

PRP-Type Direct Search Methods for Unconstrained Optimization*

Qunfeng Liu, Wanyou Cheng

College of Computer, Dongguan University of Technology, Dongguan, China

E-mail: liuqf@dgut.edu.cn

Received March 26, 2011; revised April 14, 2011; accepted April 17, 2011

Abstract

Three PRP-type direct search methods for unconstrained optimization are presented. The methods adopt three kinds of recently developed descent conjugate gradient methods and the idea of frame-based direct search method. Global convergence is shown for continuously differentiable functions. Data profile and performance profile are adopted to analyze the numerical experiments and the results show that the proposed methods are effective.

Keywords: Direct Search Methods, Descent Conjugate Gradient Methods, Frame-Based Methods, Global Convergence, Data Profile, Performance Profile

1. Introduction

Direct search methods form an important class of numerical methods for solving optimization problems. They are particularly useful in the solution of the problems where the derivatives are not available, or difficult to compute. Early direct search methods can trace back to the compass search method [1] and the pattern search method [2]. In the 1960s, the direct search methods were widely applied. Due to the lack of convergence theory, direct search methods fell out of favor with the mathematical optimization community by the early of 1970s.

In the past twenty years, direct search methods have seen a revival. Significant interests have been provoked by new convergence results. For a direct search method to achieve convergence, it often needs to employ one of the following three techniques, namely line searches, trust regions and discrete grids [3]. In this paper we will adopt the discrete grid strategy to develop some globally convergent direct search methods. Alternative approaches based on line searches and trust regions can be found in [3-6] and the references therein.

Early convergence theory based on discrete grids was established by Torczon *et al.* in [7-9]. They developed a framework for the GPS (generalized pattern search) method which contains many earlier algorithms, such as the compass search, Hooke and Jeeves' pattern search, as

special cases. The GPS methods have been extended to bound and linearly constrained problems [10,11], non-smooth problems [12] and general constrained problems [13].

The grids play an important role in the GPS methods and their extensions. In the GPS methods, each grid is often a subset of some member of a sequence of nested grids.

Coope and Price [14] has shown that the grids can be chosen more freely. The grid-based methods proposed in [14] incorporated two arbitrary finite processes which may reoriented and reshaped the grids. Partly provoked by Coope and Price's ideas, the review paper [15] presented a new unified presentation which was called GSS (generating set search) method for a large number of direct search methods including the GPS methods. The GSS methods have been also extended to linear constrained optimization [16,17].

The great freedom in the choice of grids permits to choose grids to reflect the information gathered during previous iterations, especially the possible gradient information. In [18], a concept called frame was proposed to gather the gradient information. Here a frame (see Definition 2.2) is a fragment of a grid. Loosely speaking, a frame is a set of points which surround a central point called frame center. Suppose that we already have a frame around the iterate x_k , then the set of search directions must contain all the frame points and may contain an arbitrary infinite process. If there is no search direc-

*This work was supported by the NNSF of China Grant 11071087 and 10971058.

tion such that the objective function value is sufficiently less than $f(x_k)$, then x_k is called a quasi-minimal point (see Definition 2.3). After a quasi-minimal point is founded, the frame size must be reduced for convergence.

A direct search method which confirms to the frame-based template in [18] was proposed in [19], where the PRP+ method was used to make use of the gradient information gathered during previous iterations. The presented numerical experiments in [19] show that this frame-based conjugate gradients method is effective.

In this paper, we consider the following unconstrained optimization

$$\min f(x), x \in R^n, \tag{1}$$

where the objective function $f(x)$ is continuously differential but the gradient of f is not available or computationally expensive.

Our main purpose is to construct efficient direct search methods for solving (1). We also use the frame-based template in [18], but substitute PRP+ method with some descent conjugate gradient methods to make use of the gradient information. These descent conjugate gradient methods for solving unconstrained optimization problems enjoy some nice properties and good numerical behavior. A common property of these methods is that they can generate sufficient descent directions for the objective function. In the case when exact gradients are available, descent conjugate gradient methods are computational more effective than PRP+ method [20-23]. We are going to include the descent conjugate gradient methods in the frame-based direct search framework to develop more efficient direct search methods for solving (1).

