\
&> (\mu - \zeta_2\rho)\|\alpha_j d_j\|^2\|g_j\|^\alpha
\end{aligned}$$

holds, where the second inequality can be derived from gradient Lipschitz condition (3.1). The proof is completed. □

Remark 2 The sufficient condition to guarantee the positive definiteness of sequence $\{B_j\}$ is $s_j^T y_j > 0$ $(\forall j \geq 0)$. Since $\|g_j\| \neq 0$ during the execution of the Algorithm 1, according to (3.5) and (3.11) we can always get $s_j^T y_j > 0$. Therefore, it can be known from (2.11) that

$$d_j^T g_j < 0, \quad \forall j \geq 0 \tag{3.12}$$

is true. From (3.12) and (3.4), we can easily deduce that $\{\varphi_j\}$ is a descending sequence.

Lemma 3 *Consider Lemmas 1 and 2. Suppose Assumptions 1 hold. If sequences $\{x_j\}$, $\{d_j\}$, $\{\alpha_j\}$, and $\{g_j\}$ are generated by Algorithm 1, then the following inequality holds:*

$$\sum_{j=0}^{\infty} \frac{(g_j^T d_j)^2}{\|d_j\|^2} < \infty. \tag{3.13}$$

Proof Given the condition (3.4)

$$\varphi(x_j) \geq \varphi(x_{j+1}) - \zeta_1\alpha_j g_j^T d_j, \quad \forall j \geq 0,$$

adding up the above inequalities from $k = 0$ to ∞ and using *Remark 2* can obtain

$$0 < \sum_{j=0}^{\infty}(-\alpha_j g_j^T d_j) < \infty. \tag{3.14}$$

Then, there exists a positive constant L_0 that satisfies the inequality

$$L_0\alpha_j\|d_j\|^2 \geq [g(x_j+\alpha_j d_j)-g(x_j)]^T d_j \geq -(1-\zeta_2)g_j^T d_j > 0, \quad \forall j \geq 0, \tag{3.15}$$

where the first inequality is derived from Cauchy inequality and Assumption 1. Thus, $\alpha_j \geq \dfrac{-(1-\zeta_2)g_j^T d_j}{L_0\|d_j\|^2} > 0, \forall j \geq 0$ is true, the lemma is immediately proved by combining (3.14). ◻

Remark 3 The inequality (3.13) means that

$$\frac{(g_j^T d_j)^2}{\|d_j\|^2} \to 0, \quad as \quad j \to \infty. \tag{3.16}$$

Next, we need to quote the following useful result (Lemma 3.2 in [22]).

Lemma 4 *Suppose Assumption 1 hold. If s_j and y_j follow the definitions in Algorithm 1 and B_j is updated by (2.12), for all $j \geq 0$, and there are positive constants $\mathscr{K}_1 \leq \mathscr{K}_2$ that*

$$\frac{y_j^T s_j}{\|s_j\|^2} \geq \mathscr{K}_1 \quad and \quad \frac{\|y_j\|^2}{y_j^T s_j} \leq \mathscr{K}_2, \quad \forall j \geq 0 \tag{3.17}$$

hold, then there exist constants $\mathscr{B}_1 > \mathscr{B}_2 > 0, \mathscr{B}_3 > 0$ such that, for any integer $t > 0$ and $j \in SD$,

$$\|B_j s_j\| \leq \mathscr{B}_1\|s_j\|, \tag{3.18}$$

and

$$\mathscr{B}_2\|s_j\|^2 \leq s_j^T B_j s_j \leq \mathscr{B}_3\|s_j\|^2 \tag{3.19}$$

hold for at least $[t/2]$ values of $j \in \{1, \ldots, t\}$; for any integer $t > 0$ and $j \notin SD$,

$$\|B_j s_j\| \leq \mathscr{B}_1\|s_j\|, \tag{3.20}$$

and

$$s_j^T B_j s_j \leq \mathscr{B}_3\|s_j\|^2 \tag{3.21}$$

hold for at least $[t/2]$ values of $j \in \{1, \ldots, t\}$.

Theorem 1 *Suppose Assumptions 1 and 2 hold and SD is an infinite set. Sequences $\{x_j\}$, $\{d_j\}$, $\{\alpha_j\}$, $\{g_j\}$ and $\{B_j\}$ are generated by Algorithm 1, then APT-BFGS converges globally, i.e. we have*

$$\liminf_{j\to+\infty} \|g_j\| = 0. \tag{3.22}$$

Proof Considering the proof by contradiction. Correspondingly, there exist a small constant $\epsilon_0 > 0$ such that

$$\|g_j\| \geq \epsilon_0, \forall j \tag{3.23}$$

hold. The proof will be completed in following two parts due to the positive and negative properties of the exponent of the power function $\|g(x)\|^\alpha$:

The first part ($\boldsymbol{\alpha \geq 0}$): The proof of this part can be discussed in two cases:
Case (a) $: j \in SD$, we have $\|y_j\| = \|g(x_{j+1}) - g(x_j)\| \leq L_0\|\alpha_j d_j\|$. By the Lemma 1, we have

$$\frac{s_j^T y_j}{\|s_j\|^2} = \frac{s_j^T y_j}{\|\alpha_j d_j\|^2} \geq \eta_1 \|g_j\|^\alpha \geq \eta_1 \epsilon_0^\alpha, \tag{3.24}$$

and

$$\frac{\|y_j\|^2}{s_j^T y_j} \leq \frac{L_0^2\|\alpha_j d_j\|^2}{s_j^T y_j} \leq \frac{L_0^2}{\eta_1 \epsilon_0^\alpha}. \tag{3.25}$$

