Computational Experience with Scaled Dai-Yuan Methods for Unconstrained Optimization

Amal Al-Saidi\(^a\) and Mehiddin Al-Baali\(^b\)

\(^a\)Sultan Qaboos University, Muscat, Oman. \(^b\)Email: Amal2.AISaeedi@moe.om

Abstract—Very recently, Al-Saidi and Al-Baali (2021) have illustrated the usefulness of the scaled technique for the conjugate gradient methods when introduced to the Fletcher-Reeves method for unconstrained optimization. In this paper, we study the behaviour of the scaled Dai-Yuan method with several choices for the scaling parameter. Some numerical results for a set of standard test problems are described. It is shown that appropriate choices of the scaling parameter improve the performance of the Dai-Yuan method substantially in some cases.

Keywords—Unconstrained optimization, nonlinear conjugate gradient methods, Fletcher-Reeves and Dai-Yuan methods, scaled technique.

I. INTRODUCTION

We consider the unconstrained minimization problem

\[
\min_{x \in \mathbb{R}^n} f(x),
\]

where \(f : \mathbb{R}^n \to \mathbb{R}\) is a smooth function and its gradient \(g(x) = \nabla f(x)\) is available for any value of \(x\). There are several kinds of numerical methods for solving problem (1), which include the steepest descent, Newton and quasi-Newton methods as well as the class of Conjugate Gradient (CG) methods. The CG class is one of the useful choices for solving large-scale problems, because it does not require the storage of matrices (see for example Fletcher [12], Nocedal and Wright [16], Pytlak [22] and Andrei [6]). These type of line search methods are iterative of the form

\[
x_{k+1} = x_k + \alpha_k d_k,
\]

where \(\alpha_k\) is a positive step length and \(d_k\) is a search direction. They are respectively defined such that \(f_{k+1} = f(x_{k+1})\) is sufficiently smaller than \(f_k = f(x_k)\) and the descent property

\[
d_k^T g_k < 0
\]

holds. Here \(g_k = g(x_k) \neq 0\) since \(g_k = 0\) holds at a solution of problem (1).

In the class of CG methods, the search direction \(d_k\) is defined initially by the steepest descent choice \(d_1 = -g_1\) that satisfies the descent property (3) for \(k = 1\). The other directions are defined by

\[
d_{k+1} = -g_{k+1} + \beta_k d_k,
\]

where \(\beta_k\) is the CG parameter. In particular, Fletcher and Reeves [13] propose the positive value of \(\beta_k\) as

\[
\beta_k^{FR} = \frac{\|g_{k+1}\|^2}{\|g_k\|^2},
\]

where \(\|\cdot\|\) denotes the Euclidean norm. Usually other choices for the parameter \(\beta_k\) are reduced to the FR formula if \(f(x)\) is a strictly convex quadratic function and \(\alpha_k\) is calculated by solving the minimization subproblem

\[
\alpha_k = \arg \min_{\alpha} f(x_k + \alpha d_k)
\]

exactly. Prior to Al-Baali [1] result, when the descent condition (3) does not hold with \(k\) replaced by \(k + 1\), the search direction (4) in the CG methods used to be reset to that of the steepest descent by letting \(\beta_k = 0\). However, the author shows that the sufficient descent condition

\[
d_{k+1}^T g_{k+1} = -\sigma \|g_{k+1}\|^2
\]

holds, for a certain positive constant \(c\), if the steplength \(\alpha_k\) is chosen inexactly such that the strong Wolfe conditions

\[
f_{k+1} \leq f_k + \rho \alpha_k g_k^T d_k, \quad |g_{k+1}^T d_k| \leq -\sigma g_k^T d_k
\]

are satisfied for \(0 < \rho < \sigma < \frac{1}{2}\). Using these conditions, Al-Baali also shows that the FR method is globally convergent. This first practical convergence result has been extended to the interval \(0 \leq \beta_k \leq \beta_k^{FR}\) and later to the wider interval

\[
-\beta_k^{FR} \leq \beta_k \leq \beta_k^{FR}
\]

by Touati-Ahmed and Storey [23] and Gilbert and Nocedal [14], respectively.