We will consider descent conjugate gradient methods which we call the TTPRP (three-term PRP) method proposed in [23], the TMPRP (two-term modified PRP) method proposed in [4] and the PRP-DC proposed in [24].

Let us have a simple review to these descent conjugate gradient methods. Each method generate a sequences of iterates $\{x_k\}$ by

$$x_{k+1} = x_k + \alpha_k d_k, k = 0, 1, \dots$$

where the step-length α_k is determined by some line search, and the initial direction is set to $d_0 = -\nabla f(x_0)$. In the TTPRP method, the direction $d_k (k \geq 1)$ is determined by

$$d_k = -\nabla f(x_k) + \frac{\nabla f(x_k)^T y_{k-1}}{\|\nabla f(x_{k-1})\|^2} d_{k-1} - \frac{\nabla f(x_k)^T d_{k-1}}{\|\nabla f(x_{k-1})\|^2} y_{k-1}, \tag{2}$$

where $y_{k-1} = \nabla f(x_k) - \nabla f(x_{k-1})$. In the TMPRP method,

the direction $d_k (k \geq 1)$ is determined by

$$d_k = -\nabla f(x_k) + \frac{\nabla f(x_k)^T y_{k-1}}{\|\nabla f(x_{k-1})\|^2} \left(I - \frac{\nabla f(x_k) \nabla f(x_k)^T}{\|\nabla f(x_k)\|^2} \right) d_{k-1}, \tag{3}$$

where I denotes the identity matrix. In the PRP-DC method, the direction $d_k (k \geq 1)$ is determined by

$$d_k = -\frac{y_{k-1}^T s_{k-1}}{\|\nabla f(x_{k-1})\|^2} \nabla f(x_k) + \frac{y_{k-1}^T \nabla f(x_k)}{\|\nabla f(x_{k-1})\|^2} s_{k-1} - \frac{s_{k-1}^T \nabla f(x_k)}{\|\nabla f(x_{k-1})\|^2} y_{k-1}, \tag{4}$$

where $s_{k-1} = x_k - x_{k-1}$.

As the computation of the gradient ∇f is not available in the direct search methods, we need to find some estimation of the gradient ∇f instead of ∇f in (2), (3) and (4). Based on these conjugate gradient directions, we then design direct search methods. Under mild conditions, we will prove that these methods are globally convergent. Our numerical results show that the proposed methods are efficient.

In the next section, we describe our direct search algorithm framework. In Section 3 we establish a global convergence theorem for this algorithm framework. In Section 4, we compare the performance of the proposed methods with some existing methods.

2. The Algorithm

In this section, we first introduce some concepts and then describe our algorithms.

Definition 2.1. A set of vectors V_+ is called a positive basis for R^n if and only if

- 1) every vector in R^n is a nonnegative linear combination of the members of V_+ , and
- 2) no proper subset of V_+ satisfies 1).

Positive basis was first proposed in [25], and has become an important concept in direct search methods. In general, if $\{v_1, v_2, \dots, v_n\}$ is a basis for R^n , then $\{v_1, v_2, \dots, v_n, -v_1, -v_2, \dots, -v_n\}$ is a positive basis for R^n . There are many other important positive bases, see [25] for more details.

The following two definitions define the frame and quasi-minimal frame proposed by Coope and Price in [18] and [19] respectively.

Definition 2.2. A frame round x with size $h > 0$ is defined by

$$\Phi(x, h, V_+) = \{x + hv : v \in V_+\}$$

where V_+ is a positive basis.

Definition 2.3. A frame $\Phi(x, h, V_+)$ is called quasi-minimal if and only if

$$f(x) \leq f(x + hv) + \varepsilon, \quad \forall v \in V_+, \quad (5)$$

where $\varepsilon = h^\mu$, and $\mu > 1$ is a given constant.

If a frame $\Phi(x, h, V_+)$ is quasi-minimal, then the frame center x is called a quasi-minimal point. If there is a frame point $y \in \Phi(x, h, V_+)$ such that

$$f(y) - f(x) < -\varepsilon, \quad (6)$$

then the frame is not quasi-minimal, and we say that the function value at y is sufficiently less than that of the frame center or that at the frame point y the objective function obtain sufficient decrease. In a frame which is not a quasi-minimal frame, there is at least a frame point at which the objective function can obtain sufficient decrease.

