Journal of Soft Computing and Artificial Intelligence
Research Article 2(2): pp:108-119 (2021)
Received: November 10, 2021 Accepted: December 1, 2021
JSCAI
ISSN: 2717-8226

Spectral Fletcher (CD) Algorithm for Solving Fuzzy Non-Linear Equations

[1]Mezher M. Abed , [2]Ufuk Öztürk , and [3]Hisham M. Khudhur

[1] Department Mathematics, Faculty of Science, University of Çankırı Karatekin, 18100 Çankırı, Turkey.
[2] Department Mathematics, Faculty of Science, University of Çankırı Karatekin, 18100 Çankırı, Turkey.
[3] Department Mathematics, College of Computer Science and Mathematics, University of Mosul, Mosul, Iraq.

Corresponding author: First A. Author (e-mail: hisham892020@uomosul.edu.iq).

ABSTRACT A conjugate gradient method is a powerful tool for solving large-scale miniaturization issues, with applications in arithmetic, chemistry, physics, engineering, medicine, and other fields. In this paper, we introduce a new spectral conjugate gradient algorithm, whose derivation is based on the Fletcher (CD) and Newton algorithms based on the solely coupling condition, which is introduced in this study. The significance of the research is in identifying a suitable algorithm. Because the Buckley and Qu methods are ineffectual in solving all types of ambiguous equations, and the conjugate gradient approach does not require a Hessian matrix (second partial derivatives of functions) in the solution, it is used to solve all types of ambiguous equations. The suggested method's descent property is demonstrated as long as the $\alpha_k$step size matches the strong Wolfe conditions. In many cases, numerical findings demonstrate that the novel technique is more efficient in solving nonlinear fuzzy equations than Fletcher (CD) algorithm.

KEYWORDS: algorithms, CG, Fletcher, fuzzy, numerical.

1. INTRODUCTION

Nonlinear equations can be solved using iterative approaches, such as

$$F(x) = 0 \qquad (1)$$

It has received a lot of attention in recent years. The concept of fuzzy numbers, as well as the arithmetic operations that can be performed on them, were first proposed and researched by Zadeh [1]. One of the most popular applications for calculating fuzzy numbers is non-linear equations, the parameters of which are fully or partially represented by fuzzy numbers. [2]–[4]. Buckley and Qu's standard analytical procedures [5]–[8] are only suitable for the linear and quadratic cases of nonlinear equations. Cannot be used to solve equations like

I- $\mathrm{Q}y^3 + \mathrm{V}y^2 - \mathrm{L}y = \Theta,$
II- $\Theta e^y - \mathrm{K}\mathrm{y} = \Lambda,$
III- $\Phi \mathrm{y}\csc(y) + \Upsilon y = \Omega,$
IV- $\Psi y^5 - \Omega \cot(y) = \Phi.$

where $y, \mathrm{A}, \mathrm{B}, \mathrm{H}, \Theta, \mathrm{E}, \Lambda, \Upsilon, \Omega, \Psi$, and Φ are fuzzy numbers. S. Abbasbandy and B. Asady employed Newton's method to solve a fuzzy nonlinear problem in 2004[9]. Amirah Ramli, Mohd Lazim Abdullah, and Mustafa Mamat used Quasi Newton's method to solve a fuzzy nonlinear problem in 2010 [10]. Due to the following disadvantages of the methodologies, this method is not particularly useful in practice: It necessitates the storage of the $n \times n$ matrix $[H_i]$, the computation of the elements of the matrix $[H_i]$ becomes extremely difficult and often impossible, the inversion of the matrix $[H_i]$ at each step, and the assessment of the amount $[H_i]^{-1}\nabla f_i$ at each step. The strategy is ineffective for challenges with a complex objective function with a large number of variables because of these flaws. The Steepest descent method for solving fuzzy nonlinear equations was developed by S. Abbasbandy and A. Jafarian [11]. which they published in 2004. However, because the sharpest descent direction is a local feature, this strategy is weak and inefficient in most applications. Hisham and Khalil created two conjugate gradient (CG) approach for solving fuzzy nonlinear equations in 202 [12], [13]. Mezher M. Abed, Ufuk ztürk, and Hisham M. Khudhur published Spectral CG Algorithm for Solving Fuzzy Non-linear Equations in 2022 [14]. Conjugate gradient methods have the drawback of being ineffective in some cases. Therefore, we need to use the new spectral gradient algorithm to find the roots of these equations, in this paper; We use one of the spectral conjugate gradient algorithms, because this algorithm is very efficient and fast in finding the roots of equations and also has global convergence. This paper is divided into six parts, the first part is a general introduction, previous studies, and the importance of the paper, the second part is the basics of arithmetic operations for fuzzy numbers, the third part is an explanation of the conjugate gradient algorithms, the fourth part is New Proposed Algorithm SCD-CG, the fifth part is the computational results and comparisons, and sixth part is the conclusions in addition to acknowledgments, and references.

2. PRELIMINARIES

We've gone through some basic definitions and arithmetic operations for fuzzy numbers in this section. For more information, we refer interested readers to [15].

Definition (2.1). A fuzzy number is defined as a set $j: \mathbb{R} \to I = [0,1]$ that meets the conditions listed below [16].

a. j denotes a semi-continuous upper boundary.
b. $j(x) = 0$ outside some range $[r, t]$.
c. there exist $p, q \in \mathbb{R}$ such that $r \le p \le q \le t$ and
  i. $j(x)$ is increasing monotonically on $[r, p]$.
  ii. $j(x)$ is decreasing monotonically on $[q, t]$.
  iii. $j(x) = 1,\ p \le x \le q$.

