Republic of Iraq
Ministry of Higher Education
and Scientific Research
University of Babylon
College of Engineering
Department of Electrical Engineering

Side Lobes Level Reduction of Linear Antenna Array Using Intelligent Optimization Algorithms

A Thesis
Submitted to the College of Engineering / University of Babylon in Partial Fulfillment of the Requirements for the Degree of Master in Engineering / Electrical Engineering / Communications

By

Huda Asaad Abd Al-Ameer Ali

Supervised by

Prof. Dr. Saad Saffah Hassoun

2024 A.D. **1445 A.H.**

بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيم

﴿ فَتَعَالَى اللَّهُ الْمَلِكُ الْحَقُّ وَلَا تَعْجَلْ بِالْقُرْآنِ مِنْ قَبْلِ أَنْ يُقْضَى إِلَيْكَ وَحْيُهُ وَقُلْ رَبِّ زِدْنِي عِلْمًا ﴾

صَدَقَ اللَّهُ العلي العظيم

القران الكريم–سورة طه الآية (114)

Dedications

It was challenging, but I completed it. I want to express my heartfelt dedication to my amazing parents. They truly deserve all the appreciation for their unwavering support and guidance throughout my journey to obtaining my master's degree. Thank you very much!

Words can barely express how grateful and amazed I am.

There has been a great source of inspiration, encouragement, and direction for me. You have truly inspired me to strive for success in everything I do, persevere no matter what, and I have unwavering confidence in myself. I appreciate all your kind wishes and prayers for me. They have always given me the courage I needed.

I would also like to dedicate this work to my amazing brothers and sisters, who have always believed in me and my ability to succeed.

Lastly, I would like to dedicate my work to my loyalty to Mrs. Fatima Al-Zahra (peace be upon her) and in preparation for the emergence of Imam Mahdi (may God hasten his arrival) and all the scholars.

Acknowledgments

I express my utmost gratitude and reverence to my God, "ALLAH," the Supreme Being, for bestowing upon me the ability to successfully accomplish this task, despite the numerous challenges encountered along the way.

I would like to express my sincere gratitude to my parents for their unwavering spiritual support throughout my life, as well as to my sisters and brother who have consistently believed in me along the way.

I would like to extend my utmost gratitude to my supervisor Prof. Dr. Saad Saffah Hassoun, for his unwavering support throughout my thesis. His exceptional comprehension of the subject matter and prompt responsiveness to the challenges I encountered were invaluable. Additionally, his guidance and advice have greatly contributed to my personal growth and knowledge acquisition. Furthermore, his unwavering patience, enthusiasm, motivation, and guidance proved invaluable throughout the entire duration of this thesis. He also imparted to me the necessary methodology to effectively present this thesis with utmost clarity.

My thanks also go to all the people who work within the Department of Electrical Engineering /College of Engineering/University of Babylon/Iraq for their help in many matters. While it is impossible to list everyone here, please know that your contributions are deeply appreciated and will be remembered indefinitely. Additionally, I would like to express my heartfelt thanks to my friends for their unwavering encouragement and care throughout this research.

Huda Asaad

Tables of contents

List of Tables

List of Figures

List of Abbreviations

Abbreviation	Definition
ACO	Ant Colony Algorithm
AF	Array factor
ALO	Ant Lion Optimization
BBO	Biogeography Based Optimization
CAA	Crucial Antenna Array
DE	Differential Evolution
FNBW	First Null Beam Width
FPA	Flower Pollination Algorithm
GA	Genetic Algorithm
GWO	Grey Wolf Optimization
HPBW	Half-Power Beam Width
IWO	Invasive Weed Optimization
LAAs	Linear Antenna Arrays
MFO	Moth Flame Optimization
MSLL	Maximum Side Lobe Level
MVO	Multi-Verse Optimization
PSLL	Peak Side Lobe Level
PSO	Particle Swarm Optimization
RAA	Random Antenna Array
SCA	Sine Cosine Algorithm
SMO	Spider Monkey Optimization
SSA	Sparrow Search Algorithm
TDR	Traveling Distance Rate
ULA	Uniform Linear Array
WEP	Wormhole Existence Probability
WOA	Whale Optimization Algorithm

List of Symbols	
Symbols	**Definition**
$\vec{A}$	coefficient vector
$A_{i,j}$	the value of the j^{th} variable or dimension of the i^{th} ant
$AL_{i,j}$	the value of the j^{th} variable (dimension) of i^{th} antlion
$Antlion_j^t$	the position of selected j^{th} antlion at t^{th} iteration
Ant_i^t	the position of i^{th} ant at t^{th} iteration.
a_i	the minimum of random walk of i^{th} variable
$\vec{a}$	is linearly decreased from 2 to 0 over the course of iterations
b	the geometry of the logarithmic spiral
C_1& C_2	Are the position weighting factors
$\vec{C}$	coefficient vector
$cumsum$	the cumulative sum
c^t	the minimum value among all variables at t^{th} iteration
c_i^t	the minimum of i^{th} variable at t^{th} iteration
D_i	reveals the distance from the i^{th} moth for the j^{th} flam
$D(\theta)$	Directivity of array
$\vec{D}$	Foraging behavior for wolves in the t^{th} iteration
$\overrightarrow{D_\alpha}$	Foraging behavior for wolves alpha in the t^{th} iteration
$\overrightarrow{D_\beta}$	Foraging behavior for wolves beta in the t^{th} iteration
$\overrightarrow{D_\delta}$	Foraging behavior for wolves delta in the t^{th} iteration
d	spacing between the elements in the array
d	the number of parameters
d^t	the vector including the maximum of all variables at t^{th} iteration

d_n^C	Arc distance between the n^{th} and $(n+1)^{th}$ elements
d_n^L	Spacing between the n^{th} and $(n+1)^{th}$elements
d_i^t	the maximum of i^{th} variable at t^{th} iteration
$E_{total}(\theta, \emptyset)$	Far field radiated field
F_i	the fitness value of the i^{th} sparrow
F_g	global optimal
F_j	the j^{th} flam
F_ω	the worst fitness value of the current population
g_{best}	Group best of the group
I_n^L	Excitation current of the antenna element
I_n^C	Excitation current
k	the direction of movement of the sparrows between [1, -1]
k	Wave number
L	number represents the Levy flights-based step size
L	One-dimensional matrix with all elements of 1
$L(\beta)$	Levy-flights based step size
l	the current number of iterations
lb_j	shows the lower bound of j^{th} variable
M_i	the i^{th} moth
M_{ant}	the matrix used to save the position of each ant
$M_{antlion}$	the matrix used to save the position of each antlions
N	the maximum number of flames
$NI(U_i)$	normalized inflation rate of the i^{th} universe
n	population size
P_i	the position of the location point in the i^{th} dimension
P_g	Position optimal for swarm
$P_{i,t}^d$	Individual Optimal position in d-dimension at initial position *i* to variable *t*
$P_{i,t+1}^d$	Individual Optimal position in d-dimension at initial position *i* to variable *t+1*
p	probability of switching between [0,1]

$\boldsymbol{p}$	the exploitation accuracy over the iterations
$\boldsymbol{p}$	Is a random number in [0,1].
$\boldsymbol{p_{best}}$	Personal best value of particles
$\boldsymbol{Q}$	random number
$\boldsymbol{R}$	Radial distance
$\boldsymbol{R_2}$	the alarm value between [0,1]
$\boldsymbol{R_A^t}$	the random walk around the antlion selected by the roulette wheel at t^{th} iteration
$\boldsymbol{R_E^t}$	the random walk around the elite at t^{th} iteration
$\mathbf{r}$	Random number between [0,1]
$\boldsymbol{r(t)}$	the stochastic function
$\boldsymbol{ST}$	The safety threshold between [0,1]
$\boldsymbol{S}$	the logarithmic spiral function
$\boldsymbol{s}$	pseudo-random step size
$\boldsymbol{T}$	specifies the maximum number of possible iterations
$\boldsymbol{TDR}$	Travelling Distance Rate
$\boldsymbol{t}$	current iterations
$\boldsymbol{t}$	a random number in [-1,1]
$\boldsymbol{t_{max}}$	maximum number of iterations
$\boldsymbol{U_i}$	shows the i^{th} universe
$\boldsymbol{ub_j}$	shows the upper bound of j^{th} variable
$\boldsymbol{V_i}$	Vector of velocity
$\boldsymbol{V_{i,t}^d}$	Individual Velocity in d-dimension at initial velocity *i* to variable *t* (Velocity before update)
$\boldsymbol{V_{i,t+1}^d}$	The velocity of swarm size in d-dimension in initial Velocity *i* to variable *t+1* (Velocity after update)
$\boldsymbol{WEP}$	Wormhole Existence Probability
$\boldsymbol{w}$	a constant defined based on the current iteration
$\boldsymbol{w}$	Inertia wight
X, Y, and Z	Chromosomal parameters
$\boldsymbol{X_{best}}$	the global optimal position
$\boldsymbol{X_p}$	The best position
$\boldsymbol{X_{worst}}$	The global worst position

X^*	the position vector of the finest solution found up
$X_\alpha, X_\beta, \& X_\delta$	the coordinates of alpha, beta, and delta wolves
$X(t+1)$	updated position of an omega wolf in the $(t+1)^{th}$ iteration
X_i	Swarm in space D-dimension
$X_{i,j}$	i^{th} sparrows location in the j^{th} dimension is indicated by the notation $X_{i,j}$
X_i^t	the positions of the current solution in i^{th} dimension at t^{th} iteration
$X_{i,t}^d$	location of swarm size in d-dimension in initial location *i* to variable *t* (position before the update)
$X_{i,t+1}^d$	location of swarm size in d-dimension in initial location *i* to variable *t+1* (position after update)
X_j	indicates the j^{th} parameter of best universe formed so far
$\vec{X}$	vector position of a grey wolf
$\overrightarrow{X_p}$	the vector representing the position of the prey
x_i^j	the j^{th} parameter of i^{th} universe
x_i^t	solution vector x_i at iteration t
x_k^t & x_j^t	pollination from different developing of the identical flowering plants
x_k^i	the j^{th} parameter of k^{th} universe selected by a roulette wheel selection mechanism.
α	a random integer that is uniform and falls between [0,1].
β	a random number
λ	wavelength
φ_n^L	Phase of the n^{th} antenna element
θ	Azimuth angle
$\emptyset$	Longitude angle
γ	scaling factor
ε	Value between [0,1]
$\Gamma(\beta)$	conventional gamma function
σ^2	Variance value

Abstract

It is possible to precisely and optimally construct antenna arrays like Linear Antenna Arrays (LAA) with characteristics like excellent directivity, signal gain, and wide coverage. An important feature in the design of antenna arrays is a reduction of Side Lobes Level (SLL) because it is an important factor for low-interference electromagnetic applications such as radar systems, communication systems, and wireless networks. This can be achieved by controlling and changing the amplitude, the element spacing, the position, or the phase.

In this thesis, ten optimization techniques have been applied to reduce the SLL, these optimization techniques are Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Flower Pollination Algorithm (FPA), Grey Wolf Optimization (GWO), Sparrow Search Algorithm (SSA), Moth Flame Optimization (MFO), Multi-Verse Optimization (MVO), Sine Cosine Algorithm (SCA), Whale Optimization Algorithm (WOA), and Ant Lions Optimization (ALO). LAA used in this compression was implemented with different geometry of 8, 16, 32, 64, 128, and 256 elements. In addition, the impact of changing the parameters values of each technique has been studied. These parameters are iterations, population size, max stall iterations, etc., all of which aim to reduce the side lobes better.

The results achieved by using the software package (MATLAB), in the first model, all algorithms gave the best SLL at a certain number of elements of an antenna array. All algorithms gave results that led to the maximum reduction of SLL through the distribution of antenna elements and for all their numbers perfectly, the SCA outperforms the rest of the algorithms at number of elements 256 by reducing the SLL to -29.2229 dB. This is because the elements are better distributed than the rest of the techniques and these

results change according to the specific conditions of rearranging the elements according to the type of antennas used.

The results of the second test show that changing parameters affects the reduction of the SLL although some parameters have minimal effect, this effect changes antennas' positions to better reduce SLL, these results change according to certain conditions represented by changing the geometry of the antenna array or changing the number of antenna elements.

1

CHAPTER ONE

'INTRODUCTION'

Chapter One

Introduction

1.1 Introduction

The rapid development of wireless communications such as cellular telecommunications, satellite, sonar, and radar in recent years has necessitated the efficient use of the existing frequency spectrum. Multiple antennas are used to form antenna arrays able to produce efficient and directed radiation because a single antenna cannot generate the desired radiation patterns [1].

Antenna arrays could be optimally and accurately designed to have properties such as high directivity, signal gain, and wide coverage. It is necessary to determine the element's positions, amplitudes, and phase angles to enhance the radiation pattern that characterizes antenna arrays in the far field. The region in the direction of the maximum radiation intensity is known as the main beam lobe next to the main lobe is the side lobes that should be canceled [2].

The primary objective of developing a radiation model with reduced Maximum Side Lobe (MSL) is to prevent interference with other systems of communication that operate in the same frequency range. The Half-Power Band Width (HPBW) of an antenna array refers to the angular width between points where the radiation intensity is half of the maximum. Reduction of SLL involves minimizing the radiation in undesired directions. Combining both concepts typically mean balancing a narrower main lobe (for better directionality) and reduced side lobes to enhance overall antenna performance, to provide the required high directivity for long-distance communication, a radiation pattern with a narrow HPBW is required.

Nonetheless, it is extremely challenging to construct the antenna array with a narrower HPBW and the lowest MSL, Because the MSL value decreases as the HPBW value rises. Obviously, the inverse is also true, so it is very difficult to accomplish a narrower HPBW while diminishing the MSL value [1].

Using metaheuristic algorithms, it is possible to construct an antenna array with a low MSL and a narrow HPBW despite the rise in electromagnetic pollution [3]. Based on geometric properties, antenna arrays are classified according to shape. Due to their simplicity, Linear Antenna Arrays (LAAs) are one of the most prevalent geometries in the literature [4].

To develop sufficient antenna technologies capable of the aforementioned characteristics in a circumscribed physical area determined by the flexibility request when it originates, there are still some technical challenges [5]. Wireless standards today consider numerous antenna technologies to accomplish longitudinal diversity, spatial multiplexing, and beam radiation to achieve greater precision. However, these features are acquired at the expense of increasing the system's complexity, which may not be tenable in tiny, low-cost devices that demand low power consumption [6,7].

Numerous global optimization algorithms, including Differential Evolution (DE) [8], Particle Swarm Optimization (PSO) [9], Genetic Algorithm (GA) [10], Ant Colony Algorithm (ACO) [11], Moth Flame Optimization (MFO) [12], Biogeography based optimization (BBO) [13], Grey Wolf Optimization (GWO) [14], Sparrow Search Algorithm (SSA) [15], Spider Monkey Optimization (SMO) [16], Multi-Verse Optimization (MVO) [17], Invasive Weed Optimization (IWO) [18], and Ant Lions Optimization (ALO) [19], etc. are used to minimize SLL by

using it with antenna arrays and retaining antenna properties that are amplitude, element position determination, and phase excitation to optimize the overall performance of the array [20].

1.2 Linear Antenna Arrays

Antenna arrays refer to configurations of multiple antennas working collectively as a single system. These arrays consist of two or more antennas arranged in specific geometries, such as linear, planar, or circular. By combining signals from individual antennas, antenna arrays offer benefits such as increased gain, and improved directivity.

LAA is a configuration of multiple antennas arranged in a straight line. It enhanced signals sent or received in specific directions, offering beamforming capabilities [21].

LAAs can employ electronic beam scanning techniques, where the direction of the main beam is electronically adjusted without physically reorienting the antennas. This allows for dynamic beam steering, making it useful in various applications like radar systems, and satellite communication [22].

LAA are called so because the arrangement of the antennas resembles a straight line and the physical layout of the antennas is linear, with each antenna element placed in a row or column along a common axis as shown in Fig. (1.1). This linear configuration is in contrast to other types of antenna arrays, where the antennas are arranged in different geometries. The linear arrangement of antennas in an array allows for specific directional properties and beamforming capabilities, making it a common and versatile design [23].

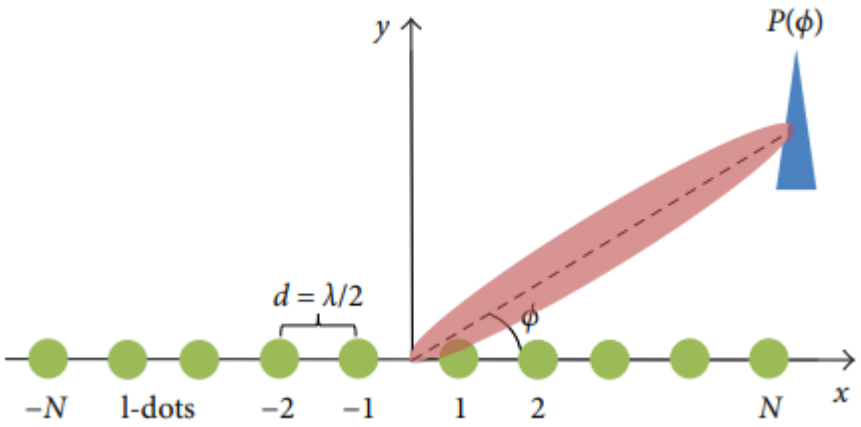


Figure (1.1). Geometry of 2N-element-symmetric LAA placed along the x-axis.

The radiated power patterns of antenna arrays can be controlled through various techniques, including element positioning, amplitude, and phase weighting, and beamforming. these techniques allow designers to tailor the radiation pattern of antenna arrays to suit different applications [21].

The LAAs have the important property is reducing the side lobe, they reduce installation limits in the traditional system and have great flexibility in controlling the excitation currents or position of elements [24]. It reduces the SLL by employing a fitness function specific to the antenna arrays. In addition, the fitness function is variable, which means that the decrease of the side lobes will also vary. It can be seen that impressive attempts have been made to reduce SLL by employing various optimization techniques [25].

1.3 Literature Review

Optimization techniques refer to a set of methods and algorithms used to find the best possible solution to a problem from a set of feasible solutions. These techniques are employed in various fields such as mathematics, and engineering.

The optimization technique's primary goal is to maximize or minimize a specific objective function while satisfying given constraints, these techniques have various applications ranging from engineering design and logistics optimization to financial modeling and machine learning [25]. A variety of optimization algorithms have emerged for antenna pattern synthesis of an array that has multiple applications. However, the implementation of these techniques is the most essential aspect [26].

In June 2012, Goswami and Mandal [10], used GA to minimize SLL by using a broadside linear antenna array Although a different number of antenna elements are used N = 12 is the best because it where GA was found to have reduced it to -13.06 dB.

Chatterjee et al., June 2013 [27], minimized SLL using a linear antenna array at N = 4 and using the PSO algorithm where SLL was reduced to -24.7854 dB. They were compared with other algorithms to find out the least SLL possible to obtain. In addition, the algorithms were compared by direction, Half power beam width, etc.

Pappula and Ghosh, June 2014 [28], introduced compared the PSO algorithm to other algorithms using a linear antenna array and showed that PSO minimized SLL to -18.92 dB, while the presented results in the paper highlight the improved performance of the antenna array synthesized using the proposed algorithms approach in comparison to traditional methods.

In April 2015, Devi et al., [29], used a GA to address the reduction of side lobes in thinned arrays. Thinned arrays are antenna arrays where some elements are selectively deactivated to achieve specific design goals tested with a different number of elements and SLL was reduced to -31.02 dB and the presented results in the paper demonstrate the capability of GA to enhance the radiation characteristics of thinned arrays.

Gangwar et al., October 2015 [30], introduced the authors utilized PSO, a nature-inspired optimization algorithm, to find optimal configurations for the thinned array that minimize side lobe levels. The study showcases the effectiveness of PSO in improving the radiation characteristics of the antenna array by reducing unwanted side lobes. The presented results in the paper highlight the capability of PSO to enhance the performance of thinned linear antenna arrays to reduce SLL at N = 100 and used several algorithms, including the PSO algorithm, where SLL was reduced to -22.58 dB.

Suggested a comparison between PSO and real-coded GA to optimize time-modulated linear antenna arrays with a focus on directivity maximization and achieving an optimal far-field pattern. Time modulation involves varying antenna parameters over time to enhance performance by Ram et al., December 2015 [31], the results showed that PSO is superior as SLL reduced to -35.21 dB at N=16, while GA decreased by -34.89 dB at N=16.

In March 2016, Saxena and Kothari [32], The authors apply FPA to find optimal configurations for the linear antenna array, aiming to enhance its performance. The study demonstrates the effectiveness of FPA in achieving optimization objectives for antenna array design FPA and PSO were compared using a linear antenna array and, the results showed that FPA is better as SLL was reduced to -23.45 dB at N=10 and PSO reduced it by its values -20.72 dB at N=10.

Saxena and Kothari, April 2016 [33], proposed LAA to reduce SLL at N = 10 and used numerous algorithms including the PSO algorithm and the GWO algorithm where GWO excelled by reducing SLL to -26.05 dB while PSO reduced it to -24.62 dB.

In September 2016, Saxena and Kothari [34], compared more than one algorithm and used LAA at N = 10. They compared the PSO algorithm to the ALO algorithm and found that the two had minimized SLL to the extent that the PSO and ALO algorithm reduced it by -20.72 dB and -23.29 dB respectively.

Das et al., September 2018 [35], The authors use MFO to find optimal configurations for both linear and circular antenna arrays, emphasizing the objective of minimizing undesired side lobes. The study demonstrates the effectiveness of MFO in achieving side lobe reduction in antenna array designs a comparison between MFO and PSO, as well as other different algorithms at N = 10 by using a linear antenna array and found that MFO is better because it reduces SLL to -26.07 dB but PSO decreases SLL to -24.62 dB.

In April 2019, Borah et al. [36], used a PSO algorithm with a linear antenna array at N = 15 to reduce side lobe level, the results showed that PSO reduced it to -16.9560 dB these results vary according to parameters such as population size and iteration.

Li and Luk, March 2020 [37], the authors discuss its effectiveness in solving electromagnetic problems, particularly in antenna design and propagation. a comparison between GWO and PSO using a linear antenna array showed that the first is the best because it reduced SLL to -21.093 dB at N=37 while PSO reduced by -17.41 dB at N=10.

In July 2020, Supriya and Rao [38], the study evaluated and compared various optimization methods employed in the design process. The authors assess the effectiveness of these techniques in enhancing the performance of linear antenna arrays a comparison was tested between FPA and GA as well as other different algorithms at N = 16 using a linear antenna array and found

that FPA is better because it reduces SLL to -34.2557 dB, and GA reduced SLL to -31.3235 dB.

In January 2021, Durmus, and Kurban [1], The paper contributed to the field by presenting findings related to the equilibrium optimization algorithm's application in the design of both linear and circular antenna arrays, offering potential advancements in antenna technology. Compared to PSO, GA, and MFO, and combined with a circular antenna array, MFO turned out to be the best as SLL reduced to -17.79 dB at N=12.

In March 2021, Albagory and Alraddady [39], Introduced a technique that aims to enhance the symmetry of antenna radiation patterns by addressing sidelobe issues. The findings contribute to the field by presenting an effective strategy for sidelobe level reduction, potentially improving the overall performance of antennas a comparison was tested between FPA and WOA as well as other different algorithms at N = 16, and found that WOA is better because it reduces SLL to -38.53 dB, and FPA reduced SLL to -36.32 dB.

Performed compared SSA, WOA, and PSO algorithms by Liang et al., June 2021 [15], and found that the best reduction for side lobe level using a linear antenna array is SSA because it reduced SLL to -23.9233 dB the modified SSA playing a crucial role in achieving effective side lobe level reduction and then PSO to -23.7487 dB and then WOA to -22.9487 dB at N=16.

In July 2022, Jamunaa et al. [40], used LAA to reduce SLL at N = 28 and used numerous algorithms including the PSO algorithm, the GWO algorithm, and the MVO algorithm where MVO excelled by reducing SLL to -22.2765 dB while GWO and PSO reduced it to -22.1221 dB and -21.843 dB respectively.

In August 2022, Singh et al. [41], suggested comparing a range of techniques including Sine Cosine Algorithm SCA, GWO, and SAA to reduce SLL using a circular antenna array and found that the best algorithm that reduced SLL were GWO because it reduced to -30.66 dB while SSA and SCA reduced it to -29.47 dB and -19.41 dB respectively at N=20.

Li, and Ouyang in 2023 [42], suggested a comparison between GA, ALO, and other algorithms by using sub-array, and found that the best to reduce SLL was ALO because it reduced SLL to -22.9 dB while GA reduced it to -20.3 dB at N = 40. The study focuses on enhancing the performance of sub-arrays within larger antenna systems by utilizing the improved Ant Lion Algorithm.

In 2023, Mohammed [43], focus is on achieving deep sidelobe reduction and a narrow beam width. Compared a range of techniques including GA to reduce SLL using sub-array which reduced the side lobe level to -30.00 dB at N = 20.

Hu et al., February 2023 [44], in another research comparison between PSO, GA, and WOA used a planer antenna array, the best was WOA because it reduced SLL to -18.52 dB while PSO and GA reduced to -15.31 dB, and -15.03 dB respectively at a number of elements 64.

Liang et al., March 2023 [45], proposed comparison was made between PSO, SSA, and WOA using a linear antenna array, and found that the best algorithms were SSA because they reduced SLL to -14.10 dB while PSO and WOA reduced to -12.24 dB and -12.40 dB at N= 16.

On 31 March 2023, Asianuba and Precious [46], The study explores the effectiveness of algorithms as optimization techniques in enhancing the performance of linear array antennas by minimizing undesired side lobes. The findings contribute to the field by presenting algorithms as a potential

method for side lobe level reduction in antenna array design by comparing a range of techniques including PSO to reduce SLL using a linear antenna array which reduced SLL to -17.26 dB at N = 10.

In July 2023, Ghattas et al. [47], The study explored the effectiveness of various optimization techniques in optimizing the performance of planar antenna arrays for beamforming applications by a comparison between PSO, and GWO by using a planer antenna array was found to be the best as SLL reduced to -14.63 dB at N=10.

The researchers mentioned works show that the reduction of SLL is significant to direct the signal in the desired direction, high quality, and concentration of energy in the main lobe and to prevent its dispersion in unintended directions, so the researchers have achieved good results but it may possible to reach better results by applying different optimization techniques as shown in Fig. (1.2). the highlighted algorithms are investigated in this thesis and they then selected from the same type have been compared under the same circumstances all have been selected from Metaheuristics type and different types due to the population-based branch and characteristics to investigate their performance is minimizing SLL.

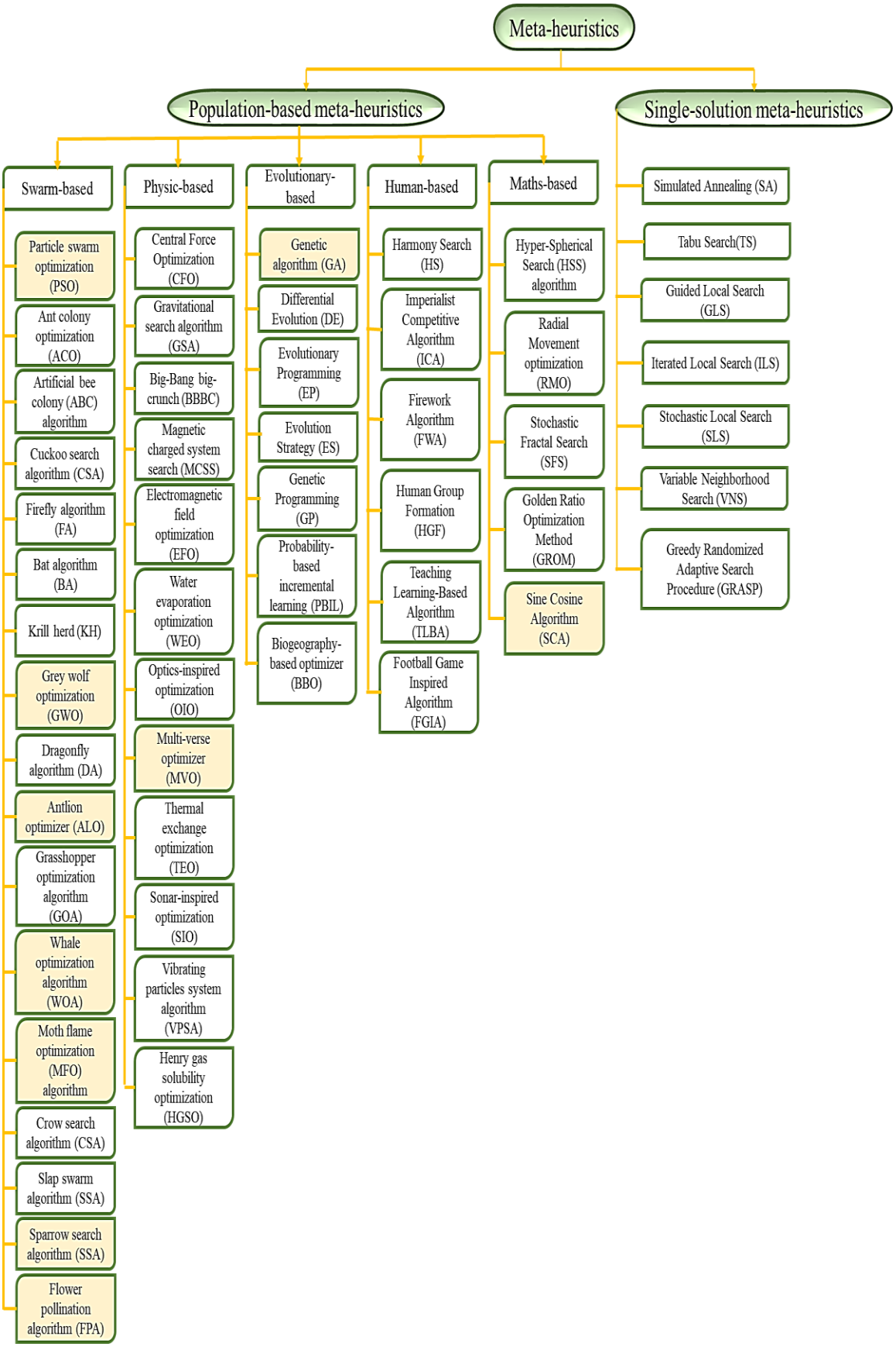


Figure (1.2). Classification of Meta-heuristics Algorithm.

1.4 Problem Statement

- The important problem is the level of the side lobes, where they take part of the power of a signal in unwanted directions, which affects the main lobe, side lobes reduce the power of transmitted signals and prevent their transmission over long distances and in a certain direction due to the concentration of power in unintended directions resulting in lower signal quality in the desired direction.

- Many optimization algorithms need to be investigated to minimize SLL by changing the values of their parameters on the performance of the antenna system.

1.5 Aims of the Work

- Based on the problem assigned in section (1.4) the work of this thesis will aim to reduce the SLL of the radiation pattern of an antenna array.
- The reduction of SLL can be achieved by examination using multiple algorithms. The algorithms used to test in this thesis PSO, GA, FPA, GWO, SSA, MFO, MVO, SCA, WOA, and ALO.
- The configuration used to test for the proposed algorithms will be in two studies.
    - The first study uses a fitness function to reduce SLL and know its position after optimization by using optimization algorithms with LAA for a different number of antenna elements and to compare the results of the proposed models and decide which one is the best for reducing the SLL.
    - The second study is to experience a change in the parameters of algorithms and find the best SLL suppression.

1.6 Outline of the Thesis

This thesis consists of five chapters as shown below:

- *Chapter one*, an introduction of the communication system and a Literature Review of the similar work of the proposed system with a description of the problem.
- *Chapter two*, presents LAA theory and a detailed explanation of optimization techniques.
- *Chapter three*, describes the use of LAA and the steps and parameters to apply optimization algorithms.
- *Chapter Four*, shows the results and discussions of the investigated system and algorithms and then compares the results with the related works.
- *Chapter Five*, illustrates the conclusions and the suggested future works.
- *References*

2

CHAPTER TWO

'BACKGROUND THEORY'

Chapter Two

Background Theory

2.1 Introduction

Electromagnetic waves in the air are sent and received at a specific frequency and bandwidth using metal objects called antennas. where antennas connect technological devices to space [2]. In the present time, antennas are one of the main components of modern wireless communication systems and are very important because they have become the subject of interest to many engineers and researchers in the field of communications. Many experiments have been conducted in previous years to show that antennas are an indispensable part of modern communication systems and devices such as portable cell phones and up to the most cutting-edge technology [48].

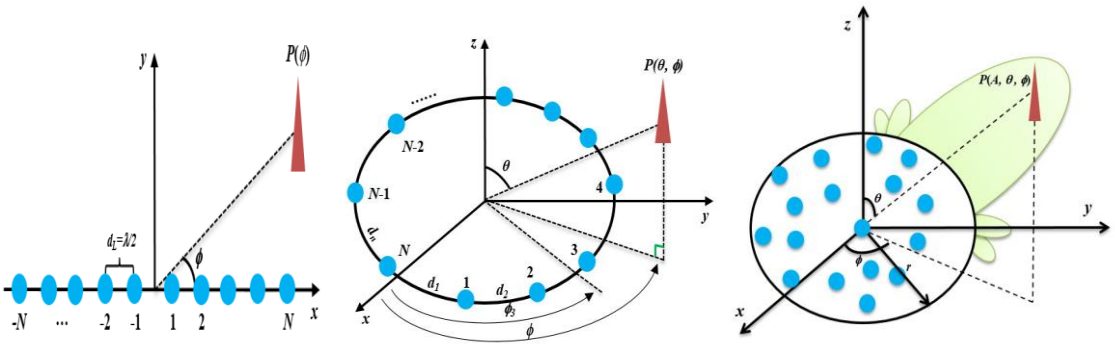
Antennas may contain one transmitter, and multiple receivers, such as industrial applications radios and remote sensors. Although the antennas vary in type, the operational format is the same since transponders are directional devices that aim to improve the main signal and reduce the side lobe [49].

Antenna arrays refer to a collection of multiple antennas that work together to enhance the performance of the overall communication system. Antenna arrays can provide various benefits, including increased signal strength, improved signal quality, and reduced interference. Optimization algorithms can optimize the performance of antenna arrays by adjusting the parameters of the antennas in the array. These algorithms can help improve the system's overall performance by finding the optimal configuration for the antenna array [50].

Several different optimization algorithms can be used to optimize antenna arrays. These algorithms can help to find the optimal antenna array configuration based on different performance criteria, such as maximizing the signal-to-noise ratio, minimizing interference, or optimizing the beam pattern [51].

To form the general pattern of the antenna array, a set of controls must be taken into account in the arrays containing identical elements that can be represented as follows [52]:

a. Arrays can be arranged in several ways (linear, circular, etc.), as shown in Fig (2.1).
b. The spacing of the elements.
c. The amplitude of each element's excitation.
d. The phase of individual parts that is excited.
e. Individual elements relative pattern.



(a) **(b)** **(c)**

Figure (2.1). Array antenna geometries: (a)Linear (b)Circular(c)Random [53].

2.2 Antenna Array

There are a range of simple devices that operate using a single antenna like a mobile phone. Although modern wireless communications require good performance such as high directivity radiation properties and narrow beam, these properties require the use of more than one antenna, because the

beam pattern of one antenna is relatively wide and with little directivity. To improve the performance and characteristics of the antenna either by increasing the antenna size or by using different forms of antenna arrays [54]. It is worth noting that increasing the size of the antenna is not feasible, so the use of several antenna arrays is the best solution to the exhibition of practical applications as it provides high gains and high direction to the main beam pattern for use in wireless applications such as long-range communication [55].

There are different types of antenna arrays, the type of array used depends on the application, and factors such as the desired coverage area, operating frequency, and physical constraints of the system are taken into consideration when designing an antenna array, antenna arrays are widely used for direction finding, beamforming, and interference cancellation in various wireless communication applications [56].

2.2.1 Beamforming

Beamforming is a technique used in antenna arrays to shape and direct the radiation pattern of the array in a specific direction. Adjusting the phase and amplitude of the signals transmitted or received by each antenna element in the array to produce constructive interference in the desired direction and destructive interference in other directions. By doing so, the signal strength in the desired direction is increased [57].

Other techniques are mutual coupling and aperture efficiency where mutual coupling refers to the interaction between two or more antennas in a system. When antennas are placed close to each other, the electromagnetic fields radiated by one antenna can influence the performance of the others, leading to mutual coupling effects, either Aperture efficiency refers to the

effectiveness of an antenna in capturing and converting incident electromagnetic waves into electrical signals.

It is especially useful in wireless communication systems, where the signal strength is often weak due to the distance between the transmitter and receiver or interference from other sources. By using beamforming techniques, the signal can be focused in the direction of the receiver, increasing the signal strength and improving the overall quality of communication [58].

2.2.2 Performance Metrics for Antenna Array

Different antenna array performance factors are taken into consideration to meet wireless communication's ongoing data traffic growth. An antenna array's performance is evaluated using a variety of factors. Some of the factors do, however, relate to one another. The properties of an antenna array's radiation pattern are typically used to gauge it. Radiation gain is used to evaluate the antenna arrays radiation pattern, SLL, directivity, beamwidth HPBW or First Null Beam Width (FNBW), and other factors [59].

These performance metrics are important for designing and optimizing the performance antenna arrays for specific applications. Different applications may have different requirements for these metrics, and it is important to choose the appropriate antenna array design based on the application's needs [60].

2.2.2.1 Antenna Array Radiation Pattern

As a graphic or mathematical description of the spatial distribution of the antenna array's radioactive energy as a function of directional space coordinates, the radiation pattern is the most important element of antenna array performance. It is possible to display the beam pattern in the systems

of spherical coordinates. It is possible to know how the antenna array radiates through the polar or spherical beam pattern and coordinates it is possible to know the parameters associated with the antenna array [48,61].

In general, the radiation pattern of an antenna array can be characterized by two primary features: the main lobe and the side lobes. The side lobes are the directions in which the antenna array radiates power that is not part of the main lobe. Side lobes are unwanted because they can cause interference with other signals and reduce the overall efficiency of the antenna array [60].

2.2.2.2 Side Lobes Level (SLL)

SLL is the performance metric of an antenna array that measures how well the radiated power of the antenna array is concentrated in the main beam. SLL is the relative height of the first side lobe down from the maximum point of the main lobe and mathematically expressed, in decibels, as the ratio of the peak of the side lobe to the peak of the main lobe [48].

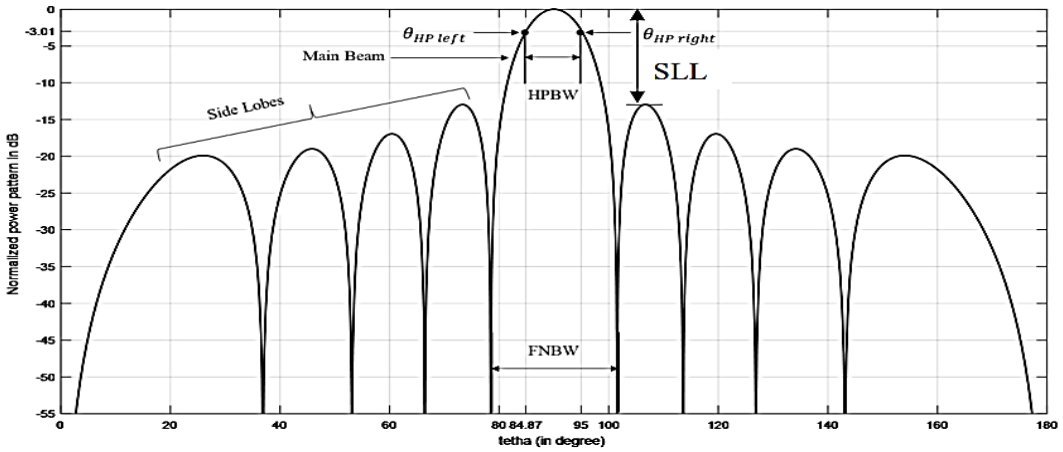


Figure (2.2). Lobes of the radiation pattern and beamwidth[48]

The SLL is an important parameter in antenna design because it determines the amount of power radiated in unwanted directions. A low SLL is desirable because it indicates that most of the energy is concentrated in the main lobe, which is the desired direction of radiation. On the other hand, a high SLL indicates that a significant amount of energy is radiated in

unwanted directions, which can cause interference and reduce the efficiency of the antenna. Fig (2.2). demonstrates the use of a graphical representation to calculate various pattern parameters, such as SLL and beamwidth, when measuring radiation pattern characteristics, SLL is given as follows [48]:

$$SLL_{dB} = 20 \times log_{10}\left(\frac{|E_{total}(\theta, \emptyset)|}{|E_{total}(\theta, \emptyset)_{max}|}\right) \quad (2.1)$$

It is possible to express the peak SLL (PSLL) [48]:

$$PSLL_{dB} = max\left(20 \times log_{10}\left(\frac{|E_{total}(\theta, \emptyset)|}{|E_{total}(\theta, \emptyset)_{max}|}\right)\right) \quad (2.2)$$

Where $E_{total}(\theta, \emptyset)$ is far field radiated field, $E_{total}(\theta, \emptyset)_{max}$ is maximum far field radiated field. $\emptyset$ is phase or longitude angle, and θ is azimuth angle.

To reduce the SLL, various techniques can be used in antenna design, these techniques aim to shape the radiation pattern to reduce the power in the side lobes and concentrate the energy in the main lobe [61].

2.2.2.3 Array Beamwidth

The beamwidth of an antenna array refers to the angular width of the primary lobe in the radiating pattern. It is a measure of the directivity or directional characteristics of the array, which is determined by the number of elements, their spacing, and the type of feeding network used to excite the elements [48].

The beamwidth of an array is to be specified as the angle between two points on the main lobe of the radiation pattern where the power is half of the maximum power. This angle is also known as the Half-Power Beam Width (HPBW) or -3 dB beamwidth. The smaller the beamwidth, the more directional the array and the higher gain in the main lobe. The beamwidth of an array can be calculated using the following formula [48]:

$$HPBW = 2 \times Arc\sin\left(\lambda/_{2d}\right) \quad (2.3)$$

Where $\lambda = f/c$ is the wavelength of the signal and d is the spacing between the elements in the array. This formula assumes a Uniform Linear Array (ULA) of isotropic elements with equal amplitudes and phases.

In practice, the beamwidth of an array can be controlled by adjusting the element spacing and the feeding network. Increasing The number of elements in an array can also reduce the beamwidth, but it comes at the cost of increased complexity, size, and cost. Beamwidth is an important parameter in antenna design, as it determines the coverage area and the directionality of the antenna.

2.2.2.4 Directivity

Directivity is a measure of the concentration of energy in a particular direction or other words the antenna's potential to direct radioactive power in a particular direction where antenna arrays are used to improve guidance well. It is a measure of the antenna's ability to convert electrical power into radiated power in a particular direction [62].

The direction can be expressed mathematically through the radioactive force of the antenna array in a particular direction compared to the radioactive force by an isotropic radiator in the same direction [62]:

$$D(\theta) = 4\pi \frac{|E_{total}(\theta,\emptyset)|^2}{\int_0^{2\pi}\int_0^{\pi}|E_{total}(\theta,\emptyset)|^2 \sin\theta \, d\theta \, d\emptyset} \quad (2.4)$$

Where $D(\theta)$ the directivity of array.

The directivity of an antenna can be increased by increasing the physical size of the antenna, increasing the operating frequency, or using directional antenna designs such as Yagi-Uda, patch antennas, or horn antennas.

Directivity is an important parameter in antenna design, as it determines the efficiency of the antenna in converting electrical power into radiated power in a specific direction [62].

2.2.3 Types of Antenna Array

Antenna arrays are a set of multiple antennas arranged in a specific pattern to enhance the performance of a wireless communication system [2].

2.2.3.1 Linear Antenna Array

LAA can create a directional beam pattern that is focused in a particular direction, and the direction of the beam can be controlled by changing the relative phase of the signals fed to each antenna or changing the position of elements [63].

2.2.3.1.1 Uniform Linear Array (ULA)

In this type of array, the currents are of equal excited and the same amplitude and evolution along the axis of the array where the elements of this array are regularly distributed that is, the distances are equal between them and each element contains the same excitement and current at the phase and the value of the spacing of the elements can change according to the design. As shown in Fig (2.3). [61].

2.2.3.1.2 Non-Uniform Linear Array

In a non-uniform linear array, the antennas are spaced non-uniformly along the line. This type of array is used to improve the performance of the ULA by reducing the sidelobe levels and improving the beam pattern [48].

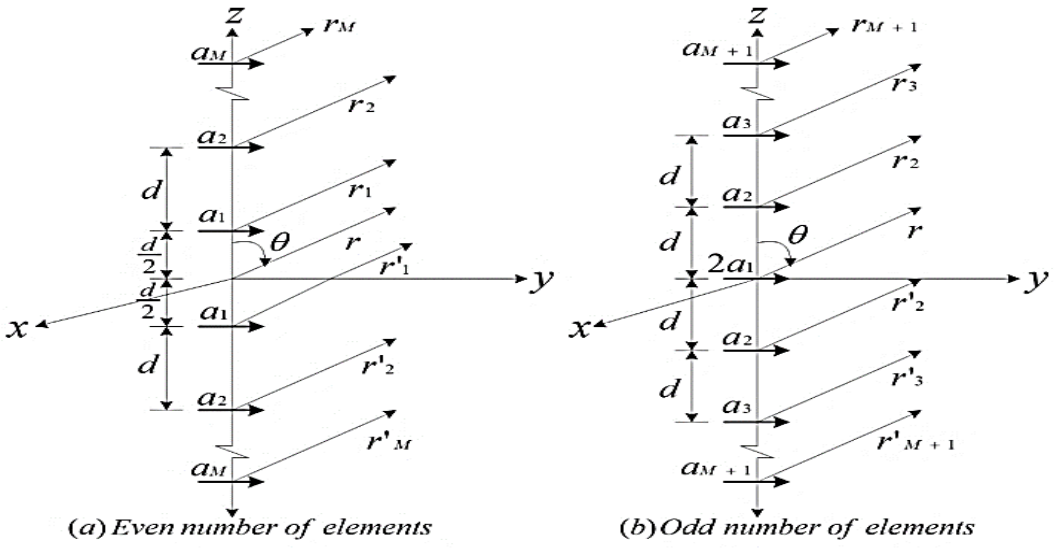


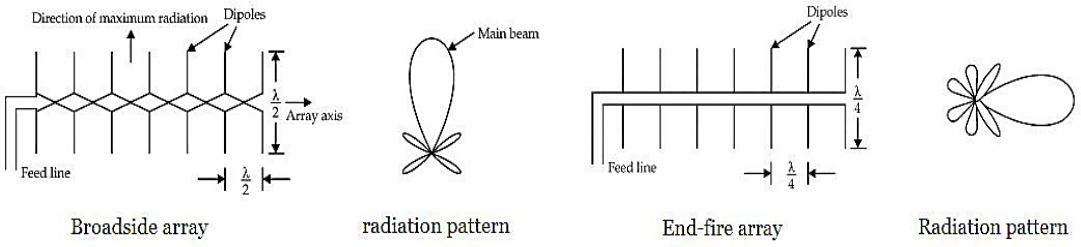
Figure (2.3). A uniform linear array [61]

2.2.3.1.3 Broadside Array

In a broadside array, the antennas are oriented perpendicular to the line of antennas. This type of array is used to create a beam pattern that is perpendicular to the line of antennas. See Fig (2.4). (a).

2.2.3.1.4 End-fire Array

In an end-fire array, the antennas are oriented along the line of antennas. This type of array is used to create a beam pattern that is directed along the line of antennas. See Fig (2.4). (b). [48].



(a) **(b)**

Figure (2.4). (a) Broadside array, (b) End-fire array [48]

2.2.4 Geometric Models and Array Factor

The antenna system includes many antenna elements that are arranged according to various geometric elements in terms of shape, such as Linear Antenna Array (LAA), Circular Antenna Array (CAA), Random Antenna Array (RAA), etc. All of which aim to get the resulting beam pattern to conform to the desired beam pattern. If it does not comply with it, optimization algorithms that help to adjust its antenna array weights will be used to obtain the required beam pattern [64].

2.2.4.1 Mathematical Model of Linear Antenna Array

A geometric model of LAA consists of the physical dimensions and positions of the individual antenna elements. The spacing between the elements, as well as their lengths and orientations, can affect the radiation pattern of the array. For example, increasing the spacing between elements can lead to a narrower main lobe and lower side lobe levels in the radiation pattern [64]. By using a mathematical function to represent the Array Factor (AF) of a LAA that takes into account the spacing and phase differences between the elements, AF can be calculated using various techniques, when using LAA of 2N isotropic elements placed symmetrically along the x-axis the expression for the AF of LAA is [64]:

$$AF_{LAA}(\theta, \varphi_n) = 2 \sum_{n=1}^{N_{LAA}} I_n \ \ cos(kx_n \ \ cos(\theta) + \varphi_n \ ) \qquad (2.5)$$

Where I_n is normalized to its maximal value of excitation current of the antenna element, x_n the position of n^{th} element in the array, $k = 2\pi/\lambda$ is the wave number. φ_n the phase of the n^{th} antenna element, assuming $\varphi_n = 0$, and $I_n = 1$ hence the relationship of AF as follows [64]:

$$AF_{LAA}(\theta) = 2\sum_{n=1}^{N_{LAA}} cos(k\, x_n\, cos(\theta)) \quad (2.6)$$

By determining the fitness function by selecting the solution for LAA elements, it is possible to enhance the beam pattern and reduce the SLL, which appears in the following format [64]:

$$Fit_{LAA} = min\,(max(20 \times log_{10}|AF_{LAA}(\theta)|)) \quad (2.7)$$

2.3 A Reduction Algorithms

Many algorithms have been used to suppress SLL as far as possible and concentrate power in the main lobe instead of sending it in an undesirable direction and these algorithms are described below.