To enforce the sufficient descent condition (7) for a preset value of the constant \(c\) and any line search technique, Al-Baali [2] replaces the search direction (4) by the scaled direction

\[
d_{k+1} = -g_{k+1} + \xi_k \beta_k d_k,
\]

where \(\xi_k\) is a scalar. This scaled direction would be close to the steepest descent one if \(\xi_k\) is chosen sufficiently close to zero. Hence, by continuity, the corresponding class of scaled CG (ScCG) methods would satisfy both the descent and convergence properties that the steepest descent method has. However, to maintain the features of the CG parameter \(\beta_k\) (or nearly so), it is preferable to choose the scaled parameter \(\xi_k\) as close as possible to one. Here, we choose the largest value of \(\xi_k\) that satisfies condition (7) and possibly condition (9) with \(\beta_k\) replaced by \(\xi_k \beta_k\). As a special case, substituting \(\beta_k = \beta_k^{FR}\) into (10), we get a search direction which defines a class of scaled FR methods. This class (referred to as ScFR) with \(\xi_k \in [0, 1]\) ensure that condition (9) holds with \(\beta_k = \xi_k \beta_k^{FR}\) so that it maintains the sufficient descent and global convergence properties that the FR method has for the strong Wolfe conditions (8) with \(\sigma < \frac{1}{2}\). The ScFR class of methods has been suggested by Al-Baali [2], analyzed with proof of the global convergence property by Al-Saidi et al. [5] and studied its behaviour for certain choices of \(\xi_k\) by Al-Saidi and Al-Baali [4]. The authors in [4] reported
encouraging numerical results and recommended their best ScFR method (referred to as ScFR2) which defines the search direction by (10) with \( \beta_k \) given by (5) and the scaling parameter by

\[
\xi_{k}^{F2R} = \begin{cases} 
\frac{(1-c)\|g_k\|^2}{\sigma d_k^T g_k} & \text{if } d_k^T g_{k+1} > (1-c)\|g_k\|^2, \\
1 & \text{otherwise}. 
\end{cases}
\]

(11)

\( \text{a}) : \) In a manner similar to that introduced to the FR method to obtain the ScFR class of methods, we replace the FR formula (5) by that of Dai and Yuan [9], given by

\[
\beta_k^{DY} = \frac{\|g_k\|^2}{d_k^T g_k},
\]

(12)

where

\[
\gamma_k = g_{k+1} = g_k.
\]

(13)

Formula (12) remains positive if the curvature condition \( d_k^T g_k > 0 \) holds (which is guaranteed for all our numerical results). Thus, substituting (12) into (10), we obtain the scaled DY (referred to as ScDY) direction.

\( \text{b}) : \) In Section II, we show that the ScDY class of methods with \( \xi_k \in [0, 1] \) maintains the sufficient descent property that the DY method has for the Wolfe conditions

\[
f_{k+1} \leq f_k + \rho \alpha_g k d_k^T d_k, \quad g_{k+1} \leq \sigma g_k d_k
\]

(14)

with \( 0 < \rho < 1/2 \) and \( \rho < \sigma < 1 \). We also enforce this property for any line search technique with sufficiently small values of the scaling parameter \( \xi_k \). In addition, that section defines some choices for \( \xi_k \). In Section III, we illustrate the way of introducing the quasi-Newton feature to \( \xi_k \) for the DY method as considered by Al-Saidi et al. [5] for the scaled CG methods. Section IV shows that the proposed class of ScDY methods maintains the global convergence property that the DY method has when the Wolfe conditions are employed with \( \sigma < 1 \). In Section V, we study the behavior of the proposed ScDY methods by applying them to a set of standard test problems. It is shown that the proposed scaling technique improves the performance of the DY method substantially in several cases. Finally, Section VI concludes the paper.