Definition (2.2). j: $\mathbb{R} \to I = [0,1]$ in parametric form

refer to the pair $(\underline{j}, \bar{j})$of $\underline{j}(\mu), \bar{j}(\mu)$, $0 \le \mu \le 1$

satisfying [16], [17],

(1) $\underline{j}(\mu)$ is a monotonically bounded growing left continuous function.

(2) $\bar{j}(\mu)$ is a monotonically bounded growing right continuous function.

(3) $\underline{j}(\mu) \le \bar{j}(\mu)$.

Definition (2.3). A classical Fuzzy number h refers to the Triangular number $j = (p, q, r)$ given as follows in equation (2)

$$j(x) = \begin{cases} \dfrac{(x-p)}{(r-p)}, & p \le x \le r \\ \dfrac{(x-q)}{(r-q)}, & r \le x \le q \end{cases} \tag{2}$$

with j(x) known as the membership function and $r \ne p, r \ne q$ [9]. This function can be written in its parameterized form as follows

$\bar{j}(\mu) = q + (r-q)\mu$
$\underline{j}(\mu) = p + (r-p)\mu$

Assume $TF(\mathbb{R})$ denotes the set of all trapezoidal fuzzy number. The following extension principle can be used to extend the scalar multiplication and addition operations to fuzzy numbers [9].

Let $= \left(\underline{j}(\mu), \overline{j}(\mu)\right)$, $k = \left(\underline{k}(\mu), \overline{k}(\mu)\right)$ with $w > 0$, the addition $(j + k)$ and multiplication by scalar w are defined as

$$(\overline{j+k})(\mu) = \overline{j}(\mu) + \overline{k}(\mu)$$

$$\left(\underline{j+k}\right)(\mu) = \underline{j}(\mu) + \underline{k}(\mu)$$

$$(\overline{wj})(\mu) = w\overline{j}(\mu)$$

$$\left(\underline{wj}\right)(\mu) = w\underline{j}(\mu).$$

3. CONJUGATE GRADIENT (CG) ALGORITHMS

The non-linear conjugate gradient (CG) scheme has the form in equation (3)

$$x_{k+1} = x_k + \alpha_k d_k, \quad k \geq 1 \tag{3}$$

Where x_1 is an initial paint, α_k is a step–length and α_k step-size that satisfy the standard Wolfe conditions in equation (4), and (5) [18], [19]

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \delta \alpha_k g_k^T d_k \tag{4}$$

$$d_k^T g(x_k + \alpha_k d_k) \geq \sigma d_k^T g_k \tag{5}$$

or strong Wolfe conditions in equation (6), and (7) [20]–[30],[31]

$$f(x_k + \alpha_k d_k) \leq f(x) + \delta \alpha_k g_k^T d_k \tag{6}$$

$$|d_k^T g(x_k + \alpha_k d_k)| \leq -\sigma d_k^T g_k \tag{7}$$

$$d_{k+1} = \begin{cases} -g_1, & k = 1 \\ -g_{k+1} + \beta_k d_k, & k \geq 1 \end{cases} \tag{8}$$

Different β_{k+1} will determine different CG methods. Some famous formula for β_{k+1}as follows:

The Fletcher and Reeves (FR) [32], Fletcher (CD) [33], Polak and Ribiere (PRP) [34], Hestenes-Stiefel (HS) [35], Dai-Yuan (DY) [36], Hisham- Khalil (KH) [12], and β is scalar.

$$\beta^{FR} = \frac{\| g_{k+1} \|^2}{\| g_k \|^2} \qquad \beta^{CD} = \frac{-\| g_{k+1} \|^2}{g_k^T d_k}$$

$$\beta^{HS} = \frac{g_{k+1}^T y_k}{y_k^T d_k} \qquad \beta^{PRP} = \frac{g_{k+1}^T y_k}{\| g_k \|^2}$$

$$\beta^{DY} = \frac{\| g_{k+1} \|^2}{y_k^T d_k} \qquad \beta^{KH} = \frac{\|g_{k+1}\|_1^2}{\|g_k\|_1^2},$$

Where $g_k = \nabla f(x_k)$, and let $y_k = g_{k+1} - g_k$.

4. NEW PROPOSED ALGORITHM SCD-CG

The search direction for the Fletcher (CD) conjugate gradient (CD-CG) method is obtained by $d_1 = -g_1$ and

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

The methods Fletcher and Reeves (FR) [32], Hisham- Khalil (KH) [12], Dai and Yuan (DY) [36], and Fletcher (CD) [33], have strong global convergence properties, but these methods have modest practical performance. While the methods Hestenes and Stiefel (HS) [35], and Polak and Ribiere (PRP) [34], are not always convergent, but they often have good computational properties see [37]. In order to obtain conjugate gradient methods with computational efficiency and good convergence properties, basically, the algorithms are found to avoid failure and to improve the performance of classical conjugated gradient algorithms. In order to accelerate the Fletcher (CD-CG) method, we use equation as follows

Let $\gamma_{k+1} = 1 + \mu_{k+1}$

Where γ_{k+1} and μ_{k+1} are two parameters, then

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

$$d_{k+1} = -(1 + \mu_{k+1}) g_{k+1} - \frac{\| g_{k+1} \|^2}{g_k^T d_k} d_k \qquad (9)$$

We incorporate the second-order information to the search direction in (9) by assuming, the direction in (9) is parallel to the Newton direction i.e

$$-G_{k+1}^{-1} g_{k+1} = -g_{k+1} - \mu_{k+1} g_{k+1} - \frac{\| g_{k+1} \|^2}{g_k^T d_k} d_k \qquad (10)$$