2.3.1 Particles Swarm Optimization

The PSO method is an intelligent technique that leverages swarm intelligence and cooperative behavior among particles. Each particle inside the swarm represents a potential solution across huge, multidimensional search spaces, which depend on the individual's social behavior such as bee swarm, fish, etc. [65]. This algorithm is an evolutionary algorithm that makes the search area very broad to reach the best solution in addition to artificial life. It is concerned with the characteristics of the applicable systems. By using a simulation approach. For instance, each swarm can be represented as a discrete point in a random position [66].

The PSO algorithm has been successfully applied in the design of antennas. This algorithm turns out to be very effective in getting the optimal solution and getting rid of the problems of optimization and is often similar to the GA. It needs physical fitness to get rid of the problems of optimization

where each particle moves randomly in space and each takes the optimal place of it, which is called the best globally [67,68].

Following a series of tests, the algorithm underwent repairs using a relative simulation so that the swarm could find the optimal point quickly. Inspired by this model, this relationship is represented by the following equation [67]:

$$v_x = v_x + 2 \times rand \times (p_{best}x - x) + 2 \times rand \times (g_{best}x - x) \quad (2.8)$$

$$x = x + v_x \quad (2.9)$$

Where p_{best} is personal best value of particles, and g_{best} is group best solution of the group.

The abstracted each individual to be a particle without mass and volume, with only velocity and position, so they called this algorithm PSO. In each generation, the particle information is combined to adjust the velocity of each dimension, which is used to compute the new position of the particle. particles change their states constantly in the multi-dimensional search space until they reach a balanced or optimal state.

The coordinates of this algorithm can be represented mathematically in continuous space, with the that N is the size of the swarm in space D-dimension is $X_i = (x_{i_1}, x_{i_2}, \dots, x_{i_d}, \dots, x_{i_D})$, individual optimal position $P_i = (p_{i_1}, p_{i_2}, \dots, p_{i_d}, \dots, p_{i_D})$, Vector of velocity is $V_i = (v_{i_1}, v_{i_2}, \dots, v_{i_d}, \dots, v_{i_D})$, position optimal for swarm $P_g = (p_{g_1}, p_{g_2}, \dots, p_{g_d}, \dots, p_{g_D})$. taking the minimizing problem as an illustration, the initial version of the PSO algorithms update formula for the individual optimal position is as follows [67]:

$$P_{i,t+1}^d = \begin{cases} X_{i,t+1}^d & if\ f(X_{i,t+1}) < f(P_{i,t}) \\ P_{i,t}^d & otherwise \end{cases} \quad (2.10)$$

Where $P_{i,t+1}^{d}$ Individual Optimal position in d-dimension at initial position *i* to variable *t+1*, $X_{i,t+1}^{d}$ location of swarm size in d-dimension in initial location *i* to variable *t+1* (position after update), $P_{i,t}^{d}$ Individual Optimal position in d-dimension at initial position *i* to variable *t*.

The optimal position of all individuals is the optimal location of the swarm. The updated velocity and position formulas are as follows [67]:

$$V_{i,t+1}^{d} = V_{i,t}^{d} + C_1 \times rand \times \left(P_{i,t}^{d} - X_{i,t}^{d}\right) + C_2 \times rand \times \left(P_{g,t}^{d} - X_{g,t}^{d}\right) \quad (2.11)$$

$$X_{i,t+1}^{d} = X_{i,t}^{d} + V_{i,t+1}^{d} \quad (2.12)$$

Since the initial version of PSO was not very effective in the optimization problem, a modified PSO algorithm appeared soon after the initial algorithm the velocity update formula was altered by the addition of inertia weight, and the new velocity update formula became [67]:

$$V_{i,t+1}^{d} = w \times V_{i,t}^{d} + C_1 \times rand \times \left(P_{i,t}^{d} - X_{i,t}^{d}\right) + C_2 \times rand \times \left(P_{g,t}^{d} - X_{g,t}^{d}\right) \quad (2.13)$$

Where $V_{i,t}^{d}$ is individual Velocity in d-dimension at initial velocity i to variable t (Velocity before update), $V_{i,t+1}^{d}$ is the velocity of swarm size in d-dimension in initial Velocity *i* to variable *t+1* (Velocity after update), *w* Inertia wight, C_1& C_2 are the position weighting factors.

Although this modified algorithm has almost the same complexity as the initial version, it has greatly improved the algorithm performance; therefore, it has achieved extensive applications.

Fig (2.5). depicts the iteration procedure of any particle for each generation. Sociological analysis of the velocity update formula reveals that the first section of this updated formula is the impact of the particle's previous velocity. It indicates that the particle is confident in its present state of motion and conducts inertial motion in accordance with its velocity. The

parameter w is known as the inertia weight, $C_1 \& C_2$ are the position weighting factors, and the rand is a random number between [0,1].

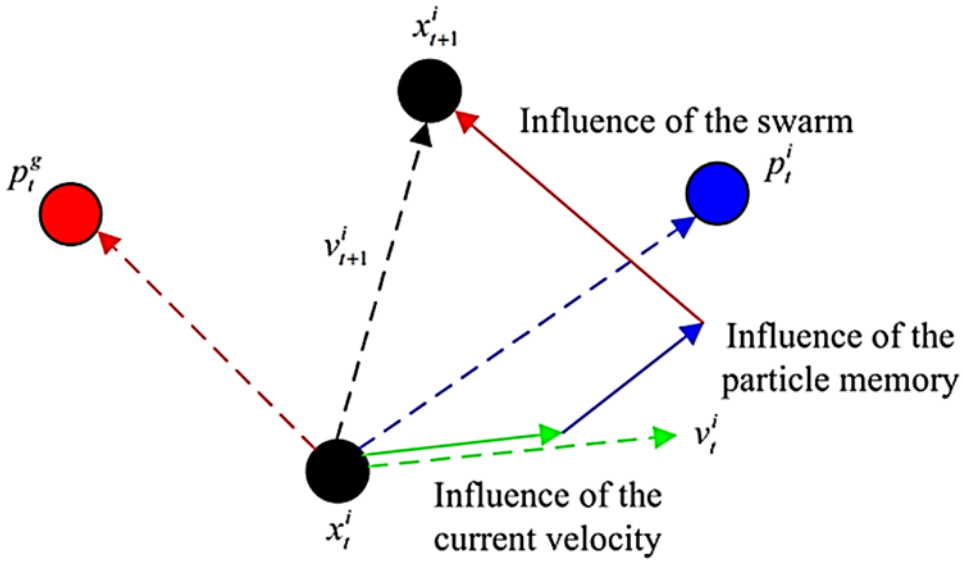


Figure (2.5). Iteration scheme of the particles [69].

The second section depends on the distance between the particle's current position and its optimal position, called the "cognitive" item. It means a particle thinking, i.e., a particle movement resulting from its own experience. Therefore, parameter C_1 is called the cognitive learning factor (also called the cognitive acceleration factor). The third section relies on the "social" factor depends on the distance between the particle's current position and the optimal global (or local) position in the swarm. It refers to the cooperation and sharing of information among the particles, specifically the movement of a particle as a result of the experiences of other particles in the swarm. The parameter C_2 is known as the social learning factor (also known as the social acceleration factor) because it stimulates the movement of a positive particle through cognition [69,70].

2.3.2 Genetics Algorithm

A Genetic Algorithm (GA) is one of the family of evolutionary algorithms that simulate the search process as an evolution through natural

selection. GA is favored over other optimization techniques because it searches a larger space and identifies the global minimum, whereas the simplex method terminates upon attaining the local minimum. GA is a prospective global optimization technique for the design of antenna arrays, among other optimization techniques [71].

GA employs techniques inspired by biology, including genetic inheritance, natural selection, mutation, and sexual reproduction (recombination, or crossover). The population is the name of GA data structure for individuals. Chromosomes are also known as individuals. Typically, each individual is depicted by binary strings. Each individual signifies a point in the search space and a candidate solution.

Steps in the GA:

Fig. (2.6) illustrates the four fundamental phases of the GA [72]:

1. **Initialization:** The spherical coordinates of a location are constructed using three values: azimuth θ, longitude $\emptyset$, and radial distance R. by the equation provided in the following manner [72]:

$$X = R \times sin\theta \times cos\emptyset \quad (2.14)$$

$$Y = R \times sin\theta \times sin\emptyset \quad (2.15)$$

$$Z = R \times cos\theta \quad (2.16)$$

Using discrete two-dimensional Fourier transform, the chromosomal parameters are X, Y, and Z. In reference [72], through the existence of individuals applied to them for enhancement and the array composition, The evolutionary algorithm was used to extend the antennas' range by increasing their frequency and decreasing their SLL. The initial phase of the GA is the initiation, which constructs chromosomes at random between

the higher and lower points of longitude and latitude to accomplish an appropriate annexation.

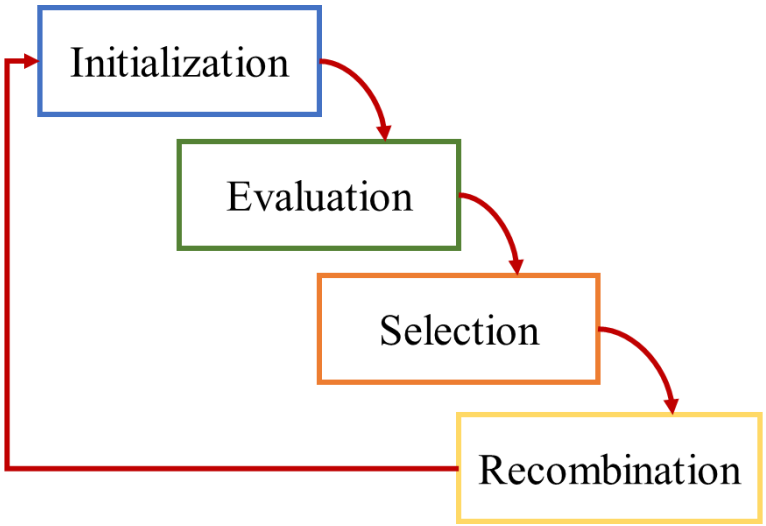


Figure (2.6). Steps in the genetic algorithm.

2. **Evaluation:** Upon initiation of implementation, it evaluates the functional fitness that is intended to expand the frequency field coverage and bring it to the level required, where the chromosome is calculated to arrange solutions as complete.
3. **Selection and Recombination:** Selection is arguably the most crucial stage, as it strengthens the control of the chosen chromosomes for the next generation to converge the algorithm. The selection of chromosomes for the succeeding generation is determined by the mutation and crossover probability.

   3.1. **Crossover:** The crossover operator generates a new chromosome by arbitrarily severing two chromosomes at one or more sites (X, Y, or Z). One or multiple crossover points are conceivable. Crossover does not result in the production of new material within a population, but it does result in the creation of new chromosomes by combining two existing ones, which enhances the average fitness of the next generation. Two types of crossovers may be distinguished: single-point crossovers, which involve the

separation of chromosomes at a single position, and multi-point crossovers, which involve the separation of chromosomes at numerous positions [73].

3.2. Mutation: The algorithm for evolution modifies the chromosome gene at random. Mutation may modify X, Y, or Z, or possibly more than one, to enhance the size of the optimal solution The mutation increases the scale of the solution space by adding new chromosomes to the population the solution space is expanded by introducing more chromosomes into the population. Using random mutations to a gene on a chromosome, it identifies new potential optimization sites [73].

The fitness of newly generated individuals is evaluated, and the fittest individuals survive for the next generation. Genes from virtuous individuals spread throughout the population, making subsequent generations more adapted to their environment.

2.3.3 Flower Pollination Algorithm

FPA is commonly used to optimize multi-objective design problems. The fundamental tenets of the FPA are grounded upon a set of four core rules [74]:

i. Biotic and cross-pollination may be seen as mechanisms contributing to the phenomenon of global pollination, wherein pollinators carrying pollen engage in Levy flights along their migratory patterns.
ii. Local pollination utilizes abiotic pollination and self-pollination.
iii. Insects, as pollinators, have the ability to promote bloom constancy, this phenomenon may be described as a reproductive likelihood that exhibits proportionality to the degree of resemblance between the two flowers in question.

iv. The regulation of the interaction or transition between local pollination and global pollination is governed by a probability associated with a switch mechanism $p \in [0,1]$.

FPA comprises core parameters of Levy-flights based step size $L(\beta)$, β is flower attraction rate, population size n , scaling factor γ, Switching Probability p , and $\varepsilon \in [0,1]$ is a uniform distribution is commonly employed in local inoculation [32].

The execution of the FPA starts by establishing the goal function and randomly initializing a population of flowers, denoted as n .The optimal solution from the initial population is determined, and defining the probability of switching $p \in [0, 1]$ it governs the choice between local or global pollination. The selection between global and local pollination is decided using a stochastic process including a random number generator. If this random number is less than the probability of switching p , then global pollination is conducted using the Eq. (2.17). Otherwise, local pollination is accomplished using the Eq. (2.21).

The mathematical expression according to represents global pollination rule (i) and flower constancy rule (iii) [32]:

$$x_i^{t+1} = x_i^t + \gamma L(g_{best} - x_i^t) \qquad (2.17)$$

Where x_i^t the solution vector x_i at iteration t, γ is a scaling factor to control step size, and g_{best} the current best solution. L The numerical value shown denotes the step size based on Levy flights, it is an estimate of the pollination intensity which corresponds to the strength of the pollination The use of a Levy flight, which L is selected from a Levy distribution, is an effective approach to represent the property of insects being able to fly great distances with varied distance increments. Utilizing the provided mathematical equation [32]:

$$L \sim \frac{\beta\Gamma(\beta)\sin\left(\frac{\pi\beta}{2}\right)}{\pi} \frac{1}{s^{1+\beta}} \quad (2.18)$$

Where $\Gamma(\beta)$ the conventional gamma function. The proposed a quick and precise algorithm for generating a stochastic variable whose probability density is close to that of a Levy stable distribution [75]. This algorithm generates in a single process the requisite Levy stable stochastic process. The determination of the pseudo-random step size s, which adheres to the Levy distribution, is performed for two Gaussian distributions U and V, as seen in the equation provided below [32]:

$$s = \frac{U}{|V^{1/\beta}|} \quad (2.19)$$

The values are taken from a Gaussian distribution that is normal with a zero mean and variance σ^2, which is given by the formula below [32]:

$$\sigma^2 = \left[\frac{\Gamma(1+\beta)}{\beta\,\Gamma(\frac{(1+\beta)}{2})} \cdot \frac{\sin(\frac{\pi\beta}{2})}{2^{(\beta-1)/2}}\right] \quad (2.20)$$

The following mathematical formulation is used for local pollination [32].

$$x_i^{t+1} = x_i^t + \varepsilon\left(x_j^t - x_k^t\right) \quad (2.21)$$

where x_k^t and x_j^t are pollen from several blooms belonging to the same kind of plant. If x_j^t and x_k^t are chosen from the same population, this is comparable to a local random walk as ε comes from a uniform distribution in [0,1].

2.3.4 Grey Wolf Optimization

The normal size of a grey wolf pack ranges from five to twelve individuals on average. The wolf pack has a well-defined hierarchical and employs collaborative strategies to pursue and capture prey, drawing

inspiration from a social organization and collective hunting behavior. This section provides a comprehensive analysis of the motivation for the GWO algorithm and its underlying mathematical model [14].

1. Inspiration of the GWO

The social structure of a grey wolf family is one of the most intriguing phenomena ever observed. All of the members are required to adhere to a very strict social hierarchy. As shown in Fig (2.7), the herd consists of four distinct types of wolves: alpha, beta, delta, and omega. A pair of dominant alphas serve as the pack's leaders. The alphas are in charge of directing the flock in foraging, feeding, migration, and other social activities. The second level of the grey wolf social hierarchy is known as the beta. Beta wolves are most likely to supplant alphas if an alpha becomes injured, ill, elderly, or dies. They facilitate the decision-making process for the alpha members and guide the subordinate wolves. The Delta wolf pack is on the third level, Deltas consist of scouts, sentinels, elders, hunters, and custodians. Omega wolves are the lowest-ranking wolves; they must submit to other dominant wolves. The omegas serve as the scapegoat.

Each wolf in the pack has a distinct role and responsibility. This variety of social hierarchies has been discovered to contribute to the cohesion and social order of the entire container. Additionally, it helps reduce conflicts and reduces the likelihood of aggressive behavior occurring among group members [14].

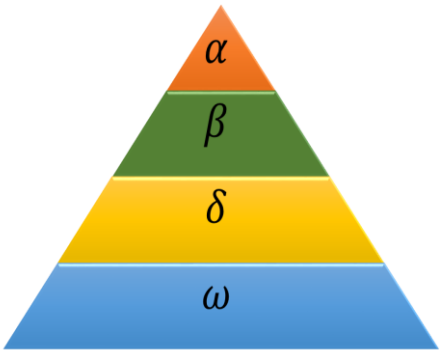


Figure (2.7). Nature depiction of the social hierarchy of grey wolves.

2. GWO Algorithm

- Social hierarchy

To mathematically characterize the social structure of grey wolves, the alpha α is presumed to be the finest solution discovered to date. The second and third-best solutions are then defined as the beta value β, delta δ, and the remaining candidate solutions are subsequently regarded to be omegas ω.

- Group hunting

The process encompasses other stages, including the encirclement of prey, the act of hunting, the attachment of prey (exploitation), and the tracking of prey (exploration).

As previously said, the primary stage of a hunt involves the identification and encircling of prey. The following pair of equations are used to quantitatively describe the phenomenon of encircling behavior [14]:

$$\vec{D} = \left|\overrightarrow{C} . \overrightarrow{X_p(t)} - \overrightarrow{X(t)}\right| \quad (2.22)$$

$$\overrightarrow{X}(t+1) = \overrightarrow{X_p}(t) - \overrightarrow{A} . \left(\vec{D}\right), \quad (2.23)$$

Where $\vec{A}$ and $\vec{C}$ are vectors of coefficients, $\vec{X}$ represents the vector position of a grey wolf, t denotes the current iterations, and $\overrightarrow{X_p}$ is the vector representing the position of the prey. Calculating the vectors $\overrightarrow{A}$ and $\overrightarrow{C}$ using [14]:

$$\overrightarrow{A} = 2\,\vec{a} . \overrightarrow{r_1} - \vec{a} \quad (2.24)$$

$$\overrightarrow{C} = 2 . \overrightarrow{r_2} \quad (2.25)$$

Where components of $\overrightarrow{r_1}, \overrightarrow{r_2}$ are random vectors in [0, 1] and $\vec{a}$ is called encircling coefficient vector iteratively decreases linearly from 2 to 0 as the following [14]:

$$\vec{a} = 2 . \left(1 - \frac{t}{T}\right) \quad (2.26)$$

Where t represents the current iteration and T represents the utmost number of iterations.

To ensure their survival within the perilous and fiercely competitive natural habitat, wolves establish collaborative social structures known as packs, which facilitate their hunting endeavors. The members of a pack typically hunt together, with the alpha typically taking the lead. The alpha can choose which prospective prey to pursue, or if the hunt is not going well, he can abandon it. On occasion, the beta and delta may also participate in the search [33].

In general, the alpha, beta, and delta are believed to have the most hunting experience. Additionally, they have a greater understanding of the prey's prospective location. Other wolves, specifically the omegas, follow the alpha, beta, and delta. Consequently, this collective foraging behavior is formulated by [14]:

$$\overrightarrow{D_\alpha} = \left|\overrightarrow{C_1}.\ \overrightarrow{X_\alpha} - \vec{X}\right|,$$

$$\overrightarrow{D_\beta} = \left|\overrightarrow{C_2}.\ \overrightarrow{X_\beta} - \vec{X}\right|,$$

$$\overrightarrow{D_\delta} = \left|\overrightarrow{C_3}.\ \overrightarrow{X_\delta} - \vec{X}\right|,$$

$$\overrightarrow{X_1} = \overrightarrow{X_\alpha} - \overrightarrow{A_1}.\left(\overrightarrow{D_\alpha}\right),$$

$$\overrightarrow{X_2} = \overrightarrow{X_\beta} - \overrightarrow{A_2}.\left(\overrightarrow{D_\beta}\right),$$

$$\overrightarrow{X_3} = \overrightarrow{X_\delta} - \overrightarrow{A_3}.\left(\overrightarrow{D_\delta}\right),$$

$$\overrightarrow{X}(t+1) = \frac{\overrightarrow{X_1}+\overrightarrow{X_2}+\overrightarrow{X_3}}{3}, \qquad (2.27)$$

Where $X(t+1)$ represents the updated position of an omega wolf in the $(t+1)^{th}$ iteration, as determined by alpha, beta, and delta in the t^{th}

iteration,$X_\alpha, X_\beta, X_\delta$, and X indicate the coordinates of alpha, beta, delta, and omega wolf.

Fig (2.8). showed elucidates how an omega wolf adjusts its spatial coordinates inside a two-dimensional plane, under the influence of the alpha, beta, and delta wolves. The collaborative behavior of alpha, beta, and delta wolves in estimating the prey's location has been seen, followed by the subsequent adjustment of the omega wolf's position [14].

The grey wolves may launch a coordinated attack on the prey after encircling it. This is achieved by manipulating $\overrightarrow{A}$ in Eq. (2.24). It is noted that the value depends on the vector of the encircling coefficient vector $\vec{a}$, which decreases linearly from 0 to 2, so the value of $\vec{A}$ ranges from $[a, -a]$.

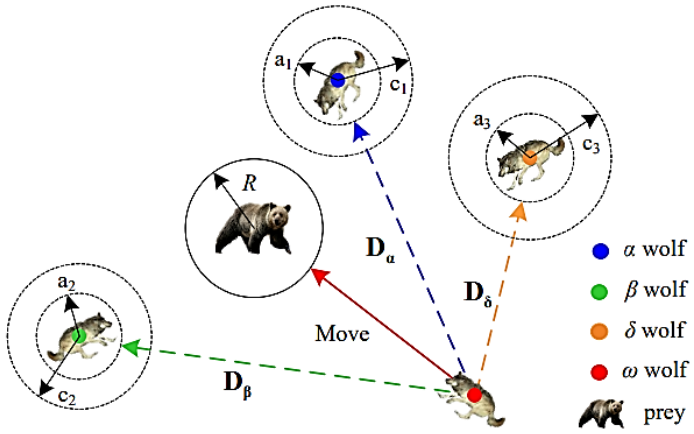


Figure (2.8). Illustration of grey wolves hunting behavior [14].

2.3.5 Sparrow Search Algorithm

The sparrow is a smart living being and always keeps its distance from others to stay safe and secure. But when there is a danger, they disperse and suffer from loneliness, and certainly according to their code of conduct, they are divided into three sections: producers, scroungers, and scouters. Assuming that there are N sparrows in a D-dimensional search space, the

position of the i^{th} sparrow can be expressed as $X_i = [X_{i,1}, X_{i,2}, X_{i,3}, \dots, X_{i,D}]$, $i = 1,2\dots, N$.

Sparrows foraging in groups can obtain food by searching or through social interaction with other group members [76]. where producers actively seek food and scroungers benefit from producers' efforts by joining or stealing. As depicted in Fig (2.9), feeding trials were designed to determine whether certain individuals favor producer or scrounger roles within a foraging community, and whether dominance rank or relatedness influences joining behavior [77].

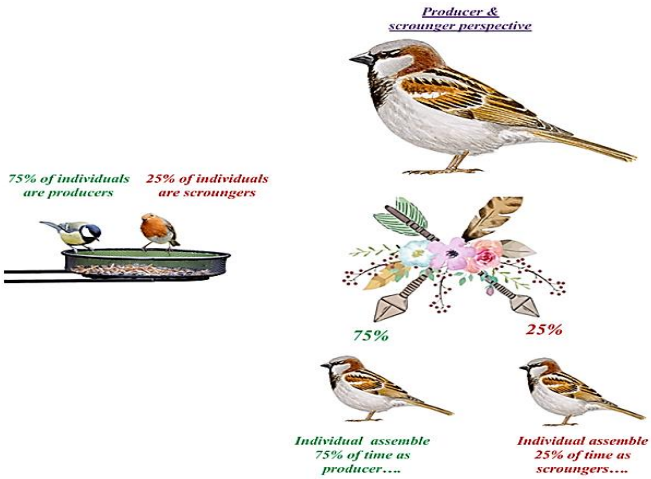


Figure (2.9). Role of producers and scroungers [77].

Producers The productive sparrow has a very extensive search logic, a very high fitness relative to other species, and is in charge of finding and supplying food for other species. They can be mathematically expressed as follows [78]:

$$X_{i,j}^{t+1} = \begin{cases} X_{i,j}^{t}.\, e^{\frac{-i}{\alpha.T}}, & R_2 < ST \\ X_{i,j}^{t} + Q.L & R_2 \geq ST \end{cases} \quad (2.28)$$

Where t and T represent the current iteration number and a maximum number of iterations resistively. α a random integer that is uniform and falls

between [0,1]. The i^{th} sparrows location in the j^{th} dimension is indicated by the notation $X_{i,j}$, and Q is a random number.

Where $R_2 \in [0,1]$ represent the alarm value and $ST \in [0,1]$ the safety threshold. When $R_2 < ST$ this means that the area is safe and producers can search for food within this area. When $R_2 \geq ST$ this means that the area is dangerous and sparrows must fly to other places to keep them safe.

Scroungers are sparrows; keep a watch on producers if they discover that the producers have discovered superior food, they will promptly abandon their current position to compete for food and become the producers themselves. The position of the Scroungers is expressed mathematically as [78]:

$$X_{i,j}^{t+1} = \begin{cases} Q.\exp\left(\dfrac{X_{worst} - X_{i,j}^{t}}{i^2}\right), & i > \dfrac{N}{2} \\ X_p^{t+1} + \left|X_{i,j}^{t} - X_p^{t+1}\right|.A^{+}.L, & i \leq \dfrac{N}{2} \end{cases} \tag{2.29}$$

Where X_{worst} and X_p represent the global worst position and best position respectively, and $A^{+} = (A\,A^{T})^{-1}$. If $N/2 < i$ it comes out that these sparrows with poor fitness require a larger area to obtain food. When $N/2 \geq i$, It turns out that these scroungers conduct local searches while being situated in a random location near the optimal location.

The Scouters are generated randomly between the producers and the scroungers, and they can perceive whether there is risk in the foraging area. The scouter model can be expressed as follows [78]:

$$X_{i,j}^{t+1} = \begin{cases} X_{best}^{t} + \beta.\left|X_{i,j}^{t} - X_{best}^{t}\right|, & F_i > F_g \\ X_{i,j}^{t} + k.\left(\dfrac{\left|X_{i.j}^{t} - X_{worst}^{t}\right|}{(F_i - F_\omega) + \varepsilon}\right), & F_i = F_g \end{cases} \tag{2.30}$$

Where X_{best} represents the global optimal position, β is a random number,$k \in [-1, 1]$ indicates the direction of movement of the sparrows. F_i represent the fitness value of the i^{th} sparrow. F_g the global optimal, F_ω the worst fitness value of the current population. when $F_g < F_i$ This indicates that the sparrow has to leave the risky region and go somewhere safer. when $F_g = F_i$ indicates that the sparrow is in the middle of the population, but it is aware of the danger and needs to be close to other sparrows to reduce the risk of predation [78].

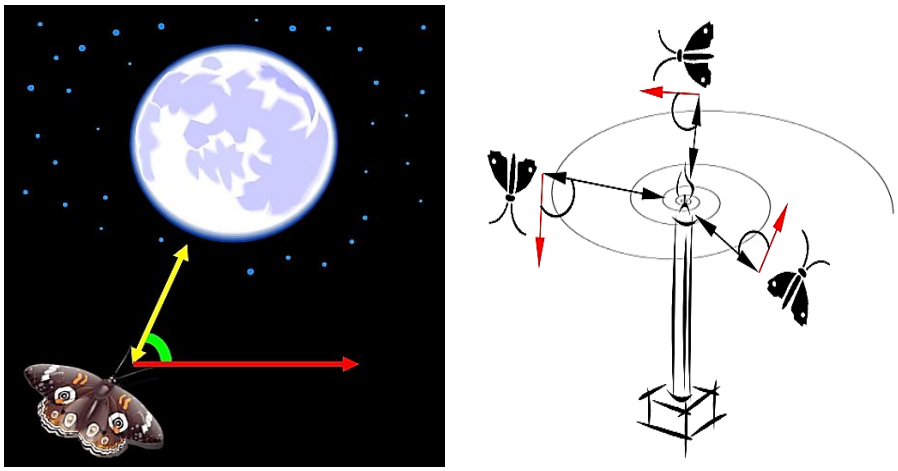
2.3.6 Moth Flame Optimization

Moths are aesthetically pleasing insects that have a striking resemblance to the butterfly family. In essence, the wild is inhabited by a staggering number of over 160,000 species of this particular insect. The life cycle of these organisms consists of two primary phases, namely larvae and adults. The metamorphosis of larvae into moths occurs via the process of cocoon formation. One of the most intriguing aspects of moths lies in their nocturnal navigation strategies. These organisms have developed the ability to harness moonlight to facilitate nocturnal flight [79].

The navigational system used was referred to as transverse orientation. This approach involves the utilization of a consistent angle for the moon by a moth, which serves as a very efficient mechanism for traversing extensive distances along a linear trajectory. As shown in Fig (2.10). (a). [80].

Although transverse orientation is known to be helpful, moths often exhibit a tendency to fly in a spiral pattern around light source. Indeed, similar behaviors are elicited by the presence of artificial illumination in moths. The limited effectiveness of the transverse orientation necessitates a significant distance between the light source and the observer to enable transit along a linear trajectory. When moths see a man-made artificial light

source, they endeavor to maintain a consistent orientation to navigate along a linear trajectory but the moths exhibit a spiral flight pattern, as seen in Fig (2.10). (b). due to their tendency to maintain a consistent angle with the nearby light source, which is in close proximity to the moon [81].



(a) **(b)**

Figure (2.10). (a)Transverse orientation. (b) . Spiral flying path around close light sources [80].

The MFO method assumes that the candidate solutions may be represented as moths, with the problem variables corresponding to the spatial locations of these moths. By altering their position vectors, the moths are capable of navigating across several dimensions, including one-dimensional, two-dimensional, three-dimensional, and hyperdimensional spaces. Due to its population-based nature, the MFO algorithm can identify optimum solutions for many problem domains. Both moths and flames may be considered as potential answers. The differentiation is contingent upon our approach and modification of them in every successive repetition. Moths serve as authentic search agents that navigate across the search space, while flames symbolize their most significant position achieved so far. Alternatively, embers might be seen as indicators or signals that moth emit while exploring the search area. Consequently, each moth seeks around a flag (flame) and upgrades it if it discovers a superior solution. This

mechanism ensures that a moth never loses its optimal solution. logarithmic spiral serves as the primary update mechanism for moths. Nonetheless, any form of the spiral may be utilized here subject to the conditions outlined below: The initial point of the spiral should begin at the insect. The position of the flame should be the concluding location of the spiral. The fluctuation of the spiral's range should not exceed the search space. Considering these factors, we define the MFO algorithm's logarithmic spiral as follows [12]:

$$M_i = S(M_i, F_j) \tag{2.31}$$

Where F_j the j^{th} flam, M_i is the i^{th} moth and S the logarithmic spiral function. The following defines the logarithmic spiral function for the MFO algorithm [12]:

$$S(M_i, F_j) = D_i . e^{bt} . \cos(2\pi t) + F_j \tag{2.32}$$

Where b is a constant that defines the form of the logarithmic spiral, t is a random number in [-1,1], and D_i reveals the distance from the i^{th} moth for the j^{th} flam. the following relationship can be expressed [12]:

$$D_i = |F_j - M_i| \tag{2.33}$$

The next position of a moth concerning a flame is defined. The *t* parameter in the spiral equation specifies how close the next position of the moth should be to the flame (t = -1 is the closest, and t = 1 is the farthest). The fundamental element of the suggested methodology is the spiral movement, as it governs the spatial adjustments made by moths in proximity to flames. The spiral equation enables a moth to navigate in a round path around a flame, rather than only in the region between the moth and the flame. Hence, the assurance of exploration and exploitation inside the search space is established [12]. As shown in Fig (2.11).

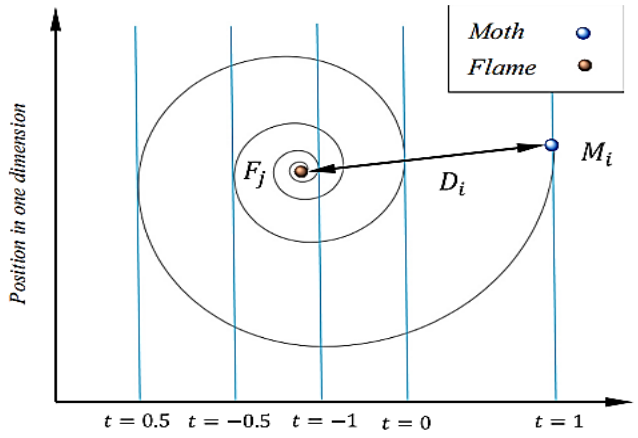


Figure (2.11). Logarithmic spiral, space around a flame, and the position concerning t [12].

The proposed position-updating procedure can ensure the safety of exploitation near the flames, to find the best solutions to improve the possibilities as we consider that the best solutions are reached through flam. Therefore, the matrix F in the above-mentioned equation comprises the n most recent optimal solutions acquired so far. During optimization, the caterpillars must alter their coordinates relative to this matrix. To underscore exploitation further, we presume that t is a random number value in the interval $[r, 1]$, where r decreases linearly from -1 to -2 over the course of an iteration. We refer to r as the convergence constant. With this procedure, moths tend to exploit their respective flames with a degree of precision proportional to the number of iterations [12].

Nonetheless, artificial moths continue to perform transverse orientation. To prevent local optimum stagnation, each moth has its flame. Because moths can only fly toward a flame and not outwards, if they are all attracted to a single flame, they will all converge on a single point in the search spaces. However, requiring them to move around different flames increases exploration of the search space and decreases the likelihood of local optima stagnation [82].

This method guarantees the exploration of the search space surrounding the finest locations found to date for the following reasons: Moths revise their positions on the most optimal solutions found to date. The sequence of flames is modified based on the optimal solution in each iteration, and the moths must adjust their positions about the modified flames. Therefore, the position updating of moths may occur around distinct flames, a mechanism that causes abrupt moth movement in search space and encourages exploration.

the position updating of moths relative to n distinct locations in the search space may hinder the exploitation of the most promising solutions. A proposed adaptive mechanism for the number of flames addresses:

$$Flame\ number = round\left(N - t \times \frac{N-1}{T}\right) \quad (2.34)$$

Where N is the maximum number of flames, t is the current number of iterations, and T is specifies the maximum number of possible iterations.

At the start of the iteration, there exists a quantity of N flames. During the last phases of the iterations, the moths alone adjust their positions relative to the most luminous flame. The ongoing decline in the quantity of flames serves to maintain a delicate equilibrium between the processes of exploration and exploitation within the search space [82].

2.3.7 Multi-Verse Optimization

The inception of the cosmos is postulated by the Big Bang hypothesis, the term multi-verse refers to the existence of other universes in addition to the universe in which we all reside [83].

In accordance with the multiverse theory, multiple universes interact and may even intersect. The multiverse theory also implies that various physical

principles may exist in separate universes. The MVO algorithm was derived from three fundamental principles of multiverse optimization, namely white holes, black holes, and wormholes. The existence of a white hole inside our universe has yet to be empirically seen. However, within the realm of theoretical physics, there is a belief that the phenomenon of the big bang may be conceptualized as a white hole. It is postulated that the Big Bang may have acted as the major catalyst for the inception of the universe [84].

Furthermore, according to the cyclic model of multi-verse theory, it is posited that the occurrence of Big Bang white holes is a consequence of the collision between parallel universes. Black holes, which have been frequently observed, operate entirely differently than white holes. Their exceptionally powerful gravitational force attracts everything, including light beams [85].

Fig (2.12). depicts conceptual representations of these three essential components of the multiverse theory. Every universe has a rate of inflation (eternal inflation) which causes its expansion in space. In terms of the formation of stars, planets, asteroids, black holes, white holes, wormholes, physical laws, and suitability for life, the universe's inflation rate is crucial. According to one of the cyclic multiverse models, multiple universes interact via white, black, and wormholes to attain a stable state. This is the identical inspiration for the conceptual and mathematical MVO algorithm [86,87].

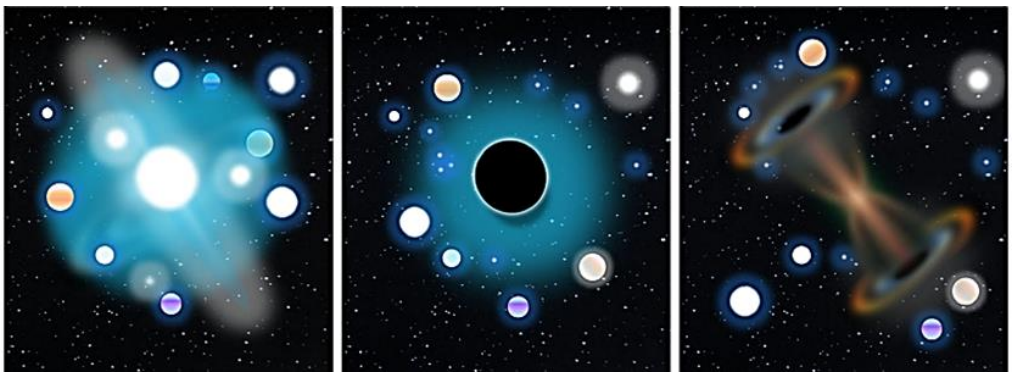


Figure (2.12). White hole, Black hole, and Wormhole [87].

The algorithm technique is partitioned into two distinct stages, namely exploration and exploitation, by the use of a population-based algorithm. The use of the notions of white holes and black holes has potential for the investigation of the search space in Multi-Verse Optimization (MVO). In contrast, wormholes facilitate the process of multi-objective optimization MVO by enabling the exploration and exploitation of search spaces. In addition, we designate each solution with an inflation rate that is proportional to the respective fitness function value of the solution. also instead of iteration, use the term time. In multiverse theory and cosmology, this is a common term. The following principles are applied to MVO universes during optimization [87]:

1. The higher the inflation rate, the greater the likelihood of a white hole.
2. The greater the inflation rate, the lower likelihood of black holes.
3. In universes with a higher inflation rate, objects are more likely to travel through white holes.
4. Universes characterized by diminished rates of inflation tend to see a higher influx of items through black holes.
5. Objects in every universe are subject to random motion.

Utilized a roulette wheel mechanism to mathematically model the white/black hole tunnels and exchange the objects of universes. At each iteration, arrange the universes according to their inflation rates and randomly select one to contain a white hole. The following measures are taken to accomplish this assuming that by [87]:

$$U = \begin{bmatrix} x_1^1 & x_1^2 & \dots & x_1^d \\ x_2^1 & x_2^2 & \cdots & x_2^d \\ \vdots & \vdots & \vdots & \vdots \\ x_n^1 & x_n^2 & \cdots & x_n^d \end{bmatrix} \tag{2.35}$$

Where n is a number of potential universes (solutions), and d is the number of parameters (variables), the position of each universe is given the following relation [87]

$$x_j^i = \begin{cases} x_k^j & r_1 < NI(U_i) \\ x_i^j & r_2 \geq NI(U_i) \end{cases} \tag{2.36}$$

Where r_1 is a random number value in [0,1], $NI(U_i)$ is normalized inflation rate of the i^{th} universe, U_i is shows the i^{th} universe, x_i^j indicates the j^{th} parameter of i^{th} universe, and x_k^i indicates the j^{th} parameter of k^{th} universe selected by a roulette wheel selection mechanism.

There is a positive correlation between a lower inflation rate and an increased probability of things successfully traversing white or black hole tunnels. This technique ensures exploration by necessitating the exchange of items across universes and the occurrence of sudden changes, which are essential for exploring the search space, to uphold the preservation of universes' variety and facilitate the process of exploitation, it is posited that each universe has wormholes, which serve as mechanisms for the random transportation of items between spatial dimensions. The things that have been moved via wormholes are represented by white dots. It is postulated that wormhole conduits are consistently built between a given universe and the most advanced universe known so far, with the purpose of facilitating localized modifications for each world and enhancing the likelihood of wormholes augmenting the pace of inflation. The formulation of this mechanism is as follows [87]:

$$x_j^i = \begin{cases} \begin{cases} x_j + TDR \times \left(\left(ub_j - lb_j \right) \times r_4 + lb_j \right) & r_3 < 0.5 \\ x_j - TDR \times \left(\left(ub_j - lb_j \right) \times r_4 + lb_j \right) & r_3 \geq 0.5 \end{cases} & r_2 < WEP \\ x_j^i & r_2 \geq WEP \end{cases} \tag{2.37}$$

Where r_2, r_3, r_4 are random numbers in [0, 1], ub_j,lb_j shows the upper and lower bound of j^{th} variable. X_j indicates the j^{th} parameter of best universe formed so far. x_i^j indicates the j^{th} parameter of i^{th} universe, and TDR, WEP are a coefficient.

The two main coefficients in this study are the Wormhole Existence Probability (WEP) and the Travelling Distance Rate (TDR). The aforementioned coefficient quantifies the likelihood of the presence of wormholes inside different universes, to accentuate exploitation as the optimization process progresses, it is essential for it to exhibit a linear growth over iterations. The velocity at which an item may be transported via a wormhole in the most extensively studied universe is influenced by the rate of transit over distance, therefore contributing to the determination of the distance rate (variation). In contrast to WEP, TDR is increased over iterations to perform a more precise exploitation/local search within the finest obtained universe. Following is the adaptive formula for both coefficients [85]:

$$WEP = min + t \times \left(\frac{max-min}{T}\right) \tag{2.38}$$

$$TDR = 1 - \frac{t^{\frac{1}{p_o}}}{T^{\frac{1}{p_o}}} \tag{2.39}$$

Where T shows the maximum iterations, t indicates the current iteration, and p_o defines the exploitation accuracy over the iterations.

2.3.8 Sine Cosine Algorithm

The optimization approach is initiated by using population optimization techniques, which include starting the process with a set of randomly generated solutions. The given set undergoes iterative evaluation by an objective function and is improved using a set of rules, which forms the core

of an optimization process. There is no guarantee of promptly achieving a solution in a single iteration. Nevertheless, when the quantity of stochastic techniques (iterations) rises, the likelihood of attaining the global optimum also grows [88,89].

Despite the variations among optimization algorithms in the field of stochastic population-based approaches, the predominant optimization procedures may be categorized into two primary stages: exploration and exploitation. During the exploration phase, optimization algorithms use a significant level of randomness to effectively merge the random solutions within the solution set. This process aims to identify and extract the most promising areas inside the search space [90].

During the exploitation phase, random solutions progress more slowly (gradually), and random variations are significantly less frequent than during the exploration phase [91]. The following are the mathematical equations for updating positions in SCA in both stages [90]:

$$X_i^{t+1} = X_i^t + r_1 \times \sin(r_2) \times \left| r_3 P_i^t - X_i^t \right| \tag{2.40}$$

$$X_i^{t+1} = X_i^t + r_1 \times \cos(r_2) \times \left| r_3 P_i^t - X_i^t \right| \tag{2.41}$$

The application of these two equations is as follows [90]:

$$X_i^{t+1} = \begin{cases} X_i^t + r_1 \times sin(r_2) \times \left| r_3 P_i^t - X_i^t \right|, & r_4 < 0.5 \\ X_i^t + r_1 \times cos(r_2) \times \left| r_3 P_i^t - X_i^t \right|, & r_4 \geq 0.5 \end{cases} \tag{2.42}$$

Where r_1, r_2, r_3 and r_4 are random numbers between [0,1], X_i^t denotes the positions of the current solution in i^{th} dimension at t^{th} iteration, and the variable P_i represents the position of the location point in the i^{th} dimension.

According to the preceding equation, the four basic parameters of SCA are denoted as r_1, r_2, r_3, and r_4. The parameter r_1 specifies the next positions

regions movement direction. The parameter r_2 specifies whether the movement should be in the direction of the objective or away from it. The parameter r_3 designates a random weight score for the target to stochastically emphasize $r_3 > 1$ or deemphasize $r_3 < 1$ the target's influence in determining the distance. Lastly, the parameter r_4 indicates equal transitions between the sine and cosine elements, as shown in Eq. (2.42) [92].

The method referred to as SCA is named based on the values of sine and cosine. Fig (2.13). This analysis illustrates the impact of the Sine and Cosine functions on Eq. (2.40) and Eq. (2.41). The provided graphic serves to visually depict how the used mathematical notations provide a distinct separation between two solutions inside the search space [92].

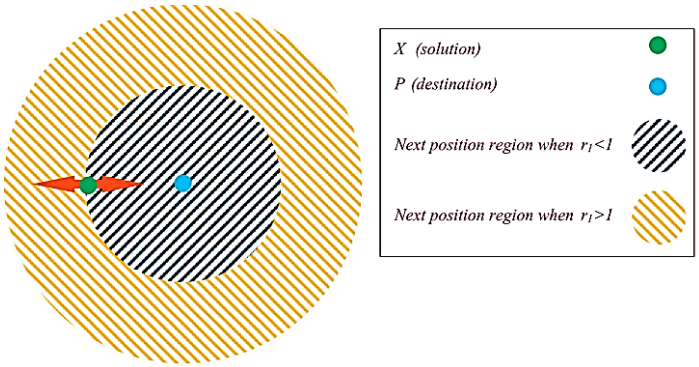


Figure (2.13). The impact of Sine and Cosine functions in Eq. (2.40) and Eq. (2.41) on the following position [92].

The cyclic structure of sine and cosine functions allows for the relocation of a solution around other solutions. This approach enables the systematic exploration of the exploitation search strategy between two potential solutions. To effectively investigate the search space, solutions must be able to seek outside the space among their similar destinations as well. This is possible by modifying the range of the sine and cosine.

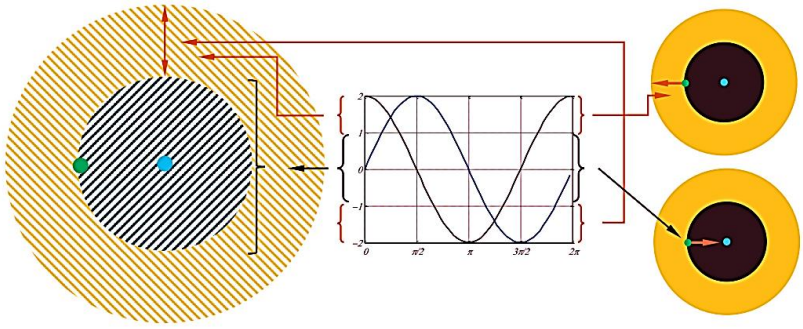


Figure (2.14). Sine and Cosine are evaluated inside the interval [-2, 2] allow a solution to go around (inside the space between them) or beyond (outside the space between them) the destination [92].

The mathematical depiction of the effects of the cosine and sine functions inside the interval [-2, 2] is shown. This mathematical model explains how altering the range area of both sine and cosine functions necessitates a different solution to update their positions outside/inside the space within itself. By giving a random number for r_2 in [2, -2] in Eq. (2.42). This tool facilitates the process of exploration and use of the search space [92].

Any algorithm must be able to strike a balance between exploration and exploitation searches to identify the most promising regions of the search space and then optimally converge to the global optimum. To balance exploration and exploitation, modifies the domains of both sine and cosine in Eqs. (2.40), (2.41), and (2.42) adaptively where [92]:

$$r_1 = a - \frac{a.t}{T} \tag{2.43}$$

Where T is the maximum number of repeated iterations, t is the number of the current iteration, a is a constant value, this equation shows decreases the range of sine and cosine functions over the course of iterations. the SCA algorithm explores the search space when the ranges of sine and cosine functions are in (1,2] and [−2, −1). However, this algorithm exploits the search space when the ranges are in the interval of [−1,1], as depicted in Fig (2.14).

2.3.9 Whale Optimization Algorithm

Research has provided evidence to support the notion that whales possess cognitive abilities, learning capabilities, evaluative skills, communication systems, and a limited capacity for emotional experiences, although far less developed compared to humans. Whales, particularly killer whales, have been found to possess the ability to construct their kind of communication [93].

The most intriguing characteristic of humpback whales is their distinctive foraging method. The act of feeding in this manner is often known as bubble-net feeding. The humpback whale prefers to pursue plankton near the surface. As shown in Fig (2.15), The act of foraging is achieved by generating discrete bubbles in a circular or '9'-shaped trajectory. this behavior was only examined via superficial observations. [93].

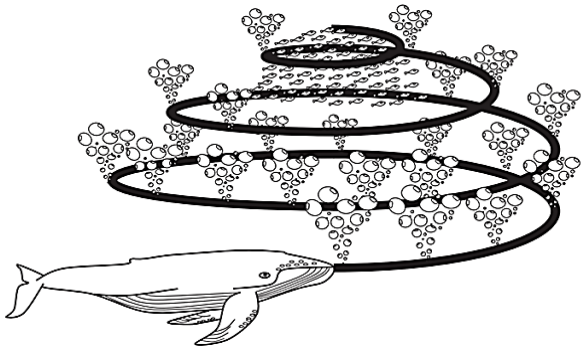


Figure (2.15). Bubble-net feeding behavior of humpback whales [93]

They discovered two maneuvers associated with bubbles and designated them as "upward spirals" and "double loops." During the first maneuver, humpback whales dive approximately 12 meters below the surface, produce spiraling clouds around their prey, and then ascend to the surface. The ultimate maneuver consists of three separate stages, namely the coral loop, lobtail, and capture loop. It is noteworthy to acknowledge that bubble-net feeding is a distinctive behavior exclusive to humpback whales [94].