Case (b) $: j \notin SD$, similar to (3.25), by the Lemma 2, we obtain

$$\frac{\|y_j\|^2}{s_j^T y_j} \leq \frac{L_0^2\|\alpha_j d_j\|^2}{s_j^T y_j} \leq \frac{L_0^2}{\eta_2 \|g_j\|^\alpha} \leq \frac{L_0^2}{\eta_2 \epsilon_0^\alpha}. \tag{3.26}$$

By (3.12) and the case of $j \notin SD$, we get

$$0 < -g_j^T d_j < \rho\alpha_j \|d_j\|^2 \, \|g_j\|^\alpha. \tag{3.27}$$

Through using the definition of P_j, we obtain the upper bound of the Euclidean norm of P_j

$$\begin{aligned}
0 < P_j &= g(\mathscr{V}_j)^T(\mathscr{V}_j - x_j) + \mu\|\mathscr{V}_j - x_j\|^2\|g(x_j)\|^\alpha \\
&= \alpha_j[g(x_j + \alpha_j d_j) - g(x_j)]^T d_j + \alpha_j g(x_j)^T d_j + \mu\|\alpha_j d_j\|^2\|g_j\|^\alpha \\
&\leq \alpha_j L_0\|\alpha_j d_j\|\|d_j\| + \alpha_j|g_j^T d_j| + \mu\|\alpha_j d_j\|^2\|g_j\|^\alpha \\
&< \|\alpha_j d_j\|^2(L_0 + (\rho+\mu)\|g_j\|^\alpha),
\end{aligned}$$

where the first inequality is caused by Cauchy inequality and (3.27). Furthermore, we have the upper bound of the Euclidean norm of s_j

$$\begin{aligned}
\|s_j\| \quad &= \quad \left\|\frac{P_j}{\|g(\mathscr{V}_j) - g(x_j)\|^2}[g(\mathscr{V}_j) - g(x_j)]\right\| \\
&= \quad \left\|\frac{P_j}{\|g(\mathscr{V}_j) - g(x_j)\|}\right\| \\
\& < \left\|\frac{\|\alpha_j d_j\|^2(L_0+(\rho+\mu)\|g_j\|^\alpha)}{\|g(\mathscr{V}_j)-g(x_j)\|}\right\| \\
&\leq \quad \left\|\frac{\|\alpha_j d_j\|^2(L_0 + (\rho+\mu)\|g_j\|^\alpha)}{L_1\|\alpha_j d_j\|}\right\| \\
&\leq \quad \|\alpha_j d_j\|\frac{L_0 + (\rho+\mu)G_b^\alpha}{L_1},
\end{aligned}$$

where the second inequality follows Assumption 1 and constant $L_1 > 0$. Therefore, by lemma 2, we get

$$\frac{s_j^T y_j}{\|s_j\|^2} \geq \frac{s_j^T y_j}{\|\alpha_j d_j\|^2}(\frac{L_1}{L_0 + (\rho + \mu)G_b^\alpha})^2 \geq \frac{L_1^2 \eta_2 \epsilon_0^\alpha}{(L_0 + (\rho + \mu)G_b^\alpha)^2}. \tag{3.28}$$

Over all, we can always have

$$\frac{s_j^T y_j}{\|s_j\|^2} \geq \lambda_1 \quad \text{and} \quad \frac{\|y_j\|^2}{s_j^T y_j} \leq \lambda_2, \tag{3.29}$$

where $\lambda_1 = \min\{\frac{L_1^2 \eta_2 \epsilon_0^\alpha}{(L_0+(\rho+\mu)G_b^\alpha)^2}, \eta_1 \epsilon_0^\alpha\}$ and $\lambda_2 = \max\{\frac{L_0^2}{\eta_1 \epsilon_0^\alpha}, \frac{L_0^2}{\eta_2 \epsilon_0^\alpha}\}$. For simplicity, we set $\mathscr{K}_1 = \lambda_1$ and $\mathscr{K}_2 = \lambda_2$. Using the lemma 4, it is obvious that there are positive constants $\mathscr{B}_1 > \mathscr{B}_2 > 0$, $\mathscr{B}_3 > 0$, such that (3.18), (3.19) holds. By combining (2.11) and (3.16), we obtain

$$\frac{(d_j^T B_j d_j)^2}{\|d_j\|^2} \to 0, \quad \text{as} \quad j \to \infty. \tag{3.30}$$

Since SD is infinite, there must be at least one subset $SD_k := \{j_k | d_{j_k}^T g_{j_k} \leq -\rho\alpha_{j_k}\|d_{j_k}\|^2\|g_{j_k}\|^\alpha, j_k \geq 0\} \subset SD$ such that

$$\frac{(d_{j_k}^T B_{j_k} d_{j_k})^2}{\|d_{j_k}\|^2} \to 0, \quad \text{as} \quad k \to \infty. \tag{3.31}$$

Then, by (3.19), we get

$$0 \leq \mathscr{B}_2^2 \|d_{j_k}\|^2 \leq \frac{(d_{j_k}^T B_{j_k} d_{j_k})^2}{\|d_{j_k}\|^2} \to 0, \quad \text{as} \quad k \to \infty, \tag{3.32}$$

which imlies that $\|d_{j_k}\| \to 0$ holds for $k \to \infty$, due to the constant $\mathscr{B}_2 > 0$. According to (2.11) and (3.18), we get

$$\|g_{j_k}\| \leq \mathscr{B}_1 \|d_{j_k}\|. \tag{3.33}$$

Similarly, by (2.11) and (3.19), we can easily get

$$\mathscr{B}_2 \|d_{j_k}\| \leq \|g_{j_k}\|, \tag{3.34}$$

where $\mathscr{B}_1 > \mathscr{B}_2 > 0$. Therefore, (3.22) holds together with (3.33) and (3.34), as $k \to \infty$, which contradicts (3.23). The proof of this part is complete.

