RAIRO-Oper. Res. 57 (2023) 817–835
https://doi.org/10.1051/ro/2023037

RAIRO Operations Research
www.rairo-ro.org

A MODIFIED NONLINEAR CONJUGATE GRADIENT ALGORITHM FOR UNCONSTRAINED OPTIMIZATION AND PORTFOLIO SELECTION PROBLEMS

Thamiso Diphofu, Professor Kaelo* and Abebe R. Tufa

Abstract. Conjugate gradient methods play a vital role in finding solutions of large-scale optimization problems due to their simplicity to implement, low memory requirements and as well as their convergence properties. In this paper, we propose a new conjugate gradient method that has a direction satisfying the sufficient descent property. We establish global convergence of the new method under the strong Wolfe line search conditions. Numerical results show that the new method performs better than other relevant methods in the literature. Furthermore, we use the new method to solve a portfolio selection problem.

Mathematics Subject Classification. 90C06, 90C30, 65K05.

Received June 5, 2022. Accepted March 25, 2023.

1. Introduction

Conjugate gradient (CG) methods have become a widely attractive option for solving large scale unconstrained optimization problems. They are easy to implement and have good global convergence properties. Unlike Newton and quasi-Newton methods, they require low memory as they do not store any matrices. Conjugate gradient methods are applicable in areas such as reconstruction of radial magnetic resonance (MR) images [41], portfolio selection [1,5,11], motion control problems [2,3], compressive sensing and image restoration problems [24,40,42].

Generally, large-scale unconstrained optimization problems are of the form

$$\min f(x), \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable function. To solve (1), CG methods follow the iterative scheme

$$x_{k+1} = x_k + \alpha_k d_k, \quad k = 0, 1, 2, \ldots,$$

where x_k and x_{k+1} are the current and next iteration points, respectively, α_k is the step length and d_k is the search direction. A single iteration moves a point x_k to a new point x_{k+1} along a search direction d_k, taking a step size α_k. Here, α_k can be determined by using an exact or inexact line search and the direction d_k is given

Keywords. Conjugate gradient, global convergence, strong Wolfe line search, portfolio selection.

Department of Mathematics, University of Botswana, Private Bag UB00704, Gaborone, Botswana.
*Corresponding author: kaelop@ub.ac.bw

© The authors. Published by EDP Sciences, ROADEF, SMAI 2023

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (https://creativecommons.org/licenses/by/4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

by

$$d_k = \begin{cases} -g_k, & k = 0, \\ -g_k + \beta_k d_{k-1}, & k \geq 1, \end{cases}$$

where $g_k = \nabla f(x_k)$ and β_k is a CG parameter which determines the CG method. Over the years, researchers have exploited numerous ways in which β_k can be chosen. Amongst those are the classical Hestenes and Stiefel (HS) [19], Fletcher and Reeves (FR) [18], Polak, Ribiére and Polyak (PRP) [32,33], Dai and Yuan (DY) [10], Liu and Storey (LS) [26] and the Conjugate Descent (CD) [17].

Research has shown that of these classical CG methods, the PRP, HS and LS methods have good computational performance, and the other three have good convergence properties. Hence, a lot of recent research has continued into constructing CG algorithms which possess both good numerical performance and convergence properties. This has led to the development of variations of CG methods such as hybrid CG methods [13,15,31], spectral CG methods [5,21] and three-term CG methods [20,23,27,36], among others.

In an attempt to improve numerical performance by incorporating both the gradient and function values information, Yin *et al.* [39] recently proposed a CG algorithm which satisfies the sufficient descent property

$$d_k^T g_k \leq -c\|g_k\|^2, \;\; c > 0, \qquad \forall k \geq 0,$$

and the trust region feature

$$\|d_k\| \leq \hat{c}\|g_k\|, \;\; \hat{c} > 0, \qquad \forall k \geq 0,$$

where $\|\cdot\|$ denotes the Euclidean norm. Their direction d_k is given as

$$d_k = \begin{cases} -g_k, & k = 0, \\ -g_k + \beta_k^{\mathrm{YMPRP}} d_{k-1}, & k \geq 1, \end{cases}$$

where

$$\beta_k^{\mathrm{YMPRP}} = \frac{\min\left\{\left|g_k^{\mathrm{T}} \tilde{y}_{k-1}^*\right|, \left|g_k^{\mathrm{T}} y_{k-1}\right|\right\}}{\max\left\{\mu\|d_{k-1}\|\|y_{k-1}\|, \|g_{k-1}\|^2\right\}}, \quad \mu > 1$$

and

$$\tilde{y}_{k-1}^* = y_{k-1} + \frac{\rho_{k-1}}{\|s_{k-1}\|^2} s_{k-1}, \quad \rho_{k-1} = 2(f_{k-1} - f_k) + (g_{k-1} + g_k)^{\mathrm{T}} s_{k-1},$$

with $s_{k-1} = x_k - x_{k-1}$ and $y_{k-1} = g_k - g_{k-1}$. Global convergence was established under the Armijo line search condition, that is, to find a step length $\alpha_k = \max\{\hat{\tau}\rho^i | i = 0, 1, 2, \ldots\}$ satisfying the Armijo condition

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \delta \alpha_k g_k^T d_k, \tag{2}$$

where $\hat{\tau} > 0$ and $\rho, \delta \in (0, 1)$.

Another way to improve the numerical performance of a CG method is to incorporate the use of a restart feature in the direction scheme [25]. Using this idea, Jiang *et al.* [23] proposed a three term CG method with a restart feature and gave their search direction as

$$d_k = \begin{cases} -g_k, & k = 0, \\ -g_k + \beta_k^{\mathrm{JJSL}} d_{k-1}, & k \geq 1 \text{ and } 0 \leq g_k^T g_{k-1} < \|g_k\|^2 \leq \|g_{k-1}\|^2, \\ -g_k + \hat{\zeta} \frac{g_k^T g_{k-1}}{\|g_{k-1}\|^2} g_{k-1}, & \text{otherwise}, \end{cases}$$

where $0 < \hat{\zeta} < 1$ and

$$\beta_k^{\text{JJSL}} = \frac{\|g_k\|^2 - g_k^T g_{k-1}}{\|g_{k-1}\|^2 - g_k^T g_{k-1}}.$$

They showed that the method is globally convergent when using the strong Wolfe line search conditions given by (2) and

$$\left| g(x_k + \alpha_k d_k)^T d_k \right| \leq \sigma \left| g_k^T d_k \right|,$$

where $0 < \delta < \sigma < 1$.

Delladji *et al.* [12] proposed a hybrid CG algorithm where the direction d_k, which satisfies the conjugacy condition $d_k^T y_{k-1} = 0$ at every iteration, is generated by the rule

$$d_k = -g_k + \beta_k^{\text{hFRBA}} d_{k-1}, \quad d_0 = -g_0.$$

Their CG parameter is taken to be the convex combination of β_k^{FR} and β_k^{BA} given as

$$\beta_k^{\text{hFRBA}} = (1 - \theta_k)\beta_k^{\text{FR}} + \theta_k \beta_k^{\text{BA}},$$

where $\beta_k^{\text{BA}} = \|y_{k-1}\|^2 / d_{k-1}^T y_{k-1}$, $\beta_k^{\text{FR}} = \|g_k\|^2 / \|g_{k-1}\|^2$ and θ_k is calculated as

$$\theta_k = \begin{cases} 0, & \text{if } \bar{\theta_k} \leq 0, \\ \bar{\theta_k}, & \text{if } 0 < \bar{\theta_k} < 1, \\ 1, & \text{if } \bar{\theta_k} \geq 1, \end{cases}$$

with

$$\bar{\theta_k} = \frac{g_k^T y_{k-1} \|g_{k-1}\|^2 - \|g_k\|^2 d_{k-1}^T y_{k-1}}{\|y_{k-1}\|^2 \|g_{k-1}\|^2 - \|g_k\|^2 d_{k-1}^T y_{k-1}}.$$

Global convergence of the method was established under the strong Wolfe line search.

One other CG method that satisfies the sufficient descent property and possesses the trust region feature is that proposed by Wu [37]. In this method, the direction is calculated as

$$d_k = \begin{cases} -g_k, & k = 0, \\ -g_k + \frac{g_k^T y_{k-1} d_{k-1} - d_{k-1}^T g_k y_{k-1}}{\gamma_1 \|g_{k-1}\|^2 + \gamma_2 \|d_{k-1}\| \|y_{k-1}\| + \gamma_3 \|d_{k-1}\| \|g_{k-1}\|}, & k \geq 1, \end{cases}$$

where $\gamma_1, \gamma_2, \gamma_3 > 0$ are constants. Global convergence of the method was established under the Wolfe line search conditions given by (2) and

$$g(x_k + \alpha_k d_k)^T d_k \geq \sigma g_k^T d_k, \tag{3}$$

where $0 < \delta < \sigma < 1$. The numerical results show that their method is effective and reliable when compared with relevant methods in the literature.

