A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF ÇANKIRI KARATEKİN UNIVERSITY

NUMERICAL AND ANALYTICAL STUDY OF SOME DESCENT ALGORITHMS TO SOLVING FUZZY OPTIMİZATION PROBLEMS

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
MATHEMATICS

BY

MOHAMMED MSALLAM MUSTAFA MUSTAFA

ÇANKIRI

2023

NUMERICAL AND ANALYTICAL STUDY OF SOME DESCENT ALGORITHMS TO SOLVING FUZZY OPTIMİZATION PROBLEMS

By Mohammed Msallam Mustafa MUSTAFA

June 2023

We certify that we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Mathematics.

Advisor **:** Assoc. Prof. Dr. Gonca DURMAZ GÜNGÖR
Co-Advisor **:** Prof. Dr. Khalil Khudhur ABBO

Examining Committee Members:

Chairman : Assoc. Prof. Dr. Faruk KARAASLAN
Department of Mathematics
Çankırı Karatekin University

Member **:** Assoc. Prof. Dr. Gonca DURMAZ GÜNGÖR
Department of Mathematics
Çankırı Karatekin University

Member **:** Asst. Prof. Dr. İlker GENÇTÜRK
Department of Mathematics
Kırıkkale University

Approved for the Graduate School of Natural and Applied Sciences

Prof. Dr. Hamit ALYAR
Director of Graduate School

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Mohammed Msallam Mustafa MUSTAFA

ABSTRACT

NUMERICAL AND ANALYTICAL STUDY OF SOME DESCENT ALGORITHMS TO SOLVING FUZZY OPTIMIZATION PROBLEMS

Mohammed Msallam Mustafa MUSTAFA

Master of Science in Mathematics

Advisor: Assoc. Prof. Dr. Gonca DURMAZ GÜNGÖR

Co-Advisor: Prof. Dr. Khalil Khudhur ABBO

June 2023

Fuzzy nonlinear equations are crucial to many aspects of daily life. Fuzzy nonlinear equations are crucial in the robotics and smart car industries, as well as in engineering, scientific challenges, mathematics, chemistry, physics, biology, machine learning, deep learning, regression classification, computer science, programming, and artificial intelligence. The development of conjugate gradient techniques to address fuzzy nonlinear optimization issues is the focus of this thesis. The purpose of employing conjugate gradient algorithms is to identify the minimal endpoint of equations or test functions, hence developing conjugate gradient algorithms to address fuzzy nonlinear optimization issues is of interest to us in this thesis. The general introduction to optimization, optimization conditions, and unrestricted optimization were all thoroughly covered in the first chapter. Following that, we briefly discussed different descent methods, including the steep descent method, conjugate gradient method, and quasi-Newton method. We also dealt with fuzzy logic and basic definitions in it .

In the second chapter, we'll show how to solve unconstrained fuzzy optimization problems with a conjugate gradient scheme Conjugate gradient approaches are one of the most commonly used nonlinear optimization methods. They are effective solutions for a broad variety of small to medium-sized optimization problems, in particular those in which the Hessian is challenging to calculate. There are a variety of conjugate gradient methods; however, we apply the well-known conjugate gradient approach to fuzzy optimization problems in this chapter. This method has a range of advantages. First, only

first-derivatives knowledge can be used to find a parameter beta. Second, the proposed algorithm is successful for problems where the Hessian is difficult to compute, the suggested scheme performs well. In order to handle fuzzy optimization problems, we created the KM1 method based on the KH algorithm, and the results were encouraging when compared to the KH algorithm. as the proposed KM1 algorithm's general descent and convergence properties were demonstrated under some hypotheses. We also suggested the KM2 algorithm, which was based on the KH method.

In the third chapter, and the numerical outcomes were contrasted with those of the KH algorithm. as the suggested KM1 algorithm's general descent and convergence features were demonstrated under several hypotheses. The created methods were quite successful at resolving issues with fuzzy nonlinear optimization.

Finally, in the fourth chapter, it deals in detail with the conclusions of the message and future works.

2023, 56 pages

Keywords: Gradient, Fuzzy, Numerical, Nonliear, Optimization

ÖZET

NUMERICAL AND ANALYTICAL STUDY OF SOME DESCENT ALGORITHMS TO SOLVING FUZZY OPTIMIZATION PROBLEMS

Muhammed Msallam Mustafa MUSTAFA
Matematik, Yüksek Lisans
Tez Danışmanı: Doç. Dr. Gonca DURMAZ GÜNGÖR
Eş Danışman: Prof. Dr. Halil Hudhur ABBO
Haziran 2023

Bulanık lineer olmayan denklemler, mühendislik, bilimsel zorluklar, matematik, kimya, fizik, biyoloji, makine öğrenimi, derin öğrenme, regresyon sınıflandırması, bilgisayar bilimi programlama, tıp ve askeri alanda yapay zekâ gibi günlük hayatımızın tüm alanlarında önemli bir rol oynar. Mühendislik endüstrileri, robotlar ve akıllı arabalar bulanık doğrusal olmayan denklemler çok önemli bir rol oynamaktadır. Bu tezde, bulanık doğrusal olmayan optimizasyon problemlerini çözmek için eşlenik gradyan algoritmaları geliştirmekle ilgileniyoruz. Bu tezde, eşlenik gradyan algoritmalarını kullanmanın amacı denklemlerin veya test fonksiyonlarının minimum son noktasını bulmak olduğundan, bulanık doğrusal olmayan optimizasyon problemlerini çözmek için eşlenik gradyan algoritmaları geliştirmekle ilgileniyoruz. İlk bölümde, optimizasyona genel giriş, optimizasyon koşulları ve kısıtlamasız optimizasyondaki temel tanımlar ayrıntılı olarak ele alındı ve ardından dik regresyon yöntemi, eşlenik gradyan yöntemi ve yarı Newtonian gibi bazı oran yöntemlerine değinildi. Yöntem. Bulanık mantık ve içindeki temel tanımları da ele aldık.

İkinci bölümde, eşlenik gradyan şemasını kullanarak kısıtlanmamış bulanık optimizasyon problemlerinin nasıl çözüleceğini göstereceğiz. Eşlenik gradyan yaklaşımları, en popüler doğrusal olmayan optimizasyon yöntemleri arasındadır. Özellikle Hessian hesaplamasının zor olduğu çok çeşitli küçük ve orta ölçekli optimizasyon problemleri için etkili çözümlerdir. Çeşitli eşlenik gradyan yöntemleri vardır; ancak, iyi bilinen eşlenik gradyan yaklaşımını bu bölümde bulanık optimizasyon problemlerine uyguluyoruz. Bu

yöntemin bir dizi avantajı vardır. İlk olarak, beta parametresini bulmak için yalnızca birinci türevlerin bilgisi kullanılabilir. İkinci olarak, önerilen algoritma, Hessian algoritmasının hesaplanmasının zor olduğu sorunları çözer ve önerilen şema iyi çalışır. Bu yüzden KH algoritmasına dayalı bulanık optimizasyon problemlerini çözmek için KM1 algoritmasını geliştirdik ve KH algoritmasına kıyasla cesaret verici sonuçlar elde ettik. Önerilen KM1 algoritmasının kapsamlı regresyon ve yakınsama özellikleri bazı hipotezlerde kanıtlanmıştır.

Üçüncü bölümde KH algoritmasına dayalı KM2 algoritmasını da önerdik ve sayısal sonuçlar KH algoritması ile karşılaştırıldı. Önerilen KM1 algoritmasının kapsamlı regresyon ve yakınsama özellikleri bazı hipotezlerde kanıtlanmıştır. Geliştirilen algoritmalar, bulanık doğrusal olmayan optimizasyon problemlerinin çözümünde oldukça verimli olmuştur.

Son olarak dördüncü bölümde ise risalenin vardığı sonuçlar ve bundan sonra yapılacak çalışmalar ayrıntılı olarak ele alınmıştır.

2023, 56 sayfa

Anahtar Kelimeler: Gradient, Fuzzy, Numerical, Nonliear, Optimization

PREFACE AND ACKNOWLEDGEMENTS

At the end of my master's academic journey, I would like to express my deep gratitude and appreciation to everyone who contributed to this path that I have overcome. I cannot thank my wonderful mentors Assoc. Prof. Dr. Gonca DURMAZ GÜNGÖR and Prof. Dr. Halil Hudhur ABBO who have given me guidance and support throughout this scientific journey.

Last, but certainly not least, I would like to give special thanks to my family and friends who have stood by me and supported me with love and dedication and a special thanks to my friend Dr. Hisham Mohammed.

Thank you everyone and I will carry this beautiful memory with me for the rest of my life.

Mohammed Msallam Mustafa MUSTAFA
Çankırı-2023

LIST OF FIGURES

LIST OF TABLE

CONTENTS

1. INTRODUCTION

Early in the nineteenth century, Euler and Lagrange used the calculus of variations to provide the first formal framework for the idea of optimization. Most of the significant advances in modern optimization theory and practice over the last 60 years may be attributed to the introduction of linear programming (LP) in the 1940s, which greatly expanded the area of the study (Khudhur 2015).

The term "mathematical programming" is a mishmash that has been around since the 1940s, long before the word "programming" came to be synonymous with computer software. It is more correct to call the process of optimizing a system "programming," rather than the more common phrase "optimization. The present context in which this term is employed is narrower than its original connotation, which incorporated connotations of algorithmic development and examination. In this particular sense, the term retains its customary definition. It is noteworthy that the word has undergone a semantic shift over time, and its current meaning no longer encompasses its earlier associations. Nonetheless, comprehending the term's past meanings may facilitate a more nuanced understanding of its significance in contemporary discourse (Nocedal and Wright 2006).

Getting to the goal of reducing a given objective function $f(x)$, where x is an actual n-dimensional vector, may be hard in a number of real-world situations. This is because the goal may be limited in different ways. In order to attain the objective of reducing a certain objective function, it is imperative to take into account the possibility of being subjected to an assortment of limitations.$f(x)$, finding the solution of x, which is a vector in n-dimensions, can prove to be a challenging undertaking in various practical scenarios. The unconstrained situation represents a significant portion of the problem space that is encountered in real-world applications because it is so straightforward to transform many constrained problems into unconstrained ones and to find solutions to them using unconstrained optimization methods; in addition, the resolution of unconstrained subproblems is required for the majority of optimization issues. As a result, we are going to concentrate on a few different approaches to solving the issue without constraints.

$$Minimize f(x), x \in \mathbb{R}^n \quad (1.1)$$

where $f: \mathbb{R}^n \to \mathbb{R}$ is a function of n variables with real-valued arguments that is sufficiently smooth on $\mathbb{R}^n$. The primary objective here is to identify a location x that satisfies the condition that f must be minimized locally $f(x^0) \leq f(x)$ for x near x^0. If $f(x^0) \leq f(x)$ for x near x^0, then x^0 is what's referred to as a "strict local minimizer" of function *f*.

The local minimization issue is not the same as the global minimization problem, which involves locating a global minimizer, which is a point and such that

$$f(x^0) \leq f(x)$$

for all $x \in \mathbb{R}^n$. This work interested with only the local minimization problems. If it is believed that at least twice a function and its first derivatives may be calculated at any position x in the space-time continuum, thereby satisfying the condition of continuous differentiability (Lenir 1981).

The possible resolutions to problem 1.1 may be separated into two different categories according to the features that they have in common with one another. Although there may be some overlap between these categories, search methods and gradient methods may be distinguished from one another. Although there may be some overlap between these categories, search methods and gradient methods can be distinguished from one another. The only thing that searches strategies do is evaluate functions, in contrast to gradient methods, which are reliant on gradient data in the form of the Jacobian gradient vector g. Search approaches do nothing more than evaluate functions. The latter category is made up of the so-called second order techniques, and they all need to make use of the Hessian matrix G in order to function properly. When continuous derivatives of a high enough order are accessible, a function is said to be smooth. This pretty broad category includes the Newton or quasi-Newton methods, great explanations, and other techniques, all of which may be found in the book by (Viviani 1985).

There are two main core types of techniques, each of which requires the assessment of gradients. In the first place are the Conjugate Gradient (CG) techniques, which were developed in 1964 by Fletcher and Reeves. In 1952, Hestenes and Stiefel came up with this strategy in order to solve linear problems. There are several different approaches to CG that are known (Viviani 1985). In general, the conjugate gradient techniques may be used to solve (1.1). More precisely, beginning with an initial point of $x_0 \in \mathbb{R}^n$, the iterations are created as follows:

$$x_{k+1} = x_k + \alpha_k d_k$$

where $d_k \in \mathbb{R}^n$ is the search direction along with the values of function f are reduced and α_k is the sky-size determined by a line search procedure tit is clear up later. The most important prerequisite is that the direction of search, denoted d_k, must be downward-pointing at each iteration of the algorithm

$$\vartheta_k^T d_k < 0 \tag{1.2}$$

which is a very important criterion concerning the effectiveness of an algorithm. In (1.2) $\vartheta \mathrm{g}_k = \nabla \mathrm{f}_k$ is the gradient of f in point x_k gradient of in Point Xiithio. In order to guarantee the global convergence sometimes it is required that the search direction $\boldsymbol{d_k}$ satisfy the sufficientdescent condition.

$$\vartheta_k^T d_k < -c\|g_k\|^2 \tag{1.3}$$

where C is a positive constant. The Quasi-Newton (QN) methods are a second class of gradient-based method that are based on the ability to approximate the curvature of nonlinear functions without explicitly generating the Hessian matrix. Davidon put out the first Quasi-Newton algorithm. He created the DFP updating formula, the first QN-algorithm, in 1959, Fletcher and Powell later made it widespread in 1963 (Powell 1977).