The second part ($\boldsymbol{\alpha < 0}$): The constant C:=max$\{G_b^\alpha, \epsilon_0^\alpha\}$ is introduced, then we have $C \geq \|g_j\|^\alpha \geq G_b^\alpha > 0$. The proof of the remaining part can also be discussed in two cases:

Case (**a**) :$j \in SD$, similar to (3.24) and (3.25), we have

$$\frac{s_j^T y_j}{\|s_j\|^2} \geq \eta_1 G_b^\alpha \quad \text{and} \quad \frac{\|y_j\|^2}{s_j^T y_j} \leq \frac{L_0^2}{\eta_1 G_b^\alpha}. \tag{3.35}$$

Case (**b**) :$j \notin SD$, similar to (3.26), by Lemma 2, we have

$$\frac{\|y_j\|^2}{s_j^T y_j} \leq \frac{L_0^2}{\eta_2 G_b^\alpha}. \tag{3.36}$$

According to the definition of s_j, we can easily get

$$\begin{aligned}
\|s_j\| &= \|\frac{P_j}{\|g(\mathscr{V}_j) - g(x_j)\|^2}[g(\mathscr{V}_j) - g(x_j)]\| \\
&< \|\frac{\|\alpha_j d_j\|^2 (L_0 + (\rho + \mu)\|g_j\|^\alpha)}{\|g(\mathscr{V}_j) - g(x_j)\|}\| \\
&\leq \|\alpha_j d_j\| \frac{L_0 + (\rho + \mu)C}{L_1}.
\end{aligned}$$

Thereby, by lemma 2, we can write

$$\frac{s_j^T y_j}{\|s_j\|^2} \geq \frac{s_j^T y_j}{\|\alpha_j d_j\|^2} (\frac{\eta_1 G_b^\alpha}{L_0 + (\rho + \mu)C})^2 \geq \frac{\eta_1^2 \eta_2 G_b^{4\alpha}}{(L_0 + (\rho + \mu)C)^2}. \tag{3.37}$$

Overall, there are obviously positive constants λ_3 and λ_4 which satisfy the following inequalities:

$$\frac{s_j^T y_j}{\|s_j\|^2} \geq \lambda_3 \quad \text{and} \quad \frac{\|y_j\|^2}{s_j^T y_j} \leq \lambda_4. \tag{3.38}$$

The rest of the proof is the same as the first part. So far, the whole proof is completed.

In the projection BFGS method [39], it is difficult to discuss the convergence rate, and the following results are a supplement in this respect.

Assumption 3 1. $\varphi(x)$ is twice continuously differentiable.

2. $\{x_j\}$ converges to an accumulation point x^* at which $g(x^*) = 0$.

3. The Hessian matrix $G(x)$ of $\varphi(x)$ is Hölder continuous and is positive definite at x^*.

Theorem 2 *Suppose Assumptions 1–3 hold and (1.5) satisfies the strong Wolfe condition. When j is large enough and sequences $\{x_j\}$, $\{d_j\}$, $\{\alpha_j\}$, $\{g_j\}$, and $\{B_j\}$ are generated by Algorithm 1, then the convergence rate of Algorithm 1 is superlinear.*

Proof According to Assumption 3 and the mean value inequality, we can easily get $y_j = [\int_0^1 G(x_j + \tau\alpha_j d_j) d\tau](\alpha_j d_j)$. Moreover, there is a constant m such that

$y_j^T(\alpha_j d_j) \geq m\|\alpha_j d_j\|^2$ i.e. $g(x_j + \alpha_j d_j)^T d_j - g(x_j)^T d_j \geq m\alpha_j\|d_j\|^2$. Due to the strong Wolfe conditions, when j is large enough, the inequalities

$$-(1+\zeta_2)g_j^T d_j \geq g(x_j+\alpha_j d_j)^T d_j - g(x_j)^T d_j \geq m\alpha_j\|d_j\|^2 \geq (1+\zeta_2)\rho\alpha_j\|d_j\|^2\|g_j\|^\alpha \tag{3.39}$$

is always satisfied. Then, we easily deduce that sufficient descent condition can always be satisfied when j is large enough, which means that the case $j \in SD$ (2.7) is always satisfied, i.e., APT-BFGS method reduces to the ordinary BFGS method. When the ordinary BFGS method is considered, its superlinear convergence has been proven (see [5, 21, 22, 44]). Therefore, the superlinear convergence of Algorithm 1 is established.□