In [16], Faramarzi and Amini proposed a modified spectral CG method and showed that the direction satisfies the sufficient descent property. Global convergence was established under the strong Wolfe line search. The method was shown to be superior when compared to other methods in the literature. And in Kou and Dai [25], a three-term CG method with a restart procedure is presented. The three-term CG method is given as

$$d_k = -g_k + \beta_k d_{k-1} + \gamma_k y_{k-1},$$

with

$$\beta_k = \max\left\{ \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}} - \left(\frac{s_{k-1}^T y_{k-1}}{\|s_{k-1}\|^2} + \frac{\|y_{k-1}\|^2}{s_{k-1}^T y_{k-1}} \right) \frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}}, \zeta \frac{g_k^T d_{k-1}}{\|d_{k-1}\|^2} \right\}, \quad 0 < \zeta < 1,$$

and

$$\gamma_k = \xi_k \frac{g_k^T d_{k-1}}{d_{k-1}^T y_{k-1}}, \quad 0 \leq \xi_k \leq 1.$$

The method uses a restart feature whenever a truncation happens, that is, when

$$\beta_k = \zeta \frac{g_k^T d_{k-1}}{\|d_{k-1}\|^2},$$

the authors set $\xi_k = 0$ and the method is restarted along the direction

$$d_k = -g_k + \zeta \frac{g_k^T d_{k-1}}{\|d_{k-1}\|^2} d_{k-1}.$$

The authors showed that due to the restart procedure, the method achieves much better results.

In the next section, we propose a new CG method that satisfies the sufficient descent condition. We then establish the global convergence of this new method under the strong Wolfe line search in Section 3. We present the numerical experiments followed by an application in portfolio optimization in Section 4. Finally, in the last section we give the concluding remarks.

2. The method

In 2001, Dai and Liao [9] proposed an extension of the Hestenes–Stiefel (HS) conjugate gradient parameter

$$\beta_k^{\mathrm{HS}} = \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}}$$

using their proposed conjugacy condition

$$d_k^T y_{k-1} = -t g_k^T s_{k-1},$$

and gave the new Dai–Liao (DL) conjugate gradient parameter

$$\beta_k^{\mathrm{DL+}} = \max\left\{ \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}}, 0 \right\} - t \frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}},$$

where $t > 0$ is a scalar. This method was shown to be globally convergent for general functions. Following this, further developments of the DL method were proposed [28,38,43]. And in Dai [8], a RMIL+ conjugate gradient parameter

$$\beta_k^{\mathrm{RMIL+}} = \begin{cases} \frac{g_k^T y_{k-1}}{\|d_{k-1}\|^2}, & \text{if } 0 \leq g_k^T g_{k-1} \leq \|g_k\|^2, \\ 0, & \text{otherwise}, \end{cases}$$

is presented, as a modification to accomplish the global convergence of the RMIL CG parameter by Rivaie *et al.* [34].

Now, motivated by the ideas of these conjugate gradient parameters discussed above, specifically using the DL numerator $g_k^T(y_{k-1} - s_{k-1})$, the HS and RMIL+ CG parameters, we construct a new CG parameter

$$\beta_k^{\mathrm{DP}} = \max\{\beta_k^{\mathrm{DP}^*}, 0\}, \tag{4}$$

where

$$\beta_k^{\mathrm{DP}^*} = \frac{\min\{g_k^T(y_{k-1} - s_{k-1}), \|g_k\|^2\}}{\|d_{k-1}\|^2} - \mu \frac{|g_k^T y_{k-1}|}{\|d_{k-1}\|\|y_{k-1}\|}, \tag{5}$$

with $\mu > 0$ a constant. We choose our direction as

$$d_k = \begin{cases} -g_k, & k = 0, \\ -g_k + \beta_k^{\mathrm{DP}} d_{k-1}, & k \geq 1. \end{cases} \tag{6}$$

Notice that from (4) and (5), we get that

$$\beta_k^{\mathrm{DP}^*} \leq \frac{\min\{g_k^T(y_{k-1} - s_{k-1}), \|g_k\|^2\}}{\|d_{k-1}\|^2} \leq \frac{\|g_k\|^2}{\|d_{k-1}\|^2},$$

meaning

$$0 \leq \beta_k^{\mathrm{DP}} \leq \frac{\|g_k\|^2}{\|d_{k-1}\|^2}. \tag{7}$$

The step length α_k is determined using the strong Wolfe line search conditions. The algorithm of our new CG method is presented as follows.

Algorithm 1. A new DP Conjugate Gradient method.

Let $k = 0$. Set $\sigma, \delta \in (0, 1), \mu > 0$ and $\epsilon > 0$. Select an initial point $x_0 \in \mathbb{R}^n$.
for $k = 0, 1, \ldots$ **do**
  If $\|g_k\| \leq \epsilon$, then stop.
  Calculate d_k by (6), where β_k is evaluated using (5) and (4).
  Determine α_k satisfying

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \delta \alpha_k g_k^T d_k \tag{8}$$

  and

$$|g(x_k + \alpha_k d_k)^T d_k| \leq \sigma |g_k^T d_k|, \tag{9}$$

  where $0 < \delta < \sigma < 1$.
  Set $x_{k+1} = x_k + \alpha_k d_k$.
  Set $k = k + 1$.
end for

---

3. Global convergence

To present the convergence analysis, we make the following fundamental assumptions about the objective function f, which have been widely used in the literature.

Assumption 3.1. *The level set*

$$\Omega = \{x \in \mathbb{R}^n : f(x) \leq f(x_0)\},$$

where x_0 is the starting point, is bounded.

Assumption 3.2. *The function $f(x)$ is continuously differentiable and its gradient is Lispchitz continuous in some neighbourhood $\mathcal{N}$ of Ω, that is, there exists a constant $L > 0$ such that*

$$\|g(x) - g(y)\| \leq L\|x - y\|, \qquad \forall x, y \in \mathcal{N}.$$

Lemma 3.1. *Let the sequences $\{x_k\}$ and $\{d_k\}$ be generated by Algorithm 1 for all $k \geq 0$ with $0 < \sigma < 1/4$. Then*

$$\frac{\|g_k\|}{\|d_k\|} < 2. \tag{10}$$

Proof. Firstly, we use the fact that for any real number σ,

$$0 < \sigma < \frac{1}{4} \implies 4\sigma - 2 < -1 \implies 2\sigma - 1 < 0 \tag{11}$$

and

$$4\sigma - 2 < -1 \implies 2\sigma - 1 < -\frac{1}{2} \implies \frac{1}{1 - 2\sigma} < 2. \tag{12}$$

We prove by induction. The result (10) follows immediately for $k = 0$. Now, suppose that (10) holds for some $k \geq 0$. Re-writing (6) for some $k + 1$ and multiplying by g_{k+1}^T, we have

$$\|g_{k+1}\|^2 = -d_{k+1}^T g_{k+1} + \beta_{k+1}^{\mathrm{DP}} g_{k+1}^T d_k. \tag{13}$$

When applying the strong Wolfe line search condition (9) and using the triangle inequality on (13), we obtain

$$\|g_{k+1}\|^2 \leq \left|g_{k+1}^T d_{k+1}\right| + \sigma\beta_{k+1}^{\mathrm{DP}}\left|g_k^T d_k\right|.$$

By (7) and the Cauchy–Schwarz inequality, the above inequality gives

$$\|g_{k+1}\|^2 \leq \|g_{k+1}\|\|d_{k+1}\| + \sigma\frac{\|g_k\|}{\|d_k\|}\|g_{k+1}\|^2, \tag{14}$$

and on dividing by $\|g_{k+1}\|$, and applying the induction process (10), leads to

$$\|g_{k+1}\| < \|d_{k+1}\| + 2\sigma\|g_{k+1}\|.$$

Therefore we obtain

$$(1 - 2\sigma)\|g_{k+1}\| < \|d_{k+1}\|.$$

By (11) and (12) we get that

$$\frac{\|g_{k+1}\|}{\|d_{k+1}\|} < \frac{1}{1 - 2\sigma} < 2, \;\; \sigma \in (0, 1/4),$$

and hence (10) has been established. This completes the proof. □

Notice that from (10), if we square both sides we obtain that

$$\frac{1}{\|d_k\|^2} < \frac{4}{\|g_k\|^2}, \qquad \text{for all } \; k \geq 0. \tag{15}$$

Lemma 3.2. *Let the sequences* $\{x_k\}$ *and* $\{d_k\}$ *be generated by Algorithm 1 with* $0 < \sigma < 1/4$*. Then the sufficient descent property*

$$g_k^T d_k \leq -c\|g_k\|^2, \tag{16}$$

for all $k \geq 0$ *and constant* $c > 0$*, holds.*

Proof. By (6), we obtain (16) immediately when $k = 0$. For $k > 0$, we have

$$d_k = -g_k + \beta_k^{\mathrm{DP}} d_{k-1},$$

and multiplying by g_k^T gives

$$g_k^T d_k = -\|g_k\|^2 + \beta_k^{\mathrm{DP}} g_k^T d_{k-1}. \tag{17}$$

The strong Wolfe condition (9) gives

$$-\sigma\beta_k^{\mathrm{DP}} \left|g_{k-1}^T d_{k-1}\right| \leq \beta_k^{\mathrm{DP}} g_k^T d_{k-1} \leq \sigma\beta_k^{\mathrm{DP}} \left|g_{k-1}^T d_{k-1}\right|. \tag{18}$$

By (17) and (18), and the Cauchy–Schwarz inequality, we obtain that

$$-\|g_k\|^2 - \sigma\beta_k^{\mathrm{DP}} \|g_{k-1}\|\|d_{k-1}\| \leq g_k^T d_k \leq -\|g_k\|^2 + \sigma\beta_k^{\mathrm{DP}} \|g_{k-1}\|\|d_{k-1}\|,$$

which on dividing by $\|g_k\|^2$, and using (7), gives that

$$-1 - \sigma\frac{\|g_{k-1}\|}{\|d_{k-1}\|} \leq \frac{g_k^T d_k}{\|g_k\|^2} \leq -1 + \sigma\frac{\|g_{k-1}\|}{\|d_{k-1}\|}.$$

Finally, using (10) in the above inequality gives

$$-1 - 2\sigma < \frac{g_k^T d_k}{\|g_k\|^2} < -1 + 2\sigma. \tag{19}$$

Therefore, (16) is obtained. □

Next, we present the well-known Zountendijk condition, which was initially discussed in [44].