II. The SCDY Class of Methods

We now define the scaled DY (ScDY) class of methods as suggested by Al-Baali [2] for the CG class of methods. The author replaces the search direction (4), for \( k \geq 1 \), by (10) where \( 0 < \xi_k \leq 1 \) is the scaling parameter, and maintains \( d_1 = -g_1 \). Since the value of \( \xi_k \) reduces this direction to that of the steepest descent, it follows by continuity that the ScDY class of methods, with values of \( \xi_k \) sufficiently close to zero, has the sufficient descent and global convergence properties that the steepest descent method has.

\( \text{a}) : \) In this paper, we extend the analysis of Al-Saidi and Al-Baali [4] for the ScFR class to ScDY. We first define some values of \( \xi_k \) that enforce the sufficient descent condition (7) as in [4], for the DY formula (12). Using this formula for \( \beta_k \) in (10) which can be substituted into (7), it follows that

\[
\xi_k(d_k^T g_{k+1}) \leq (1-c)d_k^T \gamma_k,
\]

(15)

where \( 0 < c \leq 1 \). It is assumed that the Wolfe conditions (14) are employed so that the curvature condition \( d_k^T \gamma_k > 0 \) is satisfied. Thus, it follows that (15) holds if its left hand side is nonpositive or either \( \xi_k \) or \( d_k^T g_{k+1} \) is sufficiently close to zero.

This paper assumes that the CG parameter \( \beta_k \) is given by the DY formula (12), unless otherwise stated, and defines \( \xi_k \) such that (15) holds. For convenience, we start with the following result of Al-Saidi et al. [5]:

Theorem 2.1: Consider the ScDY class of methods, defined by (2), (12) and (10) and assume that the search direction (10) with \( \xi_k = 1 \) satisfies the sufficient descent condition (7). Then, this condition remains satisfied for \( 0 \leq \xi_k \leq 1 \).

If condition (15) holds with \( \xi_k = 1 \), then we use this value so that the DY search direction is unchanged. Otherwise, we choose \( \xi_k \) such that condition (15) holds with equality. Thus, we choose for the ScDY method,

\[
\xi_k = \begin{cases} 
\frac{(1-c)d_k^T g_{k+1}}{d_k^T g_k} & \text{if } d_k^T g_{k+1} > (1-c)d_k^T \gamma_k, \\
1 & \text{otherwise.}
\end{cases}
\]

(16)

We note that \( \xi_k = 1 \) is used when \( \beta_k(d_k^T g_{k+1}) \leq 0 \), although in this case (15) holds for any value of \( \xi_k > 0 \). To obtain the least change in the DY parameter, \( c \) should be chosen close to zero. We also note that usually \( \xi_k = 1 \) for a fairly accurate line search (\( \sigma \leq 0.1 \)). Thus, we consider the outline of the method as in Algorithm ScDY. We note that Algorithm ScDY is reduced to the normal DY method if \( \xi_k = 1 \) is used for all \( k \). If in Step 2 exact line searches (or nearly so) are employed for all \( k \), then Algorithm ScDY (by Step 5) is reduced to the normal DY method. In practice, we observed that the above scaling technique improves the performance of the DY method substantially in several cases.

Algorithm ScDY

Step 1: Given \( \varepsilon > 0 \), \( 0 < c \leq 1 \) and \( x_1 \in \mathbb{R}^n \), let \( d_1 = -g_1 \) and set \( k = 1 \)

Step 2: Compute a positive step length \( \alpha_k \) which satisfies the strong Wolfe conditions (8)

Step 3: Compute a new point by \( (2) \)

Step 4: Compute CG parameter \( \beta_k \) by (12)

Step 5: Define \( \xi_k \) (e.g., by (16))

Step 6: Compute a new search direction by \( (10) \)