In this paper, we use the well-known positive basis $V_+ = \{e_1, \dots, e_n, -e_1, \dots, -e_n\}$ to define a quasi-minimal frame, where e_i is the i^{th} unit vector.

We now describe our PRP-type direct search algorithm.

Firstly, a frame is constructed centered at the current iterate x_k . Using the function values at the frame points, we estimate g_k as an approximation to the gradient $\nabla f(x_k)$. The i^{th} element of g_k is given by

$$\frac{f(x_k + h_k e_i) - f(x_k - h_k e_i)}{2h_k}$$

The next search direction is then calculated according to (2) for TTPRP (or (3) for TMPRP or (4) for PRP-DC) in which $\nabla f(x_k)$ is replaced by g_k .

We use the line search method proposed in [19] to get a step-length α_k . Specifically, we solve the following one dimension optimization problem to get α_k :

$$\min_{\alpha \in \mathbb{R}} \varphi(\alpha) = f(x_k + \alpha h_k d_k / \|d_k\|), \quad (7)$$

where the current iterate x_k , the frame size h_k and the conjugate direction d_k are given. The next iterate is then set to be $x_{k+1} = x_k + \alpha h_k d_k / \|d_k\|$. If the current frame is quasi-minimal, then decrease the frame size by letting $h_{k+1} = h_k/4$. Otherwise, if α_k is large enough such that $\alpha_k \geq 2 + 2\sqrt{n}$, then increase the frame size by letting $h_{k+1} = 5h_k/2$.

As the search direction may not be a descent direction, the step-length α_k determined by (7) may be negative.

In our method, we adopt the n -step restart strategy. Because the n -step restart conjugate gradient method possesses n -step quadratic convergence property when applied to minimizing a twice continuously differentiable function. Specifically, at iterate $kn+1$, we use $-g_{kn+1}$ as the search direction. For the purpose of convergence, we need to set x_{kn+1} equal to the lowest known point

before the next reset. That is to say, we let x_{kn+1} be determined by the following rule.

$$f(x_{kn+1}) = \min_{\substack{1 \leq i \leq n \\ (k-1)n \leq j \leq kn}} \left\{ f(x_j), f(x_j \pm h_j e_i) \right\}, \quad (8)$$

We summarize the above process as the algorithm below.

Algorithm.

- 1) Initialize: Set $k = 1, j = n + 1, h_0 = 1$.
- 2) **While** (stopping conditions do not hold) **do**
 - a) Construct a frame centered at the current iterate x_k . Calculate the function values at the frame points, and form the gradient estimation g_k .
 - b) Check stopping conditions.
 - c) Calculate the new search direction d_k .
 - d) Execute the line search process to find α_k .
 - e) Set $x_{k+1} = x_k + \alpha h_k d_k / \|d_k\|$.
 - f) If the current frame is quasi-minimal, then set $h_{k+1} = h_k/4$; else if $\alpha_k \geq 2 + 2\sqrt{n}$, then set $h_{k+1} = 5h_k/2$; else set $h_{k+1} = h_k$. Increase k by one and decrease j by one respectively.
 - g) If $j > 1$ then return to step (a); else
 - set x_{k+1} equal to the lowest known point according to (8);
 - set $j = n + 1$, increase k by one.

end.

In the above algorithm, the variable j counts the number of iterates until next restart.

3. Convergence Analysis

If we regard all the conjugate gradient steps as part of the finite process, then the general convergence result of the frame-based method will still hold [18]. In this section, we describe the convergence result specified for our algorithm.

Suppose the sequence of quasi-minimal points generated by the algorithm is denoted by $\{z_m\}$. The following theorem shows that $\{z_m\}$ must be infinite and each limit points of $\{z_m\}$ is a stationary point of the objective function.

Theorem 3.1. Assume that

- 1) the sequence of iterates $\{x_k\}$ generated by the algorithm is bounded;
- 2) f is continuously differentiable; and
- 3) $h_k \rightarrow 0$ as $k \rightarrow +\infty$.

