

Application of a globally convergent hybrid conjugate gradient method in portfolio optimization

P. MTAGULWA, P. KAELO, T. DIPHOFU AND K. KAISARA

Abstract

In this paper, we propose a modification that improves efficiency, robustness and reliability of the famous HS conjugate gradient method. In particular, we propose a hybrid of the HS and DHS methods, where DHS is another recent modification of the HS method. Irrespective of the line search, the search direction of the proposed method is sufficiently descent. Moreover, the new approach guarantees global convergence for general functions under the strong Wolfe line search. Numerical results and performance profiles are reported, and indicate that the new approach outperforms three similar methods in the literature. We also give a practical application of the new approach in minimizing risk in portfolio selection.

Mathematics Subject Classification 2000: 90C06, 90C30, 65K05

Keywords: Conjugate gradient, Global convergence, Strong Wolfe-Powell conditions, Portfolio selection.

1. INTRODUCTION

Several methods in the literature have been proposed to solve the unconstrained optimization problem

$$\min f(x), \quad (1)$$

where $x \in \mathbb{R}^n$ and $f: \mathbb{R}^n \rightarrow \mathbb{R}$ is a continuously differentiable function that is bounded from below. However, nonlinear conjugate gradient (CG) methods are considered to be the most efficient in solving large-scale unconstrained optimization problems because of their low memory requirements and simple iterations. In solving problem (1), CG methods begin from an initial guess, $x_0 \in \mathbb{R}^n$, and generate a sequence $\{x_k\}_{k=0}^{\infty}$ by using the relation

$$x_{k+1} = x_k + \alpha_k d_k, \quad (2)$$

where $\alpha_k > 0$ is known as the step size and d_k is the search direction formulated by

$$d_k = \begin{cases} -g_k, & \text{if } k = 0, \\ -g_k + \beta_k d_{k-1}, & \text{if } k > 0. \end{cases} \quad (3)$$

The scalar β_k , which differentiates CG methods, is called the conjugate gradient update parameter (coefficient), and $g_k = \nabla f(x_k)$ denotes the gradient of f at x_k .

In finding the step size α_k , inexpensive line searches such as the weak Wolfe line search conditions

$$\begin{cases} f(x_k + \alpha_k d_k) \leq f(x_k) + \delta \alpha_k g_k^T d_k \\ g_{k+1}^T d_k \geq \sigma g_k^T d_k, \end{cases}$$

the strong Wolfe line search conditions

$$\begin{cases} f(x_k + \alpha_k d_k) \leq f(x_k) + \delta \alpha_k g_k^T d_k \\ |g_{k+1}^T d_k| \leq \sigma |g_k^T d_k|, \end{cases} \quad (4)$$

or the generalised Wolfe line search conditions

$$\begin{cases} f(x_k + \alpha_k d_k) \leq f(x_k) + \delta \alpha_k g_k^T d_k \\ \sigma g_k^T d_k \leq g_{k+1}^T d_k \leq -\sigma_1 g_k^T d_k, \end{cases}$$

are often used, where $0 < \delta < \sigma < 1$ and $\sigma_1 \geq 0$. The most prominent choices for updating the CG parameter include the Fletcher-Reeves (β_k^{FR}) [16], Dai and Yuan (β_k^{DY}) [8], Conjugate Descent (β_k^{CD}) [15], Hestenes and Stiefel (β_k^{HS}) [19], Polak-Ribière-Polyak (β_k^{PRP}) [26; 27] and Liu and Storey (β_k^{LS}) [22]. Although the FR, DY and CD methods have strong convergence properties, they tend to have poor numerical performance. This is because they are prone to taking short steps which leads to jamming. On the contrary, the HS, PRP and LS methods do not always guarantee convergence. However, these methods have good numerical performance because of their ability to switch to the steepest descent direction in order to avoid short steps which result in jamming [18].

In order to improve on convergence and efficiency of conjugate gradient methods, many researchers have proposed modifications of the original schemes, such as hybrid methods (in which two or more CG coefficients are used interchangeably or as a linear combination in one algorithm), three-term methods and spectral methods [1; 5; 12; 14; 20; 28]. For example, Jiang et al. [20] modified the denominator of the PRP method and proposed the JJSL method as

$$\beta_k^{JJSL} = \frac{g_k^T (g_k - g_{k-1})}{g_{k-1}^T (g_{k-1} - g_k)} = \frac{\|g_k\|^2 - g_k^T g_{k-1}}{\|g_{k-1}\|^2 - g_k^T g_{k-1}},$$

where $\|\cdot\|$ denotes the Euclidean norm. To increase the numerical efficiency of the

JJSL method, they introduced a new restart direction given by

$$d_k = -g_k + \xi \frac{g_k^T g_{k-1}}{\|g_{k-1}\|^2} g_{k-1}$$

and constructed their search direction as follows

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

where $0 < \xi < 1$. Under the strong Wolfe conditions, the global convergence of the JJSL method was established and numerical experiments showed that it outperforms the PRP method. In 2022, Sulaiman et al. [28] constructed a three-term search direction

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

where

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

and

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

with $y_{k-1} = g_k - g_{k-1}$. The authors established global convergence under the weak Wolfe line search conditions and also showed that the method is efficient in solving unconstrained optimization problems.

In Fang et al. [14], a hybrid method that uses four different conjugate gradient coefficients interchangeably is introduced by proposing the update

$$\beta_k^{FW} = \frac{\|g_k\|^2 - \max\{0, (\|g_k\|/\|g_{k-1}\|)g_k^T g_{k-1}\}}{\max\{\mu|g_k^T d_{k-1}|, d_{k-1}^T(g_k - g_{k-1})\}},$$

where $\mu > 1$. This method satisfies the sufficient descent condition

$$d_k^T g_k \leq -c\|g_k\|^2, c > 0,$$

irrespective of the line search and its global convergence was attained with the step size yielded by the Wolfe line search. Numerical experiments indicate that the method is efficient and robust in solving a collection of unconstrained optimization problems.