Step 7: Set \( k := k + 1 \) and go to Step 1

We now consider the other choices for \( \xi_k \) as suggested for a general SCG by Al-Saidi et al. [5]. It is important to realize that the choice of \( \xi^2_k \) in (16) is independent of the line search technique. However, if the strong Wolfe conditions (8) are employed, Al-Saidi et al. suggest replacing the first case of (16) by a smaller value to obtain

\[
\xi^3_k = \begin{cases} 
\frac{(1-c)\|g_k\|^2}{\sigma d_k^T g_k} & \text{if } d_k^T g_{k+1} > (1-c)d_k^T \gamma_k, \\
1 & \text{otherwise.}
\end{cases}
\]

(17)

which satisfies the sufficient descent condition (15) with \( \xi_k = \xi_k^3 \).

Similar to [4], since \( d_k g_{k+1} \leq \|d_k\|\|g_{k+1}\| \), we derive the choice of Al-Baali [2] for the DY parameter by the following approach. The first case of (16) can be reduced to obtain

\[
\xi^4_k = \begin{cases} 
\frac{(1-c)d_k^T g_{k+1}}{\|d_k\|\|g_{k+1}\|} & \text{if } d_k^T g_{k+1} > (1-c)d_k^T \gamma_k, \\
1 & \text{otherwise.}
\end{cases}
\]

(18)
III. QUASI-NEWTON FEATURES

In a manner similar to that given by Al-Saidi et al. [5] and essentially by Al-Baali [2], for defining the search direction close to that of the quasi-Newton direction, we define the scaling parameter related to the ScDY direction as follows. Let the scaled search direction (10) with \( \beta_k \) given by (12) be written as follows

\[
d_{k+1} = -H_{k+1}(\xi_k)g_{k+1},
\]

where

\[
H_{k+1}(\xi_k) = I - \xi_k \frac{d_k g_k^T}{d_k^T g_k} \xi_k
\]

is the inverse Hessian approximation and \( I \) is the identity matrix. Since the quasi-Newton condition \( H_{k+1}(\xi_k)g_k = \alpha_k d_k = 0 \) cannot be satisfied, in general, the author defines \( \xi_k \) by solving the subproblem

\[
\min_{\xi} \|H_{k+1}(\xi)g_k - \alpha_k d_k\| = \|g_k - \alpha_k d_k - \xi g_{k+1}^T g_k \|.
\]

Hence, the solution can be written as follows

\[
\xi_k^q = \frac{d_k^T g_k (\gamma_k - \alpha_k d_k) + g_{k+1}^T g_k d_k}{\gamma_k d_k^T g_k}.
\]

(19)

To maintain the search direction different from that of the steepest descent, the values of \( \xi_k^q \) should be bounded away from zero (say, \( \xi_k^q \geq \bar{c} > 0 \)). Since, in addition, the first case of (16), (17) and (18), which define \( \xi_k^i \), for \( i = 1, 2, 3 \), respectively, enforce the sufficient descent condition (15) with equality, replacing \( \xi_k^i \), for \( i = 1, 2, 3 \), by a smaller value keeps inequality (15) fulfilled. Thus, we replace the acceptable \( \xi_k^i \) (that defined in the previous section) by

\[
\xi_k^q = \min(\xi_k^i, \max(\xi_k^q, \bar{c})),
\]

(20)

for \( i = 1, 2, 3 \). In practice, employing the strong Wolfe conditions with \( \sigma = 0.1 \), we observed that \( \xi_k^q \) works better than the other two choices.

It is worth noting that the above choices for \( \xi_k^q \), \( i = 1, 2, 3 \) also enforce the sufficient descent condition. Thus, Al-Saidi et al. [5] extend Theorem 2.1 to the following one which we state here, for completeness.

**Theorem 3.1:** Consider the ScDY class of methods, defined by (2), (12) and (10), and let \( \xi_k = \xi_k^i \) or \( \xi_k = \xi_k^q \), for \( i = 1, 2, 3 \). Then, the sufficient descent condition (7) holds for all \( k \geq 1 \).

