Hindawi
Mathematical Problems in Engineering
Volume 2018, Article ID 1523267, 8 pages
https://doi.org/10.1155/2018/1523267

Research Article

Hybrid Modification of Accelerated Double Direction Method

Milena J. Petrović,[1] **Predrag S. Stanimirović**,[2] **Nataša Kontrec**,[1] **and Julija Mladenović**[3]

[1] *University of Priština, Faculty of Science, Lole Ribara 29, 28000 Kosovska Mitrovica, Serbia*
[2] *University of Niš, Faculty of Science and Mathematics, Višegradska 33, 18000 Niš, Serbia*
[3] *Politehnika, School for New Technologies, Autoput 18, 11000 Belgrade, Serbia*

Correspondence should be addressed to Milena J. Petrović; milena.petrovic@pr.ac.rs

Received 29 May 2018; Revised 20 October 2018; Accepted 28 October 2018; Published 6 November 2018

Academic Editor: J.-C. Cortés

We present a hybridization of the accelerated gradient method with two vector directions. This hybridization is based on the usage of a chosen three-term hybrid model. Derived hybrid accelerated double direction model keeps preferable properties of both included methods. Convergence analysis demonstrates at least linear convergence of the proposed iterative scheme on the set of uniformly convex and strictly convex quadratic functions. The results of numerical experiments confirm better performance profile in favor of derived hybrid accelerated double direction model when compared to its forerunners.

1. Introduction

The main goal herein is to derive an efficient optimization method for minimization of an objective function $f : \mathbb{R}^n \longrightarrow \mathbb{R}$. Therewith, we assume the function f is uniformly convex and twice continuously differentiable. Furthermore, for the gradient and the Hessian of the function f at the k-th iterative point we use the next notation:

$$\begin{aligned} g_k(x) &= \triangledown f(x_k), \\ G_k(x) &= \triangledown^2 f(x_k). \end{aligned} \tag{1}$$

The general form of the iterations for finding the extreme values of the objective function f is given by the next expression:

$$x_{k+1} = x_k + t_k d_k, \tag{2}$$

where x_k is the current, x_{k+1} is the next iterative point, t_k is iterative step length value, and d_k is an iterative vector direction which leads us to the solution of the problem. Certainly, t_k and d_k are the most important issues of an optimization model (2) and they generate the efficiency of a relevant method. For that reason, the way of defining these two crucial elements is of great importance for each minimization scheme.

In one of the first algorithms for solving unconstrained optimization problems, denoted as the steepest descent gradient method which is exposed by Cauchy, the iteration is defined as

$$x_{k+1} = x_k - t_k g_k, \tag{3}$$

and here, the descent direction is simply presented as the negative gradient vector, while iterative step size value is calculated by the exact line search formula:

$$t_k = \arg\min_{t>0} f(x_k + t d_k). \tag{4}$$

Furthermore, in the general Newton method

$$x_{k+1} = x_k - G_k^{-1} g_k t_k, \tag{5}$$

the vector direction is calculated as the product of the inverse Hessian and the gradient of the objective function. Defining the vector direction in this way guarantees fast convergence properties, but still, practical computing of the function Hessian and its inverse can be difficult. And so, many modified Newton, Newton-conjugate, and quasi Newton schemes were developed in which the calculation of the Hessian and its inverse is, somehow, avoided.

In the quasi-Newton methods the Hessian of the goal function or its inverse is approximated by the adequately defined matrix. Using this type of methods we generally reduce the time of computations since we avoid the complicated calculations in deriving the Hessian of the objective function. Nevertheless, the methods of quasi Newton type preserve good properties of the Newton method. For these reasons, in this paper, we propose the method of quasi Newton type where the value of the iterative step size parameter t_k is obtained by the inexact Backtracking line search procedure.

In the second section we give an overview of some accelerated gradient methods and hybrid iterations. We elaborate the deriving of the hybrid accelerated double direction method and restate the algorithm in the third section of this paper. In the fourth section we give a convergence analysis regarding the proposed iteration. Numerical experiments and comparison are presented in the last section of this paper.

2. Preliminaries: Accelerated Gradient Methods and Hybrid Iterations

The authors in [1] rightfully detected a class of accelerated gradient descent methods, defined by the general iterative scheme

$$x_{k+1} = x_k - \gamma_k^{-1} t_k g_k. \tag{6}$$

In the previous expression, γ_k presents an iterative acceleration parameter which improves performance of the relevant method. A common way to determine this parameter is through the features of the second-order Taylor's series taken on appropriate scheme (6). Acceleration parameters that were computed in such way are applied in the methods described in [1–5]. According to the iteration form (6), we can conclude that the accelerated gradient methods are of the quasi-Newton type in which the approximation of the Hessian, i.e., its inverse, is obtained by the scalar matrix $\gamma_k I$, where I is appropriate identity matrix and $\gamma_k = \gamma(x_k, x_{k-1})$ is the matching acceleration parameter. Here are several expressions for defining the acceleration parameters of some accelerated gradient schemes:

$$\gamma_{k+1}^{SM} = 2\gamma_k \frac{\gamma_k \left[f\left(x_{k+1}\right) - f\left(x_k\right)\right] + t_k \left\|g_k\right\|^2}{t_k^2 \left\|g_k\right\|^2}. \tag{7}$$

(SM method [1])

$$\gamma_{k+1}^{ADD} = 2 \frac{f\left(x_{k+1}\right) - f\left(x_k\right) - \alpha_k g_k^{\mathrm{T}} \left(\alpha_k d_k - \gamma_k^{-1} g_k\right)}{\left(\alpha_k d_k - \gamma_k^{-1} g_k\right)^{\mathrm{T}} \left(\alpha_k d_k - \gamma_k^{-1} g_k\right)}, \tag{8}$$

