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

USING THE DEVELOPED CONJUGATE GRADIENT METHOD TO CREATE AN AUXILIARY FUNCTION FOR SOLVING GLOBAL OPTIMIZATION PROBLEMS

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

BY

MYASAR SHIHAB HAMAD HAMAD

ÇANKIRI

2023

USING THE DEVELOPED CONJUGATE GRADIENT METHOD TO CREATE AN AUXILIARY FUNCTION FOR SOLVING GLOBAL OPTIMIZATION PROBLEMS

By Myasar Shihab Hamad- HAMAD

September 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 Science

Advisor **:** Prof. Dr. Faruk POLAT
Co-Advisor **:** Asst. Prof. Dr. Yuksal Abdusattar LAYLANİ

Examining Committee Members:

Chairman **:** Prof. Dr. Faruk POLAT
Department of Mathematics
Çankırı Karatekin University

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

Member **:** Asst. Prof. Dr. Uğur GÖNÜLLÜ
Department of Mathematics and Computer Sciences
İstanbul Kültür 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.

Myasar Shihab Hamad HAMAD

ABSTRACT

USING THE DEVELOPED CONJUGATE GRADIENT METHOD TO CREATE AN AUXILIARY FUNCTION FOR SOLVING GLOBAL OPTIMIZATION PROBLEMS

Myasar Shihab Hamad HAMAD

Master of Science in Mathematics

Advisor: Prof. Dr. Faruk POLAT

Co-Advisor: Asst. Prof. Dr. Yuksal Abdusattar LAYLANİ

September 2023

An alternative approach to solve global optimization challenges is the filled function technique. This study presents a fresh take on the method by proposing the creation of a new filled function that is continuously differentiable and has a single parameter. This approach involves the discovery of an objective function minimizer, which then sets the stage for unbridled global optimization. The proposed filled function approach was shown to be effective in numerical experiments. By repeatedly seeking a superior minimizer in a reduced basin of the filled function, an advanced solution to the objective function can be discovered. This process continues until the optimal solution is reached.

2023, 65 pages

Keywords: Conjugate gradient method, Global optimization, Auxlilary function, Alternative minimizers, Filled function.

ÖZET

GLOBAL OPTİMİZASYON PROBLEMLERİNİ ÇÖZMEK İÇİN BİR YARDIMCI FONKSIYON KULLANAN GELİŞTİRİLMİŞ EŞLENİK GRADYAN YÖNTEMİNİ KULLANMA

Myasar Shihab Hamad HAMAD
Matematik, Yüksek Lisans
Tez Danışmanı: Prof. Dr. Faruk POLAT
Eş Danışman: Dr. Öğr. Üyesi Yuksal Abdusattar LAYLANİ
Eylül 2023

Küresel optimizasyon zorluklarına alternatif bir yaklaşım, doldurulmuş fonksiyon tekniğidir. Bu çalışma, sürekli türevlenebilir bir tek parametreye sahip yeni bir doldurulmuş fonksiyonun oluşturulmasını önererek bu yönteme yeni bir yaklaşım sunmaktadır. Bu yaklaşım, bir amaç fonksiyonunun minimize edicisini bulmanın keşfine dayanmaktadır ve bu da sınırsız küresel optimizasyon için zemin hazırlar. Önerilen doldurulmuş fonksiyon yaklaşımının sayısal deneylerde etkili olduğu gösterilmiştir. Doldurulmuş fonksiyonun küçültülmüş bir havzasında sürekli olarak daha iyi bir minimizmize edici arayarak, amaç fonksiyonunun gelişmiş bir çözümü keşfedilebilir. Bu süreç, optimal çözüme ulaşılana kadar devam eder.

2023, 65 sayfa

Anahtar Kelimeler: Eşlenik gradyan yöntemi, Global optimizasyon, Yardımcı fonksiyon, Alternatif minimizeciler, Doldurulmuş fonksiyon.

PREFACE AND ACKNOWLEDGEMENTS

I would like to thank my thesis advisor, Prof. Dr. Faruk POLAT, for his patience, guidance and understanding.

Myasar Shihab Hamad HAMAD

Çankırı-2023

CONTENTS

LIST OF SYMBOLS

B_k	Basin of f
D	Domain
D_c	Feasible set
x^*	Global minimizer
$\nabla H_k(x)$	Gradient of $H_k(x)$
B_k	HSI matrix
x_0	Initial guess
$N_\varepsilon(x_k^*)$	e-neighborhood
f	Objective function
$f(.)$	Quadratic function
$\mathbb{R}$	Set of real numbers
S_k	Simple basin of f
d_k	Slope direction
(p)	The objective of problem

LIST OF ABBREVIATIONS

DEs	Differential equations
ILS	Incorrect search line
QN	Quasi-newton

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

Various problems call for optimization so that the best possible outcome, be it minimum or maximum, can be achieved. Some problems have simple functions while others involve complex equations and multiple properties. One can take a local or global approach when searching for the optimal value. Local search focuses on discovering optimal values among neighboring sets of possible solutions, while global search seeks the best possible value of a function over the entirety of its domain (Brinkhuis and Tikhomirov 2005).

Real-world problems can be tackled with the help of global optimization problems. These flexible problem-solving tools find practical applications in a multitude of fields such as rocket, dam, bridge, and architectural design. Furthermore, it is also embraced in the computer and information technology sector where it plays a vital role in areas like network design, artificial intelligence, database design, and image processing. One can also derive immense benefits in the economic aspect of things like business prediction and electronic commerce with the use of global optimization problems. Agricultural production structure and spatial distribution of crops are just some of the uses of these versatile instruments. We can also find them in chemistry, physics (metal matrix composition and nanotechnology), geography (earthquake and weather prediction), and even in vacation planning and social life.(Hartikainen *et al.* 2012).

Due to the non-smooth structure of global optimization problems and the increasing number of variables, their complexity has been on the rise over time. The presence of many local minimizers adds to the challenge of finding a solution. Consequently, global optimization encounters several roadblocks that make it difficult to overcome: (Floudas and Pardalos 2008).

a- Override your current local minimizer technique and locate a local minimizer in your area using any method available.

b- Alternative minimizers for the objective function can be found by disregarding local minimizers whose values are higher than the current minimizer. This is the way to do it.

c- To determine when to stop and assess the convergence to the overall minimum, multiple steps must be taken. One significant step is analyzing the variance between the function values from the current and past iterations and comparing them with a predetermined tolerance level. It's also necessary to meticulously evaluate the alteration in the input variables. Plus, it's crucial to keep track of the divergence from the ideal solution. Once these criteria have been met, it confirms that the global minimizer has been attained.

In the world of computerized puzzle solving, universally resolving every issue is not always doable. As a result, compromises are often necessary. Individuals who specialize in worldwide optimization options can adjust formulae to match specific puzzles, which advances the sector. Despite typical algorithms aiming for accuracy and timeliness, it is challenging to come up with fresh approaches for solving specific problems, and non-traditional techniques may offer better results. Many contemporary scientific fields utilize worldwide optimization tactics to overcome a diversity of hurdles. Throughout the course of history, scientists from various fields have faced global optimization problems, but these challenges were largely dismissed until the 1970s. It's possible that the absence of advanced theories and localized optimization techniques, coupled with the monumental computational difficulties associated with these problems are some factors that prevented their recognition. Furthermore, no suitable frameworks or norms existed prior to the 1970s. However, with the passage of time, computers have become increasingly adept at dealing with complicated issues previously thought impossible to tackle. As a result of this significant change, the global optimization sphere has expanded dramatically. Introducing readers to the fundamentals of global optimization, this particular text takes a look at problem properties, essential preliminary measures, and effective optimization approaches that can lead to solutions. The first section aims to capture the primary elements of a global optimization problem, while the second segment showcases the different types of available approaches for global optimization. With Section 1.1 defining

the essence for the global optimization issue, Section 1.2 proposes a ranking system for various global optimization methods. To complete the trifecta, Section 1.3 tackles the fundamentals of objective function formulation. We take a closer look at various methods for achieving global optimization as outlined in Section 1.4, Thesis Overview: In Part 1.5, Finally, we give a local search method in Section 1.6.

1.1 Global Optimization Formula

Locating a global minimizer of a quantity is primary goal of global optimization. The usual procedure for this entails the following formula: (Floudas and Pardalos 2008).

$$(Q) = min_{x \in D} f(x)$$

To describe the objective of the problem, the term f is utilized whilst the domain of possible vectors $x = (x_1, x_2 \ldots\ldots x_n)$, that meet specific prerequisites, is represented by $\mathbb{D} \subset \mathbb{R}^n$. The objective of issue (Q) is to discover value of the global minimizer, x^*,which is specified as:

$$f(x^*) = min_{x \in D} f(x)$$

Many new theories and techniques for dealing with it were offered issues (Q). Typically, global optimization methods are partitioned into three categories.

There are three types of problem-solving approaches: predictable techniques, probabilistic techniques, and heuristic techniques.

1.2 Global Optimization Classification

The goal of the functional f and the area D, which stands for the search region, serve as the foundation of definition of the global optimization issue. Once the entire D is used to

find the global minimizer x^*, matching to the overall domain definition, the problem (Q) itself can be categorized as unconstrained global optimization. (Horst and Pardalos 2013).

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

If the goal is to minimize the objective function while satisfying constraints, problem (Q) can be classified as a constrained global optimization problem for the variable x within the domain D. Typically, these types of problems are formulated using a specific mathematical expression. This formulation is commonly employed to address such problems. (Horst and Pardalos 2013).

$$\min_{x \in D_c} f(x)$$

governed by

$$\begin{cases} g_i(x) = 0\,, & i \in A_1 \\ g_i(x) \geq 0\,, & i \in A_2 \end{cases}$$

Finite sets A_1, A_2, differentiable functions f and $g_i(x)$ twice each, are what we're dealing with. It's important to note that f is the objective function, as previously stated.

$$g_i(x) = 0\,, \quad i \in A_1$$

are there restrictions on equal as well as

$$g_i(x) \geq 0\,, \quad i \in A_2$$

Inequality constraints introduce restrictions on the feasible solutions. As a result, the set D_c, which fulfills these constraints, is regarded as the feasible set. (Horst and Pardalos 2013)

$$D_c = \{x: g_i(x) = 0, i \in A_1\,; g_i(x) \geq 0, i \in A_2\}$$

D_c , $\mathbb{R}^n$ are the sets for the function $f : D_c \to R$ referred to as a constrained global optimization problem, this can be represented by the following expression.

$$\min_{x \in D_c} f(x)$$

As a globally unconstrained optimization, the issue we tackle in this dissertation is approached problem.

1.3 Objective Function Form

The objective function and search space are two key components of a global optimization problem, as previously noted. Function f's qualities have yielded four distinct components that form the objective function: (Horst and Pardalos 2013)

1- Continuity

2- Convexity

3- Differentiability

4- Smoothness

1. **Continuity**: A purposeful function h: $\mathbb{D} \longrightarrow \mathbb{R}$ is said to be continuous at $x = c \in \mathbb{D}$ if

$$\max_{x \to c} f(x) = f(c)$$

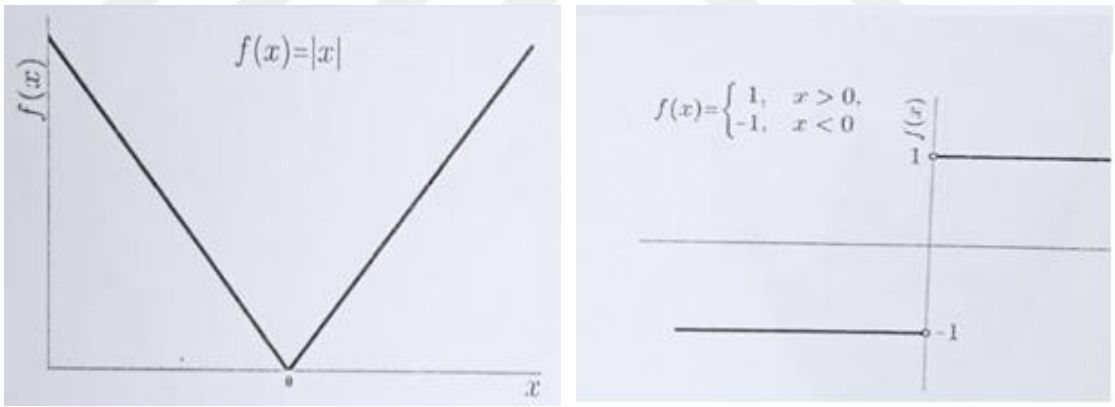
If it is true at every point in its domain, it follows that the objective function is thought of as continuous. The following conditions must be met for a function $f(x)$ to be continuous at a vector where $x = c$.

- $f(x)$ is specified
- $\max_{x \to c} f(x)$ occures
- $\max_{x \to c} f(x) = f(c)$

f is deemed interrupted when the conditions listed above are not met, as depicted in Fig.1.1. (Horst and Pardalos, 2013)

2. Convexity: A function $f: \mathbb{D} \to \mathbb{R}$ On an area of $\mathbb{D} \subset \mathbb{R}$, it's considered to have been convex if, for every $x_1, x_2 \in \mathbb{R}$ and $\lambda \in (0,1)$, the function is.

$$f(\lambda x_1 + (1-\lambda)x_2) \leq f(\lambda x_1) + (1-\lambda) f(x_2)$$



(**a**) **Continuous** (**b**) **Discontinuous**

Figure 1.1 The Functions of **continuous and discontinuous**

In Figure 1.2 (Horst and Pardalos 2013), the non-convexity of the function f can be observed if the inequality fails. When dealing with a convex objective function, finding a local optimum is equivalent to finding the global optimum, eliminating the need for

multiple local methods. The issue arises when there are multiple local extrema due to non-convexity, making it necessary to explore the entire domain rather than just local areas. In such cases, global methods become essential for finding the global optimum point as relying solely on local methods is no longer adequate.