Lemma 3.3. *Suppose Assumptions 3.1 and 3.2 hold and* d_k *is computed by Algorithm 1. Then*

$$\sum_{k=0}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} < +\infty. \tag{20}$$

In the following theorem, we establish global convergence of Algorithm 1.

Theorem 3.1. *Let Assumptions 3.1 and 3.2 hold, and the sequences* $\{x_k\}$ *and* $\{d_k\}$ *be generated by Algorithm 1. Then*

$$\liminf_{k\to\infty} \|g_k\| = 0. \tag{21}$$

Proof. We prove by contradiction. Suppose (21) does not hold, that is, there exists a constant $\bar{\omega} > 0$ such that $\|g_k\| \geq \bar{\omega}$, for all $k \geq 0$. This means

$$\frac{1}{\|g_k\|^2} \leq \frac{1}{\bar{\omega}^2} \quad \text{for all} \quad k \geq 0. \tag{22}$$

When $d_k = -g_k + \beta_k^{\mathrm{DP}} d_{k-1}$ is re-written as $d_k + g_k = \beta_k^{\mathrm{DP}} d_{k-1}$ and squared both sides, we have

$$\|d_k\|^2 = -\|g_k\|^2 - 2g_k^T d_k + \left(\beta_k^{\mathrm{DP}}\right)^2 \|d_{k-1}\|^2. \tag{23}$$

From (19), we obtain that

$$(2-4\sigma)\|g_k\|^2 < -2g_k^T d_k < (2+4\sigma)\|g_k\|^2, \qquad \text{for all} \quad k \geq 1,$$

and hence, from (23), we obtain that

$$\|d_k\|^2 < -\|g_k\|^2 + (2+4\sigma)\|g_k\|^2 + \left(\beta_k^{\mathrm{DP}}\right)^2 \|d_{k-1}\|^2,$$

which, on using (7), leads to

$$\|d_k\|^2 < (1+4\sigma)\|g_k\|^2 + \frac{\|g_k\|^4}{\|d_{k-1}\|^2}.$$

Dividing both sides of the above inequality by $\|g_k\|^4$, and using (15), we obtain

$$\frac{\|d_k\|^2}{\|g_k\|^4} < \frac{(1+4\sigma)}{\|g_k\|^2} + \frac{4}{\|g_{k-1}\|^2},$$

and using (22), it follows that

$$\frac{\|d_k\|^2}{\|g_k\|^4} < \frac{(1+4\sigma)}{\bar{\omega}^2} + \frac{4}{\bar{\omega}^2}, \qquad \text{for all} \;\; k \geq 0,$$

which implies that

$$\frac{\|g_k\|^4}{\|d_k\|^2} > \frac{\bar{\omega}^2}{5+4\sigma}, \qquad \text{for all} \;\; k \geq 0.$$

We now have that

$$\sum_{k=0}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} > \sum_{k=0}^{\infty} \frac{\bar{\omega}^2}{5+4\sigma} = \infty$$

contradicts (20). Hence (21) is true. The proof is complete. □

4. Numerical experiments

In this section, we present results obtained from running the DP method (Algorithm 1) on a set of 105 unconstrained optimization problems with dimensions varying from 60 to 10 000. We also report results of three other CG methods in the literature to compare with our method. The first two methods include IMPRP by Jian *et al.* [22] and JJSL in [23], both which use the strong Wolfe line search with $\sigma = 0.1$ and $\delta = 0.01$. And the other is the hFRBA method by Delladji [12], which also uses the strong Wolfe line search with $\sigma = 0.1$ and $\delta = 0.0001$. For our DP method, we choose $\sigma = 0.1$, $\delta = 0.01$ and $\mu = 0.2$.

The problems are taken from [4], except for Problems 13, 14 and 15, which are taken from [30]. In Table 1, we present the names of the functions (Function Name), starting points (x_0) and dimensions (Dim) of these problems. The algorithms are stopped either when the number of iterations exceeds 10 000 or when the inequality $\|g_k\| \leq \epsilon$ is satisfied, where $\epsilon = 10^{-6}$. All codes are written in MATLAB 2015b and run on a DELL desktop with Intel(R) Core(TM) i5-2400 CPU @ 3.10 GHz processor, 4 GB of RAM and Windows 10 operating system.

Table 1. Table of problems, starting points and dimensions.

Function No.	Function name	x_0	Dim
$1, 2, 3$	DIXMAANA	$[2; \ldots; 2]$	$3000, 6000, 9000$
$4, 5, 6$	DIXMAANB	$[2; \ldots; 2]$	$3000, 6000, 9000$
$7, 8, 9$	DIXMAANC	$[2; \ldots; 2]$	$3000, 6000, 9000$
$10, 11, 12$	DIXMAAND	$[2; \ldots; 2]$	$3000, 6000, 9000$
$13, 14, 15$	Penalty 1	$[1; 2; \ldots; n]$	$500, 800, 1000$
$16, 17, 18$	HIMMELBG	$[1.5; \ldots; 1.5]$	$1000, 5000, 10\,000$
$19, 20, 21$	QUARTC	$[2; \ldots; 2]$	$1000, 5000, 10\,000$
$22, 23, 24$	BDEXP	$[1; \ldots; 1]$	$1000, 5000, 10\,000$
$25, 26, 27$	Ext. DENSCHNB	$[1; \ldots; 1]$	$1000, 5000, 10\,000$
$28, 29, 30$	Ext. DENSCHNF	$[1; \ldots; 1]$	$1000, 5000, 10\,000$
$31, 32, 33$	Gen. Quartic	$[1; \ldots; 1]$	$1000, 5000, 10\,000$
$34, 35, 36$	NONSCOMP	$[3; \ldots; 3]$	$1000, 5000, 10\,000$
$37, 38, 39$	Raydan 1	$[1; \ldots; 1]$	$60, 80, 100$
$40, 41, 42$	Raydan 2	$[1; \ldots; 1]$	$1000, 5000, 10\,000$
$43, 44, 45$	Ext. Beale	$[1; 0.8; \ldots; 1; 0.8]$	$1000, 5000, 10\,000$
$46, 47, 48$	Ext. Hiebert	$[0; \ldots; 0]$	$1000, 5000, 10\,000$
$49, 50, 51$	COSINE	$[1; \ldots; 1]$	$60, 80, 100$
$52, 53, 54$	Broyden1	$[-1; \ldots; -1]$	$500, 750, 1000$
$55, 56, 57$	Broyden 2	$[-1; \ldots; -1]$	$500, 750, 1000$
$58, 59, 60$	Ext. BD1	$[0.1; \ldots; 0.1]$	$100, 250, 500$
$61, 62, 63$	Ext. Himmelblau	$[1; \ldots; 1]$	$1000, 5000, 10\,000$
$64, 65, 66$	Ext. QP2	$[1; \ldots; 1]$	$1000, 5000, 10\,000$
$67, 68, 69$	Gen. Tridiagonal 2	$[-1; \ldots; -1]$	$1000, 5000, 10\,000$
$70, 71, 72$	Diagonal 7	$[1; \ldots; 1]$	$1000, 5000, 10\,000$
$73, 74, 75$	Diagonal 8	$[1; \ldots; 1]$	$1000, 5000, 10\,000$
$76, 77, 78$	Almost Perturbed Quad.	$[0.5; \ldots; 0.5]$	$1000, 5000, 10\,000$
$79, 80, 81$	DQDRTC	$[3; \ldots; 3]$	$1000, 5000, 10\,000$
$82, 83, 84$	DIXMAANE	$[2; \ldots; 2]$	$3000, 6000, 9000$
$85, 86, 87$	DIXMAANF	$[2; \ldots; 2]$	$3000, 6000, 9000$
$88, 89, 90$	DIXMAANG	$[2; \ldots; 2]$	$3000, 6000, 9000$
$91, 92, 93$	DIXMAANH	$[2; \ldots; 2]$	$3000, 6000, 9000$
$94, 95, 96$	Ext. Rosenbrock	$[-1.2; 1; \ldots; -1.2; 1]$	$1000, 5000, 10\,000$
$97, 98, 99$	Ext. Tridiagonal 1	$[2; \ldots; 2]$	$1000, 5000, 10\,000$
$100, 101, 102$	Ext. White and Holst	$[-1.2; 1; \ldots; -1.2; 1]$	$1000, 5000, 10\,000$
$103, 104, 105$	Ext. Wood	$[-3; -1; \ldots; -3; -1]$	$1000, 5000, 10\,000$

Table 2 shows the results of the experiments in terms of the number of iterations (NI), function evaluations (FE), gradient evaluations (GE) and the time in seconds (TIME(s)) taken to solve a problem. An entry of "F" is made if the method fails to solve the problem within the maximum iterations. From Table 2, we can see that the DP method successfully solves 94% of the problems used, followed by IMPRP method at 90%, the JJSL at 89% and lastly, hFRBA method at 85%.