4 Numerical experiments

In the section, we show the numerically experimental test results for seventy-four problems (the specific problems are shown in Table 1 in [38]) from an unconstrained optimization test functions collection which presented by Andrei [1]. To test and compare the numerical performance of Algorithm 1, we show some current algorithms [22, 39, 40] that perform well and are somewhat related to technique proposed by us. Algorithms that do not use projection technique are ordinary BFGS method and a modified BFGS method of Li and Fukushima [22], which are both under the WWP line search condition, called WWP-BFGS and WWP-LFBFGS, respectively. To show the effect of Algorithm 1 with adaptive projection technique, the projection BFGS algorithm with modified WWP line technique [39] and ordinary WWP line technique [40] are also applied, which we called MWWP-PT-BFGS and WWP-PT-BFGS, respectively. For Algorithm 1, corresponding to the two parts of Algorithm 1, α is chosen as 0.1 or -0.1, representing WWP-APT-BFGS-1 and WWP-APT-BFGS-2 respectively. In fact, for different problems, there must be some optimal α in Algorithm 1 which will achieve the best experimental results. For convenience, we choose a fixed value for all experiments. Nevertheless, we can still get superior numerical results. The $Himmeblau$ stop rule is used: if $|\varphi(x_j)| > \mathscr{E}_1$ holds, let $Ter_1 = \frac{|\varphi(x_j)-\varphi(x_{j+1})|}{|\varphi(x_j)|}$; otherwise, set $Ter_1 = |\varphi(x_j) - \varphi(x_{j+1})|$. If the condition $\|g(x)\| < \mathscr{E}$ (or $Ter_1 < \mathscr{E}_2$) is true or the total number of iteration loops exceeds one thousand, this program terminates, where $\mathscr{E}_1 = \mathscr{E}_2 = 10^{-5}$ and $\mathscr{E} = 10^{-6}$. We give the parameter and initialization settings in Algorithm 1: $\zeta_1 = 0.2$, $\zeta_2 = 0.8$, $\rho = 0.7$, $\mu = 4\zeta_2\rho$, $B_0 = I$; the initialization settings for x_0 are provided by Andrei [1], and each problem has different initialization settings. Only when the numbers of searches is guaranteed to be greater than ten, the step size α_j obtained through the WWP search technique will be selected. This algorithm code is written on MATLAB R2019a software; the program runs on a PC with an Intel (R) Core (TM) i7-6700HQ CPU at 2.60GHz (8 CPUs), 2.6GHz, and 8066MB of RAM; and the operating system is Windows 10.

The overall experimental results are shown in Table 1. Some notations need to be clarified before analysis: dim, ni, nfg, and time denote the dimension of the problems being tested, the total number of algorithm iterations, the total number of functions

Table 1 The total numerical results

dim	ni	nfg	Time
WWP-APT-BFGS-1			
300	2758	10,681	85.21875
900	2864	7816	885.890625
1200	1628	8138	948.65625
2100	1344	2597	7010.34375
3000	1129	3287	10,463.26563
WWP-APT-BFGS-2			
300	2648	8636	74.515625
900	2709	7499	865.046875
1200	1631	6117	943.390625
2100	1360	2662	7984.5625
3000	1144	3256	9955.25
MWWP-PT-BFGS			
300	2537	11,121	87.59375
900	2989	8004	893.09375
1200	1634	6668	971.25
2100	1548	3030	7968.09375
3000	1203	3608	10,851.53125
WWP-PT-BFGS			
300	2302	8163	70.953125
900	3030	7959	970.484375
1200	1628	8591	933.984375
2100	1541	2936	8987.421875
3000	1167	3708	11,573.5
WWP-BFGS			
300	3523	11,784	101.9375
900	4224	10423	1339.625
1200	2696	10849	1735.484375
2100	4384	5620	21,457.42188
3000	2236	6363	19,922.40625
WWP-LFBFGS			
300	3523	11,784	97.09375
900	4224	10,423	1184.015625
1200	2696	10,849	1524.796875
2100	4384	5620	20,132.60938
3000	2236	6363	16,953.67188

and gradient computations during the iteration, and the CPU time (in seconds) of the entire iterative process respectively.

As can be seen from Table 1, the algorithm presented in the paper is clearly effective for solving all tested problems, and it is competitive with the other four algorithms in ni, nfg, and time on the tested problems. The profiles of these algorithms are compared and analyzed by the tool of Dolan and Morè [10]. The performances of algorithms WWP-APT-BFGS-1, WWP-APT-BFGS-2, MWWP-PT-BFGS, WWP-PT-BFGS, WWP-BFGS, and WWP-LFBFGS in terms of ni, nfg, and time are shown in Figs. 1, 2, and 3 respectively, where the y-axis represents the probability that the solver can achieve the best possible performance ratio, and the x-axis represents the factor (see [10] for more details). Obviously, Figs. 1 and 2 illustrate that the performance of the algorithms which use the projection technique (including adaptive projection technique) are better than those without using the projection technique, suggesting that projection techniques can effectively adapt to some bad situations to achieve our expected results. Figures 1 and 2 also indicate that WWP-APT-BFGS-1 and WWP-APT-BFGS-2 outperform other algorithms; additionally, they have more robustness than that of others. It is not difficult to see from Fig. 3 that WWP-APT-BFGS-1 and MWWP-PT-BFGS exhibit the best performance among the six algorithms, while WWP-APT-BFGS-1 and WWP-APT-BFGS-2 achieve slightly better robustness than MWWP-PT-BFGS, and all these three algorithms are far superior to the others in terms of robustness performance. Because the quadratic term in the projection formula is multiplied by an adaptive part $\|g_j\|^\alpha$, the adaptive projection algorithm should take longer CPU computation time than other projection algorithm in theory. However, as

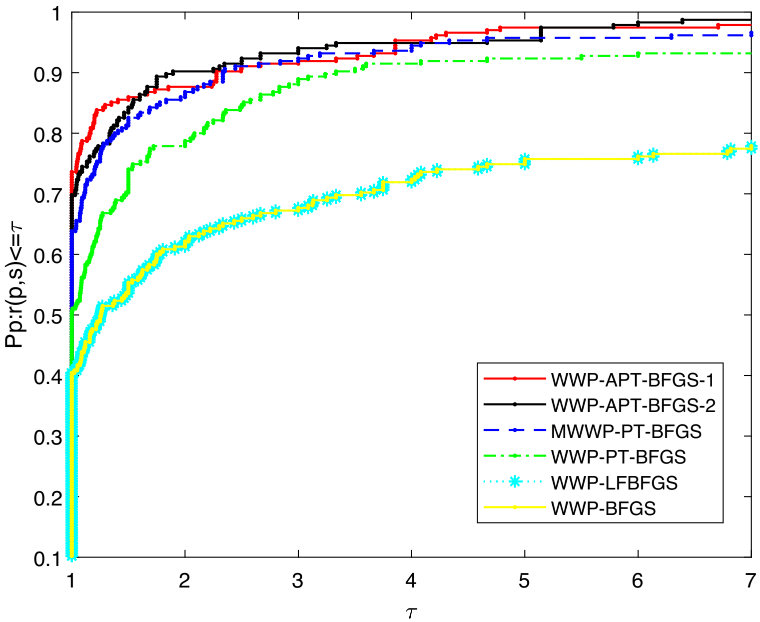

Fig. 1 Performance profiles of the WWP-APT-BFGS and four other algorithms in terms of ni