Then $\{z_m\}$ is an infinite sequence whose cluster points are stationary points of f .

Proof. Firstly, we prove that the algorithm generates an infinite sequence of quasi-minimal points. Suppose on the contrary that the sequence of quasi-minimal points is finite. Let the final quasi-minimal point be z_m . After the final quasi-minimal iterate z_m is found, the parameter

h will never decrease, and (6) will hold for all $k > m$. This implies the sequence of the function values is unbounded. It contradicts condition 1). Consequently, the sequence of quasi-minimal iterates $\{z_m\}$ is infinite.

Let z_∞ be an arbitrary cluster point of $\{z_m\}$ and the subsequence $\{z_m\}_K$ converges to z_∞ . Then for any $z_m \in \{z_m\}_K$, we have

$$f(z_m \pm \hat{h}_m e_i) \geq f(z_m) - (\hat{h}_m)^\mu, \quad i = 1, \dots, n,$$

where \hat{h}_m is the frame size corresponding to the quasi-minimal point z_m . So we get

$$\frac{f(z_m \pm \hat{h}_m e_i) - f(z_m)}{\hat{h}_m} - (\hat{h}_m)^{\mu-1} \geq 0, \quad i = 1, \dots, n$$

Letting $m \rightarrow +\infty$ with $k \in K$ we obtain

$$\nabla f(z_\infty)^T e_i \geq 0, \quad i = 1, \dots, n$$

and

$$-\nabla f(z_\infty)^T e_i \geq 0, \quad i = 1, \dots, n.$$

These two inequalities yield $\nabla f(z_\infty) = 0$. \square

4. Numerical Experiments

In this section, we report some numerical results. We compare the proposed direct search methods (denoted as TTPRP, TMPRP and PRP-DC respectively) with the preconditioned PRP+ direct search method (denoted as PRP+(pre)) proposed in [19].

We adopt both the performance profile [26] and the data profile [27] to compare the performance among different methods.

The performance profile seeks to capture how well one solver performs relative to the other selected solvers on the set of test problems. While the data profile display the raw data. In particular, the data profile seeks to tell the percentage of the problems that can be solved (for a given tolerance τ) with any given number of function evaluations. So the data profile is especially suit for the situation where the function evaluation is expensive [27].

We use the 53 smooth test problems proposed in [27].

