T.C.
KAYSERİ ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

YAPAY ZEKA OPTİMİZASYON ALGORİTMALARI İLE ANTEN DİZİ SENTEZİ

Hazırlayan
ZAFER YILDIRIM

Danışman
Dr. Öğr. Üyesi ALİ DURMUŞ

Yüksek Lisans Tezi

Temmuz, 2022
KAYSERİ

T.C.
KAYSERİ ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

YAPAY ZEKA OPTİMİZASYON ALGORİTMALARI İLE ANTEN DİZİ SENTEZİ

(Yüksek Lisans Tezi)

Hazırlayan
Zafer YILDIRIM

Tez Danışmanı
Dr. Öğr. Üyesi Ali DURMUŞ

Kayseri Üniversitesi Bilimsel Araştırma Projeleri Koordinatörlüğü tarafından FLY-2021-1050 proje numarası ile desteklenmektedir.

Temmuz, 2022
KAYSERİ

BİLİMSEL ETİĞE UYGUNLUK

Tezimde yaptığım tüm çalışmalar ve bu çalışmalar için yer alan bilgiler akademik ve etik kurallar çerçevesinde elde edilmiştir. Bu kuralların gerektirdiği üzere tezimin özünde olmayan tüm sonuç ve bilgileri referans ekleyerek bildirdim.

İmza

Zafer YILDIRIM

TEZ İNTİHAL RAPORU TESLİM FORMU

TC.

KAYSERİ ÜNİVERSİTESİ

Lisansüstü Eğitim Enstitüsü Müdürlüğü

Anabilim Dalı: Elektrik-Elektronik Mühendisliği

Program Adı: Yüksek Lisans

Tez Başlığı: Yapay Zeka Optimizasyon Algoritmaları ile Anten Dizi Sentezi

Yukarıda bilgileri verilen tez çalışmasının a) Giriş, b) Ana bölümler ve c) Sonuç kısımlarından oluşan (Kapak, Önsöz, Özet, İçindekiler ve Kaynakça hariç) toplam 95 sayfalık kısmına ilişkin 27/06/2022 tarihinde ***Turnitin*** intihal programından aşağıda belirtilen filtreleme uygulanarak alınmış olan özgünlük raporuna göre tezin benzerlik oranı**: % 3'** tür.

Filtrelemeye **alıntılar dahil** edilmiştir. Filtrelemede **yedi (7) kelimeden daha az** örtüşme içeren metin kısımları hariç tutulmuştur.

Kayseri Üniversitesi Lisansüstü Eğitim Enstitüsü Tez İntihal Raporu Uygulama Esaslarını inceledim ve bu uygulama esaslarında belirtilen azami benzerlik oranlarına göre tez çalışmasının herhangi bir intihal içermediğini, aksinin tespit edilmesi durumunda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Gereğini bilgilerinize arz ederim.

…../…../2022

Danışman	Hazırlayan
Dr. Öğr. Üyesi Ali DURMUŞ	Zafer YILDIRIM

YÖNERGEYE UYGUNLUK

"Yapay Zeka Optimizasyon Algoritmaları ile Anten Dizi Sentezi" başlıklı Yüksek Lisans Tezi, Kayseri Üniversitesi Lisansüstü Eğitim Enstitüsü Tez Önerisi ve Tez Yazma Yönergesine uygun olarak hazırlanmıştır.

Hazırlayan	**Danışman**
Zafer YILDIRIM	Dr. Öğr. Üyesi Ali DURMUŞ

Elektrik-Elektronik Mühendisliği ABD Başkanı

Doç. Dr. Engin Cemal MENGÜÇ

KABUL VE ONAY TUTANAĞI

Dr. Öğr. Üyesi Ali DURMUŞ danışmanlığında **Zafer YILDIRIM** tarafından hazırlanan **"Yapay Zeka Optimizasyon Algoritmaları ile Anten Dizi Sentezi"** adlı bu çalışma jürimiz tarafından Kayseri Üniversitesi Lisansüstü Eğitim Enstitüsü **Elektrik-Elektronik Mühendisliği** Anabilim Dalı'nda **Yüksek Lisans Tezi** olarak kabul edilmiştir.

…../…../2022

JÜRİ:

Danışman :

Üye :

Üye :

ONAY:

Bu tezin kabulü Enstitü Yönetim Kurulu'nun …. /…./…… tarih ve …….. sayılı kararı ile onaylanmıştır.

……. /…..…../ ……..…

Doç. Dr. Engin Cemal MENGÜÇ

Enstitü Müdürü

TEŞEKKÜR

Tez konusunun belirlenmesinden tezin son aşamasına gelene kadar bana yol gösteren, yoğun iş temposu arasında, çok değerli vakitlerini ayırarak bana destek ve yardımlarını esirgemeyen, akademik kariyerime başladığım günden beri tecrübeleriyle bana ışık tutan, hiçbir yardımı esirgemeyen Tez Danışmanım saygıdeğer hocam Dr. Öğr. Üyesi Ali DURMUŞ'a teşekkür ederim.

Olumlu ve yapıcı eleştirileriyle beni yönlendiren ve tecrübelerini benimle paylaşan değerli hocam, Prof. Dr. Ercan KARAKÖSE'ye ve bilgi birikimleriyle her zaman kendilerinden çok şeyler öğrendiğim ve tezimin araştırma kısmında analizleri yapmamda büyük destek olan değerli Dr. Öğr. Üyesi Rıfat KURBAN hocama, teşekkürlerimi sunarım.

Tezimin başından sonuna kadar desteklerini esirgemeyen, tezimi okuyup gereken düzeltmeleri yapmamda yardımcı olan ve beni motive eden dayım Dr. Fatih KOCAOĞLU'na, manevi olarak hep yanımda olan Dr. Öğr. Üyesi Mehmet KUZU'ya akademik kariyerime devam etmemde yardımcı olan ve en zor günlerimde verdiği desteklerle her zaman yanımda olan annem ve babam başta olmak üzere tüm aileme emekleri için çok teşekkür ederim.

YAPAY ZEKA OPTİMİZASYON ALGORİTMALARI İLE ANTEN DİZİ SENTEZİ

Zafer YILDIRIM

Kayseri Üniversitesi Lisansüstü Eğitim Enstitüsü,
Yüksek Lisans Tezi, Temmuz 2022

Danışman: Dr. Öğr. Üyesi Ali DURMUŞ

ÖZET

Bu tezde, literatüre yeni kazandırılmış yapay zeka optimizasyon algoritmaları kullanılarak doğrusal, dairesel ve eşmerkezli dairesel anten dizilerinin optimum ışıma diyagramları elde edilmiştir. Anten dizileri ile uzun mesafelere etkin ve verimli bir şekilde haberleşme sistemi kurmak için ışıma diyagramları dar yarı demet güç genişliğine ve olabildiğince bastırılmış yan demet seviyesine (SLL) sahip olmalıdır. Yan demetlerin bastırılması, aynı frekansta çalışan diğer sistemlerden gelen elektromanyetik paraziti engellemek için temel bir gerekliliktir. Dar yarı demet güç genişliği ise, anten dizilerinde yüksek yönlülük sağlamak için oldukça önemlidir. Bu yüksek yönlülüğü ve bastırılmış SLL elde edebilmek için optimize edilmesi gereken yer, genlik, elemanlar arası mesafe, faz gibi anten dizisine ait birçok parametre bulunmaktadır.

Bu çalışmada, anten dizilerinin konum, genlik ve elemanlar arası mesafe değerleri, literatürde oldukça iyi bilinen Yapay Arı Koloni Algoritması ve Parçacık Sürü Optimizasyon metodu ile literatüre yeni kazandırılmış yapay zeka optimizasyon metotlarından; Arşimet Optimizasyon Algoritması, Atomik Yörünge Araması, Kristal Yapı Algoritması, Kertenkele Sürü Algoritması, Bal Porsuğu Algoritması, Lichtenberg Algoritması kullanılarak optimum olarak belirlenmiştir. Tezde, anten dizilerini optimize edebilme yeteneğine sahip hızlı ve esnek bir arama algoritmasını tespit etmek için farklı metotlar kullanılarak doğrusal, dairesel ve eşmerkezli dairesel anten dizi problemleri üzerinde çalışmalar yapılmıştır. Yapılan çalışmalar sonucunda önerilen algoritmaların sonuçları literatürdeki metotların sonuçlarıyla karşılaştırılmış ve önerilen metotların farklı anten dizi sentez problemlerinde oldukça başarılı olduğu tespit edilmiştir.

Anahtar Kelime: Anten Dizi Sentezi, Yan Demet Seviyesi Bastırma, Yapay Zeka Optimizasyon Algoritmaları, Hesaplamalı Elektromanyetik.

ANTENNA ARRAY SYNTHESIS WITH ARTIFICIAL INTELLIGENCE OPTIMIZATION ALGORITHMS

Zafer YILDIRIM

Kayseri University Graduate Education Institute

M. S. C. Thesis, July2022

Supervisor: Asst. Prof. Ali DURMUŞ

ABSTRACT

In this thesis, optimum radiation patterns of linear, circular, and concentric circular antenna arrays are obtained by using newly introduced artificial intelligence optimization algorithms. In order to establish an effective and efficient communication system with antenna arrays over long distances, radiation pattern should have a narrow half power beam width (HPBW) and a suppressed sidelobe level (SLL) as much as possible. Suppression of sidelobe is a basic requirement to avoid electromagnetic interference from other systems operating at the same frequency. The narrow HPBW is very important for high directivity in antenna arrays. In order to achieve this high directivity and suppressed SLLs, there are many parameters of the antenna array such as position, amplitude, distance between elements, phase, which need to be optimized.

In this study, the position, amplitude, and distance between the elements of the antenna arrays are determined as optimum using ABC, PSO which is well known in the literature, and newly introduced Archimedes Optimization Algorithm, Atomic Orbital Search, Crystal Structure Algorithm, Chameleon Swarm Algorithm, Honey Badger Algorithm, Lichtenberg Algorithm. In the thesis, studies have been carried out on linear, circular, and concentric circular antenna array problems using different methods to detect a fast and flexible search algorithm capable of optimizing antenna arrays. As a result of the studies, the results of the proposed algorithms are compared with the results of the methods in the literature, and it is determined that the proposed methods are quite successful in different antenna array synthesis problems.

Keywords: Antenna Array Synthesis, Sidelobe Level Suppression, Artificial Intelligence Optimization Algorithms, Computational Electromagnetics.