A mathematical representation of encircling prey, the spiral bubble-net feeding maneuver, and the search for prey is first presented. In encircling prey, humpback whales can identify the location of prey and encircle it. Since the position of the optimal design within the search space is unknown a priori, the WOA algorithm implies that the current best candidate solution is either the target prey or is close to the optimal. Once the best search agent has been identified, the remaining search agents will attempt to adjust their positions toward the best agent. The following equations represent this behavior [94]:

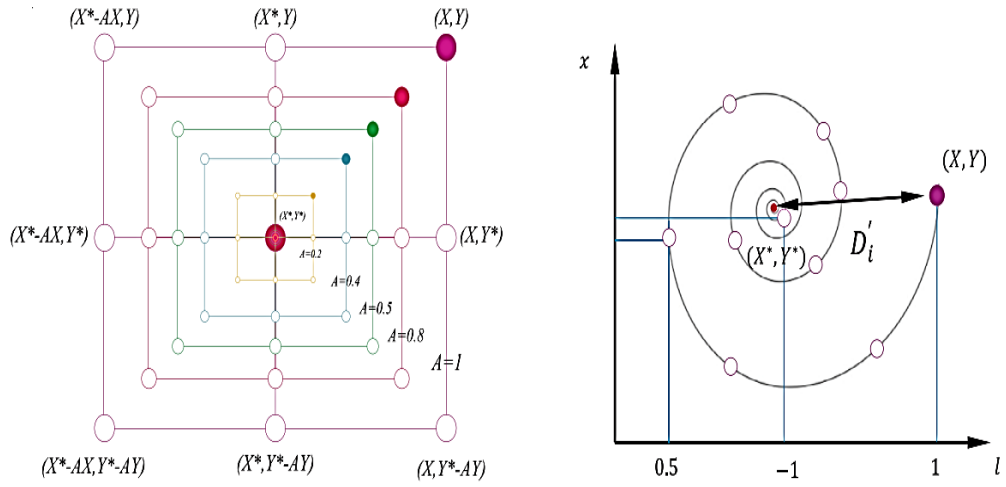
$$\vec{D} = \left|\vec{C}.\overrightarrow{X^*}(t) - \vec{X}(t)\right| \tag{2.44}$$

$$\vec{X}(t+1) = \overrightarrow{X^*}(t) - \vec{A}.\vec{D} \tag{2.45}$$

Where $\vec{A}$ and $\vec{C}$ are coefficient vectors are calculated by Eq. (2.24) and Eq. (2.25) respectively, t indicates the current iteration, X^* is the position vector of the finest solution found up, if a superior solution exists, it should be updated with every iteration, $\vec{X}$ is the position vector.

In bubble-net attack technique (exploitation phase) two approaches are designed to mathematically model the bubble-net behavior of humpback whales, Shrinking encircling mechanism, and Spiral updating position [95].

In the shrinking encircling mechanism this behavior is accomplished by reducing the value of $\vec{a}$ in Eq. (2.24). Note that $\vec{a}$ also reduces the fluctuation range of $\vec{A}$. In other words, $\vec{A}$ is a random value within the interval $[a, -a]$, which decreases from 2 to 0 during iterations. Using random values for $\vec{A}$ in the interval $[1, -1]$, the new position of a search agent can be defined anywhere between the agent's initial position and the position of the current best agent. In 2D space, Fig (2.16). (a). depicts the conceivable coordinates from (X, Y) towards (X^*, Y^*) that can be reached by $0 \leq A \leq 1$ [95].



(a) **(b)**

Figure (2.16). The Bubble-net search method has been implemented in WOA (X^* is the best solution obtained so far): (a) Shrinking encircling mechanism and (b) Spiral updating position [95].

In spiral updating position this method first calculates the distance between the cetacean located at (X, Y) and the prey located at (X^*, Y^*), as shown in Fig (2.16). (b). A spiral equation is then created between the position of the whale and its prey to replicate the helix-shaped movement of humpback whales as follows [94]:

$$\vec{X}(t+1) = \overrightarrow{D'}.e^{bt}.\cos(2\pi l) + \overrightarrow{X^*}(t) \qquad (2.46)$$

Where b is used to establish the geometry of the logarithmic spiral, l is a random number in $[1, -1]$, and $\overrightarrow{D'} = \left|\overrightarrow{X^*}(t) - \vec{X}(t)\right|$ signifies the distance between the i^{th} cetacean and its prey.

Observe that humpback whales swim around their prey in a diminishing circle and a spiraling path at the same time along a spiral path. To model this concurrent behavior, we assume a 50% chance of selecting either the decreasing encircling mechanism or the spiral model to adjust the position

of whales during optimization. The following is the mathematical model [95]:

$$\vec{X}(t+1) = \begin{cases} \overrightarrow{X^*}(t) - \vec{A}.\vec{D} & if\ \ p < 0.5 \\ \overrightarrow{D'}.e^{bt}.cos(2\pi l) + \overrightarrow{X^*}(t) & if\ \ p > 0.5 \end{cases} \quad (2.47)$$

Where p is a random number in [0,1].

In search for prey look for food (exploration phase) The identical strategy based on the $\vec{A}$ vector variant can be used to seek prey (exploration). In reality, humpback whales seek indiscriminately based on their relative positions. Therefore, we use $\vec{A}$ with random values greater than 1 or less than -1 to relocate the search agent away from a reference whale.

In contrast to the exploitation phase, in the exploration phase revise the position of a search agent based on a randomly selected search agent rather than the finest search agent discovered to date. This mechanism and $|\vec{A}| > 1$ place an emphasis on exploration and permit the WOA algorithm to conduct a global search. The mathematical model looks like this [94]:

$$\vec{D} = |\vec{C}.\overrightarrow{X_{rand}} - \vec{X}| \quad (2.48)$$

$$\vec{X}(t+1) = \overrightarrow{X_{rand}} - \vec{A}.\vec{D} \quad (2.49)$$

Where $\overrightarrow{X_{rand}}$ a position vector chosen at random from the current population (a random whale).

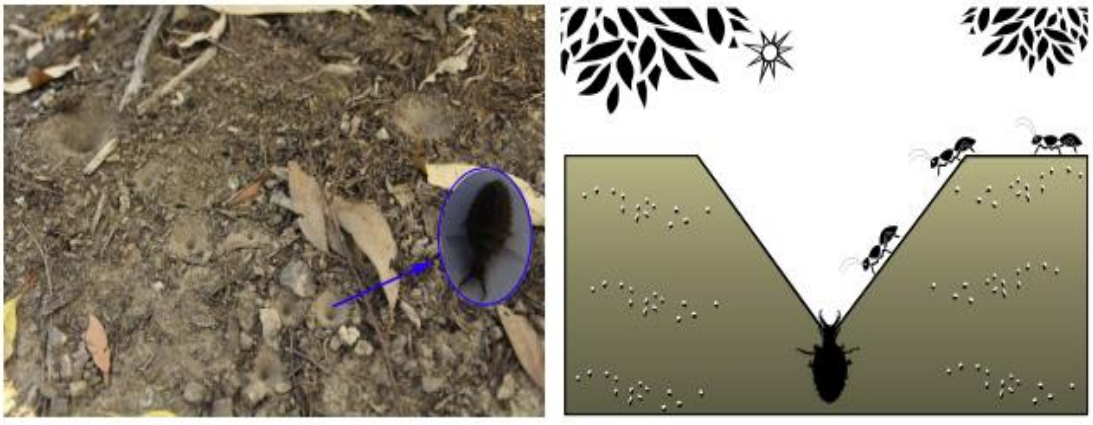
The WOA begins with a collection of random solutions. At each iteration, search agents adjust their positions relative to either a randomly selected search agent or the finest solution thus far obtained. The parameter is decreased from 2 to 0 so that exploration and exploitation can occur, respectively. When $|\vec{A}| > 1$, a random search agent is selected, and when $|\vec{A}| < 1$, the optimal solution is chosen for updating the position of the search

agents. Depending on the value of p, WOA is capable of switching between spiral and circular motion. The WOA algorithm concludes when a termination criterion is satisfied [95].

2.3.10 Ant Lion Optimization

Antlions, sometimes referred to as doodlebugs, belong to the families Myrmeleontidae and Neuroptera, which are classified as net-winged insects. Ant lions have two major stages in their life cycle: larvae and adults. The larvae have a natural lifecycle of up to three years, while adults only live for three to five weeks. Adult antlions undergo metamorphosis within a cocoon. They primarily prey on larvae, and during maturation they reproduce [96].

Their titles derive from their distinctive foraging behavior and preferred prey. An antlion larva excavates a cone-shaped hole in the sand by moving along a circular path and ejecting sand with its mouthparts. its enormous mandible [94]. Fig (2.17). (a) depicts several cone-shaped cavities of varying diameters.



(a) **(b)**

Figure (2.17). Cone-shaped traps and hunting behavior of antlions [97].

As depicted in Fig (2.17). (b), after excavating the trap, the larvae conceal beneath the bottom of the cone (as a sit-and-wait predator) and wait

for insects (particularly ants) to become ensnared in the pit. The cones edge is sufficiently pointed for insects to readily descend to the bottom of the trap. Once the antlion detects prey in the snare, it attempts to capture it. However, insects typically attempt to evade capture and escape the snare [97].

The major idea for the ALO algorithm was derived from the foraging behavior shown by antlion larvae. The ALO algorithm is designed to replicate the dynamic interaction between antlions and ants inside a snare-like environment. To simulate such interactions, ants are required to traverse the search space, while antlions are permitted to pursue them and improve their fitness by using traps. Since ants move stochastically when seeking sustenance in the wild, a random is used to model their movement as follows [97]:

$$X(t) = [0, cumsum(2r(t_1) - 1), cumsum(2r(t_2) - 1), \dots, cumsum(2r(t_n) - 1) \qquad (2.50)$$

Where n is the maximum number of iterations, $cumsum$ calculates the cumulative sum, t is the step of random walk (iteration), $r(t)$ the stochastic function defined as:

$$r(t) = \begin{cases} 1 & if\ rand\ > 0.5 \\ 0 & if\ rand \leq 0.5 \end{cases} \qquad (2.51)$$

The position of ants is saved and employed during optimization in the matrix below [97]:

$$M_{ant} = \begin{bmatrix} A_{1,1} & A_{1,2} & \dots & A_{1,d} \\ A_{2,1} & A_{2,2} & \cdots & A_{2,d} \\ \vdots & \vdots & \vdots & \vdots \\ A_{n,1} & A_{n,2} & \cdots & A_{n,d} \end{bmatrix} \qquad (2.52)$$

Where M_{ant} is the matrix used to save the position of each ant, n is the number of ants, d is the number of variables, and the symbol $A_{i,j}$ represents the value of the j^{th} variable or dimension of the i^{th} ant.

It is worth noting the resemblance seen between ants and PSO as well as individuals in GA. The positions of an ant correspond to the parameters of the solutions. In addition to ants, we presume antlions are also lurking in the search area. The following matrices are employed to store their positions [97]:

$$M_{antlion} = \begin{bmatrix} AL_{1,1} & AL_{1,2} & \dots & AL_{1,d} \\ AL_{2,1} & AL_{2,2} & \cdots & AL_{2,d} \\ \vdots & \vdots & \vdots & \vdots \\ AL_{n,1} & AL_{n,2} & \cdots & AL_{n,d} \end{bmatrix} \quad (2.53)$$

Where $M_{antlion}$ is the matrix used to save the position of each antlion, n is the number of antlions, d is the number of variables, and $AL_{i,j}$ shows the value of the j^{th} variable (dimension) of i^{th} antlion.

All random walks are founded on the Eq. (2.50). At every phase of optimization, ants revise their positions with random treks. However, since every search space has a boundary (variable range), Eq. (2.50) cannot be used directly to update the position of ants. To maintain the random walks within the search space, they are normalized using the min-max normalization of the following formula [97]:

$$X_i^t = \frac{(X_i^t - a_i) \times (d_i - c_i^t)}{(d_i^t - a_i)} + c_i \quad (2.54)$$

Where a_i is the minimum of random walk of i^{th} variable, c_i^t is the minimum of i^{th} variable at t^{th} iteration, and d_i^t indicates the maximum of i^{th} variable at t^{th} iteration.

Antlions traps affect the random journeys of ants. The following equations are proposed to mathematically model this assumption [97]:

$$c_i^t = Antlion_j^t + c^t \quad (2.55)$$

$$d_i^t = Antlion_j^t + d^t \quad (2.56)$$

Where c^t is the minimum of all variables at t^{th} iteration, d^t indicates the vector including the maximum of all variables at t^{th} iteration, c_i^t is the minimum of all variables for i^{th} ant, d_i^t is the maximum of all variables for i^{th} ant, and $Antlion_j^t$ shows the position of the selected j^{th} antlion at t^{th} iteration.

During optimization, the ALO algorithm must use a roulette wheel operator to select antlions based on their fitness. This mechanism gives sturdier antlions a greater chance of capturing ants [97]. Sliding ants toward the antlion Using the proposed mechanisms, antlions possess the ability to design traps that are commensurate with their level of fitness. and ants must move unpredictably. In order to mathematically model this behavior, the hyper-spherical radius of ants' random travels is decreased adaptively. Proposed in this regard are the following equations [96]:

$$c^t = \frac{c^t}{I} \tag{2.57}$$

$$d^t = \frac{d^t}{I} \tag{2.58}$$

$$I = 10^w \frac{t}{T} \tag{2.59}$$

Where c^t is the minimum value among all variables at t^{th} iteration, d^t indicates the vector including the maximum of all variables at t^{th} iteration, t is the current iteration, T is the maximum number of iterations, and w is a constant defined based on the current iteration ($w = 2$ when $t > 0.1T$, $w = 3$ when $t > 0.5T$, $w = 4$ when $t > 0.75T$, $w = 5$ when $t > 0.9T$, and $w = 6$ when $t > 0.95T$). Basically, the constant w can adjust the accuracy level of exploitation.

In predation and rebuilding the pit when an ant reaches the bottom of the trench and is captured by the antlion, the search is complete. The antlion then

drags the ant into the sand and consumes its corpse. For mimicking this process, it is assumed that ants capture prey when they become fitter (enter sand). To increase its chances of capturing new prey, an antlion is required to adjust its position to the most recent location of its prey. Proposed in this regard is the following equation [97]:

$$Antlion_j^t = Ant_i^t \quad if\ f(Ant_i^t) > f(Antlion_j^t) \qquad (2.60)$$

Where t shows the current iteration, $Antlion_j^t$ shows the position of selected j^{th} antlion at t^{th} iteration, and Ant_i^t indicates the position of i^{th} ant at t^{th} iteration.

Elitism is a crucial characteristic of evolutionary algorithms that enables them to maintain the optimal solution at any stage of optimization. The finest antlion obtained thus far in each iteration is kept and regarded as elite. As the finest antlion, the elite should be able to influence the movements of all insects throughout iterations. Consequently, it is supposed that every ant wanders arbitrarily around a chosen antlion simultaneously by the roulette wheel and the elite as follows [97]:

$$Ant_i^t = \frac{R_A^t + R_E^t}{2} \qquad (2.61)$$

Where R_A^t is the random walk around the antlion selected by the roulette wheel at t^{th} iteration, R_E^t is the random walk around the elite at t^{th} iteration, and Ant_i^t indicates the position of i^{th} ant at t^{th} iteration.

The aim of using these algorithms is to minimize SLL and this will be achieved by changing the positions of the antenna elements better, which results in a good SLL arrangement by following the implementation steps in Chapter three.

3

CHAPTER THREE

'INVESTIGATED OPTIMIZATION ALGORITHMS FOR SLL REDUCTION'

Chapter Three

Investigated Optimization Algorithms for SLL Reduction

3.1 Introduction

In this chapter, we will present the nutshell needed for the general system of LAA and the steps needed for each technique of optimization technique used. Where the process of optimization is to minimize SLL are secondary radiation patterns that appear alongside the main lobe and can cause interference with neighboring systems, and degrade overall system efficiency. Combining antenna systems with algorithms allows real-time monitoring of the environment and the ability to adjust the array's characteristics dynamically by changing the number of antenna elements and the values of the parameters. This adaptability enables continuous optimization of the antenna's radiation pattern, leading to reduced SLL.

3.2 Linear Antenna Array Models

To reduce the SLL in the radiation pattern by LAA, use different techniques. Some ways to reduce SLL are optimization techniques to find the best element placements and excitement while maintaining the main lobe properties required. Increasing the number of elements, the arrays and changing the position of elements can help reduce SLL and improve the direction of the main beam. Fig (3.1) shows used optimization of the antenna system.

It is crucial to remember that obtaining low side lobe levels frequently necessitates making trade-offs with other elements like the main lobe's breadth and gain. The process of improving an antenna can be complicated, thus software and modeling tools might be helpful.

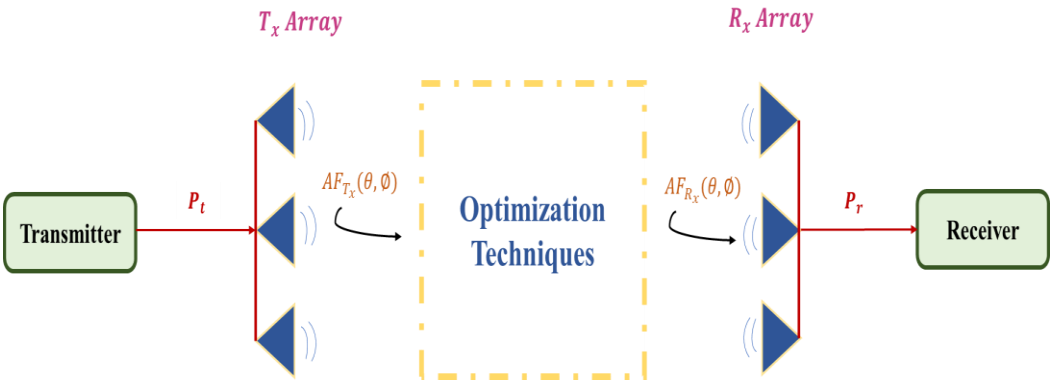


Figure (3.1). Block diagram antenna system

The AF in LAAs is typically found by engineers and researchers in the field of antenna engineering and electromagnetic theory. It involves the mathematical analysis of the array's geometry, element spacing, and radiation properties to determine how the individual antenna elements combine their radiation to form the overall radiation pattern of the array. The process of finding the array factor can involve complex mathematical calculations, and in some cases, numerical methods or software simulations may be used for accurate results. The AF can be calculated through Eq. (2.6) and will be applied to the function of fitness.

Fitness function is a mathematical function shown in Eq. (2.7) used to evaluate the performance or quality of a particular solution or candidate solution within a given problem domain. the fitness function is used to assess how well a specific arrangement of antenna elements meets the desired radiation pattern or performance criteria.

The fitness function for an LAA typically involves comparing the array's radiation pattern to a target pattern or a set of desired specifications. The closer the radiation pattern of the array aligns with the desired pattern, the higher the fitness function. The fitness function should be designed in a way that encourages convergence toward the best solution that meets the design requirements.

3.3 Systems Setup

In this thesis, an array of antennas with different numbers of elements (8,16, 32, 64, 128, and 256) are used in the proposed system. Distributed elements are arranged in linear form.

The LAA function calculates the AF for each element and then compares it with the previous array factor to give the decision of the position and feeding of each element in the array. This process will be applied to each element of the array and then plotted to compare the results.

Ten algorithms are employed to get the desired "minimum SLL" result. Based on the ideal level of the side lobe, these algorithms will select the best case for amplitude and element distribution. according to Fig. (3.2). symbolize the whole approach to antenna improvement.

3.4 Investigated Optimization

As stated previously, the proposed system employs ten proposed algorithms: PSO, GA, FPA, GWO, SSA, MFO, MVO, SCA, and ALO.

Optimization algorithms that are tested in this thesis were used to reduce the SLL of the radiation pattern of antenna array. The algorithms make iterations to evaluate the position of the antenna elements. Depending on the best fitness function values. The following is a flowchart for each algorithm, represented by the steps being taken to reach the optimal solution. Plus, tables represent the parameters of each algorithm.

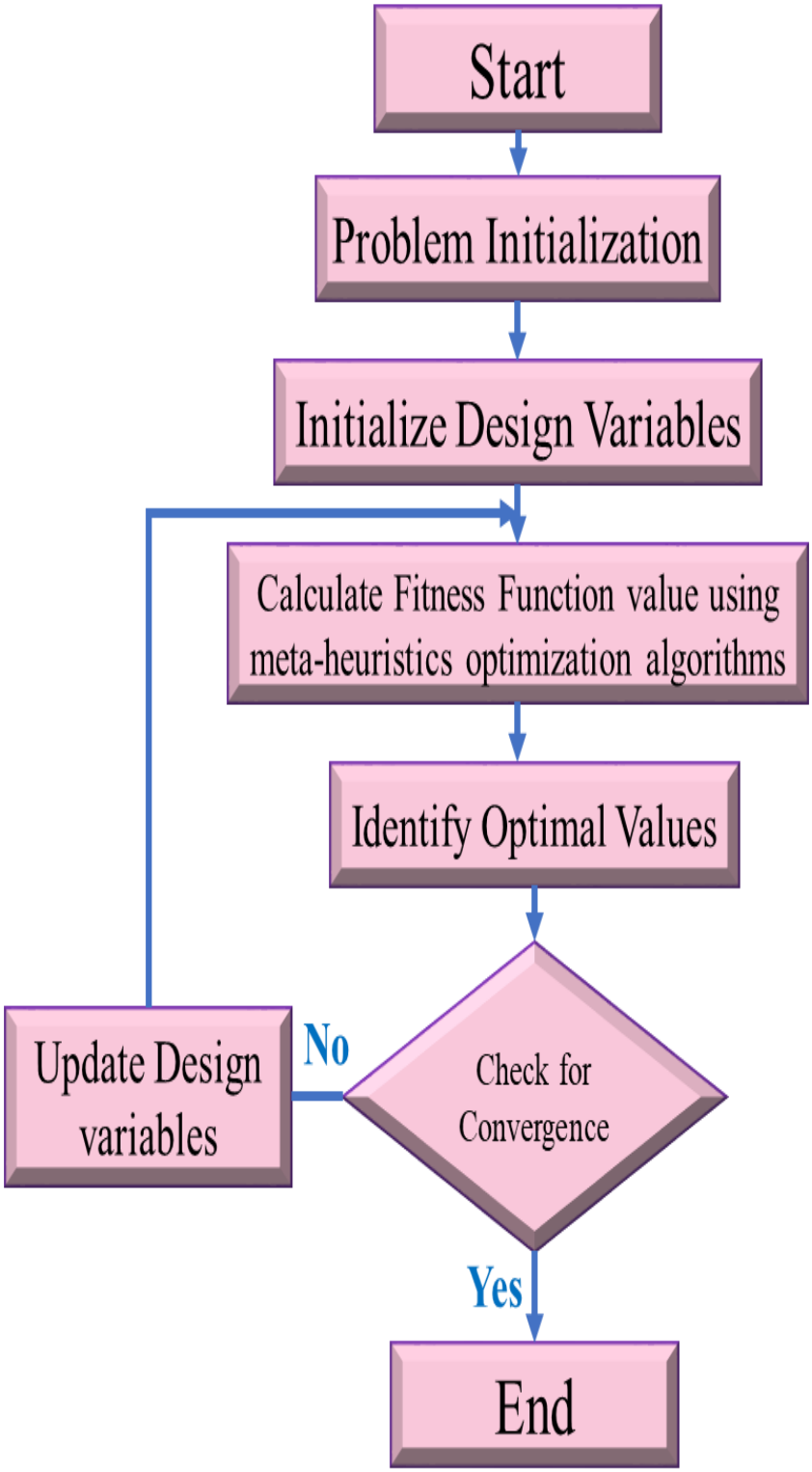


Figure (3.2) Flowchart of the antenna optimization.

3.4.1 Steps and Parameters Based on PSO Algorithm

These parameters are in Table (3.1). may be variables and the amount of this change aims to minimize SLL, for example, iterations are one of the variable parameters that has been changed several times to maximize the reduction of SLL, as for population size, and max stall iteration. Fig (3.3). shows the steps being followed to minimize SLL.

Table (3.1). Illustrates the PSO parameters

Parameters	Value
Iterations	1000
Population Size	200
Max stall iterations	50
Correction factor (C_1&C_2)	2
Inertia weight	1

The Steps to Implement the Particle Swarm Optimization Algorithm:

1. Define the optimization objective function and set the initial positions and velocities of a particle swarm to be random.
2. 2. Evaluate the fitness of each particle using the objective function, and update the optimal personal position and fitness value for every particle using Equation (2.7).
3. Identify the particle with the best fitness in the swarm and update the global best position and fitness.
4. Using Eq. (2.11), update the velocities of each particle according to its current velocity, personal the best, and global optimal position. Using Eq. (2.12), update the position of each particle in accordance with its new velocity.
5. If a termination condition has not been satisfied, return to step 2. If not, proceed to the following step.

6. Return the global best position as the optimized solution. The actual implementation might involve additional considerations, such as inertia weight, acceleration coefficients, and boundary constraints.

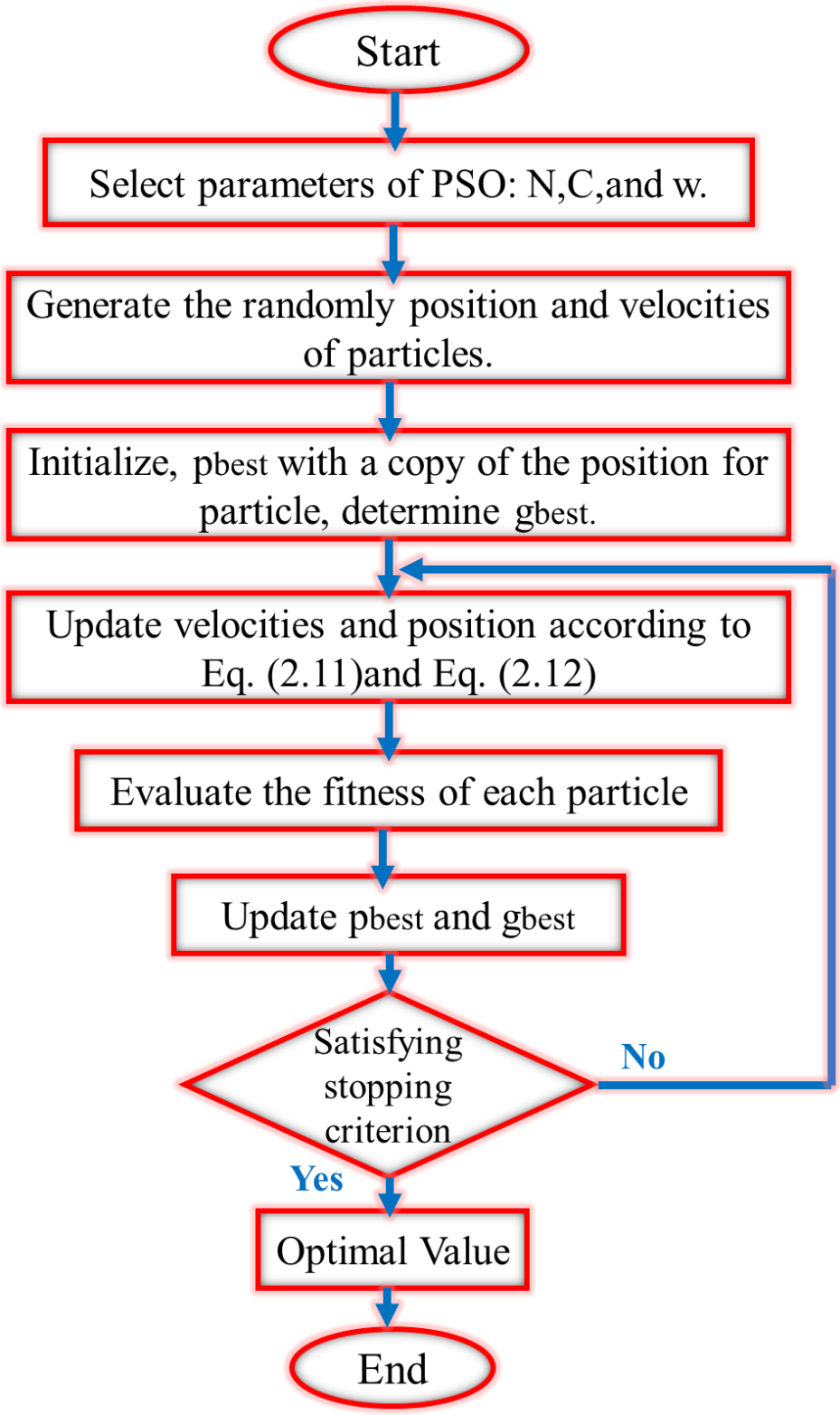


Figure (3.3) Flowchart of PSO algorithm.

3.4.2 Steps and Parameters Based on GA

These parameters are in Table (3.2). may be variables and the amount of this change aims to minimize SLL, for example, Generations is one of the variable parameters it has been changed several times to maximize the reduction of SLL, for population size and max stall generation. Fig (3.4). exhibits the processes followed to reduce SLL.

Table (3.2). Illustrates the GA parameters

Parameters	Value
Generations	1000
Population Size	200
Max stall iterations	50

The Steps to Implement the Genetic Algorithm:

1. Initialize a population of candidate solutions (chromosomes) randomly or using a heuristic and define the objective function to be optimized.
2. Evaluate the fitness of each chromosome in the population using the objective function by Eq.(2.7).
3. Select chromosomes from the population for reproduction based on their fitness, Higher fitness should increase the likelihood of selection.
4. Apply crossover (recombination) to selected pairs of chromosomes to create new offspring, Crossover can be one-point, two-point, uniform, or other strategies.
5. To foster diversity, introduce random sporadic changes (mutations) to the offspring of the genes.
6. Replace a portion of the existing population with newly-created offspring, such as strategies of generational replacement or steady-state replacement.

7. If a termination condition is met, If not converged, go back to step 2. Otherwise, proceed to the next step.
8. Return the chromosome with the highest fitness as the optimized solution. and the real implementation needs to define the specifics of encoding, crossover, mutation rates, and other parameters.

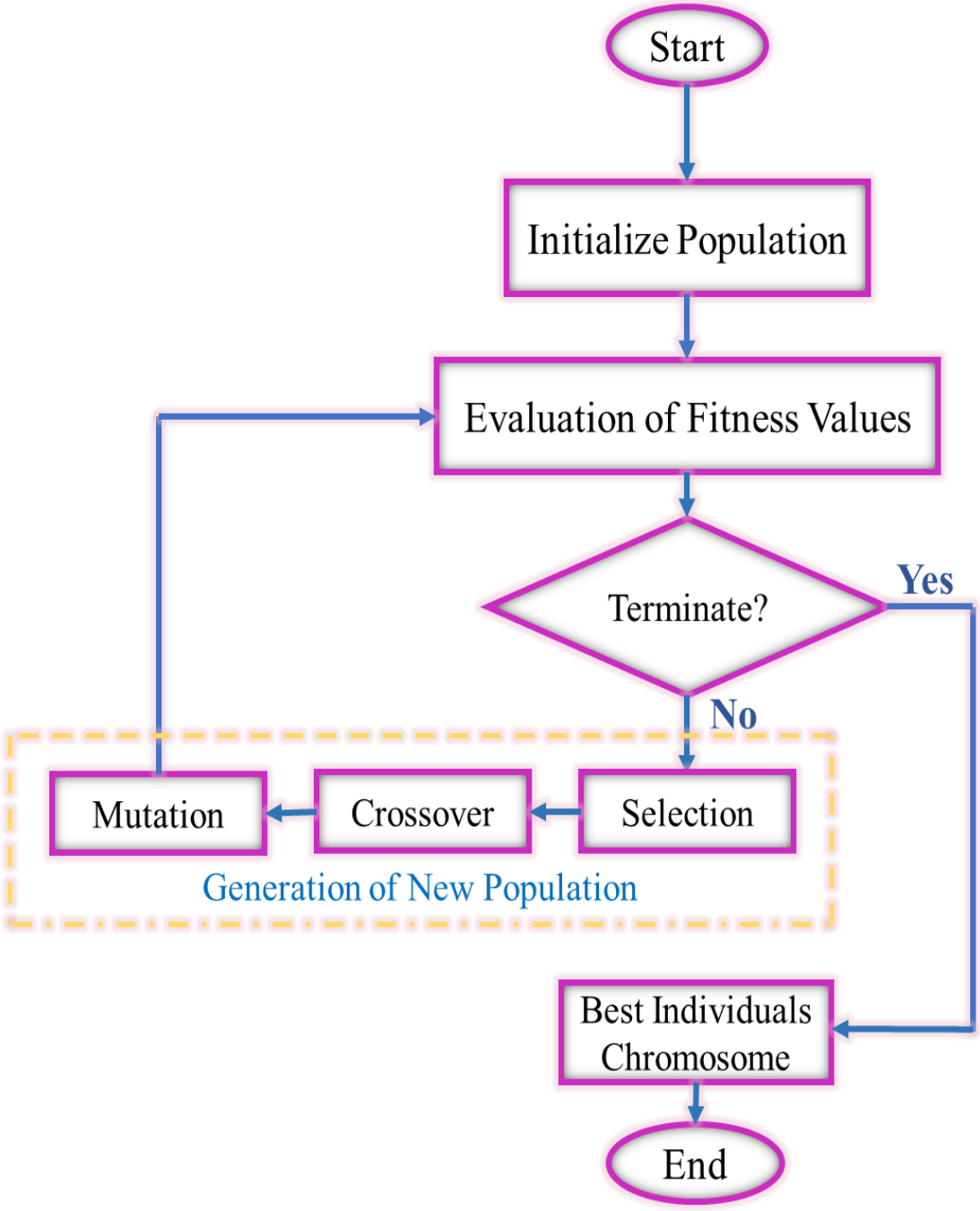


Figure (3.4) Flowchart of GA.

3.4.3. Steps and Parameters Based on FPA

The parameters are listed in Table (3.3). may be variables, this change is intended to minimize SLL; for instance, population size is one of the variable parameters that has been altered multiple times to maximize the reduction of SLL, as in iterations, probability switch for FPA and flower attraction rate**.** Fig (3.5). shows the steps that must be followed to minimize SLL.

Table (3.3). Illustrates the FPA parameters

Parameters	Value
Iterations	1000
Population Size	200
Probability Switch for FPA	0.05-0.95
Flower attraction rate	1.5
Scaling factor	0.1
Uniform distribution	0-1

The Steps to Implement the Flower Pollination Algorithm:

1. Start a population of flowers with random positional distributions or use a heuristic and define the objective function to be optimized.
2. Evaluate the fitness of each flower using the objective function by Eq. (2.7).
3. Pollinate flowers by adjusting their positions based on their fitness and neighboring flowers, use a pollination mechanism that can include exploration and exploitation.
4. Update the positions of flowers after pollination, updating based on a probability factor, where better solutions are more likely to influence others.
5. Apply a local search operation to a subset of flowers to enhance exploitation by Eq. (2.21).

6. Check if a termination condition is met, if not converged, go back to step 2. Otherwise, proceed to the next step.
7. Return the flower with the best fitness as the optimized solution. and It's important to note that Flower Pollination Algorithm parameters and mechanics can vary based on different implementations.

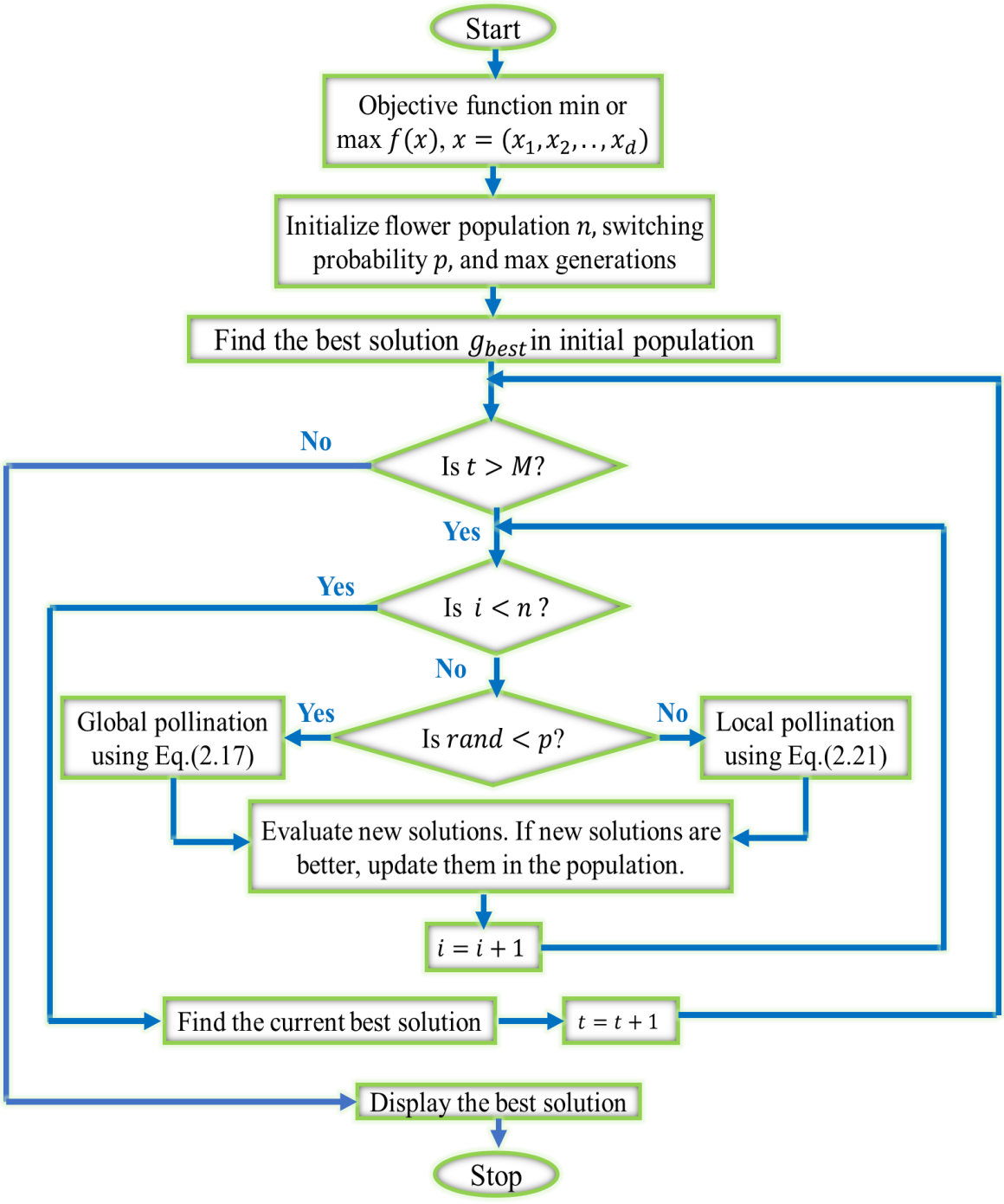


Figure (3.5) Flowchart of FPA.

3.4.4. Steps and Parameters Based on GWO Algorithm

The parameters are listed in Table (3.4). may be variables, and this change is intended to reduce SLL; for illustration, population size is one of the variable parameters that has been modified to reduce SLL, as in iterations Fig (3.6). displays the steps taken to reduce SLL.

Table (3.4). Illustrates the GWO parameters

Parameters	Value
Iterations	1000
Population Size	200
Search Space	Ub & Lb

The Steps to Implement the Grey Wolf Optimization Algorithm:

1. Start a grey wolf population with positions that are random or using a heuristic, and define the objective function to be optimized.
2. Evaluate the fitness of each grey wolf using the objective function by Eq.(2.7).
3. Identify the α, β, and δ wolves, which are the 1^{st} with the highest, 2^{nd}-highest, and 3^{rd} highest fitness, respectively. Update their positions based on their current positions and those of other wolves.
4. Calculate search equations to update the positions of the remaining wolves using alpha, beta, and delta positions as references. Update the positions of the grey wolves based on the search equations. this step balances exploration and exploitation using (2.27).
5. Check if a termination condition is met, If not converged, go back to step 2. Otherwise, proceed to the next step.
6. Return the position of the alpha wolf as the optimized solution.

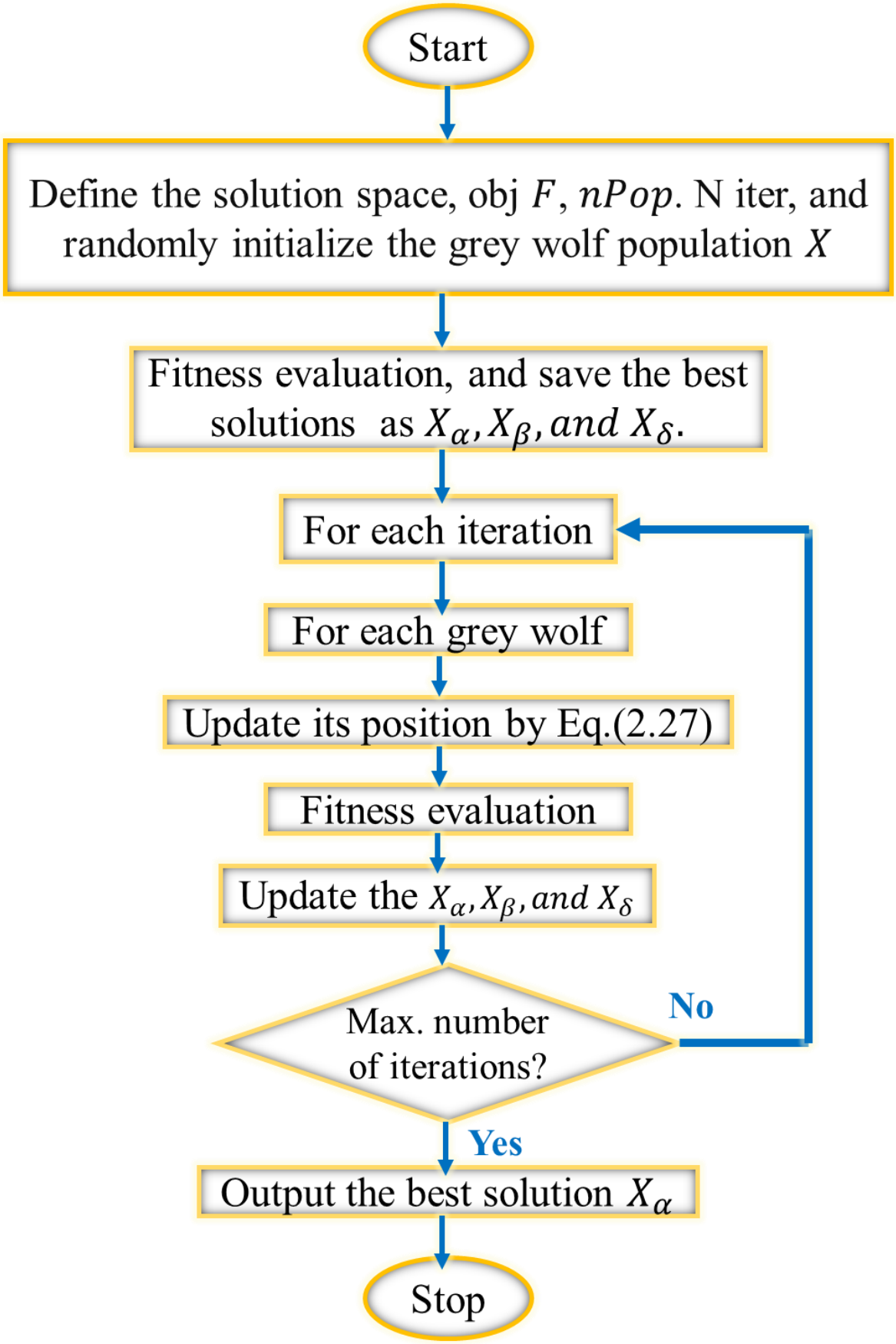


Figure (3.6) Flowchart of GWO algorithm.

3.4.5. Steps and Parameters Based on SSA

The parameters are listed in Table (3.5). may be variables, and this change is intended to reduce SLL, the population size of producers when changing its value will affect reduced SLL, as in iterations, and search space. Fig (3.7). shows the steps being followed to minimize SLL.

Table (3.5). Illustrates the SSA parameters

Parameters	Value
Iterations	1000
Population Size	200
The population size of producers	0.2
Search Space	Ub & Lb

The Steps to Implement the Sparrow Search Algorithm:

1. Create a population of particles with arbitrary positions and velocities. and define the objective function to be optimized
2. Using the objective function, assess the fitness of each particle and update the optimal position and fitness of each particle using Eq. (2.7).
3. Determine the sparrows fitness using the objective function and update each sparrows optimal personal position and fitness.
4. Identify the particle with the optimal fitness in the population and update the global best position and fitness.
5. Update each particles velocity based on its current velocity, personal the best position, and global optimal positions, and each particles position is updated based on its new velocity.
6. Verify that a termination condition is satisfied. If convergence has not occurred, return to step 3. If not, continue to the next step.
7. Return the global best position as the optimized solution.

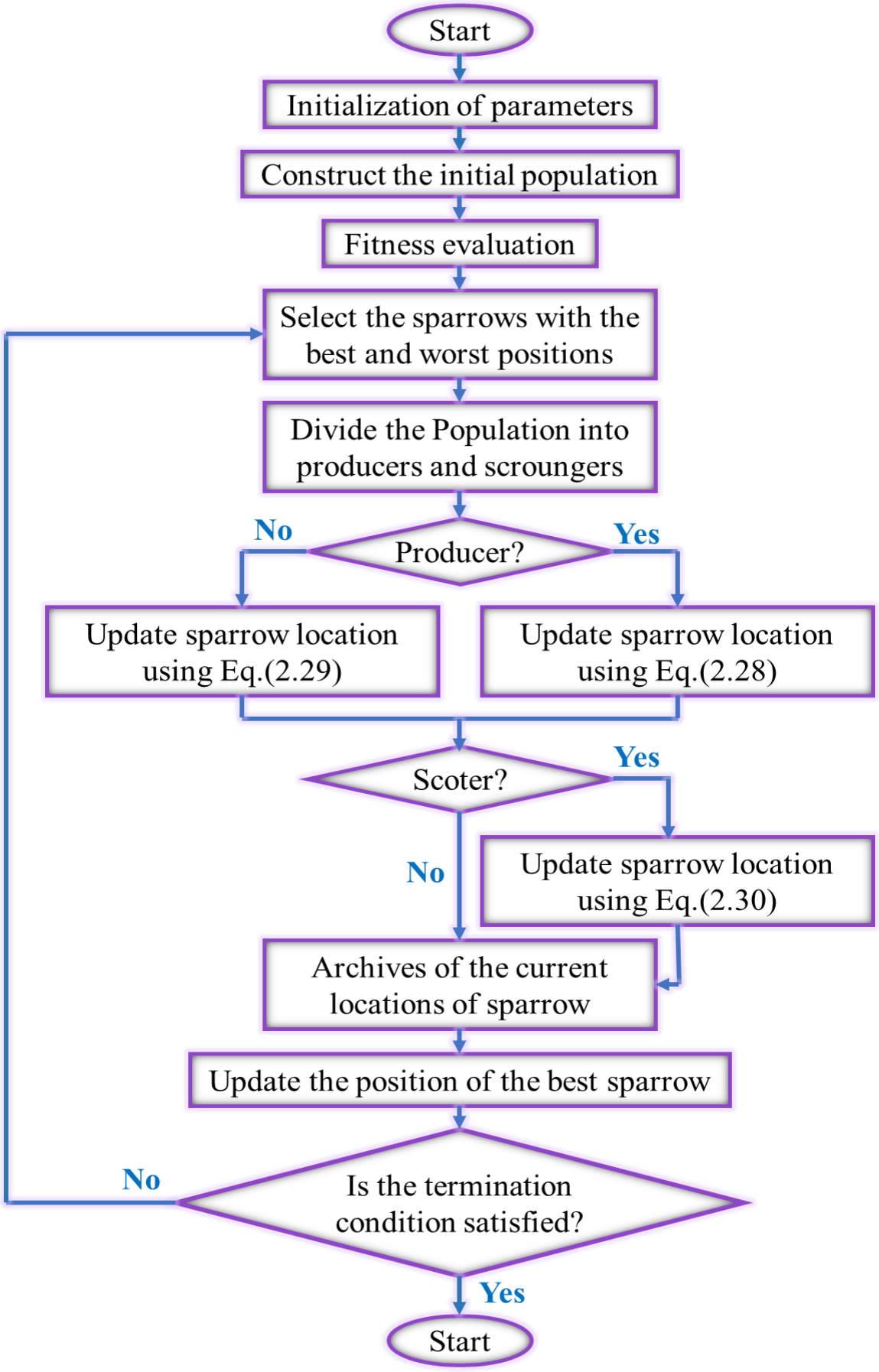


Figure (3.7) Flowchart of SSA.

3.4.6. Steps and Parameters Based on MFO Algorithm

Table (3.6). contains the parameters. may be variables, and the extent of this change is intended to reduce SLL; for example, the number of iterations when its value is changed, will affect reduced SLL, as in population size, spiral flight search, and search space. Fig (3.8).exhibits the steps followed to reduce SLL.