**Proof:** It is obvious for \( k = 1 \). For \( k \geq 1 \), \( \xi_k = \xi_k^i, 1 \leq i \leq 3 \), are chosen such that inequality (15) holds which is equivalent to the sufficient descent condition (7). Since \( \xi_k^{qi} \leq \xi_k^{q}, 1 \leq i \leq 3 \), inequalities (15) remains satisfied. Hence, condition (15) holds true for \( k \geq 1 \).

IV. GLOBAL CONVERGENCE PROPERTY

We state the result of Al-Saidi et al. [5] of imposing the global convergence property

\[
\lim_{k \to \infty} \inf_{k} \|g_k\| = 0
\]

(21)

to the ScDY methods that we proposed in the previous sections such that the sufficient descent condition (7) holds for all \( k \geq 1 \). Therefore, we first state the following standard assumptions on the objective function.

**Assumption 4.1:**

1) The level set \( \mathcal{L} = \{x : f(x) \leq f(x_1)\} \) is bounded.

2) In some neighborhood \( \mathcal{N} \) of \( \mathcal{L} \), the objective function \( f \) is continuously differentiable.

3) The gradient \( g(x) \) is Lipschitz continuous, that is there exists a positive constant \( L \) such that

\[
\|g(x) - g(\tilde{x})\| \leq L\|x - \tilde{x}\| \quad \text{for all } x, \tilde{x} \in \mathcal{N}.
\]

(22)

In some cases, we also assume that the steplength \( \alpha_k \) is chosen such that the strong Wolfe conditions (8) hold which implies the Wolfe conditions. Dai and Yuan [1] show that the CG method defined by (2), (4) and

\[
0 \leq \beta_k \leq \beta_k^{DY},
\]

(23)

is globally convergent if the weak Wolfe conditions (14) is employed. Thus, in a manner similar to that of ScFR methods, we state the following convergence result for the ScDY methods.

**Theorem 4.1:** Suppose Assumptions 4.1 holds and let the ScDY class of methods be defined by (2), (10), \( \beta_k = \beta_k^{DY} \) and \( \xi_k = \xi_k^i \) or \( \xi_k = \xi_k^q \), for \( i = 1, 2, 3 \). If the weak Wolfe conditions (14) are employed with \( \sigma < 1 \), then the ScDY methods converge globally in the sense that property (21) holds.

V. NUMERICAL RESULTS

In order to give some idea about the behaviour of the ScDY methods, we describe some numerical results obtained by applying the methods to a set of standard test problems, that is used by Al-Saidi and Al-Baali [4]. The set consists of 46 type of problems with dimension \( n = 2, 4 \) and 100. The names of the 46 test functions are given in Table 1, which have been collected by CUTE [7] and Moré, Garbow, and Hillstrom [15]. All the methods are defined by Algorithm ScDY and differ only in Steps 4 and 5. As for the ScFR methods for the purposes of an accurate comparison, we use in Step 2 for all methods the Matlab line search code of Al-Baali (which essentially complies with a similar Fortran code of Fletcher) which computes a value of the steplength \( \alpha_k \) that satisfies the strong Wolfe conditions (8) as described in [3] and [12]). For Step 0, as in [4], we let \( \epsilon = 10^{-6} \) and \( c = 0.001 \). The run is stopped not only when \( \|g_k\| \leq 10^{-6} \), as required in Step 1, but also if the number of iterations reaches 10^5. Therefore, we consider the following algorithms:

- **DY:** the usual DY method, defined by Algorithm ScDY with \( \rho = 10^{-4} \) and \( \sigma = 0.1 \) in Step 2 and \( \xi_k = 1 \) in Step 5.
- **ScDYi:** same as DY except that in Step 5, \( \xi_k = \xi_k^i, i = 1, 2, 3 \), defined by (16), (17) and (18), respectively.
- **ScDYqi:** same as ScDYi except that in Step 5, \( \xi_k^i, i = 1, 2, 3, \) is replaced by \( \xi_k^q \), defined by (19) and (20) with \( \bar{c} = 0.001 \).
Table 1: List of Test Functions