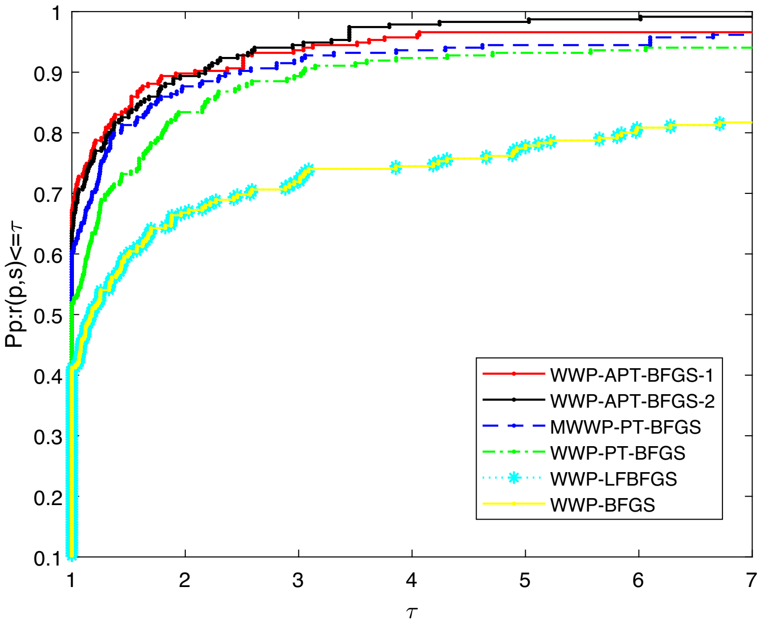

Fig. 2 Performance profiles of the WWP-APT-BFGS and four other algorithms in terms of nfg

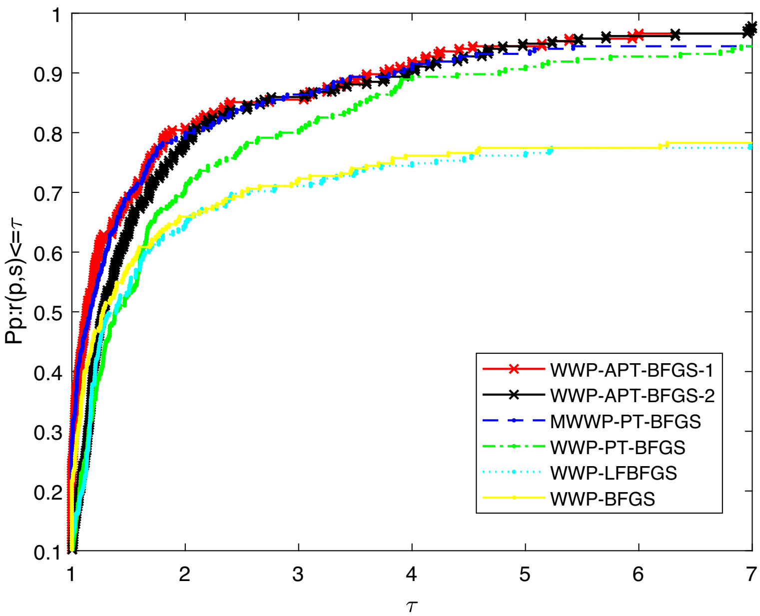

Fig. 3 Performance profiles of the WWP-APT-BFGS and four other algorithms in terms of time

long as we choose the appropriate tuning parameters α, the adaptive projection algorithm can be comparable to other projection algorithms in CPU computation time, such as WWP-APT-BFGS-1. Therefore, the selection of α is crucial for different problems.

Besides, the percentage of projection iterations in the total number of iterations will also affect the performance of the presented algorithm. For example, considering WWP-APT-BFGS-1 for all the tested problems, the average projection iteration percentage are 18%, 7%, 9%, 12%, and 4%, and the corresponding dimensions are 300, 900, 1200, 2100, and 3000 which means that the sufficient descent condition (2.7) can be met by most iterations, and a few iterations are generated by projection iteration (2.9).

5 Applications for Muskingum model

The section intends to explore the application of Algorithm 1 to the Muskingum model which is a famous hydrologic engineering application problem. Flood routing is a fundamental issue when calculating flood waveforms along open channels. As a hydrological routing method, the water storage capacity of Muskingum model is determined by the outflow and inflow of water, which is a common flood routing model, and it is defined as follows:

$$\begin{aligned}\min f\left(x_1, x_2, x_3\right) = \sum_{j=1}^{n-1} \Bigg(&\left(1-\frac{\Delta t}{6}\right) x_1 \left(x_2 I_{j+1} + \left(1-x_2\right) Q_{j+1}\right)^{x_3} \\ &- \left(1-\frac{\Delta t}{6}\right) x_1 \left(x_2 I_j + \left(1-x_2\right) Q_j\right)^{x_3} - \frac{\Delta t}{2}\left(I_j - Q_j\right) \\ &+ \frac{\Delta t}{2}\left(1-\frac{\Delta t}{3}\right)\left(I_{j+1} - Q_{j+1}\right)\Bigg)^2,\end{aligned} \tag{5.1}$$

where $\{x_i\}$ is the parameter sequence, when $i = 1, 2, 3$, x_i will represent the constant of storage time, the weight factor, and another necessary parameter when estimating respectively, $\{Q_j | j = 1, 2, \cdots, n\}$ and $\{I_j | j = 1, 2, \cdots, n\}$ are the observed sequences, representing outflow discharges and inflow discharges (at time t_j), and Δt represents the time interval. In numerical experiments, the initial selections of parameters are $\Delta t = 12$ (h) and $x = (0, 1, 1)^{\mathrm{T}}$. Moreover, the specific informations of parameters Q_j and I_j in the year 1960,1961, and 1964 respectively are presented in [26].