İÇİNDEKİLER

YAPAY ZEKA OPTİMİZASYON ALGORİTMALARI İLE ANTEN DİZİ SENTEZİ

BİLİMSEL ETİĞE UYGUNLUK ..... **i**

TEZ İNTİHAL RAPORU TESLİM FORMU ..... **ii**

YÖNERGEYE UYGUNLUK ..... **iii**

KABUL VE ONAY TUTANAĞI ..... **iv**

TEŞEKKÜR ..... **v**

ÖZET ..... **vi**

ABSTRACT ..... **vii**

İÇİNDEKİLER ..... **viii**

KISALTMA LİSTESİ ..... **x**

TABLO LİSTESİ ..... **xii**

ŞEKİL LİSTESİ ..... **viv**

GİRİŞ **1**

BİRİNCİ BÖLÜM
ANTEN DİZİSİ VE YAPAY ZEKA OPTİMİZASYON ALGORİTMALARI

1. Antenler……..…….3
   1.1. Tezin Amacı……...……...…...4
   1.2. Yapay Zeka Optimizasyon Algoritmaları…….……..……5
      1.2.1. Sürü Zekasına Dayalı Yapay Zeka Optimizasyon Algoritmaları…….6
         1.2.1.1. Parçacık Sürü Optimizasyonu……...….....…….…….…7
         1.2.1.2. Yapay Arı Kolonisi Algoritması……...9
         1.2.1.3. Bal Porsuğu Algoritması…….……12
         1.2.1.4. Bukalemun Sürüsü Algoritması…….…….....16
      1.2.2. Fizik Tabanlı Yapay Zeka Optimizasyon Algoritmaları.……...19
         1.2.2.1. Arşimet Optimizasyon Algoritması…….19
         1.2.2.2. Atomik Yörünge Araması Metodu……..……21
         1.2.2.3. Kristal Yapı Algoritması…….……..24
         1.2.2.4. Lichtenberg Algoritması…….…….....27
   1.3. Anten Dizileri……..…….…….....29
   1.4. Tez Motivasyonu…….…….....31
   1.5. Tez Düzenlemesi…….……...…...31
   1.6. Literatür Özeti…….……..31

İKİNCİ BÖLÜM
GENEL BİLGİLER

2. Anten Teorisi……..…35
2.1. Problem Formülasyonu……..……35
2.1.1. Doğrusal Anten Dizisi……..……35
2.1.2. Dairesel Anten Dizisi……..…….37
2.1.3. Eşmerkezli Dairesel Anten Dizisi…….…...38
2.2. Yapay Zeka Optimizasyon Algoritmalarının Anten Dizi Sentezine Uygulanması…….…….…40

ÜÇÜNCÜ BÖLÜM
SONUÇLAR

3. Yapılan Optimizasyon Çalışmaları ve Elde Edilen Simülasyon Sonuçları ……...42
3.1. 3.1. LAA Sentezine PSO, ABC, HBA ve CSA Metotlarının Uygulanması…….….42
3.2. LAA Sentezine AOS Metodunun Uygulanması …….....49
3.3. LAA Sentezine AOA, CryStAl ve LA Metotlarının Uygulanması…..……...51
3.3.1. 10 Elemanlı LAA Sentezine AOA, CryStAl ve LA Metotlarının Uygulanması…….…….....51
3.3.2. 16 Elemanlı LAA Sentezine AOA, CryStAl ve LA Metotlarının Uygulanması…….……...53
3.3.3. 24 Elemanlı LAA Sentezine AOA, CryStAl ve LA Metotlarının Uygulanması…….……...55
3.4. CAA Sentezine AOA, AOS, CryStAl ve LA Metotlarının Uygulanması……..58
3.4.1. 8 Elemanlı CAA Sentezine AOA, AOS, CryStAl ve LA Metotlarının Uygulanması……....58
3.4.2. 10 Elemanlı CAA Sentezine AOA, AOS, CryStAl ve LA Metotlarının Uygulanması……....61
3.4.3. 12Elemanlı CAA Sentezine AOA, AOS, CryStAl ve LA Metotlarının Uygulanması…….……...64
3.5. CCAA Sentezine AOS Metodunun Uygulanması……..67

BÖLÜM 4
SONUÇLAR VE TARTIŞMALAR

4. Sonuçlar……..……..……..….77

TEZ ÇALIŞMASINDAN ÜRETİLEN ESERLER...…….....79
KAYNAKÇA…….....80
ÖZGEÇMİŞ……...…….……...90

KISALTMALAR LİSTESİ

~2x: Yaklaşık 2 Kat

~3x: Yaklaşık 3 Kat

ABC: Yapay Arı Kolonisi (Artificial Bee Colony)

AOA: Arşimet Optimizasyon Algoritması

AOS: Atomik Yörünge Araması (Atomic Orbital Search)

BBO: Biyocoğrafya Tabanlı Optimizasyon Metodu (Biogeography-based optimization)

CAA: Dairesel Anten Dizisi (Circular Antenna Array)

CCAA: Eşmerkezli Dairesel Anten Dizisi (Concentric Circular Antenna Array)

CEC: Evrimsel Hesaplama Kongresi (Congress on Evolutionary Computation)

CryStAl: Kristal Yapı Algoritması (Crystal Structure Algorithm)

CSA: Bukalemun Sürüsü Algoritması (Chameleon Swarm Algorithm)

CSO: Kedi Sürüsü Optimizasyon Metodu (Cat Swarm Optimization)

DA: Ejderha Sineği (Dragonfly Algorithm)

DF: Dizi Faktörü

DLA: Difüzyon Sınırlı Toplama (Diffusion Limited Aggregation)

EO: Yerçekimi Optimize edici (Equilibrium Optimizer)

EP: Evrimsel Programlama

FA: Ateş böceği Algoritması (Firefly Algorithm)

GB: Giga Byte

GHz: Giga Hertz

HBA: Bal Porsuğu Algoritması (Honey Badger Algorithm)

HPBW: Yarım Dalga Güç Genişliği (Half Power Beam Width)

IoT: Nesnelerin İnterneti (Internet of Things)

IIoT: Endüstriyel Nesnelerin İnterneti (Industrial Internet of Things)

LA: Lichtenberg Algoritması

LAA: Doğrusal Anten Dizisi (Linear Antenna Array)

LF: Lichtenberg Şekli (Lichtenberg Figure)

MATLAB: Matrix Laboratory Uygulaması

MF: Maliyet Fonksiyonu

MFO: Güve-Alev Optimizasyon Algoritması (Moth-Fly Optimizer)

Ort: Ortalama

PDF: Olasılık Yoğunluk Fonksiyonu (Probability Density Function)

PO: Politika Optimizasyonu

PSO: Parçacık Sürüsü Optimizasyonu

Ref: Referans

SLL: Yan Demet Seviyesi (Sidelobe Level)

Std: Standart Sapma

TO: Taguchi Optimizasyon Metodu (Taguchi Optimization)

TABLO LİSTESİ

Tablo 1.1. PSO algoritmasında kullanılan dahili parametreler, karşılaştırılan metotlar ve test fonksiyonları……....8

Tablo 1.2. ABC algoritmasında kullanılan dahili parametreler, karşılaştırılan metotlar ve test fonksiyonları……..….11

Tablo 1.3. HBA algoritmasında kullanılan dahili parametreler, karşılaştırılan metotlar ve test fonksiyonları…….14

Tablo 1.4. CSA algoritmasında kullanılan dahili parametreler, karşılaştırılan metotlar ve test fonksiyonları…….17

Tablo 1.5. AOA algoritmasında kullanılan dahili parametreler, karşılaştırılan metotlar ve test fonksiyonları…….20

Tablo 1.6. AOA algoritmasında kullanılan dahili parametreler, karşılaştırılan metotlar ve test fonksiyonları…….24

Tablo 1.7. CryStAl algoritmasında kullanılan dahili parametreler, karşılaştırılan metotlar ve test fonksiyonları……26

Tablo 1.8. LA algoritmasında kullanılan dahili parametreler, karşılaştırılan metotlar ve test fonksiyonları…….28

Tablo 3.1. PSO, ABC, HBA ve CSA metotları ile elde edilen SLL ve CPU zamanının istatistiksel değerleri……43

Tablo 3.2. PSO, ABC, HBA ve CSA metotları ile elde edilen LAA elemanlarının konumları……44

Tablo 3.3. AOS ile elde edilen SLL ve genlik değerleri……49

Tablo 3.4. 10 elemanlı LAA için karşılaştırmalı sonuçlar…….51

Tablo 3.5. 10 elemanlı LAA'ya ait genlik değerleri……..52

Tablo 3.6. 16 elemanlı LAA için karşılaştırmalı sonuçlar…….53

Tablo 3.7. 16 elemanlı LAA'ya ait genlik değerleri……..54

Tablo 3.8. 24 elemanlı LAA için karşılaştırmalı sonuçlar…….55

Tablo 3.9. 24 elemanlı LAA'ya ait genlik değerleri……..….56

Tablo 3.10. AOA, AOS ve CryStAl metotlarının kontrol parametreleri……..…58

Tablo 3.11. $SLL_{ort\&Std}$, $HPBW_{ort\&Std}$, ve CPU $Zaman_{ort\&Std}$ 8 elemanlı CAA için 30 çalışmada elde edilen değerler…….59

Tablo 3.12. 8 elemanlı CAA için ortalama SLL'nin konumları ve genlikleri değeri…...60

Tablo 3.13. $SLL_{ort\&Std}$, $HPBW_{ort\&Std}$, ve CPU $Zaman_{ort\&Std}$ 10 elemanlı CAA için 30 çalışmada elde edilen değerler…….62

Tablo 3.14. 10 elemanlı CAA için Ortanca SLL'nin konumlarının ve genliklerinin değerleri……...63

Tablo 3.15. $SLL_{ort\&Std}$, $HPBW_{ort\&Std}$, ve CPU $Zaman_{ort\&Std}$ 12 elemanlı CAA için 30 çalışmada elde edilen değerler…….65

Tablo 3.16. 12 elemanlı CAA için Ortanca SLL'nin konumlarının ve genliklerinin değerleri……...66

Tablo 3.17. Merkez elemanı olmayan n_1=4, n_2=6, n_3=8 elemanlı CCAA'ya ait elde edilen optimal genlik ve SLL değerleri……...……..68

Tablo 3.18. Merkez elemanlı n_1=4, n_2=6, n_3=8 elemanlı CCAA'ya ait elde edilen optimal genlik ve SLL değerleri……..70

Tablo 3.19. Merkez elemanı olmayan n_1=8, n_2=10, n_3=12 Elemanlı CCAA'ya ait elde edilen optimal genlik ve SLL değerleri…….72

Tablo 3.20. Merkez elemanlı n_1=8, n_2=10, n_3=12 elemanlı CCAA'ya ait elde edilen optimal genlik ve SLL değerleri……73

ŞEKİL LİSTESİ

Şekil 1.1. Telgraf kabloları…….3

Şekil 1.2. Anten dizilerine örnek……4

Şekil 1.3. Göç eden kuş sürüsü……..8

Şekil 1.4. PSO metodunun akış diyagramı……9

Şekil 1.5. Yiyecek arayan arılar…….……..10

Şekil 1.6. ABC algoritmasının akış diyagramı……12

Şekil 1.7. Bal porsuğu……..13

Şekil 1.8. HBA algoritmasının akış diyagramı……15

Şekil 1.9. Avlanan bukalemun…….……16

Şekil 1.10. CSA'nın akış diyagramı……18

Şekil 1.11. AOA'nın işleyişi……20

Şekil 1.12. AOA algoritmasının akış diyagramı……..21

Şekil 1.13. Bir atomun elektron yoğunluğu konfigürasyonu…….……..22

Şekil 1.14. Atomlarda çekirdeğin etrafındaki elektronlar için kuantum merdiven analojisi…….……...23

Şekil 1.15. AOS yönteminin akış diyagramı…….……..24

Şekil 1.16. a) Bravis modeli kristal yapısı, b) Kristal oluşumu gösterimi……...……..25

Şekil 1.17. CryStAl algoritmasının akış diyagramı…….……26

Şekil 1.18. Lichtenberg yapısı…….……27

Şekil 1.19. LA algoritmasının akış diyagramı…….……28

Şekil 1.20. Anten dizisi gösterimi…….……..29

Şekil 1.21. Anten dizisinin ışıma diyagramı……30

Şekil 2.1. 2N elemana sahip doğrusal anten dizisinin yapısı……36

Şekil 2.2. CAA'nın geometrik yapısı……...37

Şekil 2.3. Eşmerkezli dairesel anten dizisinin yapısı……..39

Şekil 3.1. a) 10 elemanlı, b) 16 elemanlı, c) 24 elemanlı, d) 32 elemanlı LAA için PSO, ABC, HBA ve CSA ile elde edilen ışıma diyagramları……45

Şekil 3.2. a) 10 elemanlı, b) 16 elemanlı, c) 24 elemanlı, d) 32 elemanlı LAA için PSO, ABC, HBA ve CSA ile elde edilen yakınsama eğrisi……..……47

Şekil 3.3. AOS ile elde edilen 10, 16 ve 24 elemana sahip LAA'ların ışıma diyagramı gösterilmektedir……..…….50