The iterative kinds of optimization algorithms are the ones that are used the most often. They begin with a rough estimate of the value of the variable x and continue by making ever more accurate estimates (which are referred to as "iterates") until, ideally, they arrive at a solution. In order to differentiate the various algorithms, this method is used. The bulk of the strategies make use of the value of the function f, along with maybe its first and second derivatives. While some algorithms only make use of information that is now being acquired in their immediate surroundings, others only make use of information that has been obtained over the course of previous rounds (Nocedal and Wright 2006).

2. PRELIMINARLES

2.1 Optimality Conditions

Suppose that $f: \mathbb{R}^n \to \mathbb{R}$ be a continuous derivative function. The column vector containing the first order partial derivatives $f(x)$ is known as the gradient of $f(x)$

$$\vartheta(x) = \nabla f(x) = \left[\frac{\partial f(x)}{\partial x_1} \frac{\partial f(x)}{\partial x_2} \dots \dots \dots \dots \frac{\partial f(x)}{\partial x_n}\right]^T. \tag{2.1}$$

The Hessian matrix is $\nabla^2 f(x)$ of $f(x)$ are the second order partial derivatives of $f(x)$ used to define the matrix as

$$G = \nabla^2 f(x) = \begin{bmatrix} \frac{\partial^2 f(x)}{\partial {x_1}^2} \frac{\partial^2 f(x)}{\partial x_1 \partial x_2} \dots \dots \dots \dots \frac{\partial^2 f(x)}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(x)}{\partial x_2 \partial x_1} \frac{\partial^2 f(x)}{\partial^2 x_2} \dots \dots \dots \dots \frac{\partial^2 f(x)}{\partial x_2 \partial x_n} \\ \dots \dots \dots \\ \dots \dots \dots \\ \frac{\partial^2 f(x)}{\partial x_n \partial x_1} \frac{\partial^2 f(x)}{\partial x_n \partial x_2} \dots \dots \dots \dots \frac{\partial f(x)}{\partial {x_n}^2} \end{bmatrix}, \tag{2.2}$$

if f is continuously differential then G is symmetric(Witzgall and Fletcher 1989).

Definition 2.1.1 A symmetric matrix G is a positive definite matrix $x \in \mathbb{R}^n$, we have $x^T G_x > 0$ (Sun and Yuan 2006).

Theorem (2.1.1) (First-order Necessary Conditions): Suppose that $f: \mathbb{R}^n \to \mathbb{R}$ have continuous partial derivatives of first order. If x^* is the sole locally existent minimum of $f(x)$, then $\nabla f(x^*) = \vartheta(x^*) = 0$ (Khudhur 2015).

Theorem 2.1.2 (Second-order sufficient conditions): Suppose that $f: \mathbb{R}^n \to \mathbb{R}$ It has partial first and second derivatives x^* is the local minimum of $f(x)$, if

$$\nabla f(x^*) = \vartheta(x^*) = 0$$

and $\nabla^2 f(x)$ is positive definite (Khudhur 2015).

Point x^* is considered to be a stationary point of $f(x)$ if $\nabla f(x^*) = 0$ (Khudhur 2015).

2.2 Basic Definitions

In this section, we will discuss a few of the most basic results, as well as provide a few general definitions When we talk about a technique, we use the word "optimization" when we imply a process that, in some way, requires figuring out the approach that will solve a certain issue in the most effective and time-saving way possible. Mathematically speaking, what has to be done here is to determine the lowest and maximum values of a function that contains n variables, where n may be any positive number that is more than zero. This is the work that needs to be achieved (Kulkarni *et al*. 2017).

Definition 2.2.1 Suppose that the $f: \mathbb{R}^n \to \mathbb{R}$, The aim of the following unconstrained optimization problem, which has to be solved in order to acquire the minimum value of $f(x)$, is to obtain x in such a way that the function $f(x)$ is reduced to its lowest feasible value. This may be accomplished by solving the issue. x^* is the global min of $f(x)$ if

$$f(x^*) \leq f(x) \text{ for all } x \in \mathbb{R}^n \tag{2.3}$$

point x^* is a local min of $f(x)$ if there exists $\varepsilon > 0$ as such

$$f(x^*) \leq f(x), \text{for all } \lambda, \ \|x - x^*\| < \varepsilon \tag{2.4}$$

(Antoniou and Lu 2007).

Definition 2.2.2 The simplest continuous second derivative functions are quadratic ones, and they have the following form:

$$f(x) = \frac{1}{2}x^T Gx + b^T x + c, x \in \mathbb{R}^n. \quad (2.5)$$

Given a positive definite symmetric $n * n$ matrix G, a scalar c and a constant vector b are used in the formulation of a mathematical problem. The derivative with respect to the input variable of the given mathematical function is represented by the gradient. $\vartheta(x) = G_x + \mathrm{b}$, the point at which the function reaches its lowest value is where the minimum is located. x^*, where $\vartheta(x^*)$ is given by $\vartheta(x^*) = G_{x^*} + \mathrm{b} = 0$ (Scales 1985).

Definition 2.2.3 Suppose that the iterative process k leads to the convergence of the sequence $\{x_k\}$ to x^*, and further suppose that a certain condition is met $\|.\|$, there exist scalars ε, ρ and $k > k_0$ such that and $\varepsilon \in (0,1)$

$$\|x_{k+1} - x^*\| \leq \varepsilon \|x_k - x^*\|^P. \quad (2.6)$$

i. If $\rho = 1$ the convergence of sequence $\{x_k\}$ to x^* can be described as linear if the rate of convergence is constant equal 1.

ii. If $\rho = 2$ the convergence of sequence $\{x_k\}$ to x^* can be described as linear if the rate of convergence is constant equal 2.

iii. If $1< \rho < 2$ the convergence of sequence $\{x_k\}$ to x^* can be described as linear if the rate of convergence is constant equal $1< \rho < 2$ (Lenir 1981).

Definition 2.2.4 (Convex set): *A* set $S \subseteq \mathbb{R}^n$ is convex if any straight-line segment joining any two points inside S completely encloses S. Formally, two distinct points in a given context. $x, \mathrm{y} \in \mathrm{S}$ have the same $\lambda x + (1 - \lambda)y \in S, \forall\, \lambda \in [0,1]$ (Rothwell 2017).

Definition 2.2.5 Let $f: S \to \mathbb{R}^1$ be a convex set S that is not empty in $\mathbb{R}^n$. Convex on S is the description of the function f if

$$f(\lambda x_1 + (1-\lambda)x_2) \geq \lambda f(x_1) + (1-\lambda) f(x_2) \tag{2.7}$$

for each $x_1, x_2 \in S$, and for each $\lambda \in (0,1)$. If the above-mentioned inequality holds true as a strict inequality for all unique pairs, then it is considered to be satisfied of x_1 and x_2 in S for all $\lambda \in (0,1)$, then the function f is deemed to be strictly convex over S. Concave on S is how the function f is described if

$$f(\lambda x_1 + (1-\lambda)x_2) \leq \lambda f(x_1) + (1-\lambda) f(x_2) \tag{2.8}$$

for all $x_1, x_2 \in S$, and for each $\lambda \in (0,1)$. S is rigidly concave for the function f. For each different $x_1$and x_2 in S for each $\lambda \in (0,1)$if the inequality mentioned above holds (Khudhur 2015).

Definition 2.2.6 Eigenvalues and Eigenvectors: Assume that A is a square matrix of size n $\times$ n. The terms eigenvalue and eigenvector of A refer to a scalar λ and a non-zero vector u, respectively, that fulfill the equation $Au = \lambda u$. The condition of the matrix $\lambda I - A$ being singular holds equal importance in its possession of an eigenvalue, specifically det $|\lambda I - A| = 0$, whereby I represents the identity matrix of dimension n-by-n (Scales 1985).

2.3 Algorithms for Unconstrained Optimization

All optimization techniques must use iterative optimization algorithms. A local minimal x^* must be given an initial guess x_0 and a series of iterations x_{k+1}. $k = 0,1,2,\ldots$ it is expected that the sequence x_k will eventually converge to x^* as $k \to \infty$. A general technique is to first define a search direction, which is denoted by d_k, and then evaluate the new iteration, which is denoted by x_{k+1}, in terms of the search direction as well as the direction from x_k.

$$x_{k+1} = x_k + \alpha_k d_k \tag{2.9}$$

where the step size is represented by α_k and the distance traveled in the direction d_k from x_k is represented by α_k. Two queries must be addressed in order to produce the iteration step: how is the search direction d_k determined. and how is the step size α_k determined.

The algorithm's output should provide the function's local minimum. Consequently, we are guaranteed to anticipate that the function value at the future iteration will be lower than the function value at the present iteration, or that

$$f(x_{k+1}) \leq f(x_k). \tag{2.10}$$

A vector $d \in \mathbb{R}^n$ is a function's descent direction at $x \in \mathbb{R}^n$, if

$$\nabla f(x)^T d < 0. \tag{2.11}$$

One method is to solve the one-dimensional minimization problem to find the step size α_k at each iteration step

$$\min_{\alpha>0} f(x_k + \alpha_k d_k) \tag{2.12}$$

and select α_k as the minimizer. To put it another way, we choose x_{k+1}such that $f(x_{k+1})$ It can be observed that a certain value represents the lowest possible limit of $f(x_k)$. We find the length of the make d_k. It is through the present cycle of operations x_k. However, this method necessitates addressing at least one-dimensional minimization problems, which can be highly costly in many cases. The process of conducting a line search to identify an appropriate step size α is detailed in the section that follows (Moguerza and Prieto 2003, Sun and Yuan 2006).

2.3.1 Line search procedure

The algorithm of minimization is equipped with a precise line search mechanism, commonly known as ELS if $\vartheta_{k+1}^{T} d_k = 0$, for $k = 0,1,2,3, \ldots$ where d_k the search direction has been established (Steihaug and Hestenes 1982).

In light of a present iteration, it is imperative to consider alternative approaches in order to optimize the outcome x_k and A descent search direction is a crucial factor in optimization algorithms that seeks to minimize a given objective function d_k A function of the step size is established by us α by

$$\vartheta(\alpha) = f(x_k + \alpha_k d_k), \alpha > 0. \tag{2.13}$$

The function values of f along the search direction d_k through the current iterate x are denoted by ϑ. It is important to note that ϑ is a function of the step size α_k, as discussed earlier. The optimal choice for α_k, according to our previous discussion, would be as a minimizer of the function ϑ. However, it may prove to be a costly effort to locate the minimum of g. The method used to locate a point on a given curve that satisfies specific criteria is called the line search procedure. The process generally involves a series of iterations aimed at narrowing down the search range to ultimately achieve a suitable solution. To summarize, the line search procedure is employed to locate a point on the curve that meets specific criteria, and it typically involves a series of iterations to narrow down the search range and obtain a satisfactory outcome.

To commence our analysis, it is necessary to examine several characteristics of the aforementioned function $\vartheta(\alpha)$. We know that

$$\vartheta'(\alpha) = \nabla f(x_k + \alpha_k d_k)^T d_k. \tag{2.14}$$

Therefore,

$$\vartheta(0) = f(x_k), \tag{2.15}$$

$$\vartheta'(0) = \nabla f(x_k)^T d_k, \tag{2.16}$$

$$\vartheta(\alpha_k) = f(x_k + \alpha_k d_k). \tag{2.17}$$

Let d_k be a descent direction at x_k, we know that $\vartheta'(0) < 0$, which means that the function $\vartheta(\alpha)$ is decreasing at $\alpha = 0$. Therefore, it is possible to choose a positive x_k, such that $\vartheta(\alpha) < \vartheta(0)$, i.e., $f(x_k + \alpha_k d_k) < f(x_k)$.

We wish to find an α_k that is close to the minimum of $\vartheta(\alpha)$. It is required that α_k in order to ensure satisfaction of the two following conditions, a selection is made:

i.Sufficient decrease requirement, for a specific parameter$c_1 \in (0,1)$,

$$\vartheta(\alpha_k) < \vartheta(0) + c_1\vartheta'(0)\alpha_k \tag{2.18}$$

i.e.,

$$f(x_k + \alpha_k d_k) < f(x_k) + c_1\alpha_k\nabla f(x_k)^T d_k. \tag{2.19}$$

The mathematical expression denoted as $L(\alpha)$. In Equation (2.17) is the focus of our discussion. It is important to note that $L(\alpha)$ has a negative slope when d_k is set as the direction of descent. Nevertheless, due to the presence of $c_1 \in (0,1)$, it is situated above the graph of ϑ on a relatively small positive α. This observation has significant implications for the analysis of the function $L(\alpha)$ and its behavior in relation to other variables.