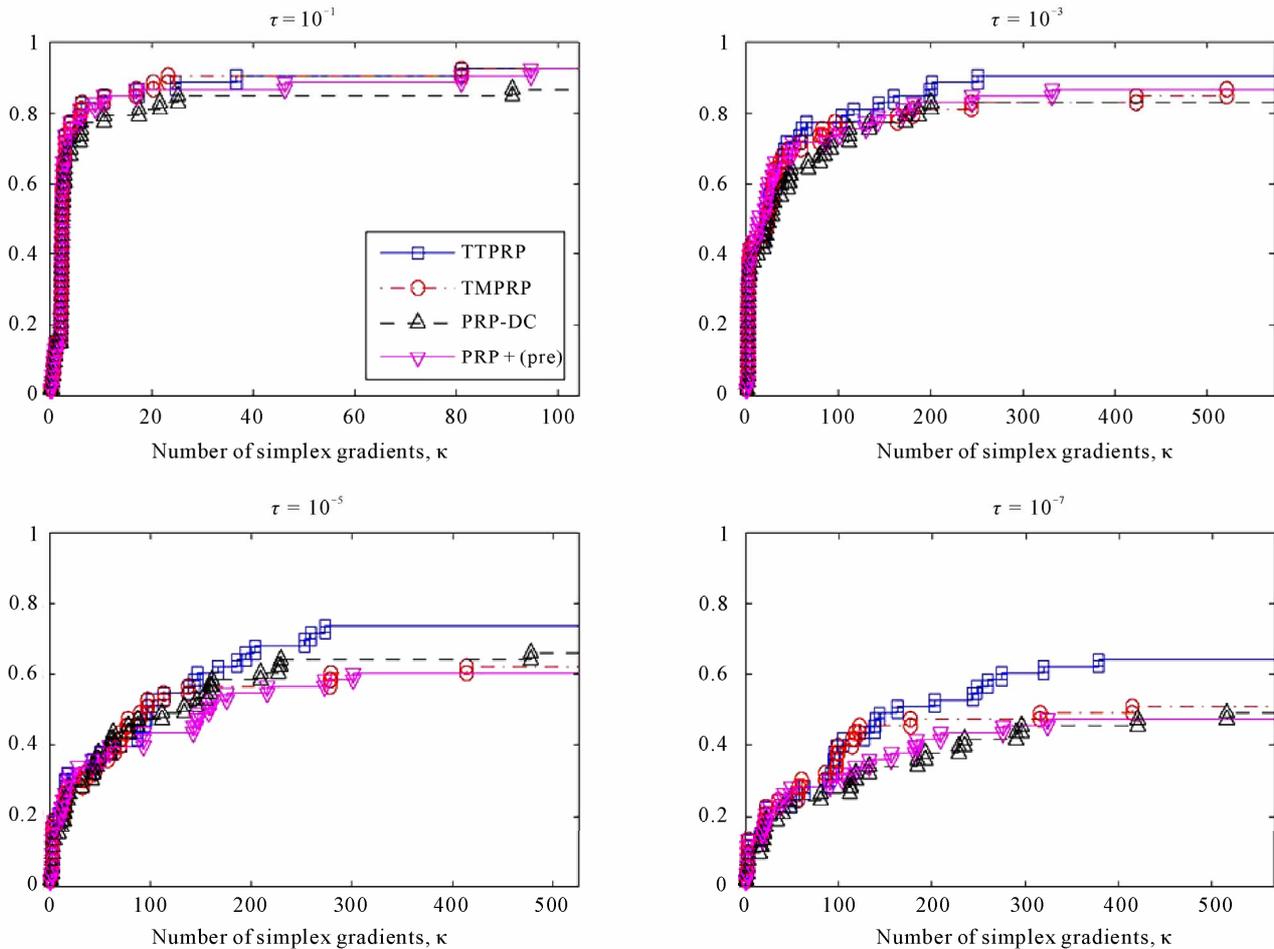


Figure 1. Data profiles show the percentage of problems solved as a function of a computational budget of simplex gradients.

The maximum dimension of these problems is 12. All methods halt when the following convergence test proposed in [27] was satisfied

$$f(x) \leq f_L + \tau(f(x_0) - f_L), \tag{9}$$

where $\tau > 0$ is a tolerance, x_0 is the starting point for the test problem, and f_L is computed for each solver as the smallest value of f obtained by any solver within 1300 function evaluations. Because we are interested in the short-term behavior of these methods as the accuracy level changes, we present the data profiles and the performance profiles for $\tau = 10^{-k}$ with $k \in \{1, 3, 5, 7\}$.

In our numerical experiments, if the current frame is quasi-minimal, then the frame size is reduced by the formula $h_{k+1} = \max\{h_{\min}, h_k/4\}$, where $h_{\min} = 10^{-10}$. The role of h_{\min} is to ensure that h would not be too close to the machine precision. If h gets too close to the machine precision then the gradient estimates may become completely inaccurate.

The codes were written in Fortran 90, and the program was run on a PC with a Genuine Intel (R) CPU (T1350@

1.86 GHz) and 504 M memory.

Figure 1 shows the data profiles for tolerance $\tau = 10^{-1}, 10^{-3}, 10^{-5}$ and 10^{-7} respectively. It can be seen that TTPRP solves the largest percentage of problems for almost all size of computational budget and levels of accuracy τ . TMPRP and PRP-DC are also comparable with PRP + (pre). It is noteworthy that the performance differences between TTPRP and other solvers tend to increase as the tolerance τ decreases and the computational budget κ increases. On the other hand, **Figure 1** also shows that within about 50 simplex gradients the performance differences between four solvers are small.

Figure 2 shows the performance profiles based on the number of function evaluations. The left side of each plot gives the percentage of the test problems which a solver can solve with the greatest efficiency; the right side gives the percentage of the test problems which a solver can solve successfully. Mainly, the right side measures the robustness of a solver.