(ADD method [4])

$$\gamma_{k+1}^{ADSS} = 2 \frac{f\left(x_{k+1}\right) - f\left(x_k\right) + \left(\alpha_k \gamma_k^{-1} + \beta_k\right) \left\|g_k\right\|^2}{\left(\alpha_k \gamma_k^{-1} + \beta_k\right)^2 \left\|g_k\right\|^2}, \tag{9}$$

(ADSS method [2])

$$\gamma_{k+1}^{TADSS} = 2 \frac{f\left(x_{k+1}\right) - f\left(x_k\right) + \psi_k \left\|g_k\right\|^2}{\psi_k^2 \left\|g_k\right\|^2}, \tag{10}$$

(TADSS method [5])

$$\gamma_{k+1}^{HSM} = 2\gamma_k \frac{\gamma_k \left[f\left(x_{k+1}\right) - f\left(x_k\right)\right] + \left(\alpha_k + 1\right) t_k \left\|g_k\right\|^2}{\left(\alpha_k + 1\right)^2 t_k^2 \left\|g_k\right\|^2}, \tag{11}$$

(HSM method [3]).

An interesting concept of merging iterations through the hybrid expression was suggested in some research articles (see [6–8]). Some of representations are given by the next set of iterations:

$$\begin{aligned} u_1 &= u \in \mathbb{C}, \\ u_{k+1} &= T u_k, \quad k \in \mathbb{N}, \end{aligned} \tag{12}$$

$$\begin{aligned} v_1 &= v \in \mathbb{C}, \\ v_{k+1} &= \left(1 - \alpha_k\right) v_k + \alpha_k T v_k, \quad k \in \mathbb{N}, \end{aligned} \tag{13}$$

$$\begin{aligned} z_1 &= z \in \mathbb{C}, \\ z_{k+1} &= \left(1 - \alpha_k\right) z_k + \alpha_k T y_k, \\ y_k &= \left(1 - \beta_k\right) z_k + \beta_k T z_k, \quad k \in \mathbb{N}, \end{aligned} \tag{14}$$

where $T : \mathbb{C} \longrightarrow \mathbb{C}$ is a mapping defined on nonempty convex subset $\mathbb{C}$ of a normed space $\mathbb{E}$, v_k, z_k, and y_k present the sequences defined by proposed iterations, and $\{\alpha_k\}, \{\beta_k\} \in (0, 1)$.

In [9] it was proved that the hybrid method

$$\begin{aligned} x_1 &= x \in \mathbb{R}, \\ x_{k+1} &= T y_k, \\ y_k &= \left(1 - \alpha_k\right) x_k + \alpha_k T x_k, \quad k \in \mathbb{N}, \end{aligned} \tag{15}$$

proposed by Picard, Mann, and Ishikawa, upgrades the hybrid models mentioned above. The authors of [3] used the advantages of the hybrid model (15) and derived a hybrid version of the accelerated gradient SM method from [1], termed the HSM method and defined by

$$x_{k+1} = x_k - \left(\alpha_k + 1\right) t_k \gamma_k^{-1} g_k. \tag{16}$$

Numerical tests from [3] confirmed that the hybrid model (16) upgrades its forerunner SM iterative rule.

3. HADD Algorithm

We are motivated by the confirmed advantages which were approved in [3] when the scheme (15) was applied on the SM method. As a result, the hybrid SM model (called HSM) was defined and tested in [3]. Herein, we apply the same hybridization strategy to the accelerated double direction method (ADD method, shortly), introduced in [4]. Derived scheme will be based on the hybrid scheme (15) and with that it keeps the accelerated features of the ADD iterations.

In order to complete the presentation, we start from the ADD iteration:

$$x_{k+1} = x_k - t_k \gamma_k^{-1} g_k + t_k^2 d_k, \tag{17}$$

```
**Require:** Objective function $f(x)$, gradient $g_k$ and stepsize $t_k$.
1: Compute
    $d_k(t) = \begin{cases} d_k^* & \text{if } k \le m-1 \\ \sum_{i=2}^{m} t^{i-1} d_{k-i+1}^* & \text{if } k \ge m, \end{cases}$
    where $t = t_k$ is the step size, $d_k^*$ is the solution of the problem $\min_{x \in R} \Phi_k(d)$, and
    $\Phi_k(d) = \nabla f(x_k)^{\mathrm{T}} d + \frac{1}{2}\gamma_{k+1} I = g_k^{\mathrm{T}} d + \frac{1}{2}\gamma_{k+1} I.$
2: Return $d_k = d_k^*$.
```

Algorithm 1: Procedure **Second direction** *(calculation of the second direction vector d_k).*

where t_k is appropriately defined step size, the first direction vector is given by $-\gamma_k^{-1} g_k = -(\gamma_k^{ADD})^{-1} g_k$, and the second one, d_k, is determined based on the next procedure **Second direction**. That procedure was introduced in [4], which was derived as a practical appearance of the more general procedure considered in [10]. The procedure **Second direction** is restated in Algorithm 1.

Remark 1. For further investigation within this topic, the second direction d_k in the ADD iteration can be defined differently. For example, in [11] the authors proposed directional k-step Newton methods for solving a single nonlinear equation in n-variables. Accordingly, they established the semi-local convergence analysis for these models, based on two different approaches. The first one is based on recurrent relations, while the other, more preferable, is established using recurrent functions. Using one (or both) approaches from [11] in determining the second direction d_k in the ADD method as well as in its hybrid version can be an interesting topic in further research.