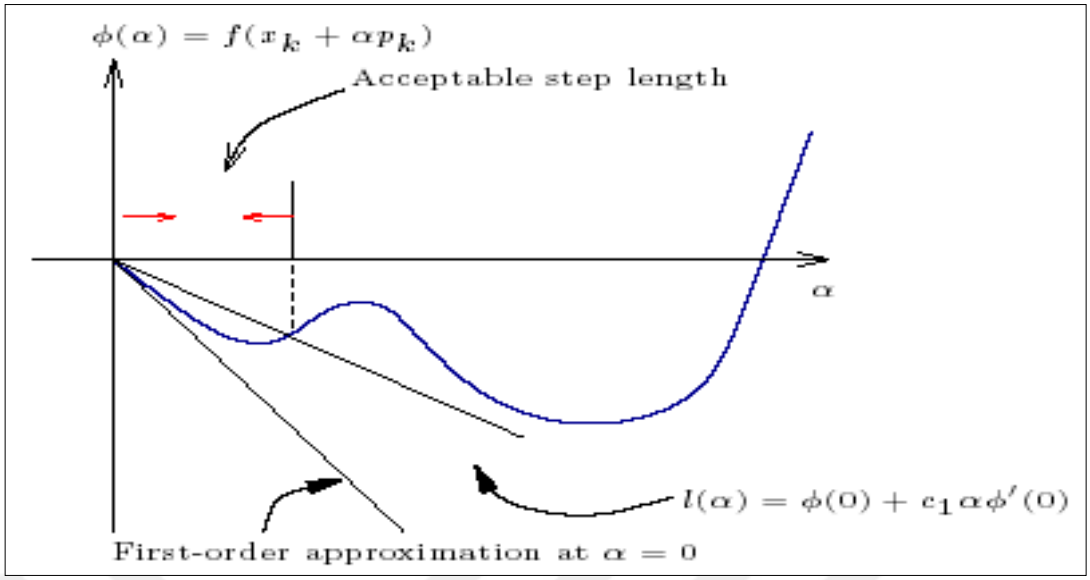

Figure 2.1 The sufficient decrease condition

The curvature condition, given a specific parameter $c_2 \in (c_1, 1)$,

$$\vartheta'(\alpha_k) \geq c_2 \, \vartheta'(0) \tag{2.20}$$

i.e.,

$$\nabla f(x_k + \alpha_k \, d_k)^T d_k \geq c_2 \, \nabla f(x_k)^T d_k. \tag{2.21}$$

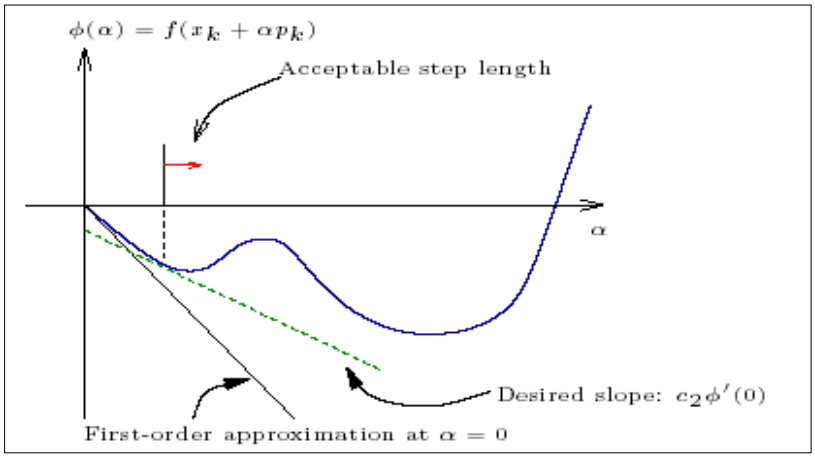

Figure 2.2 The curvature condition

Note that the left-hand-side of (2.21) is just $\vartheta'(\alpha_k)$, the condition of curvature serves to guarantee a level of smoothness in the slope of ϑ at $\alpha_k$is bigger than c_2 times the slope of ϑ at $\alpha = 0.7$.

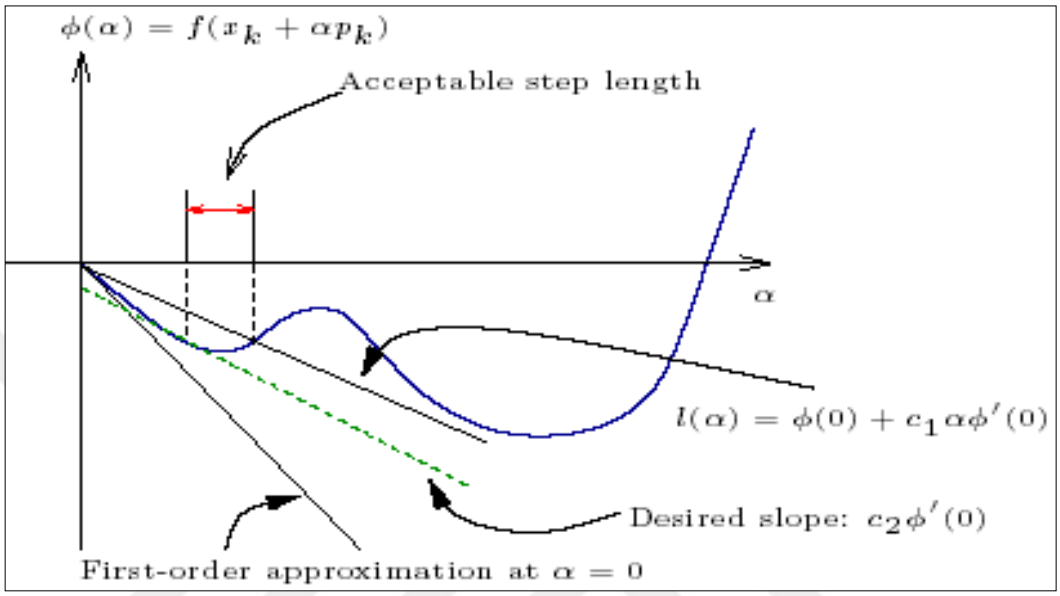

Figure 2.3 The combined wolfe condition

In the line search technique, the two conditions mentioned above, (2.19) and (2.21), are known as standard wolfe conditions. We say that the search direction d_k satisfies the sufficient descent property. i.e. it is a sufficient descent if $\vartheta_k^T d_k < -c\|\vartheta_k\|^2$ for $k = 0,1,2,\ldots$ where $c > 0$ is a constant.

2.3.2 Steepest descent method (SD)

Minimizing the goal function f via a linear approximation is the foundation of the SD method, which is applied to the situation where $x_k \in \mathbb{R}^n$ is currently located. Particularly, let us suppose that the present point x_k meets the condition $\vartheta(x_k) \neq 0$. If this is the case, then we will be able to build a linear approximation of f at x_k by using Taylor's theorem, which may be restated as follows:

The SD approach is based on minimizing the objective function f is linear approximation, at the current point $x_k \in \mathbb{R}^n$. Particularly, assume the current point x_k satisfies $\vartheta(x_k) \neq 0$, Then we'll be able to make a linear approximation of f at $x_k$by Taylor's theorem, which is stated as follows:

$$f_{k+1}(x) \equiv f(x_k) + \vartheta_k^T(x - x_k). \tag{2.22}$$

Now consider this if there exists a $d \in \mathbb{R}^n$ such that $\vartheta_k^T d < 0$ then $\exists$ an $\alpha_0 > 0$ such that (2.10) holds for all $\alpha \in (0, \alpha_0)$ (Khudhur 2015). In light of this matter, it is imperative that we select from one of the ensuing options to ensure a successful lineage. $x \in R^n$ s.t. $\vartheta_k^T(x - x_k)$ is, of course, a negative, if $g_k \neq 0$ and $\vartheta(x_k) \neq 0$. Afterwards, we'll be able to craft $\vartheta_k^T(x - x_k)$ as negative as possible (Khudhur 2015). As a result, we must limit the length of the direction $d = x - x_k$, specifically, we can examine the subsequent issue in greater detail

$$Min(\vartheta_k^T d) \text{ Subject to } \|d_k\|_2^2 \leq \|\vartheta_k\|_2^2.$$

According to the Cauchy-Schwarz inequality, the ideal response to the aforementioned issue is:

$$d_k = -\vartheta. \tag{2.23}$$

The direction d_k SD and the ensuing iteration process are referred to as

$$x_{k+1} = x_k - \alpha_k \vartheta_k \tag{2.24}$$

is called the method of SD.

2.3.3 Newton method

When making a decision regarding an appropriate search direction, it is important to note that the steepest descent method solely employs first derivatives, thus this should be taken into consideration. This approach may not necessarily provide the greatest results in every situation. The iterative process that is produced as a consequence may work more effectively than the SD method if higher derivatives are used. The Newton's technique, for example, which makes use of both the first and second derivatives, performs better than the SD method when the beginning point is located in close proximity to the minimizer.

For the time being, let's assume that G_k is definite and positive (and that it can therefore be inverted). We can find the Newton direction by figuring out the vector d minimizing $m_k(d)$. To achieve this effect, choose a gradient $m_k(d)$ w. r. t. $d = 0$. It is possible to locate Newton's direction:

$$d_k^N = -G_k^{-1}\vartheta_k. \tag{2.25}$$

The next point, x_{k+1}, may therefore be found by:

$$x_{k+1} = x_k + \alpha_k d_k^N. \tag{2.26}$$

The pure Newton method is the collective name for the iterative methods that were developed as a result, (2.25) and (2.26). If G_k^{-1} is positive definite and symmetric, the following formulation is a strict descent:

$$\vartheta_k^T d_k^N = -\vartheta_k^T G_k^{-1}\vartheta_k \leq -\lambda\|\vartheta_k\|^2 \tag{2.27}$$

where λ is smallest eigenvalue of G^{-1}, equation (2.27) is obtained by using the equality $\vartheta_k^T G_k \vartheta_k \geq \lambda\vartheta_k^T\vartheta_k$ (Witzgall and Fletcher 1989).

2.3.4 Methods based on the quasi-newton principle (QN)

In the 1960s, quasi-Newton methods, which do not require costly Hessian matrices computations and have good performance in practice, revolutionized nonlinear optimization. Several other types have been proposed, but the BFGS technique has gained more traction since the 1970s and is now often regarded as the best QN strategy (Mascarenhas 2004) the following words are used to explain the search:

$$d_k = -H_k \vartheta_k. \tag{2.28}$$

Additionally, it uses (2.28) to determine the next point. H_k is a $n \times n$ in this instance, the inverse, you can use a positive-definite symmetric matrix that will vary after each iteration to approximate the Hessian. The following rule is applied in the context of BFGS to modify H_{k+1}:

$$H_{k+1}^{BFGS} = \left(I - \frac{s_k y_k^T}{s_k^T y_k}\right) H_k \left(I - \frac{y_k s_k^T}{s_k^T y_k}\right) + \frac{s_k s_k^T}{s_k^T y_k} \tag{2.29}$$

when using H_1 as the identity matrix, H_{k+1} must be limited to satisfy either the quasi-Newtonian condition or the secant equation, which can be expressed as follows:

$$H_{k+1} y_k = s_k. \tag{2.30}$$

This condition is only true if the step is finished. The requirement of curvature is satisfied by s_k and the gradient change of y_k

$$s_k^T y_k > 0. \tag{2.31}$$

In order to maintain a consistent satisfaction of this requirement, it is imperative that the convexity be significantly high. Nevertheless, it is essential to implement equation (2.31) with caution, as it requires a careful imposition of restrictions on the line search method

used for selection. This approach is necessary in order to ensure that the method of selection does not deviate from the desired outcome α_k. In reality, (2.31) is certain to hold whether the Wolfe criteria are applied to the line search (strongly or weakly) (Khudhur 2015).

Although BFGS has certain desired characteristics, such as super linear convergence, Walter in (Mascarenhas 2004) shows that when applied to non-convex objective functions, the BFGS technique and other Broyden class methods that demand accurate line searches might not converge. the QN technique has the additional drawback of requiring handling $n \times n$ matrices. The final search direction strategy discussed in this chapter is the nonlinear conjugate gradient (CG) method. the method is thoroughly investigated in the section that follows since it is a part of our work in this area and will be covered in more detail there.

2.3.5 Conjugate Gradient Methods

They provide a group of crucial techniques for resolving the unconstrained optimization issue (1.1), particularly when n is large. They possess this form:

$$x_{k+1} = x_k + \alpha_k d_k, \quad k \geq 1 \tag{2.32}$$

where α_k is the search direction determined by d_k is the step size acquired from a line search

$$d_{k+1} = \begin{cases} -\vartheta_1, & k = 1 \\ -\vartheta_{k+1} + \beta_k d_k, & k \geq 1 \end{cases} \tag{2.33}$$

and β_k a parameter can be defined as a measurable factor or characteristic that can influence the behavior or outcome of a system or process. The most well-known formulas for β_k are those of Hestenes and Stiefel β^{HS}(Hestenes and Stiefel 1952), $\mathbb{R}$. Fletcher and

Reevesβ^{FR}(Fletcher 1964), Polak and Ribiereβ^{PR}(Polak and Ribiere 1969), and Dixonβ^{Dx}(Dixon 1975) and Liu-Storey β^{Ls}(Liu and Storey 1991):

$$d_1 = -\vartheta_1, y_k = \vartheta_{k+1} - \vartheta_k$$

$$d_{k+1} = -\vartheta_{k+1} + \frac{y_k^T \vartheta_{k+1}}{y_k^T d_k} d_k \quad (2.34.a)$$

(Hestenes and Stiefel 1952) or

$$d_{k+1} = -\vartheta_{k+1} + \frac{\vartheta_{k+1}^T \vartheta_{k+1}}{\vartheta_k^T \vartheta_k} d_k \quad (2.34.b)$$

(Fletcher 1964), or

$$d_{k+1} = -\vartheta_{k+1} + \frac{y_{k+1}^T \vartheta_{k+1}}{\vartheta_k^T \vartheta_k} d_k \quad (2.34.c)$$

(Polak and Ribiere 1969), or

$$d_{k+1} = -\vartheta_{k+1} + \frac{\vartheta_{k+1}^T \vartheta_{k+1}}{d_k^T \vartheta_k} d_k \quad (2.34.d)$$

(Dixon 1975), or

$$d_{k+1} = -\vartheta_{k+1} + \frac{\vartheta_{k+1}^T y_k}{d_k^T \vartheta_k} d_k \quad (2.34.e)$$

(Liu and Storey 1991).

2.3.6 Fuzzy logic

Fuzzy logic was first introduced in a work named "Fuzzy Sets" by Lotfi A. Zadeh, which was published in 1965. He is credited with creating fuzzy logic.