Another way to present the results is through performance profiles suggested by Dolan and Moré [14]. Let $\mathcal{P}$ be the set of problems used for testing, n_p be the number of problems in $\mathcal{P}$, $\mathcal{S}$ be the set of solvers (methods) in comparison and n_s be the number of solvers in $\mathcal{S}$. Here, $t_{p,s}$ is the number of iterations, number of function/gradient evaluations or CPU time in seconds obtained by solver $s \in \mathcal{S}$ in solving problem $p \in \mathcal{P}$. The performance between the solvers is based on the ratio

$$r_{p,s} = \frac{t_{p,s}}{\min\{t_{p,i} : 1 \leq i \leq n_s\}}.$$

TABLE 2. Table of number of iterations, function evaluations, gradient evaluations and time in seconds.

Func. No.	NI DP/hFRBA/ IMPRP/JJSL	FE DP/hFRBA/ IMPRP/JJSL	GE DP/hFRBA/ IMPRP/JJSL	TIME(s) DP/hFRBA/IMPRP/JJSL
1	7/11/9/7	16/24/20/16	11/13/12/10	0.2029203/0.4046821/0.4720448/0.0250138
2	7/11/9/7	16/24/20/16	11/13/12/10	0.0724956/0.336858/0.1255289/0.0929345
3	7/11/9/7	16/24/20/16	11/13/12/10	0.0823917/0.0913484/0.0644678/0.1055865
4	6/12/6/6	14/26/14/14	9/14/8/9	0.0407538/0.1858902/0.0173414/0.0257026
5	6/13/7/6	14/28/16/14	9/15/9/9	0.0648402/0.0926598/0.0366985/0.0572816
6	6/13/7/6	14/28/16/14	9/15/9/9	0.0861392/0.1278156/0.0512027/0.0810052
7	7/10/7/8	22/28/22/38	15/17/15/30	0.0548349/0.4013343/0.0272978/0.0805015
8	7/11/7/8	22/30/22/38	15/18/15/30	0.1016512/0.0993071/0.0897893/0.2370451
9	7/11/7/8	22/30/22/38	15/18/15/30	0.1133827/0.1312018/0.0741785/0.2281492
10	9/10/8/10	26/28/24/28	20/20/17/21	0.06481/0.4311522/0.0313413/0.0542843
11	9/10/8/10	26/28/24/27	20/20/17/18	0.1021964/0.1083686/0.0610977/0.098472
12	9/10/8/10	26/28/24/27	20/20/17/18	0.2307655/0.1340818/0.0847656/0.1473872
13	90/F/79/223	397/F/382/1038	338/F/347/983	0.2616384/F/0.2337048/0.8382861
14	31/47/35/947	153/217/174/3441	101/172/129/3396	0.1729841/0.2680589/0.1917131/6.3126537
15	75/42/65/552	359/362/324/1984	293/319/273/1945	0.7291155/0.6676884/0.5954261/5.3891444
16	5/5/5/6	70/77/70/96	70/77/70/96	0.0102769/1.4432297/0.0084413/0.0155215
17	5/5/5/6	70/77/70/96	70/77/70/96	0.0394021/0.058958/0.0281028/0.0530966
18	5/5/6/6	70/77/98/96	70/77/98/96	0.0626188/0.0714752/0.0696764/0.0934206
19	16/20/18/15	99/138/133/104	62/97/96/70	0.0366565/0.2222431/0.0480703/0.0476963
20	18/35/21/17	115/394/173/125	63/320/120/76	0.1720431/0.6492141/0.2738464/0.254786
21	19/40/F/F	133/444/F/F	72/367/F/F	0.3713113/1.4427897/F/F
22	6/5/6/5	71/51/71/52	71/51/71/52	0.0164773/0.078928/0.0153643/0.0139238
23	6/5/6/6	71/51/71/76	71/51/71/76	0.0607858/0.0811076/0.0613404/0.0967982
24	6/5/6/6	71/51/71/76	71/51/71/76	0.1148295/0.0827492/0.1191347/0.1774115
25	14/36/16/12	59/148/68/50	23/41/26/17	0.0050247/1.3506086/0.0037645/0.0042516
26	15/38/17/13	63/156/72/54	24/43/28/19	0.0122094/0.0529533/0.0089121/0.0720526
27	15/39/17/14	63/160/72/58	24/44/28/20	0.0171902/0.0350852/0.0161054/0.0226019
28	7/9/5/6	25/29/21/23	18/20/16/18	0.005381/0.0514334/0.0038077/0.0026941
29	7/9/6/6	25/29/23/23	18/20/17/18	0.0041828/0.0106663/0.0042254/0.0067368
30	7/9/6/6	25/29/23/23	18/20/17/18	0.0093518/0.0133769/0.0058994/0.0175653
31	7/17/9/11	22/41/25/36	15/24/16/23	0.0023611/0.0581958/0.0036525/0.0030596
32	7/15/8/10	22/36/24/33	15/21/16/21	0.005063/0.0197587/0.0114895/0.0143582
33	9/17/8/10	21/40/20/31	12/23/12/20	0.0122717/0.0172598/0.00509/0.0151504
34	46/74/41/39	153/253/142/143	64/107/61/69	0.0101395/0.096863/0.0100954/0.0281152
35	47/72/44/67	171/265/177/263	82/121/92/133	0.0240995/0.1111659/0.0234863/0.0677928
36	52/48/47/85	186/173/187/296	86/76/94/132	0.0478588/0.0837278/0.0404331/0.0967287
37	99/59/70/97	200/121/148/204	137/62/94/133	0.0155102/0.0582088/0.0083383/0.0124264
38	93/76/78/161	195/157/163/332	114/80/91/176	0.0097256/0.0099276/0.0080054/0.0235529
39	138/91/95/178	286/188/198/368	162/96/107/191	0.0163449/0.0139961/0.0116816/0.0273746
40	4/F/4/6	6/F/6/8	6/F/6/8	0.0009564/F/0.0008322/0.0017427
41	4/F/4/7	6/F/6/9	6/F/6/9	0.0032067/F/0.0024392/0.0170252
42	4/F/4/7	6/F/6/9	6/F/6/9	0.0060843/F/0.0155417/0.0119007
43	26/15/31/183	516/921/801/1402	485/902/765/1120	0.1681675/0.6535771/0.333305/0.521476
44	27/15/31/187	519/921/801/1412	486/902/765/1126	0.7688617/1.6262618/1.1779731/2.4944465
45	27/15/31/190	519/921/801/1421	486/902/765/1133	1.449777/2.7950663/2.2591828/4.8949942
46	F/F/F/2	F/F/F/7	F/F/F/3	F/F/F/0.0006288
47	F/F/F/2	F/F/F/7	F/F/F/3	F/F/F/0.0160677
48	F/F/F/2	F/F/F/7	F/F/F/3	F/F/F/0.0035872
49	10/42/14/10	33/129/46/33	22/46/30/23	0.0021863/0.120583/0.0030515/0.0029716
50	10/42/F/11	32/130/F/35	21/47/F/23	0.0020692/0.0060124/F/0.0025707
51	10/37/F/F	32/115/F/F	22/42/F/F	0.0038216/0.016707/F/F
52	40/72/41/62	127/229/132/196	47/85/50/72	0.1169038/0.7496422/0.4081055/0.2546672

Table 2. continued.