Where G_{k+1}^{-1} is the inverse Hessian matrix. Now suppose G_{k+1}^{-1} is symmetric $\left(G_{k+1}^{-1} = \left(G_{k+1}^{-1}\right)^T\right)$, positive definite and satisfies the Quasi-Newton condition i.e

$$G_{k+1}^{-1} y_k = s_k \qquad (11)$$

Where $s_k = x_{k+1} - x_k$ multiply both sides of (10) by and considering G_{k+1}^{-1} be symmetric and positive definite, then

$$-\left(G_{k+1}^{-1} y_k\right)^T g_{k+1} = -y_k^T g_{k+1} - \mu_{k+1} y_k^T g_{k+1} - \frac{\| g_{k+1} \|^2}{g_k^T d_k} y_k^T d_k$$

Use the relation given in (11) to get

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

Divide both sides in the above equation by $y_k^T d_k$ then

$-\frac{s_k^T g_{k+1}}{y_k^T d_k} = -\beta_{k+1}^{HS} - \mu_{k+1}\beta_{k+1}^{HS} + \beta_{k+1}^{CD}$ (where $\beta_{k+1}^{HS} = \frac{g_{k+1}^T y_k}{y_k^T d_k}$, and $\beta_{k+1}^{CD} = -\frac{\|g_{k+1}\|^2}{g_k^T d_k}$)

Or

$$-\frac{s_k^T g_{k+1}}{y_k^T d_k} = -(1 + \mu_{k+1})\beta_{k+1}^{HS} + \beta_{k+1}^{CD}$$

Or

$$(1 + \mu_{k+1})\beta_{k+1}^{HS} = \beta_{k+1}^{CD} + \frac{s_k^T g_{k+1}}{y_k^T d_k}$$

$$\therefore \gamma_{k+1} = \frac{\beta_{k+1}^{CD}}{\beta_{k+1}^{HS}} + \frac{s_k^T g_{k+1}}{y_k^T g_{k+1}} \tag{12}$$

Use this value for the γ_{k+1} in (9) then

$$d_{k+1} = -\gamma_{k+1} g_{k+1} + \beta_{k+1}^{CD} d_k \tag{13}$$

We call the algorithm defined in (12) and (13) as spectral Fletcher (SCD-CG) algorithm and we summarize it as the following algorithm SCD-CG.

Algorithm (SCD-CG):

step(1) : Initialization : select $x_1 \in R^n, \varepsilon > 0$ is a small positive real value and compute

$$d_1 = -g_1, \alpha_1 = 1/\|g_1\| \text{ and } k = 1$$

step(2) : Test for convergence: If $\|g_k\| \leq \varepsilon$ break x_k is optimal solution else go to step(3).

step(3) : Line search : calculateα_k satisfying the strong wolf conditions (6) (7) and up to date the variable $x_{k+1} = x_k + \alpha_k d_k$, calculate $f_{k+1}, g_{k+1}, y_k \text{ and } s_k$.

step(4) : Direction calculation : calculate γ_{k+1} from (12), if $\gamma_{k+1} \geq 1$ or $\gamma_{k+1} \leq 0$ set $\gamma_{k+1} = 1$ and $d_{k+1} = -\gamma_{k+1} g_{k+1} + \beta_{k+1}^{CD} d_k$ then $d_{k+1} = -\gamma_{k+1} g_{k+1}$ else $d_{k+1} = d$ and $\alpha_{k+1} = \alpha_k * \|d_k\|/\|d_{k+1}\|, k = k + 1$ go to step(2).

4.1 Descent property

In this part, we prove that our algorithm determines in equation (12) and (13) generates the descent direction for each iteration according to the following theorem.

Theorem:

consider the algorithm defined in equation (3) where d_k computed from (12) and (13). Assume that the step size α_k satisfies the strong Wolfe conditions (6) and (7). Then the search directions d_k generated by the SCD-CG algorithm are descent for all k provided $y_k^T g_{k+1} > 0$.

Proof

The prove is by indication, for k=1, $d_1 = -g_1 \rightarrow g_1^T d_1 < 0,$.

Now suppose $g_k^T d_k < 0$ or $g_k^T s_k < 0$ $s_k = \alpha_k d_k$ then for $k+1$ we have

$$d_{k+1} = -\left(\frac{\beta_{k+1}^{CD}}{\beta_{k+1}^{HS}} + \frac{s_k^T g_{k+1}}{y_k^T g_{k+1}}\right) g_{k+1} + \beta_{k+1}^{CD} d_k$$

$$g_{k+1}^T d_{k+1} = -\left(\frac{\beta_{k+1}^{CD}}{\beta_{k+1}^{HS}} + \frac{s_k^T g_{k+1}}{y_k^T g_{k+1}}\right) g_{k+1}^T g_{k+1} - \frac{\alpha_k \|g_{k+1}\|^2}{\alpha_k g_k^T s_k} g_{k+1}^T s_k$$

$$g_{k+1}^T d_{k+1} = -\left(\frac{\beta_{k+1}^{CD}}{\beta_{k+1}^{HS}} + \frac{s_k^T g_{k+1}}{y_k^T g_{k+1}}\right) g_{k+1}^T g_{k+1} - \frac{\|g_{k+1}\|^2}{g_k^T s_k} g_{k+1}^T s_k$$