More recently, Mustafa [25] introduced a new spectral LS conjugate gradient method by applying the Dai and Liao conjugacy condition ($d_k^T y_{k-1} = -t g_k^T s_{k-1}, t > 0$) [9] and consequently proposing the spectral search direction

$$d_k = - \left(\frac{t g_k^T s_{k-1}}{g_k^T y_{k-1}} + \frac{d_{k-1}^T y_{k-1}}{-d_{k-1}^T g_{k-1}} \right) g_k + \frac{g_k^T y_{k-1}}{-d_{k-1}^T g_{k-1}} d_{k-1},$$

where $s_{k-1} = x_k - x_{k-1}$. The author proved that this spectral method generates sufficient descent directions and globally converges with the strong Wolfe line search. In addition, the method performed better than the LS and CD methods in numerical experiments.

In this paper, we propose a hybrid method of the HS method and its modification (DHS method). This new hybrid is such that it possesses not only the sufficient descent property independently of any line search, but also converges globally with the strong Wolfe line search. In the next section, we present the new approach and its motivation.

2. ANOTHER HYBRID CONJUGATE GRADIENT METHOD

Over the years, many variants of the famous PRP method have been proposed in the literature, see for example [2; 11; 17; 20; 21; 24]. In 2006, Wei et al. [29] modified the PRP method by introducing a conjugate gradient coefficient which is formulated by

$$\beta_k^{WYL} = \frac{g_k^T (g_k - \frac{\|g_k\|}{\|g_{k-1}\|} g_{k-1})}{\|g_{k-1}\|^2}. \quad (5)$$

The authors proved that β_k^{WYL} is always positive and consequently has good convergence properties under the weak Wolfe line search. Motivated by their work, Zhang [30] and Dai and Wen [10] improved the WYL method by respectively proposing the NPRP and DHS methods in which the conjugate gradient coefficients are computed as

$$\beta_k^{NPRP} = \frac{\|g_k\|^2 - \frac{\|g_k\|}{\|g_{k-1}\|} |g_k^T g_{k-1}|}{\|g_{k-1}\|^2} \quad (6)$$

and

$$\beta_k^{DHS} = \frac{\|g_k\|^2 - \frac{\|g_k\|}{\|g_{k-1}\|} |g_k^T g_{k-1}|}{m |g_k^T d_{k-1}| + d_{k-1}^T y_{k-1}}, \quad (7)$$

where $m > 1$. Both modifications were shown to possess better convergence properties with a sufficient descent property and good numerical performance. Following the

work of Zhang [30], Alhawarat et al. [3] suggested the update

$$\beta_k^{PRP^{**}} = \begin{cases} \beta_k^{PRP}, & \text{if } \|g_k\|^2 > |g_k^T g_{k-1}|, \\ \beta_k^{NPRP}, & \text{otherwise.} \end{cases}$$

The authors showed that the $\beta_k^{PRP^{**}}$ method also satisfies the sufficient descent property and is globally convergent under the strong Wolfe line search. Moreover, a numerical analysis showed that $\beta_k^{PRP^{**}}$ outperforms the WYL and NPRP methods.

Motivated by the work of Alhawarat et al. [3] and that of Dai and Wen [10], we propose the update

$$\beta_k^{MHS} = \begin{cases} \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}}, & \text{if } \|g_k\|^2 \geq |g_k^T g_{k-1}|, \\ \frac{\|g_k\|^2 - \frac{\|g_k\| \|g_{k-1}\|}{\|g_k\|^2} |g_k^T g_{k-1}|}{m |g_k^T d_{k-1}| + d_{k-1}^T y_{k-1}}, & \text{otherwise,} \end{cases} \quad (8)$$

with $m > 1$ and a search direction d_k defined as

$$d_k = \begin{cases} -g_k, & \text{if } k = 0 \text{ or } |g_k^T g_{k-1}| \geq 0.5 \|g_k\|^2, \\ -\left(1 + \beta_k^{MHS} \frac{g_k^T d_{k-1}}{\|g_k\|^2}\right) g_k + \beta_k^{MHS} d_{k-1}, & \text{if } k > 0. \end{cases} \quad (9)$$

Now, with β_k and d_k defined as in (8) and (9) respectively, we present our new algorithm below.

Algorithm 1 A globally convergent HS-DHS hybrid conjugate gradient method (MHS Method)

- 1: Assign $x_0 \in \mathbb{R}^n$ and parameters $\delta \in (0, 1)$, $\sigma \in (\delta, 0.9]$, $\varepsilon > 0$ and $m > 1$. Set $d_0 = -g_0$.
 - 2: **for** $k = 0, 1, \dots$ **do**
 - 3: If $\|g_k\| \leq \varepsilon$, then stop.
 - 4: Generate d_k by (9), with β_k obtained by (8).
 - 5: Compute α_k using the strong Wolfe conditions (4).
 - 6: Set $x_{k+1} = x_k + \alpha_k d_k$.
 - 7: Set $k = k + 1$ and go to Step 3.
 - 8: **end for**
-

3. GLOBAL CONVERGENCE OF THE NEW APPROACH

We begin by introducing two standard assumptions which have been used to analyze the global convergence properties of CG methods in the literature.

ASSUMPTION 3.1. The level set

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

is bounded, where x_0 is the initial guess of the iterative relation (2).

ASSUMPTION 3.2. In some neighbourhood N of Ω , the objective function f is continuously differentiable and its gradient is Lipschitz continuous, which means there exists a constant $L > 0$ such that

$$\|g(x) - g(y)\| \leq L\|x - y\| \text{ for all } x, y \in N.$$

If **Assumptions** 3.1 and 3.2 hold, then there exists a positive constant ς such that

$$\|g(x)\| \leq \varsigma \text{ for all } x \in N. \quad (10)$$

Next, we present a lemma that ensures that our algorithm is monotone, that is to say, for any iterate x_k generated by the method, we have $f(x_{k+1}) < f(x_k)$.

LEMMA 3.1. Let the sequences $\{g_k\}$ and $\{d_k\}$ be generated by **Algorithm** 1. Then the sufficient descent condition

$$d_k^T g_k = -\|g_k\|^2, \forall k \geq 0, \quad (11)$$

holds.

PROOF.