Func. No.	NI DP/hFRBA/ IMPRP/JJSL	FE DP/hFRBA/ IMPRP/JJSL	GE DP/hFRBA/ IMPRP/JJSL	TIME(s) DP/hFRBA/IMPRP/JJSL
53	41/64/42/36	128/205/134/117	46/78/50/45	0.3675242/0.765998/0.4615662/0.4778118
54	40/71/43/37	126/228/139/119	47/87/54/46	1.1796277/2.6370915/1.5174676/1.6470474
55	25/F/F/23	85/F/F/77	35/F/F/32	0.0803381/F/F/0.112593
56	25/F/22/F	85/F/87/F	35/F/34/F	0.256402/F/0.2607659/F
57	F/F/F/F	F/F/F/F	F/F/F/F	F/F/F/F
58	15/13/12/13	73/52/53/57	35/24/26/24	0.0045616/0.1771433/0.0023328/0.0050143
59	13/23/13/14	69/139/67/78	33/72/30/35	0.0101735/0.0086152/0.003257/0.0049271
60	13/13/12/13	79/74/83/75	31/36/32/30	0.0064599/0.0046525/0.0041958/0.0043941
61	18/19/13/10	62/66/48/38	26/28/22/18	0.0058668/0.1916567/0.0031298/0.0024527
62	18/19/13/10	62/66/48/38	26/28/22/18	0.0098392/0.0619388/0.0089208/0.0215949
63	18/19/13/10	62/66/48/38	26/28/22/18	0.0150082/0.0233172/0.0111292/0.0223884
64	51/F/51/2187	518/F/466/7770	395/F/342/2250	0.0514953/F/0.3468055/0.6136087
65	39/17/35/1364	315/286/401/4938	222/243/318/1464	0.1094141/0.1587747/0.1284526/1.3967898
66	34/F/30/1562	379/F/204/5606	291/F/129/1622	0.2131359/F/0.1041339/2.8816002
67	77/45/58/129	256/144/192/521	102/54/75/263	0.0162598/0.1208952/0.0124411/0.0465381
68	72/80/50/132	241/320/157/522	97/153/57/258	0.0462831/0.1054178/0.02566/0.1376725
69	83/49/47/95	315/203/181/377	150/105/87/187	0.1139157/0.1031836/0.0611129/0.1693003
70	4/4/4/4	10/10/10/10	6/6/6/6	0.0013735/0.147752/0.0073494/0.0063595
71	4/4/4/F	10/10/10/F	6/6/6/F	0.0141843/0.0448718/0.0451508/F
72	F/4/F/F	F/10/F/F	F/6/F/F	F/0.0084402/F/F
73	4/4/4/4	10/56/10/10	6/53/6/6	0.0016365/0.1066509/0.0012644/0.0017446
74	4/4/4/4	10/102/10/10	6/100/6/6	0.004499/0.172965/0.0150549/0.0155898
75	4/F/4/4	10/F/10/10	6/F/6/6	0.0114587/F/0.008629/0.0140434
76	1346/197/1085/6915	5535/790/4391/27682	1534/199/1156/6941	0.3339241/0.2855514/0.1974453/4.7146755
77	6082/426/426/898	30156/2131/2131/4234	6978/427/427/983	7.7599342/0.6020956/0.1998345/0.5979503
78	F/637/F/F	F/3187/F/F	F/639/F/F	F/0.9729758/F/F
79	151/5/5/30	533/19/19/113	178/7/7/41	0.026889/0.0354799/0.0011612/0.0083719
80	111/5/5/34	394/19/19/124	134/7/7/41	0.0473601/0.016164/0.0030939/0.0292867
81	126/5/5/15	448/19/19/55	153/7/7/16	0.1173734/0.014348/0.0049834/0.0153896
82	2028/269/1457/503	5634/797/4165/1417	5633/796/4164/1416	10.6798254/1.9672033/6.6749984/3.1161245
83	3563/369/1876/1470	9927/1097/5368/2548	9926/1096/5367/2547	39.8975025/7.9326481/17.825717/11.7149417
84	4714/445/2624/F	13153/1325/7583/F	13152/1324/7582/F	74.264629/12.5298583/40.0671068/F
85	1977/259/1331/415	5537/768/3849/1282	5536/767/3848/1281	10.585705/2.9430651/6.1342837/2.792792
86	3334/344/1541/614	9360/1023/4426/1946	9359/1022/4425/1945	38.1727556/6.5364534/15.0202879/9.6083761
87	4883/404/2278/F	13610/1203/6525/F	13609/1202/6524/F	82.4217643/8.4244906/35.0505033/F
88	1795/234/1134/9321	5019/699/3291/9629	5016/696/3288/9626	9.1134558/1.6915959/5.180053/26.0966945
89	3569/317/1847/3488	10011/948/5304/4872	10008/945/5301/4869	44.2288471/3.7033265/18.492981/18.6556728
90	4735/376/3271/F	13276/1125/9323/F	13273/1122/9320/F	76.5671336/6.4697058/53.4717682/F
91	1409/381/931/729	3910/1093/2667/2125	3907/1090/2664/2119	6.7505212/2.4810154/4.1324368/4.0930772
92	2505/576/1761/F	7031/1671/4991/F	7028/1668/4988/F	28.1805055/6.444989/19.3682549/F
93	3762/586/1746/F	10544/1620/5050/F	10541/1617/5047/F	61.8452657/7.7555071/25.5372581/F
94	38/F/22/5414	468/F/382/19041	385/F/334/5509	0.0309567/F/0.025804/0.7420138
95	38/F/24/5414	468/F/390/19041	385/F/339/5509	0.0722536/F/0.0547002/1.7618042
96	38/F/24/5414	468/F/390/19041	385/F/339/5509	0.233811/F/0.0894104/3.0854685
97	3/11/24/1590	54/80/124/4813	52/73/113/4017	0.0178698/0.2029392/0.0433441/1.3470724
98	3/13/24/1562	54/93/124/4729	52/85/113/3947	0.0828105/0.1381151/0.1823608/6.428692
99	3/13/24/1597	54/93/124/4853	52/85/113/4054	0.145161/0.2406472/0.3419167/12.4280738
100	33/22/37/1567	351/129/198/5615	279/76/113/1774	0.0957562/0.0706111/0.0473103/0.9523107
101	33/22/38/1593	351/129/203/5706	279/76/114/1800	0.4381548/0.1357867/0.2147668/4.4442587
102	33/22/38/1567	351/129/203/5615	279/76/114/1774	0.867927/0.2509225/0.4089634/8.7084066
103	1193/118/286/4215	4748/528/1369/16922	1578/228/632/4803	0.2511895/0.0837719/0.0649949/0.6664794
104	1293/287/413/4294	4966/1194/1712/17066	1559/457/601/4838	0.6200476/0.2153503/0.1812659/1.5976578
105	1563/103/376/8203	5906/477/1756/32991	1824/212/752/8768	1.0987241/0.1047728/0.3337922/5.3335165

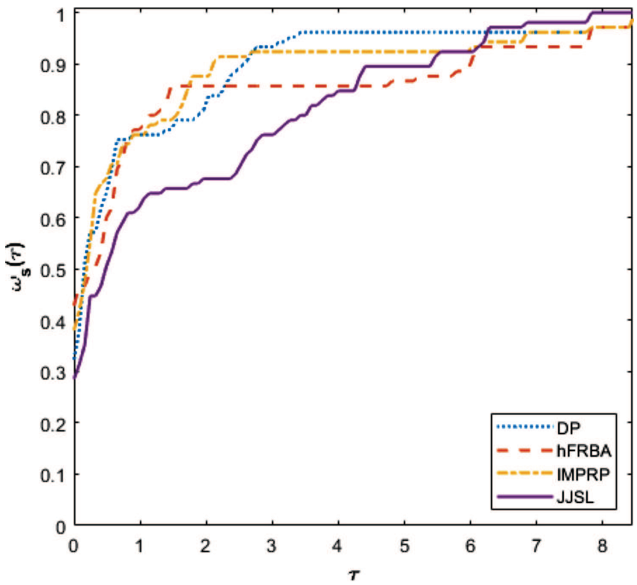

FIGURE 1. Number of iterations performance profiles.

A large enough value is chosen for $r_{p,s}$ if solver s fails to solve problem p. The performance profiles for each solver s is defined by the probability

$$\omega_s(\tau) = \frac{1}{n_p}\text{size}\{p : 1 \le p \le n_p, \ln(r_{p,s}) \le \tau\}, \quad \tau \ge 0.$$

We set $r_{p,s} = 2\max\{t_{p,s} : s \in \mathcal{S}\}$ for an entry of "F" in Table 2. The performance profiles for number of iterations, number of function evaluations, number of gradient evaluations and CPU time in seconds are shown in Figures 1, 2, 3 and 4, respectively. In all these figures, we can see that the DP method is highly competitive and efficient because its graph is always above the other graphs or among the top graphs. In particular, for values of τ between 2.3 and 5.3, where the DP is more efficient, its percentage success is as follows. In Figure 1, the DP method is the highest at 96%, followed by IMPRP with 91%, then JJSL with 87%, and finally hFRBA with 86%. For Figure 2 we have the DP method with 97%, followed by IMPRP with 93%, then JJSL with 90%, and lastly hFRBA with 88%. In Figure 3 we have the DP method with 96%, then IMPRP with 92%, followed by JJSL with 91%, and lastly hFRBA with 87%.

4.1. Application in portfolio selection

The theory of portfolio selection was initially proposed by Markowitz [29]. A stock portfolio is a group of assets or stocks owned by an investor. Investors always seek to employ the best strategy in allocating and selecting their portfolio in order to make profit while incurring some risk. A criteria that can be employed here may be one that maximizes return, minimizes risk or minimizes risk with a specific target return [6,7]. In this paper, we focus on the criteria that only minimizes risk.