Şekil 3.4. 10 elemanlı LAA'ya ait ışıma diyagramı……..……52

Şekil 3.5. 10 elemanlı LAA'ya ait yakınsama eğrisi……..……..…53

Şekil 3.6. 16 elemanlı LAA'ya ait ışıma diyagramı……..54

Şekil 3.7. 16 elemanlı LAA'ya ait yakınsama eğrisi…….55

Şekil 3.8. 24 elemanlı LAA'ya ait ışıma diyagramı…….56

Şekil 3.9. 24 elemanlı LAA'ya ait yakınsama eğrisi……57

Şekil 3.10. 8 elemanlı CAA'ya ait ışıma diyagramları……59

Şekil 3.11. 8 elemanlı CAA'ya ait yakınsama eğrisi…….60

Şekil 3.12. 8 elemanlı CAA'ya ait kutu grafiği..……..……61

Şekil 3.13. 10 elemanlı CAA'ya ait ışıma diyagramı…….……..62

Şekil 3.14. 10 elemanlı CAA'ya ait yakınsama eğrisi……...63

Şekil 3.15. 10 elemanlı CAA'ya ait kutu grafiği……..64

Şekil 3.16. 12 elemanlı CAA'ya ait ışıma diyagramları……65

Şekil 3.17. 12 elemanlı CAA'ya ait yakınsama eğrisi……..66

Şekil 3.18. 12 elemanlı CAA'ya ait kutu grafiği…….……67

Şekil 3.19. Merkez elemanı olmayan n_1=4, n_2=6, n_3=8 elemanlı CCAA'ya ait ışıma diyagramı…….69

Şekil 3.20. Merkez elemanlı n_1=4, n_2=6, n_3=8 elemanlı CCAA'ya ait ışıma diyagramı…71

Şekil 3.21. Merkez elemanı olmayan n_1=8, n_2=10, n_3=12 elemanlı CCAA'ya ait ışıma diyagramı…….73

Şekil 3.22. Merkez elemanlı n_1=8, n_2=10, n_3=12 elemanlı CCAA'ya ait ışıma diyagramı…….75

Şekil 3.23. Üç halkalı CCAA'ya ait yakınsama eğrisi……..76

GİRİŞ

Gelişen iletişim teknolojisi ile nesnelerin birbirleriyle iletişim kurma ihtiyacı her geçen gün artmaktadır. Bu durum iletişim teknolojisinin en önemli unsurlarından biri olan anten tasarımına olan ilgiyi artırmaktadır. Günümüzde tek bir anten yeterli kazanç ve yüksek yönlülük sağlayamamaktadır. Bu nedenle, belirli bir geometriye sahip ve iki veya daha fazla antenden oluşan farklı anten dizilerine ihtiyaç duyulmaktadır. Hesaplamalı elektromanyetik alanında bulunan anten dizi problemi zor tasarım problemlerinden birisidir. Klasik optimizasyon metotları ise zorlu anten dizi problemlerinin üstesinden gelmekte zorlanmaktadır. Bu yüzden anten dizilerinin sentezinde Yapay Zeka (YZ) optimizasyon algoritmalarının kullanılması literatürde oldukça yaygındır. YZ optimizasyon algoritmaları sürü zekasına dayalı, sosyal davranıştan esinlenen, fizik tabanlı ve evrimsel algoritmalar gibi esin kaynaklarına göre sınıflandırılmaktadır.

Bu tezde, anten dizilerinin genlik, konum ve elemanlar arası mesafe değerleri, literatürde oldukça iyi bilinen Yapay Arı Koloni Algoritması (Artificial Bee Colony-ABC) ve , Parçacık Sürü Optimizasyon Metodu (Particle Swarm Optimization-PSO) ile literatüre yeni kazandırılmış YZ optimizasyon algoritmalarından Arşimet Optimizasyon Algoritması (Archimedes Optimization Algorithm-AOA), Atomik Yörünge Arama Metodu (Atomic Orbital Search-AOS), Kristal Yapı Algoritması (Crystal Structure Algorithm-CryStAl), Kertenkele Sürü Algoritması (Chameleon Swarm Algorithm-CSA), Bal Porsuğu Algoritması (Honey Badger Algorithm-HBA), Lichtenberg Algoritması (Lichtenberg Algorithm-LA) kullanılarak optimum olarak belirlenmiştir. Tezde, anten dizilerini optimize edebilme yeteneğine sahip hızlı ve esnek bir arama algoritmasını tespit etmek için farklı metotlar kullanılarak doğrusal, dairesel ve eşmerkezli dairesel anten dizi problemleri üzerinde çalışmalar yapılmıştır. Yapılan çalışmalar sonucunda önerilen metotların sonuçları literatürdeki metotların sonuçlarıyla karşılaştırılmış ve önerilen metotların farklı anten dizi sentez problemlerinde oldukça başarılı olduğu tespit edilmiştir.

Birinci bölümde antenlerin tarihi gelişiminden başlanılmış, antenlere giriş yapılmış, anten dizileri anlatılmış ve bu tezde kullanılan YZ optimizasyon algoritmaları açıklanmıştır. İkinci bölümde doğrusal anten dizilerinin, dairesel anten dizilerinin ve eş merkezli dairesel

anten dizilerinin problem formülasyonları ile birlikte bu anten dizilerini optimize etmek için gerekli olan maliyet fonksiyonu verilmiştir. Üçüncü bölümde ise bahsi geçen literatüre yeni kazandırılmış algoritmalar ve literatürde oldukça iyi bilinen metotlar kullanılarak elde edilen anten dizilerinin simülasyonlarının sonuçları karşılaştırmalı olarak verilmiştir. Dördüncü bölümde elde edilen sonuçlar analiz edilmiştir.

BİRİNCİ BÖLÜM
ANTEN DİZİSİ VE YAPAY ZEKA OPTİMİZASYON ALGORİTMALARI

1. Antenler

İnsanların iletişim kurma ihtiyacı yüzyıllar önce başlamıştır. İnsanlar, jestlerle, mimiklerle ve ses tonunu ayarlayıp birbirlerine seslenerek iletişim kurmaya çalışmışlardır. Bu yöntemler belirli mesafelerle sınırlı olduğu için daha yeni tekniklere ihtiyaç duyulmuş ve bu ihtiyaç duman ile haberleşmenin gelişimine yol açmıştır. Duman ile çok daha uzak mesafelere kısa bilgilerin iletimi sağlanabilmektedir (Güneş, 2013). Fakat bu iletişim tekniğinde bilgi kalıcı olmamaktadır. Bilginin kalıcı olabilmesi ise yazının geliştirilerek elde edilen bilgilerin tablet vb. araçlara yazılmasıyla sağlanabilmiştir. Yazı ilk başta bilginin, çok daha uzun mesafelere silinmeden gönderilmesini mümkün kıldı. Daha sonra elektriğin keşfedilmesi sonucu ortaya çıkan telgraf sayesinde daha hızlı iletişim sağlanmaya başlandı (Güneş, 2013). Şekil 1.1'de telgraf kabloları gösterilmektedir:



Şekil 1.1. *Telgraf kabloları*
(https://st2.depositphotos.com/1162190/5477/i/950/depositphotos_54779581-stock-photo-old-wired-communications.jpg?download=true Erişim Tarihi:26.05.2022)

Telgrafın kullanımı uzak mesafelere en hızlı iletişimi sağlamasına rağmen çok ciddi kablo masraflarına neden olmaktaydı. Telgraf bu maliyetine rağmen sadece yazılı metinleri iletebiliyordu. Telefonların keşfi ile sesli iletişim de başlamış oldu. İnternetin keşfi ise sadece ses değil, görüntü ve yüksek boyutta veri aktarımını da mümkün kıldı. Nesnelerin İnterneti (IoT)'nin geliştirilmesiyle beraber yüksek bant genişliği ihtiyacı ortaya çıkmıştır (Painuly ve ark., 2020). Bu ihtiyacı karşılamak için antenler tek başlarına yetersiz kalmaktadır. Bu yüzden yüksek bant genişliği gerektiren durumlarda birçok antenden oluşan anten dizileri kullanılmaktadır (Balanis, 2015). Anten dizileri, anten elemanlarının dizildiği yapının şekline göre adlandırılmaktadır. Dairesel, doğrusal ve eliptik geometrilere sahip anten dizileri örnek olarak verilebilir. Şekil 1.2'de anten dizilerine örnek gösterilmiştir:



Şekil 1.2. *Anten dizilerine örnek*
(https://cdn.pixabay.com/photo/2013/03/01/18/49/antenna-88022_960_720.jpg Erişim Tarihi:26.05.2022)

1.1. Tezin Amacı

IoT ve 5G gibi daha yeni teknolojilerin ortaya çıkmasıyla daha verimli anten teknolojilerine ihtiyaç duyulmuştur. Birden çok antenden oluşan anten dizileri ile yüksek verimliliğe ulaşılabilmektedir. Anten dizilerinin sentezlenmesi için yer, genlik, elemanlar

arası mesafe gibi birçok parametrenin optimize edilmesi gerekmektedir (Durmus ve Kurban, 2021). Birçok parametrenin optimize edilmesi gerekliliği ise bu problemi oldukça zorlaştırmaktadır. Bu tezin amacı bu zorlu optimizasyon problemini en verimli şekilde çözerek yeni anten dizileri tasarlamak ve anten dizi sentezleme yeteneğine sahip olan yeni Yapay Zeka (YZ) optimizasyon algoritmalarını anten dizisine uygulayarak, anten dizisi için literatüre yeni metotlar kazandırmak ve en verimli anten dizilerini tasarlamaktır.

Anten dizisi gibi zorlu problemlerin iyi bir çözücüye ihtiyacı vardır ve dünya var olan en büyük problem çözücüdür. Doğa ve doğadaki yaşam en basit olaylardan en karmaşık olaylara kadar birçok konuda problemlerle karşılaşmıştır ve bu problemlere en uygun çözümü bulmuştur. Problemi kimi zaman yavaş yavaş çözerken kimi zaman çok hızlı çözmektedir. Tabiattaki canlılar sürü davranışlarını kullanarak probleme çözüm ararken, doğa belli bir tabiat kanununu kullanarak bu problemlere çözüm bulmaktadır. Kimisi yavaş bir şekilde çözüm arayıp en iyi sonuca ulaşırken kimisi çok daha hızlı yakınsama yaparak aynı sonuca daha hızlı ulaşabilmektedir. Bazıları ise hızlı yakınsama yaparken en iyi sonuca ulaşamamaktadır. Problemlere çözüm ararken her canlının farklı taktiği, her fizik olayının farklı doğa kanunu vardır. Bu kanunlardan esinlenilerek geliştirilen metasezgisel yapay zeka metotları yardımıyla birçok optimizasyon problemlerine çözümler bulunmuştur (Babayigit, 2018; Babayigit ve Senyigit, 2017; Balaban ve Somov, 2015; Basbug, 2017; Bouchekara, 2021; Khalid ve ark., 2021; L. Liu ve ark., 2021; Ojstersek ve ark., 2020; Tailor ve ark., 2020; Talatahari ve Azizi, 2020; Z.-J. Wang ve ark., 2020; Z. Wang ve ark., 2020; Wu ve ark., 2018).