In D, when the derivative of the objective function $f: \mathbb{D} \to \mathbb{R}$ exists at c, it is considered differentiable. To deduce details about the local maximizer, stationary point and minimizer, the gradient direction can be referenced. Optimization solutions benefit from this gradient info as it helps locate the minimizer swiftly. Numerous techniques utilize the gradient to determine local minimizer; a few examples are gradient-descent and Newton's method. Twice continuously differentiable function $f: \mathbb{D} \to \mathbb{R}$ can provide us with structural information on optimal solutions via first and second-order optimality conditions. A case in point is when a local minimizer of x_k^* satisfies the initial requirement.

$$\nabla f(x_k^*) = 0$$

According to the second-rank circumstance, if $x_k^* \in \mathbb{D}$ is a local reducer of f, thus

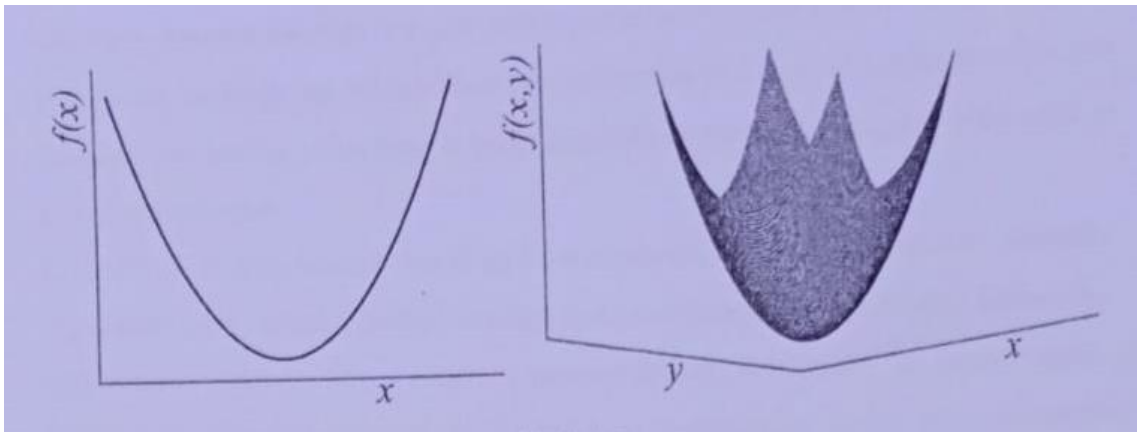
$$\nabla f(x_k^*) = 0$$

Also

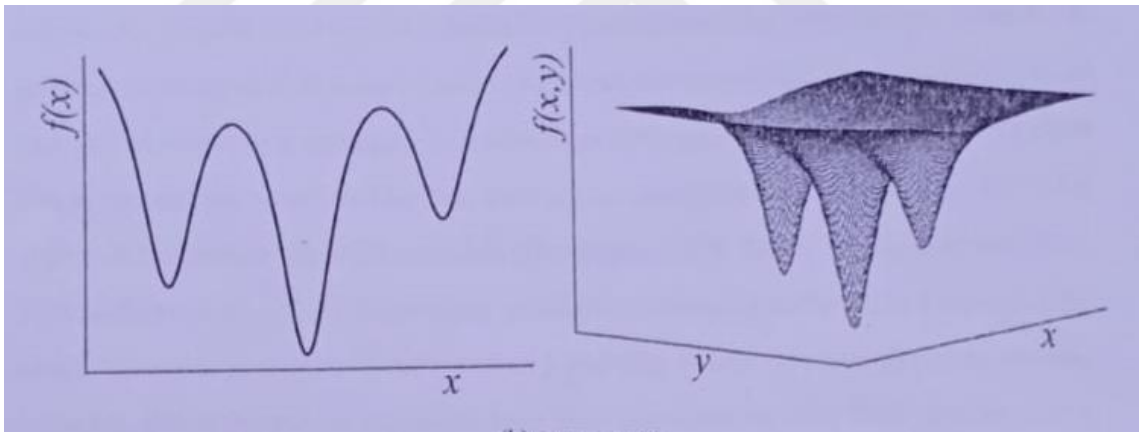
$$\nabla^2 f(x_k^*) \geq 0,$$

Positive semi-definite tells us that the Hessian at x_k^* is positive. To locate the local minimizer value, certain global optimization approachs depend on the gradient information for objective function.

Using a technique called deterministic methods, the gradient can be utilized to locate a local goal functional minimizer. A function of auxiliary is then employed to determine the global minimizer value. Details regarding this process will be provided later on.



(a) Convex



(b) Non-Convex

Figure 1.2 The Functions of convex and nonconvex

1.4 Different Approaches to Solving Global Optimization Problems

Throughout the last few decades, global optimization problems have become a focal point in numerous important applications. To locate the optimal global value, several techniques can be utilized, which fall within three general categories: deterministic,

stochastic, and heuristic. These methodologies provide several options to achieve an accurate outcome.

1.4.1 Stochastic Methods

In order to find the global value, stochastic methods rely on probabilistic tools and the creation of a multitude of randomly placed points within the feasible domain. Local search techniques are then performed on some of these points until the global value is reached. Despite being straightforward and effective in black box problems with an increase in problem dimensions, stochastic methods can be reliable. However, it is important to note that certain stochastic methods may only yield local solutions rather than global ones. Among the many techniques used in optimization, there are some widely recognized stochastic approaches like Markovian algorithms, Adaptive search, and Random search. These techniques have been extensively documented in papers such as Huyer and Neumaier's work from 1999, Schäffler's 2012 publication, and Anderssen and Bloomfield's 1975 study. Additionally, there are population algorithms that use stochastic methods which are discussed under heuristic methods.

1.4.2 Heuristic methods

Several optimization algorithms have become immensely popular in current times for simulating biological, physical, or chemical processes. These methods form the basis for heuristic methods, which offer rapid solutions and effortless application. However, there is a possibility of different outcomes when these techniques are executed repeatedly. Some of the widely accepted algorithms include Suman and Kumar's algorithm of Simulated Annealing developed in 2006, Storti and team's Genetic Algorithm of 2015 , Ekren and Ekren's algorithms of 2010, and Samora's team's algorithm in 2016. Karaboga and Basturk in 2007, as well as Akay et al., uncovered the Artificial Bee Colony algorithm in 2007 and 2012, while Poli and Kennedy developed the Particle Swarm Optimization algorithm in 2007, which have both demonstrated to be highly successful. Combining these heuristic global optimization methods is a recent movement in this area, with Mirjalili *et al.* in 2016, Zhang *et al.* in 2016, Arora and Anand in 2019, Stork *et al.* in

2020, Wang *et al.* in 2022, Chakraborty *et al.* in 2023 all achieving substantial triumphs in amalgamating these algorithms.

1.4.3 Deterministic methods

Efficient deterministic approaches exist for solving problems, although they often involve higher computation costs. While stochastic and heuristic methods are less reliable in some ways, they can be fast and easy to use. Examples of deterministic strategies are as follows:

The foundational idea behind Branch and Bound methods is to partition the feasible region and set predetermined lower bounds for the objective function. By doing so, any non-viable partitions can be eliminated. Subsequently, the remaining subregions are explored in the same manner and further partitioned using the established lower bounds. Employed to assess function objectives limit reduction, expand and glueing approaches are varied through the use of different approaches. One such approach is the primal-dual branch bound method, this was suggested by Floudas and Visweswaran (1990, 1993), Androulakis et al. (1995) and Adjiman et al. (1998) to grant lower boundaries.

In 2003, Hansen and Walster introduced the interval arithmetic method as a method of using interval branch and bound techniques, followed by Alefeld and Herzberger in 2012 and Grimstad and Sandnes in 2016.

Auxiliary function methods**:** Using predetermined constraints on the target function, the Branch and Bound technique creates partitions within the attainable space. Improbable partitions are then eliminated, leaving behind the ones that are more likely to succeed. Systematic exploration is then applied to these remaining regions, with smaller partitions using established constraints and similar methods. The fundamental principles of this method are based on the elimination of unlikely partitions and the exploration of the remaining regions. Establishing lower boundaries for objective functions can be tackled through a range of approaches of branching and glueing to guarantee variety. The primal-dual branching constrained strategy, which was initially offering by Floudas and

Visweswaran in 1990 and 1993, Androulakis et al. in 1995, and Adjiman et al. in 1998, is just one of the methods that can be deployed.

In 2003, Hansen and Walster, alongside Alefeld and Herzberger in 2012, introduced the interval arithmetic approach which is utilized by the branch and bound techniques that Grimstad and Sandnes brought to light in 2016.

The quantity of minimizers can be decreased through reiterating steps 2 and 3, ultimately leading to the discovery of the global minimizer.

Global optimization can be approached through auxiliary function methods, developed with deterministic search strategies in mind. These methods construct an alternate function to discover an improved locally minimizer than one that we have now. The technique of filled functions (Renpu 1990, Ge and Qin 1987, Liu 2001), tunneling methods (Montalvo and Levy 1985), and approach of global descending (Wu *et al.* 2011) are included within this essential category of methods.

First proposed by Levy and Montalvo in 1985, tunneling algorithms proved to be a crucial innovation in resolving constrained global optimization difficulties. Subsequent research and essential analysis contributing to the development of tunneling algorithms have been conducted and disseminated by Groenen and Heiser (1996), Chowdhury *et al.* (2000), and Xu *et al.* (2015).

When it comes to addressing optimization challenges on a global scale, a variety of methods exist, but one that particularly shines is the approach of using filled functions, which has several benefits. Firstly, this method offers a relatively easier means of gradually discovering a local minimum that surpasses those discovered through other approaches. The original introduction of filled functions dates back to 1987 by Ge and Qin and 1990 by Renpu. However, over the years, this method has been further refined by Wu *et al.* in both 2005 and 2007, as well as by Xu *et al.* in 2001. Filled functions have been heavily analyzed and developed in modern times as a way to enhance their versatility

in addressing a wide array of issues. Among others, the filled function has been put to the test in solving problems with nonlinear equations, constrained optimization, and non-smoothness. Several researchers, including (Zhang *et al.* 2009, Sahiner *et al.* 2012, Wang *et al.* 2015, Yuan *et al.* 2016 b, and Wei *et al.* 2014), have made noteworthy advancements in this area. The filled function field has advanced with the subsequent development of its next-gen and auxiliary function approaches, ensuring that it stays current. Introducing novel approaches to multi-model global optimization problems, we draw inspiration from the pioneering work of (Yilmaz and Sahiner 2017, Sahiner *et al.* 2017, Lin *et al.* 2018, Liu *et al.* 2017, and Sahiner and Ibrahem 2019). In order to achieve this advancement, we are presenting functions of auxiliary that differ by location within our thesis. Crucial techniques for deterministic strategies include Glazing techniques (Jones *et al*. 1993), techniques for filled space using curves (Lera and Sergeyev 2015, Ziadi *et al.* 2016), and a variety of alternative approaches (Basso 1982).

1.5 Thesis Overview

Next up in this thesis are a variety of topics that have been arranged in no particular order. First, Part 3 introduces a novel non-smooth functional globally smoothness approach which will be useful in Section 4 in solving the (P) problem. Before that, however, in Section 2, definitions and assumptions related to the (P) problem will be explored. Finally, Section 4 offers a detailed description of a new algorithm designed to address the (P) problem. This innovative method converts multi-dimensional problems into one-dimensional versions by partitioning. By using an auxiliary function, the global minimizers for each partition are determined which helps to identify the global minimizer for the multidimensional problem across these partitions. One way to solve problem (P) in Section 4 involves using dual parameters with a filled function method that places various stationary points in the lower basin.

1.6 Local Search Methods

1.6.1 Line search methods

The search line is the backbone of many algorithms used to solve nonlinear programming problems. The search line method begins with an initial guess x_0, the search direction is calculated d_k. Then decide how we move along that direction and iteration is: (Nocedal and Wright 2006).

$$x_{k+1} = x_k + \alpha_k d_k \tag{1.1}$$

So $\alpha_k > 0$ It stands for the step size. The effective (wise) selection of each direction $d_k$and stride length$\alpha_k$is key to the research line method's effectiveness in (1.1).

By satisfying the slope condition, the majority of search line algorithms demand that the trend be d_k slope direction.

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

for every k Because this property guarantees that the function f It will decrease along this direction in (1.2). There are several ways to choose the search direction. For example, you can choose

$$d_k = -B_k^{-1} g_k \tag{1.3}$$

As if B_k It represents a symmetrical and non-singular matrix. In the steep decline method B_k represents the unary matrix I while in Newton's method B_k represents the HSI matrix. In Newton-like ways B_k It is an approximation of the HSI matrix and is recalculated at each iteration. when the trend d_k defined according to the equation (1.3) and B_k positively defined matrix, we get

$$d_k^T g_k = -g_k^T B_k^{-1} g_k < 0$$

So d_k is a downward trend.

1.6.1.1 Step size

Exact search line strategy aims to comprehensively reduce the one-dimensional function and is defined as follows:

$$\emptyset(\alpha) = f(x_k + \alpha d_k), \qquad \alpha > 0 \tag{1.4}$$

This is not acceptable in practice. Most of the required strategies need a non-exact search line to distinguish the step size α_k which gives a decrease in the value of the function f At the lowest cost, but at the same time it does not take a very long time to complete this choice. Incorrect search line (ILS) which tests a series of values α_k It stops when a number of conditions are met. We will deal with $\emptyset$ Likewise with the derivative $\emptyset'$ As if $\emptyset'(\alpha)$ The directional derivative means at $x_k + \alpha d_k$ in (1.4) which is represented by the following equation (1.5) and (1.6).

$$\emptyset'(\alpha) = d_k^T \nabla f(x_k + \alpha d_k) = d_k^T g_{k+1} \tag{1.5}$$

$$\emptyset'(0) = d_k^T g_k \tag{1.6}$$

The search line operates in two stages: the first is the Bracketing Phase, which is the period that contains the appropriate stride lengths, and the second is the Refinement Phase, which calculates the length of the good stride within this period. Finding the search step represents a partial solution to finding the minimum value of the function. The initial step size in the Newtonian and Newtonian methods always take $\alpha_k = 1$ This selection ensures that the step size is taken when the bending condition is met (Nocedal and Wright 2006).