Consider a portfolio consisting of m stocks. Return of stock r_i, denoted $r_{it}, 1 \le i \le m$, at time t, is defined by

$$r_{it} = \frac{P_t - P_{t-1}}{P_{t-1}},$$

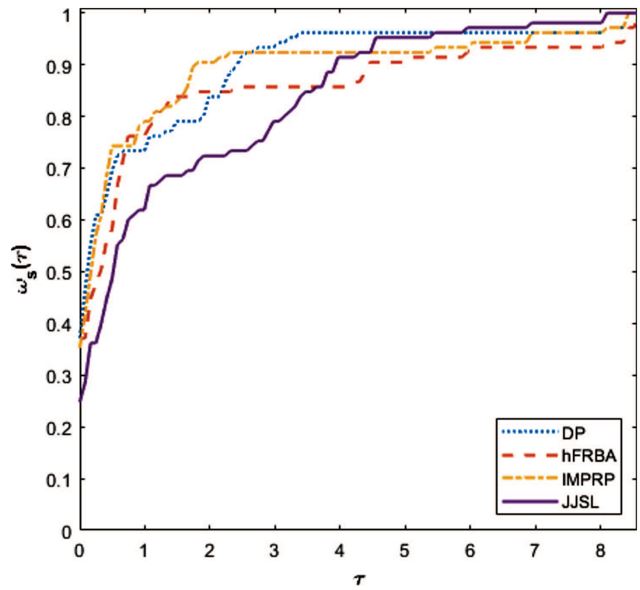

FIGURE 2. Number of function evaluations performance profiles.

where P_t and P_{t-1} is the closing price of the stock at time t and $t-1$, respectively. The mean return of the stock is defined as

$$\bar{r}_i = \frac{1}{n}\sum_{t=1}^{n} r_{it},$$

where n is the number of returns on the stock. The expected return of a portfolio of m assets is defined as

$$\bar{\mu} = E\left(\sum_{i=1}^{m} w_i R_i\right), \tag{24}$$

where R_i is the expected return of stock r_i and w_i is the corresponding weight of the stock in the portfolio. The variance of return of stock, which measures how far the asset price has moved from the mean is calculated as

$$\sigma_v^2 = \frac{1}{n-1}\sum_{t=1}^{n}(r_{it} - \bar{r}_i)^2.$$

It represents the risk of a portfolio [35]. Covariance measures the relationship between two stocks in the portfolio, a positive covariance means the stock returns move together whereas a negative covariance means stock returns move inversely. It is calculated as

$$\operatorname{cov}(r_i, r_j) = \frac{1}{n-1}\sum_{t=1}^{n}(r_{it} - \bar{r}_i)(r_{jt} - \bar{r}_j), \quad i \neq j. \tag{25}$$

We define the portfolio risk as the variance of the portfolio and denote it σ_p^2. The risk-averse portfolio optimization problem, of a portfolio with m stocks, can be formulated as

$$\begin{cases} \text{minimize} & \sigma_p^2 = X^T V X \\ \text{subject to} & \sum_{j=1}^{m} w_j = 1, \end{cases} \tag{26}$$

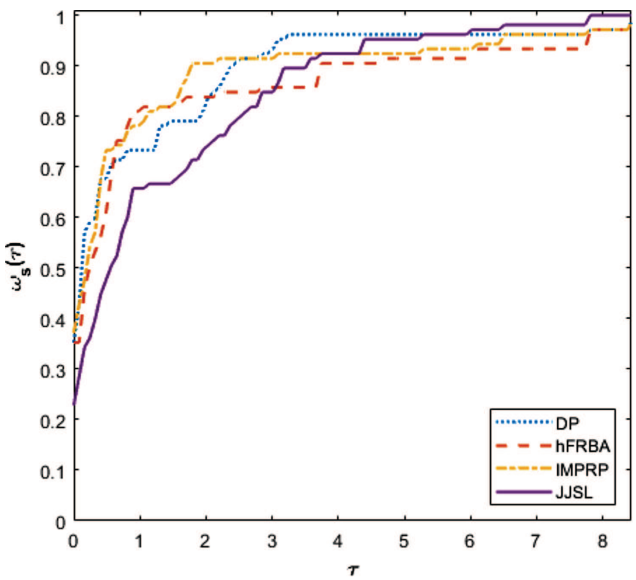

FIGURE 3. Number of gradient evaluations performance profiles.

where $X^T = [w_1, \ldots, w_m]$ and $w_1, \ldots, w_m$ are the portfolio investment weighted proportions of the stocks in the portfolio. The matrix V is the variance-covariance matrix

$$V = \begin{bmatrix} \sigma^2_{1,1} & C_{1,2} & \cdots & C_{1,m} \\ C_{2,1} & \sigma^2_{2,2} & & C_{2,m} \\ \vdots & & \ddots & \vdots \\ C_{m,1} & C_{m,2} & \cdots & \sigma^2_{m,m} \end{bmatrix},$$

where $\sigma^2_{1,1}, \sigma^2_{2,2}, \ldots, \sigma^2_{m,m}$ are the variances of the stocks and $C_{i,j}$, $i \neq j\ (i, j = 1, 2, \ldots, m)$ is the covariance between stock r_i and stock r_j computed as in (25). Here, $C_{i,j} = C_{j,i}$, hence V is symmetric.

Notice that (26) is a constrained optimization problem, which can be transformed into an unconstrained optimization problem of the form (1) by setting

$$w_m = 1 - \sum_{i=1}^{m-1} w_i.$$

Furthermore, after some algebraic computations, equation (26) can be re-written as

$$\min_{w \in \mathbb{R}^{m-1}} h(w), \tag{27}$$

where

$$h(w) = h_1(w) + h_2(w) + h_3(w),$$

with

$$h_1(w) = \sum_{i=1}^{m-1} \sum_{j=1}^{m-1} C_{i,j} w_i w_j,$$

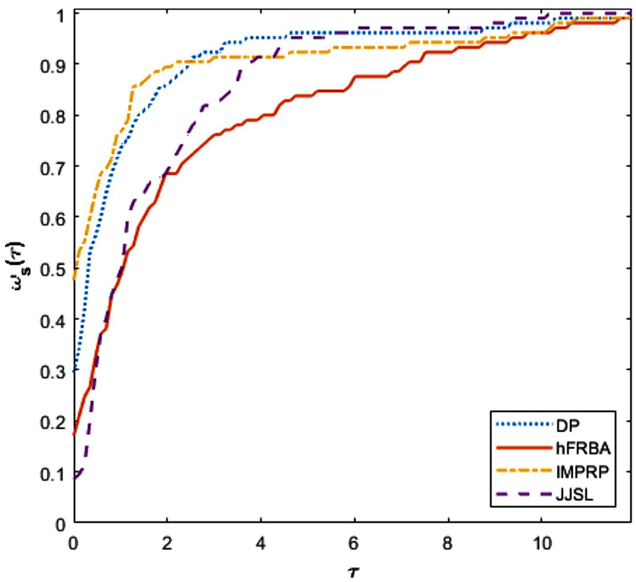

Figure 4. CPU time performance profiles.

Table 3. Table of companies.

Shoprite Holdings Ltd (SHPJ)	Gold Fields Ltd (GFIJ)
MTN Group Ltd (MTNJ)	African Rainbow Minerals Ltd (ARIJ)
Sasol Ltd (SOLJ)	Impala Platinum Holdings Ltd (IMPJ)
Anglo American Platinum Ltd (AMSJ)	Vodacom Group Ltd (VODJ)
FirstRand Ltd (FSRJ)	Discovery Holdings Ltd (DSYJ)
Reinet Investments SCA (RNIJ)	Italtile Ltd (ITEJ)
SPAR Group Ltd (SPPJ)	Investec Ltd (INLJ)
Aspen Pharmacare Holdings Ltd (APNJ)	Nedbank Group Ltd (NEDJ)
Absa Group Ltd (ABGJ)	Sanlam Ltd (SLMJ)
Nampak Ltd (NPKJ)	Bidvest Group Ltd (BVTJ)

$$h_2(w) = \left(1 - \sum_{i=1}^{m-1} w_i\right)\left(\sum_{j=1}^{m-1} [C_{m,j} + C_{j,m}]w_j\right),$$

$$h_3(w) = C_{m,m}\left(1 - \sum_{i=1}^{m-1} w_i\right)^2,$$

and $C_{1,1} = \sigma^2_{1,1}, \ldots, C_{m,m} = \sigma^2_{m,m}$.

For the numerical experiments, we choose a portfolio of 20 stocks ($m = 20$). We use weekly closing prices of 20 companies listed on the Johannesburg Stock Exchange. These companies are listed in Table 3 and the data is collected over a period spanning from 12 April 2020 to 3 April 2022 from the database `https://www.investing.com/`. Table 4 shows the mean and variances of the stocks. Table 5 shows the variance and covariances of the stocks and is set to be the symmetric matrix V in (26).

Running Algorithm 1, hFRBA, IMPRP and JJSL methods to solve (27) with initial points $w^0 = (0.1, \ldots, 0.1), (0.2, \ldots, 0.2), (0.3, \ldots, 0.3)$ and $(0.01, \ldots, 0.01)$, where $w^0 \in \mathbb{R}^{19}$, we obtained the solution

TABLE 4. Table of mean and variance.

Company	Mean	Variance	Company	Mean	Variance
SHPJ	0.005559541	0.002006374	GFIJ	0.005559541	0.005416055
MTNJ	0.010046015	0.003614818	ARIJ	0.010046015	0.002667297
SOLJ	0.013897375	0.006904869	IMPJ	0.013897375	0.00454488
AMSJ	0.005055538	0.003847571	VODJ	0.005055538	0.000598035
FSRJ	0.005354465	0.002642709	DSYJ	0.005354465	0.0025653
RNIJ	0.000955696	0.001277183	ITEJ	0.000955696	0.003734507
SPPJ	−0.001509258	0.001352478	INLJ	−0.001509258	0.00320363
APNJ	0.004324629	0.002193065	NEDJ	0.004324629	0.004754836
ABGJ	0.005917397	0.00352224	SLMJ	0.005917397	0.002216564
NPKJ	−0.001981974	0.013984408	BVTJ	−0.001981974	0.002130913

TABLE 5. Table of variance and covariance.