1.2. Yapay Zeka Optimizasyon Algoritmaları

Yapay Zeka Optimizasyon Algoritmaları bilgiye sahip olanlar ve olmayanlar olarak ikiye ayrılmaktadır (Mirjalili ve Dong, 2020). Bilgiye sahip yapay zeka optimizasyon algoritmaları metasezgisel yöntemler olarak adlandırılmaktadır (Mirjalili ve Dong, 2020). Durmaksızın her arama ajanı bilgi gönderir ve bir bilgiye dayalı olarak arama yaparlar. Bu sayede arama uzayındaki tuzakları fark edebilirler. Yerel optimum noktaları fark ederek onlardan kaçınırlar ve küresel optimum noktalara odaklanırlar. Ayrıca metasezgisel

yöntemler birden fazla arama ajanına sahiptir (Mirjalili ve Dong, 2020). Arama ajanları birbirleri arasında iletişim kurarlar ve bu iletişime göre esinlendikleri kaynağın arama stratejisine göre çözüm uzayında arama yaparlar. Bilgiye sahip olmayan yöntemler olarak bilinen klasik yöntemler tek ajana sahiptirler. Bu yüzden bilgiye sahip değillerdir ve yerel optimumlar ile küresel optimumları ayırt etme yeteneklerine de sahip değildir (Mirjalili ve Dong, 2020). Metasezgisel metotlar esin kaynaklarına göre sınıflandırılmaktadır. Fizik kanunlarından esinlenen fizik tabanlı metotlar, sürü davranışlarından esinlenen sürü zekâsı, canlıların sosyal hayattaki davranışlardan esinlenen sosyal davranış algoritmaları gibi birçok alandan esinlenilen metasezgisel yöntemler bulunmaktadır. Fizik tabanlı algoritmalara, suyun kaldırma kuvvetinden esinlenen Arşimet Optimizasyon Algoritması (AOA) (Hashim ve ark., 2021), elektronun hareketini anlatan kuantum kanunundan esinlenen Atomik Yörünge Araması Metodu (AOS) (Azizi, 2021), kristalin oluşumundan esinlenen Kristal Yapısı Algoritması (CryStAl) (Talatahari ve ark., 2021), gökyüzünde yıldırımların oluşturduğu Lichtenberg şeklinden esinlenen Lichtenberg Algoritması (LA) (Pereira ve ark., 2021); sürü zekasına sahip algoritmalara, tabiattaki kuş ve balık sürülerinin göç davranışından esinlenen Parçacık Sürü Optimizasyon Metodu (PSO) (Poli ve ark., 2007), arıların yiyecek aramasını temel alan Yapay Arı Kolonisi Algoritması (ABC) (Karaboga ve Basturk, 2008), bal porsuklarının ilginç avlanma taktiğinden geliştirilen Bal Porsuğu Algoritması (HBA) (Hashim ve ark., 2022), kertenkelelerin avlanma becerilerini gören araştırmacılar tarafından esinlenen Kertenkele Sürüsü Algoritması (CSA) (Braik, 2021) metotları; sosyal davranışlardan esinlenen metotlara, hayvan gruplarının işbirlikçi yaklaşımlarından esinlenen Dinamik Sosyal Davranış Metodu (Sankar ve ark., 2017), sosyal ağdaki kullanıcıların ilgileri ve etkileşimini inceleyen Sankar tarafından geliştirilen Paralel Sosyal Davranış Tabanlı Algoritması (Sankar ve ark., 2017), toplum içi ve toplumlar arası etkileşimi temel alan Toplum ve Medeniyet Algoritması (Ray ve Liew, 2003) metotları örnek olarak verilebilir.

1.2.1. Sürü Zekasına Dayalı Yapay Zeka Optimizasyon Algoritmaları

Sürü zekasına dayalı metasezgisel metotlar, doğadaki canlıların yiyecek arama veya avlanma davranışlarından esinlenilerek geliştirilen algoritmalardır. Bu tür algoritmalar literatürde oldukça popüler olarak kullanılmakta ve durmaksızın yenileri ortaya çıkmaktadır

(Hashim ve Hussien, 2022; Oyelade ve ark., 2022; N. Singh ve ark., 2022; Trojovský ve Dehghani, 2022; Zeidabadi ve ark., 2022). Sürü zekâsına dayalı YZ optimizasyon algoritmalarından bazıları bu çalışmada kullanılmıştır. Bunlar:

- Parçacık Sürüsü Optimizasyon Metodu
- Yapay Arı Kolonisi Algoritması
- Bal Porsuğu Algoritması
- Bukalemun Sürüsü Algoritması

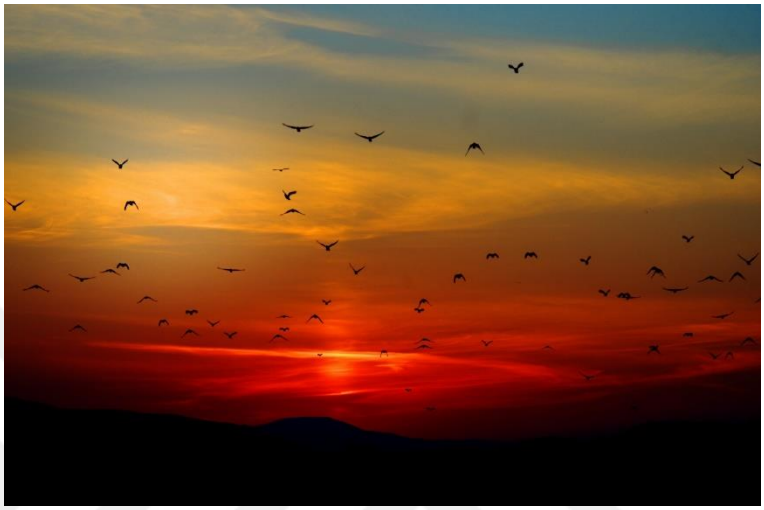
1.2.1.1. Parçacık Sürü Optimizasyon Metodu

Parçacık Sürü Optimizasyonu (Poli ve ark., 2007), literatürdeki YZ optimizasyon algoritmaları arasındaki en iyi bilinen algoritmalardan biridir. PSO algoritması, göç sırasında doğadaki kuş ve balık sürülerinin yiyecek arama davranışlarından esinlenmiştir ve bu sürülerin davranışı matematiksel olarak modellenmiştir. Bu sürülerin matematiksel modellemesi konum, uygunluk değeri ve hız olmak üzere üç özellik dikkate alınarak yapılmıştır. İlk parametre olan konum, bu sürüdeki bir üyenin arama uzayındaki yerini belirtir. İkinci parametre olan uygunluk değeri, sürüdeki kanatlıların yiyeceğe yakınlıklarına göre ne kadar uygun olduğunu gösteren değerdir. Bir kuşun uzayda sürü içindeki konumunu ve uzayda sürüdeki en iyi kuşun konumunu hesaba katarak üçüncü parametre olan hızını günceller. Bu bilgiler eşliğinde sürü üyeleri, konumlarını sürüleriyle paylaşır ve sürünün en iyi konumlara doğru ilerlemesini sağlar. Sürü ise önceki konumundan daha iyi olduğu bir konum arayarak hareket eder. PSO'nun avantajları ise yerel bir optimuma takılmaması ve sonuca daha hızlı ulaşmak için türev hesaplamamasıdır. Sürü üyeleri her zaman birbirleriyle bilgi paylaştığından, diğer üyeler bu bilgileri en iyi pozisyonlara doğru daha iyi gezinmek için kullanabilir. Aşağıda PSO'nun konum ve hız değerleri aşağıdaki denklem yardımı ile belirlenir.

$$\begin{cases} v_i \leftarrow v_i + U(0,\varphi_1)\otimes(p_i - x_i) + U(0,\varphi_2) \otimes (p_g - x_i) \\ x_i \leftarrow x_i + v_i \end{cases} \quad (1.1)$$

$U(0,\varphi_1)$, her yinelemede her parçacık için ayrı şekilde oluşturulan rasgele sayıları temsil eder. $\otimes$ çarpımı ifade eden bir simgedir. Konum x_i ile gösterilir. v_i, hızı temsil eder ve

değeri $[+V_{max}, -V_{max}]$ aralığında belirlenir. Şekil 1.3'te göç eden kuş sürüsüne örnek verilmiştir.



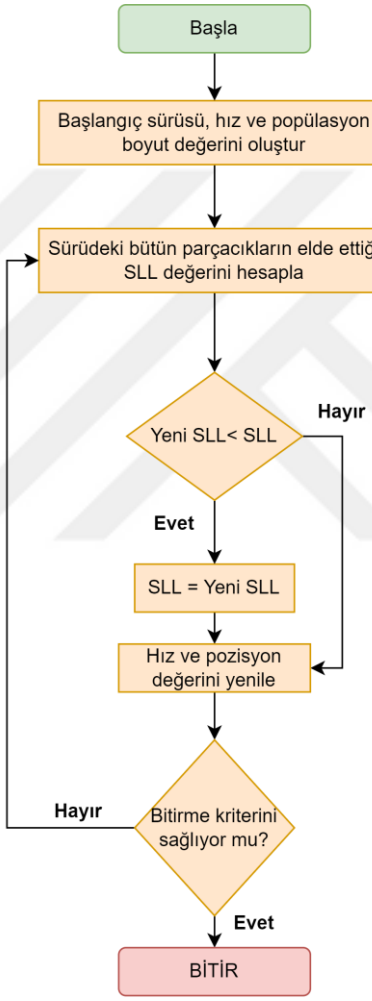
Şekil 1.3. *Göç eden kuş sürüsü (*https://www.pexels.com/tr-tr/fotograf/gun-batimi-sirasinda-dagin-uzerinde-ucan-kus-surusu-70577/ Erişim Tarihi:26.05.2022)

PSO algoritmasının problemi optimize etmek için kullandığı dahili parametreler, performansını göstermek için karşılaştırıldığı test fonksiyonları ve karşılaştırma yapıldığı diğer metotlar Tablo 1.1'de gösterilmiştir (Poli ve ark., 2007).

Tablo 1.1. *PSO algoritmasında kullanılan dahili parametreler, karşılaştırılan metotlar ve test fonksiyonları*

Algoritmalar	**Parametreler ve Değerleri**	**Karşılaştırılan metotlar**	**Uygulanan test fonksiyonları**
PSO	·α=0.1 ·β=0.005	·GOA ·SSA ·GWO ·WOA ·PSO ·SCA ·DA ·CMA-ES ·ALO ·EO ·MPA	·10 kıyaslama Fonksiyonu (CEC-2019 kıyaslama fonksiyonu) ·Welded tasarım problemi ·30 kıyaslama fonksiyonu (CEC-2017 kıyaslama fonksiyonu) ·7 gerçek tasarım problemi(basınçlı kap tasarımı, kaynaklı kiriş tasarımı, çekme/basınç yayı tasarımı, 3 bartruss tasarım problemi, Hız düşürücü problemi, Konsol kiriş tasarımı problemi ve Çok diskli debriyaj freni problemi)

PSO algoritmasının problemi optimize ederken izlediği bir yol vardır. Bu yolu akış diyagramı ile anlatabiliriz. PSO hakkında daha detaylı bilgi için ana makaleyi inceleyebilirsiniz (Poli ve ark., 2007). Şekil 1.4'te PSO metodunun anten dizi sentezinde nasıl uygulandığına dair akış diyagramı verilmiştir:



Şekil 1.4. *PSO metodunun akış diyagramı*

1.2.1.2. Yapay Arı Kolonisi Algoritması

Yapay Arı Koloni (Karaboga ve Basturk, 2008) optimizasyon yöntemi 2009 yılında Karaboğa tarafından önerilmiştir. Bu yöntem literatürde en çok bilinen sürü tabanlı algoritmalardan biridir. ABC yöntemi, arama uzayına bazı sınırlamalar getirerek arıların yiyecek arama yöntemlerine farklı bir bakış açısı getirmiştir. Bu sınırlamalar ABC'deki üye sayısı ve arı sayısı kadar tutulur.

Temelde görevi olan ve kesin bir görevi olmayan arılar olmak üzere iki türü olmakla birlikte üç çeşit arı vardır. Birinci tip, kaynağa gidip yiyecek getiren işçi arılardır. Şekil 1.5'te yiyecek arayan işçi arılar gösterilmektedir.