For $k = 0$ or $|g_k^T g_{k-1}| \geq 0.5\|g_k\|^2$, we have $d_k = -g_k$ and consequently

$$g_k^T d_k = -\|g_k\|^2.$$

Otherwise, the search direction d_k is defined by

$$d_k = -\left(1 + \beta_k^{MHS} \frac{g_k^T d_{k-1}}{\|g_k\|^2}\right) g_k + \beta_k^{MHS} d_{k-1}. \quad (12)$$

Pre-multiplying (12) by g_k^T yields

$$\begin{aligned} g_k^T d_k &= -\|g_k\|^2 \left(1 + \beta_k^{MHS} \frac{g_k^T d_{k-1}}{\|g_k\|^2}\right) + \beta_k^{MHS} g_k^T d_{k-1} \\ &= -\|g_k\|^2 - \beta_k^{MHS} g_k^T d_{k-1} + \beta_k^{MHS} g_k^T d_{k-1} \\ &= -\|g_k\|^2. \end{aligned}$$

Therefore, (11) is always satisfied and thus concludes the proof. $\square$

LEMMA 3.2. [31] Suppose that **Assumptions** 3.1 and 3.2 hold. Let the sequence $\{x_k\}$ be generated by (2), with a descent search direction d_k . If the step size α_k is computed using the strong Wolfe conditions, then the Zoutendijk condition

$$\sum_{k \geq 0} \frac{(g_k^T d_k)^2}{\|d_k\|^2} < +\infty$$

holds.

By applying the descent condition (11) we can also write the Zoutendijk condition as

$$\sum_{k \geq 0} \frac{\|g_k\|^4}{\|d_k\|^2} < +\infty. \quad (13)$$

Next we present a lemma which shows that the new approach, that is β_k^{MHS} , is always positive and bounded.

LEMMA 3.3. For any $k \geq 0$, the relation

$$0 < \beta_k^{MHS} \leq 2\beta_k^{DY} = \frac{2\|g_k\|^2}{d_{k-1}^T y_{k-1}}$$

always holds.

PROOF.

First we begin by showing that the inner product $d_{k-1}^T(g_k - g_{k-1})$ is always positive, by considering the inequality

$$g_k^T d_{k-1} \geq \sigma g_{k-1}^T d_{k-1},$$

which is a constituent of the strong Wolfe conditions (4). Subtracting $g_{k-1}^T d_{k-1}$ from both sides of the above inequality yields

$$d_{k-1}^T y_{k-1} = d_{k-1}^T (g_k - g_{k-1}) \geq (\sigma - 1) g_{k-1}^T d_{k-1}. \quad (14)$$

Since $\sigma < 1$ and our search direction satisfies the sufficient descent condition (11), it follows that

$$d_{k-1}^T y_{k-1} > 0.$$

Let us now consider the following two cases.

Case 1. If $\beta_k^{MHS} = \frac{\|g_k\|^2 - g_k^T g_{k-1}}{d_{k-1}^T y_{k-1}}$, then under the condition $\|g_k\|^2 \geq |g_k^T g_{k-1}|$, we get

$$\begin{aligned} 0 < \beta_k^{MHS} &\leq \frac{\|g_k\|^2 + |g_k^T g_{k-1}|}{d_{k-1}^T y_{k-1}} \\ &\leq \frac{2\|g_k\|^2}{d_{k-1}^T y_{k-1}}. \end{aligned}$$

Case 2. If $\beta_k^{MHS} = \frac{\|g_k\|^2 - \frac{\|g_k\|}{\|g_{k-1}\|} |g_k^T g_{k-1}|}{m|g_k^T d_{k-1}| + d_{k-1}^T y_{k-1}}$, then by Cauchy-Schwarz inequality we obtain

$$0 \leq \frac{\|g_k\|^2 - \frac{\|g_k\|}{\|g_{k-1}\|} |g_k^T g_{k-1}|}{m|g_k^T d_{k-1}| + d_{k-1}^T y_{k-1}} \leq \beta_k^{MHS} \leq \frac{\|g_k\|^2}{d_{k-1}^T y_{k-1}}.$$

Thus,

$$0 \leq \beta_k^{MHS} \leq \frac{2\|g_k\|^2}{d_{k-1}^T y_{k-1}}.$$

Hence, the lemma is proved. $\square$

THEOREM 3.4. If **Assumptions** 3.1 and 3.2 hold and the sequences $\{g_k\}$ and $\{d_k\}$ are generated by MHS method, then

$$\liminf_{k \rightarrow \infty} \|g_k\| = 0. \quad (15)$$

PROOF.

Assume that (15) does not hold. Then there exists a constant $r > 0$ such that

$$\|g_k\| \geq r, \quad (16)$$

for all $k \geq 0$. By Lipschitz continuity of the gradient, it follows that

$$\|y_{k-1}\| = \|g_k - g_{k-1}\| \leq L\|x_k - x_{k-1}\| \leq LD,$$

where $D = \max\{\|x - y\| : x, y \in \Omega\}$ is the diameter of the level set Ω . Since $\sigma \in (\delta, 0.9]$ and

$$\beta_k^{MHS} \leq \frac{2\|g_k\|^2}{d_{k-1}^T (g_k - g_{k-1})},$$

then from (10), (11) and (14), we get

$$\begin{aligned} \beta_k^{MHS} &\leq \frac{2\|g_k\|^2}{(\sigma - 1)(g_{k-1}^T d_{k-1})} \\ &= \frac{2}{1 - \sigma} \frac{\|g_k\|^2}{\|g_{k-1}\|^2} \end{aligned}$$

$$\begin{aligned}
&\leq \frac{20\varsigma^2}{r^2} \\
&= M.
\end{aligned} \tag{17}$$

Next, we show that if g_k satisfies (16), then there exists a constant $\varpi > 0$ such that

$$\alpha_k \geq \varpi > 0, \forall k \geq 0.$$

Indeed, by assigning

$$\vartheta_k \triangleq f(x_k + \alpha d_k) - f(x_k) - \delta \alpha g_k^T d_k = (1 - \delta) \alpha g_k^T d_k + o(\alpha),$$

and from the strong Wolfe conditions, we have that for sufficiently small $\alpha > 0$, the terms ϑ_k are non positive since $g_k^T d_k = -\|g_k\|^2 < 0$ according to the sufficient descent condition (11). Thus, the first inequality of the strong Wolfe conditions holds. Now, let

$$\bar{\alpha}_k = \sup\{\bar{\alpha} | f(x_k + \alpha d_k) \leq f(x_k) + \delta \alpha g_k^T d_k, \forall \alpha \in (0, \bar{\alpha}]\}. \tag{18}$$