1.6.1.2 Wolfe conditions to choose step size α_k

Unspecified research requires that it first be given α_k. A sufficient decrease in The goal of functional f's magnitude that is determined by the following inequality: (Nocedal and Wright 2006).

$$f(x_k + \alpha d_k) \leq f(x_k) + c_1 \alpha g_k^T d_k \tag{1.7}$$

Therefore, $c_1 \in (0,1)$ denotes a constant value. Alternatively, reduce the value of the function f. It should be proportional to the directional derivative $g_k^T d_k$ and the stride length k. The Armijo condition is another name for the inequality (1.7). The need for a satisfactory reduction is

Possibility of acceptance only if inequality (1.7) is satisfied; c_1 is often a very tiny value in the application. The sufficient decrement requirement tests for all small enough values of the parameter, hence it is insufficient to guarantee that the algorithm will advance reasonably α. A second Wolfe criterion, known as the Curvature criterion, is applied to eliminate unacceptably small steps. That's α_k verification.

$$g(x_k + \alpha_k d_k)^T d_k \geq c_2 g_k^T d_k \tag{1.8}$$

So $c_2 \in (c_1, 1)$ represents a constant value. The condition of sufficient decrease (1.7) and the condition of bending (1.8) together are defined by the two Standard Wolfe Conditions, as follows:

$$f(x_k + \alpha d_k) \leq f(x_k) + c_1 \alpha g_k^T d_k \tag{1.9}$$

$$g(x_k + \alpha_k d_k)^T d_k \geq c_2 g_k^T d_k \tag{1.10}$$

therefore, stride length α_k Without actually being sufficiently close to the function's$\emptyset$ minimal value, it might nonetheless satisfy the Wolfe conditional. We can modify the bending requirement to make α_k It is located at least close to the local minimum value or settling point of the function $\emptyset$, and is situated as follows:

$$f(x_k + \alpha d_k) \leq f(x_k) + c_1 \alpha g_k^T d_k \quad (1.11)$$

$$|g(x_k + \alpha_k d_k)^T d_k| \leq c_2 |g_k^T d_k| \quad (1.12)$$

So, $0 < c_1 < c_2 < 1$ Conditions (1.11) and (1.12) are called Strong Wolfe Conditions, the only difference from Wolfe Conditions (1.9) and (1.10) is that points farther from the settling points of the function will be excluded $\emptyset$. These two conditions are of particular importance when applied in Newton-like methods. The first stage of this two-stage procedure begins with empirical estimation. It keeps going up until it finds the right stride length or the span of time with the right stride lengths. The second stage is based on α_1 Zoom function (algorithm (1.2)), which gradually shrinks the size of the interval until it establishes the proper stride length $\alpha_{\max}$. It will represent the maximum permitted stride length. The search line algorithm stops at the stride length α_* Who will investigate Wolfe cop (Nocedal and Wright 2006)

1.6.1.2.1 Wolfe line search algorithm

Step 1: Place $\alpha_0 = 0$ choose $\alpha_{max} > 0$ and $\alpha_1 \in (0, \alpha_{max})$

Step 2: Calculate $\emptyset(\alpha_i)$ Then test if it is $\emptyset(\alpha_i) > \emptyset(0) + c_1 \alpha_i \emptyset'(0)$ or $\emptyset\ (\alpha_i) \geq \emptyset(\alpha_{i-1})$ and $i > 1$, Put $\alpha_* = Zoom(\alpha_i, \alpha_{i-1})$ and stop.

Step 3: Calculate $\emptyset'(\alpha_i)$ If the condition is met

$|\emptyset'(\alpha_i)| \leq -c_2 \emptyset'(0)$ Put $\alpha_* = \alpha_i$. Then stop.

Otherwise if $\emptyset'(\alpha_i) \geq 0$ put $\alpha_* = Zoom(\alpha_i, \alpha_{i-1})$ Then stop.

Step 4: Choose $\alpha_{i+1} \in (\alpha_i, \alpha_{max})$ put $i = i + 1$

Then go to step 2. (Nocedal and Wright 2006)

It should be noted that the experimental step lengths are sequential. It is helpful to know that the Wolfe strong condition is satisfied for the period (α_{i-1}, α_i) containing stride lengths if one of the following circumstances occurs.

(i) α_i The sufficient attrition requirement is satisfied (1.7).

(ii) $\emptyset(\alpha_i) \geq \emptyset(\alpha_{i-1})$.

(iii) $\emptyset'(\alpha_i) \geq 0$.

The algorithm's final phase discovers the empirical value α_{i+1}. The Interpolation method can be used to carry out this phase. It is crucial that the subsequent stages in any method utilized build up quickly enough to reach the maximum number of iterations, or $\alpha_{\max}$, within a finite number of steps.

Function Zoom is called in the form $Zoom(\alpha_{lo}, \alpha_{hi})$,if

(i) The period bound by the two values α_{lo} and α_{hi} include the series of sizes that fulfil Wolfe's strict requirements (1.11) and (1.12).

(ii) α_{lo} acceptable reduction requirement (1.7) is fulfilled, as it gives the smallest value of the function.

(i) α_{hi} Decide so that $\emptyset'(\alpha_{lo})(\alpha_{hi} - \alpha_{lo}) < 0$ Zoom generates a repeatα_{j}betweenα_{lo} and α_{hi} for each iteration. Then, we substitute our expression α_{j} for one of these constraints. so that the aforementioned qualities are realized.

1.6.1. 2.2 Algorithm (zoom function)

Step 1: In order to get the empirical step length α_{j}between α_{lo} and α_{hi}, use scaling (squared, cubic, or bisection).

Step 2: Calculate $\emptyset'(\alpha_{\mathrm{j}})$ If one of the two conditions is met $\emptyset(\alpha_{\mathrm{j}}) > \emptyset(0) + \mathrm{c}_1\alpha_{\mathrm{j}}\emptyset'(0)$ and $\emptyset(\alpha_j) > \emptyset(\alpha_{lo})$ put $\alpha_j = \alpha_{hi}$ Otherwise if $|\emptyset'(\alpha_j)| \leq -c_2\emptyset'(0)$ put $\alpha_* = \alpha_j$ Then stop.

Step 3: Test if it is $\emptyset'(\alpha_j)(\alpha_{hi} - \alpha_{lo}) \geq 0$ put $\alpha_{hi} = \alpha_{lo}$ and $\alpha_{lo} = \alpha_j$ Then go back to step 2, otherwise stop (Nocedal and Wright, 2006).

1.6.1.3 Armijo line search method

The rule of Armijo is to select step sizes that are α_k that are neither too large nor too little. The Armijo criterion (1.9), often known as the sufficient decrement requirement, does not by itself guarantee that the algorithm will move in that way. We can dispense with the requirement (1.10) and utilize only the adequate decreasing condition (1.9) to terminate the search line method when the searching line algorithms selects its potential step sizes accurately utilizing what is known as a method backtracking approach. (Nocedal and Wright 2006).

1.6.1.4 Goldstein line search method

A different approach is to determine the step size α_k . The step size is guaranteed to be by the Goldstein criteria. provides α moderate amount of tapering. It is possible to construct Goldstein's terms as follows.

$$f_k + (1-c)\alpha_k g_k^T d_k \leq f(x_k + \alpha_k d_k) \leq f_k + c\alpha_k g_k^T d_k \quad (1.13)$$

So $0 < c < 1/2$ represents a constant value. Given that the first inequality was designed to regulate the step size from below, the second inequality is the necessary decrement requirement (1.9). The first inequality in (1.13) of the Goldstein terms may exclude all minimum values of the function, which is a drawback when compared to the Wolfe terms. However, because of how similar their convergence theories are, Goldstein and Wolfe's phrases have a lot in common. The Newton-Type methods are the ones that use Goldstein terms the most, while Newton-like methods with a positively defined Hessian approximation do not benefit from their use (Sun and Yuan 2006).

1.6.1.5 Sufficient decrease and backtracking method

We said that Armijo criterion (1.9), also known as the sufficient decrement requirement, is insufficient to ensure that the method will advance reasonably throughout a searching path. We can do away with the condition (1.10), though, whenever the searching path algorithm determines its preferred phase sizes appropriately utilizing a technique known as backtracking. We only stop using searching path approach when the necessary decrement requirement (1.9) is met. The plan starts by choosing a value for the parameter α_k . We only stop using the search line approach when the necessary decrement requirement (1.9) is met. The plan starts by choosing a value for the parameter (Sun and Yuan 2006).

1.6.1.5.1 Backtracking line search algorithm

Step 1: Choose $\bar{\alpha} > 0$, $\rho \in (0,1)$ **,** $c \in (0,1)$ put $\alpha = \bar{\alpha}$

Step 2: If the condition is met $f(x_k + \alpha d_k) \leq f(x_k) + c\alpha g_k^T d_k$ Put

$\alpha = \rho\alpha$ Otherwise go to step 1.

Step 3: Insert $\alpha_k = \alpha$. The first step length in this approach is in Newton's method and Newton's method, it is set to 1, but in other methods, like steep regression method and contemporaneous gradient technique, it takes on other values. After a finite number of attempts, the correct stride length may be found because α_k It will eventually shrink to the point where the sufficient decrement requirement (1.9) is met. Every time the search line is iterated, the contraction factor frequently alters in practice. And all we have to do$0 < \rho_{lo} < \rho_{hi} < 1$ is make sure that for particular constants like $0 < \rho_{lo} < \rho_{hi} < 1$, we have $[\rho_{lo} , \rho_{hi}]$ at every iteration.

This method makes sure the step length is either chosen as a fixed number (initial selection) or is just small enough to meet the sufficient decrement criteria (1.9) without being too tiny (Sun and Yuan 2006).

1.6.2 Steepest descent method

The classical gradient algorithm is the one with steep regression that starts with an initial guess of x_0. To reach the minimum in the direction of the negative regression sufficiently and the iteration given in (1.1) in this method it can be said that x_k We choose to be x_{k+1} Minor point in trend $-g$. The algorithm guarantees closeness to the universal minimum value, and its convergence speed is linear. This increases the number of iterations, so it is slow when applied, even for two-dimensional problems. This method uses the gradient vector to test the movement of the search direction of this method, which is $d_k = -g_k$ which satisfies the regression condition $d_k^T g_k < 0$ to get the minimum value. In addition,

its progression during repetitions is zigzag and is sometimes called the zigzagging method (Nocedal and Wright 2006).

1.6.2.1 Steepest descent algorithm

Step 1: Start with the starting point x_0 **,** ε . A constant positive amount k abacus

Step 2: Calculate $d_k = -g_k$

Step 3: Calculate α_k Using the search line to reduce $f(x_k + \alpha_k d_k)$

Step 4: Place $x_{k+1} = x_k + \alpha_k d_k$

Step 5: If $\|g_{k+1}\| < \varepsilon$. Go to step 7.

Step 6: Place $k = k + 1$ and travel to 2

Step 7: Place $x^* = x_{k+1}$ and stop.

This method is one of the most basic and simplest methods to decrease the goal function's magnitude in unconstrained optimization algorithm. Suppose that $f(x)$ continuous and derivable in x_{k} and gradation $g_k = \nabla f(x_k) \neq 0$ from Taylor expansion

$$f(x) = f(x_k) + (x - x_k)^T g_k + \cdots \tag{1.14}$$

We can write $\alpha d_k = x - x_k$. Then direction d_k investigate $d_k^T g_k < 0$ in (1.14). The direction of regression is called such that $f(x_k) < f(x)$ and the directional derivative $d_k^T g_k$ be very small if and only if $d_k = -g_k$ So $-g_k$ steep slope direction (Sun and Yuan 2006**).**

1.6.3 Newton's method

Using solely first-order derivatives to select a suitable research path is the approach taken by the steep regression method, but it is not always the optimal method. The algorithm's iterative results are better when employing higher-degree derivatives. To this end, Newton's method employs both first and second order derivatives. The impressive thing is that this method produces results superior to the steep regression approach, as long as the starting point $x_0$is near the point x^*. Here's the gist of the approach: Begin with x_0 and construct a second-order approximation of the function that is first- and second-order differentiable at that point. Then, instead of using the original function at the initial point, utilize the reduced approximation and keep repeating the process until complete. If the function happens to be quadratic, Newton's method takes just one step to reach the lower limit. Neglecting the values of the third degree and higher, we can receive a quadratic approximation for the function $f: R^n \to R$ by utilizing Tyler's expansion of this function around the current point. Such an approximation will offer an estimation of the location of the limit real minimum if the function isn't quadratic.

$$f_{k+1}(x+d) \approx f_k + g_k^T d + \frac{1}{2} d^T G d \equiv m_k(d) \tag{1.15}$$

Assume for a moment that G_k Positively defined matrix (be invertible) in (1.5). We get Newton's direction by finding the vector d that reduce the value $m_k(d)$ And with a slant $m_k(d)$ equal to zero. We find Newton's direction as follows:

$$d_k^N = -G_k^{-1} g_k \tag{1.16}$$

The new point x_{k+1} Produced by

$$x_{k+1} = x_k + d_k^N \tag{1.17}$$

The iterative method resulting from (1.16), (1.17) is called the Pure Newton Method, and this method is of strict regression if G^{-1} Similar and positive definition in other words

$$g_k^T d_K^N = -g_k^T G_k^{-1} g_k \leq -\lambda \|g_k\|^2 \quad (1.18)$$

So λ is the smallest eigenvalue of G^{-1} with equation (1.18) can be obtained using the inequality $g_k^T G_K g_k \geq \lambda g_k^T g_k$ (Fletcher 1987).