Stocks	SHPJ	MTNJ	SOLJ	AMSJ	FSRJ	RNIJ	SPPJ	APNJ	ABGJ	NPKJ
SHPJ	0.002006374	0.000837112	0.000593763	0.000648987	0.000955097	0.000168641	0.000790034	0.00013383	0.00092273	0.00059234
MTNJ	0.000837112	0.003614818	0.001214182	0.000294042	0.000961105	0.000539457	0.000432094	0.000506407	0.001375278	0.002225888
SOLJ	0.000593763	0.001214182	0.006904869	0.00133449	0.002036878	0.000907544	0.000694374	0.001220733	0.002709509	0.003219875
AMSJ	0.000648987	0.000294042	0.00133449	0.003847571	0.00045991	0.000193367	0.000171579	8.3232E−05	0.000752595	0.001095483
FSRJ	0.000955097	0.000961105	0.002036878	0.00045991	0.002642709	0.000522548	0.000978319	0.000578049	0.002369969	0.002483377
RNIJ	0.000168641	0.000539457	0.000907544	0.000193367	0.000522548	0.001277183	0.000249998	0.000473485	0.000511721	0.000758463
SPPJ	0.000790034	0.000432094	0.000694374	0.000171579	0.000978319	0.000249998	0.001352478	0.00015924	0.000670225	0.000343746
APNJ	0.00013383	0.000506407	0.001220733	8.3232E−05	0.000578049	0.000473485	0.00015924	0.002193065	0.000790948	0.001792901
ABGJ	0.00092273	0.001375278	0.002709509	0.000752595	0.002369969	0.000511721	0.000670225	0.000790948	0.00352224	0.003461055
NPKJ	0.00059234	0.002225888	0.003219875	0.001095483	0.002483377	0.000758463	0.000343746	0.001792901	0.003461055	0.013984408
GFIJ	0.000124516	−0.000915278	−0.000894591	0.001032211	−0.000426813	−0.000325523	−3.4217E−05	−0.000211639	−0.001254264	−0.001316273
ARIJ	0.000403483	0.000416878	0.000752523	0.001688835	0.000352815	0.000217174	0.000272456	−3.16912E−05	0.000698653	0.000475422
IMPJ	0.000706121	0.000388573	0.001775014	0.003224171	0.000652557	0.000333218	0.000368159	0.00035048	0.000983351	0.001370068
VODJ	0.000377444	0.000623905	0.000416232	0.000193383	0.000399056	0.000266102	0.00032268	0.000108792	0.000339132	0.000287675
DSYJ	0.000908004	0.000882044	0.001867293	0.000658755	0.001800071	0.000508051	0.000733969	0.000563603	0.00204399	0.002035604
ITEJ	0.000355431	0.000325167	0.000361273	0.000200412	0.00047637	0.000190791	0.000333996	−4.89722E−05	0.000659489	4.03411E−05
INLJ	0.000728251	0.001336301	0.00227889	0.000247189	0.001759383	0.000812438	0.000455951	0.000857905	0.002200869	0.003009507
NEDJ	0.001206396	0.001714483	0.003109213	0.000953871	0.002873041	0.000624696	0.000811333	0.000813532	0.003513722	0.00386657
SLMJ	0.000851299	0.001093775	0.001601518	0.000563545	0.001926325	0.00054289	0.000862478	0.000625042	0.002003833	0.002116266
BVTJ	0.000710804	0.00102461	0.001748912	0.000701574	0.001633038	0.000329798	0.000720932	0.000647785	0.00185658	0.002340973
SHPJ	0.000124516	0.000403483	0.000706121	0.000377444	0.000908004	0.000355431	0.000728251	0.001206396	0.000851299	0.000710804
MTNJ	−0.000915278	0.000416878	0.000388573	0.000623905	0.000882044	0.000325167	0.001336301	0.001714483	0.001093775	0.00102461
SOLJ	−0.000894591	0.000752523	0.001775014	0.000416232	0.001867293	0.000361273	0.00227889	0.003109213	0.001601518	0.001748912
AMSJ	0.001032211	0.001688835	0.003224171	0.000193383	0.000658755	0.000200412	0.000247189	0.000953871	0.000563545	0.000701574
FSRJ	−0.000426813	0.000352815	0.000652557	0.000399056	0.001800071	0.00047637	0.001759383	0.002873041	0.001926325	0.001633038
RNIJ	−0.000325523	0.000217174	0.000333218	0.000266102	0.000508051	0.000190791	0.000812438	0.000624696	0.00054289	0.000329798
SPPJ	−3.4217E−05	0.000272456	0.000368159	0.00032268	0.000733969	0.000333996	0.000455951	0.000811333	0.000862478	0.000720932
APNJ	−0.000211639	−3.16912E−05	0.00035048	0.000108792	0.000563603	−4.89722E−05	0.000857905	0.000813532	0.000625042	0.000647785
ABGJ	−0.001254264	0.000698653	0.000983351	0.000339132	0.00204399	0.000659489	0.002200869	0.003513722	0.002003833	0.00185658
NPKJ	−0.001316273	0.000475422	0.001370068	0.000287675	0.002035604	4.03411E−05	0.003009507	0.00386657	0.002116266	0.002340973
GFIJ	0.005416055	0.000186894	0.001216867	0.000173432	−0.00014238	2.9133E−05	−0.000530133	−0.001125507	−0.000328221	−0.000567836
ARIJ	0.000186894	0.002667297	0.002017596	5.01539E−05	0.000521658	0.00040383	0.000172183	0.00088278	0.000429215	0.000529795
IMPJ	0.001216867	0.002017596	0.004554488	8.46686E−05	0.000928021	0.000279867	0.000733924	0.001336093	0.000686499	0.000645252
VODJ	0.000173432	5.01539E−05	8.46686E−05	0.000598035	0.000427553	0.000163902	0.000400612	0.000374885	0.000450061	0.00039834
DSYJ	−0.00014238	0.000521658	0.000928021	0.000427553	0.0025653	0.000386927	0.001403621	0.002210785	0.001807441	0.001300584
ITEJ	2.9133E−05	0.00040383	0.000279867	0.000163902	0.000386927	0.003734507	0.000613187	0.000657535	0.000497057	0.000107724
INLJ	−0.000530133	0.000172183	0.000733924	0.000400612	0.001403621	0.000613187	0.00320363	0.00262497	0.001520984	0.001169686
NEDJ	−0.001125507	0.00088278	0.001336093	0.000374885	0.002210785	0.000657535	0.00262497	0.004754836	0.002423374	0.002269596
SLMJ	−0.000328221	0.000429215	0.000686499	0.000450061	0.001807441	0.000497057	0.001520984	0.002423374	0.002216564	0.001412814
BVTJ	−0.000567836	0.000529795	0.000645252	0.00039834	0.001300584	0.000107724	0.001169686	0.002269596	0.001412814	0.002130913

$w_1 = 0.025$, $w_2 = -0.025$, $w_3 = -0.031$, $w_4 = 0.022$, $w_5 = -0.017$, $w_6 = 0.150$, $w_7 = 0.117$, $w_8 = 0.133$, $w_9 = 0.070$, $w_{10} = 0.003$, $w_{11} = 0.061$, $w_{12} = 0.095$, $w_{13} = -0.029$, $w_{14} = 0.436$, $w_{15} = -0.013$, $w_{16} = 0.091$, $w_{17} = -0.012$, $w_{18} = -0.013$, $w_{19} = -0.097$ and $w_{20} = 0.034$. Substituting these values of w_i's in (24) and (26), together with those in Tables 4 and 5, we obtain that $\bar{\mu} = 0.0018$ and $\sigma_p^2 = 0.00144$. This gives the allocation for each stock when investing under a criteria of minimizing risk as given in Table 6, with a portfolio risk of 0.00144 and an expected portfolio return of 0.0018. A negative allocation means the investor is short selling

Table 6. Table of stocks and allocations.

Company	Allocation	Company	Allocation
SHPJ	2.5%	GFIJ	6.1%
MTNJ	−2.5%	ARIJ	9.5%
SOLJ	−3.1%	IMPJ	−2.9%
AMSJ	2.2%	VODJ	43.6%
FSRJ	−1.7%	DSYJ	−1.3%
RNIJ	15%	ITEJ	9.1%
SPPJ	11.7%	INLJ	−1.2%
APNJ	13.3%	NEDJ	−1.3%
ABGJ	7%	SLMJ	−9.7%
NPKJ	0.3%	BVTJ	3.4%

the stock, that is, selling stock that one does not own or that has been acquired on loan from a broker. Notice that because of risk-return trade-off, the strategy of minimizing risk when formulating the portfolio selection problem (27) minimizes expected return as well.

5. Conclusion and future work

In this paper, we proposed a new conjugate gradient method which has a direction that satisfies the sufficient descent property. The method is based on the ideas of the DL and RMIL+ conjugate gradient parameters. Its global convergence was established under the strong Wolfe line search. The method's efficacy was tested using a number of unconstrained optimization problems. Based on the numerical results, it showed to be efficient and robust as compared to other competing methods in the literature. Furthermore, the method's applicability was explored in portfolio selection, where a risk-averse portfolio optimization problem, with m stocks, is solved by transforming it into an unconstrained optimization problem. For future work, the proposed method can be extended to solve portfolio selection problems with more practical constraints, such as, for example, restricting the minimum and maximum proportions of asserts in a portfolio. The method can also be extended to solve other practical problems that arise in motion control, compressive sensing and image deblurring.