<table>
<thead>
<tr>
<th>No.</th>
<th>Function’s Name</th>
<th>N</th>
<th>d</th>
<th>Function’s Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Quadratic function of Al-Baali</td>
<td>24</td>
<td>1</td>
<td>Truncus</td>
</tr>
<tr>
<td>2</td>
<td>Extended White and Holst</td>
<td>25</td>
<td>2</td>
<td>Zerli</td>
</tr>
<tr>
<td>3</td>
<td>Extended Rosenbrok</td>
<td>26</td>
<td>100</td>
<td>Shallow</td>
</tr>
<tr>
<td>4</td>
<td>Extended Frenelstein and Roth</td>
<td>27</td>
<td>100</td>
<td>Generalized Quartic</td>
</tr>
<tr>
<td>5</td>
<td>Extended Beale</td>
<td>28</td>
<td>100</td>
<td>Axis Parallel hyper-ellipsoid</td>
</tr>
<tr>
<td>6</td>
<td>Extended Wood</td>
<td>29</td>
<td>2</td>
<td>Less</td>
</tr>
<tr>
<td>7</td>
<td>Extended R.</td>
<td>30</td>
<td>100</td>
<td>Generalized Tridiagonal 1</td>
</tr>
<tr>
<td>8</td>
<td>Extended Generalized Tridiagonal 1</td>
<td>31</td>
<td>100</td>
<td>Generalized Tridiagonal 2</td>
</tr>
<tr>
<td>9</td>
<td>Extended Generalized Diagonal 1</td>
<td>32</td>
<td>100</td>
<td>POWER</td>
</tr>
<tr>
<td>10</td>
<td>Extended Denschnib</td>
<td>33</td>
<td>100</td>
<td>Quadratic QP1</td>
</tr>
<tr>
<td>11</td>
<td>Extended Hiebert</td>
<td>34</td>
<td>100</td>
<td>Extended quadratic penalty QP2</td>
</tr>
<tr>
<td>12</td>
<td>FLETCHER</td>
<td>35</td>
<td>100</td>
<td>Dixon and Price</td>
</tr>
<tr>
<td>13</td>
<td>Extended Powell</td>
<td>36</td>
<td>4</td>
<td>Quadratic</td>
</tr>
<tr>
<td>14</td>
<td>Extended Noncomp</td>
<td>37</td>
<td>100</td>
<td>Diagonal4</td>
</tr>
<tr>
<td>15</td>
<td>Extended Denschnib</td>
<td>38</td>
<td>4</td>
<td>Cubic</td>
</tr>
<tr>
<td>16</td>
<td>Extended quadratic penalty QP1</td>
<td>39</td>
<td>100</td>
<td>Schonert Stelitz</td>
</tr>
<tr>
<td>17</td>
<td>Extended Penalty</td>
<td>40</td>
<td>100</td>
<td>Sphere</td>
</tr>
<tr>
<td>18</td>
<td>De Jong’s</td>
<td>41</td>
<td>100</td>
<td>Sum Squares</td>
</tr>
<tr>
<td>19</td>
<td>Extended Denschnib</td>
<td>42</td>
<td>4</td>
<td>Powell</td>
</tr>
<tr>
<td>20</td>
<td>Extended Maratos</td>
<td>43</td>
<td>100</td>
<td>Qing</td>
</tr>
<tr>
<td>21</td>
<td>Extended Cabe</td>
<td>44</td>
<td>100</td>
<td>Generalized PSC1</td>
</tr>
<tr>
<td>22</td>
<td>Three bump function</td>
<td>45</td>
<td>100</td>
<td>Perturbed Quadratic</td>
</tr>
<tr>
<td>23</td>
<td>Booth</td>
<td>46</td>
<td>100</td>
<td>Diagonal 2</td>
</tr>
</tbody>
</table>