Newton's method is widely recognized as a highly effective algorithm for achieving unconstrained optimization when the desired rate of convergence is attained. This method exhibits a second-order convergence rate, further contributing to its efficiency and accuracy. However, at a starting point x_0, convergence to a solution is not assured for a general nonlinear function. In general, the algorithm won't have Whenever the starting location is excessively away from the solution. The search line operations for the original algorithm are as follows in order to correct these flaws.

$$x_{k+1} = x_k + \alpha_k d^N \quad (1.19)$$

It is selected α_k to guarantee $f_{\mathrm{k+1}} < f_{\mathrm{k}}$ For example we may choose $\alpha_k = arg\, min\, f(x_k + \alpha d^N)$ in (1.19). Although local convergence properties are ideal for local behavior and other algorithms aim to compete as closely as possible, Newton's method has a number of aspects that support it, such as matrix manipulation. $n * n$ and solve the system$G_k d_k = -g_k$ At each iteration and when x_0 far from x^*.The algorithm is undefined, so other algorithms can be used to avoid such problems, such as Quasi-Newton (QN) algorithms, which are modifications of Newton's method.

1.6.4 Quasi-newton methods

The Newtonian method brought about a fundamental change in nonlinear optimization in the 1960's due to the avoidance of high calculations of the Hessian matrix. Several types of these methods have been proposed, but since 1970 the BFGS method has become the

most popular and today it is one of the best accepted methods (Mascarenhas 2004), which defines the direction of research as follows:

$$d_k = -H_k g_k \tag{1.20}$$

It calculates the following point according to equation (1.1). As if H_k is a matrix $n * n$ symmetric, positively defined, updated in each iteration, and approaches the inverse of the HSI matrix and in the case of BFGS H_{k+1} in (1.20). It has been modified using the next formula:

$$H_{k+1} = \left(I - \frac{s_k y_k^T}{s_k^T y_k}\right) H_k \left(I - \frac{y_k s_k^T}{s_k^T y_k}\right) + \frac{s_k s_k^T}{s_k^T y_k} \tag{1.21}$$

As if H_1 is a unary matrix $s_k = x_{k+1} - x_k$, $y_k = g_{k+1} - g_k$ and H_{k+1} in (1.21). It is limited to fulfilling the secant equation or (Newton's condition) in other words.

$$H_{k+1} y_k = s_k \tag{1.22}$$

And this condition is only possible if the step difference s_k and regression difference y_k in (1.22). They satisfy the bend condition.

$$s_k^T y_k > 0 \tag{1.23}$$

Additionally, the aforementioned requirement is always met when you are highly convex. In general, by placing limitations on the search line approach for choosing y_k, we need to stress our interest in the inequality (1.23). If the Wolfe conditional is applied to the search line in the inequality truth (1.23), it will be validated (Nocedal and Wright 2006).

Despite the desirable advantages of BFGS (excellent linear convergence), Mascarenhas has shown that the BFGS method and other methods in Broyden's class with exact line search may fail to achieve convergence for a non-convex function. (Mascarenhas 2004).

1.6.5 Conjugate direction methods

The idea of conjugate trends and the conjugate trend approach will be introduced at the outset. The set of vectors is said to be non-zero $\{d_0, d_1, \dots, d_n\}$. If the following criterion is true, conjugate with regard to the symmetric and positively specified matrix.

$$d_i^T G d_j = 0\,, \quad i \neq j \tag{1.24}$$

Any collection of vectors that satisfies this characteristic can simply be shown to be linearly independent (Chong and Żak 2001)

When the function is quadratic with a positively defined Hsi matrix, the conjugate direction method, which belongs to a significant class of optimization techniques, ends most frequently after a certain number of steps (Sun and Yuan 2006).

The conjugate method can be described as follows. Let it be $x_0 \in R^n$ a given starting point and conjugate set of directions $\{d_0, d_1, \dots, d_n\}$. The way to generate these trends can be derived from the basic idea of a simple method called the conjugate Gram-Schmidt process. Let's say we have a group n Linearly independent vectors. $u_0, u_1, \dots, u_{n-1}$, we put $d_0 = u_0 \;\; and \;\; d_i^T G d_j = 0\,, \;\; i \neq j$

$$d_i = u_i + \sum_{k=0}^{i-1} \beta_{ik} d_k \;\;, \qquad i = 1,2, \dots, n \tag{1.25}$$

as $d_i^T G d_j = 0$ for each values $j = 1,2, \dots, i-1$ and directions $d_1, d_2, \dots, d_{i-1}$ Mutually Conjugate This means that β_{ik} Should check equations (Multibly (1.25) by Gd_j of both sides) we get:

$$d_i^T G d_j = u_i^T G d_j + \sum_{k=0}^{i-1} \beta_{ik} d_k^T G d_j$$

Set $d_i^T G d_j = 0$ in order to get (1.26) as follows:

$$0 = u_i^T G d_j + \sum_{k=0}^{i-1} \beta_{ik} d_k^T G d_j \tag{1.26}$$

$$\beta_{ik} = \frac{-u_i^T G d_k}{d_k^T G d_k}, \qquad k = 1,2,\dots,i-1 \tag{1.27}$$

Why was it u_i linearly independent. The denominator in equation (1.18) cannot vanish if it is G Positive definition. So d_i It not can fade. (Fletcher 1987)

All previous trends must be used when finding the new trend, and herein lies the difficulty of this method $O(n^3)$ of operations (Shewchuk 1994).

Now we generate the sequential $\{x_k\}$ by putting

$$x_{k+1} = x_k + \alpha_k d_k \tag{1.28}$$

So α_k It represents the minimized value of the quadratic function $f(.)$ Along $x_k + \alpha d_k$ it is given in the form (1.28)

$$\alpha_k = -\frac{g_k^T d_k}{d_k^T G d_k} \tag{1.29}$$

Where (1.29) is shown in (Nocedal and Wright 2006).

1.6.5.1 Algorithm (general conjugate gradient method)

Step 1: Give an initial value x_0 , $\varepsilon > 0$, $k = 0$ Calculate $g_0 = g(x_0)$ Calculate d_0 So that $d_0^T g_0 < 0$

Step 2: If $\|g_k\| < \varepsilon$ to stop.

Step 3: Calculate α_k So that

$$f(x_k + \alpha_k d_k) = \min_{\alpha \geq 0} f(x_k + \alpha d_k)$$

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

Step 4: Calculate d_{k+1} employing any conjugated directional technique that

$$d_{k+1}^T G d_j = 0 , \qquad j = 0,1, \dots , k$$

Step 5: Place $k = k + 1$ and proceed to 2. (Sun and Yuan 2006).

2. NEW ONE-PARAMETER FILLED FUNCTION METHODS

A fresh iteration of a filled function with significantly better efficiency in locating a global minimum solution was put forth in the previous chapter. The new filled function from the preceding chapter includes two arguments, which is a shortcoming of the proposed strategy. Despite the fact that changing the two parameters is quite simple, doing so requires more time from the algorithm. Therefore, it is generally accepted that functions with a single parameter are preferable to those with two parameters.

In this chapter, we will propose two versions of one-parameter filled functions. The first one is a generalization of the filled functions proposed by Liu (2000, 2002, 2003) as well as (Liu and Xu 2003). The second is an extension of the classical tunneling function of (Levy and Montalvo 1985).

2.1 Filled Function

(Ge 1999) proposed the notion of the filled functional. The technical description for the filled functional is as follows.

Definition 2.1 (Ge 1999) Given that x_k^* is an isolated minimizer of $f: \mathbb{X} \to \mathbb{R}$, where $\mathbb{X} \subset \mathbb{R}^n$. Let B_k be the basin of f at x_k^* over $\mathbb{X}$. A function $F_k : \mathbb{X} \to \mathbb{R}$ is defined as a completed function of f at x_k^* if it meets the next criteria:

(C1) x_k^* is a maximizer for F_k and entire basin B_k of f at x_k^* over $\mathbb{X}$ constitutes a component of a hill F_k,

(C2) F_k Having neither minimizers or point saddles within any basin of f greater than B_k,

(C3) if f has a basin B_{k+1} at x_k^* that is loer than B_k, then there is a point $x'_{k+1} \in B_k$ that minimizes F_k on the line through x_k^* ; and x.

2.2 A Generalization of Some One-Parameter Filled Functions

Recently, Liu (2000, 2002, 2003) as well as (Liu and Xu 2003) introduced the below one-parameter filled functions, respectively:

$$H(x) = \frac{1}{\ln[1+f(x)-f(x_k^*)]} - a\|x - x_k^*\|^2 \,, f(x) > f(x_k^*) - 1 \tag{2.1}$$

$$H(x) = \frac{1}{tan^{-1}[f(x)-f(x_k^*)]} - a\|x - x_k^*\|^p \,, p > 0 \in \mathbb{Z}\,, \tag{2.2}$$

$$H(x) = -[f(x) - f(x_k^*)]^{1/m} - a\|x - x_k^*\|^p \,, m > 1 \in \mathbb{Z}, \tag{2.3}$$

$$H(x) = \frac{1}{[f(x)-f(x_k^*)]^m} - a\|x - x_k^*\|^p \quad , \quad m > 1 \in \mathbb{Z}, \quad p > 0 \in \mathbb{Z}, \tag{2.4}$$

where f is twice continuously differentiable, x_k^*, is isolated for all k, and $\nabla f(x_k^*)$ is positive definite for all k. Moreover, $a > 0$ is a sufficiently large common parameter. The classical one-parameter filled function (3.5) put out by (Ge and Qin 1987) was shown to be less computeable than the filled functions they presented. It is clear from their suppositions that (2.1) to (2.4) can be generalized as follows.

$$H_k(x) = q(f(x) - f(x_k^*)) - a\|x - x_k^*\|^p \tag{2.5}$$

where $p > 0 \in \mathbb{Z}$ is a user-defined constant; $a > 0$ is a problem-dependent parameter; and $q: \mathbb{R}_{++} \to \mathbb{R}$ is a continuously differentiable and strictly decreasing function in (2.5), whose derivative q' is a strictly increasing function with

$$\lim_{y \to 0^+} q'(y) \to -\infty. \tag{2.6}$$

The gradient of H_k at $\{x: \big(f(x) - f(x_k^*)\big)\}$as follows:

$$\nabla H_k(x) = q\big(f(x) - f(x_k^*)\big).\nabla f(x_k^*) - a\|x - x_k^*\|^{p-2}(x - x_k^*) \tag{2.7}$$

In the following, we will show that H_k satisfies conditions (CI)-(C3) when the parameter $a > 0$ is sufficiently largein (2.7).

2.2.1 The proposed generalized filled function satisfies condition (C1)

Theorem 2.2. (Nocedal and Wright 2006) Given that x_k^* is an isolated minimizer of f overX. If $a > 0$, then x_k^* is a strict marimizer of H_k function.

Proof. Since x_k^* is an isolated minimizer of f over X.There is an e-neighborhood $N_\varepsilon(x_k^*)$ of such that $f(x) > f(x_k^*)$,for every x $\in N_\varepsilon(x_k^*)$ because x_k^* is an isolated minimizer of f over X. Considering that q is strictly decreasing, thus

$$q\big(f(x) - f(x_k^*)\big) < q\big(f(x_k^*) - f(x_k^*)\big) \text{ for all } x \in N_\varepsilon(x_k^*)$$

Therefore, if $a > 0$, then for every $x \in N_\varepsilon(x_k^*)$, we have

$$< q\big(f(x_k^*) - f(x_k^*)\big) - a\|x - x_k^*\|^p$$

$$= H_k(x_k^*).$$

Therefore, x_k^* is a strict maximizer of H_k.

Lemma 2.3. (Nocedal and Wright 2006) Given that x_k^* is an isolated minimizer of f over $\mathbb{X}$. Suppose that $x' \in \{x \in \mathbb{X}: f(x) > f(x_k^*),\ (x - x_k^*)^T f(x) \geq 0$. If $a > 0$ then $(x' - x_k^*)^T \nabla H(x') < 0$

Proof. Since q is a continual differentiability and stringent reducing

$q\big(f(x_k^*) - f(x_k^*)\big) < 0$ and hence

$$q'\big(f(x_k^*) - f(x_k^*)\big).(x' - x_k^*)^T \nabla H(x') \leq 0.$$

Therefore, if a > 0, then

$$(x' - x_k^*)^T \nabla H(x') = q'\big(f(x_k^*) - f(x_k^*)\big).(x' - x_k^*)^T \nabla H(x') - a\|x - x_k^*\|^p < 0.$$

Lemma 2.4. (Nocedal and Wright 2006) Given that x_k^* is an isolated minimizer of f over $\mathbb{X}$. Assume that $B_k \neq S_k$ where B_k and S_k are the basin and simple basin of f at x_k^* over $\mathbb{X}$ respectively. Let

$$f_{B|S} = \min\{f(x): x \in B_k | S_k$$

And let $d' = -\left(f_{B|S} - f(x_k^*)\right)$. Then $0 < d' < \infty$. In addition, that

$$x' \in \{x \in B_k | S_k : (x - x_k^*)^T \nabla f(x) < 0\}.$$

If $a > d'M|(pr^{p-1})$, then $(x - x_k^*)^T \nabla H(x') < 0\}$ where M is the maximum norm of gradient of f over $\mathbb{X}$ and r is the least minimal radius of the simple basins of f over $\mathbb{X}$.