It can be seen from **Figure 2** that TTPRP solves the largest percentage of problems for almost all size of per-

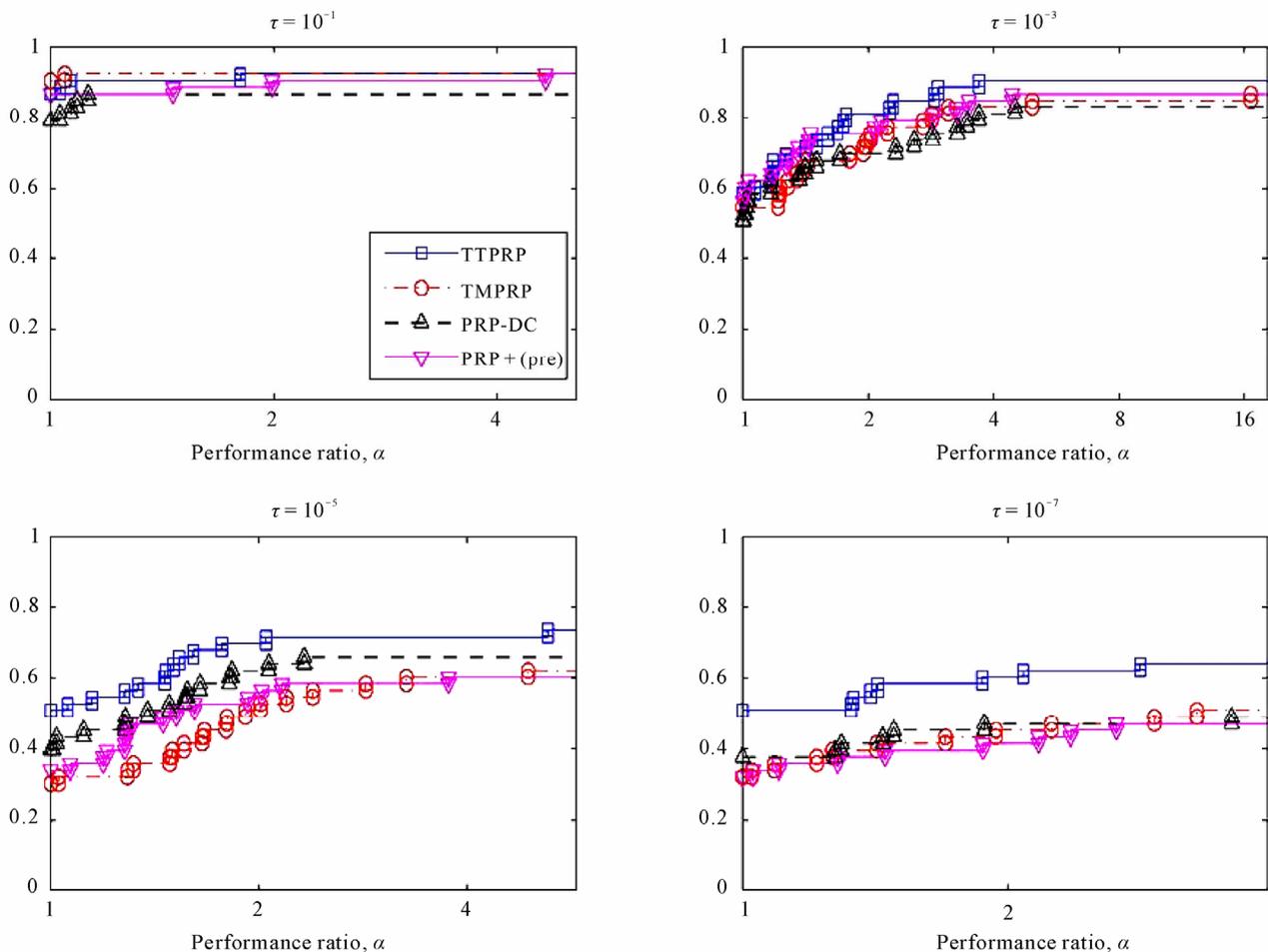


Figure 2. Performance profiles (logarithmic scale).

formance ratio and levels of accuracy τ . TMRP and PRP-DC are also comparable with PRP + (pre). It is noteworthy that the performance differences between TTPRP and other solvers still tend to increase as the tolerance τ decreases.