Fuzzy logic operates on a decision-making mechanism akin to that employed by humans. Its central focus is on ambiguous and unreliable data, which is its primary concern. This methodology is not grounded on the conventional true/false or 1/0 thinking of Boolean logic, which is an immense oversimplification of real-world complexities. Rather, it relies on the degree of truth, utilizing this measure as its foundation. The degree to which something is true is the fundamental basis for fuzzy logic, and it is this principle that sets it apart from other types of logic. Through its application, fuzzy logic allows for a more nuanced and realistic approach to decision-making, particularly in situations where the data is not entirely clear.

It demonstrates how numbers ranging from 0 to 1 can be used to represent values in fuzzy structures. Absolute truth is symbolized by the number 1.0, while absolute untruth is symbolized by the number 0.0. A numerical representation of a value in a fuzzy system is the truth value (Lee 2005).

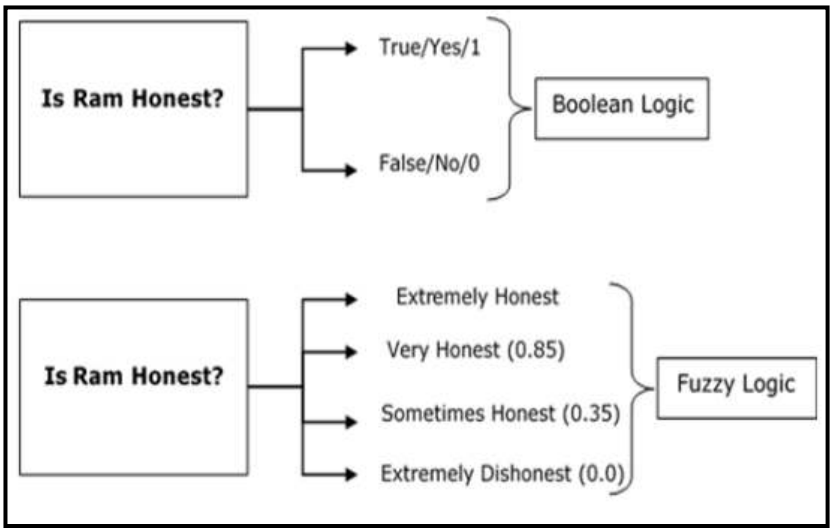

Figure2.4 Boolean logic and fuzzy logic

2.3.7 Fuzzy sets and classical sets

In classical, or crisp, sets, an element's transition from membership to non-membership in a particular set is abrupt and clearly defined (referred to as "crisp"). In a universe with fuzzy sets, this transition for an element will take place gradually. This transition between distinct degrees of membership reflects the hazy and ambiguous nature of the fuzzy set's bounds. As a result, the question of whether an element from the universe belongs to this set is answered by a function that tries to explain ambiguity and uncertainty.

The composition of the set is ambiguous and contains elements that belong to the set to different extents based on the constituent. Based on this explanation, the entities belonging to a conventional set are not acknowledged as members of the set unless their association with that set is absolute or fully realized (i.e., their association is attributed a worth of 1 in this definition). This definition of membership in a set underscores the importance of precision in defining the elements of the set.(Dubois and Prade 1978).

2.4 Membership

Utilizing the characteristic function, sometimes referred to as the discriminating function, which is a component of the membership function, it is possible to ascertain whether or not an entity x is a member of a collection A. This may be done in the context of a collection A.

2.4.1 Membership function

For set A, a membership function is defined $u_A(x)$ as follows:

$$u_A(x) = \begin{cases} 1, & \text{if } x \in A \\ 0, & \text{if } x \notin A \end{cases}.$$

The mapping from the general set x to the set $\{0,1\}$ is the definition of the function $u_A(x)$.$u_A(x)$: $x \longrightarrow \{0,1\}$ (Maleki 2002).

2.4.2 Types of membership functions

Definition 2.4.1 Triangular Membership Function: Three parameters, a, b, and c, are used to specify this function's general mathematical formula (Ganesan and Veeramani 2006):

$$u(x) = \begin{cases} 0; & x \leq a \\ \frac{x-a}{b-a}; & \text{a} \leq x \leq b \\ \frac{c-x}{c-b}; & \text{b} \leq x \leq c \\ 0; & x \geq a \end{cases}. \tag{2.35}$$

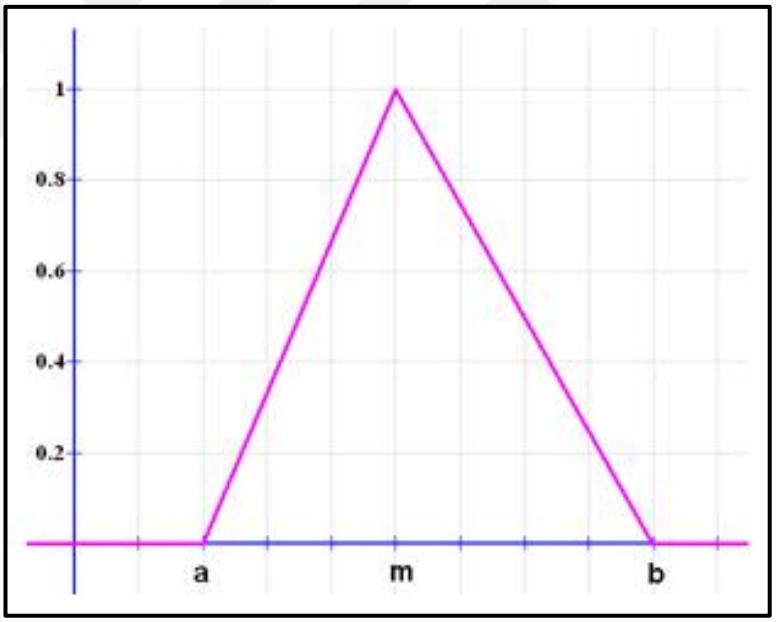

Figure 2.5 Triangular membership function

Definition 2.4.2 (Trapezoidal Membership Function) Four parameters , a, b, and c, are required in the following generic mathematical formula for this function (Ganesan and Veeramani 2006):

$$u(x) = \begin{cases} 0; & x \leq a \\ \frac{x-a}{b-a}; & \text{a} \leq x \leq b \\ 1; & \text{a} \leq x \leq b. \\ \frac{c-x}{c-b}; & \text{c} \leq x \leq d \\ 0; & x \geq d \end{cases} \tag{2.36}$$

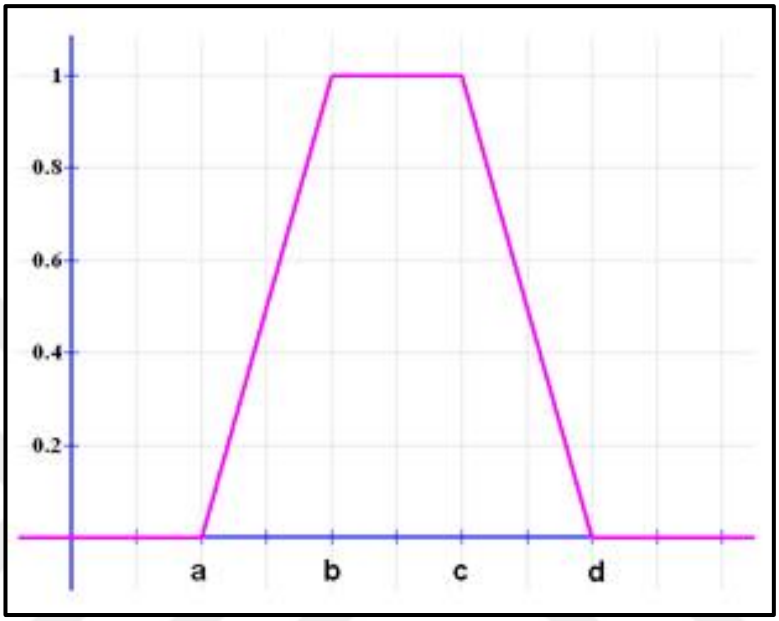

Figure 2.6 Trapezoidal membership function

Definition 2.4.3 (Gaussian Membership Function) The two parameters $\{a, b\}$ in the following general mathematical formula define this function (Ganesan and Veeramani 2006):

$$\mu(x) = e^{\frac{-(x-a)^2}{b}}. \tag{2.37}$$

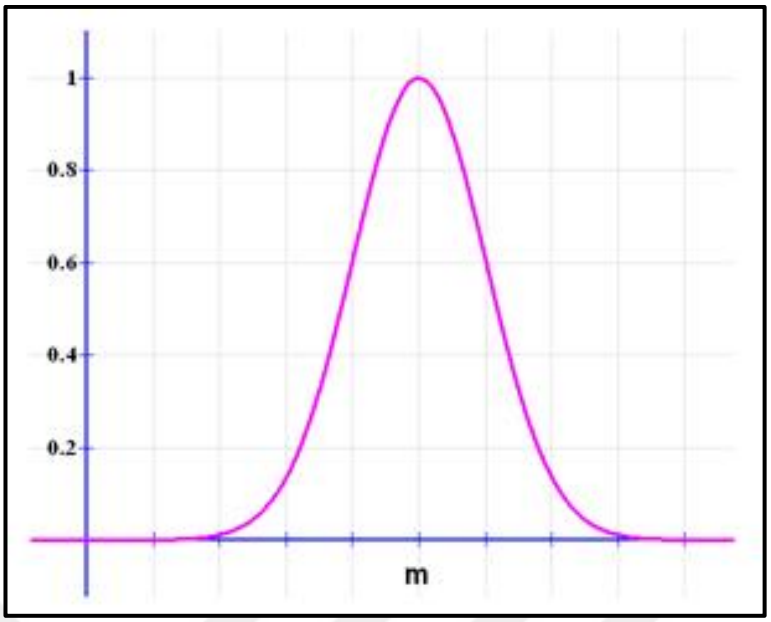

Figure 2.7 Gaussian membership function

2.4.3 Fuzzy Set

Zadeh came up with the idea of binary membership on the real continuous interval set value $[0, 1]$, with the endpoints of 0 and 1 standing in for neither membership nor participate on. This is similar to what the indicator function does for crisp sets, with the exception that an element x can have an unlimited range of values between the endpoints that indicate various degrees of membership in a set on the universe. Zadeh did this so that different "degrees of membership" could be accommodated. There is a connection between this and the manner in which the indicator function operates for crisp sets.

The membership functions of a crisp set are distinct from those of a fuzzy set, which may have an unlimited number of membership functions to represent it. A crisp set is denoted by a single membership function, whereas a fuzzy set may be represented by any number of membership functions. The following statement offers a logical representation of membership in mathematical terms: $u_A(x) \in [0,1]$.

The notation pattern for fuzzy sets that follows is as follows when the domain of discourse, x is discrete and finite

$$A = \left\{\frac{u_A(x_1)}{x_1} + \frac{u_A(x_2)}{x_2}\right\} = \sum_i \frac{u_A(x_i)}{x_i}. \tag{2.38}$$

The fuzzy set A is written as follows when the universe, x, is continuous and infinite:

$$A = \left\{\int \frac{u_A(x)}{x}\right\} \tag{2.39}$$

(Chaturvedi 2017).

3. MATERIALS AND METHODS

We come into problems with linear and non-linear programming, which are referred to using the abbreviation (LNPP) when we are attempting to optimize systems in the real world. Real numbers are used to represent a significant amount of the coefficients in almost all optimization problems. On the other hand, the coefficients that are involved in the objective and constraint functions have a nature that is imprecise in the real world; as a result, they have to be presented in a form that is fuzzy in order to effectively reflect the situation. This is necessary in order to ensure that the scenario is accurately portrayed. Bellman and Zadeh were the first persons to tackle the problems associated with fuzzy optimization. (Bellman Re and Zadeh La 1970). In the next chapter, we will illustrate how to employ a conjugate gradient approach to solve issues involving unconstrained fuzzy optimization. One of the most common applications of nonlinear optimization techniques is the conjugate gradient method; for additional information on this topic, take a look at the following: (Bazaraa, Sherali and Shetty 2013, Nocedal and Wright 2006). They are effective solutions for a wide range of small to medium-sized optimization problems, in particular those in which calculating the Hessian presents a challenge. There are a variety of conjugate gradient methods; however, we apply the well-known conjugate gradient approach to fuzzy optimization problems in this chapter. This method has a range of advantages. First, only first-derivatives knowledge can be used to find a parameter beta. Second, the proposed algorithm is successful for problems where the Hessian is difficult to compute, the suggested scheme performs well.

3.1 Fundamental Concepts and Preliminaries

Assume that $\mathbb{R}$ is a set of real numbers. On $\mathbb{R}$, a fuzzy set u is a mapping. Let $\mathbb{R}$ be the set of real numbers. $u: \mathbb{R} \longrightarrow [0,1]$, the membership function of a fuzzy set u defines its properties $u: \mathbb{R} \longrightarrow [0,1]$, which each x in $\mathbb{R}$ is associated with a real number μ_u in $[0,1]$.

Let u be a fuzzy set. The $\alpha - cut$ or $0 - level$ of the fuzzy set u is defined as follows: $[u]^{\alpha} = \{x \in \mathbb{R}: \mathrm{u(x)} \geq \alpha\}$. Furthermore, u support is indicated by

$$S(u) = \{x \in \mathbb{R}: \mathrm{u(x)} > \alpha\},$$

the $0 - level$ of u is defined by the closure of support u, $[u]^0 = cl\big(S(u)\big)$.

Definition 3.1.1 (Bector and Chandra 2005): A fuzzy number is defined as a fuzzy set u on $\mathbb{R}$ that satisfies the following properties:

i- u is normal, i.e. $\exists x_0 \in \mathbb{R}$ in the context that $u(x_0) = 1$.

ii- u is a fuzzy convex set, i.e., $u(\lambda x + (1-\lambda)y \geq min\{u(x), u(y)\}$, whenever $x, y \in \mathbb{R}$ and $\lambda \in [0,1]$.

iii-u is upper semi-continuous on $\mathbb{R}$.

iv-$[u]^0$ is a compact set.