Table (3.6). Illustrates the MFO parameters

Parameters	Value
Iterations	1000
Population Size	200
Search Space	Ub & Lb
Spiral Flight Search	1

The Steps to Implement the Moth Flame Optimization Algorithm:

1. Randomly position a population of moths to begin, and define the objective function to be optimized.
2. Evaluate the fitness of each moth using the objective function by Eq.(2.7).
3. Update the positions of moths based on their current positions, the best moth's position, and the light intensity (fitness).
4. Adjust the light intensity based on the fitness of the moths, Light intensity can be used to guide moths toward better solutions.
5. Update moth positions by moving them toward the light source (better solutions).
6. Check if a termination condition is met, If not converged, go back to step 2. Otherwise, proceed to the next step.
7. Return the position of the moth with the best fitness as the optimized solution.

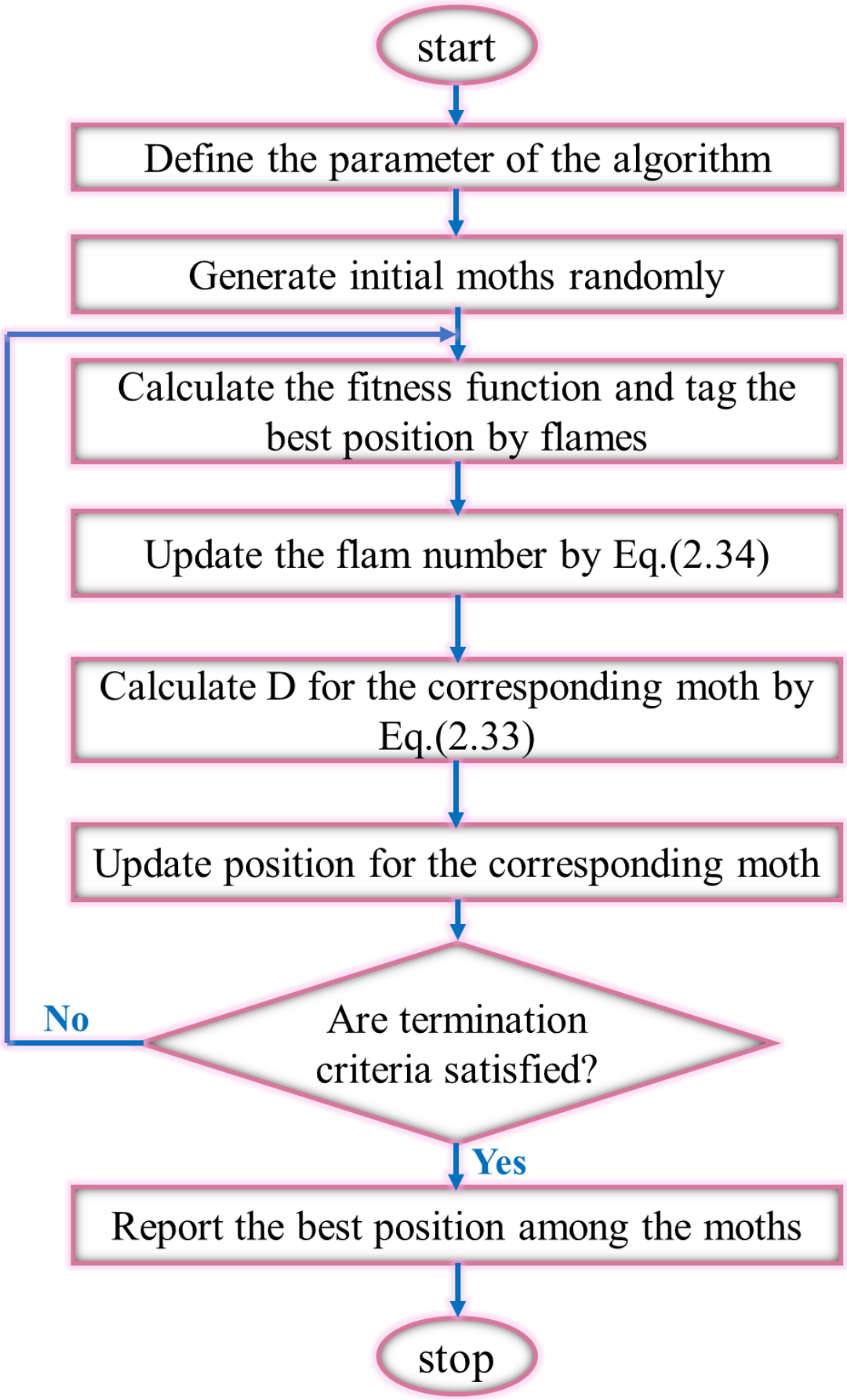


Figure (3.8) Flowchart of MFO algorithm.

3.4.7. Steps and Parameters Based on MVO Algorithm

Table (3.7). contains the parameters. may be variables, and the extent of this change is intended to reduce SLL; for example, the number of exploitation accuracy over the iterations when its value is changed will have an effect on reduced SLL, as in iterations, and population size. Fig (3.9). displays the processes taken to reduce SLL.

Table (3.7). Illustrates the MVO parameters

Parameters	Value
Iterations	1000
Population Size	200
Search Space	Ub & Lb
Exploitation Accuracy Over the Iterations	6
Max_WEP & Min_WEP	1 & 0.2

The steps to Implement the Multiverse Optimization Algorithm:

1. Randomize the initial positions of a population of universes or use a heuristic and define the objective function to be optimized.
2. Evaluate the fitness of each universe using the objective function by Eq.(2.7).
3. Identify the universe with the best fitness (brightest universe) in the population and Update the position of the brightest universe based on the current positions of all universes.
4. Update the positions of all universes based on the positions of the brightest universe and other universes.
5. Check if a termination condition is met. If not converged, go back to step 2. Otherwise, proceed to the next step.
6. Return the position of the brightest universe as the optimized solution.

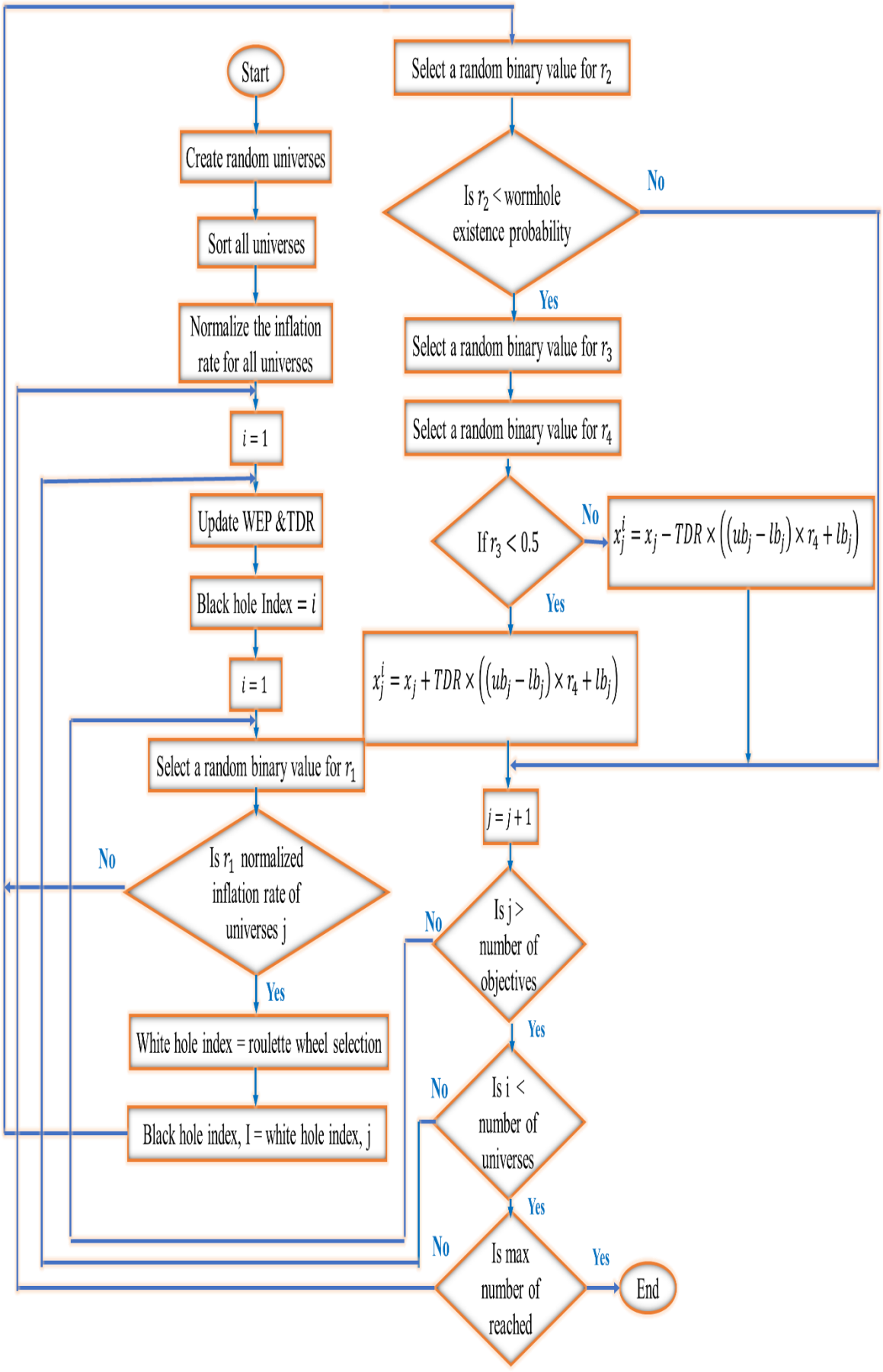


Figure (3.9) Flowchart of MVO Algorithm.

3.4.8 Steps and Parameters Based on SCA

The parameters are shown in Table (3.8). may be variables, and the extent of this change is intended to reduce SLL; for example, the number of current iterations, when its value is altered, will have an effect on decreased SLL, as in population size and iteration. Fig (3.10). shows the steps that must be followed to minimize SLL.

Table (3.8). Illustrates the SCA parameters

Parameters	Value
Iterations	1000
Population Size	200
Current iteration	2
Search Space	Ub & Lb

The Steps to Implement the Sine Cosine Algorithm:

1. Initialize a population of solutions with random values or use a heuristic and define the objective function to be optimized.
2. Evaluate the fitness of each solution using the objective function by Eq.(2.7).
3. Identify the population solution with the highest fitness and update the global optimal solution and fitness.
4. Update the values of solutions using the sine and cosine functions. the updated values are influenced by the global best solution and the current iteration.
5. Check if a termination condition is met, If not converged, go back to step 2. Otherwise, proceed to the next step.
6. Return the global best solution as the optimized solution.

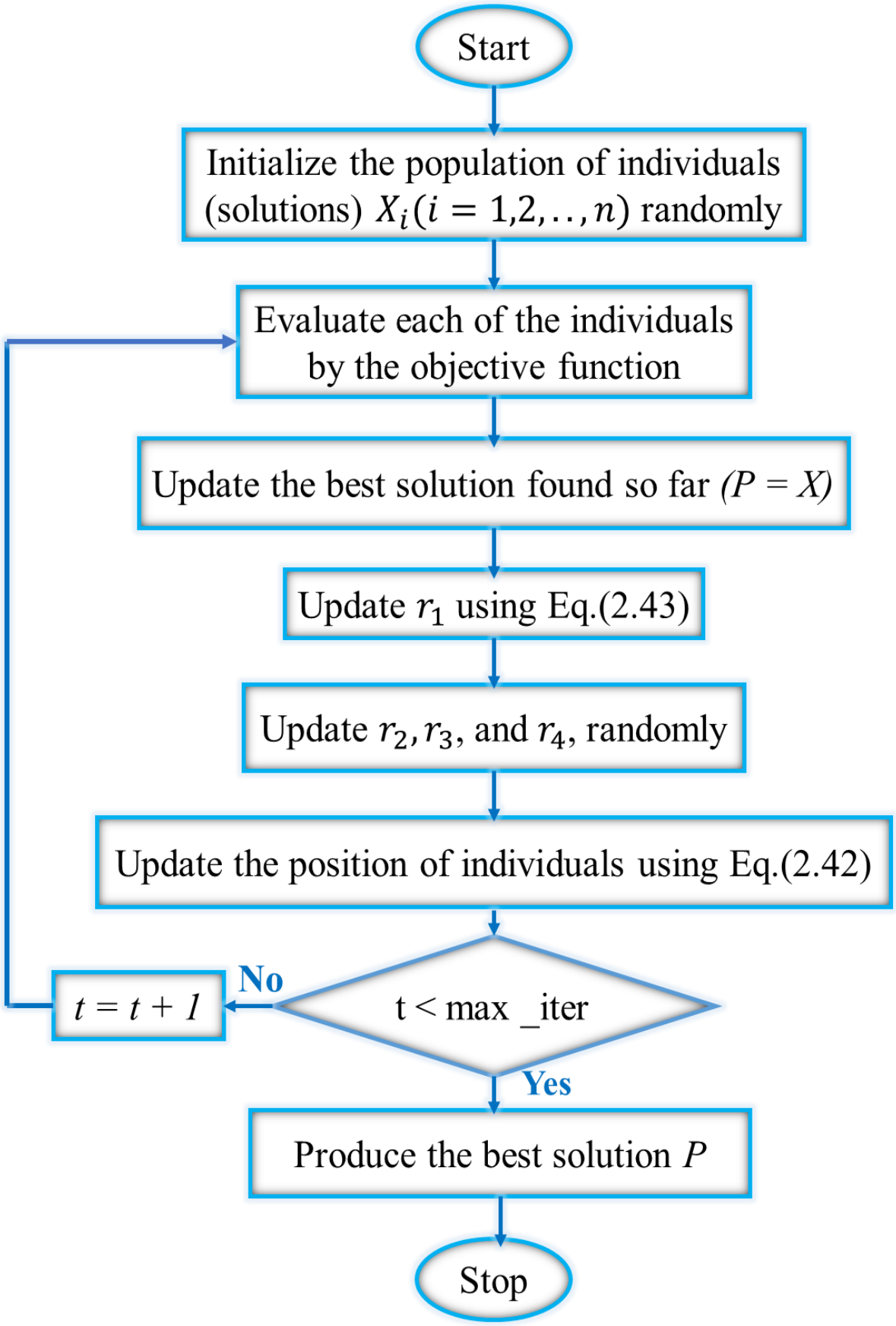


Figure (3.10). Flowchart of SCA.

3.4.9 Steps and Parameters Based on WOA

Table (3.9). contains the parameters. may be variables, and the extent of this change is intended to reduce SLL; for example, the number of spiral update exponent when its value is changed will have an effect on reduced SLL, as in iterations, population size, probability, and search space. Fig (3.11). exhibits the steps followed to reduce SLL.

Table (3.9). Illustrates the WOA parameters

Parameters	Value
Iterations	1000
Population Size	200
Spiral update positions	1
p	Probability is a random number between [0,1]
Search Space	Ub & Lb

The Steps to Implement the Whale Optimization Algorithm:

1. Randomly position a population of whales to begin and define the objective function to be optimized.
2. Evaluate the fitness of each whale using the objective function using Eq.(2.7).
3. Identify the population solution with the highest fitness and update the global optimal solution and fitness.
4. Update the positions of whales based on their current positions and the global best solution.
5. Use different equations for different types of whales.
6. Simulate the encircle prey behavior of whales to explore the solution space.
7. Simulate the bubble-net hunting behavior of whales to exploit promising regions.

8. Check if a termination condition is met, If not converged, go back to step 2. Otherwise, proceed to the next step.
9. Return the global best solution as the optimized solution.

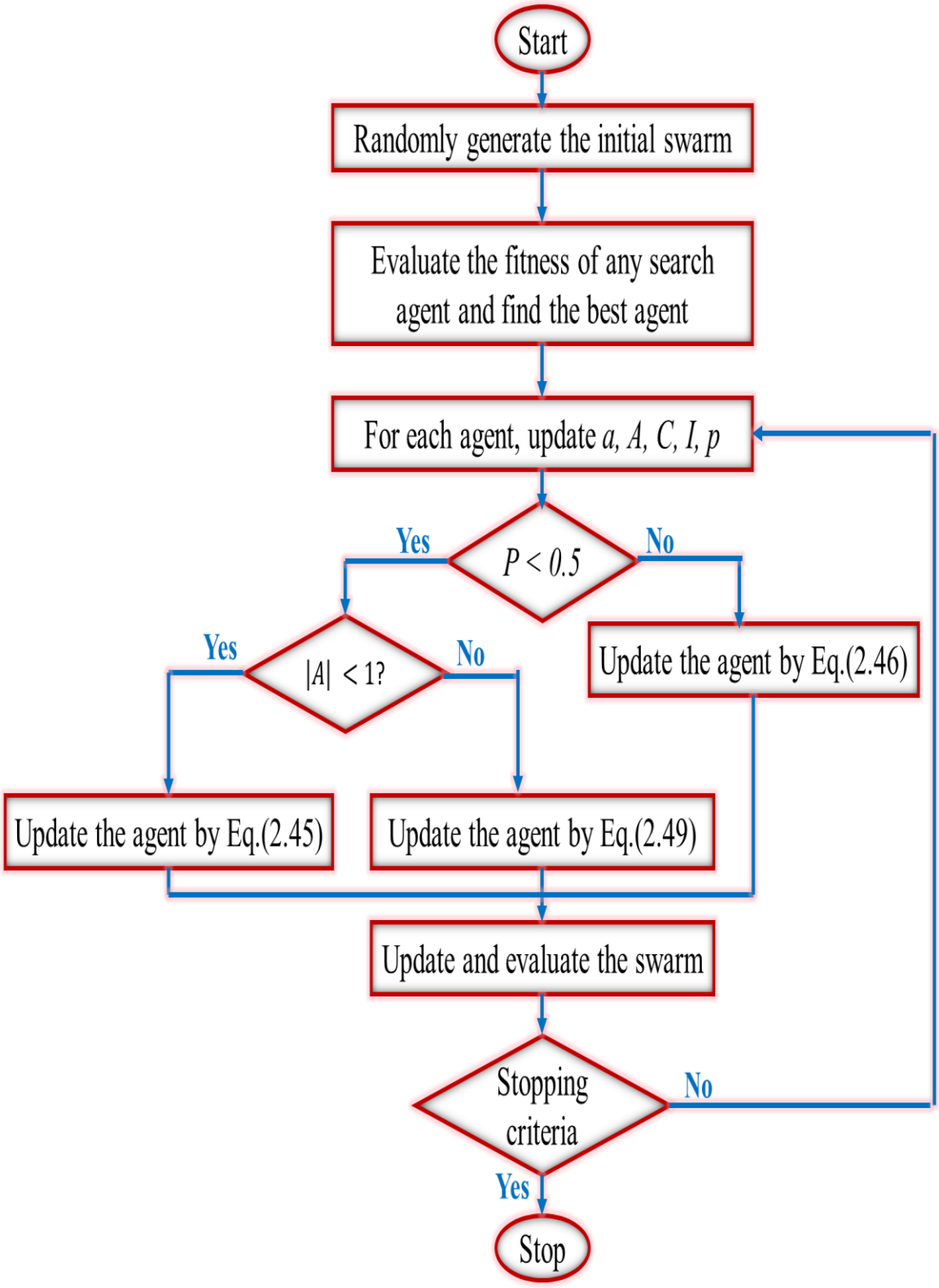


Figure (3.11). Flowchart of WOA.

3.4.10 Steps and Parameters Based on ALO Algorithm

The parameters are shown in Table (3.10). may be variables, and the extent of this change is intended to reduce SLL; for example, the number of iterations, when its value is altered, will have an effect on decreased SLL, as in population size. Fig (3.12). displays the steps taken to reduce SLL.

Table (3.10). Illustrates the ALO parameters

Parameters	Value
Iterations	1000
Population Size	200
Search Space	Ub & Lb

The Steps to Implement the Ant Lion Optimization Algorithm:

1. Initialize a population of ants and ant lions with random positions and define the objective function to be optimized.
2. Evaluate the fitness of each ant and ant lion using the objective function using Eq.(2.7).
3. Identify the population solution with the highest fitness and update the global optimal solution and fitness.
4. Allow ants to move towards better solutions using a probabilistic movement strategy.
5. Update the levels of the ant lions based on the fitness of ants in their vicinity and levels guide ant lions toward promising areas.
6. Move ant lions according to their levels, encouraging them to explore areas with higher ant activity.
7. Check if a termination condition is met, If not converged, go back to step 2. Otherwise, proceed to the next step.
8. Return the position of the best ant lion as the optimized solution.

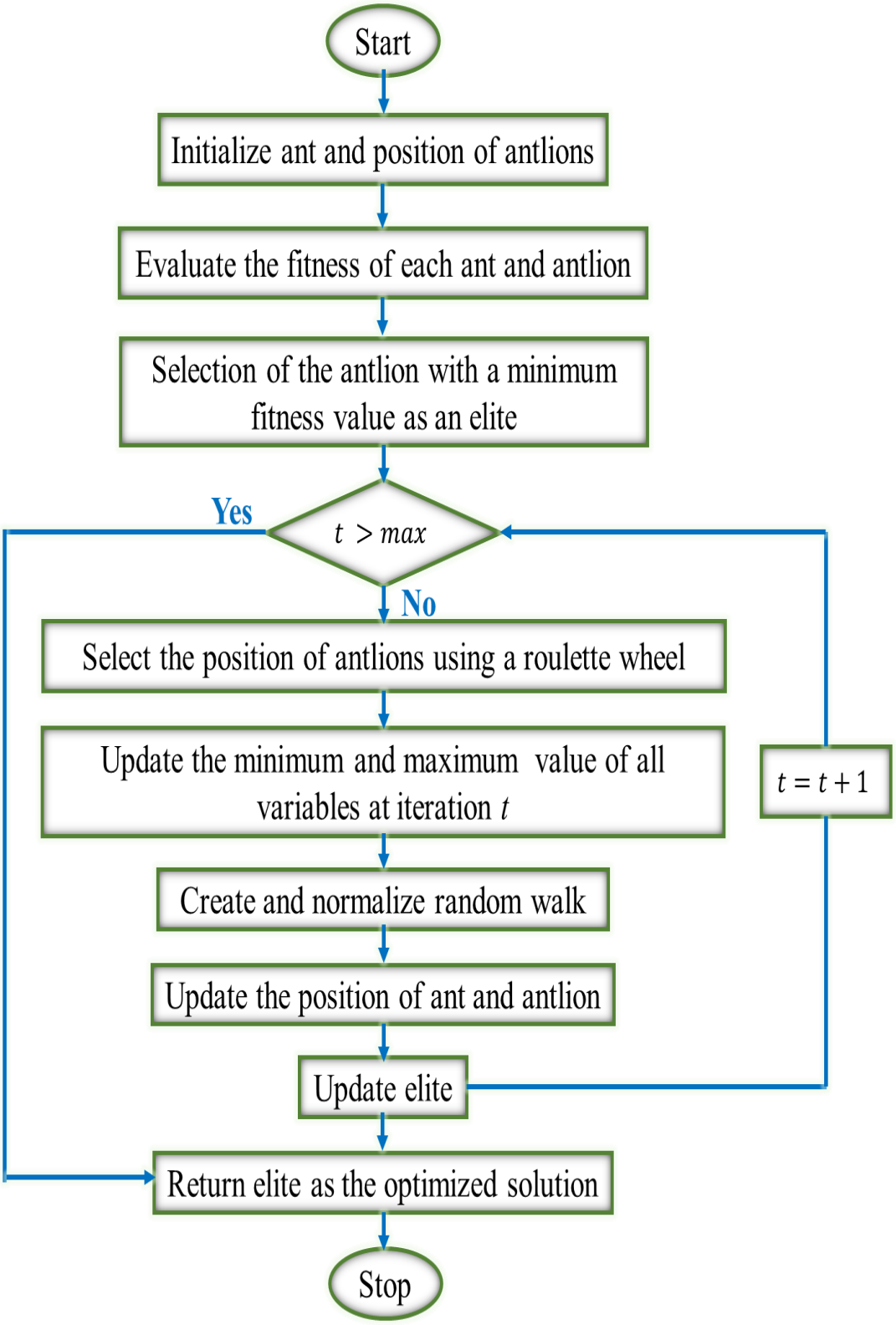


Figure (3.12). Flowchart of ALO algorithm.

All the previous algorithms will be discussed in the results will be shown in the next chapter.

4

CHAPTER FOUR

'RESULTS AND DISCUSSION'

Chapter Four

Results and Discussion

4.1 Introduction

There are many optimization algorithms, that are investigated in this work to minimize the SLL. These algorithms were used with the LAA beamformer to get the best radiation pattern with different element numbers of antenna N=8,16,32,64,128, and 256.

In addition to studying antenna positions before and after optimization and studying the impact of parameters on the reduction of SLL for each algorithm separately, using the software packages (MATLAB).

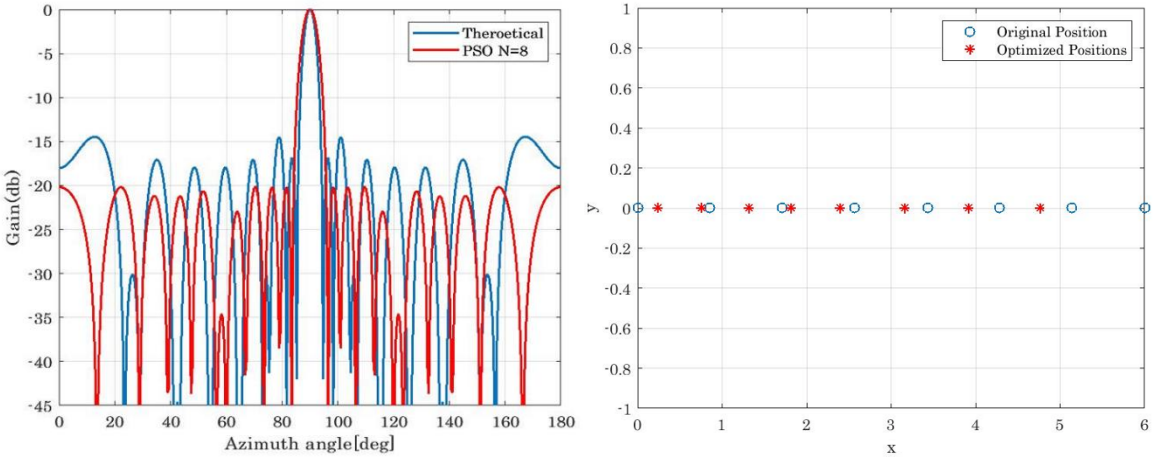
4.2 Reduction SLL for Linear Antenna Arrays Using Optimization Techniques

Algorithms have been tested for use with a beam LAA to reduce SLL and this is achieved by using an objective function. Reducing SLL to the maximum will change the distribution position and the amplitude of the elements in each element of the antenna array.

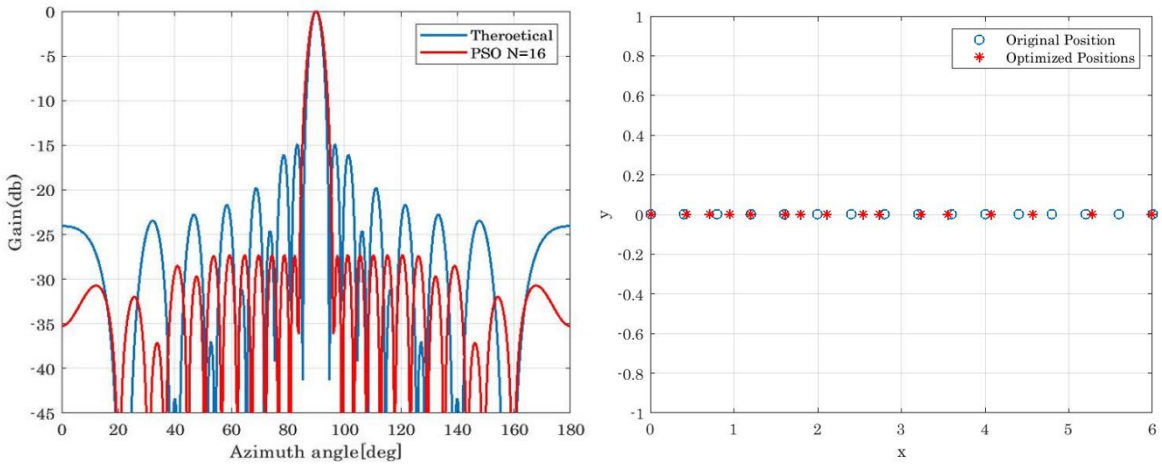
4.2.1 Results of Particles Swarm Optimization Algorithm

Based on the steps in section (3.4.1) the results were reached in Fig (4.1). (a, b, c, d, e, and f). Shows positions change for each antenna element, the amount of attrition of SLL is shown at each number of antenna elements where at N = 8, SLL reduced its theoretical value from -16.8470 dB to -20.1984 dB. At N = 16, SLL decreased significantly by -27.2992 dB while the theoretical beam pattern was -14.8815 dB. SLL decreased from -14.0977 dB to -27.3525 dB at N = 32. at N = 64, SLL decreased from -13.6939 dB to -27.8289 dB.

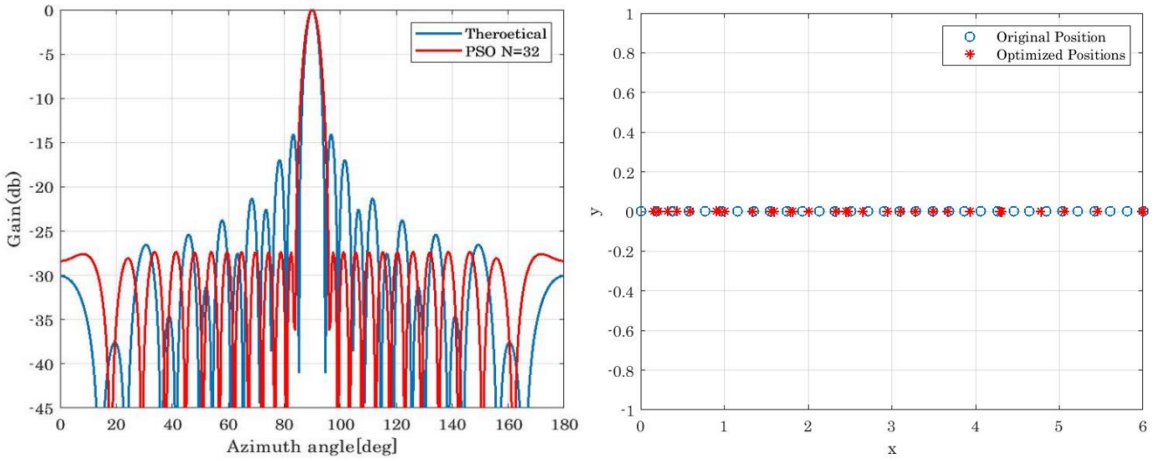
At N = 128 SLL decreased from a theoretical beam of -13.4779 dB to -28.3277 dB. The best decrease was obtained at N = 256 where SLL reduced to -28.5405 dB.



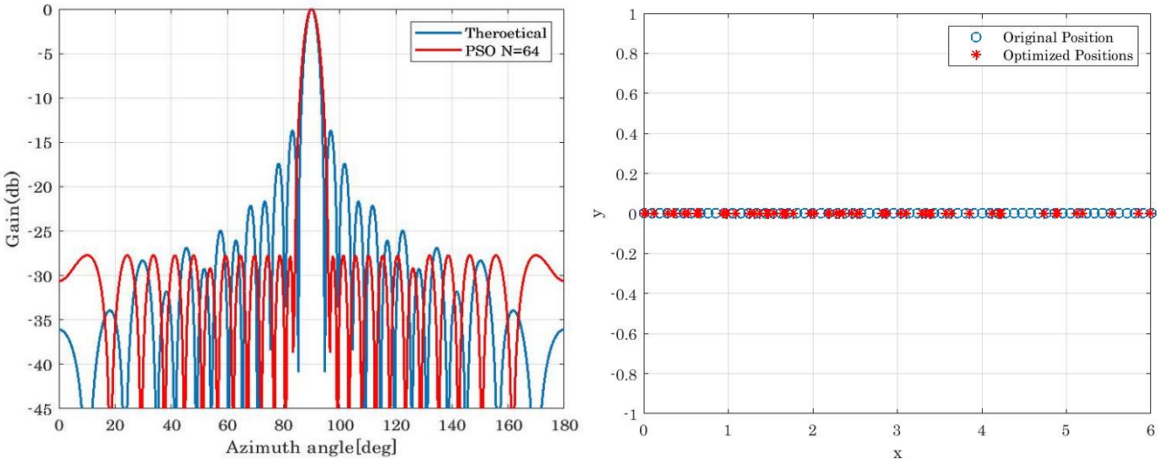
a. N =8 Elements



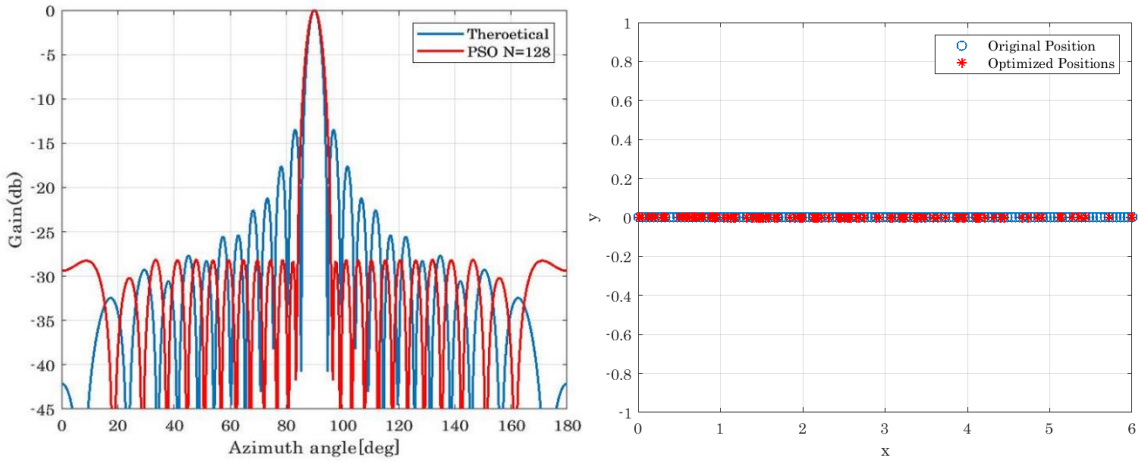
b. N =16 Elements



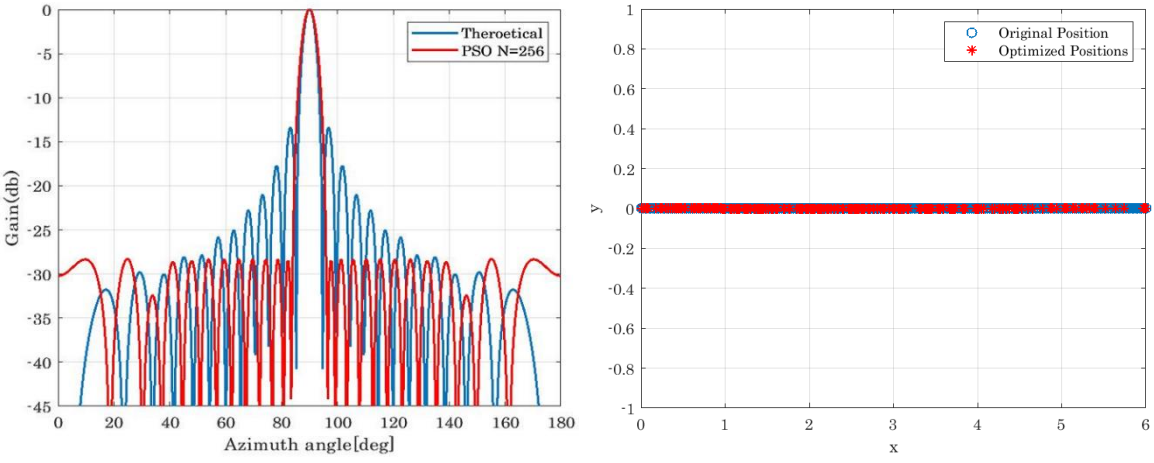
c. N =32 Elements



d. N =64 Elements



e. N =128 Elements



f. N =256 Elements

Figure (4.1). PSO with different numbers of elements

Note that the higher number of antennas leads to a better main lobe and less SLL, leading to less inter symbol interference as shown in Fig (4.2) shows all cases of PSO to reduced SLL.

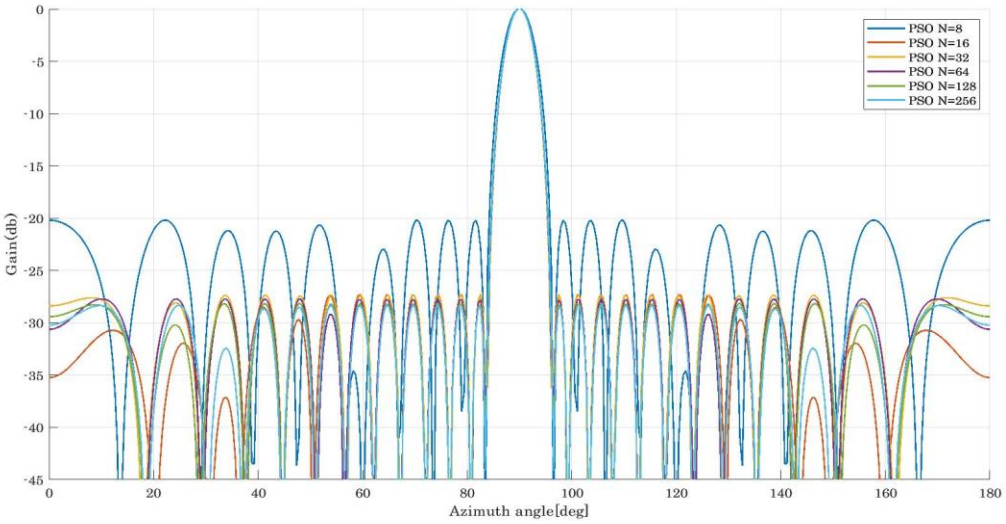
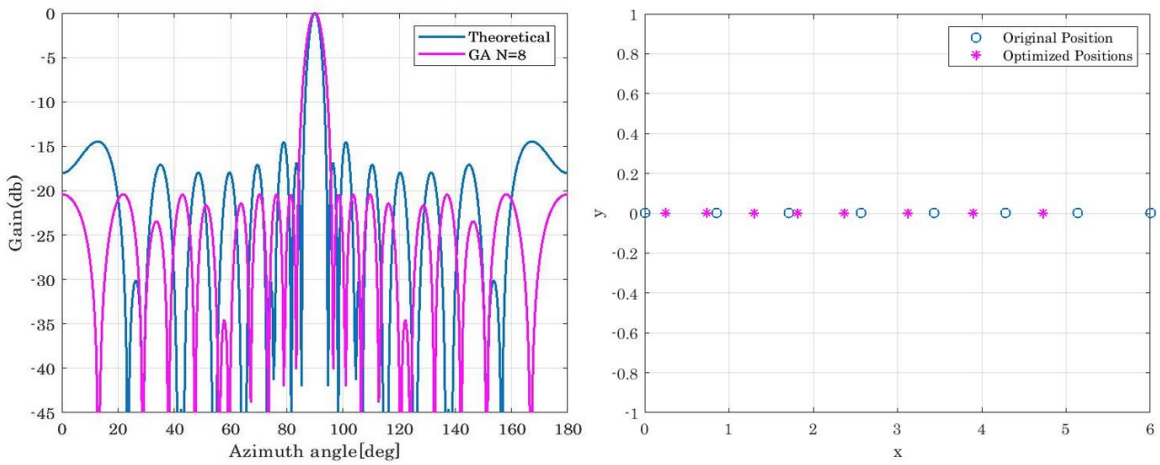


Figure (4.2). PSO comparison by the number of elements

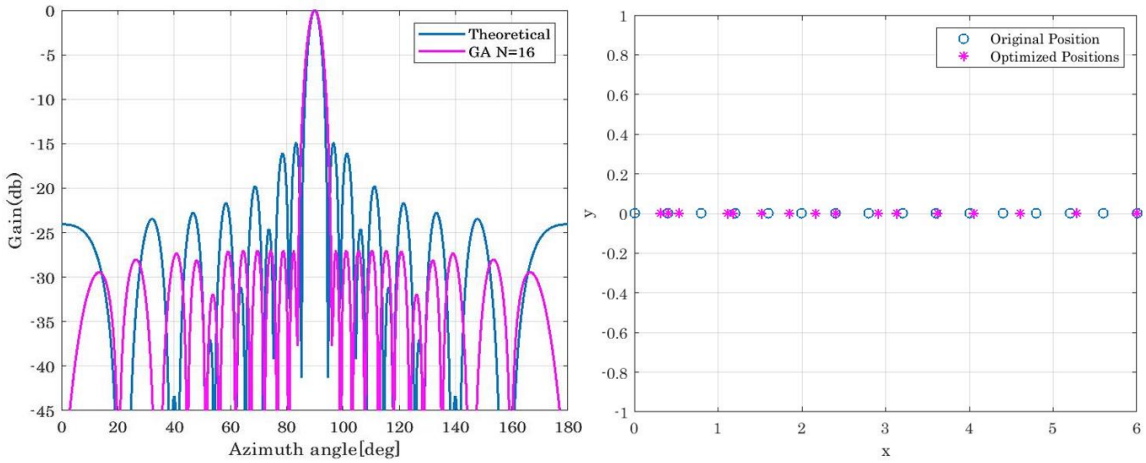
4.2.2 Results of Genetic Algorithm

Based on the steps in section (3.4.2) the results were reached when using the GA algorithm to minimize SLL as shown in Fig (4.3). (a, b, c, d, e, and f). Which shows positions change for each number of antenna elements, the amount of attrition of SLL is shown at N = 8 SLL reduced its theoretical value from -16.8470 dB to -20.4335 dB. At N = 16, SLL decreased significantly by -26.9987 dB while the theoretical beam pattern was -14.8815 dB. SLL decreased from -14.0977 dB to -27.8764 dB at N = 32.

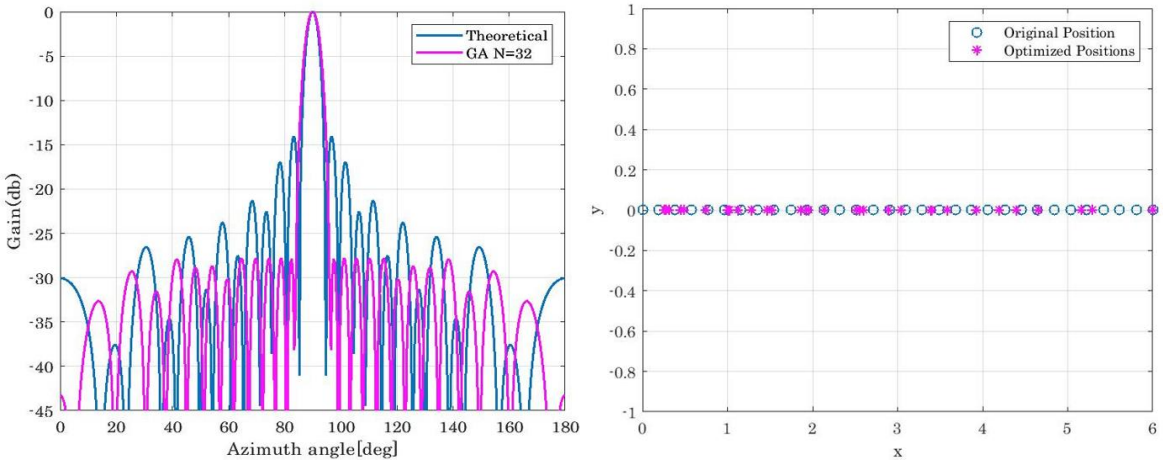
At N = 64, SLL decreased from -13.6939 dB to -28.0044 dB, at N = 128 SLL decreased from a theoretical beam of -13.4779 dB to -28.5568 dB. The best decrease was obtained at N = 256 where SLL reduced to -28.6204 dB.



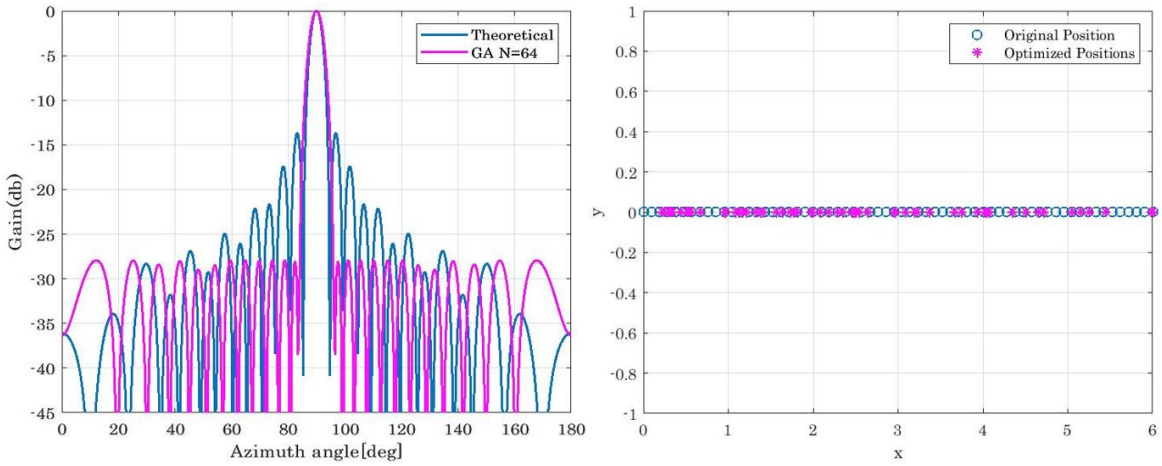
a. N =8 Elements



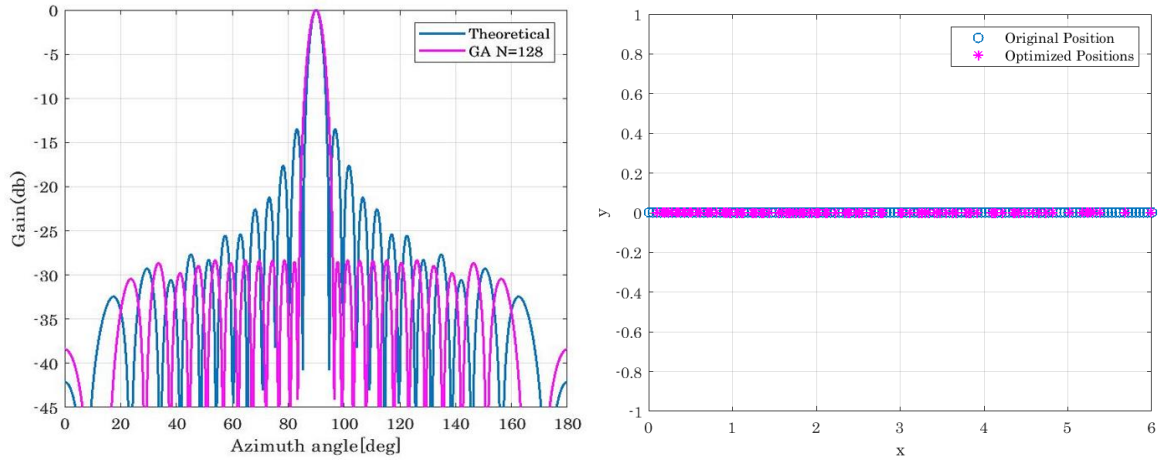
b. N =16 Elements



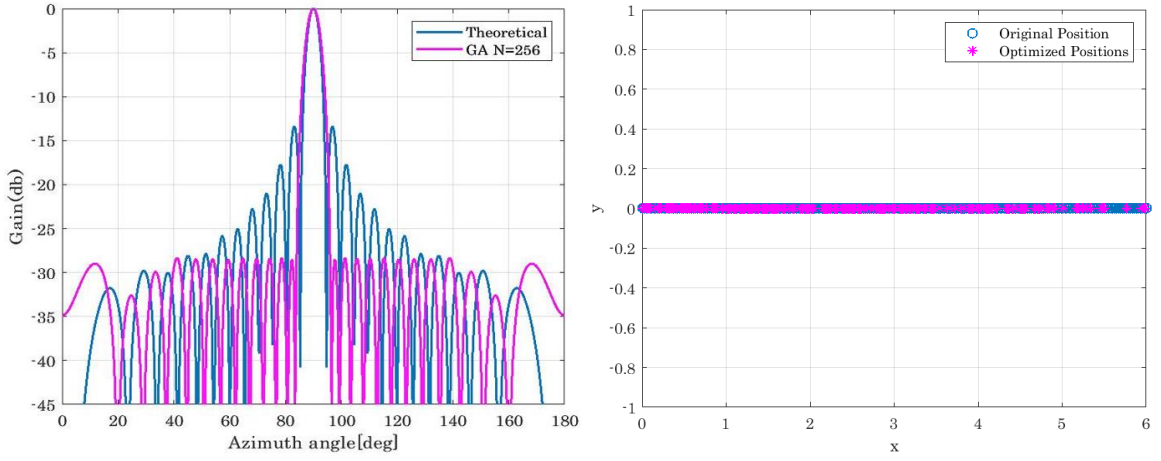
c. N =32 Elements



d. N =64 Elements



e. N =128 Elements



f. N =256 Elements

Figure (4.3). GA with different numbers of elements

Note that the higher number of antennas leads to a better main lobe and less SLL, leading to less inter symbol interference as shown in Fig (4.4) shows all cases of GA to reduced SLL. GA is better than PSO at its best value, at N = 256.