Applying the hybrid scheme (15) on the iterative rule (17), we get the hybrid iterative scheme

$$\begin{aligned} x_1 &= x \in \mathbb{R}, \\ x_{k+1} &= T y_k = y_k - t_k \gamma_k^{-1} g_k + t_k^2 d_k, \\ y_k &= (1-\alpha_k) x_k + \alpha_k T x_k \\ &= (1-\alpha_k) x_k + \alpha_k \left(x_k - t_k \gamma_k^{-1} g_k + t_k^2 d_k\right) \\ &= x_k - \alpha_k t_k \gamma_k^{-1} g_k + \alpha_k t_k^2 d_k, \quad k \in \mathbb{N}. \end{aligned} \tag{18}$$

After replacing the third expression from the set of equations (18) into the second one, the next iterative rule follows:

$$x_{k+1} = x_k - (\alpha_k + 1) t_k \gamma_k^{-1} g_k + (\alpha_k + 1) t_k^2 d_k. \tag{19}$$

To simplify further calculation, we will use a constant value for the parameter $\alpha_k \in (0,1)$ in (19), just like the authors did in [1, 9]. So, in (19), instead of $\alpha_k + 1 \in (1,2)$ we simply take $\alpha \in (1,2)$. Now we can restate a hybrid ADD method, or the HADD iterative scheme, as follows:

$$x_{k+1} = x_k - \alpha t_k \gamma_k^{-1} g_k + \alpha t_k^2 d_k, \quad \alpha \in (1,2). \tag{20}$$

Yet, we need to determine the iterative value of the accelerated parameter $\gamma_k = \gamma_k^{HADD}$. As we mentioned previously, this parameter can be appropriately defined using Taylor's expansion of the proposed iteration (20) in two successive iterative points:

$$\begin{aligned} f(x_{k+1}) \approx{} & f(x_k) + g_k^{\mathrm{T}} \alpha \left(t_k^2 d_k - t_k \gamma_k^{-1} g_k\right) + \frac{1}{2} \\ & \cdot \alpha^2 \left(t_k^2 d_k - t_k \gamma_k^{-1} g_k\right)^{\mathrm{T}} \nabla^2 f(\xi) \\ & \cdot \left(t_k^2 d_k - t_k \gamma_k^{-1} g_k\right). \end{aligned} \tag{21}$$

The parameter ξ in the previous expansion fulfills the condition

$$\begin{aligned} & \xi \in [x_k, x_{k+1}], \\ & \xi = x_k + \kappa (x_{k+1} - x_k) = x_k + \kappa \left(\alpha t_k^2 d_k - \alpha t_k \gamma_k^{-1} g_k\right), \\ & \qquad 0 \le \kappa \le 1. \end{aligned} \tag{22}$$

In the next relation we substitute the value $\bigtriangledown^2 f(\xi)$ from (21) by the scalar matrix $\gamma_{k+1} I$, which leads to

$$\begin{aligned} & f(x_{k+1}) \\ & \quad = f(x_k) + g_k^{\mathrm{T}} \alpha \left(t_k^2 d_k - t_k \gamma_k^{-1} g_k\right) \\ & \qquad + \frac{1}{2} \alpha^2 \gamma_{k+1} \left(t_k^2 d_k - t_k \gamma_k^{-1} g_k\right)^{\mathrm{T}} \left(t_k^2 d_k - t_k \gamma_k^{-1} g_k\right). \end{aligned} \tag{23}$$

From (23), it is possible to derive the approximation factor $\gamma_{k+1} = \gamma_{k+1}^{HADD}$ of the HADD scheme:

$$\begin{aligned} \gamma_{k+1}^{HADD} &= \gamma_{k+1} \\ &= 2 \frac{f(x_{k+1}) - f(x_k) - \alpha g_k^{\mathrm{T}} \left(t_k^2 d_k - t_k \gamma_k^{-1} g_k\right)}{\alpha^2 t_k^2 \left(t_k d_k - \gamma_k^{-1} g_k\right)^{\mathrm{T}} \left(t_k d_k - \gamma_k^{-1} g_k\right)}. \end{aligned} \tag{24}$$

With the aim of preserving the Second-Order Necessary Condition and Second-Order Sufficient Condition, we assume positivity of the acceleration parameter: $\gamma_{k+1} > 0$. In practical computation, it is possible that (24) generates negative value for γ_{k+1}. We resolve this situation by taking $\gamma_{k+1} = 1$ in such cases. As a consequence, then the first

```
Require: Objective function $f(x)$, the direction $d_k$ of the search at the point $x_k$, and numbers $0 < \sigma < 0.5$ and $\beta \in (0,1)$.
$t = 1$.
While $f(x_k + td_k) > f(x_k) + \sigma t g_k^T d_k$, take $t := t\beta$.
Return $t_k = t$.
```

Algorithm 2: The Backtracking line search procedure.

```
Require: $0 < \rho < 1$, $0 < \tau < 1$, $x_0 \in \mathbb{R}^n$ and $\alpha \in (1,2)$.
1: Set $k = 0$. For given $x_0$, take $\gamma_0 = 1$ and calculate $f(x_0)$ and $g(x_0)$.
2: If $\|g_k\| < \epsilon$, then return $x_k$, $f(x_k)$ and stop the algorithm, else continue by the next step.
3: Compute the iterative step length, $t_k$, by Algorithm 2.
4: Compute second vector direction, $d_k$, by the rule Second direction described in Algorithm 1.
5: Compute $x_{k+1}$ using (20), then calculate $f(x_{k+1})$ and $g_{k+1}$.
6: Calculate the approximation parameter $\gamma_{k+1}$ using (24).
7: If $\gamma_{k+1} < 0$ take $\gamma_{k+1} = 1$.
8: Set $k := k + 1$ and go to Step 2.
9: Return $x_{k+1}$ and $f(x_{k+1})$.
```

Algorithm 3: HADD method.

vector direction becomes the negative gradient vector $-g_k$. In this special case, the next iterative point of the iteration (20) becomes

$$x_{k+2} = x_{k+1} - \alpha t_{k+1} g_{k+1} + \alpha t_{k+1}^2 d_{k+1}. \tag{25}$$

In order to present the main HADD algorithm, we need two additional auxiliary procedures. The first one is previously displayed Algorithm 1, by which we calculate the second vector direction, d_k. The second procedure is the Backtracking line search algorithm for calculating the iterative step size value.