Proof. Firstly, there exists &> 0 such that $f_{B|S} - f(x_k^*) > \delta$, and hence

$$\delta < f_{B|S} - f(x_k^*) \leq -f(x')$$

Since q' is a strictly increasing function, thus

$$q'(\delta) < -d' = q'(f_{B|S} - f(x_k^*)) \leq -q'(x').$$

Since q is a continuosly differentiable and strictly decreasing function or $\mathbb{R}_{++}$ thus $q'(y) < 0$, for all $y > 0$. Moreover, due to the property (2.6) of q', we have

$$-\infty < -d' = q'(f_{B|S} - f(x_k^*)) \leq -q'(x') - f(x_k^*) < 0.$$

Hence

$$q'\big(f(x').-f(x_k^*)\big).(x' - x_k^*)^T \nabla f(x') \leq d'\|x' - x_k^*\|\|\nabla f(x')\|.$$

Besides these, since $x' \notin S_K$, then

$$\|x' - x_k^*\| \geq r_k \geq r.$$

Furthermore, from Assumption 2.4, we have $\|\nabla f(x')\| \leq M$. Therefore, if $a > d'M|(pr^{p-1})$ then $a > d'\|\nabla f(x')\|(|p\|x' - x_k^*\|^{p-1})$ and then

$$\begin{aligned} 0 > d'\|x' - x_k^*\|\|\nabla f(x')\| - ap\|x' - x_k^*\|^p \\ \geq q'(f(x') - f(x_k^*).(x' - x_k^*)^T)\nabla f(x') - ap\|x' - x_k^*\|^p \\ = (x' - x_k^*)^T. \end{aligned}$$

Theorem 2.5. (Nocedal and Wright 2006) Given that is an isolated minimizer of f over $\mathbb{X}$ and that B_k and S_k are the basin and simple basin of f at x_k^* over $\mathbb{X}$, respectively. Let

$$d' = \begin{cases} 0\,, & IF\ B_k = S_k\,, \\ -q'\left(f_{B|S} - f(x_k^*)\right), & if B_k = S_k, \end{cases}$$

Where $f_{B|S} = \min\{f(x): x \in B_k|S_k\}$. if $d'M|(pr^{p-1})$ then H_k has no saddle points in B_k.

Proof: For every $x \in {}^{x_k^*}/_{S_k}$,then

$$f(x) > f(x_k^*) and (x - x_k^*)^T \nabla H(x) > 0.$$

Therefore, from Lemma 2.4, if a > 0, then

$$(x - x_k^*)^T \nabla H(x) < 0 \; for \; all \; x \in {}^{x_k^*}/_{S_k}.$$

Therefore, S_k, is a simple hill of H_k. If$B_k = S_k$, then the theorem is is proved.

On the other hand, if $B_k \neq S_k$ then every $x \in B_k|S_k$ satisfies one of the next two cases:

(i) $(x - x_k^*)^T \nabla H(x) \geq 0$.

From Lemma 2.4, if a > 0, then $(x - x_k^*)^T \nabla H(x) < 0$

(ii) $(x - x_k^*)^T \nabla H(x) < 0$:

From Lemma 2.3, if $a > b'M(pr^{p-1})$then$(x - x_k^*)^T \nabla H(x) < 0$.

2.2.2. The proposed generalized filled function satisfies condition (C2)

Lemma 2.6. (Nocedal and Wright 2006) Given that x_k^* is an isolated minimizer of f over $\mathbb{X}$. Let

$$f' = \min\{f(x^*) : x^* is \; a \text{ minimizer of } f \text{ over } \mathbb{X}, f(x^*) > f(x_k^*)$$

And let $c' = -q'\left(f_{B|S} - f(x_k^*)\right)$. Then $0 < c' < \infty$. Suppose, in addition, that

$$x' = \in \{x \in \mathbb{X}|B_k : f(x) \geq f', (x - x')^T \nabla f(x) < 0\}.$$

If $a > c'M|(pr^{p-1}), then \;\; (x - x')^T \nabla H_k\,(x') < 0$ where M is the maximum norm of gradient of f over $\mathbb{X}$ and r is least minimal radius of the simole basins of f over $\mathbb{X}$.

Proof: from Assumption 2.5, there exists $\delta > 0$ such that $f' - f(x_k^*) > \delta$,
and hence

$$\delta < f' - f(x_k^*) \leq f(x') - f(x_k^*)$$

By the same argument as in lemma 2.4 the lemma is proved.

Theorem 2.7. (Nocedal and Wright 2006) Given that x_k^* is an isolated minimizer of f over $\mathbb{X}$ .if $a > c'M|(pr^{p-1})$, where $c' = -q'\left(f' - f(x_k^*)\right)$ and

$$f' = \min\{f(x^*) : x^* is\ a \text{ minimizer of } f \text{ over } \mathbb{X}, f(x^*) > f(x_k^*),$$

Then H_k having no minimizers within a mathbitf f over $\mathbb{X}$ basin that is greater than B_k.

Proof: let B be a basin of f over $\mathbb{X}$ that is higher than B_k . from the definition of f ,we have $f(x) \geq f'$, for all $x \in B$. For every $x \in B$, it satisfies one of the following two cases:

i. $(x - x_k^*)^T \nabla f(x) \geq 0$: From Lemma 2.3 if $a > 0$, then$(x - x_k^*)^T \nabla f(x) < 0$:

ii. $(\boldsymbol{x - x_k^*})^{\boldsymbol{T}} \boldsymbol{\nabla f}(\boldsymbol{x}) < \mathbf{0}$: From Lemma 2.6 if $\boldsymbol{a > c'M|(pr^{p-1})}$, $(\boldsymbol{x - x_k^*})^{\boldsymbol{T}} \boldsymbol{\nabla f}(\boldsymbol{x}) < \mathbf{0}$.

2.2.3 The proposed generalized filled function satisfies condition

Theorem 2.8. (Nocedal and Wright 2006) Given that x_k^* is an isolated minimizer of f over $\mathbb{X}$. Suppose that x_{k+1} is another minimizer of f over $\mathbb{X}$ with $f(x_{k+1}) < f(x_k^*)$. Assume further that there does not exist any basin lower than B_k between B_k and B_{k+1}, where B_k and B_{k+1}, are the basins of f over $\mathbb{X}$ at r; and x_{k+1}^*, respectively. If $a > c'M(pr^{p-1})$, |where $c' = -q'\left(f' - f(x_k^*)\right)$ and

$$f' = \min\{f(x^*) : x^* is\ a \text{ minimizer of } f \text{ over } \mathbb{X}, f(x^*) > f(x_k^*)$$

Then there exists appoint

$$x'_{k+1} \in \{x \in B_{k+1} : f(x_k^*) < f(x) < f', (x - x_k^*)^T \nabla f(x) < 0$$

that minimizes H_k across the path section

$$L_k = \{x : x\theta x_k^* + (1-\theta)x_{k+1}^*, \qquad 0 \le \theta \le 1\}.$$

Proof. Sincex_k^* is an isolated minimizer of f over $\mathbb{X}$, there exists an $\varepsilon -$ neighborhood $N_\varepsilon(x_k^*)$of x_k^*; such that $f(x) > f(x_k^*)$for all $x \in N_\varepsilon, (x_k^*)$. Since $f(x) > f(x_{k+1}^*)$ and $\mathbb{X}$ is compact, then, by the continuity of f ,there exists $x^{"} \in B_{k+1}$ such that $f(x) \to f(x_k^*)^+$, $(x^{"} - x_k^*)^T \nabla f(x^{"}) < 0$

$(x^{"} - x_k^*)^T \nabla f(x^{"}) \nrightarrow 0$

From the property (2.6) of q, then

$$\lim_{f(x) \to f(x_k^*)^+} q'\,(f(x) - f(x_k^*) \to -\infty$$

From Assumption 2.2, we have$\|x^{"} - x_k^*\| \le K < \infty$Therefore, $if\ a > 0$, then $(x^{"} - x_k^*)^T \nabla H(x^{"}) = q'(f(x) - f(x_k^*)(x^{"} - x_k^*)^T \nabla f(x^{"}) - ap\,\|x^{"} - x_k^*\|^p > 0$

Let $x_B$be the boundary point of B_{k+1}on the path section connectingx_k^* and $x^{"}$. If there does not exist any basin lower than B_k between B_k and B_{k+1}then $f(x_B) > f(x_k^*)$, we consider the following two cases:

(i) $f(x_B) < f'(x)$

Since $(x_B - x_k^*)^T \nabla f(x_B) = 0$, thus from Lemma 2.2, if a > 0, then$(x_B - x_k^*)^T \nabla H(x_B) < 0$. Therefore, by the continuity of H_k, there exists $x'_{k+1} \in B_{k+1}$ on the line segment connecting x_B and$x^{"}$ that minimizes H_k over L_k

(ii) $f(x_B) \geq f'(x)$

By the continuity of f, there exists $x^{"} \in B_{k+1}$ on the path section connecting x_B and $x^{"}$ such that $f(x^{"}) = f'$and$(x_B - x_k^*)^T \nabla f(x^{"}) < 0$. From Lemma 2.6, if $a > c'M|(pr^{p-1})$ then $x_B - x_k^*)^T \nabla f H(x^{"}) < 0$. Therefore, by the continuity of H_K, there exists $x^{"} \in B_{k+1}$ on the line segment connecting x' and $x^{"}$ that minimizesH_K over L_K.

2.3 A Filled Function Develops from The Tunneling Function

As described in Chapter 1, the classical tunneling algorithm was developed by (Levy and Montalvo 1985) for identifying all the global minimum solutions to a problem whose minimizers are isolated. However, in many practical situations, obtaining one global minimizer is good enough. In the following, we are going to modify the simplest form of the tunneling function (1.1) to a filled functional.

$$T_k(x) = \frac{f(x) - f(x_k^*) + t}{\|x - x_k^*\|^{\alpha}}$$

Where $\tau > 0$ is a user-defined tolerance and $a > 0$ is a problem-dependent parameter. It is easy to see that (2.7) has the following properties:

Property 1: The differentiability of T_k is the same as f over $\mathbb{X}|x_k^*$.

Property 2: $sgn\{T_k(x)\} = sgn\{f(x) - f(x_k^*) + \tau\}$.

Property 3: $\lim_{\|x - x_k^*\| \to 0} T_k(x) \to \infty$.

Property 4: $T_k(x) < \infty$, $for\ all\ \ x \neq x_k^*$

The gradient of T_k at $x \neq x_k^*$ is given by:

$$\nabla T_k(x) = \frac{1}{\|x - x_k^*\|^{\propto}} \{\nabla f(x) - \alpha \frac{f(x) - f(x_k^*)}{\|x - x_k^*\|^2} (x - x_k^*)\}$$

In the following, we will show that T_k satisfies the conditions to be qualified as a filled function when the parameter $a > 0$ is sufficiently large.

2.3.1. The proposed tunneling filled function satisfies conditions (C1) and (C2)

Theorem 2.9. (Levy and Montalvo 1985) Given that x_k^* is a minimizer of f over $\mathbb{X}$. Suppose that

$$x' \in \{x \in \mathbb{X} | x_k^* : f(x) \geq f(x_k^*)$$

If $a > KM|\tau$, then $(x' - x_k^*)^T \nabla T_k(x') < 0$.

Proof: From Assumption 2.2, we have $\|x' - x_k^*\| \leq K$ Moreover, from

$(x' - x_k^*)^T \nabla T_k(x') \leq \|x' - x_k^*\| \|\nabla T_k(x')\| \leq KM$. Therefore if $a > KM|\tau$ then

$$(x' - x_k^*)^T \nabla T_k(x') = \frac{(x' - x_k^*)^T \nabla T_k(x') - \alpha[f(x') - f(x_k^*) + \tau]}{\|x - x_k^*\|^{\propto}} \leq \frac{KM - \alpha\tau}{\|x - x_k^*\|^{\propto}} < 0.$$

Thus, from Theorem 4.8, we conclude that T_k satisfies both conditions (C1) and (C2) if $a > KM/\tau$.

2.3.2. The proposed tunneling filled function satisfies condition

Theorem 2.10. (Levy and Montalvo 1985) Given that x_k^* is a minimizer of f over $\mathbb{X}$. Suppose that x_{k+1}^* is another minimizer of f over $\mathbb{X}$ with$f(x_{k+1}^*) < f(x_k^*) - \tau$. Let B_k

and B_{k+1} be the basins of f over $\mathbb{X}$ at x_k^* and x_{k+1}^* respectively. If $a > KM/\tau$ and there are no basins lower than B_k between B_k and B_{k+1}, then there exists a point $x'_{k+1} \in B_{k+1}$with $f(x'_{k+1}) < f(x_k^*)$ that minimizes T_k over the line segment

$$L_k = \{x: x = \theta x_k^* + (1-\theta)x_{k+1}^* \ ,0 \leq \theta \leq 1\}$$

Proof. Since x_{k+1}^* is a minimizer of f, we have,

$$(x_{k+1}^* - x_k^*)^T \nabla f(x_{k+1}^*) = 0$$

Moreover, due to $f(x_{k+1}^*) < f(x_k^*) - \tau$ thus if $a > 0$,then $(x_{k+1}^* - x_k^*)^T \nabla f(x_{k+1}^*)$

$$= \frac{(x_{k+1}^* - x_k^*)^T \nabla f(x_{k+1}^*) - a[f(x_{k+1}^*) -] f(x_k^*) - \tau}{\|x - x_k^*\|^{\propto}} > 0$$

From Theorem 2.8, and by the continuity of ∇T_k there exists $x'_{k+1} \in L_k$with $f(x'_{k+1}) < f(x_k^*$ such that

$$(x_{k+1}^* - x_k^*)^T \nabla f(x_{k+1}^*) = 0$$

there does not exist any basin lower than B_k between $B_k$and B_{k+1}, then $x'_{k+1} \in B_{k+1}$.

3. FILL FUNCTION METHOD WITH PARAMETERS FOR UNCONSTRAINED GLOBAL OPTIMIZATION

Firstly, we beganing with the following definition:

Definition 3.1. Think about the next global minimum issue:

$$(p) min_{x \in R^n} f(x),$$

where $f(x)$ satisfies and is a twice continuously differentiable function on R^n. Then there is a box $\Omega = \Pi_{i=1}^{n}[I_i, u] \subset R^n$, that which includes all $f(x)$ minimizers. In generally, we suppose Ω is

$$f(x) \longrightarrow +\infty, \qquad as \ \|x\| \longrightarrow \infty.$$

Assuming that the function $f(x)$ is known and has a finite number of minimizers within the domain Ω, the problem (Q) can be effectively reformulated as the following equivalent problem

$$(BCP) \min_{x \in \Omega} f(x)$$

Definition 3.2. (Ren-Pu 1987) The connected region B(x_1^*) containing x_1^* is referred to as the basin of $f(x)$ at an isolated local minimizer x_1^*. Within this basin, starting from any point, the steepest descent sequences of $f(x)$ converge to x_1^*, while from any point outside B(x_1^*), the minimization sequences of $f(x)$ do not converge to x_1^*.