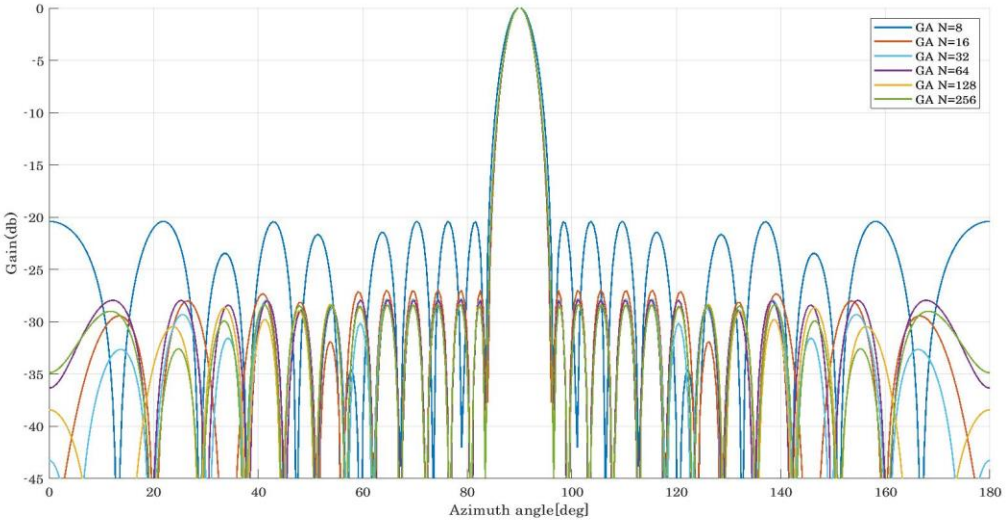
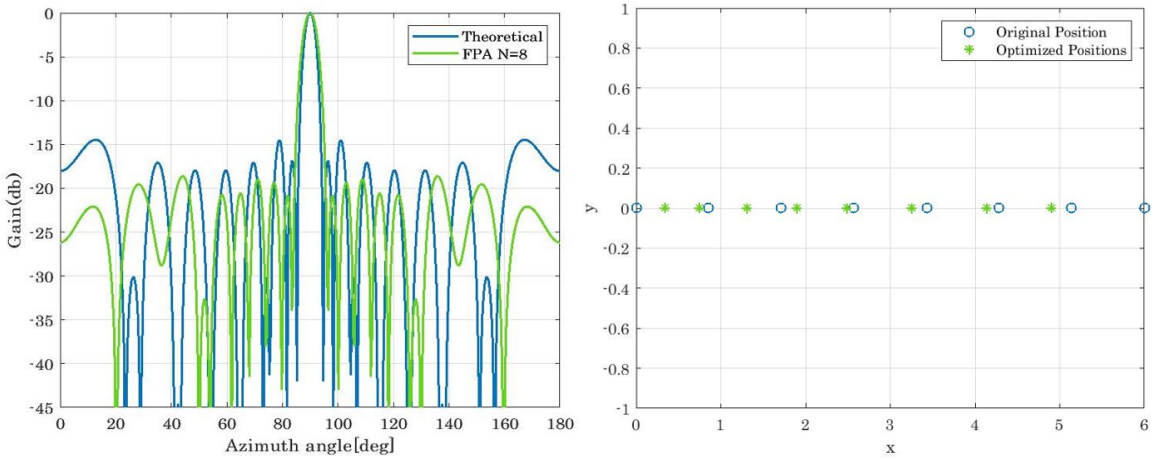


Figure (4.4). GA comparison by the number of elements.

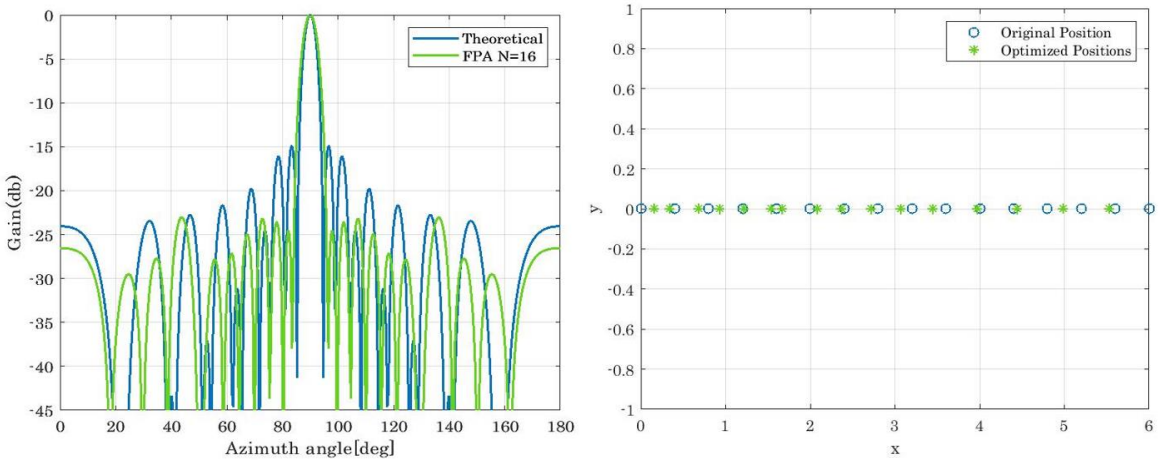
4.2.3 Results of Flower Pollination Algorithm

Based on the steps in section (3.4.3) the results were reached in Fig (4.5). (a, b, c, d, e, and d). Displays a change in positions for each number of antenna elements, the amount of attrition of SLL is shown where at N = 8, SLL reduced its theoretical value from -16.8470 dB to -20.8492 dB. At N = 16, SLL decreased significantly by -24.5472 dB while the theoretical beam pattern was -14.8815 dB. SLL decreased from -14.0977 dB to -28.3071 dB at N = 32, it is the best reduction obtained when using FPA despite the use of more antenna elements but it did not give a better result than at N = 32.

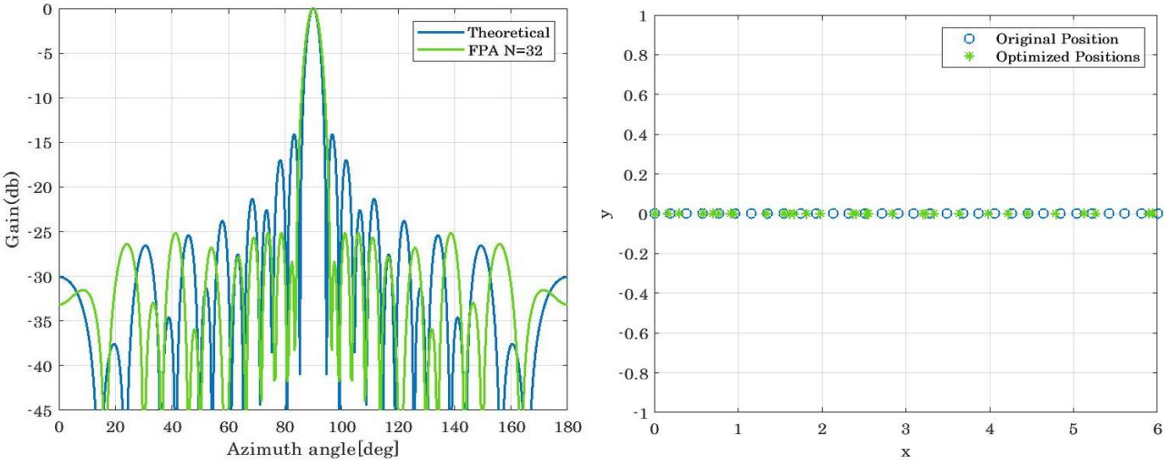
At N = 64, SLL decreased from -13.6939 dB to -28.0148 dB. At N = 128 SLL decreased from a theoretical beam of -13.4779 dB to -26.4663 dB. The decrease was obtained at N = 256 where SLL reduced to -23.0646 dB.



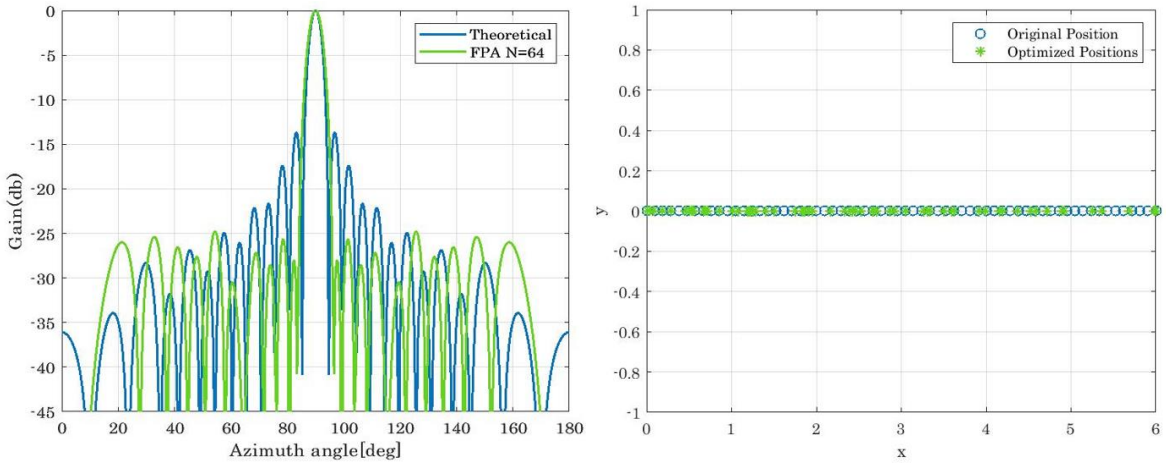
a. N =8 Elements



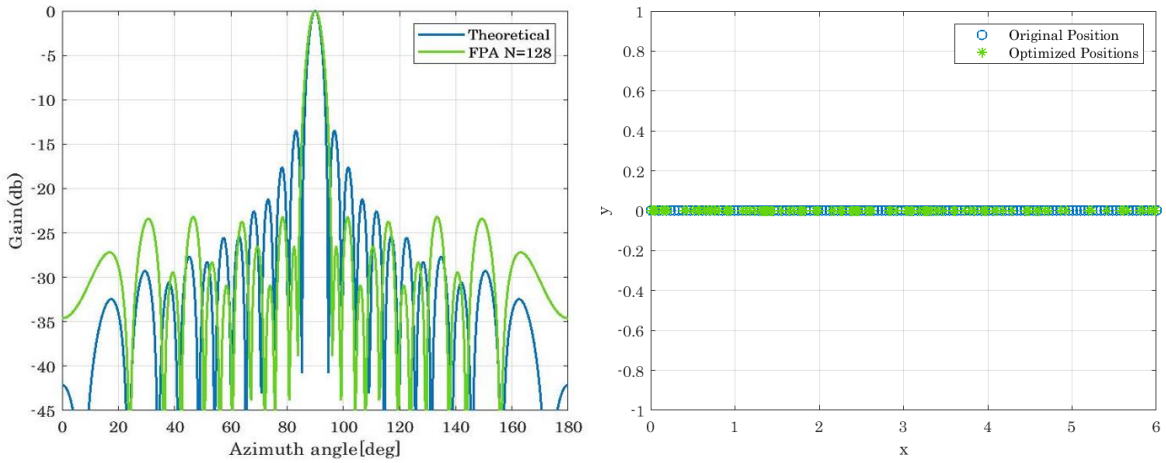
b. N =16 Elements



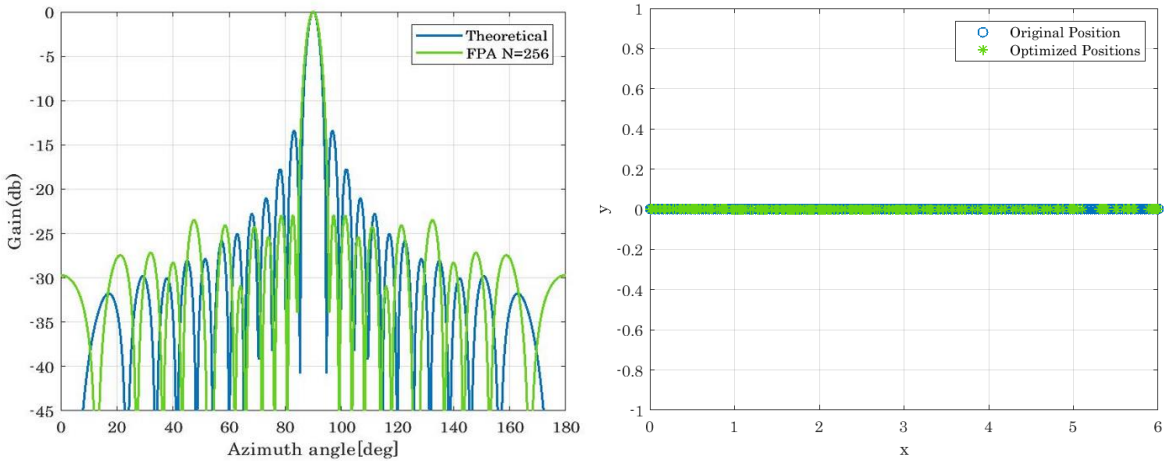
c. N =32 Elements



d. N =64 Elements



e. N =128 Elements



f. N =256 Elements

Figure (4.5). FPA with different numbers of elements

Note that the higher number of antennas leads to a better main lobe and less SLL, leading to less inter symbol interference as shown in Fig (4.6) shows all cases of FPA. Note that FPA is better than PSO and GA at N = 32 where SLL has been minimized but GA is still superior at N = 256.

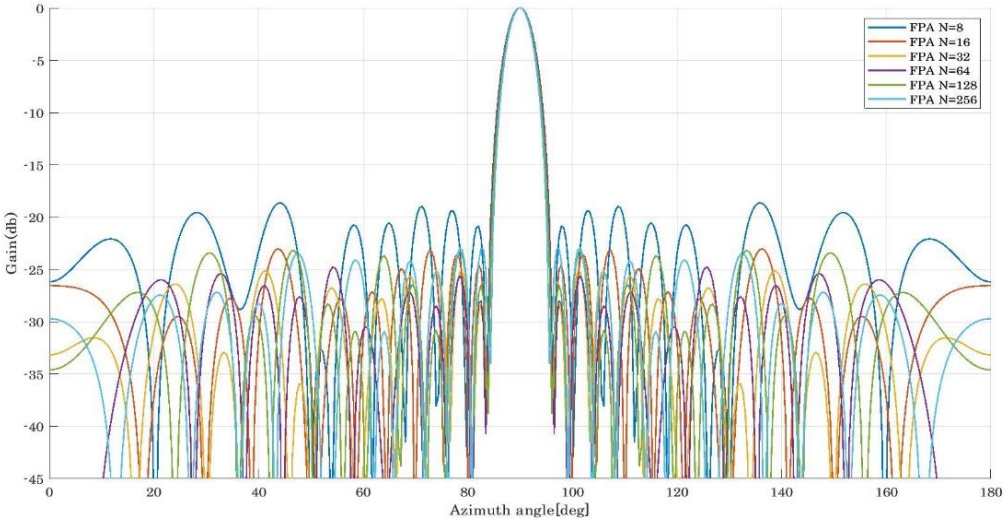
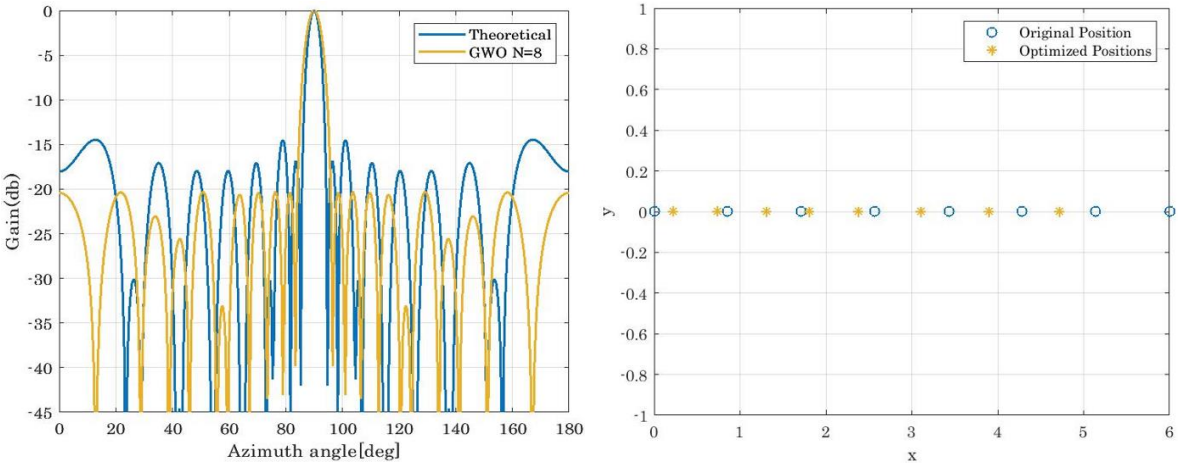


Figure (4.6). FPA comparison by the number of elements.

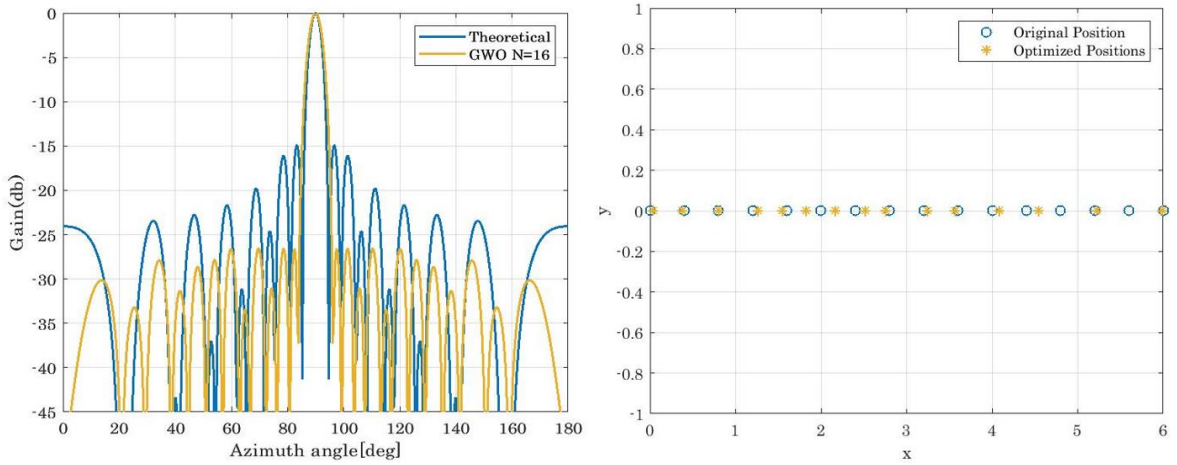
4.2.4 Results of Grey Wolf Optimization Algorithm

Based on the steps in section (3.4.4) the results were reached in Fig (4.7). (a, b, c, d, e, and f). Shows positions change for each antenna element and each number of antenna elements displays the quantity of SLL attrition where at N = 8, SLL reduced its theoretical value from -16.8470 dB to -20.3723 dB where SLL reduced. At N = 16, SLL decreased significantly by -26.5385 dB while the theoretical beam pattern was -14.8815 dB. SLL decreased from -14.0977 dB to -27.3854 dB at N = 32. At N = 64, SLL decreased from -13.6939 dB to -27.7335 dB.

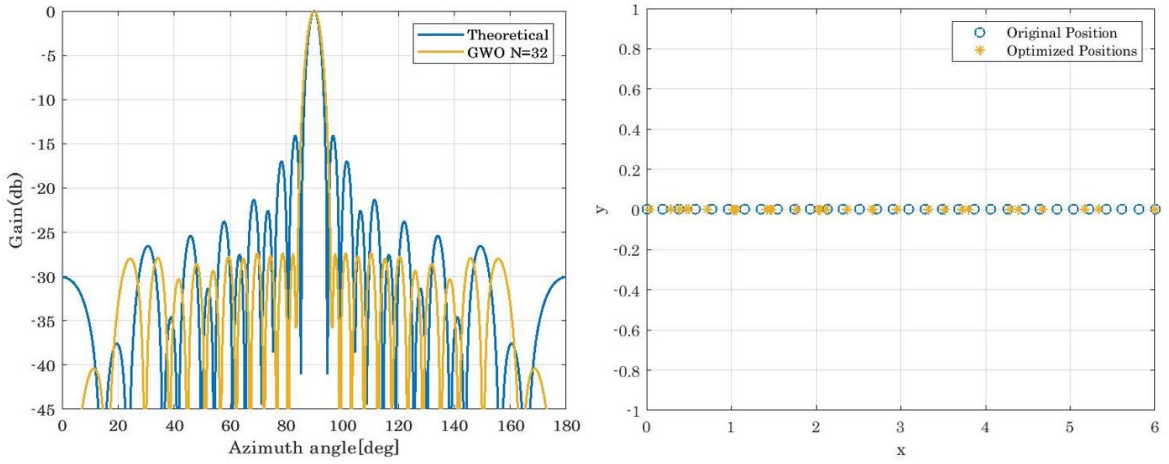
At N = 128 SLL decreased from a theoretical beam of -13.4779 dB to -28.2732 dB. While the decrease was obtained at N = 256 where SLL reduced to -28.3367 dB, it is the best reduction obtained when using GWO, However, GA remains preferable to other algorithms PSO, FPA, and GWO.



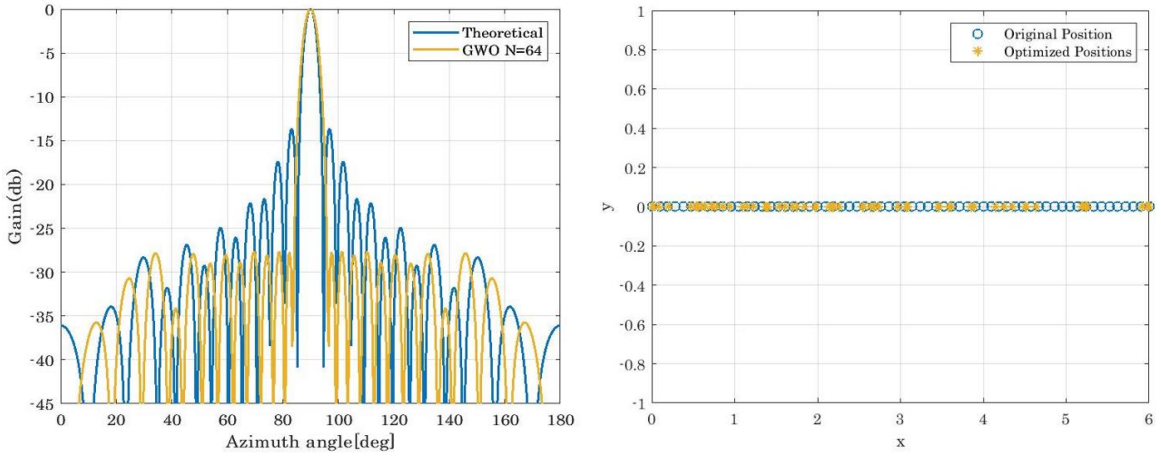
a. N =8 Elements



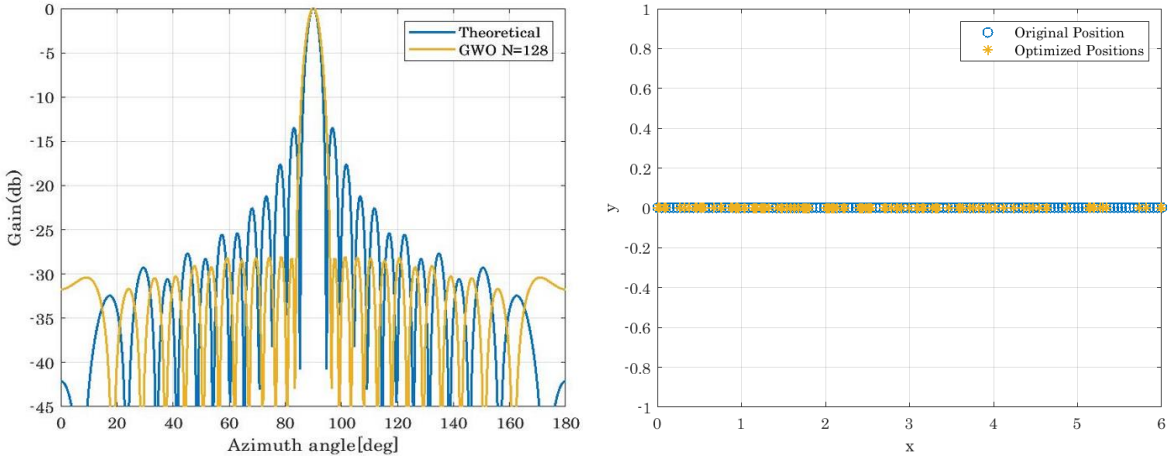
b. N =16 Elements



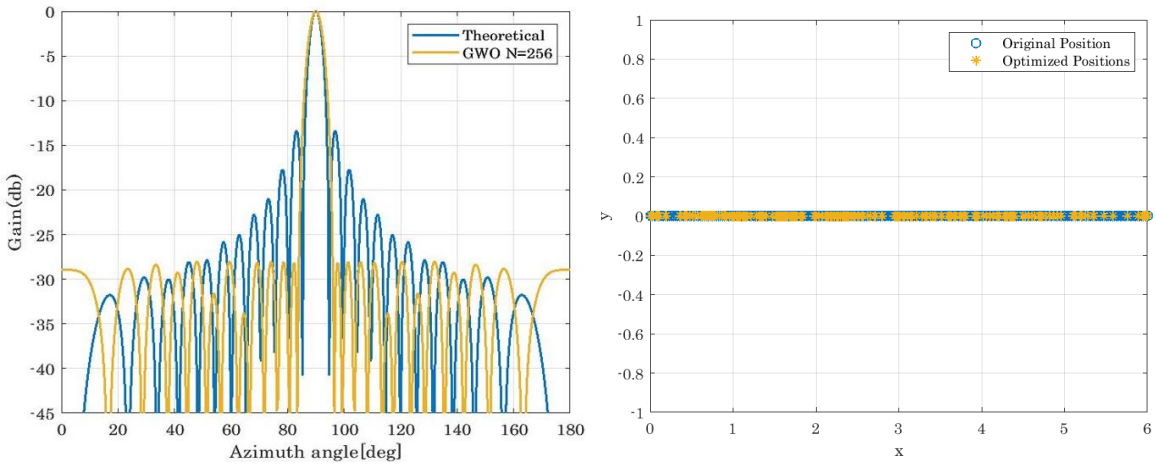
c. N =32 Elements



d. N =64 Elements



e. N =128 Elements



f. N =256 Elements

Figure (4.7). GWO with different numbers of elements

Note that the higher number of antennas leads to a better main lobe and less SLL, leading to less inter symbol interference as shown in Fig (4.8) shows all cases of GWO.

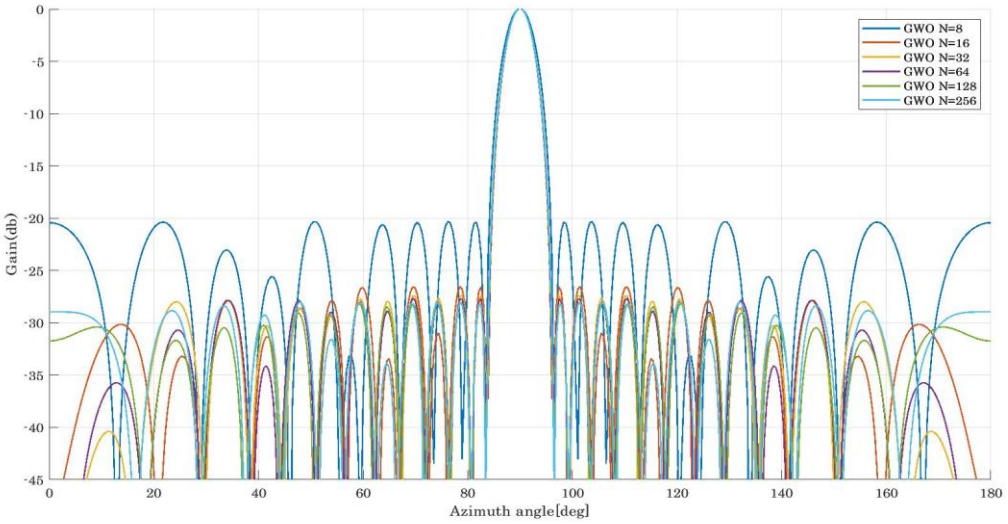
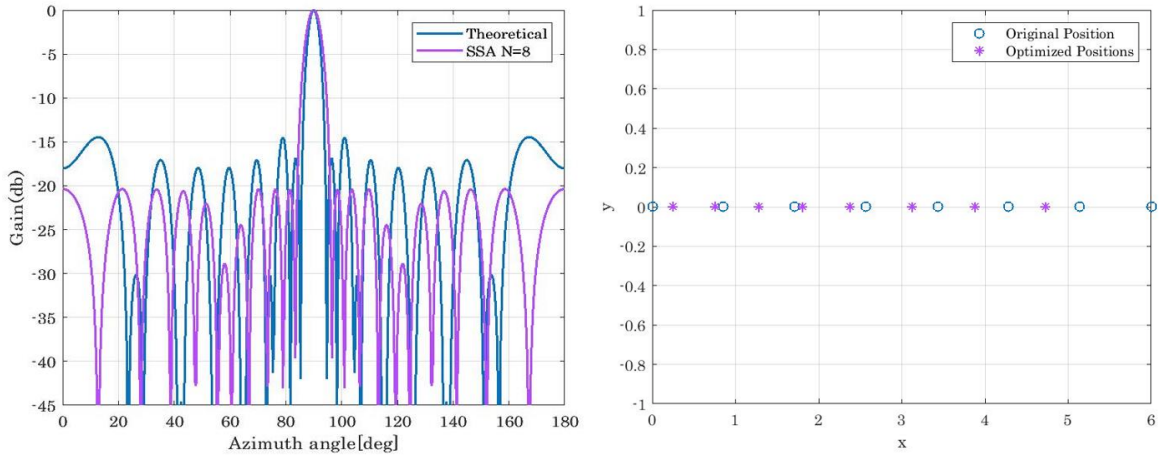


Figure (4.8). GWO comparison by the number of elements.

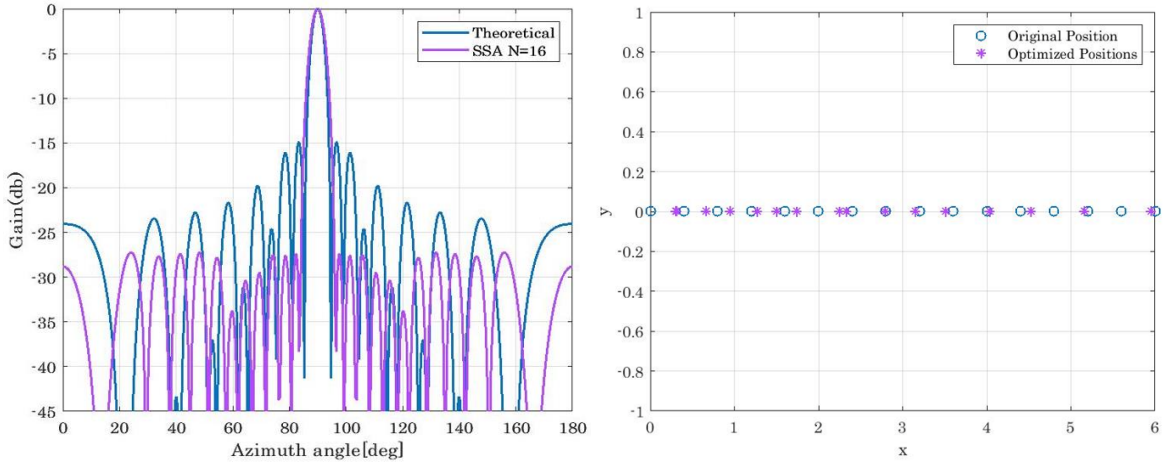
4.2.5 Results of Sparrow Search Algorithm

Based on the steps in section (3.4.5) the results were reached in Fig (4.9). (a, b, c, d, e, and f). Shows positions change for each antenna element and each number of antenna elements indicates the SLL attrition rate where at N = 8, SLL reduced its theoretical value from -16.8470 dB to -20.4192 dB. At N = 16, SLL decreased significantly by -27.4011 dB while the theoretical beam pattern was -14.8815 dB. SLL decreased from -14.0977 dB to -27.7574 dB at N = 32.

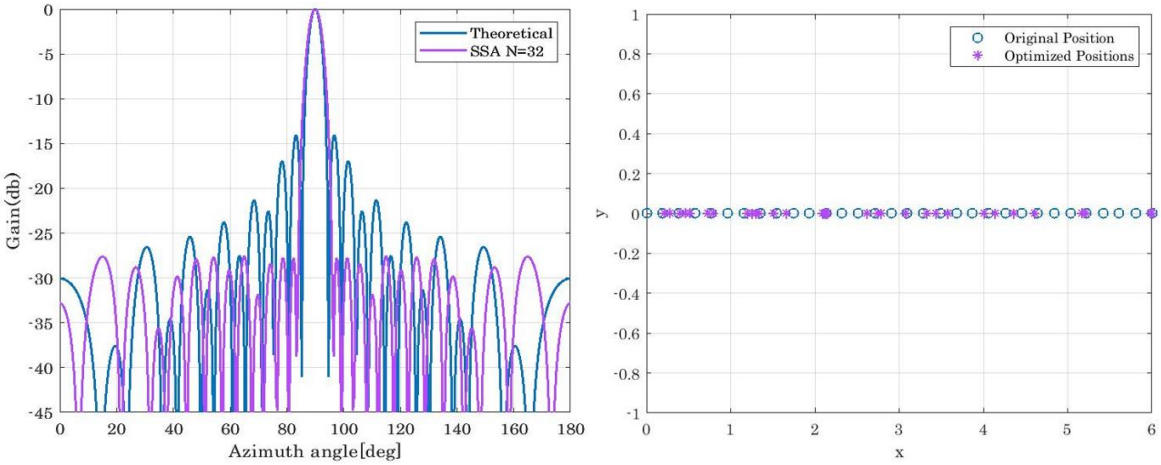
At N = 64, SLL decreased from -13.6939 dB to -28.1870 dB. At N = 128 SLL decreased from a theoretical beam of -13.4779 dB to -28.2192 dB. While the decrease was obtained at N = 256 where SLL reduced to -28.3453 dB, it is the best reduction obtained when using SSA, However, GA remains superior to PSO, FPA, GWO, and SSA.



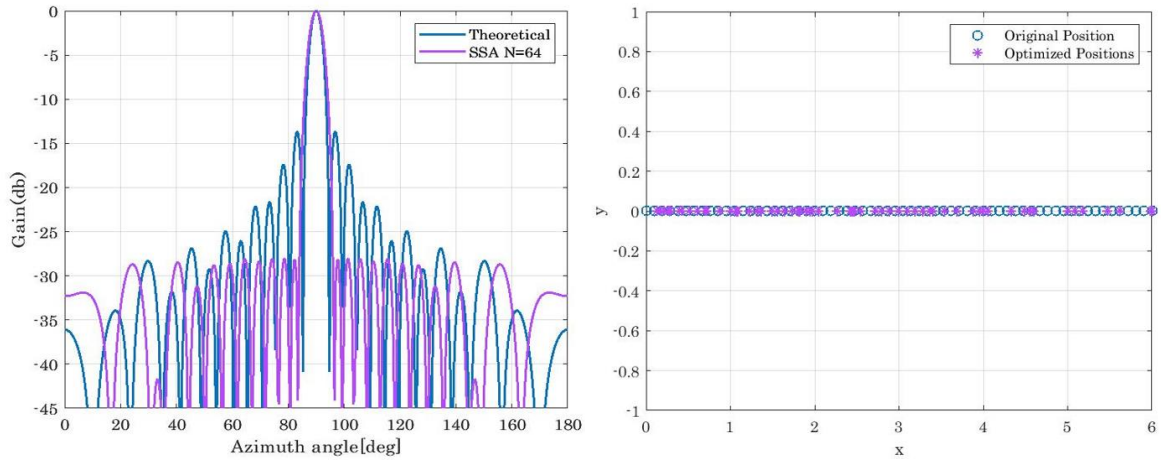
a. N =8 Elements



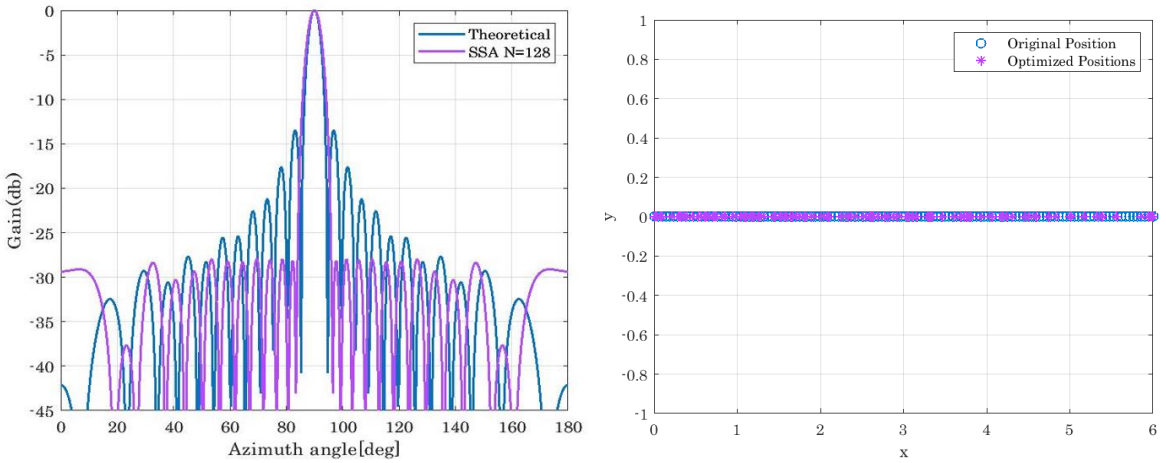
b. N =16 Elements



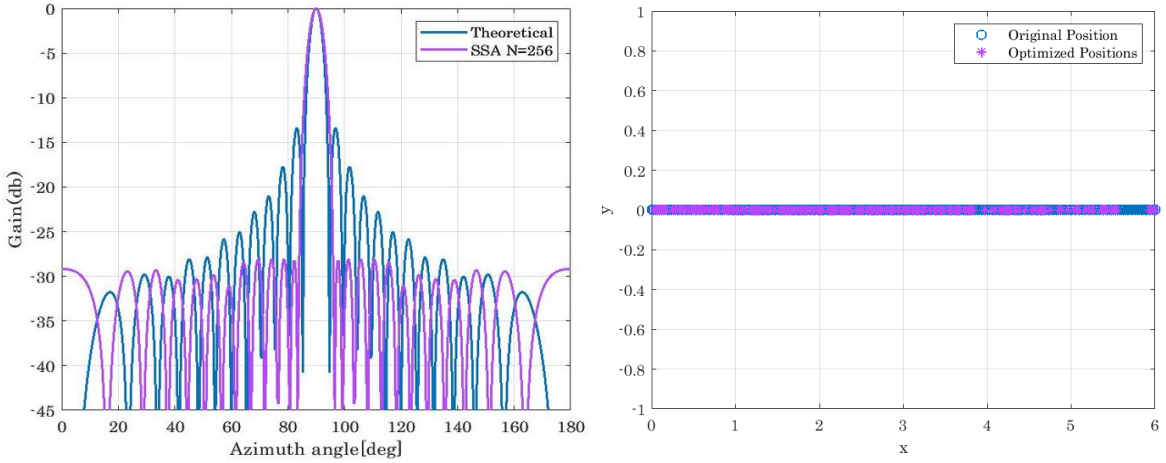
c. N =32 Elements



d. N =64 Elements



e. N =128 Elements



f. N =256 Elements

Figure (4.9). SSA with different numbers of elements

Note that the higher number of antennas leads to a better main lobe and less SLL, leading to less inter-symbol interference as shown in Fig (4.10) illustrates all SSA instances.

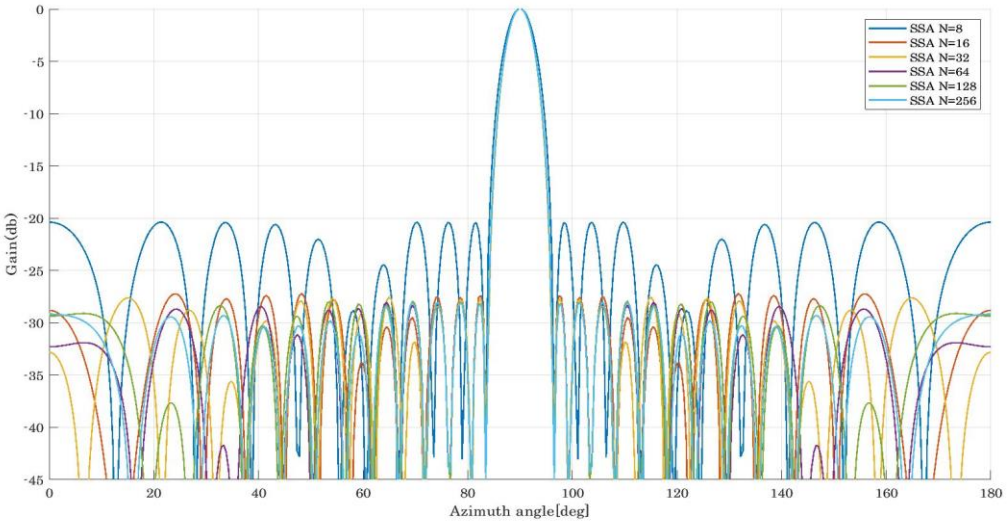
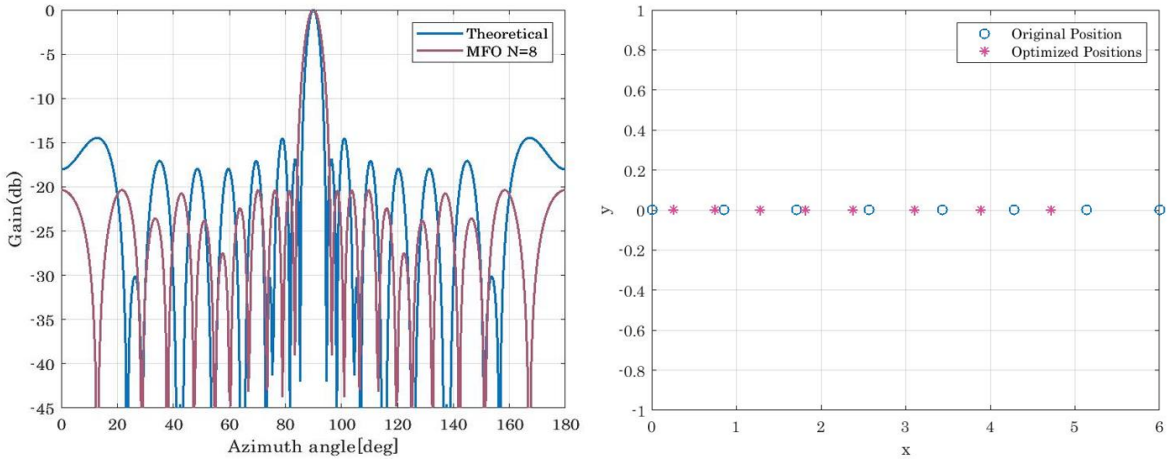


Figure (4.10). SSA comparison by the number of elements.

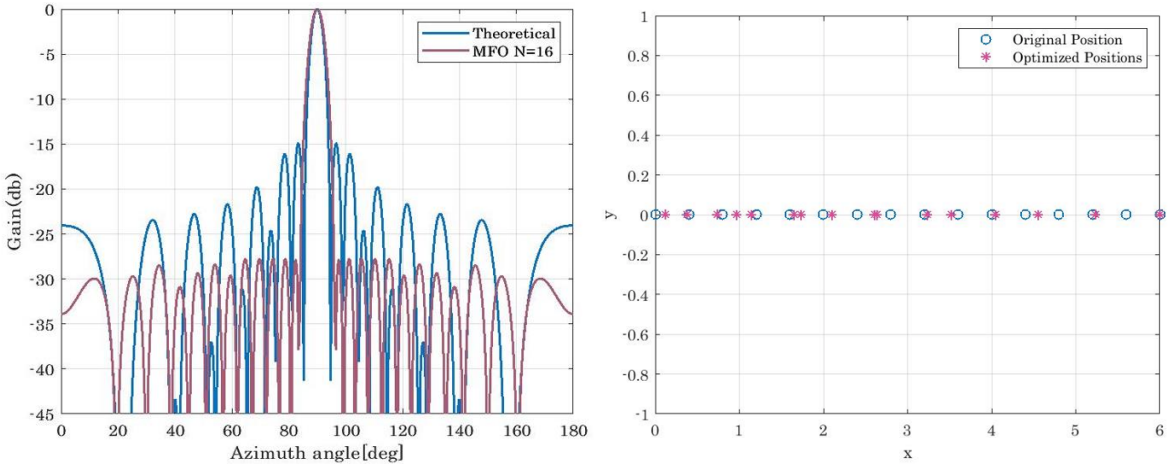
4.2.6 Results of Moth Flam Optimization Algorithm

Based on the steps in section (3.4.6) the results were reached in Fig (4.11). (a, b, c, d, e, and f). Shows positions change for each antenna element and each number of antenna elements displays the quantity of SLL attrition at N = 8, SLL will be reduced to -20.4166 dB after the beam pattern value has been -16.8470 dB. While SLL dropped to -27.8226 dB at N = 16 after the theoretical pattern -14.8815 dB. SLL decreased from -14.0977 dB to -27.9496 dB at N = 32.

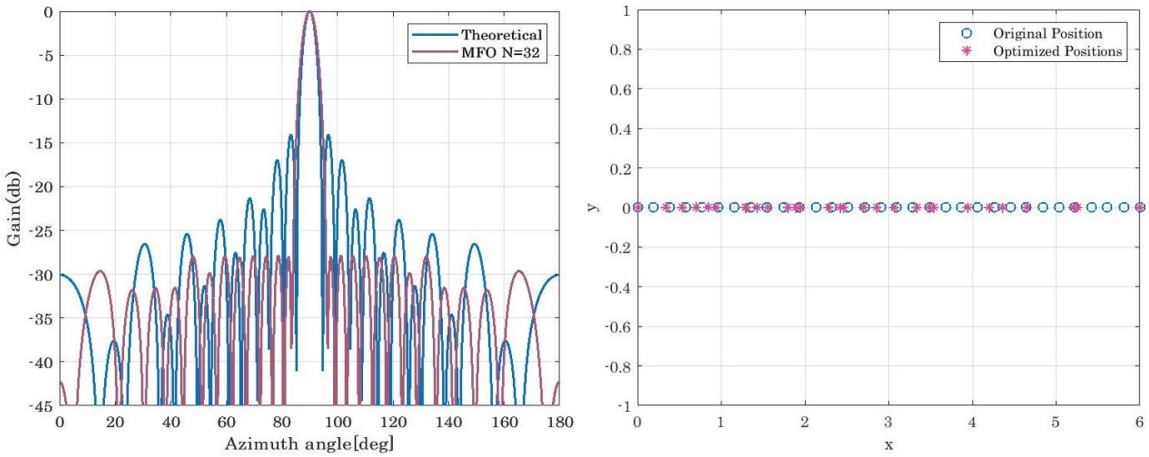
At N = 64,128, and 256, SLL dropped to -27.9725 dB, -28.3859 dB, and -24.7649 dB respectively. The best drop-in SLL when using MFO was found to have occurred at N = 128. Although by increasing the number of antenna elements we can get adverse results this is because despite the complexity of the system as shown in Fig (4.11).e.