Algorithm 3 describes the main algorithm, termed the HADD algorithm.

4. Convergence of the HADD Method

The convergence properties of the established HADD iterative method are considered on the set of uniformly convex and strictly convex quadratic functions. In the case of uniformly convex functions the statements are the same as exposed in [1, 4]. For that reason, we just restate the following lemma, in which decreasing of the objective function in two successive points is estimated with respect to the HADD scheme. Thereupon, the upcoming theorem confirms linear convergence of our hybrid accelerated model.

Lemma 2. *Suppose the function f is twice continuously differentiable and uniformly convex on $\mathbb{R}^n$. With that, let the sequence $\{x_k\}$ be generated by Algorithm 3. Then the next estimation is true*

$$f(x_k) - f(x_{k+1}) \geq \mu \|g_k\|^2, \tag{26}$$

where

$$\mu = \left\{ \frac{\sigma}{M}, \frac{\sigma(1-\sigma)}{L}\beta \right\}. \tag{27}$$

Theorem 3. *For the twice continuously differentiable and uniformly convex function f on $\mathbb{R}^n$ and the sequence $\{x_k\}$ generated by Algorithm 3, the following holds:*

$$\lim_{k\longrightarrow\infty} \|g_k\| = 0. \tag{28}$$

Therewith, the sequence $\{x_k\}$ converges to the optimal solution at least linearly.

We show now that the iteration (20) is convergent regarding the set of strictly convex quadratic functions

$$f(x) = \frac{1}{2}x^{\mathrm{T}}Ax - b^{\mathrm{T}}x. \tag{29}$$

In (29), it is assumed that A is a real $n \times n$ symmetric positive definite matrix and that vector $b \in \mathbb{R}^n$ is given. The smallest and the largest eigenvalues of the matrix A, respectively, are denoted by λ_1 and λ_n.

Lemma 4. *Let f be the strictly convex quadratic function defined by (29), where $A \in \mathbb{R}^{n\times n}$ is a symmetric positive definite matrix. Let λ_1 and λ_n be the smallest and the largest eigenvalues of A. Then, the following inequalities are valid for the hybrid accelerated gradient model (20):*

$$\lambda_1 \leq \frac{\gamma_{k+1}}{t_{k+1}} \leq \frac{4\lambda_n}{\sigma}, \quad k \in \mathbb{N}, \tag{30}$$

Proof. Let us calculate the difference in two successive iterative points of the goal function (29):

$$\begin{aligned} f(x_{k+1}) - f(x_k) = {} & \frac{1}{2}x_{k+1}^{\mathrm{T}}Ax_{k+1} - b^{\mathrm{T}}x_{k+1} - \frac{1}{2}x_k^{\mathrm{T}}Ax_k \\ & + b^{\mathrm{T}}x_k. \end{aligned} \tag{31}$$

Including the iteration (20) we continue computations:

$$\begin{aligned}
f\left(x_{k+1}\right)-f\left(x_k\right) &= \frac{1}{2}\left(x_k-\alpha t_k\gamma_k^{-1}g_k+\alpha t_k^2 d_k\right)^{\mathrm{T}} \\
&\quad\cdot A\left(x_k-\alpha t_k\gamma_k^{-1}g_k+\alpha t_k^2 d_k\right) \\
&\quad-b^{\mathrm{T}}\left(x_k-\alpha t_k\gamma_k^{-1}g_k+\alpha t_k^2 d_k\right)-\frac{1}{2}x_k^{\mathrm{T}}Ax_k+b^{\mathrm{T}}x_k \\
&= \frac{1}{2}x_k^{\mathrm{T}}Ax_k-\frac{1}{2}\alpha t_k\gamma_k^{-1}g_k^{\mathrm{T}}Ax_k+\frac{1}{2}\alpha t_k^2 d_k^{\mathrm{T}}Ax_k-\frac{1}{2} \\
&\quad\cdot\alpha t_k\gamma_k^{-1}g_k^{\mathrm{T}}Ax_k+\frac{1}{2}\alpha^2 t_k^2\gamma_k^{-2}g_k^{\mathrm{T}}Ag_k-\frac{1}{2} \\
&\quad\cdot\alpha t_k^3\gamma_k^{-1}d_k^{\mathrm{T}}Ag_k+\frac{1}{2}\alpha t_k^2 d_k^{\mathrm{T}}Ax_k-\frac{1}{2}\alpha^2 t_k^3\gamma_k^{-1}d_k^{\mathrm{T}}Ag_k \\
&\quad+\frac{1}{2}\alpha^2 t_k^4 d_k^{\mathrm{T}}Ad_k-b^{\mathrm{T}}x_k+\alpha t_k\gamma_k^{-1}b^{\mathrm{T}}g_k-\alpha t_k^2 b^{\mathrm{T}}d_k \\
&\quad-\frac{1}{2}x_k^{\mathrm{T}}Ax_k+b^{\mathrm{T}}x_k.
\end{aligned} \tag{32}$$