Undoubtedly, $\bar{\alpha}_k > 0$ and

$$\bar{\bar{\alpha}} = \inf_k \bar{\alpha}_k > 0$$

by (16). From the mean value theorem, we get

$$f(x_k + \bar{\alpha}_k d_k) = f(x_k) + \bar{\alpha}_k g(x_k + \rho_k d_k)^T d_k, \rho_k \in (0, \bar{\alpha}_k).$$

Consequently, there exists a constant δ_1 , $0 < \delta < \delta_1 < \sigma < 1$, for which the inequality

$$\sigma g_k^T d_k \leq g(x_k + \rho_k d_k)^T d_k = \delta_1 g_k^T d_k \leq \delta g_k^T d_k.$$

From the strong Wolfe conditions, assume we have

$$g(x_k + \alpha d_k)^T d_k < \sigma g_k^T d_k,$$

and set

$$\underline{\alpha}_k = \sup\{\underline{\alpha} | g(x_k + \alpha d_k)^T d_k < \sigma g_k^T d_k, \forall \alpha \in (0, \underline{\alpha}]\}. \tag{19}$$

Clearly $\underline{\alpha}_k > 0$, and by (16), we obtain

$$\underline{\underline{\alpha}} = \inf_k \underline{\alpha}_k > 0.$$

Using (18) and (19), and the fact that $0 < \delta < \sigma < 1$, we obtain that

$$\underline{\alpha}_k \leq \rho_k \leq \bar{\alpha}_k \text{ for all } k \geq 0.$$

Therefore, there exists a constant $\varpi > 0$ such that

$$\bar{\alpha} \geq \alpha_k \geq \varpi \geq \underline{\alpha} > 0 \text{ for all } k \geq 0.$$

From (9) we have

$$d_k = - \left(1 + \beta_k^{MHS} \frac{g_k^T d_{k-1}}{\|g_k\|^2} \right) g_k + \beta_k^{MHS} d_{k-1},$$

and hence,

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

Applying Cauchy-Schwarz inequality and (17) we get

$$\begin{aligned} \|d_k\| &\leq \|g_k\| + M \|g_k\| \frac{|g_k^T d_{k-1}|}{\|g_k\|^2} + M \|d_{k-1}\| \\ &\leq \|g_k\| + 2M \|d_{k-1}\| \\ &\leq \varsigma + 2M \left(\frac{x_k - x_{k-1}}{\alpha_k} \right) \\ &\leq \varsigma + 2M \left(\frac{D}{\varpi} \right) \\ &= E, \end{aligned}$$

which implies that

$$\sum_{k \geq 0} \frac{1}{\|d_k\|^2} = +\infty. \quad (20)$$

On the other hand, from Zoutendijk condition (13), and (16), we have

$$r^4 \sum_{k \geq 0} \frac{1}{\|d_k\|^2} \leq \sum_{k \geq 0} \frac{\|g_k\|^4}{\|d_k\|^2} < +\infty,$$

which contradicts (20). Hence claim (15) is true and the proof is complete. $\square$

4. NUMERICAL EXPERIMENTS

In this section, we test the efficiency and robustness of our proposed MHS method, by comparing its numerical performance to the methods proposed by Dai and Wen [10], Alhawarat et al. [3] and Wei et al. [29], herein denoted as DHS, PRP** and NPRP respectively, on a set of 52 unconstrained test problems selected from [4]. Iterations are terminated when $\|g_k\| \leq 10^{-5}$ and moreover, if a method does not reach a solution within 10000 iterations we consider it to have failed to solve the problem. All algorithms are coded in MATLAB R2019a and the four methods were implemented with the strong Wolfe line search conditions with parameters $\delta = 0.01$ and $\sigma = 0.1$. For MHS and DHS, the parameter m is set to $m = 1.5$.

Numerical results are reported in Table 1, where “-” means that the method failed to solve the problem within 10000 iterations. Number of iterations, function evaluations and gradient evaluations are denoted as NI, FE and GE, respectively. From the table, we observe that the MHS method successfully solved all the 52 problems, whereas PRP** failed to solve one problem, NPRP method failed to solve two problems and DHS method failed to solve three problems. For each problem, we bolded the minimum iterations and function/gradient evaluations so that the best performer can easily be located. Numerical results in Table 1 indicate that the MHS method is competitive as there is a considerable number of problems for which it gives lower number of iterations and gradient/function evaluations.

In addition, for a clear picture of the performance difference between MHS, PRP**, NPRP and DHS methods, we implement a performance profile tool suggested by Dolan and Moré [13]. In Figures 1, 2 and 3, we compare the performance of the MHS, PRP**, NPRP and DHS methods based on number of iterations, gradient evaluations and function evaluations, respectively. In these figures, the top curve represents the method that solved the most problems within a factor $\tau > 0$. All three figures indicate that the MHS method outperforms the DHS, PRP** and NPRP methods with respect to number of iterations, gradient/function evaluations. Therefore, the figures indicate that our method is efficient and robust in solving the given set of unconstrained optimization problems.