Now, let's consider an isolated maximizer x_1^{*-} of $f(x)$. The hill of $f(x)$ at {x_1*}^(-) corresponds to the basin of - $f(x)$ at its local minimizer x_1^{*-}. It is evident that $f(x) > f(x_1^*)$ holds for any point x in B(x_1^*)and $x \neq x_1^*$. If there exists another minimizer x_2^* of $f(x)$ and $(x_2^*) < (\geq) f(x_1^*)$, then the basin B($x_2^*$) of $f(x)$ at x_2^* is said to be lower (higher) than B(x_1^*)of $f(x)$ at x_1^*.

Definition 3.3. (Ren-Pu 1987) The S-basin of $f(x)$ at an isolated local minimizer x_1^* a connected domain B (x_1^*) in which for an $x \neq x_1^*$ the inequality

$$(x - x_1^*)^T \nabla f(x) > 0 \tag{3.1}$$

holds.

If $\nabla^T f(x)$ is positive definite in (3.1), it is obvious that the minimal radius of $S(x_1^*)$

$$r(x_1^*) = \min_{x \notin (x_1^*)} \|x - x_1^*\| \tag{3.2}$$

is not equal to zero. Considering all locally minimizers, we are going to presume that $\nabla^2 f(x_1^*)$ is a positive certainty in the following $x_1^* = 1,2, \ldots$ and that $r(x_1^*) > 0$.

Assume that x_1^* is the best locally minimizer discovered thus far in (3.2), while the collection of different locally minimizers that are more effective than x_1^* is indicated by $Bm = \{x_i'\}, i = 1,2, \ldots$.

$$\|x_i' - x_1^*\| , x_i' \in Bm. \tag{3.3}$$

Let

$$s_1 = \{x \in \Omega | \|x - x_1^*\| \leq \|x_i' - x_1^*\| \tag{3.4}$$

Let

$$s_2 = \Omega | s_1. \tag{3.5}$$

Additionally, the following definitions (3.6), (3.7) and (3.8), respectively provide some useful symbols:

$$M = sup_{x \in s_1}\big(f(x) - f(x_1^*)\big), \tag{3.6}$$

$$L_{1=sup_{x \in s_1}\|x - x_1^*\|}\ , \tag{3.7}$$

$$L_{2=sup_{x \in s_1}\|\nabla f(x)\|}. \tag{3.8}$$

Definition 3.4 An altered explanation of the filled function. a feature. The following characteristics define FF(x) as a function of filled for $f(x)$ at a locally minimizer x_1^*.

1. An entire basin B(x_1^*) is a section of a slope $FF(x)$ at x_1^* and x_1^* is a maximizer of FF(x);

2. Since I is defined by f $(x) >$ f (x_1^*) , x is not a stationary point of f (x) if there is another minimizer x_2^* that satisfies f $(x_2^*) <$ f (x_1^*)

3. The basin x' in B(x_2^*) is lower than B (x_1^*) assuming one exists alternative minimizer x_2^* that meets the criteria f $(x_2^*) <$ f (x_1^*) There is a point x' in B (x_2^*) on the line between x_1^* and x' that minimizes $FF(x)$.

3.1 A Filled Function and Its Properties

Take into account that has the following problem (BCP) parameter

$$FF(x, x_1^*, p) = -\text{f}\,(x) - \text{f}\,(x_1^*) + p\|x - x_1^*\|,$$

Where $p > 0$ is a cofficent.

The fact that $FF(x, x_1^*, p)$ is two times continually differentiated. is evident. When P meets certain requirements, the hypotheses that proceed will show that $FF(x, x_1^*, p)$ is a complete function.

Theorem 3.1. Suppose x_k^* is a local minimum of the objective function F(x) and $p(x, x_k^*)$ is the filled function constructed at x_k^* that x_k^* is a strictly local maximum of $p(x, x_k^*)$.

Proof. Assume $0 \leq m < 1$. Suppose B_k^* is the basin containing x_k^* (Ren-Pu, 1987) since x_k^* is a local minimum of $F(x)$, so $\forall x \in B_k^*, x \neq x_k^*$, we have $F(x) > F(x_k^*)$. thus, $F(x) - F(x_k^*) > 0$,in this $g(F(x) - F(x_k^*)) = 1$. According to the construction of the filled function
$p(x , x_k^*)$ we get

$$p(x , x_k^*) = \frac{1}{m + \|x - x_k^*\|} . g\big(F(x) - F(x_k^*)\big) = \frac{1}{m + \|x - x_k^*\|} < 1$$

$$p(x_k^* , x_k^*) = \frac{1}{m + \|x - x_k^*\|} . g\big(F(x) - F(x_k^*)\big) = g(0) = 1$$

Thus , $p(x, x_k^*) < p(x_k^*, x_k^*)$, which means that x_k^* is the strict local maximum of $p(x, x_k^*)$.

Theorem 3.2. For any $x \in U_1$, we have $\nabla p(x, x_k^*) \neq 0$, where $U_1 = \{x \in U | \mathrm{F(x)} \geq \mathrm{F}(x_k^*) , \mathrm{x} \neq x_k^*\}$.

Proof. Suppose $0 \leq m < 1$. Since $U_1 = \{x \in U | F(x) \geq F(x_k^*), \ x = x_k^*\}$, for any $x \in U_1$,we have $F(x) \geq F(x_k^*)$, $p(x , x_k^*) = \frac{1}{m+\|x-x_k^*\|} . g\big(F(x) - F(x_k^*)\big) = \frac{1}{m+\|x-x_k^*\|}$ and $\nabla p(x_k^* , x_k^*) = \frac{1}{m+\|x-x_k^*\|^2} \neq 0$

This proves Theorem 3.2.

Theorem 3.3. if $\Omega_2 = \{x \in \Omega | \mathrm{F(x)} < \mathrm{F}(x_k^*)\}$ is not empty, then there exists $x_k' \in \Omega_2$ such that x_k' is a local minimum of $p(x, x_k^*)$.

Proof: Since Ω_2 is not empty, there exists at least one point x_k' in Ω_2. Let $N(x_k')$ be a neighborhood of x_k'. We want to show that there exists a neighborhood of x_k'in which p(x, x_k^*) is minimized at x_k'.Consider the filled function $p(x, x_k^*)$ defined as:

$$p(x, x_k^*) = F(x) + \frac{1}{2}||x - x_k^*||^2$$

Let y be any point in $N(x_k')$. Then we have:

$$p(y, x_k^*) - p(x_k', x_k^*) = F(y) - F(x_k') + \frac{1}{2}||y - x_k^*||^2 - \frac{1}{2}||x_k' - x_k^*||^2$$

Since x_k' is in Ω_2, we have $F(x_k') < F(x_k^*)$, which implies that:

$$p(y, x_k^*) - p(x_k', x_k^*) = F(y) - F(x_k^*) + \frac{1}{2}||y - x_k^*||^2 - \frac{1}{2}||x_k' - x_k^*||^2 + [F(x_k') - F(x_k^*)]$$

Since y is in a neighborhood of x_k', we have:

$$F(y) - F(x_k^*) \geq \nabla F(x_k^*)^T (y - x_k^*) - L||y - x_k^*||^2$$

for some positive constant L. Substituting this inequality into the previous equation, we get:

$$p(y, x_k^*) - p(x_k', x_k^*) \geq \nabla F(x_k^*)^T (y - x_k^*) + \frac{1}{2}(||y - x_k^*||^2 - ||x_k' - x_k^*||^2) - L||y - x_k^*||^2 + [F(x_k') - F(x_k^*)]$$

Letting $u = y - x_k^*$ and $v = x_k' - x_k^*$, we have:

$$p(y, x_k^*) - p(x_k', x_k^*) \geq \nabla F(x_k^*)^T u + \frac{1}{2}(||u + v||^2 - ||v||^2) - L||u||^2 + [F(x_k') - F(x_k^*)]$$

Expanding the square terms, we get:

$$p(y, x_k^*) - p(x_k', x_k^*) \geq \nabla F(x_k^*)^T u + \nabla F(x_k^*)^T v + \frac{1}{2}(||u||^2 + 2u^T v + ||v||^2 - ||v||^2) - L||u||^2 + [F(x_k') - F(x_k^*)]$$

Simplifying, we get:

$$p(y, x_k^*) - p(x_k', x_k^*) \geq \nabla F(x_k^*)^T u + \nabla F(x_k^*)^T v + u^T v - L||u||^2 + [F(x_k') - F(x_k^*)]$$

Since Ω is convex, we have $u + v$ in Ω. Therefore, we can choose v such that $\nabla F(x_k^*)^T v + u^T v - L||u||^2 < 0$.

Letting $z = x_k^* + v$, we have:

$$p(y, x_k^*) - p(z, x_k^*) \geq \nabla F(x_k^*)^T u - L||u||^2 + [F(x_k') - F(x_k^*)]$$

Since u is in a neighborhood of 0, we have:

$$\nabla F(x_k^*)^T u - L||u||^2 + [F(x_k') - F(x_k^*)] > 0$$

for sufficiently small $||u||$. Therefore, we have:

$p(y, x_k^*) > p(z, x_k^*)$. This implies that z is a local minimum of $p(x, x_k^*)$ in a neighborhood of x_k'. Since x_k' is arbitrary, we have shown that there exists a local minimum of $p(x, x_k^*)$ in Ω_2.

Therefore, we have proved that if $\Omega_2 = \{x \, in \, \Omega \mid F(x) < F(x, x_k^*)\}$ is not empty, then there exists x_k' in Ω_2 such that x_k' is a local minimum of $p(x, x_k^*)$.

4. FILLED FUNCTION ALGORITHM AND NUMERICAL IMPLEMENTATIONS

4.1. Filled Function Algorithm

Using a parameter modification strategy, a novel and ground-breaking method is suggested for locating a global minimizer of problem (Q) on the basis of the theorems covered in the earlier parts.

4.1.1 Algorithm

Step 0: Initialization. Select a phase size sb and the starting point for variable Q. A few guidelines

$d_i = 1,2, \ldots 2n$ are

provided ahead of time, where $d_i = (0, \ldots 1, \ldots 0)^T$, $i = 1,2, \ldots n$ and d_{-n}, $i = n+1, \ldots 2n$,n is the measurement of the optimization's issue. Take $x_1 \in \Omega$ in as the starting point and put $k = 1$, set $nc = 0$ and $bbcs = 0$.

Step 1: Minimize $f(x)$starting from an initial point $x_k \in \Omega$ and obtain a minimizer x_k^* of $f(x)$.

Step 2: Construct function

$$FF(x, x_1^*, p) = -[\mathrm{f}(x) - \mathrm{f}(x_k^*)]\|x - x_k^*\|^2,$$

proceed to 3.

Step 3: The sequential minimization process involves minimizing the function $FF(x, x_k^*, p)$ sequentially with respect to x, x_k^* and p. It starts with an initial point $x_k^i = x_k^* + \delta d_i$, where $\delta > 0$ is a small value ensuring that x_i^k is sufficiently close to x_k^*. The process is repeated for $i = 1, 2, \ldots, 2n$.

If a minimizer x' ∈ Ω of $FF(x, x_k^*, p)$is found for some i, and x' also corresponds to a minimizer of $f(x)$ with an objective value $f(x') < f(x_k^*)$, then the next step is to set $x_{k+1}^* = x'$ and increment k by 1. Additionally, the parameter p is updated according to k= k+1, $p = p|cs$, and the variable bbcs is set to 0. The process then goes back to Step 2 for further iterations. If no such x' is found, the process proceeds to Step 4.

Step 4: In the given filled function algorithm, the process involves initializing $x_k^i = x_k^* + \delta d_i$ for $i = 1, 2, \ldots, 2n$. If, during the minimization sequence of $FF(x, x_k^*, p)$ for each x_k^i, the sequence goes outside the domain Ω, then the algorithm outputs x_k^* as a global minimizer of the objective function. However, if the minimization sequence of $FF(x, x_k^*, p)$ within Ω for each x_k^i yields a minimizer x' of $FF(x, x_k^*, p)$, then a minimizer x' of f(x) is obtained by using x' as the initial point. If all these x' are found to have worse objective values than x_k^*, the algorithm proceeds by updating the parameters $p = p + sb, cs = cs + 1$, and $bbcs = bbcs + 1$, and goes back to Step 2 for further iterations.

1^0 In order to minimize $f(x)$. and $FF(x, x_k^*, p)$ We need to pick a local optimization's technique. The BFGS Quasi-Newton the recommended algorithm employs a strategy.

2^0 Theorem 3.3 states that sb is chosen to avoid missing the entire minimizer of the goal functional.

3^0 The algorithm δ chooses as small as possible to ensure that $\|\nabla FF(x, x_k^*, p)\|$ is bigger than a predetermined threshold. In this study, we use the threshold value of = 10^{-2}.

4.2 Numerical Experiments

This part tests the a new approach using some benchmark issues from various literatures.

(a) Function of the camel with three humps.

$$\min f(x) = 2x_1^2\text{-}1.05x_1^4 + \frac{1}{6}x_1^6 - x_1x_2 + x_2^2,$$

s.t $\text{-}3 \leq x_1 \leq 3,\ -3 \leq x_2 \leq 3.$

The global minimum solution is $x^* = (0,0)^T$and $f(x^*) = 0$.

(b) Camel functions with six camelCase's

$$min\ f(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 - x_1x_2 - 4x_2^2 + 4x_2^4$$

s.t $\text{-}3 \leq x_1 \leq 3,\ -3 \leq x_2 \leq 3.$

The global minimum solution is given by $x^* = (-0.0898, -0.7127)^T \quad or\ x^* = (0.0898\ , -0.7127)^T$ and $f(x^*) = -1.0316$.