We used the useful performance profiles tool of Dolan and More [11], which compares some solvers on a set of problems in terms of the number of line searches, function evaluations and gradient evaluations as well as the CPU time in seconds, required to solve the problems. For convenience, we repeat the definition here as follows. The performance profile $P_M(\tau)$, $\tau \geq 0$, is defined by the formula

$$P_M(\tau) = \frac{\text{number of problems where } \log_2 n \leq \tau}{\text{total number of problems}}$$

where $\tau_{P_M}$ is the performance ratio of the number of line search, function evaluations and gradient evaluations (or the time) required to solve problem $p$ by method $M$ to the lowest number of line search, function evaluations and gradient evaluations (or the time) required to solve problem $p$. The ratio $\tau_{P_M}$ is set to infinity (or some large number) if method $M$ fails to solve problem $p$. The values of $P_M(\tau)$ at $\tau = 0$ gives the percentage of test problems for which the method $M$ is best and the value for $\tau$ large enough is the percentage of test problems that the M method can solve. Thus, a solver with high values of $P_M(\tau)$ or one that located at the top right of the figure performs better than the lower one.

$a)$ : We now show that each of the above classes of the scaled DY methods is reduced to a single method if the value of $c$ is sufficiently small.

**Theorem 5.1**: If the strong Wolfe conditions (8) hold with $d_{k+1}^{T}g_k > 0$ and $c$ is chosen such that

$$c \leq \frac{1}{1 + \sigma},$$

then

$$\xi_k^i = 1, \quad \xi_k^{qi} = \min(1, \max(\xi^q_k, \tilde{c})), \quad \xi_k^{q2} = 1,$$

for $i = 1, 2, 3$.

**Proof:** It follows from the definition of $\gamma_k$ by (13), the second condition in (8) and the assumption $d_{k+1}^{T}g_k > 0$ that

$$\frac{d_{k+1}^{T}g_k}{d_{k}^{T}g_k} = 1 - \frac{d_{k}^{T}g_k}{d_{k+1}^{T}g_k} \geq 1 + \frac{1}{\sigma}. \quad (26)$$

Since the curvature condition $d_{k}^{T}g_k > 0$ holds, (26) and (24) yield

$$\frac{d_{k+1}^{T}g_k}{d_{k}^{T}g_k} \leq \frac{\sigma}{\sigma + 1} = 1 - \frac{1}{\sigma} \leq 1 - c. \quad (27)$$

Hence,

$$d_{k}^{T}g_k \leq (1 - c)d_{k}^{T}g_k$$

which yields by (16), (17), (18) and (20) that the two equations in (2) hold for all $i$.

This result shows that condition (24) holds for our choice of $\sigma = 0.1$ and $c = 0.001$. Hence, it reduces the ScDY$i$ algorithms to the DY method, for all $i$, and the ScDY$qi$ algorithms to that defined in (2). Therefore, we consider the comparison of ScDY$q3$, DY and ScFR2 as shown in Figures

![Figure 1](image1.png)  
Comparison among DY, ScDY$q3$ and ScFR2 for NLS.

![Figure 2](image2.png)  
Comparison among DY, ScDY$q3$ and ScFR2 for NF.

![Figure 3](image3.png)  
Comparison among DY, ScDY$q3$ and ScFR2 for NG.
We show that introducing a simple scaling technique to the well-known Dai-Yuan method maintains its global convergence property and improves its performance substantially in several cases. It is also shown that the proposed scaled DY methods perform better than the best scaled FR method of Al-Saidi and Al-Baali [4]. However, further experiments would be desirable.

VI. CONCLUSION

We show that introducing a simple scaling technique to the well-known Dai-Yuan method maintains its global convergence property and improves its performance substantially in several cases. It is also shown that the proposed scaled DY methods perform better than the best scaled FR method of Al-Saidi and Al-Baali [4]. However, further experiments would be desirable.

REFERENCES

6