Table I. Numerical results of the methods

Function	Dim	MHS			PRP**			NRP			DHS		
		NI	FE	GE	NI	FE	GE	NI	FE	GE	NI	FE	GE
Ext. Freudenstein & Roth Cube	100	8	39	14	8	39	13	10	51	18	10	51	18
	10000	1360	7050	2100	-	-	-	1942	9724	3004	1992	9852	2952
Ext. Rosenbrock	20000	33	168	74	30	166	73	41	213	88	26	138	64
Ext. White & Holst	1000	31	165	68	28	137	53	39	209	90	34	193	89
Ext. Beale	50000	17	63	40	15	67	39	14	43	25	14	43	26
Ext. Penalty	500	11	50	15	10	47	14	12	53	16	12	53	16
Perturbed Quad	7000	470	2821	922	470	2821	922	837	5016	1556	1160	6947	2165
Raydan2	10	3	5	5	3	5	5	3	5	5	3	5	5
Diagonal 2	5000	324	1278	1275	320	1331	1329	413	1832	1829	319	1385	1383
Hager	200	25	61	33	25	61	33	26	63	34	26	63	34
Gen. Tridiagonal 1	7000	20	60	24	20	60	24	19	57	24	19	57	24
Ext. Tridiagonal 1	20000	7	43	37	11	45	37	6	29	23	7	43	37
Ext TET	2000	6	16	10	6	16	10	6	16	10	6	16	10
Gen. Tridiagonal 2	1000	39	128	46	38	125	45	41	149	60	42	145	56
Diagonal 4	20000	6	13	7	6	14	8	6	14	7	6	14	7
Diagonal 5	5000	2	5	5	2	5	5	2	5	5	2	5	5
Ext. Himmelblau	2000	8	32	13	9	35	14	8	32	13	8	32	13
Gen. White & Holst	100	1965	11074	3027	1813	9948	2532	2495	13535	3541	2476	13301	3441
Gen. PSC1	1000	11	62	57	11	76	71	12	74	69	15	119	112
Ext. PSC1	1000	8	26	14	7	25	14	8	26	14	8	26	14
Ext. Powell	2000	75	306	179	184	611	299	61	245	142	62	250	145
FHess1	50	334	1088	539	324	1033	496	324	1014	540	546	1683	868
FHess2	500	1333	8716	1793	2389	16677	3369	1208	7449	1609	1084	6662	1431
Ext. BD1	50000	9	35	22	10	37	23	10	36	22	10	36	22
Ext. Maratos	100	12	53	18	11	48	17	13	54	19	11	48	17
Perturbed Quad Diagonal	100000	354	1419	397	475	2278	592	412	1667	463	371	1527	430
Ext. Wood	50000	47	220	82	50	239	77	45	217	84	59	274	99
Quadratic QF1	500	122	609	361	122	609	361	209	929	434	184	831	388
Ext. QP2	5000	48	304	125	28	178	77	54	341	146	56	351	148
Quadratic QF2	500	215	1084	222	215	1084	222	236	1190	251	205	1033	241
ENGVAL1	1000	21	52	28	20	56	32	20	53	30	20	53	30
Ext. Tridiagonal 2	1000	25	52	27	26	54	28	27	56	30	26	54	29
FLETCHCR	1000	2715	16381	4759	2980	18331	5398	-	-	-	-	-	-
BDQRTIC	50	84	364	121	83	367	127	79	323	108	73	312	108
Tridia	500	476	2820	915	577	3361	1052	662	3638	926	629	3450	864
NONDIA	20000	7	55	8	7	55	8	7	55	8	7	55	8
DQDRTIC	5000	12	48	15	17	69	25	12	47	14	10	41	14
NONDQUAR	1000	957	2800	2612	1269	3370	3167	1262	4270	3918	1488	4980	4533
Almost Perturbed Quad	20000	807	4845	810	807	4845	810	2556	15459	2643	3031	18337	3154
Perturbed Trid Quad	10000	530	3181	531	530	3181	531	1516	9094	1558	1556	9332	1596
LIARWHD	500	15	81	23	18	96	32	17	92	27	17	90	26
NONSCOMP	20000	41	151	62	46	166	65	40	143	58	41	153	63
QUARTC	7000	3	22	20	3	22	20	3	22	20	3	22	20
SINQUAD	800	152	842	350	120	630	267	160	856	365	135	717	305
Ext. Denschnb	50000	6	17	10	6	17	10	6	17	10	6	17	10
Dixon3dq	5000	2500	5012	2513	2500	5012	2513	-	-	-	-	-	-
Cosine	5000	8	27	19	8	28	20	8	27	19	8	27	19
Biggsb1	500	500	1007	508	500	1007	508	4348	8677	6368	3606	7196	5271
Gen. Quartic	50000	6	18	11	8	20	11	6	18	11	6	18	11
FHess3	10000	2	13	3	2	13	3	2	13	3	2	13	3
Ext. Denschnf	50000	13	68	30	15	81	35	13	69	30	13	69	30
Ext. Quad Penalty QP1	500	6	25	15	6	25	15	6	24	14	-	-	-

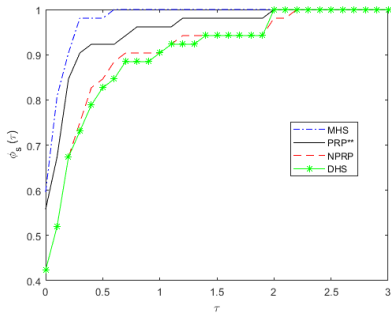


Fig. 1. Performance profile of number of iterations

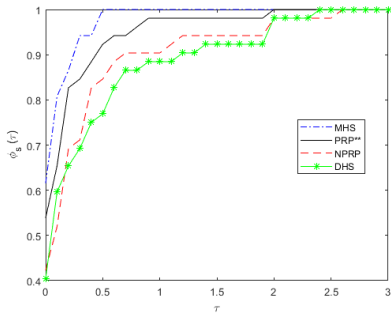


Fig. 2. Performance profile of gradient evaluations

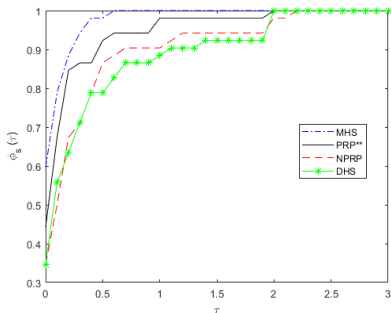


Fig. 3. Performance profile of function evaluations

4.1. Application in Portfolio Selection

In this subsection, we use our newly proposed conjugate gradient method to solve a portfolio selection problem. The theory of portfolio selection was initially discussed by Markowitz [23]. A portfolio is a collection of n stocks or assets $A_1, A_2, \dots, A_n$, that an investor has invested in. Consider a stock A_i , $1 \leq i \leq n$, the rate of return R_i of stock A_i at time t is calculated as