5. Discussion

Based on the Coope-Price direct search framework, we proposed three PRP-type direct search methods. All of them employ a kind of descent conjugate gradient direction. When exact gradients are available, descent conjugate gradient methods can generate sufficient descent directions for the objective function, and numerical results have shown that they are often computationally more effective than PRP+ method.

In this paper, the gradient information of the objective function is not available. We estimate the gradients based on the function values obtained at the maximal positive basis. These gradients estimated may be not accurate very much, but global convergence can be ensured under the Coope-Price direct search framework. In other words, convergence is guaranteed by the frame-based nature of the algorithms, not the fact that they mimic a conjugate gradients method.

The accuracy of the gradients estimated may also affect the descent property of the descent conjugate gradient directions. However, the numerical results show that the proposed PRP-type direct search methods are promising and competitive, especially the TTPRP method.

6. References

- [1] W. C. Davidon, "Variable Metric Method for Minimization," *SIAM Journal on Optimization*, Vol. 1, No. 1, 1992, pp. 1-17. [doi:10.1137/0801001](https://doi.org/10.1137/0801001)
- [2] R. Hooke and T. A. Jeeves, "Direct Search Solution of Numerical and Statistical Problems," *Journal of the ACM*, Vol. 8, No. 2, 1961, pp. 212-229. [doi:10.1145/321062.321069](https://doi.org/10.1145/321062.321069)
- [3] M. J. D. Powell, "Direct Search Algorithms for Optimization Calculations," *Acta Numerica*, Vol. 7, No. 1, 1998, pp. 287-336. [doi:10.1017/S0962492900002841](https://doi.org/10.1017/S0962492900002841)
- [4] A. Conn, K. Scheinberg and P. L. Toint, "On the Convergence of Derivative-Free Methods for Unconstrained Optimization," In: M. D. Buchmann and A. Iserles, Eds., *Approximation Theory and Optimization*, Cambridge University Press, Cambridge, 1997.
- [5] B. Karasozen, "Survey of Trust-Region Derivative Free Optimization Methods," *Journal of Industrial and Management Optimization*, Vol. 3, No. 2, 2007, pp. 321-334.
- [6] M. J. D. Powell, "A View of Algorithms for Optimization without Derivatives," *Technical Report DAMTP2007/NA03*, Department of Applied Mathematics and Theoretical Physics, University of Cambridge, Cambridge, 2007.
- [7] R. M. Lewis and V. Torczon, "Rank Ordering and Positive Bases in Pattern Search Algorithms," *Technical Report 96-71*, Institute for Computer Applications in Science and Engineering, NASA Langley Research Center, Hampton, 1996.
- [8] V. Torczon, "On the Convergence of the Multidirectional Search Algorithm," *SIAM Journal on Optimization*, Vol. 1, No. 1, 1991, pp. 123-145. [doi:10.1137/0801010](https://doi.org/10.1137/0801010)
- [9] V. Torczon, "On the Convergence of Pattern Search Algorithms," *SIAM Journal on Optimization*, Vol. 7, No. 1, 1997, pp. 1-25. [doi:10.1137/S1052623493250780](https://doi.org/10.1137/S1052623493250780)
- [10] R. M. Lewis and V. Torczon, "Pattern Search Algorithms for Bound Constrained Minimization," *SIAM Journal on Optimization*, Vol. 9, No. 4, 1999, pp. 1082-1099. [doi:10.1137/S1052623496300507](https://doi.org/10.1137/S1052623496300507)
- [11] R. M. Lewis and V. Torczon, "Pattern Search Methods for Linearly Constrained Minimization," *SIAM Journal on Optimization*, Vol. 10, No. 3, 2000, pp. 917-941. [doi:10.1137/S1052623497331373](https://doi.org/10.1137/S1052623497331373)
- [12] C. Audet and J. E. Dennis Jr., "Analysis of Generalized Pattern Searches," *SIAM Journal on Optimization*, Vol. 13, No. 3, 2003, pp. 889-903. [doi:10.1137/S1052623400378742](https://doi.org/10.1137/S1052623400378742)
- [13] C. Audet and J. E. Dennis Jr., "Mesh Adaptive Direct Search Algorithms for Constrained Optimization," *SIAM Journal on Optimization*, Vol. 17, No. 1, 2006, pp. 188-217. [doi:10.1137/040603371](https://doi.org/10.1137/040603371)
- [14] I. D. Coope and C. J. Price, "On the Convergence of Grid-Based Methods for Unconstrained Optimization," *SIAM Journal on Optimization*, Vol. 11, No. 4, 2001, pp. 859-869. [doi:10.1137/S1052623499354989](https://doi.org/10.1137/S1052623499354989)
- [15] T. G. Kolda, R. M. Lewis and V. Torczon, "Optimization by Direct Search: New Perspectives on Some Classical and Modern Methods," *SIAM Review*, Vol. 45, No. 3, 2003, pp. 385-482. [doi:10.1137/S003614450242889](https://doi.org/10.1137/S003614450242889)
- [16] T. G. Kolda, R. M. Lewis and V. Torczon, "Stationary Results for Generating Set Search for Linearly Constrained Optimization," *SIAM Journal on Optimization*, Vol. 17, No. 4, 2006, pp. 943-968. [doi:10.1137/S1052623403433638](https://doi.org/10.1137/S1052623403433638)
- [17] R. M. Lewis, A. Shepherd and V. Torczon, "Implementing Generating Set Search Methods for Linearly Constrained Minimization," *SIAM Journal on Scientific Computing*, Vol. 29, No. 6, 2007, pp. 2507-2530. [doi:10.1137/050635432](https://doi.org/10.1137/050635432)
- [18] I. D. Coope and C. J. Price, "Frame Based Methods for Unconstrained Optimization," *Journal of Optimization Theory and Applications*, Vol. 107, No. 2, 2000, pp. 261-274. [doi:10.1023/A:1026429319405](https://doi.org/10.1023/A:1026429319405)
- [19] I. D. Coope and C. J. Price, "A Direct Search Frame-Based Conjugate Gradients Method," *Journal of Computational Mathematics*, Vol. 22, No. 4, 2004, pp. 489-500.
- [20] W. Y. Cheng, "A Two-Term PRP-Based Descent Method," *Numerical Functional Analysis and Optimization*, Vol. 28, No. 11-12, 2007, pp. 1217-1230. [doi:10.1080/01630560701749524](https://doi.org/10.1080/01630560701749524)
- [21] W. W. Hager and H. Zhang, "A New Conjugate Gradient