The actual observed data and the data calculated by Algorithm 1 are presented in Figs. 4, 5, and 6, corresponding to three different years respectively, and the final parameter estimation results are listed in Table 2, including the estimated results of the other two algorithms for reference. The following conclusions can be drawn by these numerical results: (i) All three figures illustrate the excellent performance of Algorithm 1; (ii) it is clear that the proposed adaptive projection BFGS method is very efficient for estimating the parameters in the Muskingum model.

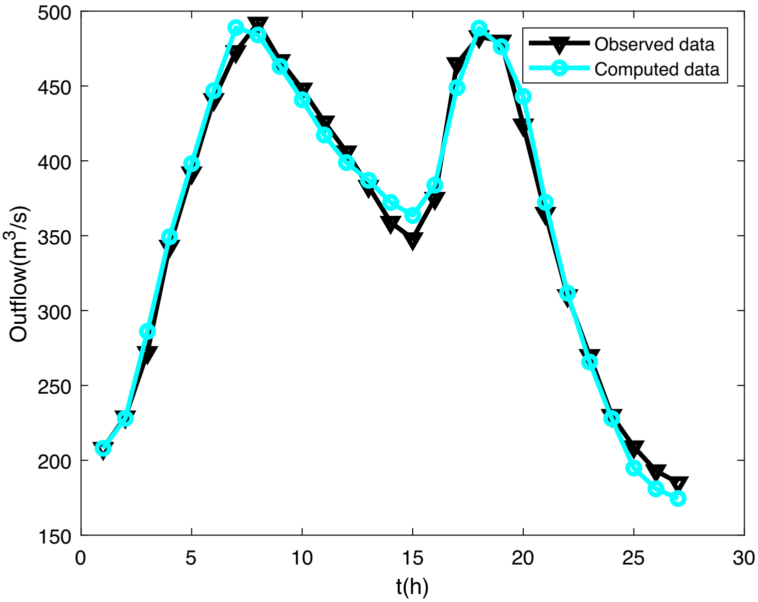

Fig. 4 Performance of APT-BFGS in 1960

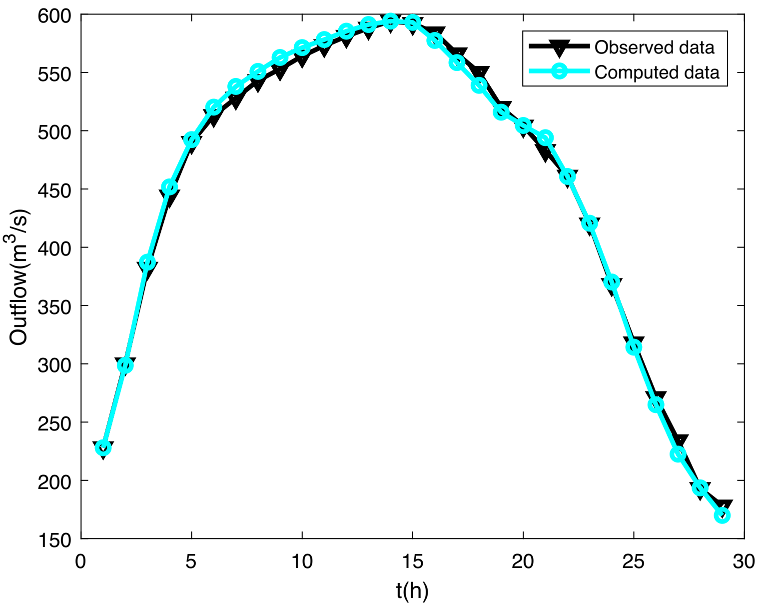

Fig. 5 Performance of APT-BFGS in 1961

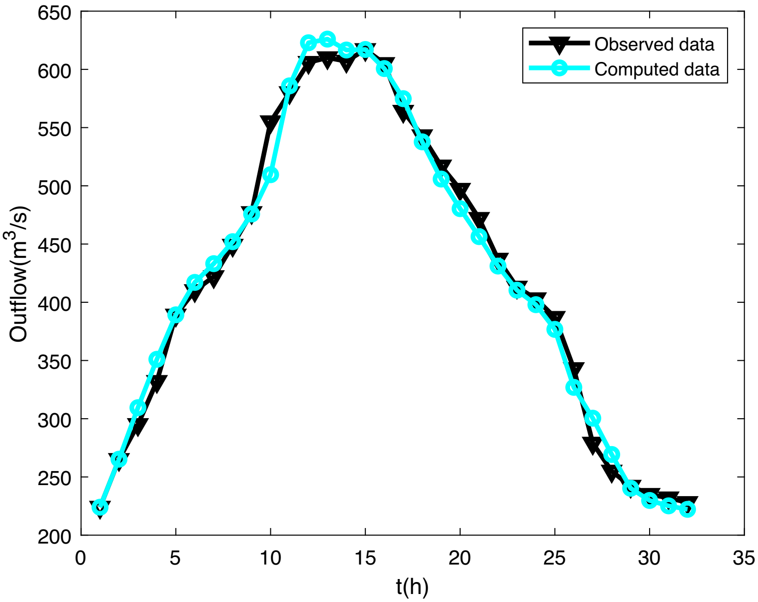

Fig. 6 Performance of APT-BFGS in 1964

6 Conclusions and future works

The purpose of the paper is to discuss whether the BFGS method is globally convergent for non-convex problems under Wolfe search conditions, and an adaptive projection BFGS method is proposed for solving it, which possesses the following properties: (i) When a special sufficient descent condition is satisfied, the ordinary BFGS iteration formula is updated as usual; otherwise, an adaptive projection technique will be used to generate the next point; (ii) APT-BFGS method converges globally under the WWP line search condition for general functions and the convergence rate is superlinear; (iii) this algorithm is so universal that the adaptive projection formulas includes several kinds of projection formulas mentioned in this paper, and at the same time, it has the flexibility to solve different problems to show the corresponding better efficiency. At last, encouraging experimental results of the proposed algorithm are reported when compared with other four algorithms.

As a future work, the following points are considered: (i) It should be mentioned that after defining a suitable surface, the convergence of the conjugate gradient method

Table 2 Parameter estimation results of the three algorithms

Algorithm	x_1	x_2	x_3
BFGS [15]	10.8156	0.9826	1.0219
HIWO [25]	13.2813	0.8001	0.9933
APT-BFGS	11.1850	1.0038	0.9994

(Polak-Ribière-Polyak) under the Wolfe search conditions can also be proved by the same steps described in [43]; (ii) whether this projection formula framework can be applied to other areas such as stochastic optimization; (iii) it is possible to come up with a new projection formulation to weaken some assumptions, such as the Lipschitz-smooth condition, which is also what we are working on; (iv) we want to prove that we can find at least one step-size α_j that satisfies (3.4).