The family of all fuzzy numbers on $\mathbb{R}$ is denoted by $F(\mathbb{R})$. $[u]^\alpha$is a condensed interval in the space $\mathbb{R}$ for all of the values $\alpha \in [0,1]$, as can be shown by definition, and therefore the $\alpha - level$ of u is a fuzzy number that is defined by $[u]^\alpha = \left[\underline{u}_\alpha, \overline{u}_\alpha\right]$, where $\underline{u}_\alpha, \overline{u}_\alpha \in \mathbb{R}$ for all $\alpha \in [0,1]$.

Let's say u and v are two ambiguous numbers. The addition and scalar multiplication of the $\alpha - \ level$ sets in $F(\mathbb{R})$ are defined as follows, respectively:

$$[u+v]^\alpha = \left[\underline{u}_\alpha, \underline{v}_\alpha, \overline{u}_\alpha, \overline{v}_\alpha\right] \tag{3.1}$$

and

$$[\lambda u]^\alpha = \left[min\{\lambda\underline{u}_\alpha, \lambda\overline{u}_\alpha\}, max\{\lambda\underline{u}_\alpha, \lambda\overline{u}_\alpha\}\right] \tag{3.2}$$

where $\lambda \in \mathbb{R}$ and $\alpha \in [0,1]$.

Definition 3.1.2 (Bector and Chandra 2005)The formula for determining a triangular fuzzy number is $u = (a, b, c)$, where a, b, and c are three real values and the membership function is. Triangular fuzzy numbers are a kind of fuzzy numbers.

$$\mu_u(x) = \begin{cases} \frac{x-a}{b-a}, & a \le x \le b \\ \frac{c-x}{c-b}, & b \le x \le c \\ 0, & otherwise \end{cases} \tag{3.3}$$

a triangular fuzzy number $u = (a, b, c)$ has $\alpha - level$ set, which is given by:

$$[u]^{\alpha} = [(1-\alpha)a + \alpha b, (1-\alpha)c + \alpha b], \quad \forall \alpha \in [0,1].$$

Definition 3.1.3 (Bector and Chandra 2005) In $F(\mathbb{R})$, let u and v represent two fuzzy numbers. Hence $[u]^{\alpha} = \left[\underline{u}_{\alpha}, \overline{u}_{\alpha}\right]$ and $[v]^{\alpha} = \left[\underline{v}_{\alpha}, \overline{v}_{\alpha}\right]$ are two intervals in $\mathbb{R}$ for all $\alpha \in [0,1]$. This is how we describe

$$u \le v \leftrightarrow [u]^{\alpha} \le [v]^{\alpha}, \alpha \in [0,1] \leftrightarrow \underline{u}^{\alpha} \le \underline{v}^{\alpha} \text{and} \overline{u}^{\alpha} \le \overline{v}^{\alpha}, \forall \alpha \in [0,1]$$

and

$$u < v \leftrightarrow u \le v \text{ and } u \ne v \leftrightarrow [u]^{\alpha} \le [v]^{\alpha}, \forall \alpha \in [0,1] \text{and } \exists \alpha^* \in [0,1]\ s.t.\ \underline{u}^{\alpha^*} \le \underline{v}^{\alpha^*} \text{ or } \overline{u}^{\alpha^*} \le \overline{v}^{\alpha^*}.$$

Definition 3.1.4 (Stefanini 2010) Given $u, v \in F(\mathbb{R})$. The generalized Hukuhara difference (gH-difference for short) between u and v is the fuzzy number w, if

$$u \ominus_{gh} v = w \Leftrightarrow \begin{cases} u = v + w \quad or \\ v = u + (-1)w. \end{cases}$$

Using the α −levels we have

$$[u \ominus_{gh} v] = [u]^{\alpha} \ominus_{gh} [v]^{\alpha} = \left[\min\{\underline{u}_{\alpha} - \underline{v}_{\alpha}, \overline{u}_{\alpha} - \overline{v}_{\alpha}\}, \max\{\underline{u}_{\alpha} - \underline{v}_{\alpha}, \overline{u}_{\alpha} - \overline{v}_{\alpha}\}\right],$$

$$\forall \alpha \in [0,1]$$

where $[u]^{\alpha} \ominus_{gh} [v]^{\alpha}$ between two intervals is the gH-difference see (Lupulescu 2013, Tanaka *et al*. 1973) .

3.2 Fuzzy Optimization and Fuzzy Differentiable Functions

In this part, the definition of differentiability of fuzzy functions is introduced first in this section. Later on, we'll look at fuzzy optimization problems and define non-dominated solutions.

3.2.1 Fuzzy Functions That Are Differentiable

From this point forward, the letter X will designate an open subset of the set $\mathbb{R}^n$. In accordance with the specification, a fuzzy function that is defined on X is denoted by the mapping $F: X \longrightarrow F(\mathbb{R})$. X_c is the symbol that represents the family of all bounded closed intervals in $\mathbb{R}$. Each fuzzy function has a name that we associate with it $F: X \longrightarrow F(\mathbb{R})$, interval-valued functions are a type of function that has a value between two intervals $F: X \longrightarrow X_c$ given by $F_{\alpha}(x) = [F(x)]^{\alpha}$. For each $\alpha \in [0,1]$, we denote $F_{\alpha}(x) = [F(x)]^{\alpha} = \left[\underline{f}^{\alpha}(x), \overline{f}^{\alpha}(x)\right]$. The endpoint functions $\underline{f}^{\alpha}(x), \overline{f}^{\alpha}(x): X \longrightarrow \mathbb{R}$, $F_{\alpha}(x)$ is functions, respectively, are referred to as upper and lower functions.

Definition 3.2.1.1 Let $X \subset \mathbb{R}$ and $F: X \longrightarrow F(\mathbb{R})$ to be a fuzzily defined function. Furthermore, consider the probability that $x_0 \in X$ and h must be in such a way that $x_0 + h \in X$. At x_0, F's generalized Hukuhara derivative (gH-derivative) is defined as

$$F(x) = \lim_{h \to 0} \frac{F(x_0+h) \ominus_{gh} F(x_0)}{h}. \quad (3.4)$$

If $\grave{F}(x) \in F(\mathbb{R})$ satisfying (3.4) exists, if F is generalized Hukuhara differentiable (gH-differentiable) at x_0, then it is said to be gH-differentiable. We assume F is gH-differentiable over X if it is gH-differentiable at any $x \in X$ (Bede and Stefanini 2013).

Definition 3.2.1.2 In $\mathbb{R}$, consider X to be an open set. a function with an interval value $F: X \longrightarrow X_c$ is gH-differentiable at $x_0 \in X$, if (3.4) which exists in the metric space with respect to the limit (X_c, H), the gH-difference between intervals is used to calculate the difference (Tanaka *et al*. 1973).

Theorem 3.2.1.1 Let $F: X \longrightarrow F(\mathbb{R})$ be a nebulous function If F is gH-differentiable, then the interval-valued function must be gH-differentiable $F_\alpha: X \longrightarrow X_c$is gH-differentiable for each $\alpha \in [0,1]$. Moreover, $\grave{F}_\alpha(x) = \left[\grave{F}(x)\right]^\alpha = [(\underline{f}^\alpha\grave{)}(x), (\overline{f}^\alpha\grave{)}(x)]$ (Arana-Jiménez *et al*. 2015).

Theorem 3.2.1.2 Let $F: X \longrightarrow F(\mathbb{R})$ to be a fuzzily defined function If F is gH-differentiable at $x_0 \in X$, one of the following situations holds for each $\alpha \in [0,1]$

$\underline{f}^\alpha$ and $\overline{f}^\alpha$ are differentiable at x_0 and

$$\left[\grave{F}(x)\right]^\alpha = [\min\{(\underline{f}^\alpha\grave{)}(x_0), (\overline{f}^\alpha\grave{)}(x_0)\}, \max\{\left(\underline{f}^\alpha\grave{)}(x_0), \left(\overline{f}^\alpha\grave{)}(x_0)\right\}\right],$$

$$(\underline{f}^\alpha\grave{)} - (x_0), (\overline{f}^\alpha\grave{)} - (x_0), (\underline{f}^\alpha\grave{)} + (x_0) \text{ and } (\overline{f}^\alpha\grave{)} + (x_0)$$

exist and satisfy

$$(\underline{f}^\alpha\grave{)} - (x_0) = (\overline{f}^\alpha\grave{)} + (x_0) \text{ and } (\underline{f}^\alpha\grave{)} + (x_0) = (\overline{f}^\alpha\grave{)} - (x_0).$$

Moreover,

$$\left[\grave{F}(x)\right]^{\alpha} = [\min\{(\underline{f}^{\alpha}\grave{)} - (x_0), (\overline{f}^{\alpha}\grave{)} - (x_0)\}, \max\{(\underline{f}^{\alpha}\grave{)} + (x_0), (\overline{f}^{\alpha}\grave{)} + (x_0)\}]$$

$$= [\min\{(\underline{f}^{\alpha}\grave{)} + (x_0), (\overline{f}^{\alpha}\grave{)} + (x_0)\}, \max\{(\underline{f}^{\alpha}\grave{)} - (x_0), (\overline{f}^{\alpha}\grave{)} - (x_0)\}]$$

(Chalco-Cano *et al*. 2015).

Definition 3.2.1.3 Let $F: X \longrightarrow F(\mathbb{R})$ be a fuzzy function and let $x^0 = (x_1^0, x_2^0, \dots, x_n^0)$ be a part of X that can't be moved. Take the fuzzy feature into consideration $h(x_i) = F(x_1^0, x_2^0, \dots, x_n^0)$. We assume that F has the ith partial gH-derivative at x_0 if h is gH-differentiable at x_i^0 (indicate by $\frac{\partial F}{\partial x_i}(x^0)$) and $\frac{\partial F}{\partial x_i}(x^0) = \grave{(h)}(x_i^0)$ (Arana-Jiménez *et al*. 2015).

Definition 3.2.1.4 (Arana-Jiménez *et al*. 2015): Let $F: X \longrightarrow F(\mathbb{R})$ to be a fuzzily defined function F represents the gradient of $\tilde{\nabla}F(x^0)$, at x_0,

$$\tilde{\nabla}F(x^0) = \left(\frac{\partial F}{\partial x_1}(x_0), \frac{\partial F}{\partial x_2}(x_0), \dots, \frac{\partial F}{\partial x_n}(x_0)\right)$$

the $\alpha - level$ set of $\tilde{\nabla}F(x^0)$ is a term that is identified and denoted by

$$[\tilde{\nabla}F(x^0)]^{\alpha} = \left([\frac{\partial F}{\partial x_1}]^{\alpha}(x_0), [\frac{\partial F}{\partial x_2}]^{\alpha}(x_0), \dots, [\frac{\partial F}{\partial x_n}]^{\alpha}(x_0)\right),$$

where

$$[\frac{\partial F}{\partial x_i}]^{\alpha} = \left[\frac{\partial \underline{F}^{\alpha}}{\partial x_i}, \frac{\partial \overline{F}^{\alpha}}{\partial x_i}\right].$$

Theorem 3.2.1.3 Let $F: X \longrightarrow F(\mathbb{R})$ to be a fuzzily defined function. If F is gH-differentiable at a certain point, then $x_0 \in X$following that, for each $\alpha \in [0,1]$, the real-valued function $\underline{f}^{\alpha} + \overline{f}^{\alpha}: X \longrightarrow \mathbb{R}$ is differentiable at x_0 moreover (Chalco-Cano *et al.* 2015),

$$\frac{\partial \underline{F}^{\alpha}}{\partial x_i}(x_0) + \frac{\partial \overline{F}^{\alpha}}{\partial x_i}(x_0) = \frac{\partial\left(\underline{f}^{\alpha} + \overline{f}^{\alpha}\right)}{\partial x_i}(x_0). \tag{3.5}$$

Definition 3.2.1.5 Let $F: X \longrightarrow F(\mathbb{R})$ to be a fuzzily defined function. If $\tilde{\nabla} F(x^0)$ is gH-differentiable at x_0, then the function $F(x^0)$ is gH-differentiable for each $i\frac{\partial F}{\partial x_i} X \longrightarrow F(\mathbb{R})$. We assume that F is twice gH-differentiable at x_0 if it is gH-differentiable at x^0. The partial derivative of gH is denoted by $\frac{\partial F}{\partial x_i}$ the symbol gH by

$$D_{ij}^2 F(x^0) = \frac{\partial^2 F}{\partial x_i \partial x_j}(x^0), \;\; i \neq j$$

and

$$D_{ij}^2 F(x^0) = \frac{\partial^2 F}{\partial x_i^2}(x^0), \;\; i = j.$$

Suppose for all $i, j = 1, 2, \ldots, n$, the partial derivative of a cross-partial $\frac{\partial^2 F}{\partial x_i \partial x_j}$ is continuous from X to $F(\mathbb{R})$, we say F is twice continuous differentiable (Chalco-Cano *et al.* 2015).

Theorem 3.2.1.4 Let $F: X \longrightarrow F(\mathbb{R})$ to be a fuzzily defined function If F is differentiable by m times gH at $x_0 \in X$following that, for each $\alpha \in [0, 1]$, the function that has a real value $\underline{f}^{\alpha}, \overline{f}^{\alpha}: X \longrightarrow \mathbb{R}$ is differentiable by m-times at x_0 (Chalco-Cano *et al.* 2015).