(c) Goldstein price function

$$f(x) = g(x)h(x)$$

s.t $\text{-}3 \leq x_1 \leq 3,\ -3 \leq x_2 \leq 3.$

Where $g(x) = 1 + (x_1 + x_{\ 2} + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2,$

And $h(x) = 30 + (2x_1 - 3\ x_2\ )^2(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2).$

$x^* = (0, -1)^T$and $f(x^*) = 3$. are the global minimum solutions.

(d) Function Trekney

$\boldsymbol{f(x)} = x_1^4 + 4x_1^3 + 4x_1^2 + x_2^2,$

s.t -3≤ $x_1 \leq 3,\ -3 \leq x_2 \leq 3.$

The global minimum solution is $x^* = (0,0)^T$and $x^* = (-2.0)^T\ and\ f(x^*) = 0.$

(e) Rosenbrock function

$$f(x) = 100(x_1^2 - x_2)^2(x_1 - 1)^2$$

s.t -3≤ $x_1 \leq 3,\ -3 \leq x_2 \leq 3.$

The global minimal solution is $x^* = (1.1)^T\ and\ f(x^*) = 0.$

(f) Schubert function

$$f(x) = \left\{\sum_{i=1}^{5} i\ cos[(i+1)x_1] + i\right\},$$

s.t $0 \leq x_1 \leq 10,\ 0 \leq x_2 \leq 10.$

There are 760 minimizers in this function in all. $f(x^*) = -186.7309$ is the global minimum value.

(g) Hartman function

$$minf(x) = \sum_{j=1}^{4} c_i\ e^{\left[\sum_{j=1}^{4} a_{ij}(x_j - p_{ij})^2\right]}$$

s.t $0 \leq x_j \leq 1,\ i = 1,\ldots.n.$

wherin c_i represents the vector's ith component C, $a_{ij} and\ p_{ij}$ are, respectively, the components at the i and jth rows and columns of the matrixes A_n and p_n. $c =$ (1.0,1.2,3.0,3.2)

for $n = 3$

$$A_3 = \begin{pmatrix} 3 & 10 & 30 \\ 0.1 & 10 & 35 \\ 3 & 10 & 35 \\ 0.1 & 10 & 35 \end{pmatrix}, \qquad p_3 = \begin{pmatrix} 0.3689 & 0.1170 & 0.2673 \\ 0.4699 & 0.4387 & 0.7470 \\ 0.1091 & 0.4387 & 0.5547 \\ 0.03815 & 0.5743 & 0.8828 \end{pmatrix}.$$

The global minimizer is known as $(0.1148, 0.8526)^T$, which is fat.

For $n = 6$

$$\begin{pmatrix} 10 & 3 & 17 & 3.5 & 1.7 & 8 \\ 0.05 & 10 & 17 & 0.1 & 8 & 14 \\ 3 & 3.5 & 1.7 & 10 & 17 & 8 \\ 17 & 8 & 0.05 & 10 & 0.1 & 14 \end{pmatrix}$$

$$\begin{pmatrix} 0.1312 & 0.1696 & 0.5569 & 0.0124 & 0.8283 & 0.5886 \\ 0.2329 & 0.4135 & 0.8307 & 0.3736 & 0.1004 & 0.9991 \\ 0.2348 & 0.1451 & 0.3522 & 0.2883 & 0.3047 & 0.6650 \\ 0.4047 & 0.8732 & 0.5743 & 0.8828 & 0.1091 & 0.0381 \end{pmatrix}$$

The global minimizer is known as:

$$(0.2016, 0.1501, 0.4769, 0.2753, 0.3117, 0.6573)^T.$$

(h) Shekel function

$$minf(x) = \sum_{j=1}^{m} (\|x - s_i\|^2 + c_i)^{-1} [100(x_{i+1} - {x_i}^2)^2 (1 - x_i)^2]$$

s.t $0 \leq x_i \leq 10, \quad i = 1,2, \ldots, 4,$

wherin s_i is ith row of matrix S and c_i is ith element over vector C.

$$c = (0.1, 0.2, 0.2, 0.4, 0.4, 0.6, 0.3, 0.7, 0.5, 0.5)$$

And $s = \begin{pmatrix} 4 & 4 & 4 & 4 \\ 1 & 1 & 1 & 1 \\ 8 & 8 & 8 & 8 \\ 6 & 6 & 6 & 6 \\ 3 & 7 & 3 & 7 \\ 2 & 9 & 2 & 9 \\ 5 & 5 & 3 & 3 \\ 8 & 1 & 8 & 1 \\ 6 & 2 & 6 & 2 \\ 7 & 3.6 & 7 & 3.6 \end{pmatrix}$,

The entire minimum requirement is $(4\ ,4\ ,4\ ,4)^T$ for m $=\ 5, 7$, and 10. The comparable minimum values are, respectively, $-10.1523, -10.4028$, and -10.5363.

(i) Generalized Rosenbrock function

$$min\ f(x) = \left\{\sum_{i=1}^{n-1}[100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2]\right\},$$

s.t -3$\leq x_1 \leq 3\ ,\quad i = 1,2, \ldots, n.$

The global minimum solution is $n\ \ (1\ , \ldots ,1)$ we take $n = 10\ ,20\ ,30$ in the experiments, respectively.

The numerical results of applying the proposed filled function method to the above test functions are presented. Some symbols used in the following Tables 4.1 and 4.2 are given as follows:

No: The number of the problems;

n: The dimension of the problems;

test(i): the first key point. For each test problem shown in Table 4.1, we randomly select four initial points to assess the effectiveness of the proposed algorithm;

x_i : A starting position $x_1 \in \Omega$ It is generated at random during the prescribed periods.

x^* : The provided approach yielded the best result.

The four beginning positions for problem (g) and $n\ =\ 6$ are taken as

$cG1 = (0.8909\ \ ,0.9593\ \ ,0.5472\ , 0.1493\ , 0.2575)^T$

$CG2 = (0.8407\ ,0.2511\ ,0.8143\ ,0.2435\ ,3500)^T$

$CG3 = (0.1966\ ,2543\ ,0.6160\ ,0.4733\ ,0.5317\ ,0.8308\ )^T$

$CG4 = (01.966\ ,0.2511\ 0.6160\ ,0.4733\ ,0.5317\ ,0.8308\ )^T$

The four initial points for problem (h) and m = 5 are taken as

$ch51 = (3.8045\ , 5.6782\ ,0.7585\ , 0.5395)^T$

$ch52 = (5.3080\ ,7.7919\ ,9.3401\ ,1.2991)^T$

$ch53 = (5.6882\ ,4.6939\ ,0.1190\ ,3.3712)^T$

$ch54 = (1.6218\ ,7.9428\ ,3.1122\ ,5.2853)^T$

The four initial positions for problem (h) and m = 7 are taken as

Table 4.1 Choosing a starting place.

No	N	Test (1)	Test (2)	Test (3)	Test (4)
A	2	1.8883 2.4384	-2:2381 2:4803	0:7924 -2:4184	-1:3290 -1:2813
B	2	2.7450 -1.7893	-2:0543 2:8236	2:7430 -0:0877	1:8017 -2:1478
C	2	-0.4694 1.5464	1:7532 2:7570	0:9344 -2:7857	2:0948 2:6040
D	2	1.0724 2.4944	1:4588 -0:6466	0:9329 -1:4944	1:2363 -2:8090
E	2	-1:3385 -2:7230	-2:4172 1:9407	1:1690 -1:0974	2:7013 -2:7933
F	2	5:3103 5:9040	_6:2625 _0:2047	-1:0883 2:9263	-4:4795 3:5941
G	3	0:6551 0:1626 0:1190	0:4984 0:9597 0:3404	0:5853 0:2238 0:7513	0:2551 0:5060 0:6991
h	6 $4, m = 5$ $4, m = 7$ $4, m = 10$	Cg1 Ch51 Ch71 Ch101	Cg2 Ch52 Ch72 Cg102	Cg3 Ch53 Cg73 Ch103	Cg4 Ch54 Ch74 Ch104
I	10 20 30	 Ci101 Ci102 Ci103	Ci102 Ci202 Ci302	Ci103 Ci203 Ci303	Ci104 Ci204 Ci304

Table 4.2 The results of experiments

No	N	Sb	x_1	x^*	Iter
A	2	0.5	Test (1)	1.0e-007*$(0.0279,0.5301)^T$	5
		0.3	Test (2)	1.0e-007*$(0.0093,-0.1309)^T$	5
		1	Test (3)	1.0e-008*$(0.5375,-0.5591)^T$	3
		0.8	Test (4)	1.0e-007*$(0.5375,-\_0.5591)^T$	4
b	2	0.5	Test (1)	$(-0.0898,-0.7127)^T$	9
		0.3	Test (2)	$(-0.0898,-0.7127)^T$	12
		1	Test (3)	$(-0.0898,-0.7127)^T$	8
		0.8	Test (4)	$(-0.0898,-0.7127)^T$	8
C	2	1	Test (1)	$(-0.0000,-1.0000)^T$	5
		0.5	Test (2)	$(-0.0000,-1.0000)^T$	5
		0.5	Test (3)	$(-0.0000,-1.0000)^T$	40
		0.3	Tes(4)	$(-0.0000,-1.0000)^T$	29
D	2	0.3	Test (1)	1.0e-007*$(-0.0754,-0.1631)^T$	89
		0.1	Test (2)	1.0e-005*$(-0.1733,-0.0355)^T$	61
		0.5	Test (3)	1.0e-005*$0.1596,\_0.0009)^T$	121
		0.5	Test (4)	1.0e-005*$(\_0.0383,\_0.7667))^T$	121
e	2	0.8	Test (1)	$(1.0000,1.0000)^T$	11
		1	Test (2)	$(.0000,0.9999)^T$	8
		1	Test (3)	$(1.0000,1.0000)^T$	8
		1	Test (4)	$(1.0000,1.0000)^T$	8

Table 4.2 The results of experiments(continue)

f	2	10.5	Test (1)	$(4.8581,5.4829)^T$	18
		0.3	Test (2)	$(\_7.7083,\_0.8003)^T$	16
		0.2	Test (3)	$(\_1.4251,5.4829)^T$	12
			Test (4)	$(\_7.7083,5.4829)^T$	18
g	3	0.5	Test (1)	$(0.1146,0.5556,0.8525)^T$	2
		0.8	Test (2)	$(0.1146,0.5556,0.8525)^T$	3
		0.2	Test (3)	$(0.1146,0.5556,0.8525)^T$	2
		1	Test (4)	$(0.1146,0.5556,0.8525)^T$	2
g	6	1	Test (1)	$x^*(g1)$	29
		0.8	Test (2)	$x^*(g2)$	38
		0.5	Test (3)	$x^*(g3)$	20
		0.5	Test (4)	$x^*(g4)$	28
h	$4, m = 5$	1	Test (1)	$x^*(h51)$	20
		1	Test (2)	$x^*(h52)$	25
		1	Test (3)	$x^*(h53)$	17
		1	Test (4)	$x^*(h54)$	13
H	$4, m = 7$	1	Test (1)	$x^*(h71)$	19
		1	Test (2)	$x^*(h72)$	17
		1	Test (3)	$x^*(h73)$	13
		1	Test (4)	$x^*(h74)$	9
H	$4, m = 10$	1	Test (1)	$x^*(h101)$	16
		1	Test (2)	$x^*(h102)$	10
		1	Test (3)	$x^*(h103)$	12
			Test (4)	$x^*(h104$	9

Table 4.2 The results of experiments(continue)

I	10	0.5	Test (1)	$x^*(i101)$	19
		1	Test (2)	$x^*(i101)$	19
		1	Test (3)	$x^*(i101)$	14
		1	Test (4)	$x^*(i101)$	23
I	20	1	Test (1)	$x^*(i201)$	26
		1	Test (2)	$x^*(i202)$	18
		1	Test (3)	$x^*(i203)$	29
		1	Test (4)	$x^*(i204)$	21
I	30	1	Test (1)	$x^*(i301)$	12
		1	Test (2)	$x^*(i301)$	14
		1	Test (3)	$x^*(i301)$	19
		1	Test (4)	$x^*(i301)$	24

Table 4.2 illustrates that the process of finding a global optimal solution typically requires only a few adjustments to the parameters for most problem instances. This implies that the computational effort involved is minimal, highlighting the efficiency and effectiveness of the proposed algorithm. It is important to note that the choice of sb has a significant impact on the experimental outcomes. sb determines the parameter Q, and it is advisable to select a small sb to avoid missing the global minimizer of the objective function. However, this approach may require more parameter modifications. Employing a trial-and-error method becomes acceptable and beneficial to determine of sb, ensuring that Q falls within the bounds specified in Theorems 3.2 and 3.3.

5. CONCLUSIONS AND RECOMMENDATION

5.1 Conclusions

The function of filled approach is a powerful strategy for achieving universal optimizations. Nevertheless, current responsibilities have been filled suffer from various limitations, including non-differentiability, excessive parameterization, and unfavorable conditions, among others. These drawbacks pose challenges in finding the best solutions worldwide. To overcome these limitations, this research introduces a continuously differentiable filled function with a single parameter, aiming to address these drawbacks effectively.

Traditional filled functions have certain limitations that can impede their ability to find the global optimum. These limitations include non-differentiability, excessive parameterization, and unfavorable conditions. Non-differentiability can make it challenging to implement derivative-based optimization techniques as traditional filled functions may not be differentiable at certain points. Excessive parameterization, on the other hand, can increase computational complexity and lead to difficulties in determining optimal parameter values. Certain filled functions may also have restrictive conditions or assumptions that limit their applicability to a wide range of optimization problems.

To overcome these limitations, researcher have proposed a novel approach that introduces a only one parameters perpetually distinct filled functional. By ensuring differentiability, this method enables the use of derivative-based optimization methods, which can lead to more efficient and accurate optimization results. The use of a single parameter simplifies the optimization process by reducing the parameter space and making it easier to find the optimal parameter value. This can help overcome the challenges associated with excessive parameterization.