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

where P_t and P_{t-1} are the prices of stock A_i at times t and $t-1$, respectively. The rate of return of the portfolio is calculated as the weighted sum of the returns of each stock, given as

$$R = \sum_{i=1}^n w_i R_i,$$

where w_i is the weight percentage of stock A_i in the portfolio and $\sum_{i=1}^n w_i = 1$. The expected return of the portfolio is given by

$$\mu = E \left(\sum_{i=1}^n w_i R_i \right) = \sum_{i=1}^n w_i \mu_i, \quad (21)$$

where μ_i is the expected return of asset A_i defined by

$$\mu_i = E(R_i).$$

The mean return of a stock is given by

$$\tilde{r}_i = \frac{1}{T} \sum_{t=1}^T R_{it}$$

and the variance of the return of stock A_i , which measures how far the asset price has moved from the mean, is given by

$$\sigma_i^2 = \text{Var}(R_i) = \frac{1}{T-1} \sum_{t=1}^T (R_{it} - \tilde{r}_i)^2.$$

The co-variance of returns of stocks A_i and A_j is calculated as

$$\text{Cov}(R_i, R_j) = \frac{1}{T-1} \sum_{t=1}^T (R_{it} - \tilde{r}_i)(R_{jt} - \tilde{r}_j), \quad i \neq j.$$

Co-variance measures correlation between rate of returns of stock prices of two assets, a positive variance means a positive correlation and a negative variance means there is negative correlation between prices of the two stocks.

Here we consider a portfolio of 10 stocks listed in the S&P 500 stock index. The companies include Apple Inc. (APPL), Microsoft Corporation (MSFT), Amazon.com Inc. (AMZN), Alphabet Inc. (GOOG), Meta Platforms Inc. (META), Johnson & Johnson (JNJ), Berkshire Hathaway Inc. (BRKA.VI), JP Morgan Chase & Co. (JPM), The Procter & Gamble Company (PG) and General Motors (GM). We used the closing stock prices ranging from 26 September 2021 to 26 September 2023 taken from <https://finance.yahoo.com>.

Table II shows the mean and variances of the returns of the stocks and Table III shows the variances and co-variances of the returns of the stocks.

Table II. Table of mean and variances ($\times 10^{-4}$)

	APPL	MSFT	AMZN	GOOG	META	JNJ	BRKA	JPM	PG	GM
mean	-4.6612	-7.0640	-2.2980	-9.2840	-25.9965	1.8781	-12.0634	-11.4596	-3.6807	4.7360
variance	3.0742	3.5099	5.8564	4.6995	11.6633	1.0838	1.2633	2.3182	0.9966	4.8154

Table III. Table of variance and co-variances ($\times 10^{-4}$)

	APPL	MSFT	AMZN	GOOG	META	JNJ	BRKA	JPM	PG	GM
APPL	3.0742	2.2490	2.1133	2.4473	3.0133	0.4270	1.1028	1.0390	0.7497	1.7873
MSFT	2.2490	3.5099	2.9942	2.9835	3.4248	0.3511	0.9771	0.9002	0.7498	1.4894
AMZN	2.1133	2.9942	5.8564	3.4461	4.2973	0.2860	1.0305	1.0243	0.6586	2.0264
GOOG	2.4473	2.9835	3.4461	4.6995	4.1597	0.3844	1.1185	0.8281	0.6118	1.5545
META	3.0133	3.4248	4.2973	4.1597	11.6633	0.4227	1.1587	1.0980	0.7308	2.0572
JNJ	0.4270	0.3511	0.2860	0.3844	0.4227	1.0838	0.4409	0.3805	0.4359	0.4277
BRKA	1.1028	0.9771	1.0305	1.1185	1.1587	0.4409	1.2633	1.1590	0.5121	1.3645
JPM	1.0390	0.9002	1.0243	0.8281	1.0980	0.3805	1.1590	2.3182	0.5063	1.7551
PG	0.7497	0.7498	0.6586	0.6118	0.7308	0.4359	0.5121	0.5063	0.9966	0.4701
GM	1.7873	1.4894	2.0264	1.5545	2.0572	0.4277	1.3645	1.7551	0.4701	4.8154

We formulate our problem in order to minimize risk when investing in our portfolio. Note that other investing strategies may focus on maximizing profits or maximizing profits while incurring some pre-specified level of risk [6; 7]. The risk of a portfolio is also known as the variance of the portfolio's returns, and is calculated as

$$\sigma_p^2 = \text{Var} \left(\sum_{i=1}^n w_i R_i \right) = \sum_{i=1}^n \sum_{j=1}^n w_i w_j \text{Cov}(R_i, R_j). \quad (22)$$

Our objective is to determine the optimal investment strategy (weight proportion of each stock in the portfolio) by solving the minimization problem

$$\min \sigma_p^2 = \text{Var} \left(\sum_{i=1}^n w_i R_i \right) \quad (23)$$

subject to

$$\sum_{i=1}^n w_i = 1. \quad (24)$$

Let $\{w_1, w_2, \dots, w_{10}\}$ be the weight proportions allocated to APPL, MSFT, AMZN, GOOG, META, JNJ, BRKA, JPM, PG and GM, respectively. By setting $w_{10} = 1 - \sum_{i=1}^9 w_i$ and using Table III, we can transform (23) into an unconstrained optimization problem

$$\min_{w \in \mathbb{R}^9} X^T V X, \quad (25)$$

where $X^T = [w_1, w_2, \dots, w_9]$ and V is the variance and co-variance symmetric matrix taken to be the data in Table III. Algorithms MHS, PRP**, NPRP and DHS were all used to solve (25) using 30 different initial points $s_1 = [0.01, 0.01, \dots, 0.01]^T$, $s_2 = [0.1, 0.1, \dots, 0.1]^T$, $s_3 = [0.05, 0.05, \dots, 0.05]^T$ and s_p , $p = 4, 5, \dots, 30$, all of which were randomly generated between $(0, 1)$ such that (24) holds. All methods succeeded in finding the solution $w_*^1 = -0.034173336$, $w_*^2 = -0.007875761$, $w_*^3 = 0.022017771$, $w_*^4 = 0.02926685$, $w_*^5 = -0.014739443$, $w_*^6 = 0.35831347$, $w_*^7 = 0.18551769$, $w_*^8 = 0.0704214$ and $w_*^9 = 0.399232137$. By using the information in Tables II and III, the solution together with (21) and (22), we obtained the portfolio expected return of $\mu = 3.603 \times 10^{-4}$ and risk of $\sigma_p^2 = 3.39 \times 10^{-4}$. Table IV shows the companies and their allocations for an investment strategy that seeks to minimize risk. Negative allocations mean the investor “short sells” the company. Short selling refers to when an investor borrows and sells stock they do not own in order to gain or profit from a future decline in the price of the stock by buying back the same number of stocks they initially borrowed at a lower price. The profit made is realized from the difference between the selling price and the buying price.