- Method with Guaranteed Descent and an Efficient Line Search,” *SIAM Journal on Optimization*, Vol. 16, No. 1, 2005, pp. 170-192. [doi:10.1137/030601880](https://doi.org/10.1137/030601880)
- [22] W. W. Hager and H. Zhang, “A Survey of Nonlinear Conjugate Gradient Methods,” *Pacific Journal of Optimization*, Vol. 2, 2006, pp. 35-58.
- [23] L. Zhang, W. J. Zou and D. H. Li, “A Descent Modified Polad-Ribiere-Polyak Conjugate Gradient Method and Its Global Convergence,” *IMA Journal of Numerical Analysis*, Vol. 26, 2006, pp. 629-640. [doi:10.1093/imanum/drl016](https://doi.org/10.1093/imanum/drl016)
- [24] N. Andrei, “A Modified Polak-Ribiere-Polyak Conjugate Gradient Algorithm for Unconstrained Optimization,” *Optimization: A Journal of Mathematical Programming and Operational Research*, Accepted.
- [25] C. Davis, “Theory of Positive Linear Dependence,” *American Journal of Mathematics*, Vol. 76, No. 4, 1954, pp. 733-746. [doi:10.2307/2372648](https://doi.org/10.2307/2372648)
- [26] E. D. Dolan and J. J. More, “Benchmarking Optimization Software with Performance Profiles,” *Mathematical Programming*, Vol. 91, No. 2, 2002, pp. 201-213. [doi:10.1007/s101070100263](https://doi.org/10.1007/s101070100263)
- [27] J. J. More and S. M. Wild, “Benchmarking Derivative-Free Optimization Algorithms,” *SIAM Journal on Optimization*, Vol. 20, No. 1, 2009, pp. 172-191. [doi:10.1137/080724083](https://doi.org/10.1137/080724083)