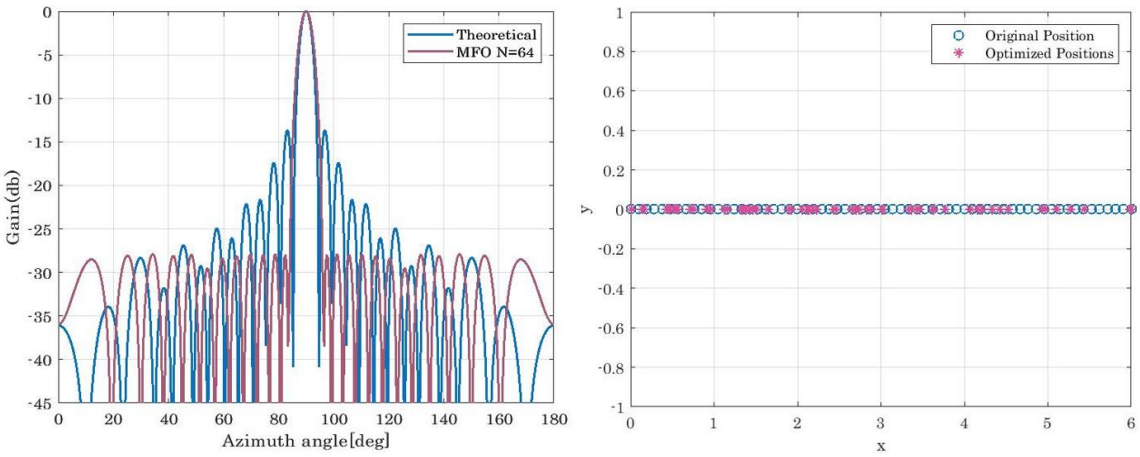
a. N =8 Elements



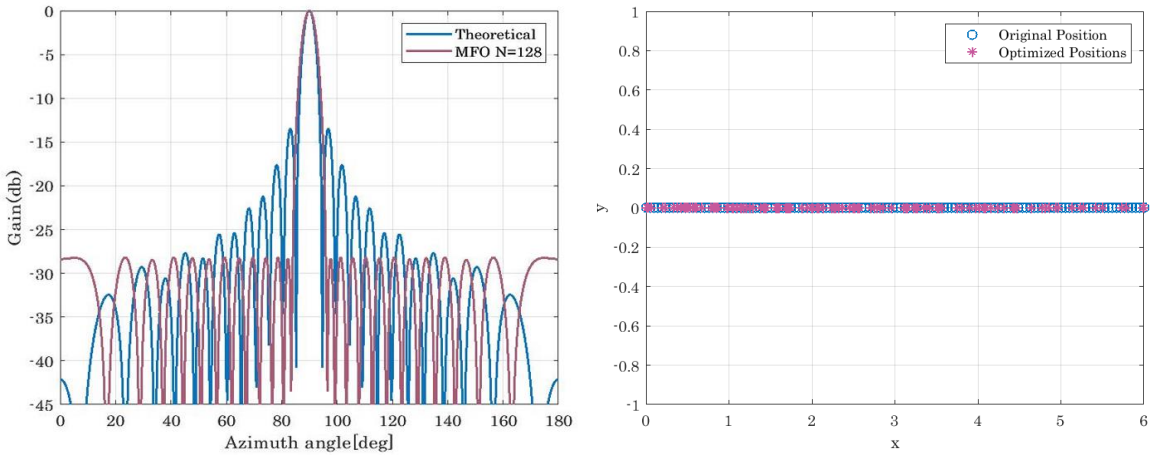
b. N =16 Elements



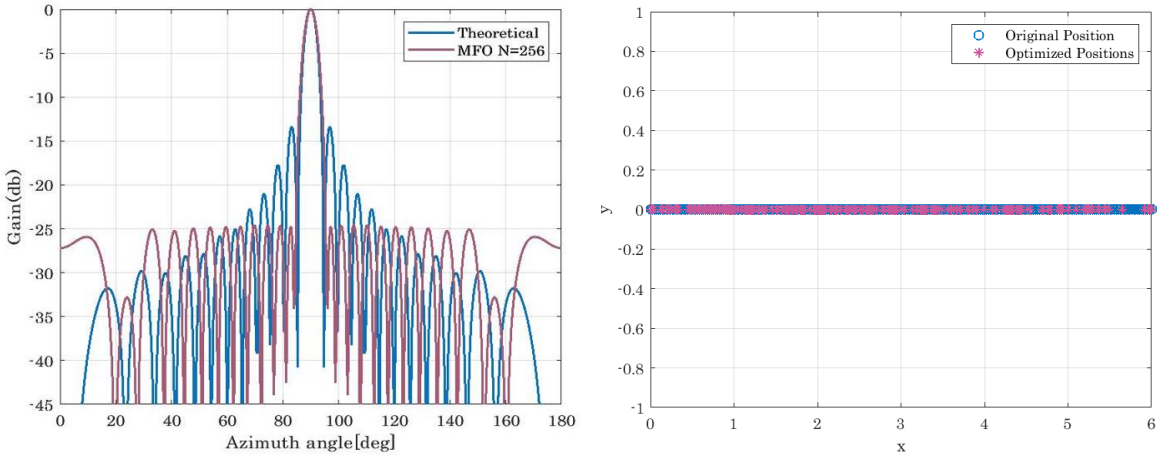
c. N =32 Elements



d. N =64 Elements



e. N =128 Elements



f. N =256 Elements

Figure (4.11). MFO with different numbers of Elements

Note that the higher number of antennas leads to a better main lobe and less SLL, leading to less inter symbol interference as shown in Fig (4.12) illustrates all MFO instances.

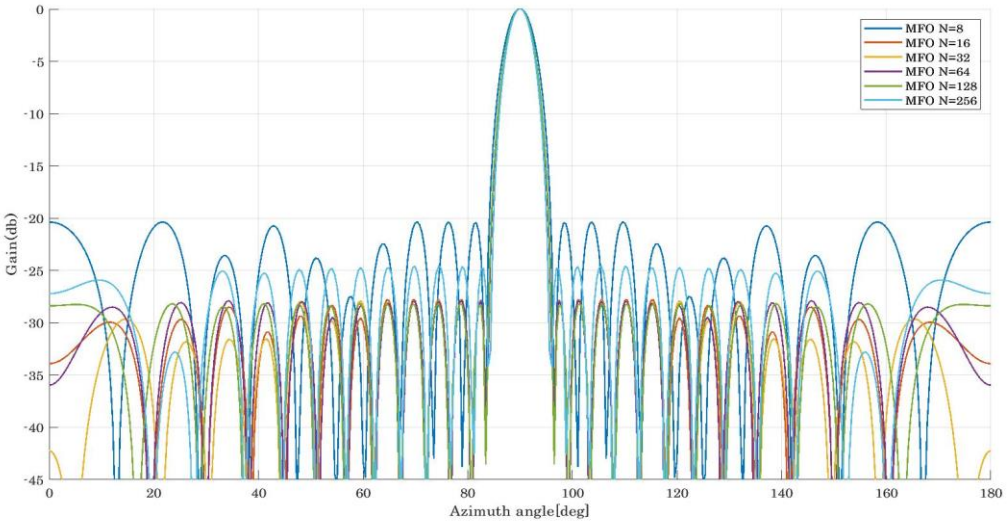
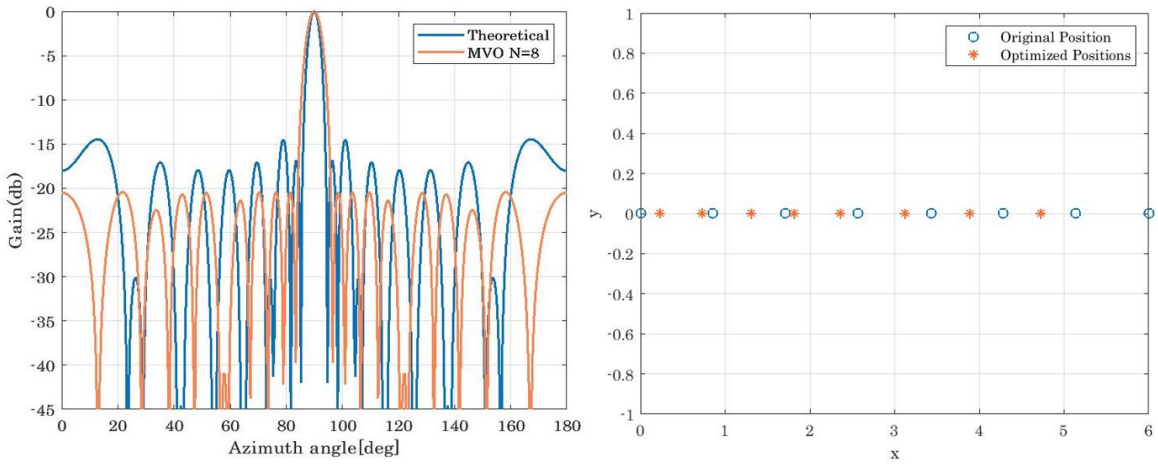


Figure (4.12). MFO comparison by the number of elements

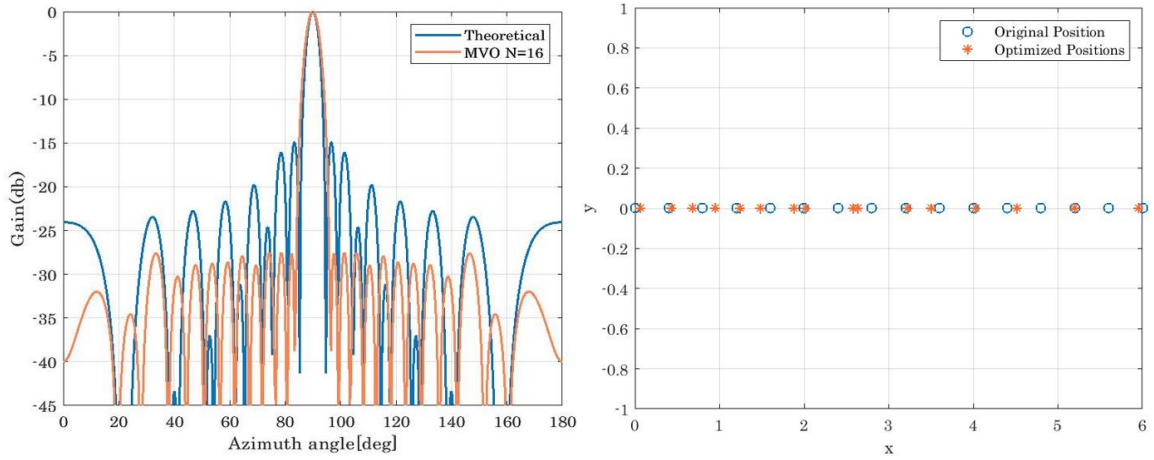
4.2.7 Results of Multi-Verse Optimization Algorithm

Based on the steps in section (3.4.7) the results were reached in Fig (4.13). (a, b, c, d, e, and f). Displays change in positions for each antenna element the amount of attrition of SLL is shown at each number of antenna element. At N = 8, SLL will be reduced to -20.4588 dB after the beam pattern value has been -16.8470 dB. While SLL dropped to -27.6097 dB. At N = 16 after the theoretical pattern -14.8815 dB. SLL decreased from -14.0977 dB to -28.0246 dB at N = 32.

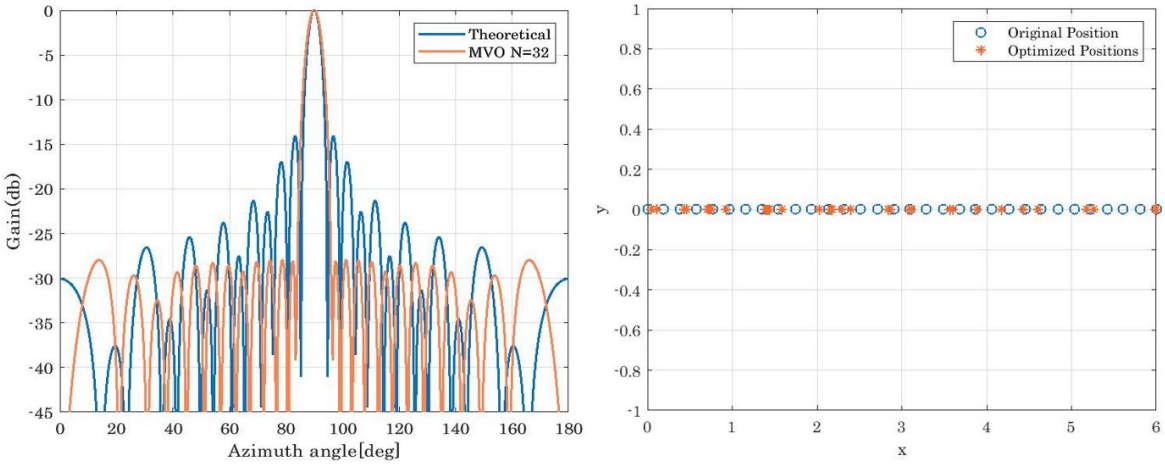
At N = 64,128, and 256, SLL dropped to -28.2811 dB, -28.4366 dB and-28.3551 dB respectively. The best reduced SLL when using MVO was found to have occurred at N = 128. although by increasing the number of antenna elements we can get adverse results this is because despite the complexity of the system as shown in Fig (4.13).e.



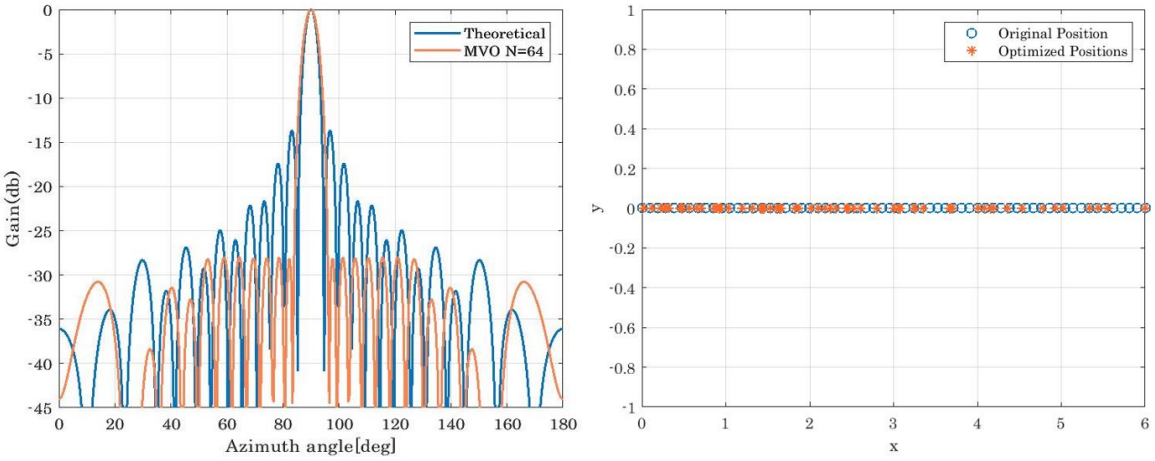
a. N =8 Elements



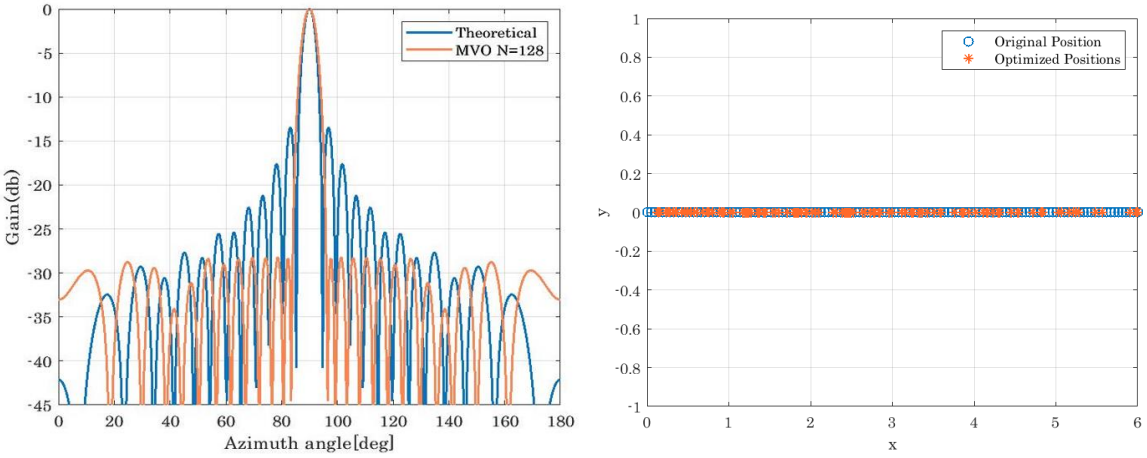
b. N =16 Elements



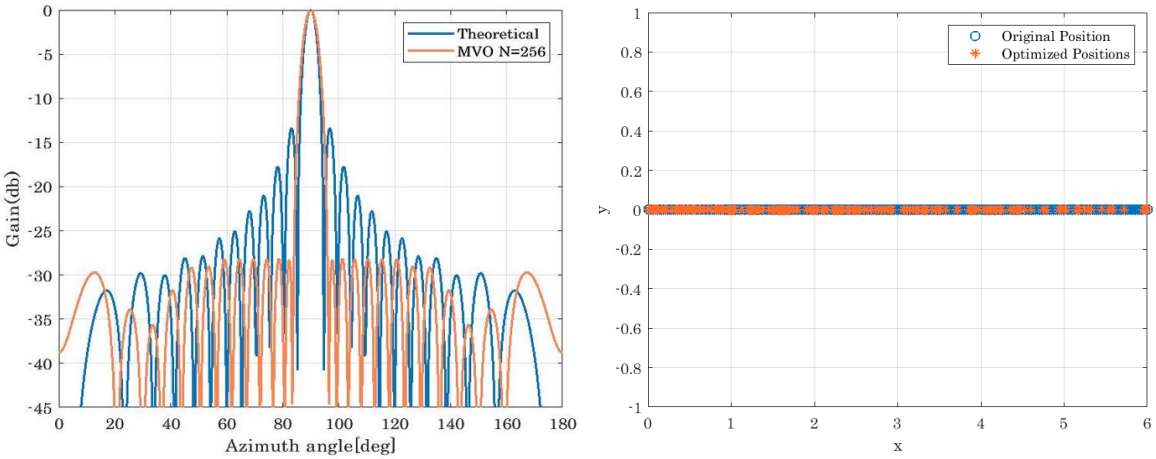
c. N =32 Elements



d. N =64 Elements



e. N =128 Elements



f. N =256 Elements

Figure (4.13). MVO with different numbers of elements.

Note that the higher number of antennas leads to a better main lobe and less SLL, leading to less inter symbol interference as shown in Fig (4.14) illustrates all MVO instances.

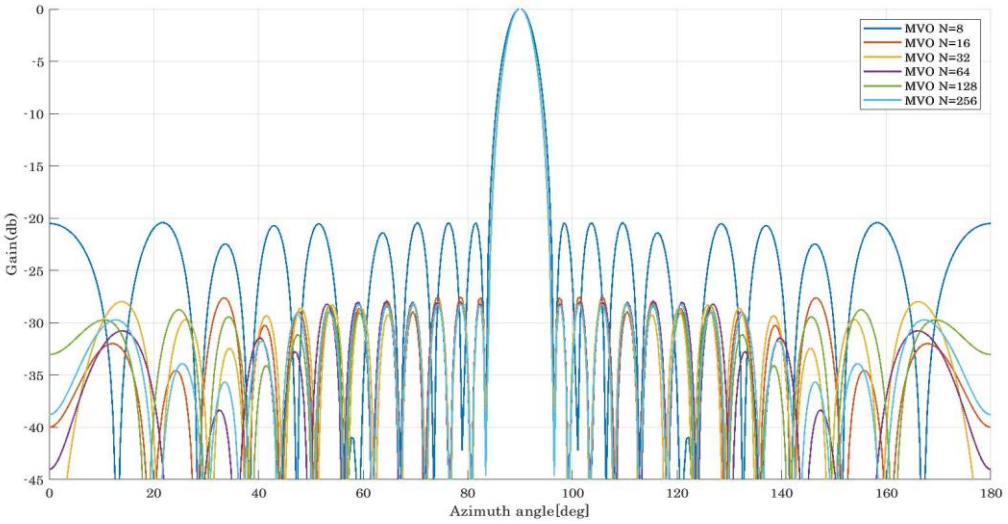
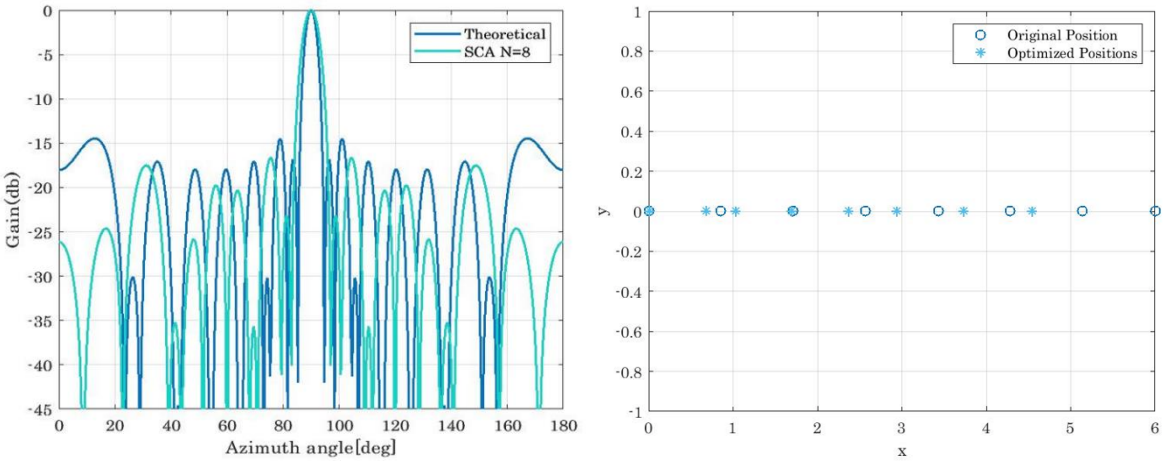


Figure (4.14). MVO Comparison by the Number of Elements

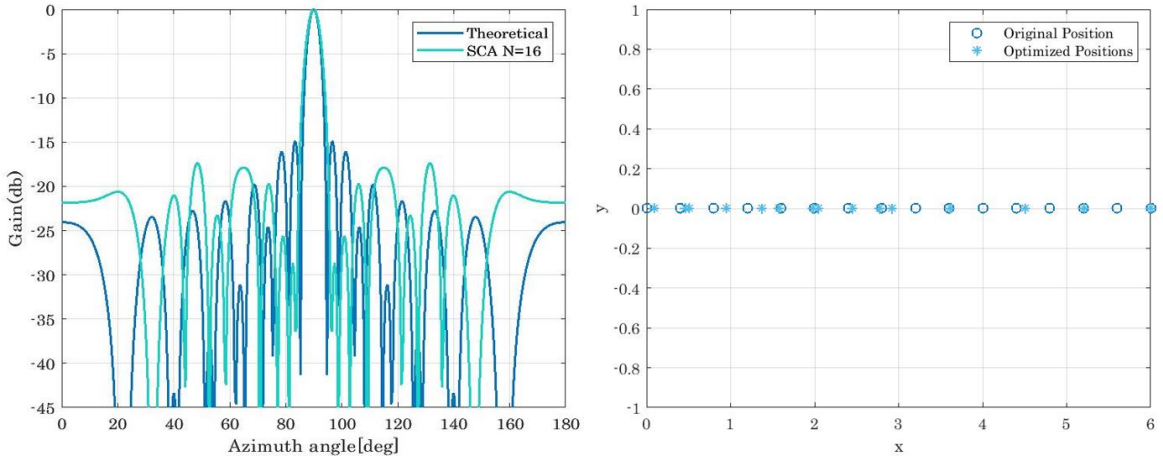
4.2.8 Results of Sine Cosine Algorithm

Based on the steps in section (3.4.8) the results were reached in Fig (4.15). (a, b, c, d, e, and f). Shows positions change for each antenna element the amount of attrition of SLL is shown at each number of antenna elements where at N = 8, SLL reduced its theoretical value from -16.8470 dB to -23.1215 dB it is considered the best value reached at N = 8. At N = 16, SLL decreased significantly by -28.6351 dB while the theoretical beam pattern was -14.8815 dB. SLL decreased from -14.0977 dB to -22.8923 dB at N = 32.

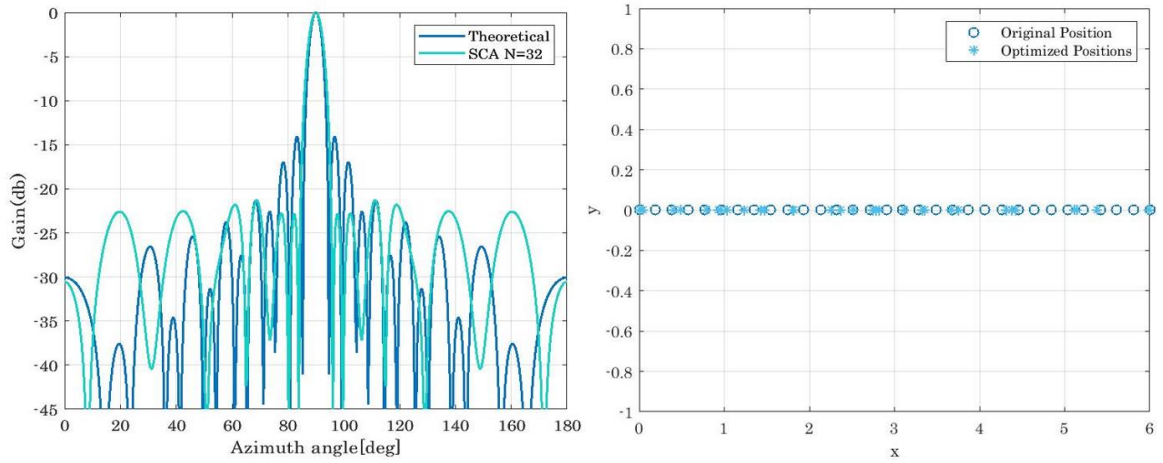
At N = 64, SLL reduced from -13.6939 dB to -26.2954 dB. At N = 128 SLL decreased from a theoretical beam of -13.4779 dB to -28.5741 dB. The best decrease was obtained at N = 256 where SLL reduced to -29.2229 dB. SCA is better than all algorithms at its best value at N = 256.



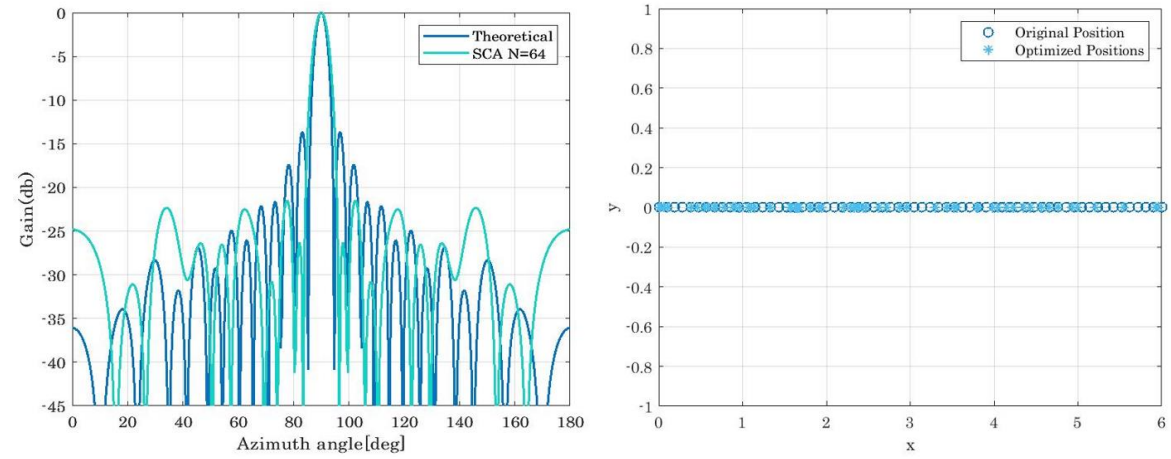
a. N =8 Elements



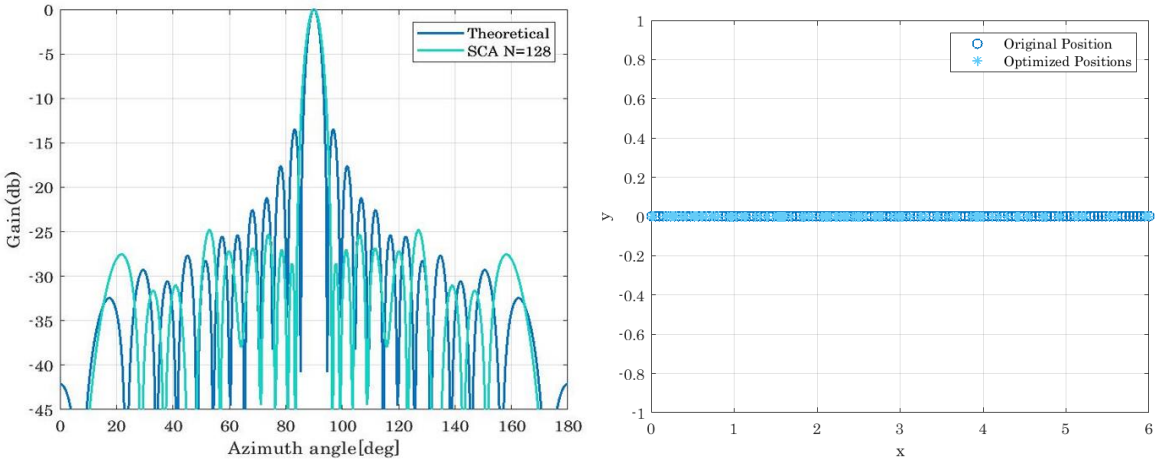
b. N =16 Elements



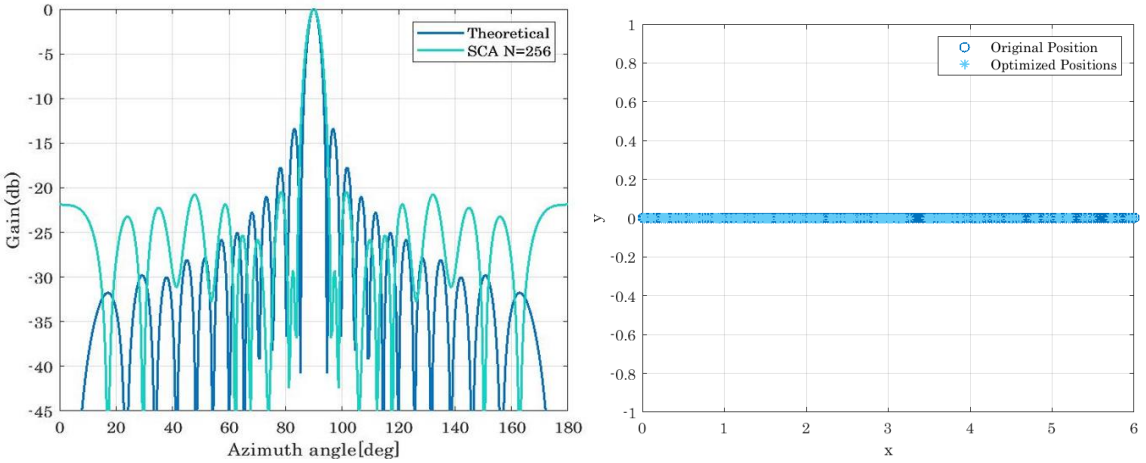
c. N =32 Elements



d. N =64 Elements



e. N =128 Elements



f. N =256 Elements

Figure (4.15). SCA with different numbers of elements.

Note that the higher number of antennas leads to a better main lobe and less SLL, leading to less inter symbol interference as shown in Fig (4.16) shows all cases of SCA.

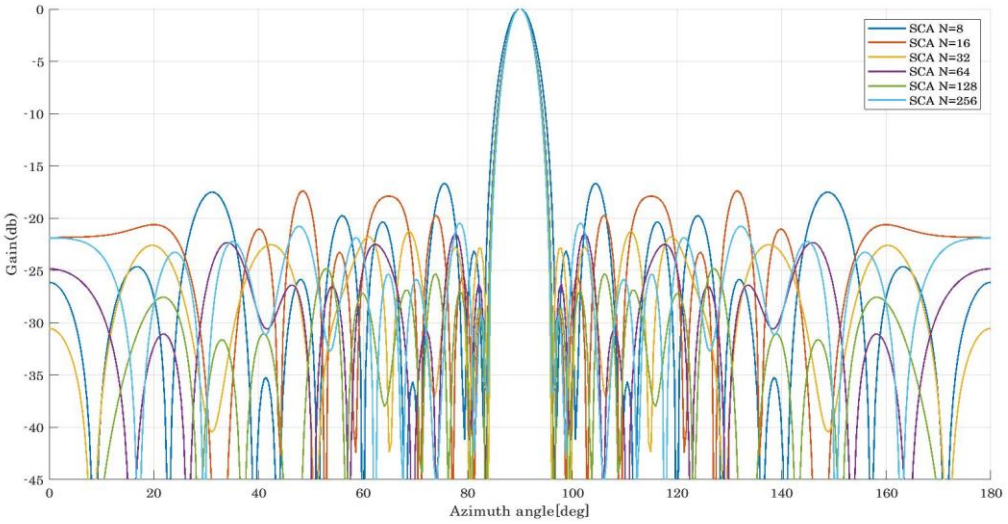
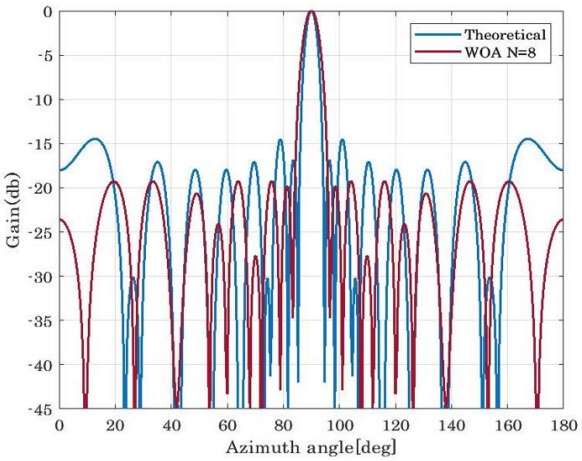


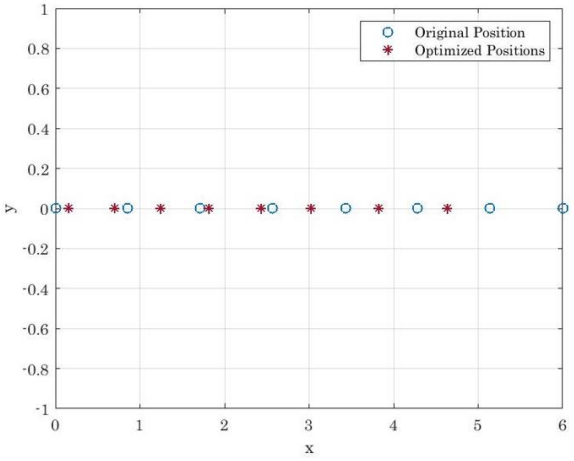
Figure (4.16). Shows SCA comparison by the number of elements.

4.2.9 Results of Whale Optimization Algorithm

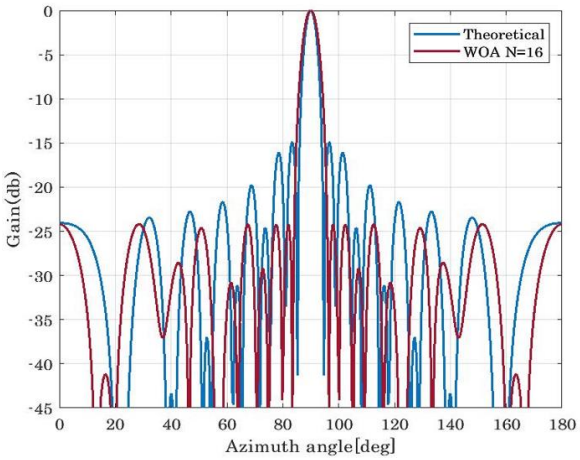
Based on the steps in section (3.4.9) the results were reached in Fig (4.17). (a, b, c, d, e, and f). Displays a change in positions for each antenna element and each number of antenna elements displays the quantity of SLL attrition at N = 8, SLL will be reduced to -19.8081 dB after the beam pattern value has been -16.8470 dB. While SLL dropped to -24.2012 dB at N = 16 after the theoretical pattern -14.8815 dB. SLL decreased from -14.0977 dB to -25.2061 dB.

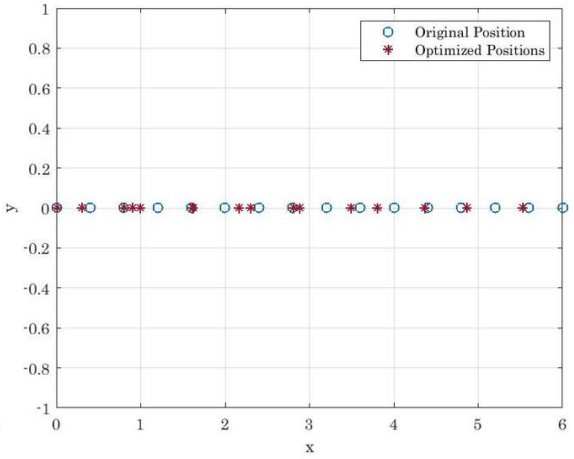
At N = 32. At N = 64,128, and 256, SLL dropped to -25.3559 dB, -26.4136 dB, and -26.7006 dB respectively. The best drop-in SLL when using WOA was found to have occurred at N = 256, but it's the least reduced SLL among other algorithms.



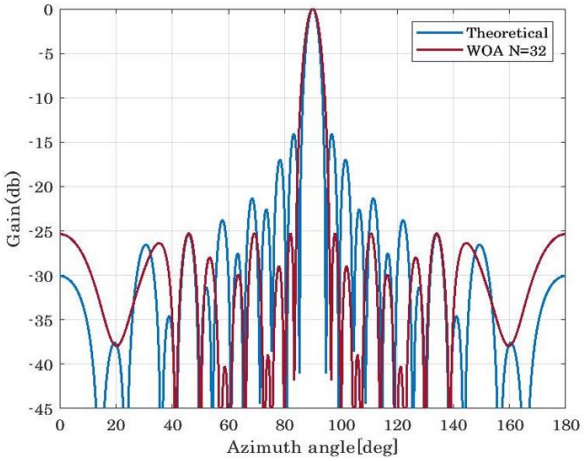


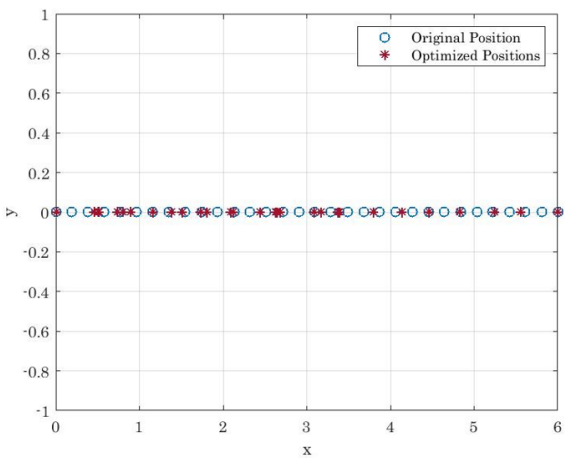
a. N =8 Elements



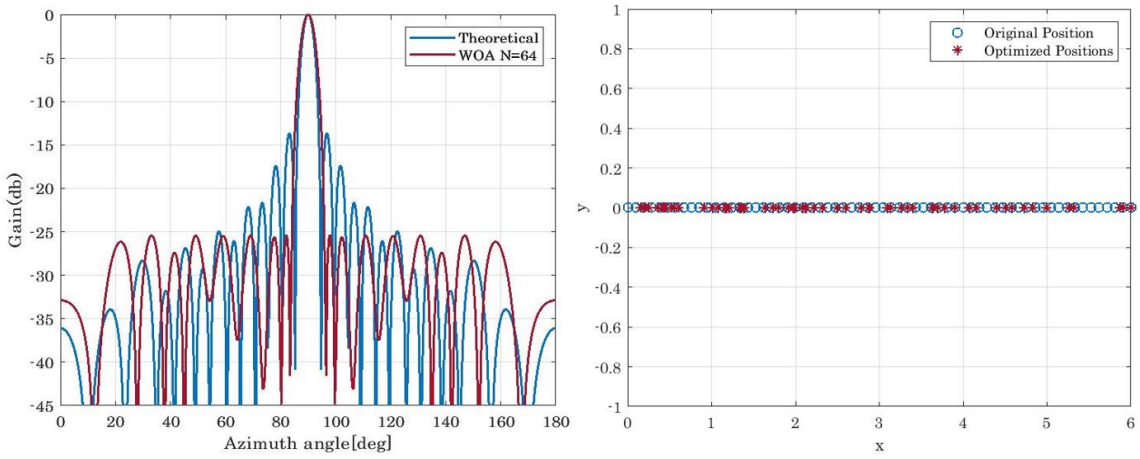


b. N =16 Elements

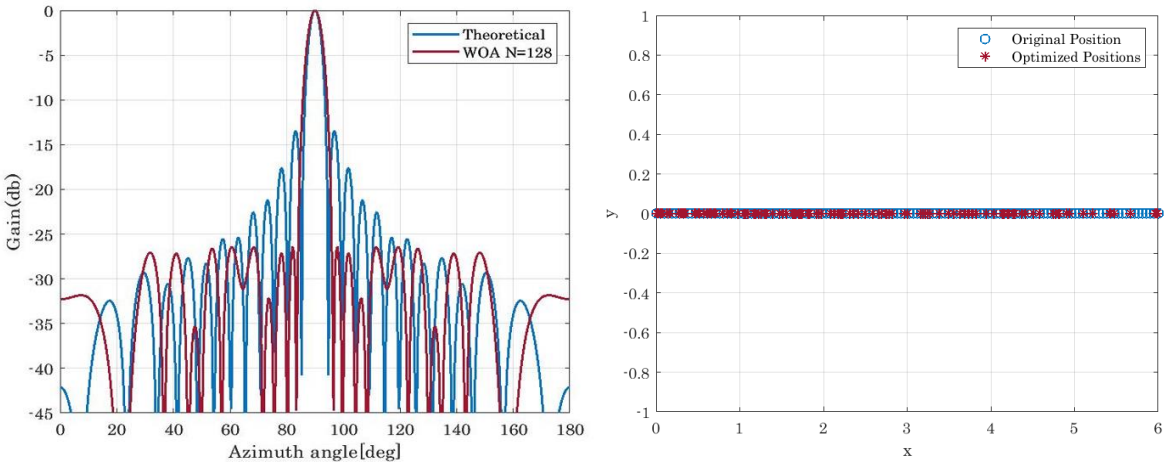




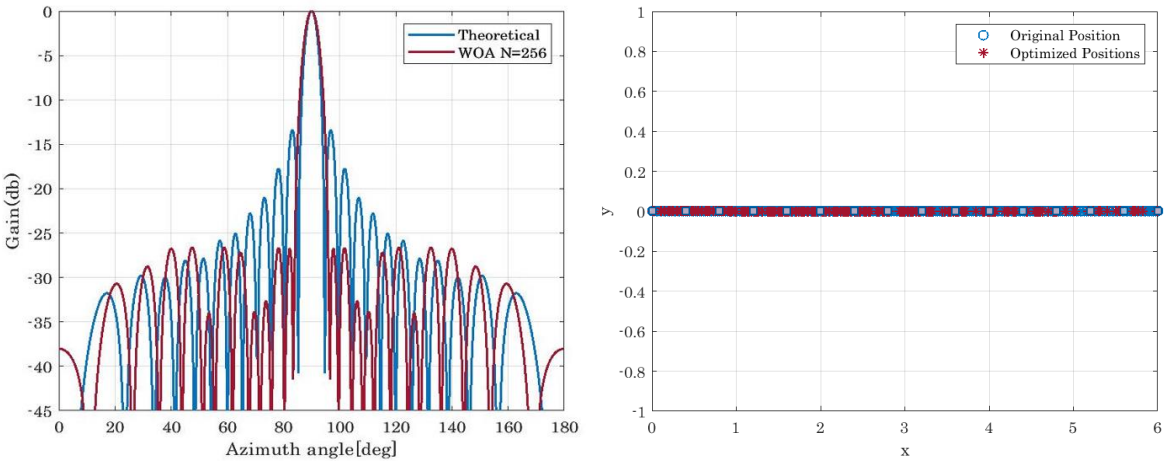
c. N =32 Elements



d. N =64 Elements



e. N =128 Elements



f. N =256 Elements

Figure (4.17). WOA with different numbers of elements.

Note that the higher number of antennas leads to a better main lobe and less SLL, leading to less inter symbol interference as shown in Fig (4.18) illustrates all WOA instances.

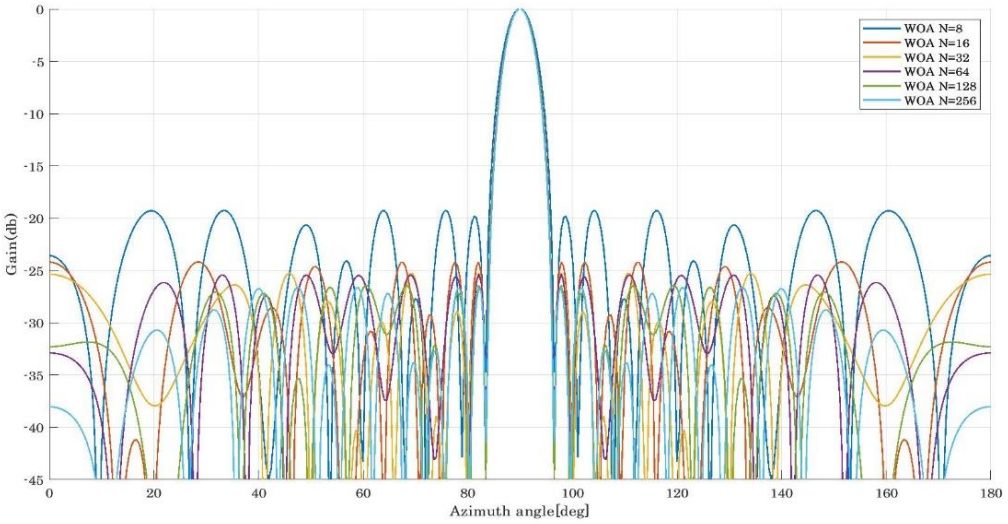
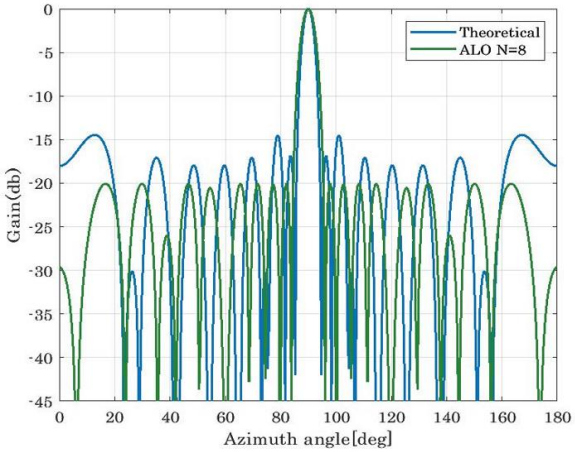


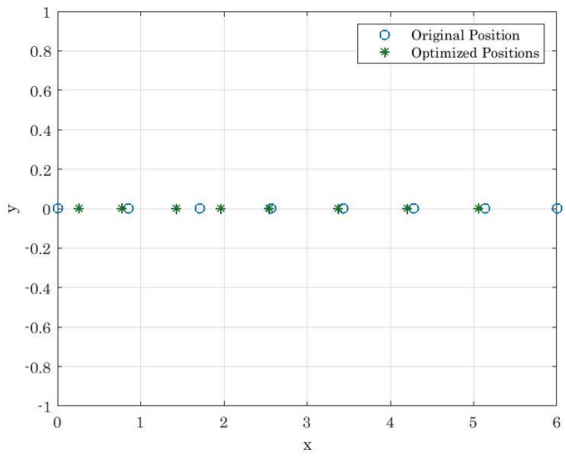
Figure (4.18). WOA comparison by the number of elements.

4.2.10 Results of Ant Lion Optimization Algorithm

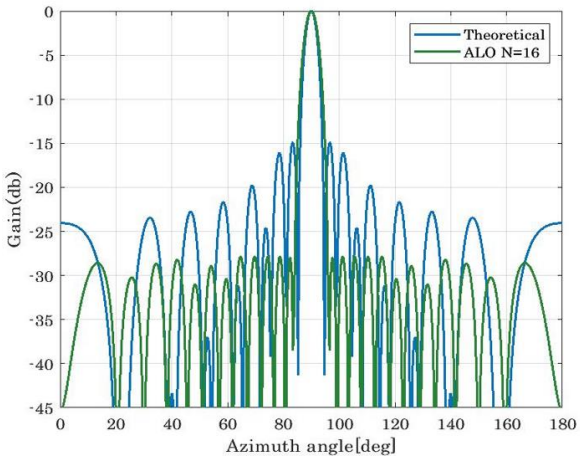
Based on the steps in section (3.4.10) the results were reached in Fig (4.19). (a, b, c, d, e, and f). Shows positions change for each antenna element, the amount of attrition of SLL is shown at each number of antenna elements where at N = 8, SLL will be reduced to -20.0224 dB after the beam pattern value has been -16.8470 dB. While SLL dropped to -27.9199 dB at N = 16 after the theoretical pattern -14.8815 dB. SLL decreased from -14.0977 dB to -27.9448 dB at N = 32.

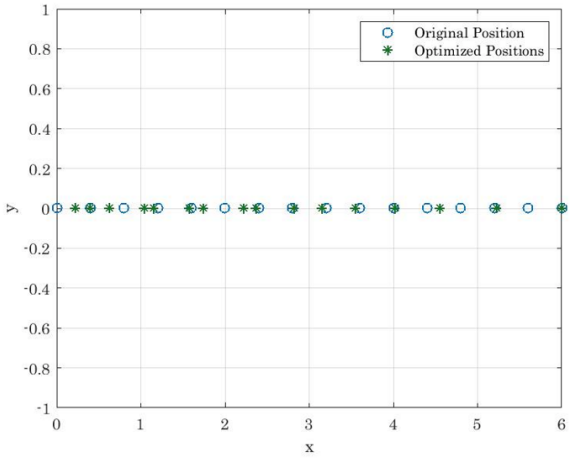
At N = 64,128, and 256, SLL dropped to -28.2614 dB, -28.3031 dB, and -27.7431 dB respectively. The best reduced SLL when using ALO was found to have occurred at N = 128. Although by increasing the number of antenna elements we can get adverse results this is because despite the complexity of the system as shown in Fig (4.19). e.