Divide both sides by $\frac{\|g_{k+1}\|^2}{g_k^T s_k}$

$$\frac{g_k^T s_k}{\|g_{k+1}\|^2} d_{k+1}^T g_{k+1} = -\frac{g_k^T s_k}{\|g_{k+1}\|^2}\left(\frac{\beta_{k+1}^{CD}}{\beta_{k+1}^{HS}} + \frac{s_k^T g_{k+1}}{y_k^T g_{k+1}}\right) g_{k+1}^T g_{k+1} - g_{k+1}^T s_k$$

$$\frac{g_k^T s_k}{\|g_{k+1}\|^2} d_{k+1}^T g_{k+1} \le -\frac{g_k^T s_k}{\|g_{k+1}\|^2}\left(-\frac{y_k^T s_k}{\alpha_k g_{k+1}^T y_k}\frac{\alpha_k \|g_{k+1}\|^2}{g_k^T s_k} + \frac{s_k^T g_{k+1}}{y_k^T g_{k+1}}\right) g_{k+1}^T g_{k+1} - s_k^T g_{k+1}$$

$$\frac{g_k^T s_k}{\|g_{k+1}\|^2} d_{k+1}^T g_{k+1} \le -\frac{g_k^T s_k}{\|g_{k+1}\|^2}\left(-\frac{y_k^T s_k}{g_{k+1}^T y_k}\frac{\|g_{k+1}\|^2}{g_k^T s_k} + \frac{s_k^T g_{k+1}}{y_k^T g_{k+1}}\right) g_{k+1}^T g_{k+1} - s_k^T g_{k+1}$$

$$\frac{g_k^T s_k}{\|g_{k+1}\|^2} d_{k+1}^T g_{k+1} \le -\left(-\frac{y_k^T s_k}{g_{k+1}^T y_k} + \frac{g_k^T s_k}{\|g_{k+1}\|^2}\frac{s_k^T g_{k+1}}{y_k^T g_{k+1}}\right) g_{k+1}^T g_{k+1} - s_k^T g_{k+1}$$

$$\frac{g_k^T s_k}{\|g_{k+1}\|^2} d_{k+1}^T g_{k+1} \le -\left(-\frac{y_k^T s_k}{g_{k+1}^T y_k} g_{k+1}^T g_{k+1} + g_k^T s_k \frac{s_k^T g_{k+1}}{y_k^T g_{k+1}}\right) - s_k^T g_{k+1}$$

$$\frac{g_k^T s_k}{\|g_{k+1}\|^2} d_{k+1}^T g_{k+1} \le -\left(-\frac{y_k^T s_k}{g_{k+1}^T y_k} g_{k+1}^T g_{k+1} + g_k^T s_k \frac{s_k^T y_k}{y_k^T g_{k+1}}\right) - s_k^T y_k$$

$\because s_k^T g_{k+1} = s_k^T g_{k+1} - s_k^T g_k + s_k^T g_k = s_k^T y_k + s_k^T g_k < s_k^T y_k$

$$\frac{g_k^T s_k}{\|g_{k+1}\|^2} d_{k+1}^T g_{k+1} \le -\left(-\frac{y_k^T s_k}{g_{k+1}^T y_k} g_{k+1}^T g_{k+1} + g_k^T s_k \frac{s_k^T y_k}{y_k^T g_{k+1}}\right) - s_k^T y_k$$

$$\frac{g_k^T s_k}{\|g_{k+1}\|^2} d_{k+1}^T g_{k+1} \le -\frac{1}{y_k^T g_{k+1}}\left(-y_k^T s_k g_{k+1}^T g_{k+1} + g_k^T s_k s_k^T y_k\right) - s_k^T y_k$$

$$\frac{g_k^T s_k}{\|g_{k+1}\|^2} d_{k+1}^T g_{k+1} \le -\frac{s_k^T y_k}{y_k^T g_{k+1}}\left(-g_{k+1}^T g_{k+1} + g_k^T s_k - y_k^T g_{k+1}\right)$$

$$d_{k+1}^T g_{k+1} \le -\frac{s_k^T y_k}{y_k^T g_{k+1}}\left(-g_{k+1}^T g_{k+1} + g_k^T s_k - y_k^T g_{k+1}\right)\frac{\|g_{k+1}\|^2}{g_k^T s_k}$$

$$d_{k+1}^T g_{k+1} \le -\frac{s_k^T y_k}{y_k^T g_{k+1}}\left(-g_{k+1}^T g_{k+1} + g_k^T s_k - g_{k+1}^T g_{k+1} + g_k^T g_k\right)\frac{\|g_{k+1}\|^2}{g_k^T s_k}$$

$$d_{k+1}^T g_{k+1} \le -\frac{s_k^T y_k}{y_k^T g_{k+1}}\left(-2g_{k+1}^T g_{k+1} + g_k^T s_k + g_k^T g_k\right)\frac{\|g_{k+1}\|^2}{g_k^T s_k}$$

$$d_{k+1}^T g_{k+1} \le -\frac{s_k^T y_k}{y_k^T g_{k+1}}\left(-\frac{2\left(g_{k+1}^T g_{k+1}\right)^2}{g_k^T s_k} + g_k^T s_k\frac{\|g_{k+1}\|^2}{g_k^T s_k} + g_k^T g_k\frac{\|g_{k+1}\|^2}{g_k^T s_k}\right)$$

$$d_{k+1}^T g_{k+1} \le -\frac{s_k^T y_k}{y_k^T g_{k+1}}\left(-\frac{2\left(g_{k+1}^T g_{k+1}\right)^2}{g_k^T s_k} + \|g_{k+1}\|^2 + g_k^T g_k\frac{\|g_{k+1}\|^2}{g_k^T s_k}\right)$$

$$d_{k+1}^T g_{k+1} \le -\frac{s_k^T y_k}{y_k^T g_{k+1}}\|g_{k+1}\|^2\left(-\frac{2\|g_{k+1}\|^2}{g_k^T s_k} + 1 + \frac{g_k^T g_k}{g_k^T s_k}\right)$$

$$d_{k+1}^T g_{k+1} \le -\frac{s_k^T y_k}{y_k^T g_{k+1}}\|g_{k+1}\|^2\left(1 + \frac{g_k^T g_k}{g_k^T s_k} - \frac{2\|g_{k+1}\|^2}{g_k^T s_k}\right)$$