Acknowledgements The authors are grateful to the editor and anonymous reviewers for their suggestions, which improve this paper greatly.

Author Contributions The contributions can be divided into the following parts: Gonglin Yuan: conceptualization, methodology, software, supervision. Xiong Zhao: data curation, writing—original draft preparation, writing—reviewing and editing, software, validation, visualization, investigation, formal analysis. Kejun Liu: writing—original draft preparation, data curation. Xiaoxuan Chen: writing—review and editing.

Funding This work is supported by Guangxi Science and Technology base and Talent Project (Grant No. AD22080047), the Special Funds for Local Science and Technology Development Guided by the Central Government (No. ZY20198003), the Innovation Funds of Chinese University (Grant No. 2021BCF03001), and the special foundation for Guangxi Ba Gui Scholars.

Data availability The data that support the findings of this study are available on request from the corresponding author, upon reasonable request.

Declarations

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Conflict of interest The authors declare no competing interests.

References

1. Andrei, N.: An unconstrained optimization test functions collection. Adv. Model. Optim. **10**(1), 147–161 (2008)
2. Andrei, N.: An adaptive scaled BFGS method for unconstrained optimization. Num. Algo. **77**(2), 413–432 (2018)
3. Broyden, C.G.: The convergence of a class of double-rank minimization algorithms 1.general considerations. IMA J Appl. Math. **6**(1), 76–90 (1970)
4. Broyden, C.G., Dennis, J.E., Jr., Moré, J.J.: On the local and superlinear convergence of quasi-Newton methods. IMA J Appl. Math. **12**(3), 223–245 (1973)
5. Byrd, R.H., Nocedal, J.: A tool for the analysis of quasi-Newton methods with application to unconstrained minimization. SIAM J Numer. Anal. **26**(3), 727–739 (1989)
6. Byrd, R.H., Nocedal, J., Yuan, Y.-X.: Global convergence of a class of quasi-Newton methods on convex problems. SIAM J Numer. Anal. **24**(5), 1171–1190 (1987)
7. Dai, Y.: A perfect example for the BFGS method. Math. Prog. **138**(1), 501–530 (2013)
8. Dennis, J.E., Moré, J.J.: A characterization of superlinear convergence and its application to quasi-Newton methods. Math. Comput. **28**(126), 549–560 (1974)
9. Dennis, J.E., Jr., Moré, J.J.: Quasi-Newton methods, motivation and theory. SIAM Rev. **19**(1), 46–89 (1977)
10. Dolan, E.D., Moré, J.J.: Benchmarking optimization software with performance profiles. Math. Prog. **91**(2), 201–213 (2002)
11. Fletcher, R.: A new approach to variable metric algorithms. Comp. J **13**(3), 317–322 (1970)