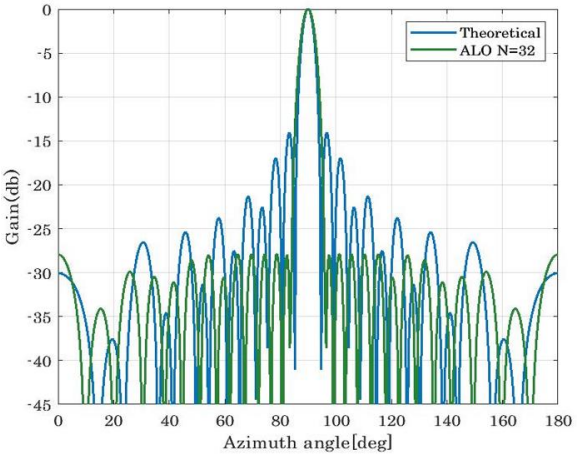


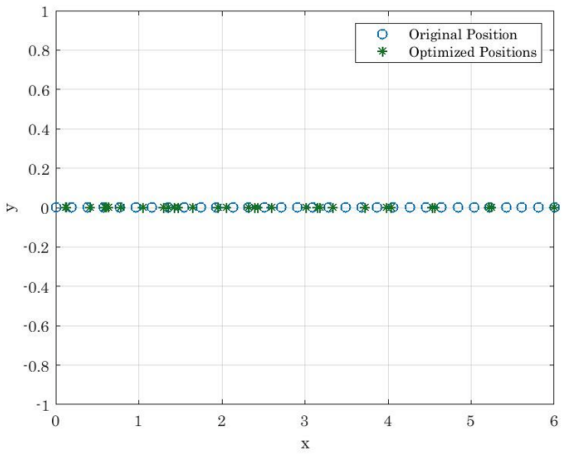
a. N =8 Elements



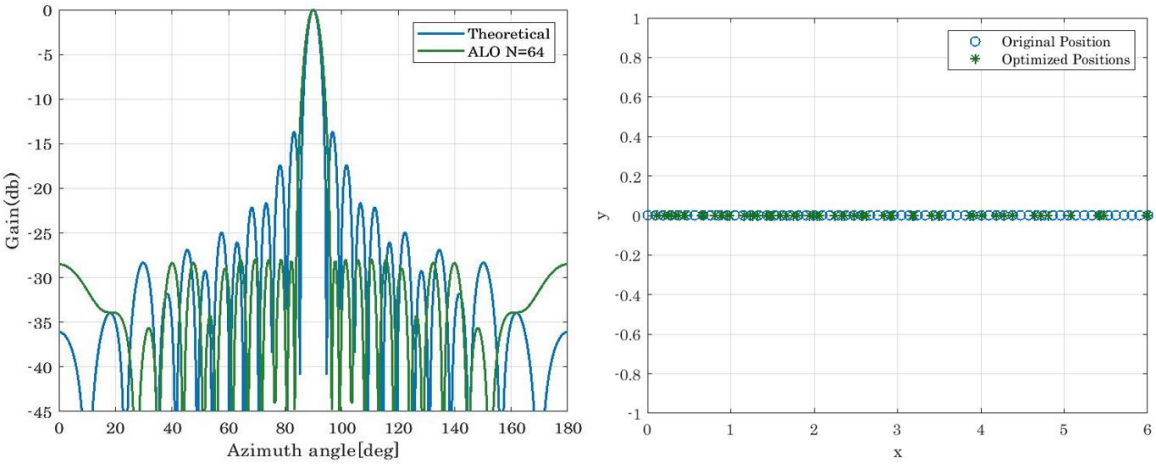


b. N =16 Elements

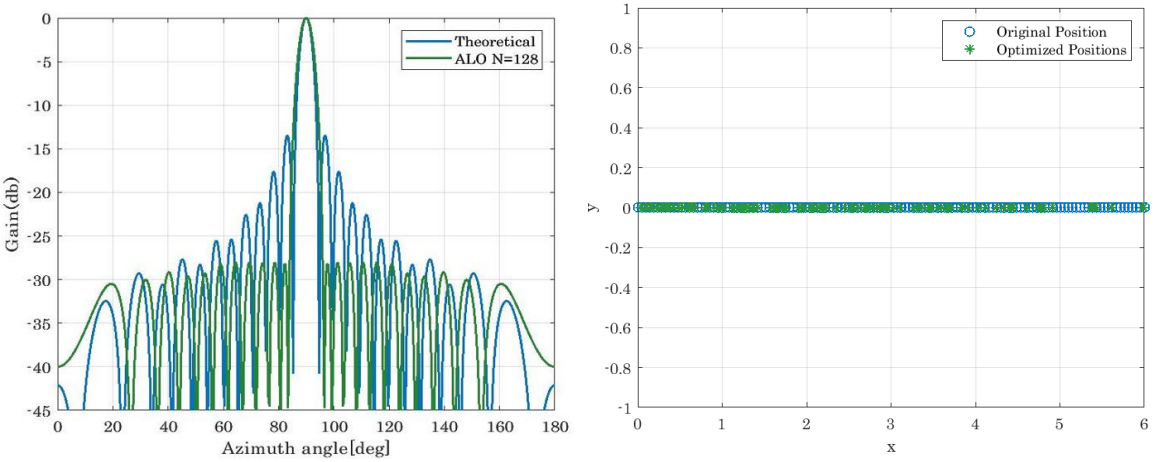




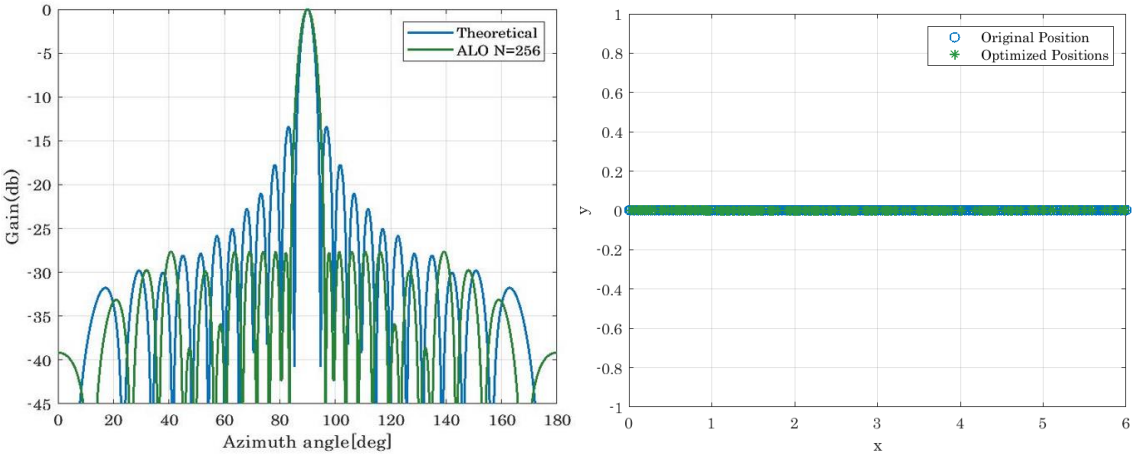
c. N =32 Elements



d. N =64 Elements



e. N =128 Elements



f. N =256 Elements

Figure (4.19). ALO with different numbers of elements.

Note that the higher number of antennas leads to a better main lobe and less SLL, leading to less inter symbol interference as shown in Fig (4.20) illustrates all ALO instances.

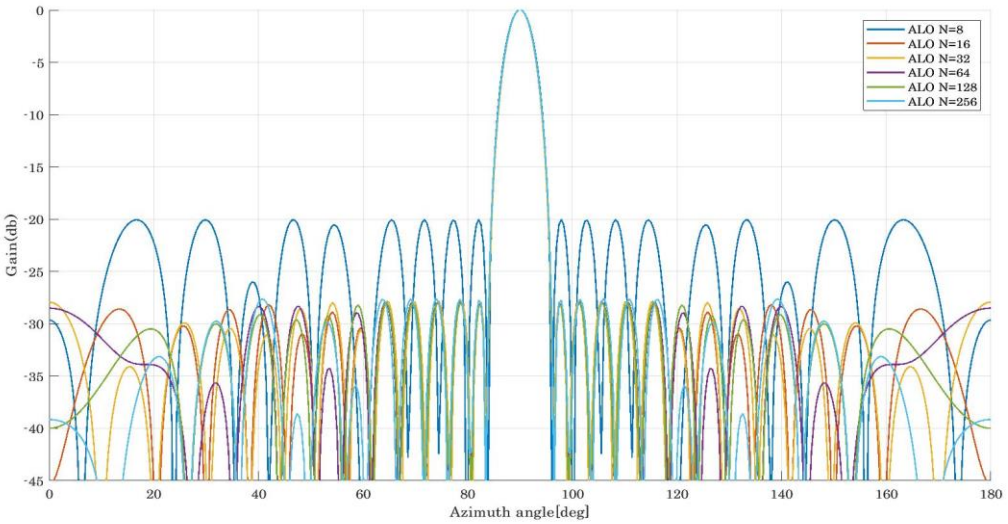


Figure. (4.20). ALO comparison by the number of elements.

4.3 Comparison Between Optimization Techniques

When comparing these algorithms at N = 8 we note SCA is better where SLL is reduced to -23.1215 dB and then FPA reduced SLL to -20.8492 dB. and then MVO where SLL is reduced to -20.4588 dB and then followed by the rest of the algorithms down to WOA is the lowest SLL reduction. As shown in Fig (4.21). when comparing all algorithms in N = 8.

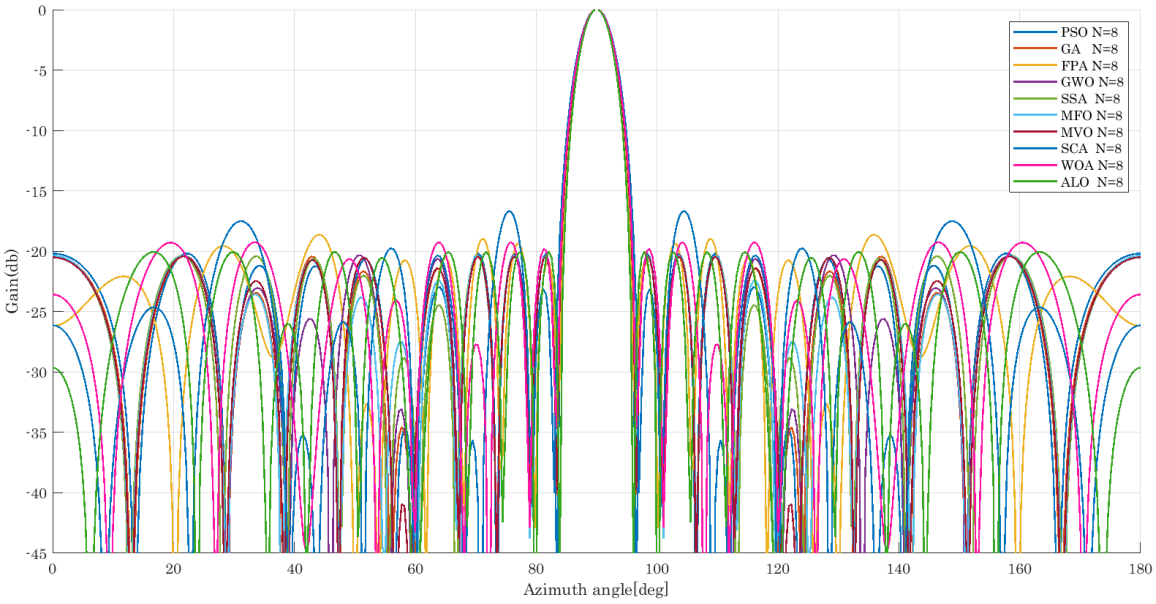


Figure (4.21). Comparison between algorithms at N = 8.

At N = 16, SCA is still superior as SLL is reduced to -28.6351 dB, followed by ALO which decreased SLL to -27.9199 dB, followed by MFO which decreased to -27.8226 dB, and then the rest of the algorithms. As shown in Fig (4.22). when comparing all algorithms N=16.

FPA is the best algorithm when N = 32 where SLL dropped to -28.3071 dB, while MVO worked to reduce SLL to -28.0246 dB, it is the lowest after FPA while ALO follows MVO outperformance where SLL reduced to -27.9448 dB, and then followed by the rest of the algorithms down to SCA which It's the least superior at N = 32. As shown in Fig (4.23).

When comparing the results of the algorithms at N = 64, the best result was found is MVO, where it reduced SLL to -28.2811 dB, followed by ALO, reducing it to -28.2614 dB, and then coming in order SSA, FPA, and GA, where SLL is -28.1870 dB, -28.0148 dB, and -28.0044 dB respectively dropped. As shown in Fig (4.24). when comparing all algorithms in N = 64.

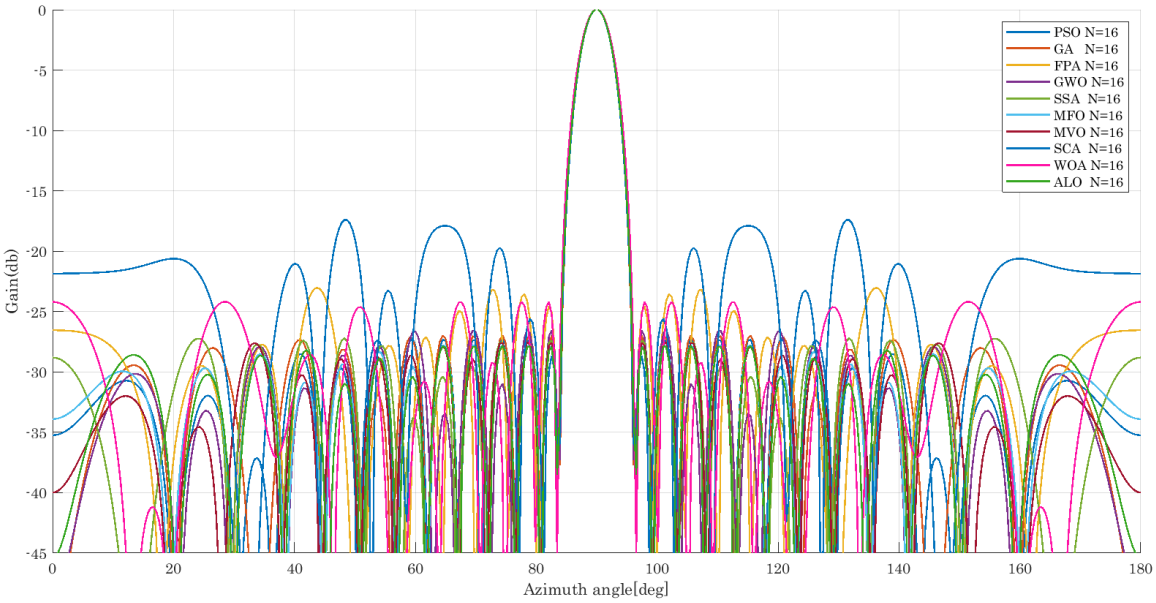


Figure (4.22). Comparison between algorithms at N = 16.

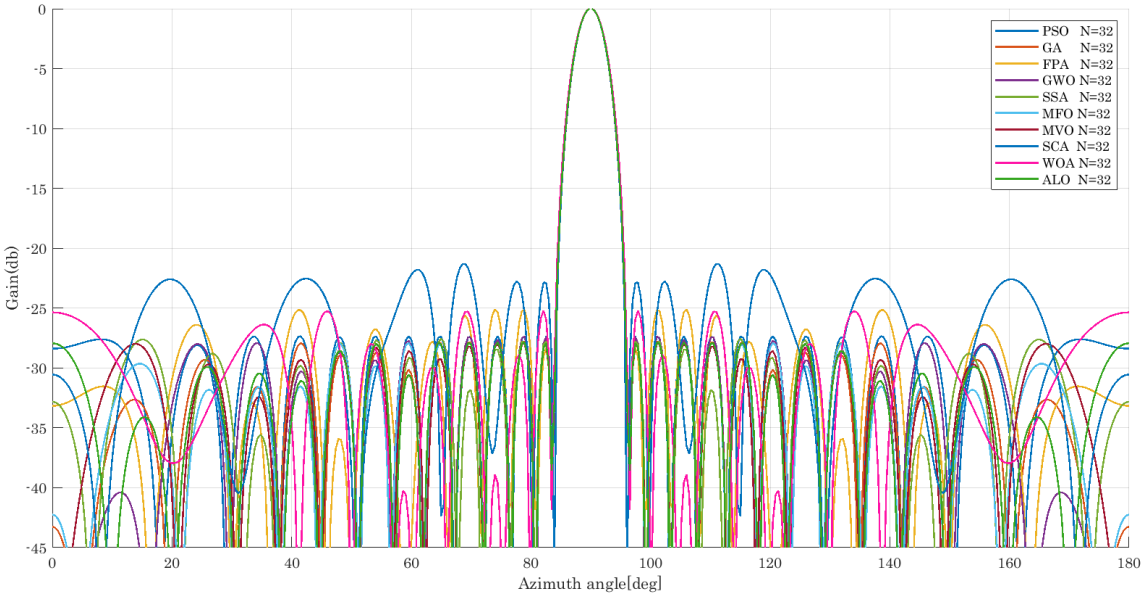


Figure (4.23). Comparison between algorithms at N = 32.

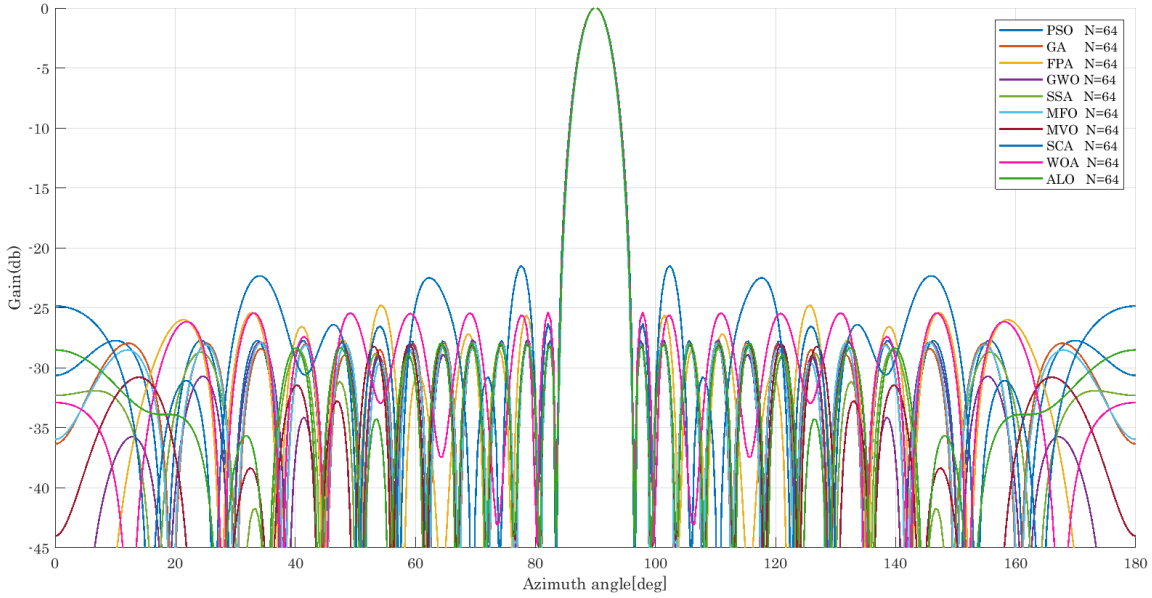


Figure (4.24). Comparison between algorithms at N = 64.

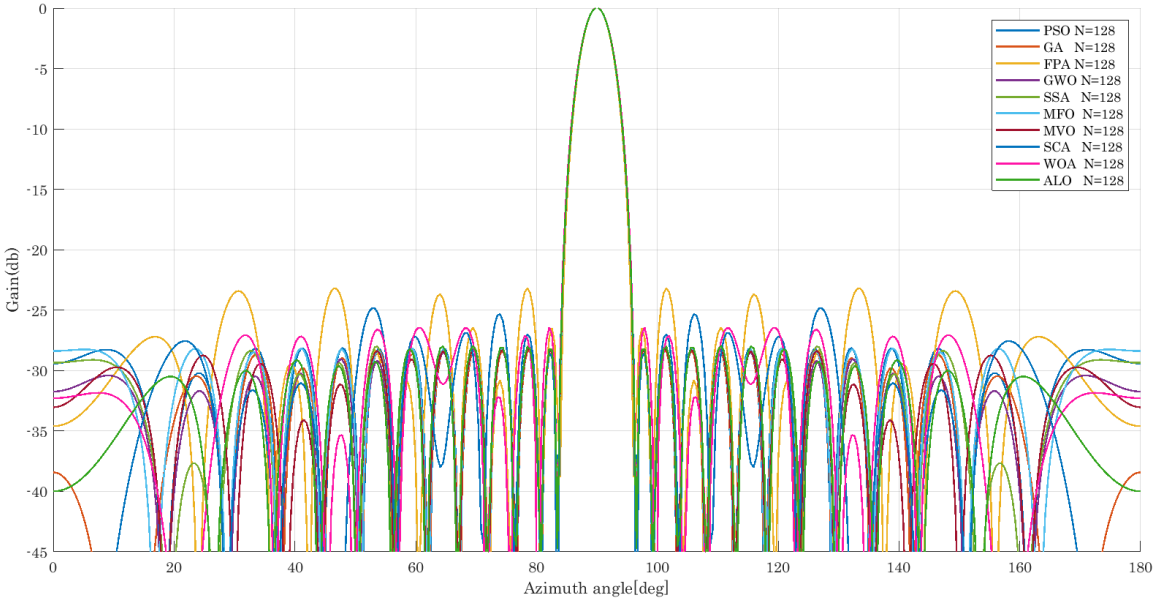


Figure (4.25). Comparison between algorithms at N = 128.

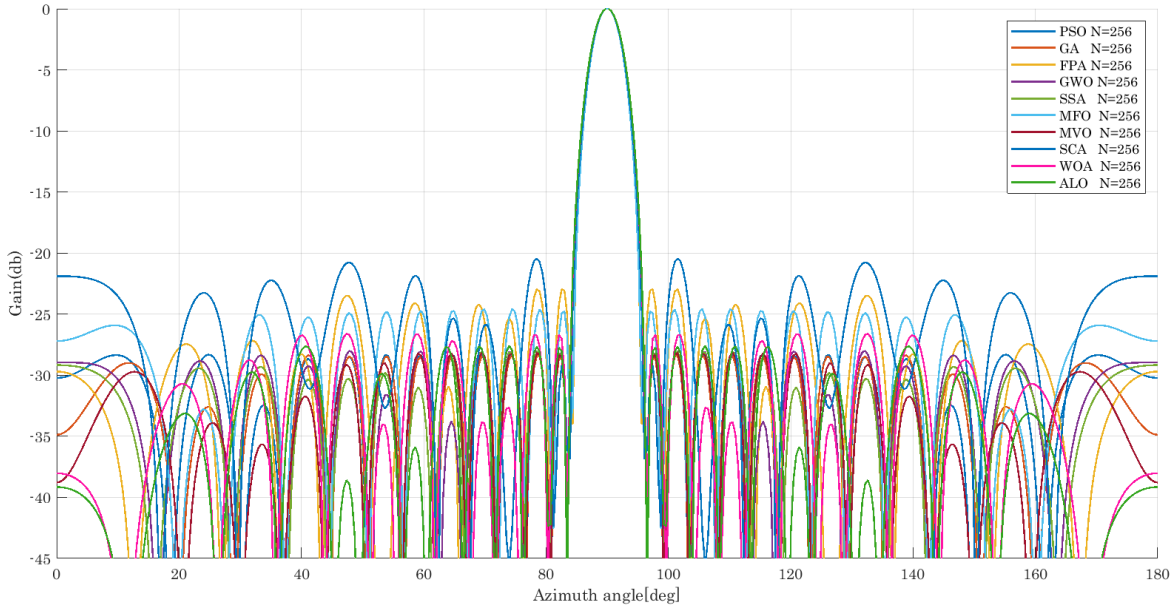


Figure (4.26). Comparison between algorithms at N = 256.

Comparing the results of the algorithms at N = 128, the best result was found to be reused SCA, which reduced SLL to -28.5741dB, followed by GA, which reduced SLL to -28.5568 dB, and then MVO, MFO, and PSO, where SLL fell by -28.4366 dB, -28.3859 dB, and -28.3277 dB, respectively. As shown in Fig (4.25). when comparing all algorithms in N = 128.

SCA is the best algorithm when N = 256 where SLL dropped to -29.2229 dB, while GA worked to reduce SLL to -28.6204 dB, it is the lowest after SCA while PSO follows GA outperformance where SLL reduced to -28.5405 dB and then followed by the rest of the algorithms down to FPA which It's the least superior at. As shown in Fig (4.26). when comparing all algorithms in N =256. Table. (4.1). shows all the results reached using all algorithms. This shows that the best algorithm to reduce SLL to the maximum is SCA at N = 256 and as much as -29.2229 dB.

Table. (4.1). Show all results of each algorithm

Algorithm	Number of Element						Gain (dB)
	8	16	32	64	128	256	
Theoretical	-16.847	-14.8815	-14.0977	-13.6939	-13.4779	-13.3727	
PSO	-20.1984	-27.2992	-27.3525	-27.8289	-28.3277	-28.5405	
GA	-20.4335	-26.9987	-27.8764	-28.0044	-28.5568	-28.6204	
FPA	-20.8492	-24.5472	-28.3071	-28.0148	-26.4663	-23.0646	
GWO	-20.3723	-26.5385	-27.3854	-27.7335	-28.2732	-28.3367	
SSA	-20.4192	-27.4011	-27.7574	-28.187	-28.2192	-28.3453	
MFO	-20.4166	-27.8226	-27.9496	-27.9725	-28.3859	-24.7649	
MVO	-20.4588	-27.6097	-28.0246	-28.2811	-28.4366	-28.3551	
SCA	-23.1215	-28.6351	-22.8923	-26.2954	-28.5741	-29.2229	
WOA	-19.8081	-24.2012	-25.2061	-25.3559	-26.4136	-26.7006	
ALO	-20.0224	-27.9199	-27.9448	-28.2614	-28.3031	-27.7431	

The decrease of SLL reached in the results is aimed at reducing overlap with other signals and increasing the gain in directing the signal in the right direction as well as increasing signal to noise ratio.

4.4 Reduce Side Lobes Level for LAA Based on the Effects of Parameters of Optimization Techniques

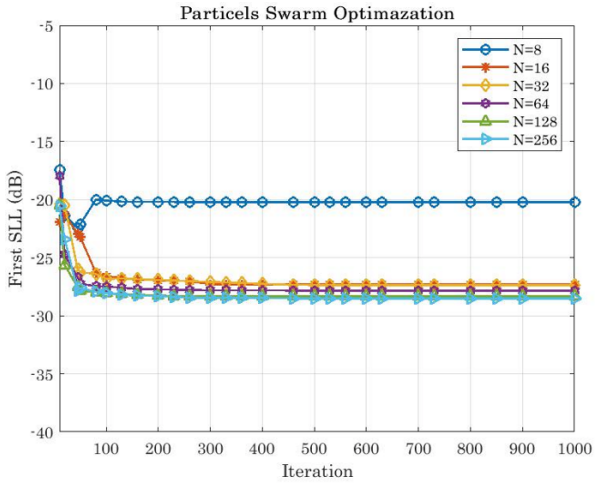
When studying the impact of parameters on the minimization of SLL for each element of the antenna, some of them were found to affect the reduction and others not to affect significantly, but a slight change.

4.4.1 The Parameters Affecting Reduced SLL by PSO

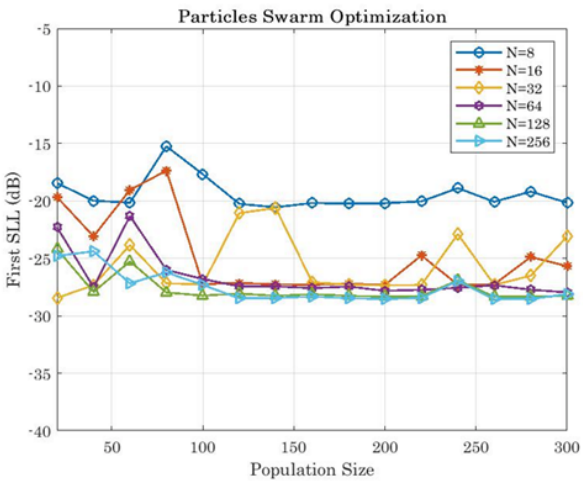
In Fig (4.27). (a). (b). and (c), shows PSO tested of the iterations, population size, and max stall iterations as parameters influencing the reduction of SLL. In N=8, the effect of the iterations is slightly wobbly at first and then fully stabilizes and it is considered the highest reduction obtained at iteration 45 and then settles after iteration 200 to iteration 1000

at -20.1984 dB. At N = 16 it is unstable at first until iteration 530 and beyond it is stable. At N = 32.64,128, and 256 it is quite stable where the best SLL reduction is -27.8289 dB at iteration 630, -27.8289 dB at iteration 800, and -28.3277 dB at iteration 700, and -28.5405 dB at iteration 800 respectively. When comparing the effect of iteration in the collection of antenna elements, the maximum reduction is at N = 256 at iteration 800, with a value of -28.5405 dB.

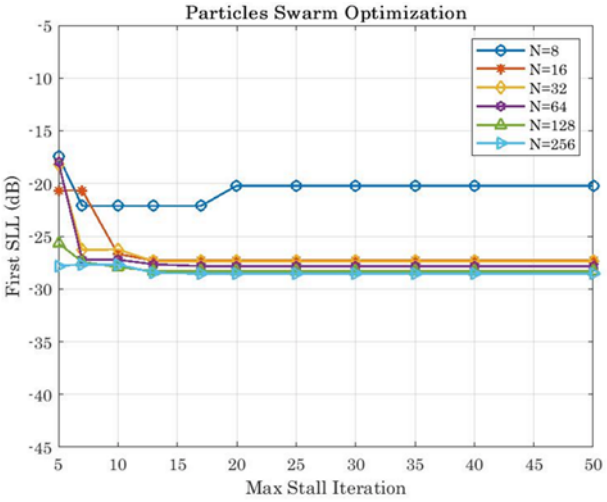
The effect of population size at N = 8 is stable, except at population sizes 80 and 100 at a value of -15.2463 dB, -17.6758 dB. At N = 16, the size of the population fluctuates up to the population size of 80, with a value of -17.3626 dB, and beyond this point, the population size is stable down to the size of the population 220, then it goes back to the fluctuation. At N = 32 the population size is wobbly but the greatest amount of SLL reduction is -28.4393 dB at population size 20. At N=64,128, and 256, they are unstable to reduced SLL initially and after that goes to stable but the highest drop values are -28.4393 dB at population size 20, -27.9617 at the population size of 300, -28.3373 dB and -28.5471 dB respectively at population size 260. The max stall iteration is stable except for fluctuating at certain points.



a. Iteration



b. Population size



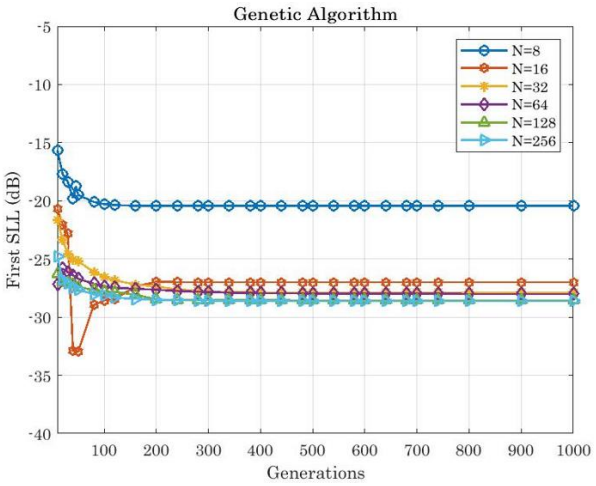
c. Max stall iteration

Figure (4.27). Affect parameters of PSO.

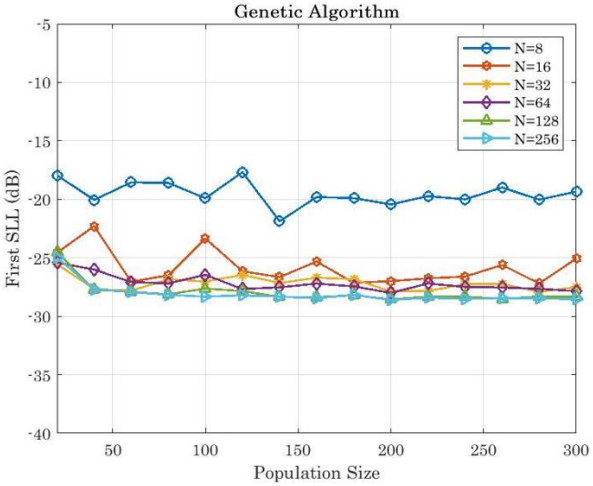
4.4.2 The Parameters Affecting Reduced SLL by GA

In Fig (4.28). (a). (b). and (c). Shown GA tested the Generations, population size, and max stall generation as parameters influencing the reduction of SLL. In N=8, the effect of the generation is slightly wobbly at first and then fully stabilizes and the best drop of SLL is -20.4335 dB at generation 240. At N = 16 it is unstable at first until the 120 generation and beyond it is stable and the best reduction SLL is -32.9523 dB at generation 50. At N = 32,64,128, and 256, it is quite stable where the SLL is descending, the best SLL reduction is -27.8764 dB at generation 500, -28.0044 dB at 900 generation 900, -28.5568 dB at generation 680, and -28.6204 dB at generation 1000, respectively.

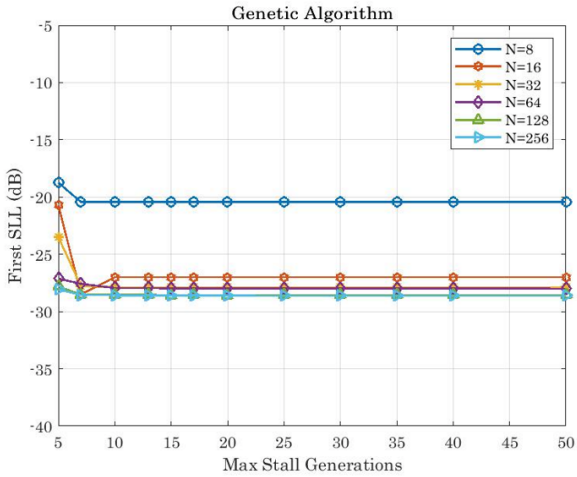
The effect of population size at N = 8 is stable, except at population size 120 and a value of reduction SLL to -17.6770 dB. In N = 16 the size of the population is fluctuating where is stable and the other is not. At N = 32,64,128, and 256 the reduction of SLL is unstable but the highest drop values are -27.8764 dB, -28.0044 dB, -28.5568 dB, and -28.6204 dB respectively at population size 200. The effect of max stall generation is very little.



a. Generations



b. Population size



c. Max stall generation.

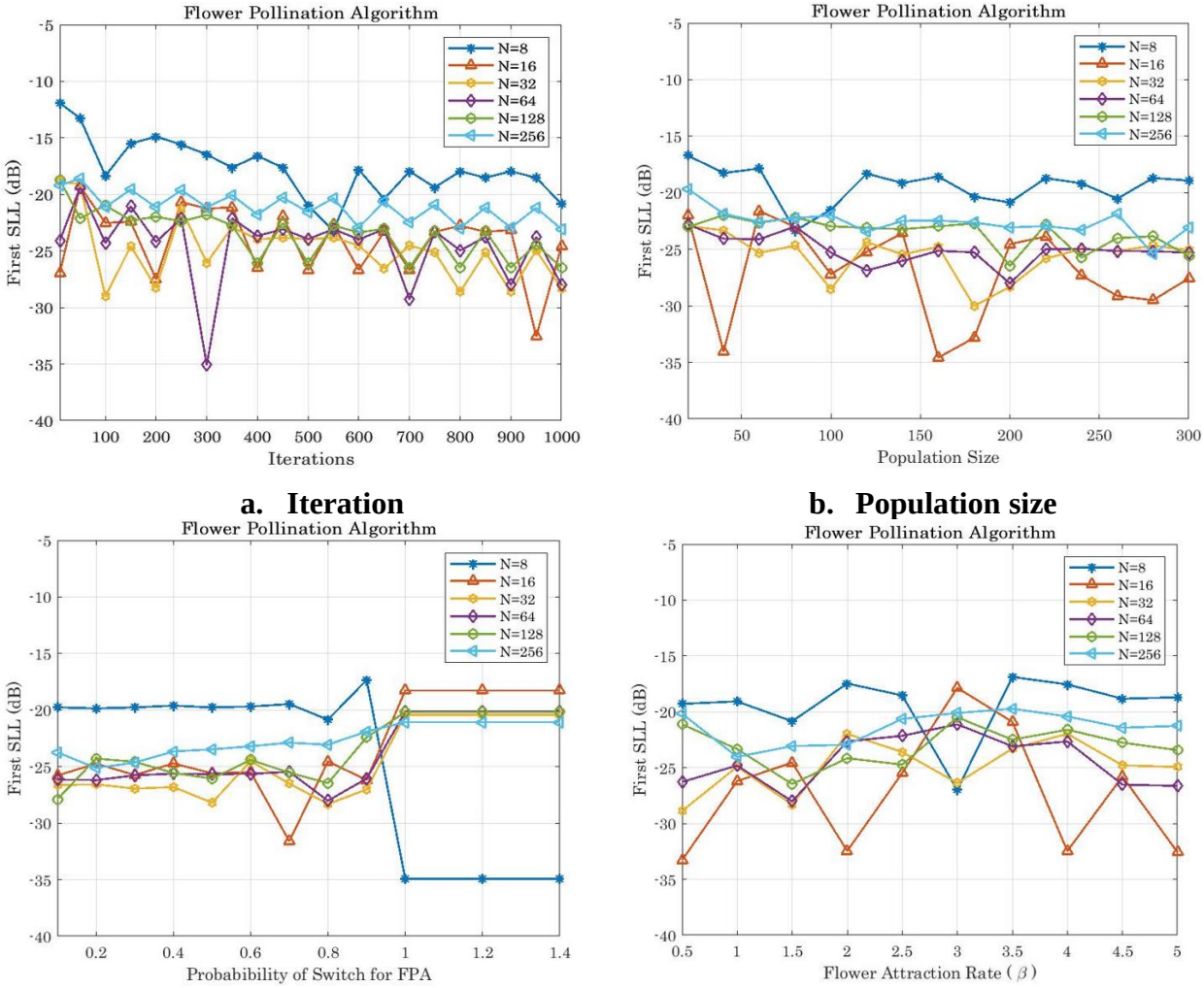
Figure (4.28). Affect parameters of GA.

4.4.3 The Parameters Affecting Reduced SLL by FPA

In Fig (4.29). (a). shown the effect of iteration is very wobbly, that is, it looks like a zig-zag. In N = 8 the highest SLL reduction is -23.2012 dB at iteration 550. At N = 16 at iteration 950 the best reduction for SLL is -32.5694 dB. At N = 32,64,128, and 256. The best SLL reduction is -29.0334 dB in iteration 100, -35.0696 dB at iteration 300, -26.4663 dB at iteration 700, and -23.0646 dB at iteration 1000 respectively.

The effect of population size N = 8,16,32,64,128, and 256, is very wobbly where the best value SLL has been reduced is -23.3463 dB at population size 80, -34.5790 dB at population size 160, -28.5110 dB at

population size 100, -28.0148 dB at population size 200, -26.4663 dB at population size 200 and -25.4162 dB at population size 280. The probability effect at N = 8 is stable except at probability 0.9 and beyond is fully stable for a probability greater than 1 and the best reduction of SLL is -34.9451 dB, at N = 16 the effect is inclined to stability except at probability 0.7 where SLL falls to -31.6000 dB. At N = 32,64,128, and 256 the effect is slightly wobbly and then fully stabilizes when the probability is 1. The effect of the flower attraction rate is very unstable, it changes the amount of SLL at each change in the flower attraction rate so it is very influential and the best reduction is -33.2830 dB at N=16 at the flower attraction rate 0.5. As shown in Fig (4.29). (b), (c), and (d).



a. Iteration **b. Population size**

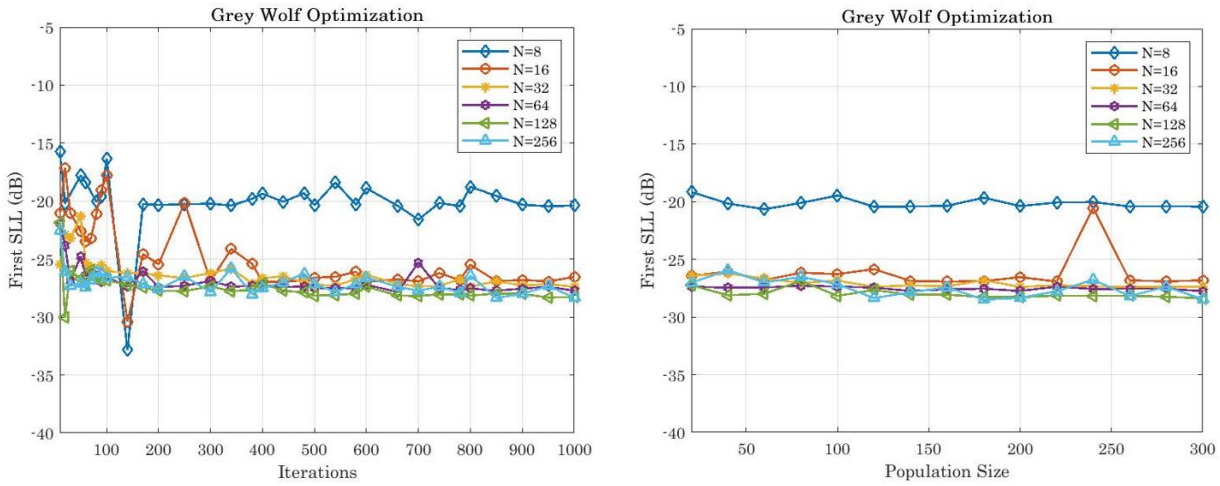
c. Probability of switch for FPA **d. Flower attraction rate**

Figure (4.29). Affect parameters of FPA

4.4.4 The Parameters Affecting Reduced SLL by GWO

In the GWO test of the iterations, and population size. In Fig (4.30). (a) shows the effect of iteration N= 8, and 16 is slightly wobbly except when iteration 140 the drop of SLL to -32.8479 dB, and -30.4126 dB respectively, and after this iteration is also wobbly but less, at N = 32,64,128, and 256, it has little instability, so reducing SLL is a bit convergent.

The effect of population size at N = 8,32,64,128, and 256 is very stable except at N = 16 its fluctuation is stable at the beginning until the population size is 240 so that SLL has decreased to -20.5262 dB, except that its effect is stable. As shown in Fig (4.30). (b).



a. Iterations **b. Population size**

Figure (4.30). Affect parameters of GWO.

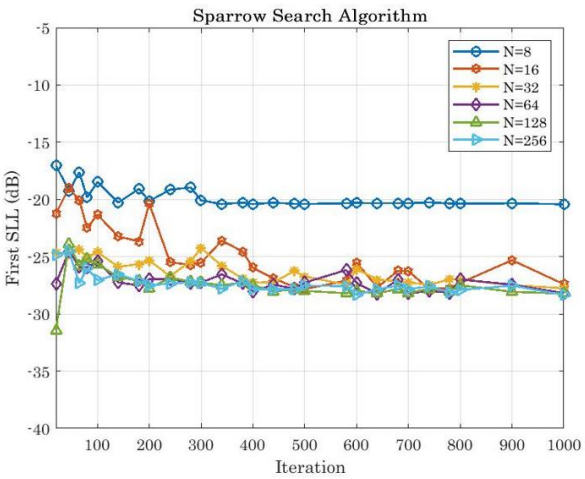
4.4.5 The Parameters Affecting Reduced SLL by SSA

In Fig (4.31). (a). (b). and (c). shown SSA test of the iterations, population size, and population size of the producers. The effect of iteration starts with instability at N = 8 until it reaches iteration 300 and then stabilizes and the highest SLL reduction is -20.4192 dB at iteration 1000. At N = 16, and 32 the best SLL reached is -27.8284 dB at iteration 500 and -27.7574 dB at iteration 1000 respectively.

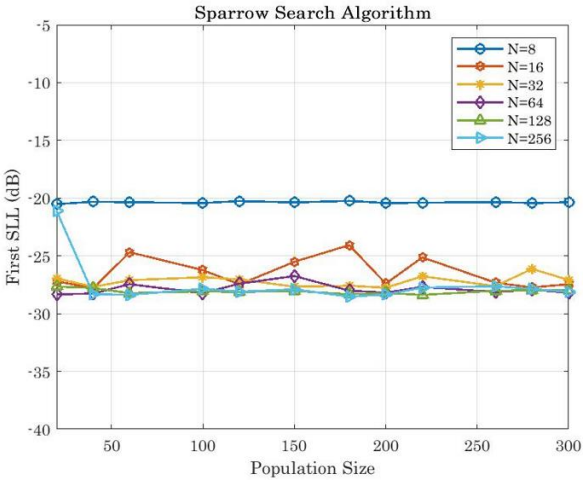
At N = 64,128,256 it looks a little wobbly but it is inclined to stability and the best SLL reduction is -28.2436 dB at iteration 640, -31.4211 dB at iteration 20, and -28.3453 dB at iteration1000 respectively.

The effect of population size is very small at N = 8 where it looks like a straight line either, at N = 16 the effect is slightly exposed and the best reduction is -27.7986 dB at population size 400, either at, N = 32,64, and 128, The effect of population size is very slightly curvy. At N = 256 the effect of population size is in the line, except at population size 20.

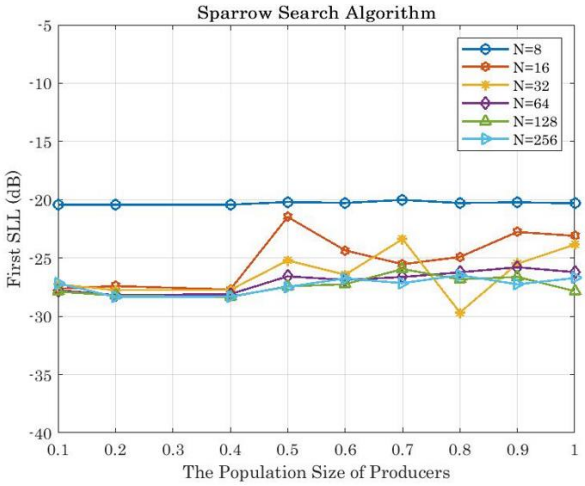
The effect of population size of producers at N = 8 is very slight, that is, it appears in a straight line, either at N = 16,32,64,128, and 256 the effect is initially a straight line, and then at a certain point it starts to fluctuate, where its highest values are reached to reduce SLL is at N = 32 and minimize its value -29.6727 dB at a population size of producers 0.9.



a. Iterations



b. Population size



c. Population size of producers

Figure (4.31). Affect parameters of SSA.

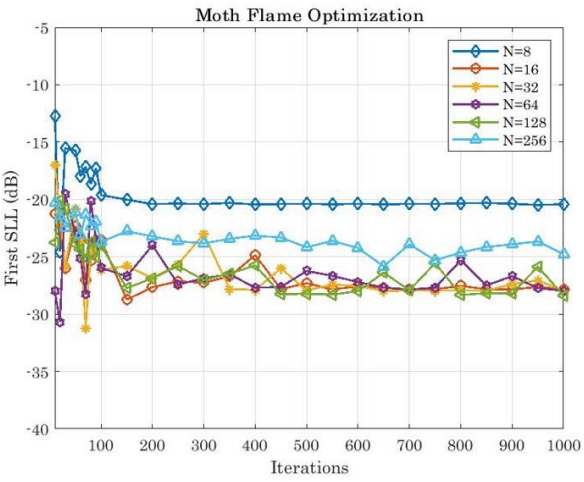
4.4.6 The Parameters Affecting Reduced SLL by MFO

In Fig (4.32). (a). (b). and (c). shown MFO test of the iterations, the population size, and the Spiral flight search. The effect of iteration starts with instability at N = 8 until it reaches iteration 100 and then stabilizes and the highest SLL reduction is -24.5652 dB at iteration 20. At N = 16, and 32 the effect looks wobbly. the best SLL reached is -28.7355 dB at iteration 150 and -31.2213 dB at iteration 70 respectively. At N = 64,128, and 256 it looks wobbly and the best SLL reduction is -30.6858 dB at iteration 20, -28.3859 dB at iteration 800, and -25.8379 dB at iteration 650 respectively.

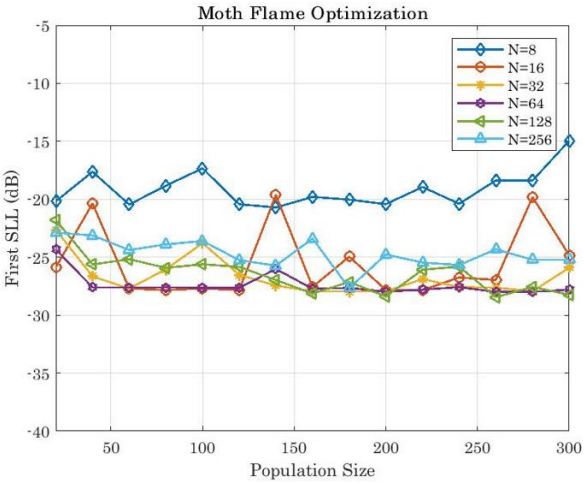
The effect of population size at N = 8 is slightly curvy and the greatest reduction of SLL is -20.7112 dB at population size 140; at N = 16, and 32 it is more diminished as the best values reached to reduce SLL are -27.8343 dB at population size 80 and -27.9496 dB at population size 200 respectively. When N = 64,128, and 256 the effect starts wobbly and then stabilizes.