Use the Cuchy-Schuarz inequality then

$$d_{k+1}^T g_{k+1} \le -\frac{s_k^T y_k}{y_k^T g_{k+1}}\|g_{k+1}\|^2\left(1 + \frac{g_k^T g_k}{g_k^T s_k} - \frac{2\|g_{k+1}\|^2}{g_k^T s_k}\right)$$

$$= -\frac{s_k^T y_k}{y_k^T g_{k+1}}\|g_{k+1}\|^2\left(1 + \frac{g_k^T g_k}{g_k^T s_k} - 2\beta_{k+1}^{CD}\right)$$

$$d_{k+1}^T g_{k+1} \le -\frac{s_k^T y_k}{y_k^T g_{k+1}}\|g_{k+1}\|^2\left(1 + \frac{g_k^T g_k}{g_k^T s_k} - 2\beta_{k+1}^{CD}\right)$$

$s_k^T y_k > 0$ by Wolfe condition and $y_k^T g_{k+1} > 0$ by assumption

$\therefore d_{k+1}^T g_{k+1} < 0$

The proof is complete.

5. COMPARISONS AND COMPUTATIONAL RESULTS

This part presents the performance of Matlab 2021b implementation with hp laptop Ram 4GB, and hard 500GB of our new spectral conjugate gradient algorithm (SCD-CG) on a set of Fuzzy Nonlinear Equations taken from [7]. We compared the performance of this algorithm against the Fletcher algorithm (CD), and the algorithms were compared with the number of iterations, the optimal value of the function, and the optimal value of the variables as shown in Table (1). These algorithms are implemented with standard Wolfe conditions with $\rho = 0.1$ and $\sigma = 0.11$ where the initial step-size $\alpha = \frac{1}{\|g_k\|}$ and initial guess for other iterations i.e. $(k > 1)$ is $\alpha_k = \alpha_{k-1}\frac{\|d_{k-1}\|}{d_k}$. In the all cases the stopping criterion is $\|g_{k+1}\| \le 10^{-6}$ and maximum number of iteration is 2000. Our comparison includes the following.

1- It:- Number of Iterations
2- x-best:- optimal Variable
3- f-best:- optimal Function Value

Tables (1), show the details of the results for (SCD-CG) algorithms versus FR-CG algorithm. The numerical solutions were also plotted in Figs (1), (2), and (3).

Example 1: Consider the fuzzy nonlinear equation [7]

$(3,4,5)x^2 + (1,2,3)x = (1,2,3)$

Without any loss of generality, assume that x is positive, and then the parametric form of this equation is as follows:

$$\begin{cases}(3+r)\underline{x}^2(r) + (1+r)\underline{x}(r) - (1+r) = 0,\\ (5-r)\overline{x}^2(r) + (3-r)\overline{x}(r) - (3-r) = 0.\end{cases}$$

The above system needs initial values as follows. For $r = 1$

$$\begin{cases}4\underline{x}^2(1) + 2\underline{x}(1) - 2 = 0,\\ 4\overline{x}^2(1) + 2\overline{x}(1) - 2 = 0,\end{cases}$$

For $r = 0$

$$\begin{cases}3\underline{x}^2(0) + \underline{x}(0) - 1 = 0,\\ 5\overline{x}^2(0) + \overline{x}(0) - 3 = 0\end{cases}$$

With initial values

$x_0 = (\underline{x}(0), \underline{x}(1), \overline{x}(1), \overline{x}(0)) = (0.434, 0.5, 0.5, 0.681).$

Example 2: Consider the fuzzy nonlinear equation [7]

$(4,6,8)x^2 + (2,3,4)\,x - (8,12,16) = (5,6,7)$

Without any loss of generality, assume that x is positive, and then the parametric form of this equation is as follows:

$$\begin{cases}(4+2r)\underline{x}^2(r) + (2+r)\underline{x}(r) - (3+3r) = 0,\\ (8-2r)\overline{x}^2(r) + (4-r)\overline{x}(r) - (9-3r) = 0.\end{cases}$$

The above system needs initial values as follows. For $r = 1$

$$\begin{cases}6\underline{x}^2(1) + 3\underline{x}(1) - 6 = 0,\\ 6\overline{x}^2(1) + 3\overline{x}(1) - 6 = 0,\end{cases}$$

For $r = 0$

$$\begin{cases}4\underline{x}^2(0) + 2\underline{x}(0) - 3 = 0,\\ 8\overline{x}^2(0) + 4\overline{x}(0) - 9 = 0,\end{cases}$$

With initial values

$x_0 = (\underline{x}(0), \underline{x}(1), \overline{x}(1), \overline{x}(0)) = (0.651, 0.7808, 0.7808, 0.8397).$

Example 3: Consider the fuzzy nonlinear equation [7]

$(1,2,3)x^3 + (2,3,4)x^2 + (3,4,5) = (5,8,13)$

Without any loss of generality, assume that x is positive, and then the parametric form of this equation is as follows:

$$\begin{cases}(1+r)\underline{x}^3(r) + (2+r)\underline{x}^2(r) - (2+2r) = 0,\\ (3-r)\overline{x}^3(r) + (4-r)\overline{x}^2(r) - (8-4r) = 0.\end{cases}$$

The above system needs initial values as follows. For $r = 1$

$$\begin{cases}2\underline{x}^3(1) + 3\underline{x}^2(1) - 4 = 0,\\ 2\overline{x}^3(1) + 3\overline{x}^2(1) - 4 = 0,\end{cases}$$

For $r = 0$

$$\begin{cases}\underline{x}^3(0) + 2\underline{x}^2(0) - 2 = 0,\\ 3\overline{x}^3(0) + 4\overline{x}^2(0) - 8 = 0.\end{cases}$$

With initial values

$x_0 = (\underline{x}(0), \underline{x}(1), \overline{x}(1), \overline{x}(0)) = (0.76, 0.91, 0.91, 1.06)$.