Şekil 1.5. *Yiyecek arayan arılar (* https://www.pexels.com/tr-tr/fotograf/pembe-beyaz-yaprakli-cicek-46534/ Erişim Tarihi:26.05.2022)

İkinci tip olan gözcü arılar, onları yönlendirmek ve kaynakların daha hızlı toplanmasına yardımcı olmak için hangi kaynağa gideceklerini belirler. Üçüncü tür ise kendi kaynaklarını yetersiz gören kâşif arılar, yeni kaynaklar bulmak için herhangi bir gözcü veya refakatçi arıdan etkilenmeden keşfe çıkan arılardır. Ayrıca işçi arılar yiyecek topladıktan sonra kovana geldiklerinde kovandaki arılarla besin kaynağının yeri ve kalitesi hakkında bilgi paylaşırlar. Diğer arılar da bu bilginin varlığında hareket eder. Arılar balın kalitesini kaynağında değerlendirerek en iyi kaynağa doğru yol alır ve bu süreç sayesinde en iyi sonuca hızlı bir şekilde yaklaşırlar. Karaboğa tarafından yapılan bu sınırlamalar, besin kaynaklarının sayısını yapay arı kolonisinin üyelerine eşit tutarak benzersiz bir algoritma sunmuştur. İşçi arılar önceden keşfettiği besin kaynağının (x_{mi}) bulunduğu çevrede daha fazla nektar bulunan yeni besin kaynaklarını (u_{mi}) denklem 1.2 yardımıyla ararlar.

$$u_{mi} = x_{mi} + \emptyset_{mi}(x_{mi} - x_{ki}) \quad (1.2)$$

x_{ki}, rastgele olarak seçilmiş besin kaynaklarını temsil etmektedir. $\emptyset_{mi}$, rastgele katsayı değerlerini göstermektedirler. Tablo 1.2'de bahsedilen ABC algoritmasında kullanılan dahili parametreler, karşılaştırılan metotlar ve performansını test etmek için uygulanan test fonksiyonları gösterilmiştir (Karaboga ve Basturk, 2008):

Tablo 1.2. *ABC algoritmasında kullanılan dahili parametreler, karşılaştırılan metotlar ve test fonksiyonları*

Algoritmalar	**Parametreler ve Değerleri**	**Karşılaştırılan metotlar**	**Uygulanan test fonksiyonları**
ABC	·Pop=100 ·İşçi arı=50 ·Gözlemci arı (49) ·Rast. sc (1) ·St sınırı (50xBoyut)	·5x5 SOM-ES ·7x7 SOM-ES ·NG-ES(m=10) ·NG-ES(m=20) ·CMA-ES	·XOR sorunu için Sinir Ağı Eğitimi ·Kaynaklı Kiriş Tasarımı ·50 kalite test fonksiyonu

Tablo 1.2'den de anlaşıldığı üzere ABC algoritması birçok test fonksiyonuna uygulanarak performans testi yapılmıştır. Ayrıca başka metotlar ile karşılaştırılarak performansı analiz edilmiştir. Parametreleri incelendiğinde algoritma içinde birçok parametre barındırmaktadır. ABC algoritması hakkında daha detaylı bilgiye ulaşmak için ana makalesini inceleyebilirsiniz (Karaboga ve Basturk, 2008). Şekil 1.6'da ise ABC algoritmasının en iyi SLL'ye ulaşmak için izlediği yolu gösteren akış diyagramı verilmiştir:



Şekil 1.6. *ABC algoritmasının akış diyagramı*

1.2.1.3. Bal Porsuğu Algoritması

Bal Porsuğu Algoritması (Hashim ve ark., 2022), doğada bal porsuğu olarak bilinen canlının yiyecek arama ve avlanma davranışlarından esinlenilerek literatüre kazandırılmıştır. Bal porsuğu tabiatta korkusuzluğuyla bilinir ve bazı Afrika çöllerinde ve yağmur ormanlarında bulunur; Güneybatı Asya ve Hindistan'da bulunan bir canlı türüdür. Kendi arama stratejisi ile tehlikeli yılanlar dahil 67 farklı türü bulur ve avlar. Bal porsuğu Şekil 1.7'de gösterilmektedir:



Şekil 1.7. *Bal porsuğu*
(https://www.nationalgeographic.com/animals/mammals/facts/honey-badger Erişim Tarihi: 26.05.2022)

Bal, bal porsuğunun en sevdiği yiyeceklerden biridir. Bu hayvan balı bulmak için bir alet kullanabilir. Bal porsuğunun doğada iki çeşit yiyecek arama ve avlanma şekli vardır. İlki kazma aşaması olarak adlandırılır. Kazma aşaması, güçlü koku alma duyusu ile avını arayarak bulur. Kazarak ve dolaşarak avına yaklaşır. İkinci aşama bal aşaması olarak adlandırılır. Bu aşamada bal porsuğunun koku alma duyusu bal aramak için yeterli olmadığı için rehbere ihtiyaç duyar. Bir kuş tarafından yönlendirilen bal porsuğu, rehber kuş ile ortak bir arama ve avlanma stratejisine sahiptir. Bu stratejide bal porsuğu rehber kuşu takip eder. Rehber kuşu bu stratejide arama görevini üstlenir. Bal porsuğundan farklı bir arama stratejisi sunan rehber kuş sayesinde balı bulur. Ama kuş bala ulaşamaz. Burada bal porsuğunun alet kullanma yeteneği devreye girer ve bala ulaşır. Bu sayede bal porsuğu yetersiz kaldığı arama alanlarındaki eksikleri bir rehber kuş sayesinde tamamlar. Bal porsuğu algoritmasının bazı dâhili parametreleri, performansını test etmek için karşılaştırıldığı metotlar ve test fonksiyonları vardır. Bunlar Tablo 1.3'te gösterilmektedir (Hashim ve ark., 2022):

Tablo 1.3. *HBA algoritmasında kullanılan dahili parametreler, karşılaştırılan metotlar ve test fonksiyonları*

Algoritmalar	**Parametreler ve Değerleri**	**Karşılaştırılan metotlar**	**Uygulanan test fonksiyonları**
HBA	·Hb say. =50 ·β=6 ·C=2	·SA ·EHO ·WOA ·GOA ·PSO ·CMA-ES ·L-SHADE ·MFO ·TEO ·HHO	·CEC-2017 standart test fonksiyonu ·24 adet kalite test fonksiyonu ·Basınçlı kap tasarım problemi ·Kaynaklı kiriş tasarım problemi ·Gerginlik/basınç yayı tasarım sorunu ·Hız düşürücü tasarım sorunu

HBA birçok mühendislik problemine uygulanmış ve iyi sonuçlar elde edilmiştir. Bal porsuğunun benzersiz arama ve kullanma yeteneği şu şekilde hesaplanır:

$$Div_j = \frac{1}{N}\sum_{i=1}^{N} median\left(x^j\right) - x_i^j, \tag{1.3}$$

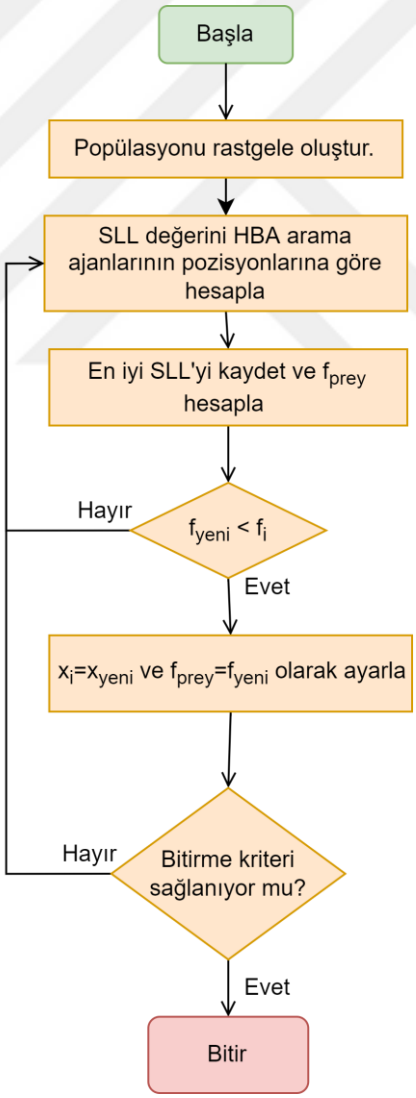
$$Div^t = \frac{1}{D}\sum_{j=1}^{D} Div_j,\ t = 1,2,\dots,t_{max} \tag{1.4}$$

i ve j sırasıyla bal porsuğunun sırasını ve boyutunu temsil eder. x_i^j, bal porsuğu çözüm adayını gösterir. Div_j, boyutlar için ortalama çeşitliliktir. Div^t, tüm iterasyonların sonunda alınan bu boyutsal çeşitliliğin (D) ortalamasıdır. Çeşitlilik hesaplandıktan sonra sömürü yüzdesini hesaplamak ve araştırmak için aşağıdaki denklem kullanılır (Hashim ve ark., 2022):

$$Exploration\% = \frac{Div^t}{\max(Div)} \times 100; \tag{1.5}$$

$$Exploitation\% = \frac{|Div^t - \max(Div)|}{\max(Div)} \times 100 \quad (1.6)$$

Maksimum iterasyonda $\max(Div)$, çeşitlilik t_{max} olarak ifade edilir. HBA algoritması hakkında daha detaylı bilgi edinmek için ana makalesini inceleyebilirsiniz (Hashim ve ark., 2022). HBA algoritmasının en iyi SLL'yi elde etmek için izlediği yolu gösteren akış diyagramı Şekil 1.8'de gösterilmektedir.



Şekil 1.8. *HBA algoritmasının akış diyagramı*

1.2.1.4 Bukalemun Sürü Algoritması

Ortama uyum sağlamak için renk değiştirme yetenekleriyle bilinen bukalemunlar iyi avcılardır. Bu özellik bukalemunları iyi birer kâşif yapar. CSA'nın yaratıcıları, bukalemunların bu keşif özelliği ile avlarını keşfetmek için özel bir keşif yeteneğine sahip olduğundan bahseder (Braik, 2021).

Bukalemunların bir diğer özelliği ise iki gözünün birbirinden bağımsız hareket edebilmesidir. Geniş açılı gözleriyle bir yandan hareket ederken avlarını takip edebilirler. Bukalemunlar avlanmak için dillerini çok hızlı kullanırlar. Bu yetenekleri sayesinde iyi bir avcı olduğu iddia edilebilir. Dili hızlı kullanabilme yeteneği, hızlı yakınsamaya benzetilebilir. Ayrıca gözlerinin 360 derecelik keşif kabiliyeti ile arama alanını iyi keşfedebilirler. Avlanan bir bukalemunun resmi Şekil 1.9'da gösterilmektedir:



Şekil 1.9. *Avlanan bukalemun* (https://www.pexels.com/tr-tr/fotograf/yesil-ve-sari-bukalemun-close-up-photography-70556/ Erişim Tarihi: 26.05.2022)

Bu algoritmada iki tür keşif ve dört tür avlanma yeteneğinden bahsedilmiştir. Keşif yetenekleri çöl keşfi ve ağaç keşfidir. İlk avlanma yetenekleri, gözlerini geri çevirmek ve avlarının dikkatini çekmeden hızla yakınlaşmaktır. İkinci yeteneği ise gözünü sağa çevirerek görsel olarak avlarını yönlendirmektir. Üçüncü yeteneği ise uzun dili sayesinde avını yaklaşmadan uzaktan avlayabilmesidir. Dördüncü yeteneği, avına yaklaşarak avlanmaktır. Bu algoritma keşif kabiliyeti ile tüm uzayı keşfederek avlanmaya katkı sağlar.

Tüm alana hâkim olan bukalemun, hızla avına yakınsar ve hızlı bir yakınsama eğrisine sahiptir. CSA CEC-2015 ve CEC-2017 test paketleri dahil olmak üzere toplam 67 test problemiyle karşılaştırılmıştır. Karşılaştırılan fonksiyonlar, dahili parametreler ve karşılaştırıldığı metotlar Tablo 1.4'te gösterilmektedir (Braik, 2021).