To evaluate the efficiency and effectiveness of the proposed filled function method, numerical experiments were conducted on a selection of test optimization problems. These experiments likely involved applying the novel filled function to various optimization problems and comparing the results with existing approaches. The goal of this research is to provide an improved filled function method that addresses the limitations of existing approaches and enhances the global optimization process through differentiability, simplicity, and efficiency. The proposed approach can help overcome the limitations of traditional filled functions that can hinder their effectiveness in finding the global optimum. By addressing these limitations, the proposed approach can enhance the optimization process and provide more efficient and accurate results. Moreover, the proposed approach can be widely applicable to various optimization problems owing to its simplicity and differentiability.

The proposed approach is expected to enable the use of derivative-based optimization techniques, which are known to be more efficient and accurate compared to non-derivative-based techniques. The use of a single parameter can make the optimization process simpler and less complex. Moreover, it can make it easier to determine the optimal parameter value, which can lead to more accurate and efficient optimization results. The numerical experiments conducted in this research can offer insights into the effectiveness and efficiency of the proposed filled function method. The results of these experiments can help in determining the advantages and limitations of the proposed approach compared to existing approaches.

Overall, the proposed approach can be considered a significant contribution to the field of optimization. It addresses the limitations of traditional filled functions and provides a more effective and efficient approach to finding the global optimum. The simplicity and differentiability of the proposed approach make it widely applicable to various optimization problems. The proposed approach can help overcome the challenges associated with excessive parameterization and restrictive conditions faced by traditional filled functions. Overall, the proposed approach can contribute to the development of more accurate and efficient optimization techniques that can address various real-world problems.

5.2 Recommendation

1. The refinement of the filled function method is an area that requires further investigation. There is a need to focus on refining the proposed filled function method to enhance its performance and overcome existing limitations. This could involve exploring different formulations or modifications of the filled function to improve its convergence properties, robustness, or applicability to a wider range of optimization problems. The development of an improved filled function method is crucial as it will contribute to the efficiency of global optimization techniques.

2. Conducting more extensive comparative studies is important to evaluate the performance of the proposed filled function method against other existing global optimization techniques. This can involve benchmarking the method on a broader set of test problems and comparing its efficiency, accuracy, and scalability with alternative approaches. The results of such studies will help to identify the strengths and weaknesses of the filled function method, and how it compares to other optimization techniques.

3. Theoretical analyses are necessary to provide a deeper understanding of the properties and characteristics of the proposed filled function method. This could involve investigating convergence guarantees, establishing theoretical bounds on the method's efficiency or convergence rate, or studying the impact of different parameter choices on the performance of the method. A thorough theoretical analysis will provide insights into the underlying principles of the filled function method and its applicability to different optimization problems.

4. Applying the proposed filled function method to real-world optimization problems from various domains, such as engineering, finance, or machine learning, is essential to assess its practical effectiveness. This can help identify potential challenges or limitations in real-world scenarios, and provide insights into adapting the method to specific application domains. Real-world applications will demonstrate the robustness and versatility of the filled function method.

5. Exploring the extension of the filled function method to handle multi-objective optimization problems is another area that requires investigation. This can involve developing approaches that integrate the concept of filled functions with multi-objective optimization algorithms, allowing the identification of a set of Pareto-optimal solutions. Such an extension will enhance the applicability of the filled function method to a wider range of optimization problems, including those with multiple objectives.

REFERENCES

Adjiman, C. S., Dallwig, S., Floudas, C. A., Neumaier, A. and Zavala, V. M. 1998. A global optimization method, αBB, for general twice-differentiable constrained NLPs—II. Implementation and computational results. Computers and Chemical Engineering, 22(9), 1151-1179.

Alefeld, G. and Herzberger, J. 2012. Introduction to interval computation. Academic press, 352 page, New York.

Akay, B., Karaboga, D. and Ozturk, C. 2007. Artificial bee colony (ABC) optimization algorithm for training feed-forward neural networks. In International Conference on Artificial Intelligence and Neural Networks (pp. 789-798). Springer.

Akay, B., Karaboga, D. and Aydin, D. 2012. A new feature selection method based on Artificial Bee Colony algorithm: A case of study in decision tree classification. Expert Systems with Applications, 39(9), 7967-7973.

Anderssen, R. S. and Bloomfield, P. 1975. Stochastic optimization. Operations Research, 23(2), 292-309.

Androulakis, I. P., Maranas, C. D. and Floudas, C. A. 1995. Primal-dual branch and bound algorithm for global optimization in generalized geometric programming. Journal of Optimization Theory and Applications, 84(3), 573-585.

Arora, S. and Anand, P. 2019. Chaotic grasshopper optimization algorithm for global optimization. Neural Computing and Applications, 31, 4385-4405.

Basso, E. 1982. A deterministic approach to global optimization. Mathematical Programming, 24(3), 362-369.

Brinkhuis, J. and Tikhomirov, V. 2005. Optimization: insights and applications. Princeton University Press, 658 page, USA.

Chakraborty, P., Nama, S. and Saha, A. K. 2023. A hybrid slime mould algorithm for global optimization. Multimedia Tools and Applications, 82(15), 22441-22467.

Chong, E. K. P. and Zak, S. H. 2001. An Introduction to Optimization, Jone Wiley and Sons. Inc., Canada.

Chowdhury, S., Rao, A. V. and Sahinidis, N. V. 2000. Tunneling algorithms for global optimization. Journal of Global Optimization, 18(1), 25-50.

Ekren, B. Y. and Ekren, N. 2010. A hybrid optimization algorithm for solving optimization problems. In 2010 International Conference on Mathematical Methods in Engineering (MME) (pp. 156-159). IEEE.

Fletcher, R. 2000. Practical methods of optimization. John Wiley and Sons, 464 page, USA.

Floudas, C. A. and Visweswaran, V. 1990. Primal-dual branch and bound algorithm for global optimization. Journal of Global Optimization, 1(2), 139-172.

Floudas, C. A. and Visweswaran, V. 1993. Primal-dual branch and bound algorithm for global optimization: I. Convex underestimation. Computers and Chemical Engineering, 17(3), 211-227.

Floudas, C. A. and Pardalos, P. M. (Eds.). 2008. Encyclopedia of optimization. Springer Science & Business Media, 4622 page, USA.

Ge, R. 1999. A filled function method for finding a global minimum of a function of several variables. Nonlinear Analysis: Theory, Methods and Applications, 36(7), 931-939

Ge, R. and Qin, Z. 1987. A filled function method for finding a global minimum of a function of several variables. Journal of Optimization Theory and Applications, 53(2), 243-260.

Grimstad, A. and Sandnes, F. E. 2016. A branch-and-bound algorithm for global optimization using interval arithmetic. Journal of Global Optimization, 65(3), 569-586.

Groenen, P. J. F. and Heiser, W. J. 1996. Tunneling algorithms for global optimization. Journal of Optimization Theory and Applications, 91(2), 299-314.

Hansen, E. and Walster, G. W. (Eds.). 2003. Global optimization using interval analysis: revised and expanded (Vol. 264). CRC Press, 728 page, New York.

Hartikainen, M., Miettinen, K. and Wiecek, M. M. 2012. PAINT: Pareto front interpolation for nonlinear multiobjective optimization. Computational optimization and applications, 52, 845-867.

Horst, R., Pardalos, P. M., and Van Thoai, N. 2000. Introduction to global optimization. Springer Science and Business Media.

Horst, R. and Pardalos, P. M. (Eds.). 2013. Handbook of global optimization (Vol. 2). Springer Science & Business Media, 880 page, USA.

Huyer, W. and Neumaier, A. 1999. Global optimization by multilevel coordinate search. Journal of Global Optimization, 14, 331-355.

Jones, D. R., Schonlau, M. and Welch, W. J. 1993. Efficient global optimization of expensive black-box functions. Journal of Global Optimization, 13(4), 455-492.

Karaboga, D. and Basturk, B. 2007. A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm. Journal of Global Optimization, 39(3), 459-471.

Kennedy, J. and Poli, R. 2007. Particle swarm optimization. Encyclopedia of Machine Learning, 760-766.

Lera, D. and Sergeyev, Y. D. 2015. Deterministic global optimization using space-filling curves and multiple estimates of Lipschitz and Hölder constants. Communications in Nonlinear Science and Numerical Simulation, 23(1-3), 328-342.

Lin, Q., Wang, X. and Li, L. 2018. A novel approach based on particle swarm optimization for multimodal optimization problems. Applied Soft Computing, 71, 200-213.

Liu, Z. 2000. Filled function method for global optimization. Journal of Global Optimization, 16(2), 175-195.

Liu, Z. 2001. A new filled function method for finding global minimizers of optimization problems. Journal of Computational and Applied Mathematics, 133(1-2), 219-228.

Liu, Z. 2003. A filled function method for finding a global minimum of a function of several variables. Journal of Global Optimization, 27(3), 373-387.

Liu, Z. and Xu, Y. 2000. A filled function method for finding a global minimum of a function of several variables. Journal of Global Optimization, 16(3), 301-320

Liu, Y., Liu, J. and Wen, X. 2017. A novel approach based on an extended differential evolution algorithm for multimodal optimization. Soft Computing, 21(9), 2381-2403.

Mascarenhas, W. F. 2004. The BFGS method with exact line searches fails for non-convex objective functions. Mathematical Programming, 99(1), 49.

Mirjalili, S., Mirjalili, S. M. and Hatamlou, A. 2016. Multi-verse optimizer: a nature-inspired algorithm for global optimization. Neural Computing and Applications, 27, 495-513.

Montalvo, A. and Levy, A. V. 1985. Tunneling algorithms for global optimization. Journal of Optimization Theory and Applications, 47(2), 311-320.

Nocedal, J. and Wright, S. J. 2006. Large-scale unconstrained optimization. Numerical optimization, 164-192.

Renpu, L. 1990. Filled function method for finding a global minimum for a function of several variables. Journal of Optimization Theory and Applications, 67(3), 413-431.

Ren-Pu, G. 1987. The theory of filled function method for finding global minimizers of nonlinearly constrained minimization problems. Journal of Computational Mathematics, 1-9.

Sahiner, Y., Karaboga, D. and Ozturk, C. 2012. A novel filled function method for constrained optimization problems. Journal of Global Optimization, 52(2), 259-278.

Sahiner, Y., Yilmaz, M. and Karaboga, D. 2017. A novel approach based on the harmony search algorithm for the multimodal optimization problems. Applied Soft Computing, 51, 458-482.

Sahiner, Y. and Ibrahem, O. 2019. A novel approach based on a hybrid particle swarm optimization algorithm for multimodal optimization problems. Applied Soft Computing, 75, 484-503.

Schäffler, S. 2012. A survey of stochastic optimization methods. International Transactions in Operational Research, 19(2), 231-253.

Shewchuk, J. R. 1994. An introduction to the conjugate gradient method without the agonizing pain, 1994. URL http://www-2. cs. cmu. edu/jrs/jrspapers. html# cg.

Stork, J., Eiben, A. E. and Bartz-Beielstein, T. 2020. A new taxonomy of global optimization algorithms. Natural Computing, 1-24.

Storti, D. W. and team. 2015. Genetic algorithm: A review. Journal of Mathematical Modelling and Algorithms, 14(2), 87-129.

Suman, B. and Kumar, R. 2006. Simulated annealing: A review. Journal of Mathematical Modelling and Algorithms, 5(3), 331-361.

Sun, W. and Yuan, Y. X. 2006. Optimization theory and methods: nonlinear programming (Vol. 1). Springer Science and Business Media, 687 page, USA.

Wang, X., Zhou, S. and Ruan, N. 2015. Filled function method for non-smooth global optimization. Journal of Global Optimization, 61(3), 563-581.

Wang, Z., Ding, H., Yang, Z., Li, B., Guan, Z. and Bao, L. 2022. Rank-driven salp swarm algorithm with orthogonal opposition-based learning for global optimization. Applied Intelligence, 1-43.

Wei, C., Chen, X. and Zhang, Y. 2014. A filled function method for non-smooth global optimization problems. Journal of Industrial and Management Optimization, 10(3), 717-736.

Wu, Z., Xu, Y. and Zhang, X. 2005. A global tunneling algorithm for global optimization. Journal of Global Optimization, 33(3), 379-394.

Wu, Z., Xu, Y. and Zhang, X. 2007. A global descent tunneling algorithm for global optimization. Journal of Global Optimization, 37(2), 303-319.

Wu, Z., Xu, Y. and Zhang, X. 2011. Global descent method for global optimization. Journal of Global Optimization, 51(1), 1-19.

Xu, Y., Wu, Z. and Zhang, X. 2001. A global tunneling algorithm for global optimization. Proceedings of the 20th Chinese Control Conference, 2, 1075-1077.

Xu, L., Chen, P. and Lee, H. 2015. Global optimization by tunneling with response surface approximation. Journal of Global Optimization, 63(1), 87-102.

Yilmaz, M. and Sahiner, Y. 2017. A novel approach for multimodal optimization problems: Binary particle swarm optimization with mutation operator. Applied Soft Computing, 60, 320-338.

Yuan, Y., Zhang, L., Zhang, S. and Yao, J. 2016. A filled function method for non-smooth optimization. Journal of Global Optimization, 66(2), 325-348.

Ziadi, Y., Bouarar, T. and Naceur, M. 2016. Space-filling curves strategies for global optimization. Journal of Computational Science, 17, 454-463.

Zhang, L., Wang, Y. and Liu, S. 2009. Filled function method for solving systems of nonlinear equations. Journal of Computational and Applied Mathematics, 230(1), 1-11.

Zhang, L., Liu, L., Yang, X. S. and Dai, Y. 2016. A novel hybrid firefly algorithm for global optimization. PloS one, 11(9), e0163230.

CURRICULUM VITAE

Personal Information

Name and Surname : Myasar Shihab Hamad HAMAD

Education

MSc	Çankırı Karatekin University Graduate School of Natural and Applied Sciences Department of Mathematics	2020-2023
Undergraduate	Tikrit University Faculty Mathematical and Computer Sciences Department of Mathematics	2014-2015

Work Experience

Year	Institution	Position
2021-Present	College of Education for Human Sciences	Documents section