The effect of spiral flight search at N = 8 is semi-straight as the highest SLL reduction is -20.4166 dB at spiral flight search 1 either at N = 16 is in line form except at point spiral flight search 2.5 reduce SLL to -32.4737dB. At N = 32 it is a gradually diminishing SLL line, at N = 64, and 128 is also

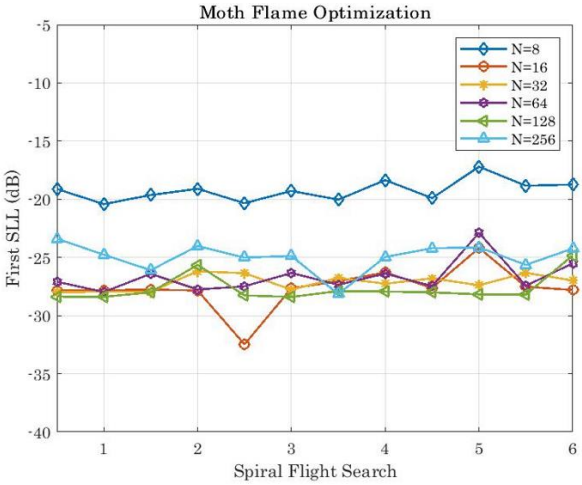
a line, except at 5 and 2 points, respectively, at N = 256 the effect is wobbly, and the greatest reduction of SLL is -28.1469 dB.



a. Iterations



b. Population size



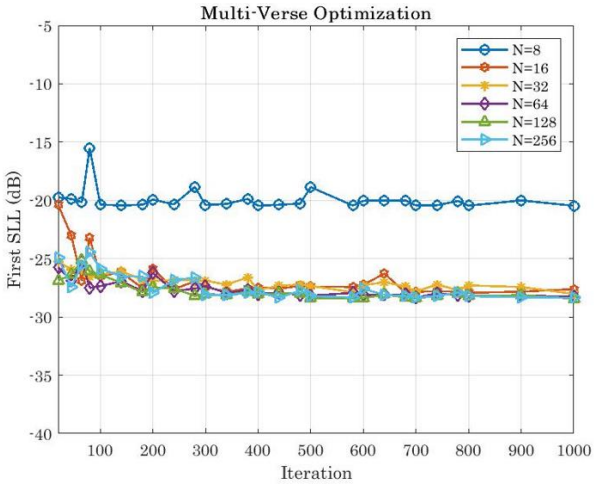
c. Spiral flight search

Figure (4.32). Affect parameters of MFO.

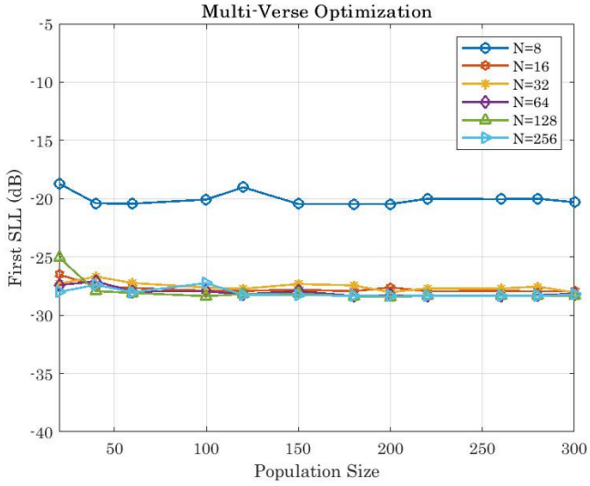
4.4.7 The Parameters Affecting Reduced SLL by MVO

In the MVO test of the iterations, the population size, and exploitation accuracy over the iterations. The effect of iteration at N = 8, and 16 on SLL reduction is stable except at iteration 80 and valued at -15.5056 dB and -23.2186 dB respectively. At, N = 32,64,128, and 256, the effect at first is slightly zigzag and then stabilizes with converging values. As shown in Fig (4.33). (a). The effect of population size on the reduction of SLL in this algorithm is similar to each particular number of antenna elements Fig

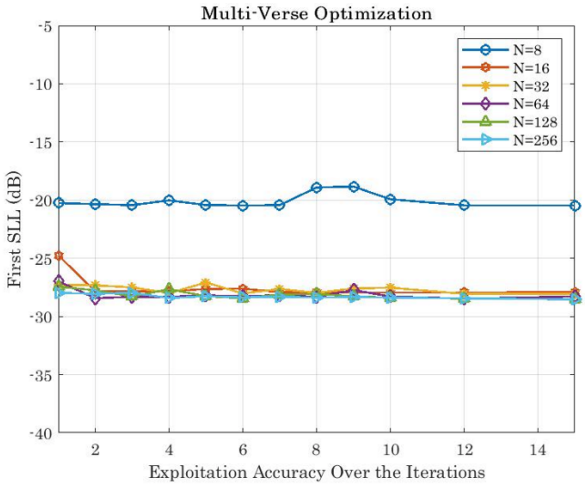
(4.33). (b). shows the stability of the effect, with converging values that are almost straight lines. It is the same effect when changing in exploitation accuracy over the iterations Fig (4.33). (c) shows the effect of the parameter that reduces SLL descending except at N = 8 has a different effect at 8 and 9 points and then rests at point 10.



a. Iterations



b. Population size



c. Exploitation accuracy over the iterations

Figure (4.33). Affect parameters of MVO.

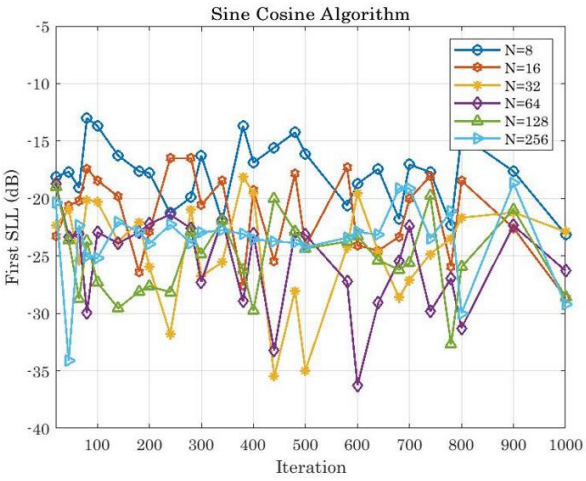
4.4.8 The Parameters Affecting Reduced SLL by SCA

In Fig (4.34). (a). (b). and (c). shown SCA test of the iterations, the population size, and current iterations. The effect of iteration in this algorithm is extremely winding as it is somewhat similar to the sawtooth and the best reduction of SLL at N = 8,16, and 32 is -23.1215 dB at iteration

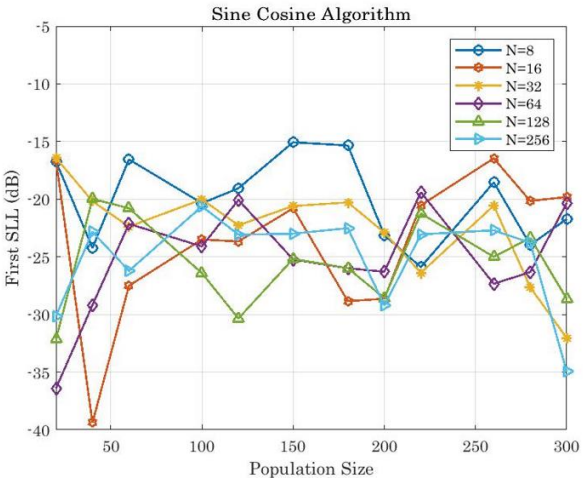
1000, -28.6351 dB at iteration 1000 and -35.5021 dB at iteration 440. At N = 64,128, and 256, the best diminishing SLL is -36.2625 dB at iteration 600, -32.7131 dB at iteration 780, and -34.1101 dB at iteration 45 respectively.

The effect of population size is very scattered is unstable as the best decrease of SLL at N = 8,16, and 32 is -24.2190 dB at population size 40, -39.3863 dB at population size 40, and -32.0400 dB at population size 300. At N = 64,128, and 256 the best SLL reduction is -36.4046 dB at population size 20, -32.1507 dB at population size 20, and -34.9485 dB at population size 300 respectively.

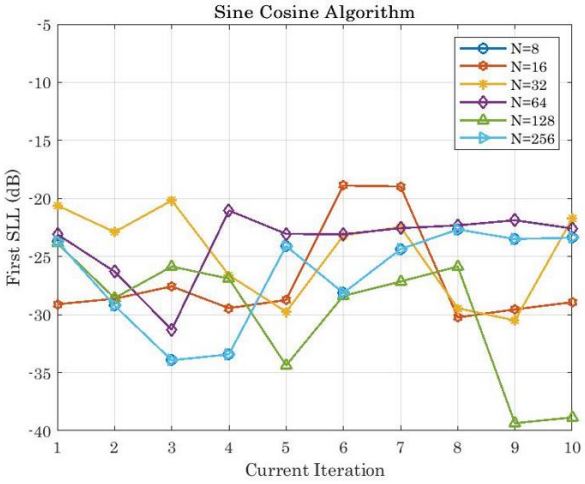
The current iteration effect is also winding and the best reduction of SLL at N = 8,16,32 is -30.4496 dB at current iteration 6, -30.2365 dB at current iteration 8, and -30.5028 dB at iteration 9. At N = 64,128, and 256, the best diminishing SLL is -31.3000 dB at current iteration 3, -39.3541 dB at current iteration 9, and -3.9338 dB at current iteration 3 respectively.



a. Iterations



b. Population size



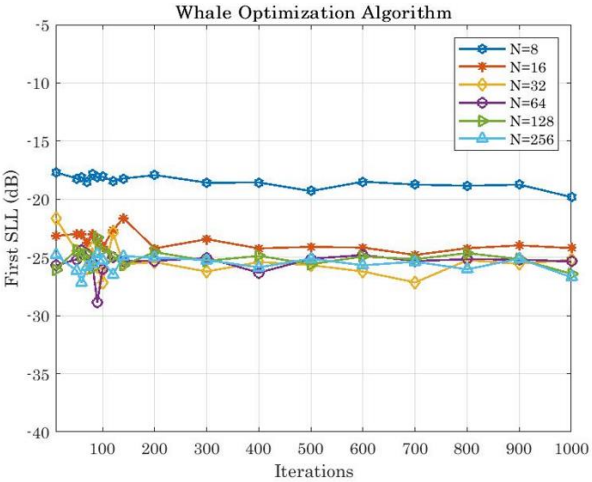
c. Current iterations

Figure (4.34). Affect parameters of SCA.

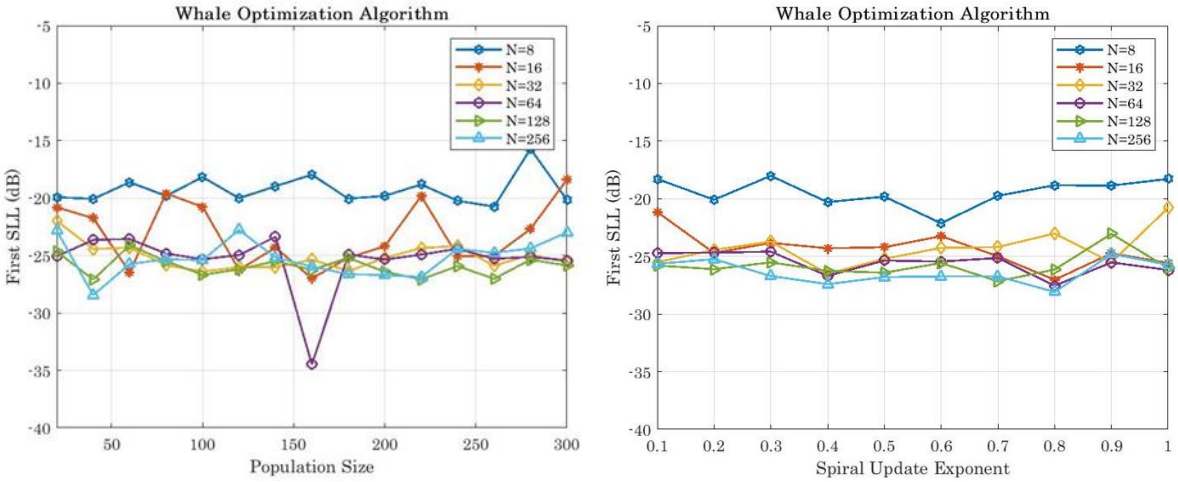
4.4.9 The Parameters Affecting Reduced SLL by WOA

In Fig (4.35). (a). (b). and (c). shown WOA test of the iterations, population size, and spiral update exponent. The effect of iteration in this algorithm is stable except the first iterations is slightly winding and the best SLL decreases at N = 64 by -28.8688 dB at iteration 90.

The effect of population size is zigzag in all antenna elements as the best reducing SLL obtained at N = 64 at population size 160 and by -34.4808 dB. and the effect of the spiral update exponent is slightly winding in its effect on SLL reduction and the highest amount to reduce it is -28.0820 at N = 256 at spiral update exponent is 2.3.



a. Iterations



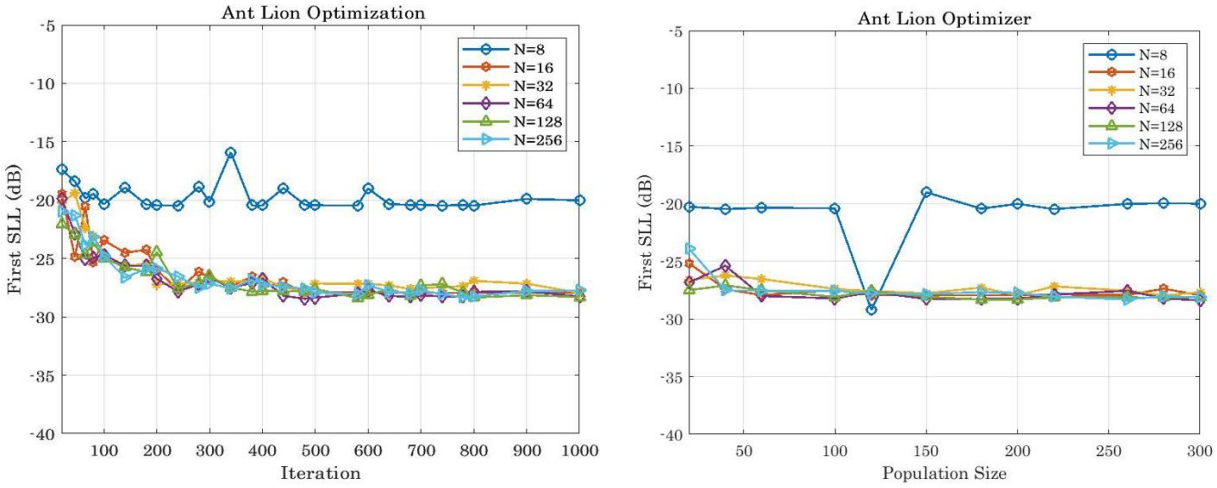
b. Population size **c. Spiral update exponent**

Figure (4.35). Affect parameters of WOA.

4.4.10 The Parameters Affecting Reduced SLL by ALO

In the ALO test of the iterations, and the population size. In Fig (4.36). (a) and (b). The effect at N = 8 is stable except at iteration 280 and the worst reduction of SLL is where its value is -18.8643 dB. At the rest of the values of the number of antenna elements, the effect of the iterations is stable, that is, the decrease of the SLL is gradual.

When the population size effect at N = 8 is stable except at population size 120 and is the most effective SLL reduction at -29.2334 dB. At all other values of the number of antenna elements, the effect of population size is stable, meaning that the SLL decreases progressively and with converging values.



a. Iterations **b. Population size**

Figure (4.36). Affect parameters of ALO.

4.5 Comparison With the Related Works

The following Table (4.2). represents a comparison between the proposed algorithms and some related previous works this comparison is based on the calculated reduction SLL of each algorithm.

Table (4.2). Show the comparison with previous studies

No.	Researcher Name	Type of antenna	Optimization Technique	Reduction SLLs	Best results
1.	Goswami and Mandal [10].	LAA	GA	-13.06 dB at N=12	When comparing GA with more than one algorithm shows it is better than reducing SLL to the maximum.
2.	Chatterjee et al [27].	LAA	PSO	-24.7854 dB at N=4	PSO Reduced SLL as far as possible.
3.	Ram et al [31].	TMLAA	PSO RGA	-35.21 dB at N=16 -34.89 dB at N=16	The results showed that PSO reduced SLL better than when using RGA by -35.21 dB.
4.	Saxena, and Kothari [32].	LAA	GA FPA	-20.72 dB at N=10 -23.45 dB at N=10	FPA is better than GA, reduced SLL at specific values but FPA outperformed GA where the value reached -23.45 dB.
5.	Saxena, and Kothari [33].	LAA	PSO GWO	-24.62 dB at N=10 -26.05 dB at N=10	GWO is the best because it has reduced SLL better than PSO.
6.	Das et al [35].	LAA	PSO MFO	-24.62 dB at N=10 -26.07 dB at N=10	MFO is the best because it has reduced SLL better than PSO.
7.	Borah et al [36].	LAA	PSO	-16.9560 dB at N=15	PSO Reduced SLL as far as possible.
8.	Li and Luk [37]	LAA	GWO PSO GA	-21.093 dB at N=37 -17.41 dB at N=10 -20.562 dB at N=37	Although the number of antenna elements varies, GWO is the best at N=37.
10.	Durmus, and Kurban [1]	CAA	MFO GA PSO	-17.79 dB at N=12 -11.80 dB at N=12 -13.67 dB at N=12	MFO is the best because it has minimized SLL to the maximum.
12.	Liang et al [15].	LAA	SSA WOA PSO	-23.9233 dB at N=16 -22.9487 dB at N=16 -23.7487 dB at N=16	SSA reduced SLL to -23.9233 dB, so it is considered the best, followed by PSO, and then WOA.
13.	Jumunaa et al [40].	LAA	PSO GWO MVO	-21.8430 dB at N=28 -22.1221 dB at N=28 -22.2765 dB at N=28	MVO is the best because it has reduced SLL more than PSO and GWO.
14.	Singh et al. [41].	CAA	SCA GWO SSA	-29.47 dB at N=20 -30.66 dB at N=20 -19.41 dB at N=20	GWO is the best because it has minimized SLL to the maximum.

15.	Hu et al [44].	PAA	GA PSO WOA	-15.03 dB at N=64 -15.31 dB at N=64 -18.52 dB at N=64	WOA reduced SLL to -18.52 dB so it is considered the best, followed by PSO, and then GA.
16.	Asianuba and Precious [46].	LAA	PSO	-17.26 dB at N=10	PSO Reduced SLL as far as possible.
17.	Ghattas et al. [47].	PAA	GWO PSO	-25.11 dB at N=10 -25.00 dB at N=10	GWO is better because it has reduced SLL more and to the maximum.
18.	This Thesis	LAA	PSO GA FPA GWO SSA MFO MVO SCA WOA ALO	-28.5405 dB at N=256 -28.6204 dB at N=256 -28.3071 dB at N=32 -28.3367 dB at N=256 -28.3453 dB at N=256 -28.3859 dB at N=128 -28.4366 dB at N=128 -29.2229 dB at N=256 -26.7006 dB at N=256 -28.3031 dB at N=128	Although the antenna elements differ to reach the best low SLL SCA is the best at N = 256.

The results obtained in the references above, some of which gave better results than the 10 algorithms due to the use of a different fitness function in addition to changing the values of the parameters and varying the number of antenna elements. Some of the above references when compared with the results of this thesis, the 10 algorithms have been distinguished by their superiority in reducing SLL to the greatest extent possible and for a different number of antenna elements .

5

CHAPTER FIVE

'CONCLUSION AND SUGGESTED FUTURE WORKS'

Chapter Five

Conclusion and Suggested Future Works

5.1 Conclusion

In this thesis, ten optimization algorithms were investigated to reduce the side lobes of the beam pattern and assemble the majority of the energy in the beam pattern without dispersing it into undesirable directions.

1. One particular fitness function for the 10 algorithms as well as changing the locations of antenna elements. SCA, FPA, and MVO were the most effective algorithms when evaluating algorithms for the various numbers of antenna elements N = 8,16,32,64,128, and 256. because they reduced SLL to the maximum amount and the least loss of power.

2. When using LAA with algorithms and changing the number of antenna elements, it affects the reduction of SLL, thereby sending a strong beam pattern with minimal energy loss, The maximum SLL reduction will alter the distribution position and amplitude of the elements in each element of the antenna array leading to high gain and direction.

3. In SCA the best reduced SLL is at N = 256, decreasing to -29.2229 dB, in FPA the best SLL reduction is -28.3071 dB at N = 32, and in MVO the best result at N = 128, reducing SLL to -28.4366 dB.

4. The parameters of each algorithm have been changed separately and their impact on SLL reduction has been studied. Each parameter has been shown to affect the amount of SLL reduction for each number of antenna elements and there is a set of parameters that have an infinite number of tests to reduce the side lobes. The best algorithm that has reduced SLL is SCA based on changing parameters, namely iteration, population size, and current iteration.

5. In FPA The effect of 4 parameters of iteration, population size, probability, and flower attraction rate were tested and it was found that the best SLL reduction of -32.5694 dB in iteration 300 at N=64, -34.5790 dB in population size 160 at N=16, -34.9451 dB in probability 1 at N=8, and -33.2830 dB in flower attraction rate 0.5 at N=16.

6. In WOA The effect of 3 parameters of iteration, population size, and spiral update exponent was tested and it was found that the best SLL reduction of -28.8688 dB in iteration 90 at N=64, -34.4808 dB in population size 160 at N=64, and -28.0820 dB in spiral update exponent 2.3 at N=256.

7. In SCA The effect of 3 parameters of iteration, population size, and current iteration was tested and it was found that the best SLL reduction of -36.2625 dB in iteration 600 at N=64, -39.3863 dB in population size 40 at N=16, and -39.3541 dB in current iteration 9 at N=128.

8. The best technique was SCA, where the effect of population size was the greatest on the drop in SLL, where it was reduced to -39.3863 dB at population size 40 at N = 16.

5.2 Suggested Future Works

1. It is possible to use a different objective function on the 10 algorithms or to use different algorithms on the same objective function. The difference in objective function depends on the system applied to optimization techniques.

2. Optimization Algorithms are likely to continue evolving, with potential improvements in efficiency, scalability, and adaptability to different problem domains. Researchers might explore new approaches like hybrid approaches that combine multiple techniques.

3. Optimization algorithms could be further optimized for specific hardware architectures or parallel processing to exploit the full potential of modern computing technologies, they may also be refined to handle large-scale optimization problems.

4. It is possible to change the values of the influential parameters to the desired application to reach the best results such as gain, direction and energy system.

References

[1] A. Durmus, and R. Kurban, "Optimum design of linear and circular antenna arrays using equilibrium optimization algorithm," *International Journal of Microwave and Wireless Technologies*., Vol. 13, No. 9, pp. 986 - 997, January 2021.

[2] C. A. Balanis, "Antenna theory: analysis and design cover the applications of antenna arrays, including direction finding, beamforming, and smart antennas", (4th ed.), Chapter 13, New Jersey: John Wiley & Sons, Inc., 2016.

[3] R. J. Mailloux, "Phased array antenna handbook," *Artech house, 2017.*

[4] K. Guney, A. Durmus, S. Basbug, "A Plant Growth Simulation Algorithm For Pattren Nulling of Linear Antenna Arrays By Amplitude Control," *Progress In Electromagnetics Research B*., Vol. 17, No., pp. 69–84, 2009.

[5] M. D. Renzo, H. Haas, A. Ghrayeb, S. Sugiura, and L. Hanzo, "Spatial Modulation for Generalized MIMO: Challenges, Opportunities, and Implementation," *Proceedings of the IEEE*, Vol. 102, No.1, pp. 56-103, January 2014.

[6] R. W. Heath, N. G. Prelcic, S. Rangan, W. Roh, and A. M. Sayeed, "An Overview of Signal Processing Techniques for Millimeter Wave MIMO Systems," *IEEE journal of selected topics in signal processing*, Vol. 10, No. 3, pp. 436-453, April 2016.

[7] R. M. Rial, C. Rusu, N. Prelcic, A. Alkhateeb, and R. W. H. Jr, "Hybrid MIMO Architectures for Millimeter Wave Communications: Phase Shifters or Switches?" *IEEE Access*., Vol. 4, pp. 247-267, January 2016.

[8] A. Chowdhury, R. Giri, A. Ghosh, S. Das, A. Abraham and V. Snasel, "Linear Antenna Array Synthesis using Fitness-Adaptive Differential Evolution Algorithm." In *IEEE Congress on Evolutionary Computation*, Barcelona, Spain, pp. 1-8, July 2010.

References

[9] J. Kennedy, and R. Eberhart, "Particle Swarm Optimization," *Proceedings of ICNN'95-international conference on neural networks*, Vol. 4, pp. 1942-1948, 1995.

[10] B. Goswami, and D. Mandal, "A genetic algorithm for the level control of nulls and side lobes in linear antenna arrays." *Journal of King Saud University-Computer and Information Sciences*., Vol. 25, No. 2, pp. 117-126, 2012.

[11] S. Mosca and M. Ciattaglia, "Ant Colony Optimization to Design Thinned Arrays," *2006 IEEE Antennas and Propagation Society International Symposium*. Albuquerque, NM, USA, pp. 4675-4678, October 2006.

[12] S. Mirjalili, "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm," *Knowledge-based systems*., Vol. 89, pp. 228-249, 2015.

[13] U. Singh, H. Kumar, and T. S. Kamal, "Linear array synthesis using biogeography-based optimization," *Progress In Electromagnetics Research*, Vol. 11, pp. 25-36, 2010

[14] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Expert Syst. Appl.*, Vol. 69, pp. 46-61, 2014.

[15] Q. Liang, B, Chen, H. Wu, C. Ma, and S. Li, "A Novel Modified Sparrow Search Algorithm with Application in Side Lobe Level Reduction of Linear Antenna Array," *Wireless Communications and Mobile Computing*., Vol. 2021, pp. 1-25, 2021.

[16] H. Wu, Y. Yan, C. Liu, and J. Zhang, "Pattern synthesis of sparse linear arrays using spider monkey optimization," *IEICE Transactions on Communications*, Vol. E100.B, No. 3, pp. 426– 432, 2017.

[17] A. E. Taser, K. Guney, and E. Kurt, "Circular Antenna Array Synthesis Using Multiverse Optimizer," *International Journal of Antennas and Propagation*., Vol. 2020, pp.1-10, 2020.

[18] G. Sun, Y. Liu, H. Li, S. Liang, A. Wang, and B. Li, "An antenna array sidelobe level reduction approach through invasive weed optimization," *International Journal of Antennas and Propagation*, Vol. 2018, pp. 1-16, 2018.

[19] S. Mirjalili, "The Ant Lion Optimizer," *Advances in engineering software*., Vol.83, pp. 80-98, 2015.

[20] D. Mandal, A. K. Bhattacharjee, and S. P. Ghoshal, "Linear Antenna Array Synthesis Using Improved Particle Swarm Optimization," In: *2011 Second International Conference on Emerging Applications of Information Technology*. *IEEE*., Kolkata, India, pp. 365-368, 2011.

[21] Š. Hajach, and B. Peter, "Antenna Arrays—Application Possibility For Mobile Communication." *Journal of Electrical Engineering.*, Vol. 53, pp. 332-335., 2002.

[22] P. F. McManamon et al, "A Review of Phased Array Steering for Narrow-Band Electrooptical Systems," *Proceedings of the IEEE*., Vol. 97, No.6, pp. 1078-1096, 2009.

[23] R. L. Haupt, "Antenna arrays: a computational approach," John Wiley & Sons, 2010.

[24] R.Tej D, S. K Ch K, and S. K. Kotamraju, "An improved peak side lobe reduction method for linear array antenna for military applications," *International Journal of Intelligent Computing and Cybernetics*, Vol. 14, No. 2, pp. 186-198, 2021.

[25] S. Binitha, and S. S. Sathya, "A Survey of Bio inspired Optimization Algorithms," *International journal of soft computing and engineering*., Vol. 2, Issue. 2, 2012.

[26] A. B. Gabis, Y. Meraihi, S. Mirjalili, A. R. Cherif, "A comprehensive survey of sine cosine algorithm: variants and applications," *Artificial Intelligence Review*, Vol. 54, No. 7, pp. 5469-5540, 2021.

[27] S. Chatterjee, S. Chatterjee, and D. R. Poddar, "Side Lobe Level Reduction of a Linear Array using Chebyshev Polynomial and Particle Swarm Optimization." *International Journal of Computer Applications,* pp. 0975-8887, 2013.

[28] L. Pappula and D. Ghosh, "Linear antenna array synthesis using cat swarm optimization", *AEU-International Journal of Electronics and Communications*, Vol. 68, No.6, pp. 540-549, 2014.

[29] G. S. K. Gayatri Devi, G. S. N. Raju, and P. V. Sridevi, "Application of Genetic Algorithm for Reduction of Sidelobes from Thinned Arrays," *Adv. Model. Anal. B.*, Vol. 58, No. 1, pp. 35-52, 2015.

[30] V. S. Gangwar, A. K. Singh, E. Thomas, S. P. Singh, " Side Lobe Level Suppression in a Thinned Linear Antenna Array Using Particle Swarm Optimization." In *2015 International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT),* pp. 787-790, Davangere, India, 2015.

[31] G. Ram, D. Mandal, R. Kar, and S. P. Ghoshal, "Directivity maximization and optimal far-field pattern of time modulated linear antenna arrays using evolutionary algorithms," *AEU-International Journal of Electronics and Communications*., Vol. 69, No. 12, pp. 1800-1809, 2015.

[32] P. Saxena, and A. Kothari, "Linear antenna array optimization using flower pollination algorithm," *SpringerPlus*, Vol. 5, pp. 1-15, 2016.

[33] P. Saxena and A. Kothari, "Optimal Pattern Synthesis of Linear Antenna Array Using Grey Wolf Optimization Algorithm", *International Journal of Antennas and Propagation*, pp.11, 2016.

[34] P. Saxena and A. Kothari, "Ant Lion Optimization algorithm to control side lobe level and null depths in linear antenna arrays." *AEU-International Journal of Electronics and Communications*, Vol. 70, No.9, pp. 1339-1349, 2016.

[35] A. Das, D. Mandal, S.P. Ghoshal, and R. Kar, "Moth flame optimization based design of linear and circular antenna array for side lobe reduction," International Journal of Numerical Modelling: Electronic Networks, Devices and Fields., Vol. 32. No.1, September 2018.

[36] S. S. Borah, A. Deb, and J. S. Roy, "Design of Thinned Linear Antenna Array using Particle Swarm Optimization (PSO) Algorithm", *Asian Journal for Convergence In Technology (AJCT) ISSN*, pp.2350-1146, 2019.

[37] X. Li, and K. M. Luk, "The Grey Wolf Optimizer and its Applications in Electromagnetics," *IEEE transactions on antennas and propagation*., Vol. 68, No. 3, pp. 2186-2197, 2020.

[38] M. Supriya, and G. V. S. P. Rao, "Comparative Analysis of Optimization Techniques Used In Linear Antenna Array Design," *International Journal of Engineering Research and Applications.*, Vol. 10, No. 7, pp. 07-11, July 2020.

[39] Y. Albagory and F. Alraddady, "An Efficient Approach for Sidelobe Level Reduction Based on Recursive Sequential Damping," *Symmetry*., Vol. 13, No. 3, pp. 480, February 2021.

[40] D. Jamunaaa, G.K. Mahantia, and F.N. Hasoonb, "Multi-verse optimization algorithm for optimal synthesis of phase-only reconfigurable linear array of mutually coupled parallel half-wavelength dipole antennas placed at finite distances from the ground plane", *Scientia Iranica*, Vol.29, No.4, pp. 1915-1924, 2022.

[41] H. Singh, S. Singh, A. Gupta, H. Singh, A. Gehlot, · J. Kaur, "Design and synthesis of circular antenna array using artificial hummingbird optimization algorithm", Journal of Computational Electronics, Vol. 21, No. 6, pp. 1293-1305, 2022.

[42] Z. Li, and J. Ouyang, "Optimal Design Method for Sub-array Beamforming Based on an Improved Ant Lion Algorithm", *Conference Series*, Vol. 2437, No. 1, pp. 012106, IOP Publishing, 2023.

[43] J. R. Mohammed, "Phased Sub-arrays Pattern Synthesis Method with Deep Sidelobe Reduction and Narrow Beam Width", IETE Journal of Research, Vol.69, No. 2, pp. 1081-1087, 2023.

[44] H. Hu, H. Li, L. Zhao, X. Jiang, J. Yang, and Y. Zhang, "Low-Sidelobe Pattern Synthesis for Spaceborne Array Antenna Based on Improved Whale Optimization Algorithm," *Applied Sciences*., Vol. 13, No. 5, pp. 2841, 2023.

[45] Q. Liang, H.Wu, and B. Chen, " Design of Linear and Circular Antenna Arrays for Side Lobe Reduction Using a Novel Modified Sparrow Search Algorithm", *Wireless Personal Communications*, Vol. 130, No. 2, pp. 1045-1069, 2023.

[46] I. Asianuba, D. Precious, "Cockroach Swarm Optimization for Side Lobe Level Reduction in Linear Array Antenna", *International Journal of Electronics and Communication Engineering*, Vol.10, Issue. 3, pp. 23-29, ISSN: 2348-8549, March 2023.

[47] N. Ghattas, A. M. Ghuniem, A. A. Abdelsalam, and A. Magdy, "Planar Antenna Arrays Beamforming Using Various Optimization Algorithms," *IEEE Access*., Vol. 11, pp. 68486-68500, 2023.

[48] W. L. Stutzman, G. A. Thiele, "Antenna Theory and Design," 3rd ed. Hoboken: John Wiley & Sons, Inc., 2013.

[49] S. J. Orfanidis, "Electromagnetic Waves and Antennas," New Jersey: Rutgers University, 2004.

[50] P. Pal, and D. Mondal, "Adaptive Antenna Array Pattern Synthesis for Suppressed Sidelobe Level and Controlled Null Using Genetic Algorithm," in Conf. *2014 International Conference for Convergence for Technology*., Pune, India, 2014, pp. 1-6.

[51] L. A. Greda, A. Winterstein, D. L. Lemes, and M. V. T. Heckler, "Beamsteering and Beamshaping Using a Linear Antenna Array Based on Particle Swarm Optimization," *IEEE Access*., Vol. 7, pp. 141562-141573, October 2019.

[52] F. J. A. Pena, J. A. R. Gonzalez, E. V. Lopez, and S. R. Rengarajan, "Genetic Algorithms in the Design and Optimization of Antenna Array Patterns," *IEEE Transactions on Antennas and Propagation*., Vol.47, No. 3, pp. 506-510, March 1999.

[53] S. Liang, Z. Fang, G. Sun, Y. Liu, G. Qu, and Y. Zhang, "Sidelobe Reductions of Antenna Arrays via an Improved Chicken Swarm Optimization Approach," *IEEE Access.*, Vol.8, pp. 37664-37683, February 2020.

[54] J. A. Bickford, A. E. Duwel, M. S. Weinberg, R. S. McNabb, D. K. Freeman, and P. A. Ward, "Performance of Electrically Small Conventional and Mechanical Antennas," *IEEE Transactions on Antennas and Propagation.*, Vol. 67, No. 4, pp. 2209 - 2223, April 2019.

[55] U.A. Bakshi, A.V. Bakshi, K. A. Bakshi. Antenna & Wave Propagation. Technical Publications Pune, 2009.

[56] Y. S. Wen, and N. Z. Ping, "A Review of the Four Dimension Antenna Arrays," *Journal of Electronic Science and Technology of China*., Vol. 4, No. 3, pp. 193-201, September 2006.

[57] V. Trees, H. L, "Optimum array processing: Part IV of detection, estimation, and modulation theory," John Wiley & Sons, 2002.

[58] A. Alkhateeb, O. E. Ayach, G. Leus, and R. W. H. Jr, "Channel Estimation and Hybrid Precoding for Millimeter Wave Cellular Systems," *IEEE Journal of Selected Topics in Signal Processing*., Vol.8, Issue.5, pp. 831-846, 01 July 2014.

[59] R. L. Haupt, "Antenna Arrays-A Computational Approach," New Jersey: John Wiley & Sons, 2010.

[60] M. Moubadir, A. Mchbal, N. A. Touhami, and M.Aghoutane, "A Switched Beamforming Network for 5G Modern Wireless Communications Applications," *Procedia Manufacturing*., Vol. 32, pp. 753-761, 2019.

[61] J. A. Bickford, A. E. Duwel, M. S. Weinberg, R. S. McNabb, D. K. Freeman, and P. A. Ward, "Performance of Electrically Small Conventional and Mechanical Antennas," *IEEE Transactions on Antennas and* Propagation., Vol. 67, No. 4, pp. 2209 - 2223, April 2019.

[62] C. To Tai, and C. Pereira, "An approximate formula for calculating the directivity of an antenna," *IEEE Transactions on Antennas and Propagation*, Vol. 24, No. 2, pp. 235-236, 1976.

[63] J. W. Rajashree. "Study on Nature Based Search Algorithms for Solving Optimization Problems with Emphasis on Application to Electromagnetic Antenna Arrays." PhD desertation, Symbiosis International University, Pune, 2013.

[64] H. Asaad and S. S. Hreshee, "A Review About The Optimization Algorithm for SLL Reduction in PAA", *2022 3rd Information Technology To Enhance e-learning and Other Application (IT-ELA)*, pp. 215-221, Baghdad, Iraq, 2022.

[65] P. Nandi, and J. S. Roy, "Side Lobe Reduction of Phased Array Antenna using Genetic Algorithm and Particle Swarm Optimization*," International Journal of Microwave And Optical Technology*, Vol.11, No.3, May 2016.

[66] A. Recioui, "Sidelobe Level Reduction in Linear Array Pattern Synthesis Using Particle Swarm Optimization", *Journal of Optimization Theory and Applications*, Vol. 153, pp.497-512, 2012.

[67] M. M. Khodier, and C. G. Christodoulou, "Linear array geometry synthesis with minimum sidelobe level and null control using particle swarm optimization," *IEEE Transactions on Antennas and propagation*., Vol.53, No. 8, pp. 2674-2679, August 2005.

[68] J. Robinson and Y. R. Samii, "Particle swarm optimization in electromagnetics," *IEEE Transactions on Antennas and Propagation*., Vol. 52, No. 2, pp.397-407, February 2004.

[69] Y. Shi, and R. Eberhar, "A modified particle swarm optimizer," In *1998 IEEE international conference on evolutionary computation proceedings. IEEE world congress on computational intelligence*., pp.69-73, Anchorage, Alaska, USA, May 1998.

[70] C. Y. Chen, and F. Ye, "Particle swarm optimization algorithm and its application to clustering analysis," In *2012 Proceedings of 17th Conference on Electrical Power Distribution*. *IEEE*., Tehran, Iran, pp. 36-37, May 2012.

[71] M. A. Panduro, A. L. Mendez, R. Dominguez, and G. Romero, "Design of non-uniform circular antenna arrays for side lobe reduction using the method of genetic algorithms", *AEU-International Journal of Electronics and Communications*, Vol. 60, No. 10, pp. 713-717, 2006.

[72] S. Katoch, S. S. Chauhan, and V. Kumar, "A review on genetic algorithm: past, present, and future," *Multimedia tools and applications.*, Vol. 80, pp. 8091-8126, October 2021.

[73] S. Kiehbadroudinezhad, N. K. Noordin1, A. Sali, and Z. Z. Abidin, "OPTIMIZATION OF AN ANTENNA ARRAY USING GENETIC ALGORITHMS," *The Astronomical Journal.*, Vol.147, No. 6, pp. 13, June 2014.

[74] X. S. Yang, "Flower pollination algorithm for global optimization," International *conference on unconventional computing and natural computation*. Berlin, Heidelberg: Springer Berlin Heidelberg, Vol.7445, pp. 240-249, 2012.

[75] X. S. Yang, M. Karamanoglu, X. He, "Flower pollination algorithm: a novel approach for multiobjective optimization," *Engineering optimization.*, Vol.46, No. 9, pp.1222-1237, October 2013.

[76] C. Zhang, and S. Ding, "A stochastic configuration network based on chaotic sparrow search algorithm," *Knowledge-Based Systems*., Vol. 220, pp. 106924, 2021.

[77] J. Xue, B. Shen, and A. Pan, "An intensified sparrow search algorithm for solving optimization problems," *Journal of Ambient Intelligence and Humanized Computing*, Vol. 14, No. 7, pp. 9173-9189, 2023.

[78] Q. Liang, B. Chen, H. Wu, C. Ma, and S. Li, "A novel modified sparrow search algorithm with application in side lobe level reduction of linear antenna array," *Wireless Communications and Mobile Computing.*, Vol. 2021, pp. 1-25, June 2021.

[79] T. J. Smith and M. H. Villet, "Parasitoids associated with the diamondback moth, *Plutella xylostella (*L.)," in the Eastern Cape, South Africa." *The Management of Diamondback Moth and Other Crucifer Pests. Proceedings of the Fourth International Workshop on Management of the Diamondback Moth and Other Crucifer Pests. Department of Natural Resources and Environment. Melbourne, Australia*, pp. 249–253, 2001.

[80] K. J. Gaston, J. Bennie, T. W. Davies, and J. Hopkins, "The ecological impacts of nighttime light pollution: a mechanistic appraisal," *Biological reviews.*, Vol. 88, No. 4, pp. 912-927, April 2013.

[81] K. D. Frank, C. Rich, and Longcore, "Effects of artificial night lighting on moths," *Ecological consequences of artificial night lighting.*, Vol.13, pp.305-344, 2006.

[82] C. Li, Z. Niu, Z, Song, B, Li, J. Fan, and P. X. Liu, "A Double Evolutionary Learning Moth-Flame Optimization for Real-Parameter Global Optimization Problems," *IEEE Access.*, Vol.6, pp. 76700-76727, November 2018.

[83] A. E. Taser, K. Guney, and E. Kurt, "Circular antenna array synthesis using multiverse optimizer," *International Journal of Antennas and Propagation*., Vol. 2020, pp. 1-10, Aug 2020.

[84] D. M. Eardley, "Death of White Holes in the Early Universe," *Physical Review Letters.*, Vol. 33, No. 7, pp. 442, Aug. 1974.

[85] P. C. W. Davies, "Thermodynamics of black holes," *Reports on Progress in Physics*., Vol. 41, No. 8, pp. 1313, January 1978.

[86] A. H. Guth, "Eternal inflation and its implications," *Journal of Physics A: Mathematical and Theoretical*., Vol. 40, No. 25, pp. 6811–6826, June 2007.

[87] P. J. Steinhardt, and N. Turok, "The cyclic model simplified," *New Astronomy Reviews.*, Vol. 49, No. 2-6, pp. 43-57, May 2005.

[88] S. M. Mirjalili, S. Z. Mirjalili, S. Saremi, and S. Mirjalili, "Sine Cosine Algorithm: Theory, Literature Review, and Application in Designing Bend Photonic Crystal Waveguides," *Nature-inspired optimizers: theories, literature reviews and applications.*, Vol. 811, pp. 201-217, 2020.

[89] A. Banerjee, and M. Nabi, "Re-Entry Trajectory Optimization for Space Shuttle using Sine-Cosine Algorithm," In *2017 8th international conference on recent advances in space technologies (RAST). IEEE.*, Istanbul, Turkey, pp. 19-22, Jun 2017.

[90] H. Chen, A. A. Heidari, X. Zhao, L. Zhang, and H. Chen, "Advanced orthogonal learning-driven multi-swarm sine cosine optimization: Framework and case studies," *Expert Systems with Applications*., Vol. 144, pp. 113113, 15 April 2020.

[91] A. Fatlawi, A. Vahedian, and N. K. Bachache, "Optimal Camera Placement Using Sine-Cosine Algorithm," In *2018 8th International Conference on Computer and Knowledge Engineering (ICCKE).*, *Mashhad, Iran*, October 2018.

[92] S. Mirjalili, "SCA: A Sine Cosine Algorithm for solving optimization problems", *Knowledge-based systems*, Vol. 96, pp. 120-133, March 2015.

[93] C. Zhang, X. Fu, L. P. Ligthart, S. Peng, and M. Xie, "Synthesis of Broadside Linear Aperiodic Arrays With Sidelobe Suppression and Null

Steering Using Whale Optimization Algorithm," *IEEE Antennas and Wireless Propagation letters.*, Vol.17, No. 2, pp. 347-350, February 2018.

[94] M. A. Almagboul, F. Shu, Y. Qian, X. Zhou, J. Wang, and J. Hu, "Atom search optimization algorithm-based hybrid antenna array receive beamforming to control sidelobe level and steering the null," *AEU-International Journal of Electronics and Communications.*, Vol. 111, pp.152854, November 2019.

[95] S. Mirjalili, and A. Lewis, "The Whale Optimization Algorithm," *Advances in engineering software*., Vol. 95, pp. 51-67, May 2016.

[96] L. Abualigah, M. Shehab, M. Alshinwan, S. Mirjalili , and M. Abd Elaziz, "Ant Lion Optimizer: A Comprehensive Survey of Its Variants and Applications," *Archives of Computational Methods in Engineering*., Vol. 28, pp. 1397-1416, 2021.

[97] A. A. Heidari, H. Faris, S. Mirjalili, I. Aljarah, and M. Mafarja, " Ant Lion Optimizer: Theory, Literature Review, and Application in Multi-layer Perceptron Neural Networks," *Nature-Inspired Optimizers: Theories, Literature Reviews and Applications.*, Vol.811, pp. 23–46, February 2019.

الخلاصة

من الممكن بناء مصفوفات هوائيات مثل مصفوفات الهوائيات الخطية (LAA) بدقة وبتصميم مثلى، مع خصائص مثل التوجيه الممتاز، وزيادة الإشارة، والتغطية الواسعة. وتعتبر إحدى السمات الهامة في تصميم مصفوفات الهوائيات هي تقليل مستوى الفصوص الجانبية(SLL) ؛ لأنها تعتبر عاملًا هامًا في تطبيقات الكهرومغناطيسية ذات التداخل المنخفض، مثل أنظمة الرادار وأنظمة الاتصالات والشبكات اللاسلكية. يمكن تحقيق ذلك من خلال التحكم والتغيير في السعة وتباعد العناصر والموقع أو الطور.

في هذه الرسالة، تم تطبيق عشر تقنيات تحسين لتقليل مستوى الفصوص الجانبية (SLL). هذه التقنيات هي تقنية تحسين سرب الجسيمات (PSO)، وخوارزمية الجينات (GA)، وخوارزمية تلقيح الزهور (FPA)، وتحسين ذئب الرمادي (GWO)، وخوارزمية البحث عن الطيور (SSA)، وتحسين لهب العث (MFO)، وتحسين الكون المتعدد (MVO)، وخوارزمية التمام والجيب تمام (SCA)، وخوارزمية تحسين الحوت (WOA)، وتحسين أسود النمل (ALO). تم تنفيذ مصفوفة الهوائيات الخطية المستخدمة في هذا الضغط بأشكال هندسية مختلفة من 8، 16، 32، 64، 128، و256 عنصرًا. بالإضافة إلى ذلك، تم دراسة تأثير تغيير قيم المعلمات لكل تقنية، وتشمل هذه المعلمات التكرار، حجم السكان، أقصى تكرار للتوقف، وما إلى ذلك، والتي تهدف جميعها إلى تقليل مستوى الفصوص الجانبية بشكل أفضل.

تحقيق النتائج باستخدام حزمة البرمجيات (MATLAB)، في النموذج الأول، أعطت جميع الخوارزميات أفضل مستوى للفصوص الجانبية في عدد معين من عناصر مصفوفة الهوائيات. أظهرت جميع الخوارزميات نتائج أدت إلى التقليل الأقصى لمستوى الفصول الجانبية من خلال توزيع عناصر الهوائيات، ولجميع أعدادها بشكل مثالي، حيث تتفوق خوارزمية SCA على باقي الخوارزميات عند N=256 من خلال تقليل مستوى الفصول الجانبية إلى -29.2229 ديسيبل. وذلك لأن العناصر يتم توزيعها بشكل أفضل من باقي التقنيات وتتغير هذه النتائج وفقًا للظروف الخاصة بإعادة ترتيب العناصر حسب نوع الهوائيات المستخدمة.

نتائج الاختبار الثاني تظهر أن تغيير المعلمات يؤثر على تقليل مستوى الفصوص الجانبية على الرغم من أن بعض المعلمات لها تأثير ضئيل، حيث يغير هذا التأثير مواقع الهوائيات لتحسين تقليل مستوى الفصول الجانبية. وتتغير هذه النتائج وفقًا لظروف معينة تمثلها تغيير هندسة مصفوفة الهوائيات أو تغيير عدد عناصر الهوائي..

جمهورية العراق
وزارة التعليم العالي والبحث العلمي
جامعة بابـل
كلية الهندسة
قسم الهندسة الكهربائية

خفض مستوى الفصوص الجانبية لمصفوفة الهوائي الخطي باستخدام خوارزميات التحسين الذكي

رسالة مقدمـــة الى كليـــة الهندســـة / جامعـــــة بابـــل

كجزء من متطلبات نيل درجـــــــة الماجستير في الهندسة /

الهندسة الكهربائية/ الاتصالات

من قبـل

هدى أسعد عبد الأمير علي

بإشـــراف

أ.د. سعد سفاح حسون

١٤٤٥ هـ ٢٠٢٤ م