12. Fletcher, R. An overview of unconstrained optimization. Algorithms for Continuous Optimization, pp. 109–143 (1994)
13. Fu, Z., Ren, K., Shu, J., Sun, X., Huang, F.: Enabling personalized search over encrypted outsourced data with efficiency improvement. IEEE Trans. Parallel Distrib. Syst. **27**(9), 2546–2559 (2015)
14. Fu, Z., Wu, X., Guan, C., Sun, X., Ren, K.: Toward efficient multi-keyword fuzzy search over encrypted outsourced data with accuracy improvement. IEEE Trans. Inform. Forensics Secur. **11**(12), 2706–2716 (2016)
15. Geem, Z.W.: Parameter estimation for the nonlinear Muskingum model using the BFGS technique. J Irrigation Drainage Eng. **132**(5), 474–478 (2006)
16. Goldfarb, D.: A family of variable-metric methods derived by variational means. Math. Comput. **24**(109), 23–26 (1970)
17. Griewank, A.: The global convergence of partitioned BFGS on problems with convex decompositions and Lipschitzian gradients. Math. Prog. **50**(1), 141–175 (1991)
18. Griewank, A., Toint, P.L.: Local convergence analysis for partitioned quasi-Newton updates. Numerische Mathematik **39**(3), 429–448 (1982)
19. Grippo, L., Lucidi, S.: A globally convergent version of the Polak-Ribiére conjugate gradient method. Math. Prog. **78**(3), 375–391 (1997)
20. Gu, B., Sun, X., Sheng, V.S.: Structural minimax probability machine. IEEE Trans. Neural Netw. Learn. Syst. **28**(7), 1646–1656 (2016)
21. Li, D., Fukushima, M.: A modified BFGS method and its global convergence in nonconvex minimization. J Comput. Appl. Math. **129**(1–2), 15–35 (2001)
22. Li, D., Fukushima, M.: On the global convergence of the BFGS method for nonconvex unconstrained optimization problems. SIAM J Optim. **11**(4), 1054–1064 (2001)
23. Liu, J., Li, S.: A projection method for convex constrained monotone nonlinear equations with applications. Comput. Math. Appl. **70**(10), 2442–2453 (2015)
24. Moscariello, A., Takx, R.A., Schoepf, U.J., Renker, M., Zwerner, P.L., O'Brien, T.X., Allmendinger, T., Vogt, S., Schmidt, B., Savino, G., et al.: Coronary CT angiography: image quality, diagnostic accuracy, and potential for radiation dose reduction using a novel iterative image reconstruction technique–comparison with traditional filtered back projection. European Radiology **21**(10), 2130–2138 (2011)
25. Ouyang, A., Liu, L.-B., Sheng, Z., Wu, F. A class of parameter estimation methods for nonlinear Muskingum model using hybrid invasive weed optimization algorithm. Mathematical Problems in Engineering, 2015, (2015)
26. Ouyang, A., Tang, Z., Li, K., Sallam, A., Sha, E.: Estimating parameters of Muskingum model using an adaptive hybrid PSO algorithm. Int. J Pattern Recogn. Artif. Intell. **28**(01), 1459003 (2014)
27. Pan, Z., Lei, J., Zhang, Y., Sun, X., Kwong, S.: Fast motion estimation based on content property for low-complexity H. 265/HEVC encoder. IEEE Trans. Broadcasting **62**(3), 675–684 (2016)
28. Pan, Z., Zhang, Y., Kwong, S.: Efficient motion and disparity estimation optimization for low complexity multiview video coding. IEEE Trans. Broadcasting **61**(2), 166–176 (2015)
29. Powell, M. J. A new algorithm for unconstrained optimization. In Nonlinear Programming, pp. 31–65. Elsevier (1970)
30. Powell, M. J. Some global convergence properties of a variable metric algorithm for minimization without exact line searches. In Nonlinear Programming, SIAM-AMS Proceedings, vol. 9 (1976)
31. Shanno, D.F.: Conditioning of quasi-Newton methods for function minimization. Math. Comput. **24**(111), 647–656 (1970)
32. Toint, P.L.: Global convergence of the partitioned BFGS algorithm for convex partially separable optimization. Math. Prog. **36**(3), 290–306 (1986)
33. Wang, C., Wang, Y., Xu, C.: A projection method for a system of nonlinear monotone equations with convex constraints. Math. Method Oper. Res. **66**(1), 33–46 (2007)
34. Wei, Z., Li, G., Qi, L.: New nonlinear conjugate gradient formulas for large-scale unconstrained optimization problems. Appl. Math. Comput. **179**(2), 407–430 (2006)
35. Wei, Z., Yu, G., Yuan, G., Lian, Z.: The superlinear convergence of a modified BFGS-type method for unconstrained optimization. Comput. Optim. Appl. **29**(3), 315–332 (2004)
36. Xia, Z., Wang, X., Sun, X., Liu, Q., Xiong, N.: Steganalysis of LSB matching using differences between nonadjacent pixels. Multimed. Tool Appl. **75**(4), 1947–1962 (2016)
37. Xia, Z., Wang, X., Zhang, L., Qin, Z., Sun, X., Ren, K.: A privacy-preserving and copy-deterrence content-based image retrieval scheme in cloud computing. IEEE Trans. Inform. Forensics Secur. **11**(11), 2594–2608 (2016)

38. Yuan, G., Li, P., Lu, J. The global convergence of the BFGS method with a modified WWP line search for nonconvex functions. Numerical Algorithms, pp. 1–13 (2022)
39. Yuan, G., Sheng, Z., Wang, B., Hu, W., Li, C.: The global convergence of a modified BFGS method for nonconvex functions. J Comput. Appl. Math. **327**, 274–294 (2018)
40. Yuan, G., Wang, X., Sheng, Z.: The projection technique for two open problems of unconstrained optimization problems. J Optim. Theory Appl. **186**(2), 590–619 (2020)
41. Yuan, G., Wei, Z.: Convergence analysis of a modified BFGS method on convex minimizations. Comput. Optim. Appl. **47**(2), 237–255 (2010)
42. Yuan, G., Wei, Z., Lu, X.: Global convergence of BFGS and PRP methods under a modified weak Wolfe-Powell line search. Appl. Math. Model. **47**, 811–825 (2017)
43. Yuan, G., Wei, Z., Yang, Y.: The global convergence of the Polak-Ribiére-Polyak conjugate gradient algorithm under inexact line search for nonconvex functions. J Comput. Appl. Math. **362**, 262–275 (2019)
44. Yuan, G., Zhang, M., Zhou, Y.: Adaptive scaling damped BFGS method without gradient Lipschitz continuity. Appl. Math. Lett. **124**, 107634 (2022)
45. Zhang, L., Zhou, W.: Spectral gradient projection method for solving nonlinear monotone equations. J Comput. Appl. Math. **196**(2), 478–484 (2006)
46. Zhang, S.: Recent progresses on real-time 3D shape measurement using digital fringe projection techniques. Opt. Lasers Eng. **48**(2), 149–158 (2010)