Table (1) of numerical results for examples above

Examples	*CD ALGORITHM*			*SCD-CG ALGORITHM*		
	It	*x-best*	*f-best*	*It*	*x-best*	*f-best*
1	8	0.4343 0.5000 0.5000 0.5307	8.1709e-014	9	0.4343 0.5000 0.5000 0.5307	2.7933e-020
2	14	0.6514 0.7808 0.7808 0.8397	5.9506e-010	9	0.6514 0.7808 0.7808 0.8397	1.8868e-011
3	135	0.8393 0.9108 0.9108 1.0564	1.4978e-008	10	0.8393 0.9108 0.9108 1.0564	1.6416e-011

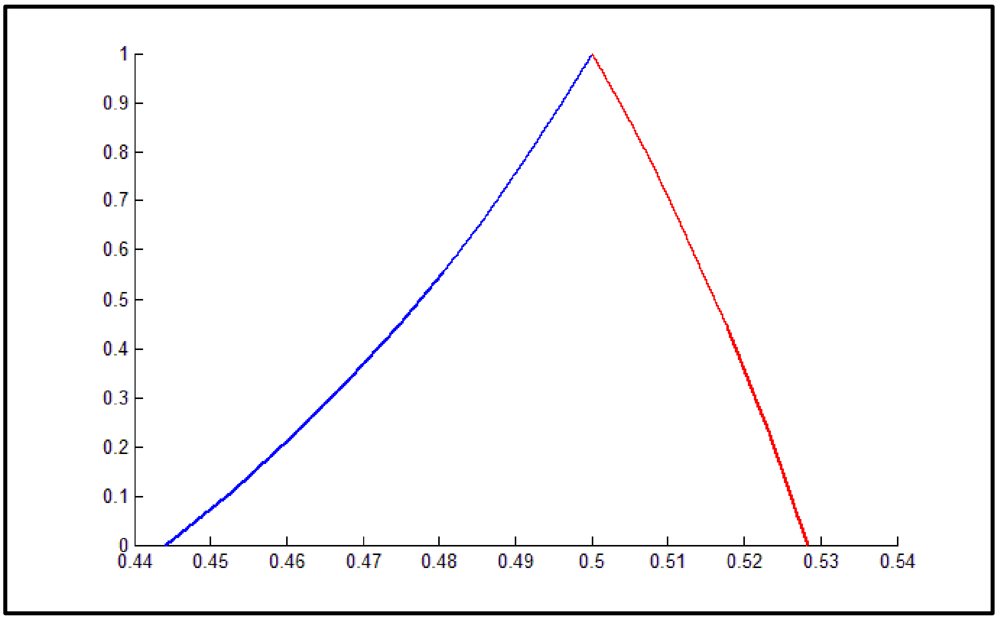

Figure 1. Drawing Solution to Example 1. using SCD-CG Algorithm

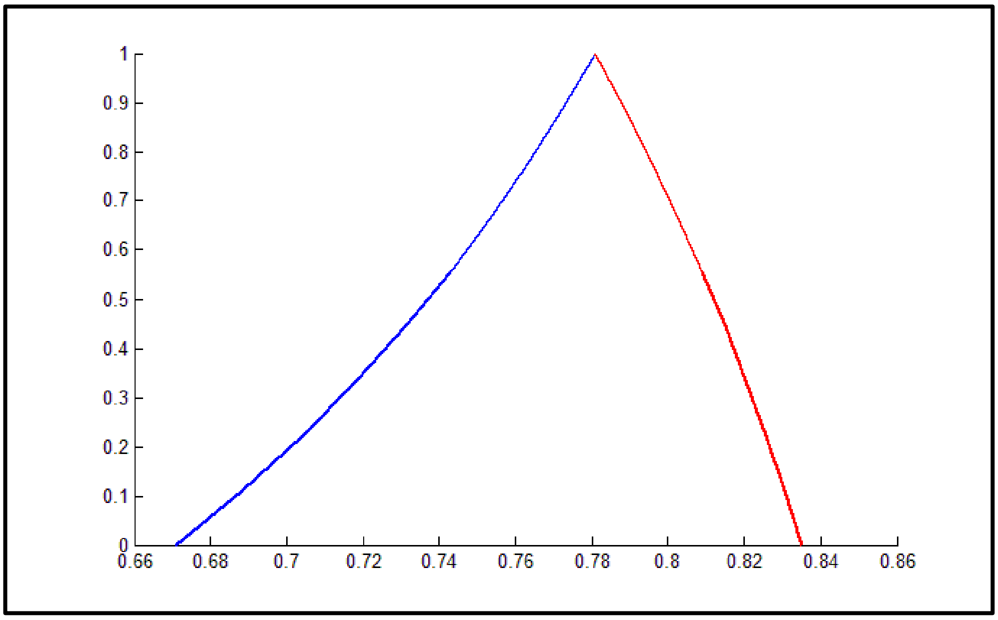

Figure 2: Drawing Solution to Example 2. using SCD-CG Algorithm

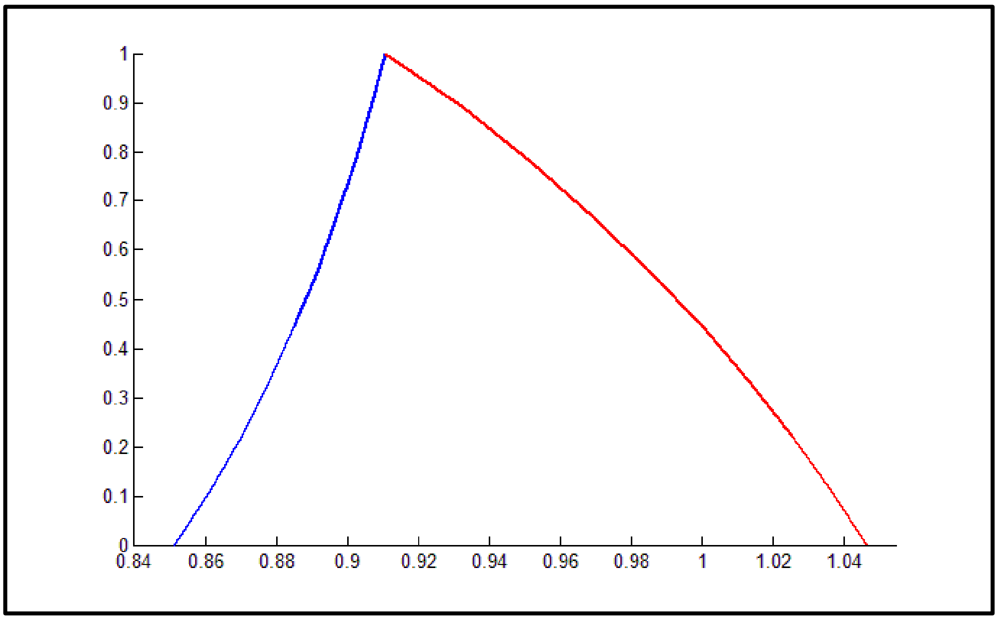

Figure 3: Drawing Solution to Example 3. using SCD-CG Algorithm

6. CONCLUSIONS

The main purpose of this article was to apply the spectral conjugate gradient algorithm that can be used to solve fuzzy nonlinear equations as an alternative to the usual analytical technique. The ambiguous nonlinear problem was transformed into a parametric formula and then solved using spectral conjugate gradient algorithms. The numerical results showed that the Spectral Fletcher algorithm (SCD-CG) performs very encouragingly in all the tested problems that have been solved, and this algorithm can also be used in other fields such as artificial neural networks, fuzzy neural networks, swarming algorithms, as well as in solving optimization problems.

ACKNOWLEDGMENTS

I would like to thank the College of Computer Science and Mathematics at Mosul University, and the Institute of Science and Technology in Çankiri Karatekin University for providing resources that have helped me increase the quality of my work.

REFERENCES

[1] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning-III," *Inf. Sci. (Ny).*, vol. 9, no. 1, pp. 43–80, 1975, doi: 10.1016/0020-0255(75)90017-1.

[2] R. Badard, "The law of large numbers for fuzzy processes and the estimation problem," *Inf. Sci. (Ny).*, vol. 28, no. 3, 1982, doi: 10.1016/0020-0255(82)90046-9.