Table IV. Table of companies and allocation strategies

Company	Allocation
Apple Inc (APPL)	-3%
Microsoft Corporation (MSFT)	-1%
Amazon.com Inc (AMZN)	2%
Alphabet Inc (GOOG)	3%
Meta Platforms Inc (META)	-1%
Johnson & Johnson (JNJ)	36%
Berkshire Hathaway Inc. (BRKA.VI)	19%
JP Morgan Chase & Co. (JPM)	7%
The Procter & Gamble Company (PG)	40%
General Motors (GM)	-2%

Table V. Table of results for MHS, PRP**, NPRP and DPRP

s_p	MHS			PRP**			NPRP			DPRP		
	NI	FE	time(s)	NI	FE	time(s)	NI	FE	time(s)	NI	FE	time(s)
s_1	5	64	0.005288	5	64	0.004133	5	64	0.004676	5	64	0.003556
s_2	6	70	0.003422	6	70	0.004538	6	70	0.004453	6	70	0.005684
s_3	7	77	0.004541	7	75	0.005398	11	121	0.007433	9	102	0.008995
s_4	6	70	0.009871	6	70	0.015019	6	70	0.01191	6	70	0.010097
s_5	7	85	0.011576	13	148	0.01496	7	85	0.010684	7	85	0.009259
s_6	6	69	0.005796	6	66	0.00292	6	70	0.003295	6	68	0.002667
s_7	7	83	0.003963	11	125	0.006356	7	83	0.004119	7	83	0.004308
s_8	8	92	0.006616	8	91	0.004882	9	104	0.003456	8	92	0.00413
s_9	9	98	0.003916	10	114	0.005292	9	103	0.005005	10	113	0.004202
s_{10}	9	99	0.00686	10	113	0.011658	9	99	0.006805	9	99	0.007764
s_{11}	6	70	0.005086	7	75	0.007906	7	75	0.00652	6	65	0.004774
s_{12}	7	77	0.00732	7	78	0.006049	7	75	0.005324	7	75	0.004647
s_{13}	8	89	0.005108	8	91	0.005202	7	83	0.003531	7	83	0.003669
s_{14}	7	79	0.003241	7	76	0.005085	7	78	0.003135	7	78	0.003904
s_{15}	8	92	0.004877	8	92	0.005433	10	111	0.004382	10	109	0.004631
s_{16}	4	49	0.003995	4	49	0.001346	4	49	0.002177	4	49	0.001707
s_{17}	8	90	0.003924	11	119	0.007074	9	104	0.004725	9	104	0.004761
s_{18}	10	114	0.004696	10	112	0.006424	10	107	0.004834	15	161	0.007155
s_{19}	6	70	0.003935	7	80	0.0041	6	70	0.003014	6	70	0.002852
s_{20}	8	92	0.00287	10	110	0.004868	8	90	0.005498	10	111	0.007206
s_{21}	8	92	0.004874	11	128	0.004986	8	93	0.004226	8	93	0.003665
s_{22}	6	65	0.002654	7	78	0.00334	6	66	0.003247	6	70	0.003581
s_{23}	7	78	0.002639	9	100	0.004577	6	68	0.003484	7	78	0.003617
s_{24}	6	70	0.002791	8	91	0.004222	6	70	0.002859	6	70	0.003558
s_{25}	7	78	0.003009	8	88	0.003725	9	98	0.004161	9	96	0.004978
s_{26}	8	89	0.00411	9	98	0.004024	9	99	0.003986	9	97	0.004543
s_{27}	8	91	0.004967	10	112	0.004178	10	115	0.006298	8	89	0.003592
s_{28}	7	77	0.003451	6	69	0.002903	9	104	0.004206	9	104	0.006452
s_{29}	4	50	0.002292	4	50	0.002473	4	50	0.002461	4	50	0.002594
s_{30}	7	78	0.005794	7	77	0.004806	8	90	0.004313	8	90	0.004338

Table V presents the performance metrics of various optimization methods, including MHS, PRP**, NPRP, and DPRP, across different starting points (s_p). The metrics evaluated include "NI" (number of iterations), "FE" (number of function evaluations), and "TIME" (CPU time in seconds). The best results are written in bold. The MHS method performed the lowest number of iterations for 86% of the different starting points, followed by NPRP at 66%, DPRP at 63%, and PRP** at 50%. This suggests that MHS generally requires fewer iterations to converge to the solution compared to the other methods. In terms of function evaluations (FE), MHS demonstrated superior performance for 53% of the starting points. NPRP and DPRP followed closely behind, each achieving 46%, while PRP** trailed with 43%. This indicates that MHS tends to achieve convergence with fewer function evaluations in a

significant portion of the cases. Considering CPU time (TIME), MHS emerged as the most time-efficient method, consuming the least amount of CPU time for 43% of the starting points. The second best method was DPRP at 27%, followed by NPRP at 23% and lastly PRP** at 7%. In summary, MHS stands out for its efficiency in terms of iterations, function evaluations and time consumed in seconds across the majority of starting points.

5. CONCLUSION

In this paper, we presented a hybrid of the famous HS method and its modification which is denoted as DHS method. The new approach (MHS method) possesses good performance and is globally convergent under the strong Wolfe line search conditions. Furthermore, the practical applicability of the MHS method is explored in optimizing risk in portfolio selection. The new approach showed to be effective in solving the portfolio selection problem.