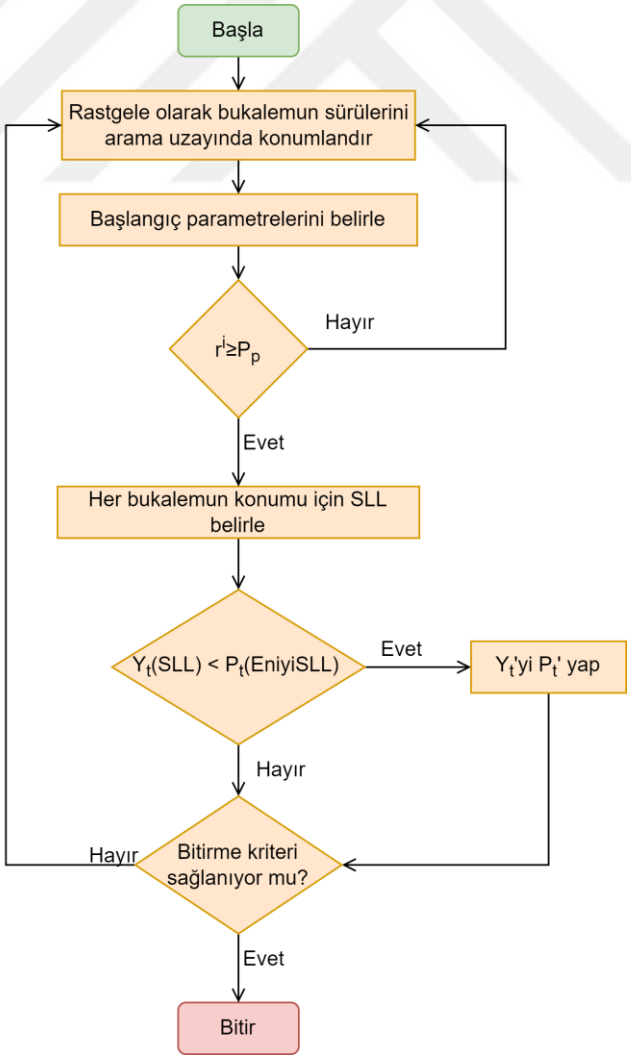
Tablo 1.4. *CSA algoritmasında kullanılan dahili parametreler, karşılaştırılan metotlar ve test fonksiyonları*

Algoritmalar	**Parametreler ve Değerleri**	**Karşılaştırılan metotlar**	**Uygulanan test fonksiyonları**
CSA	·p_1=0.25 ·p_2=1.50 ·p =1.0 ·c_1=1.75 ·c_2=1.75	·SSA ·PSO ·MFO ·EO ·MVO ·SCA ·GSA ·GA ·HS ·GWO	·Tek modüllü, çok modüllü, hibrit ve kompozisyon test fonksiyonları (CEC 2015) ·C-f1, C-f2, C-f3, C-f4-f6, C-f8, C-f9 C-f12 ve C-f14 kalite testi fonksiyonu kısıtlama yönetimi ·Kaynaklı kiriş tasarım problemi ·Basınçlı kap tasarım problemi ·Gerginlik/basınç yayı tasarım sorunu ·Hız düşürücü tasarım sorunu ·Rulmanlı rulman tasarım problemi

CSA'nın keşfi aşağıdaki gibi hesaplanabilir:

$$Y_{t+1}^{i,j} = \begin{cases} Y_t^{i,j} + p_1\left(P_t^{i,j} - G_t^j\right)r_2 + p_2\left(G_t^j - Y_t^{i,j}\right)r_1, & r_i \geq P_p \\ Y_t^{i,j} + \mu\left(\left(u^j - l^j\right)r_3 + l_b^j\right) sgn(rand - 0.5), & r_i < P_p \end{cases} \quad (1.7)$$

i bukalemun sırasını gösterir. *t*, iterasyon sayısını temsil ederken, *j*, t'nin problem boyutunu sembolize eder. $Y_{t+1}^{i,j}$ ve $Y_t^{i,j}$ sırasıyla bir sonraki iterasyonda olacak konumu ve mevcut konumu ifade eder. $P_t^{(i,j)}$, t'deki en iyi konuma sahip bukalemundur. G_t^j, t'inci iterasyona kadar tüm iterasyonlardaki en iyi konumu belirtir. p_1 ve p_2, keşif yeteneğini kontrol etmek için var olan sayılardır. 0 ile 1 aralığında seçilen r_1, r_2 ve r_3 rastgele sayılardır. P_p ise bukalemun avlanma olasılığını temsil eder. $sgn(rand - 0.5)$, arama ve bulma yeteneğini kontrol eden [-1,1] aralığında değişen değerdir. μ iterasyonların bir fonksiyonu olarak tanımlanan parametredir. Daha fazla bilgiye ulaşmak için ana makaleye bakabilirsiniz (Braik, 2021). Şekil 1.10'da CSA'nın optimum tasarımını elde ederken ki akış diyagramı verilmiştir:



Şekil 1.10. *CSA'nın akış diyagramı*

1.2.2. Fizik Tabanlı Yapay Zeka Optimizasyon Algoritmaları

Fizik tabanlı yapay zeka algoritmaları, doğada var olan fizik kanunlarının matematiksel modellerinden esinlenilerek algoritma haline getirilmiş yapılardır. Fizik tabanlı yöntemler, ardında yatan kanunlara dayandırılarak sağlam temeller üzerine kurulmuş algoritmalardır. Durmadan literatüre yenileri katılmakta olup makaleler incelendiğinde performans testleri oldukça iyi sonuçlar vermektedir. Bu çalışmada fizik tabanlı algoritmaları öne çıkartılmasının en önemli sebebi en iyi sonuçların fizik tabanlı metotlar ile elde edilmiş olmasıdır.

Fizik tabanlı algoritmalara örnek verilecek olursa:

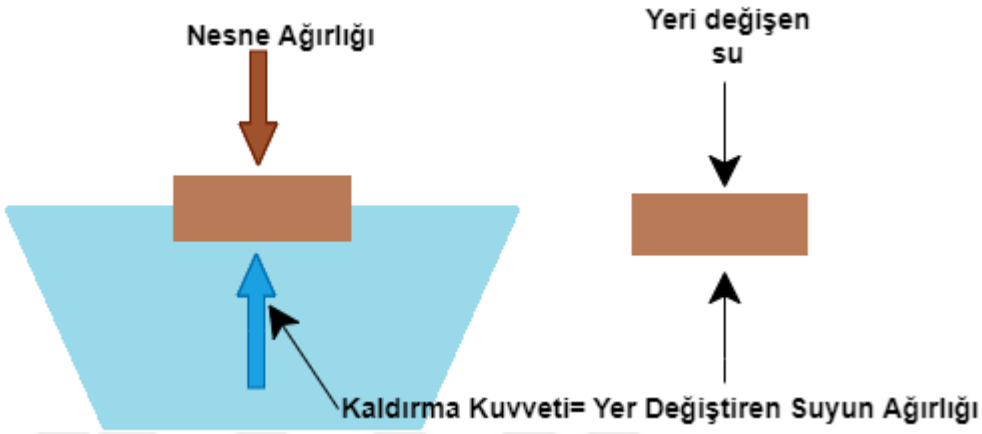
- Arşimet optimizasyon metodu (Hashim ve ark., 2021)
- Atomik yörünge araması (Azizi, 2021)
- Kristal yapı algoritması (Talatahari ve ark., 2021)
- Lichtenberg algoritması (Pereira ve ark., 2021)

Yukarıda bahsedilen algoritmalar 2021 yılında sonra literatüre yeni kazandırılmış fizik tabanlı YZ optimizasyon algoritmalarıdır.

1.2.2.1. Arşimet Optimizasyon Metodu

AOA sıvıların kaldırma kuvvetinden esinlenerek Hashim ve diğerleri tarafından keşfedilmiştir (Hashim ve ark., 2021). Bu algoritmada ilk olarak suya hayali bir cisim atılır. Düşen nesne yüzebilir veya suya batmış olabilir. AOA, cismi suyun üzerinde dengede tutmaya çalışarak optimum değeri bulmaya çalışır. Bu algoritmanın temel olarak iki öğesi vardır. Birincisi akışkan bir madde, ikincisi ise katı bir cisimdir. Katı cisim sıvının içine bırakıldığında, sıvının ağırlığından daha ağır değilse yüzer haldedir. Tersi durumda ise denge durumuna ulaşana kadar sıvının içinde hareket edecektir. Başka bir deyişle, cismin suda bir kuvveti vardır. Bu kuvveti sıfıra indirmek ve bu kuvveti 0'da sabit tutarak dengede tutmak gerekir. Bunu sağlamak için gerekli bazı parametreler vardır. AOA, bu nesneyi optimize etmek için üç parametre kullanır. Bunlar hacim, yoğunluk ve ivmedir. burada v hacimdir, p yoğunluktur, a ivmedir ve sırasıyla b ve 0 alt simgeleri sıvı ve batık nesneyi

temsil eder. AOA'nın işleyişini temsil eden görsel Şekil 1.11'de gösterilmiştir (Hashim ve ark., 2021):



Şekil 1.11. *AOA'nın işleyişi*

Bahsi geçen denklem ise Denklem 1.6'daki gibi yazılabilir:

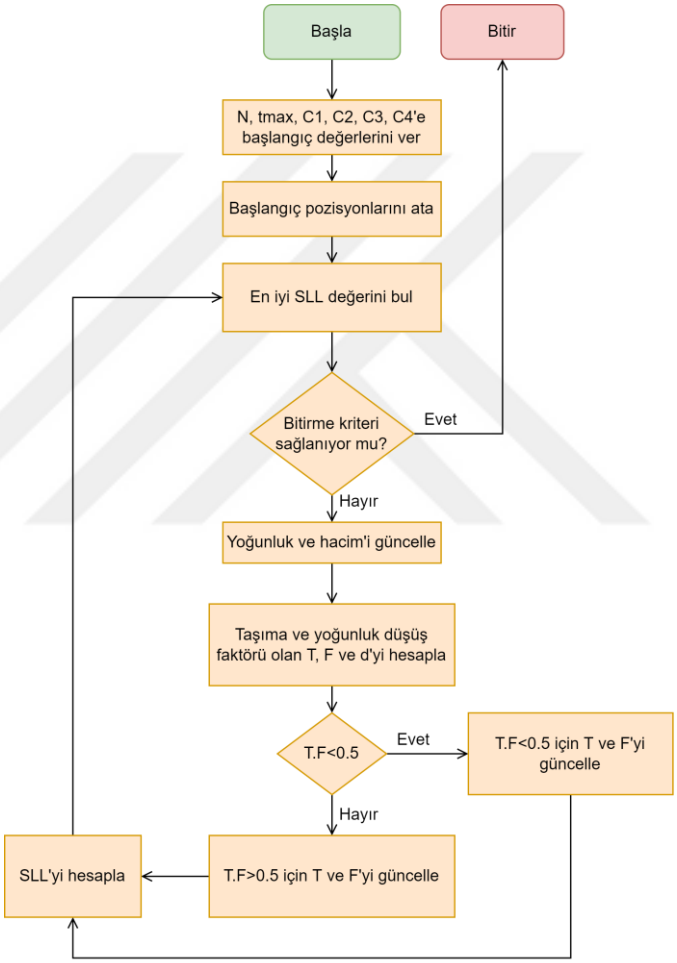
$$a_0 = \frac{p_b v_b a_b}{p_0 v_0} \tag{1.8}$$

YZ optimizasyon algoritmaları çalıştırılırken her algoritmanın farklı dâhili parametreleri bulunmaktadır. Ayrıca her algoritma farklı yöntemler ile karşılaştırılır. Bu karşılaştırmada farklı sayıda ve türde test fonksiyonları kullanılır. Tablo 1.5'te AOA algoritmasının dahili parametreleri karşılaştırıldığı fonksiyonlar ve metotlar verilmektedir (Hashim ve ark., 2021):

Tablo 1.5. *AOA algoritmasında kullanılan dahili parametreler, karşılaştırılan metotlar ve test fonksiyonları*

Algoritmalar	**Parametreler ve Değerleri**	**Karşılaştırılan metotlar**	**Uygulanan test fonksiyonları**
AOA	Obj Num.=30 C1 = 2 C2 = 6 C3 = 2 C4 = 0.5	·WOA ·PSO ·GA ·SCA ·L-SHADE ·L-SHADE-EpSin ·WOA ·HHO ·EO	·30 CEC-2017 fonksiyonu. ·Gerilim/basınç yayı tasarımı. ·Kaynaklı kiriş tasarımı. ·Basınçlı kap tasarımı. ·Hız düşürücü tasarım.