[3] P. Diamond, "Fuzzy least squares," *Inf. Sci. (Ny).*, vol. 46, no. 3, pp. 141–157, 1988.

[4] M. Friedman, M. Ming, and A. Kandel, "Fuzzy linear systems," *Fuzzy sets Syst.*, vol. 96, no. 2, pp. 201–209, 1998.

[5] J. J. Buckley and Y. Qu, "Solving linear and quadratic fuzzy equations," *Fuzzy Sets Syst.*, vol. 38, no. 1, pp. 43–59, 1990, doi: 10.1016/0165-0114(90)90099-R.

[6] J. J. Buckley and Y. Qu, "On using α-cuts to evaluate fuzzy equations," *Fuzzy Sets and Systems*, vol. 43, no. 1. Elsevier, p. 125, 1991, doi: 10.1016/0165-0114(91)90026-M.

[7] J. J. Buckley and Y. Qu, "Solving fuzzy equations: A new solution concept," *Fuzzy Sets Syst.*, vol. 39, no. 3, pp. 291–301, 1991, doi: 10.1016/0165-0114(91)90099-C.

[8] J. J. Buckley and Y. Qu, "Solving systems of linear fuzzy equations," *Fuzzy Sets Syst.*, vol. 43, no. 1, pp. 33–43, 1991, doi: 10.1016/0165-0114(91)90019-M.

[9] S. Abbasbandy and B. Asady, "Newton's method for solving fuzzy nonlinear equations," *Appl. Math. Comput.*, vol. 159, no. 2, pp. 349–356, Jul. 2004, doi: 10.1016/j.amc.2003.10.048.

[10] M. Mamat, A. Ramli, and M. L. Abdullah, "Broyden's method for solving fuzzy nonlinear equations," *Adv. Fuzzy Syst.*, 2010, doi: 10.1155/2010/763270.

[11] S. Abbasbandy and A. Jafarian, "Steepest descent method for solving fuzzy nonlinear equations," *Appl. Math. Comput.*, vol. 174, no. 1, pp. 669–675, Jul. 2006, doi: 10.1016/j.amc.2005.04.092.

[12] H. M. Khudhur and K. K. Abbo, "A New Type of Conjugate Gradient Technique for Solving Fuzzy Nonlinear Algebraic Equations," in *Journal of Physics: Conference Series*, 2021, vol. 1879, no. 2, doi: 10.1088/1742-6596/1879/2/022111.

[13] H. M. Khudhur and K. K. Abbo, "New hybrid of Conjugate Gradient Technique for Solving Fuzzy Nonlinear Equations," *J. Soft Comput. Artif. Intell.*, vol. 2, no. 1, pp. 1–8, 2021.