Acknowledgements. The authors would like to thank the anonymous referees for their valuable comments and suggestions that improved the quality of this work.

References

[1] A.B. Abubakar, P. Kumam, M. Malik, P. Chaipunya and A. H. Ibrahim, A hybrid FR-DY conjugate gradient algorithm for unconstrained optimization in portfolio selection. *AIMS Math.* **6** (2021) 6506–6527.

[2] A.B. Abubakar, P. Kumam, M. Malik and A.H. Ibrahim, A hybrid conjugate gradient based approach for solving unconstrained optimization and motion control problems. *Math. Comput. Simul.* **201** (2022) 640–657.

[3] A.B. Abubakar, M. Malik, P. Kumam, H. Mohammad, M. Sun, A.H. Ibrahim and A.I. Kiri, A liu-Storey-type conjugate gradient method for unconstrained minimization problem with application in motion control. *J. King Saud Univ. Sci.* **34** (2022) 101923.

[4] N. Andrei, An unconstrained optimization test functions collection. *Adv. Model. Optim.* **10** (2008) 147–161.

[5] A.M. Awwal, I.M. Sulaiman, M. Malik, M. Mamat, P. Kumam and K. Sitthithakerngkiet, A spectral RMIL+ conjugate gradient method for unconstrained optimization with applications in portfolio selection and motion control. *IEEE Access* **9** (2021) 75398–75414.

[6] M.C. Bartholomew-Biggs, Nonlinear Optimization with Financial Applications. Springer Science & Business Media (2006).

[7] M.C. Bartholomew-Biggs and S.J. Kane, A global optimization problem in portfolio selection. *Comput. Manag. Sci.* **6** (2009) 329–345.

[8] Z. Dai, Comments on a new class of nonlinear conjugate gradient coefficient with global convergence properties. *Appl. Math. Comput.* **276** (2016) 297–300.

[9] Y.H. Dai and L.Z. Liao, New conjugacy conditions and related nonlinear conjugate gradient methods. *Appl. Math. Optim.* **43** (2001) 87–101.

[10] Y.H. Dai and Y. Yuan, A nonlinear conjugate gradient method with a strong global convergence property. *SIAM J. Optim.* **10** (1999) 177–182.

[11] J. Deepho, A.B. Abubakur, M. Malik and I.K. Argyros, Solving unconstrained optimization problems via hybrid CD-DY conjugate gradient methods with applications. *J. Comput. Appl. Math.* **405** (2022) 113823.

[12] S. Delladji, M. Belloufi and B. Sellami, New hybrid conjugate gradient method as a convex combination of FR and BA methods. *J. Inf. Optim. Sci.* **42** (2021) 591–602.

[13] S.S. Djordjević, New hybrid conjugate gradient method as a convex combination of LS and FR methods. *Acta Math. Sci.* **39** (2019) 214–228.

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

[15] A.H. Ibrahim, P. Kumam, A. Kamandi and A.B. Abubakar, An efficient hybrid conjugate gradient method for unconstrained optimization. *Optim. Methods Softw.* **37** (2022) 1370–1383.

[16] P. Faramarzi and K.A. Amini, Modified spectral conjugate gradient method with global convergence. *J. Optim. Theory Appl.* **182** (2019) 667–690.

[17] R. Fetcher, Practical Methods of Optimization Vol. 1: Unconstrained Optimization. John Wiley & Sons, New York (1987).

[18] R. Fletcher and C.M Reeves, Function minimization by conjugate gradient. *Comput. J.* **7** (1964) 149–154.

[19] M. Hestenes and E. Steifel, Method of conjugate gradients for solving linear systems. *J. Res. Nat. Bur. Stan. Sect. B* **49** (1952) 409–436.

[20] B. Ivanov, P.S. Stanimirović, B.I. Shaini, H. Ahmad and M. Wang, A novel value for the parameter in the Dai–Liao-type conjugate gradient method. *J. Func. Space* **2021** (2021) 6693401.

[21] J. Jian, L. Yang, X. Jiang, P. Liu and M. Liu, A spectral conjugate gradient method with descent property. *Mathematics* **8** (2020) 280.

[22] J. Jian, P. Liu, X. Jiang and B. He, Two improved nonlinear conjugate gradient methods with the strong Wolfe line search. *Bull. Iran. Math. Soc.* **48** (2022) 2297–2319.

[23] X. Jiang, J. Jian, D. Song and P. Liu, An improved Polak–Ribière–Polyak conjugate gradient method with an efficient restart direction. *Comput. Appl. Math.* **40** (2021) 174.

[24] M. Koorapetse, P. Kaelo, S. Lekoko and T. Diphofu, A derivative-free RMIL conjugate gradient projection method for convex constrained monotone equations with applications in compressive sensing. *Appl. Numer. Math.* **165** (2021) 431–441.

[25] C.X. Kou and Y.H. Dai, A modified self-scaling memoryless Broyden–Fletcher–Goldfarb–Shanno method for unconstrained optimization. *J. Optim. Theory Appl.* **165** (2015) 209–224.

[26] Y. Liu and C. Storey, Efficient generalized conjugate gradient algorithms, Part 1: theory. *J. Optim. Theory Appl.* **69** (1991) 129–137.

[27] J.K. Liu, Y.X. Zhao and X.L. Wu, Some three-term conjugate gradient methods with the new direction structure. *Appl. Numer. Math.* **150** (2019) 433–443.

[28] J. Lu, G. Yuan and Z. Wang, A modified Dai–Liao conjugate gradient method for solving unconstrained optimization and image restoration problems. *J. Appl. Math. Comput.* **68** (2022) 681–703.

[29] H. Markowitz, Portfolio selection. *J. Finan.* **7** (1952) 77–91.

[30] J.J. Morè, B.S. Garbow and K.E. Hillstrom, Testing unconstrained optimization software. *ACM Trans. Math. Softw. (TOMS)* **7** (1981) 17–41.

[31] P. Mtagulwa and P. Kaelo, An efficient mixed conjugate gradient method for solving unconstrained optimisation problems. *East Asian J. Appl. Math.* **11** (2021) 421–434.

[32] E. Polak and G. Ribiére, Note sur la covergence de directions conjugées. *Rev. Fren. Inf. Rech. Oper. 3e Année* **16** (1969) 35–43.

[33] T. Polyak, The conjugate gradient method in extreme problems. *USSR Comp. Math. Math. Phys.* **9** (1969) 94–112.

[34] M. Rivaie, M. Mustafa, L.W. June and I. Mohd, A new class of nonlinear conjugate gradient coefficient with global convergence properties. *Appl. Math. Comput.* **218** (2012) 11323–11332.

[35] S. Roman, Introduction to the Mathematics of Finance: from Risk Management to Options Pricing. Springer Science & Business Media (2004).

[36] Q. Tian, X. Wang, L. Pang, M. Zhang and F. Meng, A new hybrid three-term conjugate gradient algorithm for large-scale unconstrained problems. *Mathematics* **9** (2021) 1353.

[37] Y. Wu, A modified three-term PRP conjugate gradient algorithm for optimization models. *J. Inequl. Appl.* **2017** (2017) 97.

[38] S. Yao, Q. Feng, L. Li and J. Xu, A class of globally convergent three-term Dai–Liao conjugate gradient methods. *Appl. Numer. Math.* **151** (2020) 354–366.

[39] J. Yin, L. Wang and X. Jiang, A modified PRP conjugate gradient method with Armijo line search for large-scale unconstrained optimization, in 36th ChiCC. Vol. 36. IEEE Xplore (2017) 2568–2571.

[40] J. Yin, J. Jian, X. Jiang, M. Liu and L. Wang, A hybrid three-term conjugate gradient projection method for constrained nonlinear monotone equations with applications. *Numer. Algorithm* **4** (2021) 1–30.

[41] T. Yuan, D. Bi, L. Pan, Y. Xie and J. Lyu, A compressive hybrid conjugate gradient image recovery approach for radial MRI. *Imaging Sci. J.* **66** (2018) 278–288.

[42] G. Yuan, T. Li and W. Hu, A conjugate gradient algorithm for large-scale nonlinear equations and image restoration problems. *Appl. Numer. Math.* **147** (2020) 129–141.

[43] Y. Zheng, A new family of Dai–Liao conjugate gradient methods for unconstrained optimization. *RAIRO: Oper. Res.* **55** (2021) 3281–3291.

[44] G. Zoutendijk, Nonlinear programming, computational methods, in Integer and Nonlinear Programming, edited by J. Abadie. North-Holland, Amsterdam (1970) 37–86.

Please help to maintain this journal in open access!

This journal is currently published in open access under the Subscribe to Open model (S2O). We are thankful to our subscribers and supporters for making it possible to publish this journal in open access in the current year, free of charge for authors and readers.

Check with your library that it subscribes to the journal, or consider making a personal donation to the S2O programme by contacting subscribers@edpsciences.org.

More information, including a list of supporters and financial transparency reports, is available at https://edpsciences.org/en/subscribe-to-open-s2o.