REFERENCES

- A. B. Abubakar, P. Kumam, M. Malik, P. Chaipunya and A. H. Ibrahim, *A hybrid FR-DY conjugate gradient algorithm for unconstrained optimization with application in portfolio selection*. AIMS Math, 6(6) (2021), 6506-6527.
- O. J. Adeleke, M. O. Olusanya and I. A. Osinuga, *A PRP-HS Type Hybrid Nonlinear Conjugate Gradient Method for Solving Unconstrained Optimization Problems*, In: Silhavy, R., Silhavy, P., Prokopova, Z (eds) Intelligent Systems Applications in Software Engineering. CoMeSySo 2019 (2019). Advances in Intelligent Systems and Computing, 1046, Springer, Cham.
- A. Alhawarat, M. Mamat, M. Rivaie and Z. Salleh, *An efficient hybrid conjugate gradient method with the strong Wolfe-Powell line search*, Math. Probl. Eng., 2015: 103517 (2015).
- N. Andrei, *An unconstrained optimization test functions collection*, Adv. Model. Optim., 10(1) (2008), 147–161.
- 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.
- M. C. Bartholomew-Biggs, *Nonlinear Optimization with Financial Applications*. Kluwer Academic Publishers, Boston (2006).
- M. C. Bartholomew-Biggs and S. J. Kane, *A global optimization problem in portfolio selection*. Comput. Manag. Sci., 6 (2009) 329-345.
- Y. H. Dai, *A family of hybrid conjugate gradient methods for unconstrained optimization*, Math. Comput., 72(243) (2003), 1317-1328.
- Y. H. Dai and L. Liao, *New conjugacy conditions and related nonlinear conjugate gradient methods*, Appl. Math. Optim., 43 (2001), 87-101.

-
- Z. Dai and F. Wen, *Another improved Wei-Yao-Liu nonlinear conjugate gradient method with sufficient descent property*, Appl. Math. Comput., 218(14) (2012), 7421-7430.
- T. Diphofu and P. Kaelo, *Another three-term conjugate gradient method close to the memoryless BFGS for large scale unconstrained optimization problems*, Mediterr. J Math., 18:211 (2021).
- T. Diphofu, P. Kaelo and A. R. Tufa, *A modified nonlinear conjugate gradient algorithm for unconstrained optimization and portfolio selection problems*, RAIRO Oper. Res., 57(2) (2023), 817-835.
- E. D. Dolan, J. J. Moré, *Benchmarking optimization software with performance profiles*, Math. Program., 91 (2002), 201-214.
- M. Fang, M. Wang, M. Sun and R. Chen, *A Modified Hybrid Conjugate Gradient Method for Unconstrained Optimization*, J. Math., 2021(1) (2021), 1-9.
- R. Fletcher, *Practical methods of optimization, Unconstrained Optimization*, John Wiley & Sons, New York 1(1987).
- R. Fletcher and C. M. Reeves, *Function minimization by conjugate gradients*, Computer J., 7(2) (1964), 149-154.
- H. Guan and S. Wang, *A Modified Conjugate Gradient Method for Solving Large-Scale Nonlinear Equations*, Math. Probl. Eng., 2021:9919595 (2021), <https://doi.org/10.1155/2021/9919595>.
- W. W. Hager and H. Zhang, *A survey of nonlinear conjugate gradient methods*, Pacific J. Optim., 2 (2011), 35-58.
- M. R. Hestenes and E. Stiefel, *Methods of conjugate gradients for solving linear systems*, J Res. Nat. Bureau Stand., 49(6) (1952), 409-436.
- 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(174) (2021), 1-24.
- X. Jiang, W. Liao, J. Yin, J. Yin and J. Jian, *A new family of hybrid three-term conjugate gradient methods with applications in image restoration*, Numer. Algor., 91(2) (2022), 1-31.
- Y. Liu and C. Storey, *Efficient generalized conjugate gradient algorithms, Part 1: Theory*, J Optim. Theory Appl., 69(1) (1991), 129-137.
- H. Markowitz, *Portfolio Selection*, J. Finance, 7(1) (1952), 77-91.
- P. Mtagulwa and P. Kaelo, *An efficient modified PRP-FR hybrid conjugate gradient method for solving unconstrained optimization problems*, Appl. Numer. Math., 145 (2019), 111-120.
- A. A. Mustafa, *New Spectral LS conjugate gradient method for nonlinear unconstrained optimization*, Int. J Comput. Math., 100 (2023), 838-846.
- E. Polak and G. Ribière, *Note sur la convergence de directions conjuguées*, Rev. Francaise Informat Recherche Opérationnelle, 3e Année, 16 (1969), 35-43.
- B. T. Polyak, *The conjugate gradient method in extreme problems*, USSR Comput. Math. Math. Phys, 9(4) (1969), 94-112.
- I. M. Sulaiman, M. Malik, A. M. Awwal, P. Kumam, M. Mamat and S. Al-Ahmad, *On three-term conjugate gradient method for optimization problems with applications on COVID-19 model and robotic motion control*, Adv Cont Discr Mod., 2022:1 (2022). <https://doi.org/10.1186/s13662-021-03638-9>.
- Z. Wei, S. Yao and L. Liu, *The convergence properties of some new conjugate gradient methods*, Appl. Math. Comput., 183(2) (2006), 1341-1350.
- L. Zhang, *An improved Wei-Yao-Liu nonlinear conjugate gradient method for optimization computation*, Appl. Math. Comput., 215(6) (2009), 2269-2274.

G. Zoutendijk, *Nonlinear programming, computational methods*, In: J. Abadie Ed., *Integer and Nonlinear Programming*, North-Holland, Amsterdam, 1970, pp. 37-86.

P. Mtagulwa
Department of Mathematics,
Botho University,
P.O. Box 501564, Gaborone, Botswana

P. Kaelo
Department of Mathematics,
University of Botswana,
P/Bag UB00704, Gaborone, Botswana
Email: kaelop@ub.ac.bw

T. Diphofu
Department of Mathematics,
University of Botswana,
P/Bag UB00704, Gaborone, Botswana

K. Kaisara
Department of Mathematics,
Botho University,
P.O. Box 501564, Gaborone, Botswana