[14] M. M. Abed, U. Öztürk, and H. Khudhur, "Spectral CG Algorithm for Solving Fuzzy Non-linear Equations," *Iraqi J. Comput. Sci. Math.*, vol. 3, no. 1, 2022.

[15] L. A. Zadeh, "Fuzzy sets," *Inf. Control*, vol. 8, no. 3, pp. 338–353, 1965.

[16] D. J. Dubois, *Fuzzy sets and systems: theory and applications*, vol. 144. Academic press, 1980.

[17] H. Zimmermann, "Fuzzy set theory and its applications," *Int. Ser. Manag.*, 1991.

[18] B. Hassan, H. Jabbar, and A. Al-Bayati, "A new class of nonlinear conjugate gradient method for solving unconstrained minimization problems," 2019, doi: 10.1109/ICCISTA.2019.8830657.

[19] B. A. Hassan, "A new type of quasi-newton updating formulas based on the new quasi-newton equation," *Numer. Algebr. Control Optim.*, vol. 10, no. 2, pp. 227–235, 2020, doi: 10.3934/naco.2019049.

[20] WOLFE P, "CONVERGENCE CONDITIONS FOR ASCENT METHODS," *SIAM Rev.*, vol. 11, no. 2, pp. 226–235, 1969, doi: 10.1137/1011036.

[21] H. M. Azzam, H. N. Jabbar, and K. K. Abo, "Four–Term Conjugate Gradient (CG) Method Based on Pure Conjugacy Condition for Unconstrained Optimization," *Kirkuk Univ. Journal-Scientific Stud.*, vol. 13, no. 2, pp. 101–113, 2018, doi: 10.32894/kujss.2018.145720.

[22] Y. A. Laylani, K. K. Abbo, and H. M. Khudhur, "Training feed forward neural network with modified Fletcher-Reeves method," *J. Multidiscip. Model. Optim.*, vol. 1, no. 1, pp. 14–22, 2018, [Online]. Available: http://dergipark.gov.tr/jmmo/issue/38716/392124#article_cite.

[23] A. S. Ahmed, "Optimization Methods For Learning Artificial Neural Networks," University of Mosul, 2018.

[24] A. S. Ahmed, H. M. Khudhur, and M. S. Najmuldeen, "A new parameter in three-term conjugate gradient algorithms for unconstrained optimization," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 23, no. 1, 2021, doi: 10.11591/ijeecs.v23.i1.pp338-344.

[25] H. M. Khudhur, "Numerical and analytical study of some descent algorithms to solve unconstrained Optimization problems," University of Mosul, 2015.

[26] K. K. Abbo and H. M. Khudhur, "New A hybrid conjugate gradient Fletcher-Reeves and Polak-Ribiere algorithm for unconstrained optimization," *Tikrit J. Pure Sci.*, vol. 21, no. 1, pp. 124–129, 2015.

[27] K. K. Abbo and H. M. Khudhur, "New A hybrid Hestenes-Stiefel and Dai-Yuan conjugate gradient algorithms for unconstrained optimization," *Tikrit J. Pure Sci.*, vol. 21, no. 1, pp. 118–123, 2015.

[28] K. K. ABBO, Y. A. Laylani, and H. M. Khudhur, "A NEW SPECTRAL CONJUGATE GRADIENT ALGORITHM FOR UNCONSTRAINED OPTIMIZATION," *Int. J. Math. Comput. Appl. Res.*, vol. 8, pp. 1–9, 2018.

[29] K. K. Abbo, Y. A. Laylani, and H. M. Khudhur, "Proposed new Scaled conjugate gradient algorithm for Unconstrained Optimization," *Int. J. Enhanc. Res. Sci. Technol. Eng.*, vol. 5, no. 7, 2016.

[30] Z. M. Abdullah, M. Hameed, M. K. Hisham, and M. A. Khaleel, "Modified new conjugate gradient method for Unconstrained Optimization," *Tikrit J. Pure Sci.*, vol. 24, no. 5, pp. 86–90, 2019.

[31] H. M. Khudhur and K. K. Abbo, "A New Conjugate Gradient Method for Learning Fuzzy Neural Networks," *J. Multidiscip. Model. Optim.*, vol. 3, no. 2, pp. 57–69, 2020.

[32] R. Fletcher and C. M. Reeves, "Function minimization by conjugate gradients," *Comput. J.*, vol. 7, no. 2, pp. 149–154, 1964, doi: 10.1093/comjnl/7.2.149.

[33] C. Witzgall and R. Fletcher, "Practical Methods of Optimization.," *Math. Comput.*, vol. 53, no. 188, p. 768, 1989, doi: 10.2307/2008742.

[34] E. Polak and G. Ribiere, "Note sur la convergence de méthodes de directions conjuguées," *Rev. française d'informatique Rech. opérationnelle. Série rouge*, vol. 3, no. 16, pp. 35–43, 1969, doi: 10.1051/m2an/196903r100351.

[35] M. R. Hestenes and E. Stiefel, "Methods of conjugate gradients for solving linear systems," *J. Res. Natl. Bur. Stand. (1934).*, vol. 49, no. 6, p. 409, 1952, doi: 10.6028/jres.049.044.

[36] Y. H. Dai and Y. Yuan, "A nonlinear conjugate gradient method with a strong global convergence property," *SIAM J. Optim.*, vol. 10, no. 1, pp. 177–182, 1999, doi: 10.1137/S1052623497318992.

[37] Z.-J. Shi and J. Guo, "A new algorithm of nonlinear conjugate gradient method with strong convergence," *Comput. Appl. Math.*, vol. 27, pp. 93–106, 2008.