3.3 Fuzzy Optimization (FO)

The following fuzzy optimization problem is considered (FOP):

$$(FOP) min_{x \in X} F(x) \tag{3.6}$$

where the goal function is $F: X \longrightarrow F(\mathbb{R})$ is a function with a fuzzily defined value and $X \subseteq \mathbb{R}^n$, F is domain, which is believed to be an open set, is we'll assume that F is gH-differentiable for the rest of the chapter.

Definition 3.3.1 Let $X \subseteq \mathbb{R}^n$ be a set that is open. We state this $x^* \in X$ is a FOP (3.6) solution that is not dominated locally if no such thing exists $x \in N_s(x^*) \bigcap X$ such that $F(x) < F(x^*)$, where $N_s(x^*)$ is an ϵ −neighborhood (Pirzada and Pathak 2013).

Theorem 3.3.1 Let $X \subseteq \mathbb{R}^n$ be an open set and $F: X \longrightarrow F(\mathbb{R})$ to be a fuzzy function. If x^* is the real-valued function's local minimizer $\underline{f}^{\alpha} + \overline{f}^{\alpha}, \forall \alpha \in [0,1]$, then x^* is a FOP (3.6) solution that is not dominated locally (Chalco-Cano *et al*. 2015).

Theorem 3.3.2 Let $\alpha \in [0,1]$ and the function that has a real value $\underline{f}^{\alpha} + \overline{f}^{\alpha}: X \longrightarrow \mathbb{R}$ at x_0, it must be differentiable. If there is a vector d that has the property of $(\nabla \left(\underline{f}^{\alpha} + \overline{f}^{\alpha}\right)(x_0))^T d < 0,$ then there exists a $\delta > 0$ such that $\left(\underline{f}^{\alpha} + \overline{f}^{\alpha}\right)(x_0 + \lambda d) < \left(\underline{f}^{\alpha} + \overline{f}^{\alpha}\right)(x_0)$ for any $\lambda \in (0, \delta)$. Therefore that, d is the direction of descent of $\underline{f}^{\alpha} + \overline{f}^{\alpha}$ (Ghaznavi and Hoseinpoor 2017)**.**

Proof. Let $\alpha \in [0,1]$ and $(\nabla \left(\underline{f}^{\alpha} + \overline{f}^{\alpha}\right)(x_0))^T d < 0$, By the differentiability of $\underline{f}^{\alpha} + \overline{f}^{\alpha}$ at x_0, we have

$$\left(\underline{f}^{\alpha} + \overline{f}^{\alpha}\right)(x_0 + \lambda d) = \left(\underline{f}^{\alpha} + \overline{f}^{\alpha}\right)(x_0) + \lambda(\nabla \left(\underline{f}^{\alpha} + \overline{f}^{\alpha}\right)(x_0))^T d + \lambda \|d\| o(x_0; \lambda d),$$

where $o(x_0;\lambda d) \to 0$ as $\lambda \to 0$. We get

$$\frac{\left(\underline{f}^{\alpha}+\overline{f}^{\alpha}\right)(x_0+\lambda d)-\left(\underline{f}^{\alpha}+\overline{f}^{\alpha}\right)(x_0)}{\lambda}=(\nabla\left(\underline{f}^{\alpha}+\overline{f}^{\alpha}\right)(x_0))^T\,d+\|d\|o(x_0;\lambda d),\lambda\neq 0$$

since $(\nabla\left(\underline{f}^{\alpha}+\overline{f}^{\alpha}\right)(x_0))^T\,d<0$ and $o(x_0;\lambda d)\to 0$ as $\lambda\to 0$, there is a $\delta>0$ such that $(\nabla\left(\underline{f}^{\alpha}+\overline{f}^{\alpha}\right)(x_0))^T\,d+\|d\|o(x_0;\lambda d)<0,\lambda\in(0,\delta)$. Therefore, the proof is completed.

We presume we can compute $F(x_k),\tilde{\nabla}F(x_k)$ at each x_k point. We can compute F because it is gH-differentiable, as shown by Theorems 3.4 and (3.3), $\nabla\left(\underline{f}^{\alpha}+\overline{f}^{\alpha}\right)(x_k)$.

3.4 Conjugate Gradient (CG) Algorithms

Equation (3) represents the non-linear conjugate gradient (CG) structure

$$x_{k+1}=x_k+\alpha_k d_k,\ \ k\geq 1 \tag{3.7}$$

where x_1 represents the initial paint, while $\alpha_k$is a step–length, and α_k step-size in equations (3,8) and (3,9) that satisfies the common Wolfe criterion (Wolfe 1969, Ahmed *et al.* 2021)

$$f(x_k+\alpha_k d_k)\leq f(x_k)+\delta\alpha_k g_k^T d_k \tag{3.8}$$

$$d_k^T g(x_k+\alpha_k d_k)\geq\sigma d_k^T g_k \tag{3.9}$$

or strong Wolfe conditions in equation (3,10), and (3,11) (Wolfe 1969, Abbo and Khudhur 2015-2018; Abbo, *et al*. 2016; Abbo et al. 2018, Laylani *et al*. 2018, M. Azzam *et al*.2018, Abdullah *et al*. 2019; Khudhur and Abbo 2020)

$$f(x_k + \alpha_k d_k) \leq f(x) + \delta \alpha_k g_k^T d_k \tag{3.10}$$

$$|d_k^T g(x_k + \alpha_k d_k)| \leq -\sigma d_k^T g_k \tag{3.11}$$

$$d_{k+1} = \begin{cases} -g_1, & k = 1 \\ -g_{k+1} + \beta_k d_k, & k \geq 1 \end{cases}. \tag{3.12}$$

Different β_{k+1} will specify or determine various CG methods. Some well known formula for β_{k+1}as follows:

The Fletcher and Reeves (FR) (Fletcher 1964), Fletcher (CD) (Witzgall and Fletcher 1989), Polak and Ribiere (PRP) (Polak and Ribiere 1969), Hestenes-Stiefel (HS) (Hestenes and Stiefel 1952), Dai-Yuan (DY) (Dai and Yuan 1999), Hisham- Khalil (KH) (Khudhur and Abbo 2021a), 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=∇f(x_k), and let y_k=$g_{k+1} - g_k$.

3.5 The First Proposed New Algorithm

In this chapter, a new algorithm of conjugate gradient algorithms is proposed based on the Hisham and Khalil (KH) algorithm in order to overcome the slow convergence of

Hisham and Khalil (KH) algorithm and reach the solution with the least number of iterations.

$$d_{k+1} = -g_{k+1} + \beta_k^{(KH)} d_k \tag{3.13}$$

$$\beta^{KM} = \frac{\|g_{k+1}\|}{\|g_k\|_1} - r\frac{|d_k^T g_{k+1}|}{\|g_k\|_1} \tag{3.14}$$

where r is pocitirc scalar.

3.6 Algorithm (KM1-CG)

First step: Initialization: choose $x_1 \in R^n, \varepsilon > 0$ is a just a small positive real number, then calculate

$$d_1 = -g_1, \alpha_1 = \frac{1}{\|g_1} \|\ and\ k = 1.$$

Second step: Convergence testing: If $\| g_k \| \leq \varepsilon$ break z_k is optimum solution else go to next step.

Third step: Line search: computed α_k meeting the strong wolf criteria or conditions (3,10) and (3,11), then update the variable $x_{k+1} = x_k + \alpha_k d_k$, compute f_{k+1}, g_{k+1}, y_k and s_k.

Fourth step: Direction computation: compute utilizing (3.12) (3.13).

For Simplicity we call the method with $\beta = \beta^{KH}$ method KH

$$d_{k+1} = -g_{k+1} + \beta_k^{(KH)} d_k$$

$$\beta^{KH} = \frac{\|g_{k+1}\|}{\|g_k\|_1} - r\frac{|d_k^T g_{k+1}|}{\|g_k\|_1}$$

where r is pocitirc scalar.

Theorem 3.6.1 Assume that FR (Fletcher - Reeves) conjugate method is implement with the strong Wolfe line search (3,10) and (3,11) when $0 < P < \sigma < Y$. Then FR method generates descent directions satisfying the following inequalities

$$-\frac{1}{1-\sigma} \leq \frac{g_k^T d_k}{\|g_k\|_2^2} \leq \frac{2\sigma - 1}{1-\sigma}$$

(Al-baali 1985). Now, we can prove the descent property for the suggested by β^{KH} method by the following theorem.

Theorem 3.6.2 Assume that the KM1 method is implemented with the strong wolf line search. Then the KM1 method generates sufficient descent directions.

Proof: The Proof is by induction for $k = 0$,

$$d_1 = -g_1 \rightarrow g_1^T d_1 = -\|g_1\|_1^2 \leq -c\|g_1\|_1^2$$

when $0 < c < 1$. Suppose that $g_K^T d_K \leq -c\|g_1\|_1^2$. Note that for $k+1$ we have

$$d_{k+1} = -g_{k+1}\left[\frac{\|g_{k+1}\|_1}{\|g_k\|_1} - r\frac{|d_k^T g_{k+1}|}{\|g_k\|_1^2}\right].$$

Multiply both sides by g_{K+1}^T, then

$$g_{k+1}^T d_{k+1} = -\|g_{k+1}\|_2 + \left[\frac{\|g_{k+1}\|_1^2}{\|g_k\|_1^2} - r\frac{|g_{k+1}^T d_k|}{\|g_k\|_1^2}\right] d_k^T g_{k+1}.$$

Use the relation $\|x_k\|_2 \leq \|x_k\|_1 \leq \sqrt{n}\|x_k\|_2$ for any $X \in R^n$

$$\therefore g_{k+1}^T d_{k+1} \leq -\|g_{k+1}\|_2 + \left[\frac{\sqrt{n}\|g_{K+1}\|_2^2}{\sqrt{n}\|g_K\|_2^2} - r\frac{|g_{k+1}^T d_k|}{\sqrt{n}\|g_K\|_2^2}\right]$$

or

$$d_{k+1}^T g_{k+1} = -\|g_{k+1}\|_2^2 + \frac{\|g_{k+1}\|_2^2 d_k^T g_{k+1}}{\|g_K\|_2^2} - r\,\frac{(d_k^T g_{k+1})^2}{\sqrt{n}\|g_K\|_2^2}$$

the last term is positive. Then We omit

$$d_{k+1}^T g_{k+1} \leq -\|g_{k+1}\|_2^2 + \frac{\|g_{k+1}\|_2^2 d_k^T g_{k+1}}{\|g_K\|_2^2}$$

the fore by theorem (3.6.1) the Sufficient descent holds. To prove the global Convergence for the KH method we need the following assumption Assumption (CG)

i. Let $\mathcal{S} = \{x \epsilon R^n | f(x) \leq f(x_0)|\}$ is bounded. i. e there exists a constant $B > 0$ s.t

$$|x| \leq \mathrm{B}\ \forall\ X \in \mathcal{S} \ \ldots\ldots\ (a)$$

ii. In Some neighborhood N of $\mathcal{S}$. f is Continuously differentiable and its gradient is Lipschitz continuous i.e. there exists $L > 0$ s.t

$$\|g(x) - g(y)\| \leq L\|x - y\|. \quad \ldots\ldots (1)$$

Note that (i) and (ii) imply that there exists $r > 0$ s.t $\|g(x)\| \leq r, \forall x \in \mathcal{S}$.

Theorem 3.6.3 Suppose that assumption CG holds - Consider any Conjugate gradient method that is $(x_{k+1} = x_k + \alpha_k d_k)$ and $(d_{k+1} = g_{k+1} + \beta d_k)$ where $|\beta| \leq \beta^{FR}$ and the step-size α_K is determined by the strong wolfe line search with $0 < P < \sigma < \frac{1}{2}$. Then $liminf \|g_k\| = 0$ (Gilbert and Nocedal 1992).

Note that Theorem (3.6.2) applicable to the KH method because

$$\beta^{KH} = \frac{\|g_{k+1}\|_2^2}{\|g_k\|_1^2} - \alpha \frac{(d_k^T g_{k+1})^2}{\sqrt{n}\|g_k\|_1^2} < \frac{\|g_{k+1}\|_1^2}{\|g_k\|_1^2} = \beta^{FR}$$

and hence KH method is globally convergent $liminf \|g_k\| = 0$.

3.7 Numerical Examples

In this section we mention some examples to demonstrate the speed of convergence of the proposed algorithms and compare them with other algorithms in the same field as shown in the table below, the clear superiority of the proposed algorithms can be noticed compared to KH method to solve some Fuzzy Optimization Problems (FOP). All examples are applied with MATLAB (R2021b). The stop criterion is applied in this case $\| g_{k+1} \| < 10^{-6}$.

Iterations: - number of iterations

X_optimal:- optimal variable

F_optimal:- optimal function value

Example 3.7.1 Take a look at the following illustration of a nonlinear programming issue that involves fuzzy parameters:

$$(FOP)\min_{x\in\mathbb{R}^2} F(x)$$
$$F(x) = \left(-\frac{1}{2},\frac{1}{2},\frac{3}{2}\right)x_1^4 + \left(0,\frac{25}{2},25\right)x_1^2 + \left(-\frac{99}{4},-\frac{99}{10},\frac{99}{20}\right)x_1x_2 + (-2,2,6)x_2^2 + (-40,-16,8)x_1 + (-32,16,32).$$

It goes like this using fuzzy arithmetic:

$$\int_0^1 \left(\overline{f}^{\alpha} - \underline{f}^{\alpha}\right)(x)\, d\alpha = x_1^4 + 25x_1^2 - 19.8x_1x_2 + 4x_2^2 - 32x_1 + 16$$

$$x_0 = \begin{bmatrix}0\\3\end{bmatrix}, x^* = \begin{bmatrix}1.9585\\4.8474\end{bmatrix}$$

(Ghaznavi and Hoseinpoor 2017).