Applying the equality $g_k = Ax_k - b$ and the symmetry property of A, we get

$$\begin{aligned}
f\left(x_{k+1}\right)-f\left(x_k\right) &= -\alpha t_k\gamma_k^{-1}g_k^{\mathrm{T}}Ax_k+\alpha t_k^2 d_k^{\mathrm{T}}Ax_k \\
&\quad-\alpha^2 t_k^3\gamma_k^{-1}d_k^{\mathrm{T}}Ag_k \\
&\quad+\frac{1}{2}\alpha^2 t_k^2\gamma_k^{-2}g_k^{\mathrm{T}}Ag_k \\
&\quad+\frac{1}{2}\alpha^2 t_k^4 d_k^{\mathrm{T}}Ad_k+\alpha t_k\gamma_k^{-1}b^{\mathrm{T}}g_k \\
&\quad-\alpha t_k^2 b^{\mathrm{T}}d_k.
\end{aligned} \tag{33}$$

The right hand side of the previous expression can be further transformed as follows:

$$\begin{aligned}
f\left(x_{k+1}\right)-f\left(x_k\right) &= \alpha t_k\gamma_k^{-1}\left(b^{\mathrm{T}}g_k-x_k^{\mathrm{T}}Ag_k\right) \\
&\quad+\alpha t_k^2\left(x_k^{\mathrm{T}}Ad_k-b^{\mathrm{T}}d_k\right)-\alpha^2 t_k^3\gamma_k^{-1}d_k^{\mathrm{T}}Ag_k+\frac{1}{2} \\
&\quad\cdot\alpha t_k^2\gamma_k^{-2}g_k^{\mathrm{T}}Ag_k+\frac{1}{2}\alpha^2 t_k^4 d_k^{\mathrm{T}}Ad_k \\
&= \alpha t_k g_k^{\mathrm{T}}\left(t_k d_k-\gamma_k^{-1}g_k\right)-\frac{1}{2} \\
&\quad\cdot\alpha^2 t_k^2\gamma_k^{-1}g_k^{\mathrm{T}}A\left(t_k d_k-\gamma_k^{-1}g_k\right)+\frac{1}{2} \\
&\quad\cdot\alpha^2 t_k^3 d_k^{\mathrm{T}}A\left(t_k d_k-\gamma_k^{-1}g_k\right) \\
&= \left(\alpha t_k g_k^{\mathrm{T}}-\frac{1}{2}\alpha^2 t_k^2\gamma_k^{-1}g_k^{\mathrm{T}}A+\frac{1}{2}\alpha^2 t_k^3 d_k^{T}A\right) \\
&\quad\cdot\left(t_k d_k-\gamma_k^{-1}g_k\right).
\end{aligned} \tag{34}$$

The replacement of $f(x_{k+1}) - f(x_k)$ by the right hand side of (34) into (24) leads us to

$$\gamma_{k+1} = 2\frac{\left(\alpha t_k g_k^{\mathrm{T}}-(1/2)\,\alpha^2 t_k^2\gamma_k^{-1}g_k^{\mathrm{T}}A+(1/2)\,\alpha^2 t_k^3 d_k^{T}A\right)\left(t_k d_k-\gamma_k^{-1}g_k\right)-\alpha t_k g_{\mathrm{k}}^{T}\left(t_k d_k-\gamma_k^{-1}g_k\right)}{\alpha^2\left(t_k^2 d_k-t_k\gamma_k^{-1}g_k\right)^{\mathrm{T}}\left(t_k^2 d_k-t_k\gamma_k^{-1}g_k\right)}. \tag{35}$$

After some calculations, we obtain

$$\gamma_{k+1} = \frac{\left(t_k d_k-\gamma_k^{-1}g_k\right)^{\mathrm{T}}A\left(t_k d_k-\gamma_k^{-1}g_k\right)}{\left(t_k d_k-\gamma_k^{-1}g_k\right)^{\mathrm{T}}\left(t_k d_k-\gamma_k^{-1}g_k\right)}. \tag{36}$$

Previous expression confirms that γ_{k+1} is the Rayleigh quotient of the real symmetric matrix A at the vector $t_k d_k - \gamma_k^{-1} g_k$, which leads us to the conclusion

$$\lambda_1 \le \gamma_{k+1} \le \lambda_n, \quad k \in \mathbb{N}. \tag{37}$$

The left hand side in inequalities (30) arrives from the fact $0 \le t_{k+1} \le 1$. To prove the right hand side of (30), we use the estimation [[3], eq. (3.8)]:

$$t_k > \frac{\beta(1-\sigma)\gamma_k}{\alpha L}. \tag{38}$$

Previous inequality implies

$$\frac{\gamma_{k+1}}{t_{k+1}} < \frac{L\alpha}{\beta(1-\sigma)}. \tag{39}$$

We can approximate the Lipschitz constant L by the largest eigenvalue λ_n and use the fact that $\alpha \le 2$, $0 < \sigma < 0.5$ and $\beta \in (\sigma, 1)$. Then (39) is restated to

$$\frac{\gamma_{k+1}}{t_{k+1}} < \frac{L\alpha}{\beta(1-\sigma)} < \frac{2\lambda_n}{\sigma\cdot 0.5} = \frac{4\lambda_n}{\sigma}. \tag{40}$$

Estimation of the Lipschitz constant L by the largest eigenvalue λ_n is certainly valid since the matrix A is symmetric and $g(x) = Ax - b$. From these two facts we conclude that

$$\begin{aligned}
\left\|g(x)-g(y)\right\| &= \left\|Ax-Ay\right\| = \left\|A(x-y)\right\| \\
&\le \left\|A\right\|\left\|x-y\right\| = \lambda_n\left\|x-y\right\|,
\end{aligned} \tag{41}$$

which completes the proof. □

Theorem 5. *Let the iterations (19) be applied on strictly convex quadratic function f given by the expression (29). Suppose that the condition*

$$\lambda_n < \frac{2\lambda_1}{\alpha} \tag{42}$$

holds for the largest and the smallest eigenvalues of symmetric positive definite matrix A. Then, the following estimations are true:

$$\left(p_i^{k+1}\right)^2 \le \delta^2 \left(p_i^k\right)^2, \qquad \left(q_i^{k+1}\right)^2 \le \lambda_n^2 \left(q_i^k\right)^2, \tag{43}$$

where

$$\delta = \max\left\{1 - \frac{\sigma\lambda_1}{2\lambda_n}, \frac{\lambda_n}{\lambda_1} - 1\right\} \tag{44}$$

and $p_i^k, q_i^k \in \mathbb{R}, k, i, n \in \mathbb{N}$. *Therewith*

$$\lim_{k\longrightarrow\infty} \|g_k\| = 0. \tag{45}$$

Proof. Let us consider the orthonormal system of eigenvectors $\{v_1, v_2, \ldots, v_n\}$ of the matrix A. Thereon, we construct the sequence of values $\{x_k\}$ by applying Algorithm 3 on strictly convex quadratic function f defined by (29). Then, for some $k \in \mathbb{N}$ and for some constants $p_1^k, p_2^k, \ldots, p_n^k, q_1^k, q_2^k, \ldots, q_n^k \in \mathbb{R}$ it follows that

$$g_k = Ax_k - b = \sum_{i=1}^{n} p_i^k v_i, \qquad d_k = \sum_{i=1}^{n} q_i^k v_i. \tag{46}$$

Applying (20), further we conclude that

$$\begin{aligned} g_{k+1} &= A\left(x_k - \alpha t_k \gamma_k^{-1} g_k + \alpha t_k^2 d_k\right) - b \\ &= \left(I - \alpha t_k \gamma_k^{-1} A\right) g_k + \alpha t_k^2 A d_k. \end{aligned} \tag{47}$$

Having in mind the representation (46), one can verify

$$\begin{aligned} g_{k+1} &= \sum_{i=1}^{n} p_i^{k+1} v_i \\ &= \sum_{i=1}^{n} \left(\left(1 - \alpha t_k \gamma_k^{-1} \lambda_i\right) p_i^k + \alpha t_k^2 \lambda_i q_i^k\right) v_i \\ &= \sum_{i=1}^{n} \left(1 - \alpha t_k \gamma_k^{-1} \lambda_i\right) p_i^k v_i + \alpha t_k^2 \sum_{i=1}^{n} \lambda_i q_i^k v_i. \end{aligned} \tag{48}$$

To prove

$$\left(p_i^{k+1}\right)^2 \le \delta^2 \left(p_i^k\right)^2, \left(q_i^{k+1}\right)^2 \le \lambda_n^2 \left(q_i^k\right)^2, \tag{49}$$

i.e., the inequalities (43), we only need to verify that

$$\left|1 - \alpha \frac{\lambda_i}{\gamma_k t_k^{-1}}\right| \le \delta \tag{50}$$

since $|\lambda_i| \le \lambda_n$ for all $i \in \{1, 2, \ldots, n\}$. There are two possibilities:

(1) $\alpha\lambda_i \le t_k^{-1}\gamma_k$.

This case implies the next set of inequalities:

$$\begin{aligned} 1 > \alpha \frac{\lambda_i}{\gamma_k t_k^{-1}} &= \frac{\alpha\lambda_i}{\gamma_k} \cdot t_k \ge \frac{\alpha\lambda_1}{\gamma_k} \frac{\beta(1-\sigma)\gamma_k}{L\alpha} \\ &= \frac{\lambda_1 \beta (1-\sigma)}{L} > \frac{\lambda_1 \sigma \cdot 0.5}{\lambda_n} = \frac{\lambda_1}{2\lambda_n}\sigma. \end{aligned} \tag{51}$$

As a consequence, we can conclude that

$$1 - \alpha \frac{\lambda_i}{\gamma_k t_k^{-1}} \le 1 - \frac{\lambda_1}{2\lambda_n}\sigma \le \delta. \tag{52}$$

(2) $\alpha\lambda_i \ge t_k^{-1}\gamma_k$.

In this case, one can verify the following estimations:

$$1 < \frac{\alpha\lambda_i}{\gamma_k t_k^{-1}} \le \frac{\alpha\lambda_n}{\lambda_1} \Longrightarrow \left|1 - \alpha \frac{\lambda_i}{\gamma_k t_k^{-1}}\right| \le \frac{\alpha\lambda_n}{\lambda_1} \le \delta. \tag{53}$$

The representation (46) and the fact that $\{v_1, v_2, \ldots, v_n\}$ is an orthonormal system of eigenvectors lead to the next conclusion

$$\|g_k\|^2 = \sum_{i=1}^{n} \left(d_i^k\right)^2. \tag{54}$$

Now, knowing that the parameter δ under condition $\lambda_n < 2\lambda_1/\alpha$ satisfies $0 < \delta < 1$, we confirm that the final statement is true.

$$\lim_{k\longrightarrow\infty} \|g_k\| = 0 \tag{55}$$

□

Remark 6. The assumption (42) used in the previous theorem is required in order to prove that the HADD process is convergent for the strictly convex quadratics. Therewith, knowing that the hybrid parameter $\alpha \in (1, 2)$ implies $\lambda_n/\lambda_1 \in (1, 2)$ points to the conclusion that Theorem 5 is applicable to very few cases. However, this is not entirely so since we choose only one particular value $\alpha \in (1, 2)$ for the practical computations. Regarding this matter, the authors in [3] numerically confirmed that the optimal value of the hybrid parameter α is the one close to the left limit of the interval $(1, 2)$, i.e., the value which is very close to 1. Therefore, we choose $\alpha = 1.1$ for numerical tests displayed in the next section. Choosing the similar values for hybrid parameter α, the condition (42) becomes very close to the condition $\lambda_n < 2\lambda_1$, used in [12], under which Q-linear convergence rate of the preconditioned BB method was established.