AOA hakkında daha detaylı bilgiye ulaşmak için ana makalesini inceleyebilirsiniz (Hashim ve ark., 2021). Şekil 1.12'de AOA algoritmasının antenin ışıma diyagramında en iyi SLL'yi elde etmek için izlediği yolu gösteren akış diyagramı verilmektedir:

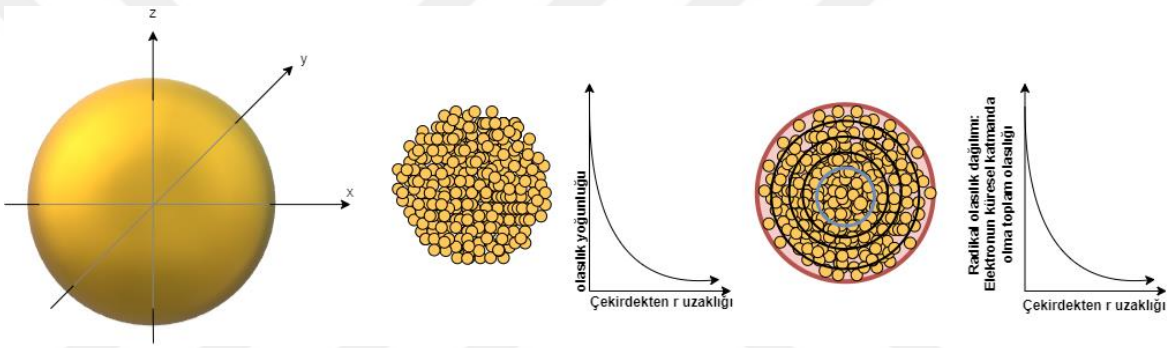


Şekil 1.12. *AOA algoritmasının akış diyagramı*

1.2.2.2. Atomik Yörünge Araması

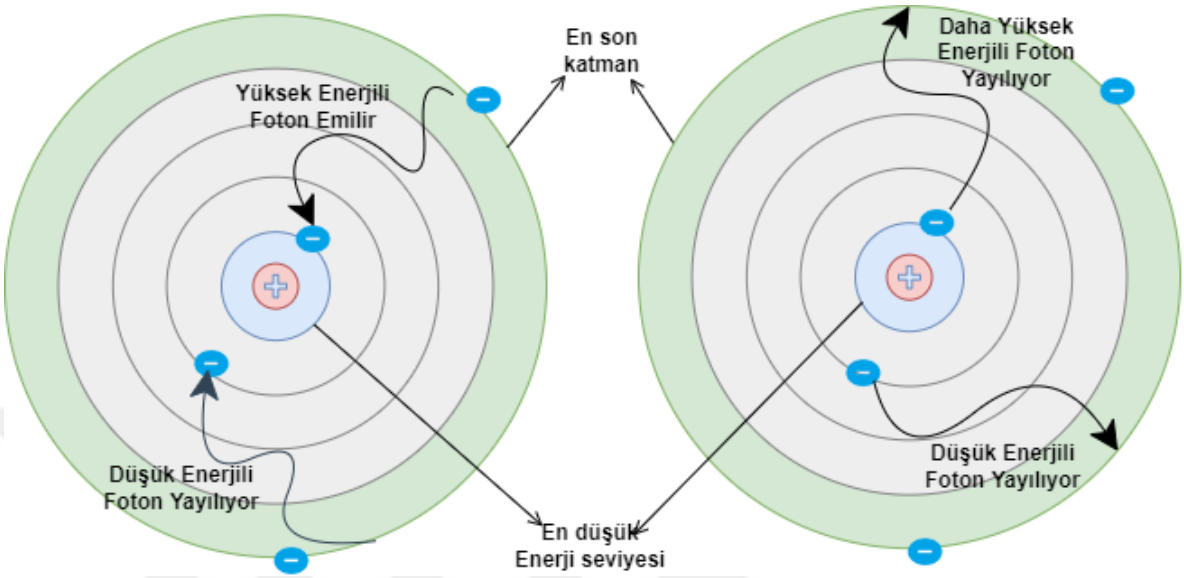
Atomik Yörünge Araması klasik atom modelinden esinlenerek Mehdi Azizi tarafından literatüre dahil edilmiştir (Azizi, 2021). AOS optimizasyon yöntemi, kuantum mekaniğinin bazı temel ilkelerine ve elektronların çekirdek etrafındaki hareketinin davranışına dayanmaktadır. Bu algoritmada belirtilen arama uzayı, eş merkezli hayali yörüngelere bölünmüş, çekirdeğin etrafındaki elektron bulutu uzayıdır. AOS'ta belirtilen hayali atomik

yörünge, Şekil 1.13'teki atom çekirdeğinin etrafındaki hayali bir yörünge olan fiziksel alanı temsil eder. Elektronların konum bilgisi mevcut değildir. Bunun yerine elektronların özel bir olasılık yoğunluk fonksiyonu (PDF) vardır. Şekil 1.13'te elektron yoğunluk fonksiyonu verilmiştir. Pozisyon değiştirirken elektronların zamana bağlı fotoğrafları çekildiğinde, bunlar bir atom yükü bulutu gibidir. Bu nedenle konum bilgisine ihtiyaç duymadan elektronların yoğun olduğu yerlerin olasılığı hesaplanarak gerekli bilgi elde edilir.



Şekil 1.13. *Bir atomun elektron yoğunluğu konfigürasyonu*

Herhangi bir mesafede çekirdeğin yörüngesinde dönen bir elektronun konumunun olasılığını bulmak için, arama uzayı eşit yarıçaplı sanal yörüngelere bölünür. Hayali yörüngelerin yarıçapı (r) eşit olduğundan, çekirdeğe yakın hayali yörüngenin hacmi, dışarıdaki yörüngeden daha küçüktür. Bu, her katmanın hacminin önceki katmandan daha hızlı arttığı anlamına gelir. Dolayısıyla içeride kalan yörüngede aranan elektronu bulmak, dışarıda kalan yörüngeye göre daha zordur. Elektronlar, temel enerji durumunda ve kabuğun içinde başlar. *n*, kuantum sayısını belirtir ve değeri belirlenir. Bu sayı elektronların enerjisi ile belirlenir. Elektronların enerjisi ile elektronun yarı çapı doğru orantılıdır. Enerjisi artarsa yarıçapı büyür, azalırsa yarıçapı küçülür. Elektron, farklı yörüngeler arasında geçiş işlemini gerçekleştirmek için uyarılır. Bir elektron uyarıldığında, bağ enerjisinden daha az enerjiye sahipse, dış yörüngede daha yüksek bir konuma hareket edecektir. Ancak, bağlanma enerjisinden daha yüksek enerji ile yüklenirse, daha düşük bir konuma sahip olacaktır. Bu durum Şekil 1.14'te resmedilmiştir:



Şekil 1.14. *Atomlarda çekirdeğin etrafındaki elektronlar için kuantum merdiven analojisi*

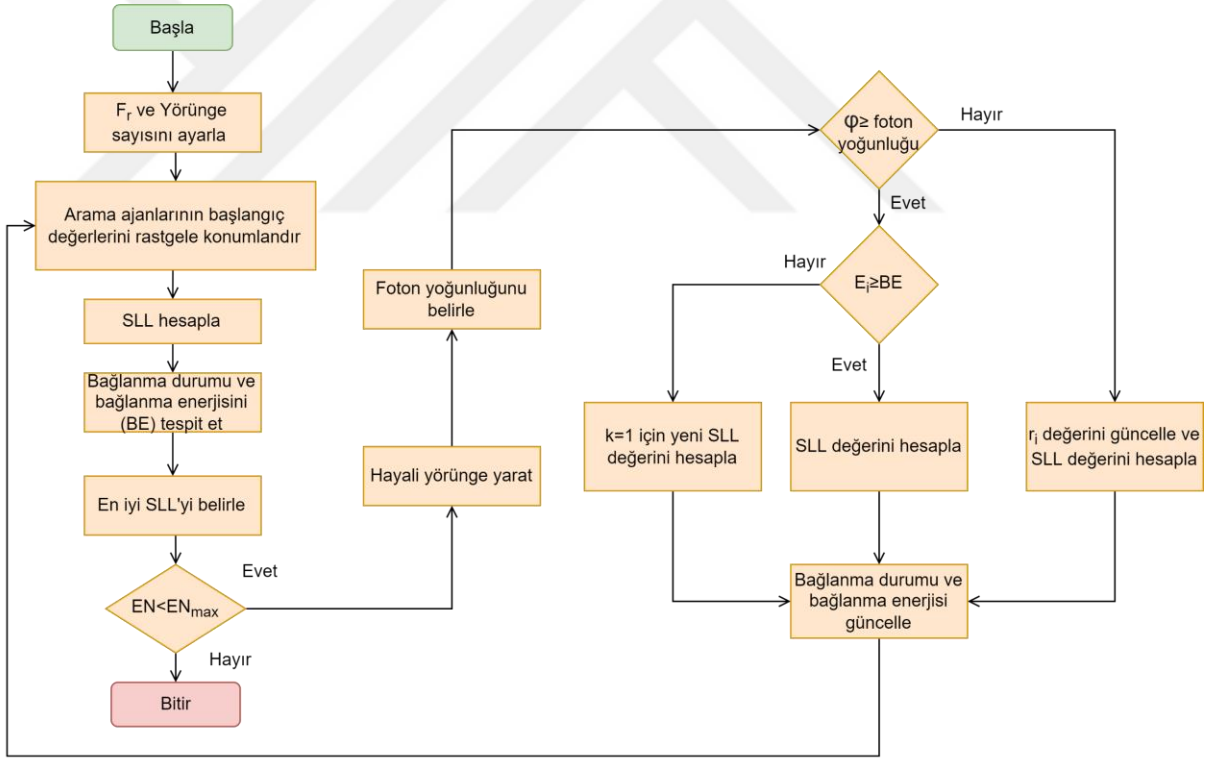
Başlangıç konumu aşağıdaki denklem yardımıyla rastgele olarak tespit edilir:

$$x_i^j(0) = x_{i,min}^j + rand.\left(x_{i,max}^j - x_{i,min}^j\right), \begin{cases} i = 1,2,3,\dots,m. \\ j = 1,2,3,\dots,d. \end{cases} \quad (1.9)$$

Adayların $x_i^j(0)$ başlangıç konumunu temsil etmektedir. $x_{i,min}^j$ minimum limiti ifade etmektedir. $x_{i,max}^j$ maksimum limiti göstermektedir. x_i^j 0 ile 1 arasında rastgele değeri (φ) olan bir sayıdır. Bu optimizasyon işlemi yapılırken kullanılan dahili parametreler ve önerilen parametre değerleri Tablo 1.6'da gösterilmiştir. AOS'a ait detaylı bilgiye ulaşmak için Azizi'nin tarafından yazılan ana makaleyi inceleyebilirsiniz (Azizi, 2021). AOS algoritmasının en iyi SLL'ye ulaşmak için izlediği yolu gösteren akış diyagramı Şekil 1.15'te gösterilmektedir:

Tablo 1.6. *AOS algoritmasında kullanılan dahili parametreler, karşılaştırılan metotlar ve test fonksiyonları*

Algoritmalar	Parametreler ve Değerleri	Karşılaştırılan metotlar	Uygulanan test fonksiyonları
AOS	·FR=0.1 ·LN=5	·GA ·BBBC ·MVO ·CSA ·SCA ·BIA ·WDO ·ASO	·20 matematiksel test fonksiyonu ·30 kıyaslama fonksiyonu (CEC-2017 kıyaslama fonksiyonu). ·5 gerçek tasarım problemi (hız düşürücü, basınçlı kap sorunu, kaynaklı kiriş sorunu, sıkıştırma yayı, çok diskli debriyaj freni sorunu)