Example 3.7.2 Take a look at the following illustration of a nonlinear programming issue that involves fuzzy parameters:

$$(FOP)\min_{x\in\mathbb{R}^2} F(x),$$
$$F(x) = (2.9,3,3.15)x_1^2 + (4.7,5,5.35)x_2^2 + (-2,1,2)x_3^2.$$

It goes like this using fuzzy arithmetic:

$$\int_0^1 \left(\overline{f}^{\alpha} - \underline{f}^{\alpha}\right)(x)\, d\alpha = 6{,}025x_1^2 + 10.025x_2^2 + x_3^2$$

$$x_0 = \begin{bmatrix}1\\1\\2\end{bmatrix}, x^* = \begin{bmatrix}0\\0\\0\end{bmatrix}$$

(Ghaznavi and Hoseinpoor 2017).

Example 3.7.3 Take a look at the following illustration of a nonlinear programming issue that involves fuzzy parameters:

(FOP) $\min_{x\in\mathbb{R}^2} F(x)$

$$F(x) = (-2,1,2)x_1^4 + (-12,-4,4)x_1^3 + \left(-\frac{25}{2},\frac{25}{2},\frac{75}{25}\right)x_1^2 + (0,2,4)x_2^2 + (-6,-2,2)x_1x_2 + (-48,-16,16)x_1 + (-32,16,32).$$

It goes like this using fuzzy arithmetic:

$$\int_0^1 \left(\overline{f}^{\alpha} - \underline{f}^{\alpha}\right)(x)\, d\alpha = x_1^4 - 8x_1^3 + 25x_1^2 + 4x_2^2 - 4x_1x_2 - 32x_1 + 16$$

$$x_0 = \begin{bmatrix}1\\1\end{bmatrix},\ x^* = \begin{bmatrix}2.0451\\1.0225\end{bmatrix}$$

(Ghaznavi and Hoseinpoor 2017).

Table 3.1 Numerical results for examples

EXAMPLE	ALGORTHIMES	ITERATIONS	F_OPTIMAL	X_OPTIMAL
1	KH	52	-30.050963054721556	[1.958547737339172; 4.847405665633175]
	KM1	15	-30.050963054852538	[1.958545353387495; 4.847396030816483]
2	KH	24	1.321183162872330e-09	[2.006028929957328; 1.003014450667364]
	KM1	16	1.031921215144393e-10	[1.996812810421163; 0.998406386132323]
3	KH	42	0	[0;0;0]
	KM1	19	0	[0;0;0]

4. RESULTS AND DISCUSSION

Estimation the non-linear unconstrained optimization problem

$$minf(x), x \in \mathbb{R}^n. \tag{4.1}$$

where $f: \mathbb{R} \longrightarrow \mathbb{R}^n$ is a continuously differentiable function that is constrained from the bottom.

There are many different methods for solving the problem (4.1) (Hager and Zhang 2006, Abbo *et al*. 2016; Abbo and Khudhur 2018). Conjugate gradient (CG) algorithms are particularly appealing to us since they need less memory and have strong local and global convergence properties (Al-baali 1985 Dai and Yuan 1996; Abbo *et al*. 2016; Khudhur and Abbo 2021a; Ibraheem and Khudhur 2022, Khudhur and Ibraheem 2022). For solving the problem (4.1),we estimation the CG method , which starts from an initial point $x_1 \in \mathbb{R}^n$ and generates a sequence $\{x_k\} \in \mathbb{R}^n$ as following:

$$x_{k+1} = x_k + \alpha_k d_k \tag{4.2}$$

where $\alpha_k > 0$isastep size, recipientfromthelinesearch, and directionsdkaregivenby (Abbo and Khudhur 2015, M. Azzam *et al.* 2018, Khudhur and Abbo 2021b).

$$d_1 = -g_1 \ ,$$

$$d_{k+1} = -g_{k+1} + \beta_k d_k \tag{4.3}$$

in the equation (4.3), $g_k = f(x_k)$, $s_k = x_{k+1} - x_k$ and β_k is the conjugate gradient parameter. The standard Wolfe line search conditions are repeatedly used in the conjugate gradient methods ,the reconditions are given by (Abbas Hassan 2019)

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \delta \alpha_k g_k^T d_{k,} \tag{4.4}$$

$$g(x_k + \alpha_k d_k)^T d_k \geq \sigma g_k^T d_k, \tag{4.5}$$

where d_k is descent direction i e $g_k^T d_k < 0$ and $0 < \rho < \sigma < 1$. Strong Wolfe conditions contain of (4.4) and the next stronger version of (4.5) (Wolfe 1969, Rothwell 2017).

$$|g(x_k + \alpha_k d_k)^T d_k| \leq -\sigma g_k^T d_k \tag{4.6}$$

various choices of the scalar β_k exist which give different performance on non- quadratic functions, yet they are equivalent for quadratic functions. In order to select the parameter β_k for the method in current paper, we mention the following choices:

Table 4.1 Numerical results for examples

No	**(Formula)**	**(Author)**
1.	$\beta_k^{HS} = \frac{g_{k+1}^T y_k}{d_k^T y_k}$	(Hestenes-Stiefel (HS,1952) (Hestenes and Stiefel 1952)
2.	$\beta_k^{FR} = \frac{g_{k+1}^T g_{k+1}}{g_k^T g_k}$	(Fletcher-Reeves (FR), 1964 (Fletcher 1964)
3.	$\beta_k^{PRP} = \frac{g_{k+1}^T y_k}{g_k^T g_k}$	(Polak-Ribière (PR),1969 (Polak and Ribiere 1969)
4.	$\beta_k^{CD} = -\frac{g_{k+1}^T g_{k+1}}{d_k^T g_k}$	(Fletcher (CD),1987 (Witzgall and Fletcher 1989)

5.	$\beta_k^{LS} = -\frac{g_{k+1}^T y_k}{d_k^T g_k}$	(Liu-Storey (LS), 1991 (Liu and Storey 1991)
6.	$\beta_k^{DY} = \frac{g_{k+1}^T g_{k+1}}{d_k^T y_k}$	Dai-Yuan (DY, 1999) (Dai and Yuan 1999)
7.	$\beta_k^{DL} = \frac{y_k^T g_{k+1}}{d_k^T y_k} - t\frac{s_k^T g_{k+1}}{d_k^T y_k}$	(Dai & Liao (DL,2001) (Dai and Liao 2001)
8.	$\beta^{KH} = \frac{\|g_{k+1}\|_1^2}{\|g_k\|_1^2}$	(Hisham- Khalil (KH)) (Khudhur and Abbo 2021a)

where $y_k = g_{k+1} - g_k$. According to the balance, these algorithms may be divided into two categories: algorithms with the numerator of $g_{k+1}^T g_{k+1}$ in the numerator of β_k, and algorithms with the numerator of $g_{k+1}^T y_k$ in the numerator of parameter β_k. The first CG method using β^{FR}(FR) for a nonlinear function, which was developed by (Fletcher 1964). β^{DY} (DY) method proposed by (Dai and Yuan 1999) having strong convergence theory, but all of these methods are prone to going off the rails. With $g_{k+1}^T g_{k+1}$ in the numerator of β_k, these methods have strong convergence theory on the β^{HS} (HSCG) suggested by (Hestenes and Stiefel 1952), β^{PR} (PR) developed and β^{LS} (LS)derived by (Liu and Storey 1991), and β^{DL} (DL) suggested by (Dai and Liao 2001). A built-in restart function is available for procedures with $g_{k+1}^T y_k$ in the numerator of parameter β_k. This feature is intended to alleviate the strays' phenomena. When the step $s_k = x_{k+1} - x_k$ is tiny, the numerator of β_k factor y_k approaches 0. As a result, β_k decreases in size and the new direction d_{k+1} in (3) becomes effectively the steepest descent (SD) direction. HS, PR, DL, and LS methods automatically change β_k to avoid strays, and their performance is better than the performance of a method with $g_{k+1}^T g_k$ in the numerator of β_k, which is better (M. Abed *et al*. 2022).

4.1 New Algorithm

$$d_{k+1} = -g_{k+1} + \beta_k^{KM2} d_k \tag{4.7}$$

$$\beta_k^{KH} = \frac{\|g_{k+1}\|_1^2}{\|g_k\|_1^2}$$

$$\beta_k^{KM2} = \beta_k^{KH} - \alpha \frac{|g_{k+1}^T d_k|}{d_k^T y_k} \tag{4.8}$$

Theorem 4.1.1 Suppose that the conjugate gradient FR is implemented with the strong Wolfe line search where $0 < \sigma < \frac{1}{2}$, then FR method generates descent directions $d_k$satisfying the following

$$-\frac{1}{1-\sigma} \leq \frac{g_k^T d_k}{\|g_k\|} \leq \frac{2\sigma-1}{1-\sigma} \qquad \text{for all } \ k \geq 0$$

(Al-baali 1985).

Theorem 4.1.2 The new modified algorithm fulfills the strong Wolf conditions found in equations (4.5) and (4.6) we put the new algorithm in equation (4.2) where the parameter and search direction are calculated from equations (4.7) and (4.8) to be the new search direction are descent for all k provided $g_{n+1}^T d_{k+1} < 0$.

Proof: For $k = 0$ then $d_1^T g_1 < -\|g_1\|_2^2 < 0$. Assume that $g_k^T d_k < 0$ for $k + 1$, we have

$$g_{k+1}^T d_{k+1} = -\|g_{k+1}\|_2^2 + \left[\frac{\|g_{k+1}\|_1^2}{\|g_k\|_1^2} - \alpha \frac{|d_k^T g_{k+1}|}{d^T g}\right] d_k^T g_{k+1}$$

$$\therefore \|x\|_2 \leq \|x\|_1 \leq \sqrt{n}\|x\|_2$$

$$\therefore\ g_{k+1}^T d_{k+1} \le -\|g_{k+1}\|_2^2 + \frac{\|g_{k+1}\|_2^2 d^T g_{k+1}}{\|g_K\|_2^2} - \alpha \frac{\left(g_{k+1}^T d_{k+1}\right)^2}{d^T g}.$$

Since α is positive ant $d^T g > 0$ by wolfe line search

$$\therefore\ g_{k+1}^T d_{k+1} \le -\|g_{k+1}\|_2^2 + \frac{\|g_{k+1}\|_2^2}{\|g_K\|_2^2} d_k^T g_{k+1}.$$

By Al-Baali theorem (Al-baali 1985) $d_k^T g_{k+1} < 0$. Assumption (GG):

1. the level set $\mathcal{S} = \{x \in \mathbb{R}^n | f(x) \le f(x_1)\}$ is bounded i.e there exists a constant $B > 0$ so that $\|X\| \le B, \forall\, X \in \mathcal{S}$.

2. In some neiborhood N of S. F is continuouety differentiable and its gradient is Lipschitz ontinuous i.e there exists a constant $L > 0$ so that $\|g(x) - g(y)\| \le L\|x - y\|$ for all $x, y \in N$.

Note that from 1 and 2, ∃ a constant $\Gamma > 0$ so that $\|g(x)\| \le \Gamma$.

Theorem 4.1.3 Suppose that assumption CG holds. Concider any conjugate gradient method ($x_{k+1} = x_k + \alpha_k d_k$, $d_{k+1} = -g_{k+1} + \beta d_k d_k$) where $|\beta| \le \beta^{FR}$ and where the step-size $\alpha_k$determined by strong wolfe line search (4.5) , (4.6) with $0 < \rho < \sigma < \frac{1}{2}$ then lim inf $\|g_k\| = 0$.

4.2 Numerical Examples

In this section we mention some examples to demonstrate the speed of convergence of the proposed algorithms and compare them with other algorithms in the same field as shown in the table below, the clear superiority of the proposed algorithms can be noticed compared to other algorithm KH to solve some Fuzzy Optimization Problems (FOP). All examples are applied with MATLAB (R2009b). The stop criterion is applied in this case

$\| g_{k+1} \| < 10^{-6}$. For more on these algorithms(Steepest Descent (SD), BB1 and BB2), see (Barzilai and Borwein 1988)(M. Azzam *et al*. 2018, Abdullah *et al*. 2019, Khudhur 2015).

Iterations:- number of iterations

X_optimal:- optimal variable

F_optimal:- optimal function value

Table 4.2 Numerical results for examples

EXAMPLE	ALGORTHIMES	ITERATIONS	F_OPTIMAL	X_OPTIMAL
1	KH	25	-30.050963054721556	[1.958547737339172; 4.847405665633175]
	KM2	18	-30.050963055039965	[1.958547836345693; 4.847406123173498]
2	KH	24	1.321183162872330e-09	[2.006028929957328; 1.003014450667364]
	KM2	10	6.151524118536145e-06	[1.954216783850243; 0.977771316663116]
3	KH	42	0	[0;0;0]
	KM2	9	0	[0;0;0]

5. CONCLUSIONS AND RECOMMENDATION

In this dissertation different types of methods which are used to get the solution of fuzzy optimization problems. have been discussed. These methods are implemented by using Matlab language, and processor core i5 with Ram 4G and hard 500G in an hp pavilion laptop. The comparison between the methods was carried out depending on using the number of iterations and number of function evaluations in classical methods.

In chapter two, we have proposed new modified type of CG algorithms for solving fuzzy optimization. The numerical results indicated the efficiency of these method in solving the given nonlinear fuzzy optimization functions compared to algorithm FR.

In chapter three, we have suggested new type of CG algorithms for solving fuzzy optimization. The numerical results indicated the efficiency of these two methods in solving the given nonlinear fuzzy equations compared to algorithm FR.

1. Putting into practice the novel algorithms created in this dissertation and contrasting them with previous inexact line search methods like Backtracking or Armijo line searches.