5. Computational Tests and Comparisons

The performance of the C++ implementation of derived HADD model is investigated on a set of 630 test unconstrained optimization problems picked from [13]. We conduct the testings on a Workstation Intel Celeron 1.6 GHz. The following stopping criteria are used:

$$\|g_k\| \le 10^{-6}, \qquad \frac{|f(x_{k+1}) - f(x_k)|}{1 + |f(x_k)|} \le 10^{-16}. \tag{56}$$

Table 1: Comparison between the HADD, ADD, and HSM methods regarding the minimal number of function evaluations.

Comparative methods	HADD	ADD	HSM	=
Number of function evaluations	142	0	18	50

The values of the Backtracking parameters are set up as follows: $\sigma = 0.0001$ and $\beta = 0.8$.

We compare the hybrid accelerated HADD method with its forerunner ADD scheme, as well as with the hybrid accelerated HSM method. The number of function evaluations is the performance profile measured in all tests. The dominance of the ADD method regarding the number of iterations among the other comparative models was confirmed in [4]. However, from that research we do not have any information about the behavior of the ADD method when the number of function evaluations is involved. With respect to this parameter, the HSM scheme upgrades the accelerated SM method as well as Nesterov's line search algorithm; see [3]. For these reasons, our experimental goal is to numerically prove better performance feature of the HADD method, considering the number of function evaluations, when compared with the ADD and the HSM method.

In Table 1, we display the number of problems, out of 630, for which an algorithm achieved the minimum number of function evaluations. In the same table, we also display the total number of problems for which all three algorithms achieved an equal number of function evaluations. Based on the results displayed in this table, it is obvious that the HADD scheme convincingly outperforms the other two comparative models.

For more clear visualization of the performance of the HADD algorithm versus the ADD and the HSM algorithms, we display in Figure 1 the Dolan-Moré's performance profile subject to the number of function evaluations metric. As we can see, the HADD scheme is more robust and therewith more efficient than the other two methods.

Obtained numerical results confirm that applied hybridization process is a good way to improve some important characteristics of chosen accelerated methods. Preferable outcomes of the HADD scheme, regarding analyzed characteristic, come from the properly chosen hybrid value α, together with derived accelerated parameter γ^{HADD}. Good convergent properties of defined HADD process can be a reason for applying proposed hybridization on some other gradient and accelerated gradient models.

6. Conclusion

We present a hybrid accelerated double direction gradient method for solving unconstrained optimization problems. The HADD method is derived by applying good properties of the hybrid representation introduced in [3] in conjunction with the form of double direction optimization model with accelerated parameter presented in [4]. The convergence of defined optimization model is provided on the set of uniformly convex and strictly convex quadratic functions.

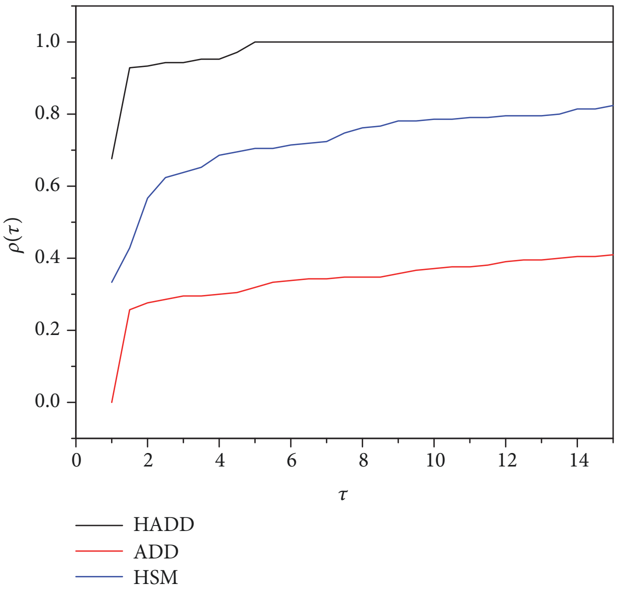


Figure 1: Performance profile for the HADD, ADD, and HSM methods regarding the number of function evaluations metric.

The HADD scheme reserves preferable features of both forerunner methods. Therewith, according to conducted numerical experiments, it outperforms the ADD and HSM methods regarding the requested number of function evaluations. We evaluated the Dolan-Moré performance profiles of comparative methods and showed that the HADD iteration is the most efficient compared to the other two algorithms.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The first and the third authors acknowledge support from the internal research project IS01-17 supported by the Faculty of Sciences and Mathematics, University of Priština, Serbia. The second author gratefully acknowledges support from the Project supported by Ministry of Education and Science of Republic of Serbia, Grant No. 174013.

References

[1] P. S. Stanimirović and M. B. Miladinović, "Accelerated gradient descent methods with line search," *Numerical Algorithms*, vol. 54, no. 4, pp. 503–520, 2010.

[2] M. J. Petrović, "An accelerated double step size model in unconstrained optimization," *Applied Mathematics and Computation*, vol. 250, pp. 309–319, 2015.

[3] M. J. Petrovic̀, V. Rakocevic̀, N. Kontrec, S. Panic̀, and D. Ilic̀, "Hybridization of accelerated gradient descent method," *Numer. Algor*, 2018.

[4] M. J. Petrović and P. S. Stanimirović, "Accelerated Double Direction Method for Solving Unconstrained Optimization Problems," *Mathematical Problems in Engineering*, vol. 2014, Article ID 965104, 8 pages, 2014.

[5] P. S. Stanimirović, G. V. Milovanović, and M. J. Petrović, "A Transformation of Accelerated Double Step Size Method for Unconstrained Optimization," *Mathematical Problems in Engineering*, vol. 2015, Article ID 283679, 8 pages, 2015.

[6] S. Ishikawa, "Fixed points by a new iteration method," *Proceedings of the American Mathematical Society*, vol. 44, pp. 147–150, 1974.

[7] W. R. Mann, "Mean value methods in iteration," *Proceedings of the American Mathematical Society*, vol. 4, pp. 506–510, 1953.

[8] E. Picard, "Memoire sur la theorie des equations aux derivees partielles et la methode des approximations successives," *Journal de Mathématiques Pures et Appliquées*, vol. 6, pp. 145–210, 1890.

[9] S. H. Khan, "A Picard-Mann hybrid iterative process," *Fixed Point Theory and Applications*, vol. 2013, article 69, 2013.

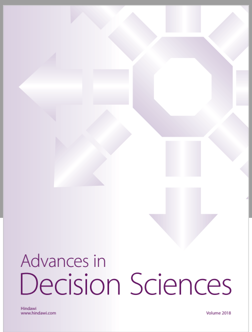
[10] N. I. Djuranović-Miličić and M. Gardaševic-Filipović, "A multistep curve search algorithm in nonlinear optimization: nondifferentiable convex case," *Facta Universitatis, Series: Mathematics and Informatics*, vol. 25, pp. 11–24, 2010.

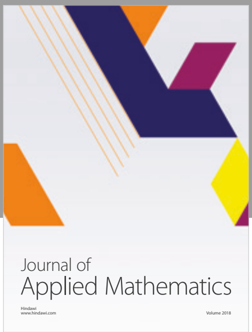
[11] A. Kumar, D. K. Gupta, E. Martinez, and S. Singh, "Directional k-step Newton methods in n variables and its semilocal convergence analysis," *Mediterranean Journal of Mathematics*, vol. 15, no. 2, Art. 34, 25 pages, 2018.

[12] B. Molina and M. Raydan, "Preconditioned Barzilai-BORwein method for the numerical solution of partial differential equations," *Numerical Algorithms*, vol. 13, no. 1-2, pp. 45–60, 1996.

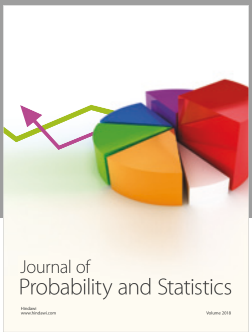
[13] N. Andrei, "An unconstrained optimization test functions collection," *Advanced Modeling and Optimization*, vol. 10, no. 1, pp. 147–161, 2008.

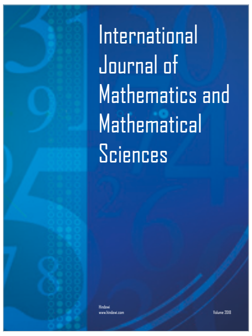
Advances in
Operations Research


Advances in
Decision Sciences

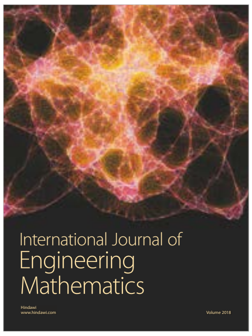

Journal of
Applied Mathematics

The Scientific
World Journal

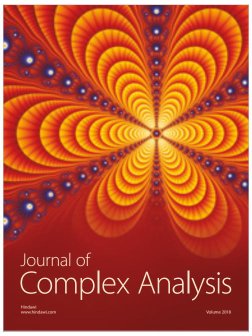

Journal of
Probability and Statistics

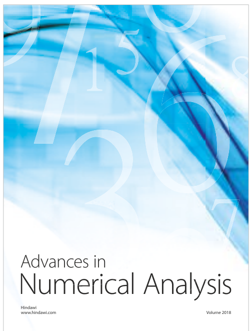

International
Journal of
Mathematics and
Mathematical
Sciences

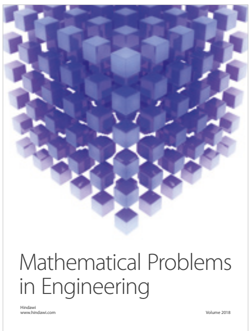
Journal of
Optimization

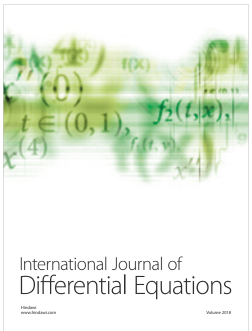

International Journal of
Engineering
Mathematics

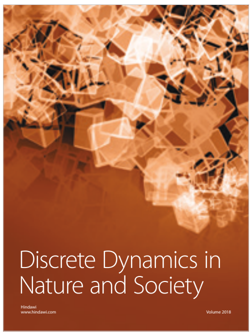
International Journal of
Analysis

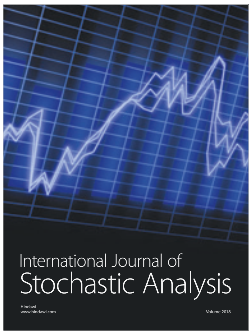

Journal of
Complex Analysis


Advances in
Numerical Analysis


Mathematical Problems
in Engineering

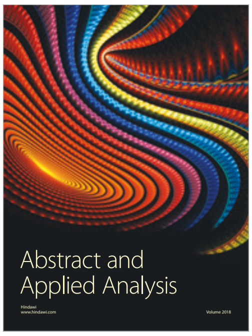

International Journal of
Differential Equations

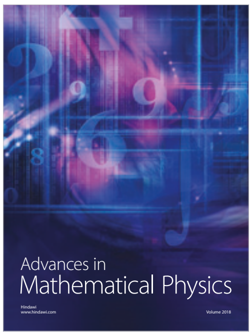

Discrete Dynamics in
Nature and Society


International Journal of
Stochastic Analysis

Journal of
Mathematics

Journal of
Function Spaces


Abstract and
Applied Analysis


Advances in
Mathematical Physics