2. The techniques that were proposed in this research can not only be used to train FFNN to solve issues involving classification or prediction, but they can also be used to train FFNN to solve problems involving other sorts of difficulties.

3. Both limited and unconstrained optimization can make use of the entire work.

4.Implementing the new algorithms developed in this study in Regression for fuzzy data.

5.The algorithms proposed in this work can be used to train Fuzzy neural networks in classification for fuzzy data.

6.Implement the modification of this study in developed Algorithms Inspired by Nature.

7.All developed algorithms can be used to solve fuzzy optimization problems and solve nonlinear fuzzy equations.

REFERENCES

Abbas Hassan, B. 2019. "A new formula for conjugate parameter computation based on the quadratic model", Indonesian Journal of Electrical Engineering and Computer Science, 13(3), p. 954. doi: 10.11591/ijeecs.v13.i3.pp954-961.

Abbo, K. K. and Khudhur, H. M. 2015. "New A hybrid conjugate gradient Fletcher-Reeves and Polak-Ribiere algorithm for unconstrained optimization", Tikrit Journal of Pure Science, 21(1), pp. 124–129.

Abbo, K. K. and Khudhur, H. M. 2018 "New A hybrid Hestenes-Stiefel and Dai-Yuan conjugate gradient algorithms for unconstrained optimization", Tikrit Journal of Pure Science, 21(1), pp. 118–123.

Abbo, K. K., Laylani, Y. A. and Khudhur, H. M. 2016 "Proposed new Scaled conjugate gradient algorithm for Unconstrained Optimization", International Journal of Enhanced Research in Science, Technology & Engineering, 5(7).

Abbo, K. K., Laylani, Y. A. and Khudhur, H. M. (2018) "A New Spectral Conjugate Gradient Algorithm For Unconstrained Optimization", International Journal of Mathematics and Computer Applications Research (IJMCAR), 8, pp. 1–9.

Abdullah, Z. M. *et al.* (2019) "Modified new conjugate gradient method for Unconstrained Optimization", Tikrit Journal of Pure Science, 24(5), pp. 86–90.

Ahmed, A. S., Khudhur, H. M. and Najmuldeen, M. S. (2021) "A new parameter in three-term conjugate gradient algorithms for unconstrained optimization", Indonesian Journal of Electrical Engineering and Computer Science, 23(1), p. 338. doi: 10.11591/ijeecs.v23.i1.pp338-344.

Al-baali, M. (1985) "Descent property and global convergence of the fletcher-reeves method with inexact line search", IMA Journal of Numerical Analysis, 5(1), pp. 121–124. doi: 10.1093/imanum/5.1.121.

Antoniou, A. and Lu, W. S. (2007) Practical Optimization, Practical Optimization: Algorithms and Engineering Applications. Boston, MA: Springer US. doi: 10.1007/978-0-387-71107-2.

Arana-Jiménez, M. *et al*. (2015) "Generalized convexity in fuzzy vector optimization through a linear ordering", Information Sciences, 312, pp. 13–24. doi: 10.1016/j.ins.2015.03.045.

Barzilai, J. and Borwein, J. M. (1988) “Two-point step size gradient methods”, IMA Journal of Numerical Analysis, 8(1), pp. 141–148. doi: 10.1093/imanum/8.1.141.

Bazaraa, M. S., Sherali, H. D. and Shetty, C. M. (2013) Nonlinear programming: theory and algorithms. John Wiley & Sons.

Bector, C. R. and Chandra, S. (2005) Fuzzy mathematical programming and fuzzy matrix games, Choice Reviews Online. Springer. doi: 10.5860/choice.43-0362.

Bede, B. and Stefanini, L. (2013) “Generalized differentiability of fuzzy-valued functions”, Fuzzy Sets and Systems, 230, pp. 119–141. doi: 10.1016/j.fss.2012.10.003.

Bellman Re And Zadeh La (1970) “Decision-Making In A Fuzzy Environment”, Management Science, 17(4), pp. 91–126. doi: 10.1142/9789812819789_0004.

Chalco-Cano, Y., Silva, G. N. and Rufián-Lizana, A. (2015) “On the Newton method for solving fuzzy optimization problems”, Fuzzy Sets and Systems, 272(1), pp. 60–69. doi: 10.1016/j.fss.2015.02.001.

Chaturvedi, D. K. (2017) Modeling and simulation of systems using MATLAB® and simulink®, Modeling and Simulation of Systems Using MATLAB and Simulink. doi: 10.1201/9781315218335.

Dai, Y. H. and Liao, L. Z. (2001) “New conjugacy conditions and related nonlinear conjugate gradient methods”, Applied Mathematics and Optimization, 43(1), pp. 87–101. doi: 10.1007/s002450010019.

Dai, Y. H. and Yuan, Y. (1996) “Convergence properties of the Fletcher-Reeves method”, IMA Journal of Numerical Analysis, 16(2), pp. 155–164. doi: 10.1093/imanum/16.2.155.

Dai, Y. H. and Yuan, Y. (1999) “A nonlinear conjugate gradient method with a strong global convergence property”, SIAM Journal on Optimization, 10(1), pp. 177–182. doi: 10.1137/S1052623497318992.

Dixon, L. C. W. (1975) “Conjugate gradient algorithms: Quadratic termination without linear searches”, IMA Journal of Applied Mathematics (Institute of Mathematics and Its Applications), 15(1), pp. 9–18. doi: 10.1093/imamat/15.1.9.

Dubois, D. and Prade, H. (1978) “Operations on fuzzy numbers”, International Journal of Systems Science, 9(6), pp. 613–626. doi: 10.1080/00207727808941724.

Fletcher, R. (1964) “Function minimization by conjugate gradients”, The Computer

Journal, 7(2), pp. 149–154. doi: 10.1093/comjnl/7.2.149.

Ganesan, K. and Veeramani, P. (2006) “Fuzzy linear programs with trapezoidal fuzzy numbers”, in Annals of Operations Research. Springer, pp. 305–315. doi: 10.1007/s10479-006-7390-1.

Ghaznavi, M. and Hoseinpoor, N. (2017) “A quasi-Newton method for solving fuzzy optimization problems”, Journal of Uncertain Systems, 11(1), pp. 3–17.

Gilbert, J. C. and Nocedal, J. (1992) “Global Convergence Properties of Conjugate Gradient Methods for Optimization”, SIAM Journal on Optimization, 2(1), pp. 21–42. doi: 10.1137/0802003.

Hager, W. W. and Zhang, H. (2006) “A new conjugate gradient method with guaranteed descent and an efficient line search”, SIAM Journal on Optimization, 16(1), pp. 170–192. doi: 10.1137/030601880.

Hestenes, M. R. and Stiefel, E. (1952) “Methods of conjugate gradients for solving linear systems”, Journal of Research of the National Bureau of Standards, 49(6), p. 409. doi: 10.6028/jres.049.044.

Ibraheem, K. I. and Khudhur, H. M. (2022) “Optimization Algorithm Based On The Euler Method For Solving Fuzzy Nonlinear Equations”, Eastern-European Journal of Enterprise Technologies, 1(4–115), pp. 13–19. doi: 10.15587/1729-4061.2022.252014.

Khudhur, H. M. (2015) Numerical and analytical study of some descent algorithms to solve unconstrained Optimization problems, University of Mosul college Computer Sciences and Mathematics Department of Mathematics Iraq. University of Mosul.

Khudhur, H. M. and Abbo, K. K. (2020) “A New Conjugate Gradient Method for Learning Fuzzy Neural Networks”, Journal of Multidisciplinary Modeling and Optimization, 3(2), pp. 57–69.

Khudhur, H. M. and Abbo, K. K. (2021a) “A New Type of Conjugate Gradient Technique for Solving Fuzzy Nonlinear Algebraic Equations”, Journal of Physics: Conference Series, 1879(2), p. 022111. doi: 10.1088/1742-6596/1879/2/022111.

Khudhur, H. M. and Abbo, K. K. (2021b) “New hybrid of Conjugate Gradient Technique for Solving Fuzzy Nonlinear Equations”, Journal of Soft Computing and Artificial Intelligence, 2(1), pp. 1–8.

Khudhur, H. M. and Ibraheem, K. I. (2022) “Metaheuristic optimization algorithm based on the two-step Adams-Bashforth method in training multi-layer perceptrons”, Eastern-European Journal of Enterprise Technologies, 2(4 (116)), pp. 6–13. doi: 10.15587/1729-4061.2022.254023.

Kulkarni, A. J., Krishnasamy, G. and Abraham, A. (2017) “Introduction to Optimization”, in Intelligent Systems Reference Library, pp. 1–7. doi: https://dl.acm.org/doi/book/10.5555/38120.

Laylani, Y. A., Abbo, K. K. and Khudhur, H. M. (2018) “Training feed forward neural network with modified Fletcher-Reeves method”, Journal of Multidisciplinary Modeling and Optimization, 1(1), pp. 14–22. Available at: http://dergipark.gov.tr/jmmo/issue/38716/392124#article_cite.

Lee, K. H. (2005) “First course on fuzzy theory and applications”, Choice Reviews Online, 42(10), pp. 42-5917-42–5917. doi: 10.5860/choice.42-5917.

Lenir, A. (1981) “Variable Storage CG-Methods”, University of leed, Dep. Of Computer Studies, M.Sc. Thesis.

Liu, Y. and Storey, C. (1991) “Efficient generalized conjugate gradient algorithms, part 1: Theory”, Journal of Optimization Theory and Applications, 69(1), pp. 129–137. doi: 10.1007/BF00940464.

Lupulescu, V. (2013) “Hukuhara differentiability of interval-valued functions and interval differential equations on time scales”, Information Sciences, 248(3–4), pp. 50–67. doi: 10.1016/j.ins.2013.06.004.

M. Abed, M., Öztürk, U. and M. Khudhur, H. (2022) “Spectral CG Algorithm for Solving Fuzzy Nonlinear Equations”, Iraqi Journal for Computer Science and Mathematics, 3(1), pp. 1–10. doi: 10.52866/ijcsm.2022.01.01.001.

M. Azzam, H., N. Jabbar, H. and K. Abo, K. (2018) “Four–Term Conjugate Gradient (CG) Method Based on Pure Conjugacy Condition for Unconstrained Optimization”, Kirkuk University Journal-Scientific Studies, 13(2), pp. 101–113. doi: 10.32894/kujss.2018.145720.

Maleki, H. (2002) “Ranking functions and their applications to fuzzy linear programming”, Far East Journal of Mathematical Sciences, 4, pp. 283–301.

Mascarenhas, W. F. (2004) “The BFGS method with exact line searches fails for non-convex objective functions”, Mathematical Programming, 99(1), pp. 49–61. doi:

10.1007/s10107-003-0421-7.

Moguerza, J. M. and Prieto, F. J. (2003) “Combining search directions using gradient flows”, Mathematical Programming, 96(3), pp. 529–559. doi: 10.1007/s10107-002-0367-1.

Nocedal, J. and Wright, S. J. (2006) “Numerical optimization”, in Springer Series in Operations Research and Financial Engineering. Springer Science & Business Media, pp. 1–664. doi: 10.1201/b19115-11.

Pirzada, U. M. and Pathak, V. D. (2013) “Newton Method for Solving the Multi-Variable Fuzzy Optimization Problem”, Journal of Optimization Theory and Applications, 156(3), pp. 867–881. doi: 10.1007/s10957-012-0141-3.

Polak, E. and Ribiere, G. (1969) “Note sur la convergence de méthodes de directions conjuguées”, Revue française d”informatique et de recherche opérationnelle. Série rouge, 3(16), pp. 35–43. doi: 10.1051/m2an/196903r100351.

Powell, M. J. D. (1977) “Restart procedures for the conjugate gradient method”, Mathematical Programming, 12(1), pp. 241–254. doi: 10.1007/BF01593790.

Rothwell, A. (2017) “Numerical methods for unconstrained optimization”, in Solid Mechanics and its Applications. Van Nostrand Reinhold New York, pp. 83–106. doi: 10.1007/978-3-319-55197-5_4.

Scales, L. E. (1985) Introduction to non-linear optimization. Macmillan International Higher Education.

Stefanini, L. (2010) “A generalization of Hukuhara difference and division for interval and fuzzy arithmetic”, Fuzzy Sets and Systems, 161(11), pp. 1564–1584. doi: 10.1016/j.fss.2009.06.009.

Steihaug, T. and Hestenes, M. (1982) “Conjugate Direction Methods in Optimization.”, Mathematics of Computation, 38(157), p. 332. doi: 10.2307/2007488.

Sun, W. and Yuan, Y.-X. (2006) Optimization Theory and Methods, Optimization Theory and Methods. Springer Science & Business Media. doi: 10.1007/b106451.

Tanaka, H., Okuda, T. and Asai, K. (1973) “On fuzzy-mathematical programming”, Journal of Cybernetics, 3(4), pp. 37–46. doi: 10.1080/01969727308545912.

Viviani, G. L. (1985) “Practical Optimization”, IEEE Power Engineering Review, PER-5(7), p. 33. doi: 10.1109/MPER.1985.5528460.

Witzgall, C. and Fletcher, R. (1989) “Practical Methods of Optimization”, Mathematics of Computation, 53(188), p. 768. doi: 10.2307/2008742.

Wolfe, P. (1969) “Convergence Conditions for Ascent Methods”, SIAM Review, 11(2), pp. 226–235. doi: 10.1137/1011036.

CURRICULUM VITAE

Personal Information

Name and Surname : Mohammed Msallam Mustafa MUSTAFA

Education

MSc	Çankırı Karatekin University Graduate School of Natural and Applied Sciences Department of Mathematics	2020-2023
Undergraduate	Al-Mustansiriya University, College of Geosciences	2016