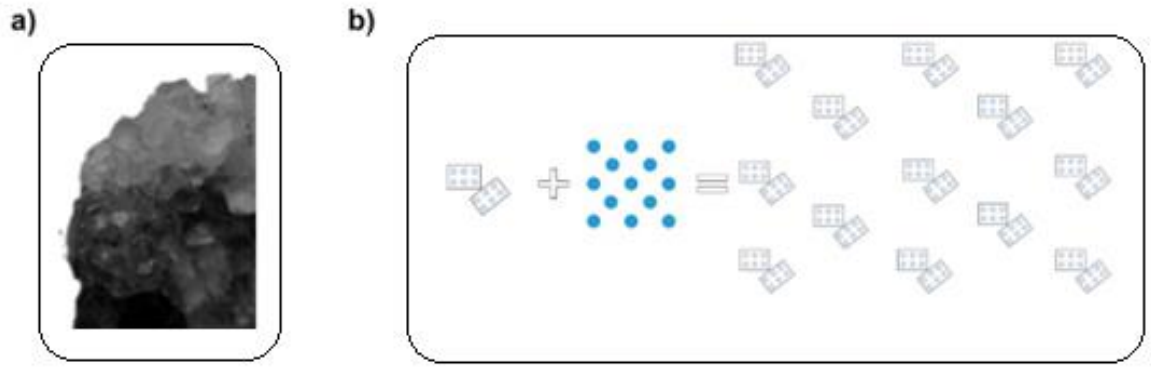


Şekil 1.15. *AOS yönteminin akış diyagramı*

1.2.2.3. Kristal Yapı Algoritması

CryStAl algoritması saf kuvars gibi kristallerin oluşumunun altında yatan prensip olan kafes noktalarına temel ekleme yönteminden esinlenerek oluşturulmuştur (Talatahari ve ark., 2021). Önceden belirlenmiş alanlarda periyodik bir dizi nokta görüntüleyen ancak

malzemedeki atomların kesin yerleşimlerini tanımlayamayan bir "kafes", bir kristalin temel bileşenidir. Kuvars kristalinin oluşumu altında fiziğin bir algoritmaya dönüştürülmesiyle oluşturulur. Kristal yapısının konfigürasyonlarını oluşturmak için matematiksel bir modele ihtiyaç vardır. Literatürde birçok kristal modeli vardır. CryStAl algoritmasında araştırmacılar Bravais adı verilen bir modelden ilham almışlardır. Bravais modelinde, her kafes noktasının bir vektör olarak kafes noktalarının konumuyla belirlendiği sonsuz bir kafes şekli göz önüne alınarak periyodik bir kristal yapı oluşturulur. Şekil 1.16'da Bravis modeli kristali ve oluşumunu gösteren yapı verilmiştir.



Şekil 1.16. *a) Bravis modeli kristal yapısı, b) Kristal oluşumu gösterimi* (Talatahari ve ark., 2021)

Bravis modeli kısaca aşağıdaki denklemde formüle edilmiştir:

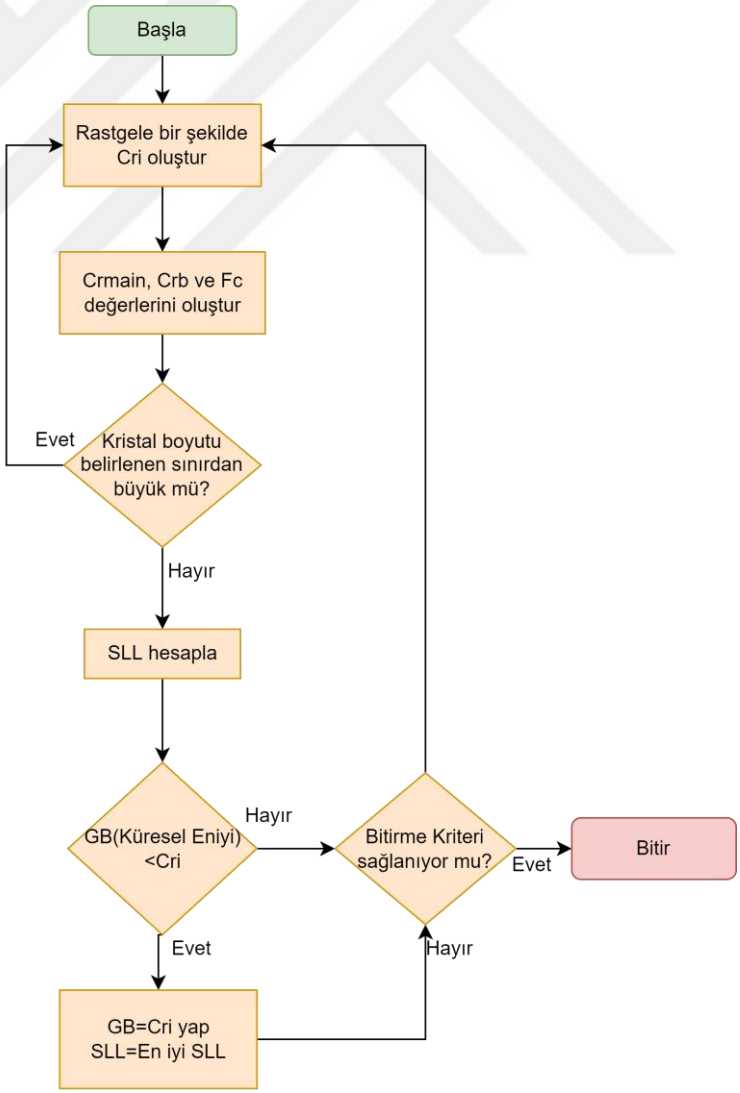
$$r = \sum n_i a_i \tag{1.10}$$

burada i kristal köşelerin sayısıdır. n_i, n. kristalin köşesinin sayısıdır, a_i birincil kristalografik yönlerle birlikte en kısa vektördür. Tablo 1.7'de CryStAl algoritmasında kullanılan dahili parametreler ve karşılaştırılan metotlar ile karşılaştırıldığı test fonksiyonları verilmiştir. Ayrıca CryStAl ile ilgili daha detaylı bilgi edinmek için Talatahari ve arkadaşlarının yazdığı makaleyi inceleyebilirsiniz (Talatahari ve ark., 2021).

Tablo 1.7. *CryStAl algoritmasında kullanılan dahili parametreler, karşılaştırılan metotlar ve test fonksiyonları*

Algoritmalar	Parametreler ve Değerleri	Karşılaştırılan metotlar	Uygulanan test fonksiyonları
CryStAl	·Cri=30	·PSO ·MVO ·ABC ·MFO ·FA ·BA ·SSA ·SCA ·GA ·HS ·ACO	·30 kıyaslama fonksiyonu (CEC-2017 kıyaslama fonksiyonu). ·6 tasarım problemi (3-5-7-9-11-13 seviyeli invertörler için senkron optimum darbe genişliği modülasyon problemleri).

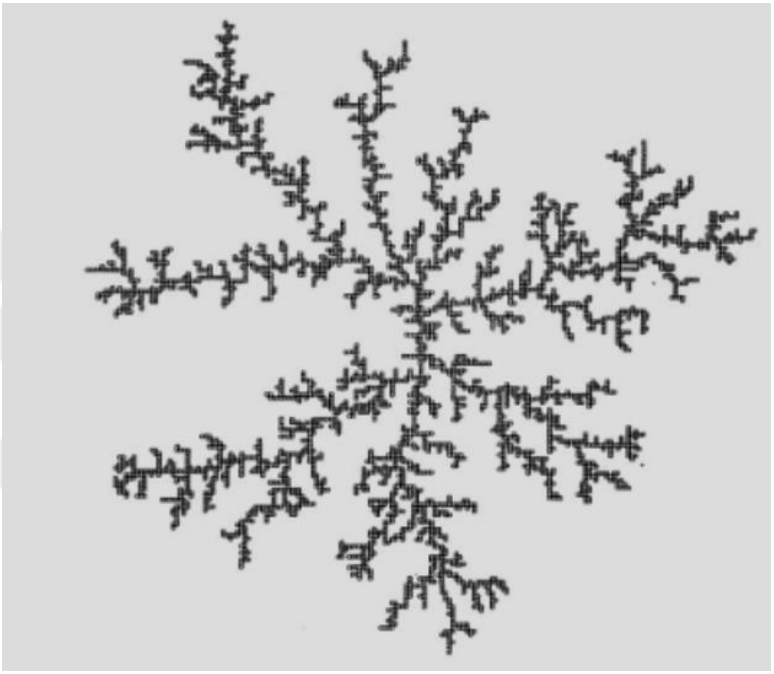
Şekil 1.17'de CryStAl algoritmasının en iyi SLL'yi elde etmek için izlediği yolu gösteren akış diyagramı verilmiştir:



Şekil 1.17. *CryStAl algoritmasının akış diyagramı*

1.2.2.4. Lichtenberg Algoritması

Lichtenberg şekli (LF) ise Witten ve Sander tarafından geliştirilen modele dayanmaktadır (Witten Jr ve Sander, 1981). LF Şekil 1.18'de gösterilmektedir:



Şekil 1.18. *Lichtenberg yapısı* (Pereira ve ark., 2021)

LA algoritması yazarları, Difüzyon Sınırlı Toplama teorisinden ilham almıştır (Pereira ve ark., 2021). DLA, ortada konumlandırılmış bir numara ile tanımlanan bir parçacık ile bir harita gibi oluşturulmuş 0'lar ve 1'lerden oluşan bir matrise sahiptir. Küme, tek değerli matris değerlerinden ve 0 değerine sahip boş alanlardan oluşmaktadır. 1 değerine sahip her matris elemanı bir küme parçacığını temsil eder ve kümedeki (N_c) sayısı programın başlangıcında belirtilir. Oluşturma yarıçapı (R_c), şeklin oluşturulduğu alanı tanımlar ve satır ve sütun numaralarının R_c'nin (çapının) iki katına eşit bir matris oluşturmak için kullanılır. LF algoritmasının matematiksel formu aşağıda verilen denklemdeki gibidir:

$$D = \frac{\ln(N_{cluster})}{\ln(R_{cluster})} \tag{1.11}$$

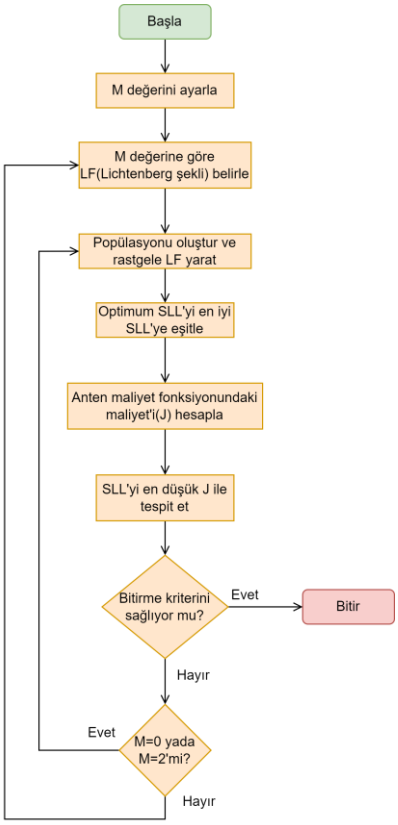
Tablo 1.8'de algoritmada kullanılan dahili parametreler karşılaştırılan metotlar ve test

fonksiyonları verilmiştir (Pereira ve ark., 2021). LA algoritmasına ait daha fazla bilgiye ulaşmak için ana makaleyi inceleyebilirsiniz (Pereira ve ark., 2021).

Tablo 1.8. *LA algoritmasında kullanılan dahili parametreler, karşılaştırılan metotlar ve test fonksiyonları*

Algoritmalar	**Parametreler ve Değerleri**	**Karşılaştırılan metotlar**	**Uygulanan test fonksiyonları**
LA	·$R_{c=}$150 ·$N_{p=}$105 ·S=1 ·Ref=0.2 ·M=0	·WOA ·PSO ·GA ·SCA ·L-SHADE ·L-SHADE-EpSin ·WOA ·HHO ·EO	Toplam 30 klasik test fonksiyonu ve 2 boyutlu 10 test fonksiyonu.

Algoritmanın en iyi SLL değerini elde etmek izlediği yolu gösteren akış diyagramı Şekil 1.19'da gösterilmektedir:



Şekil 1.19. *LA algoritmasının akış diyagramı*

1.3. Anten Dizileri

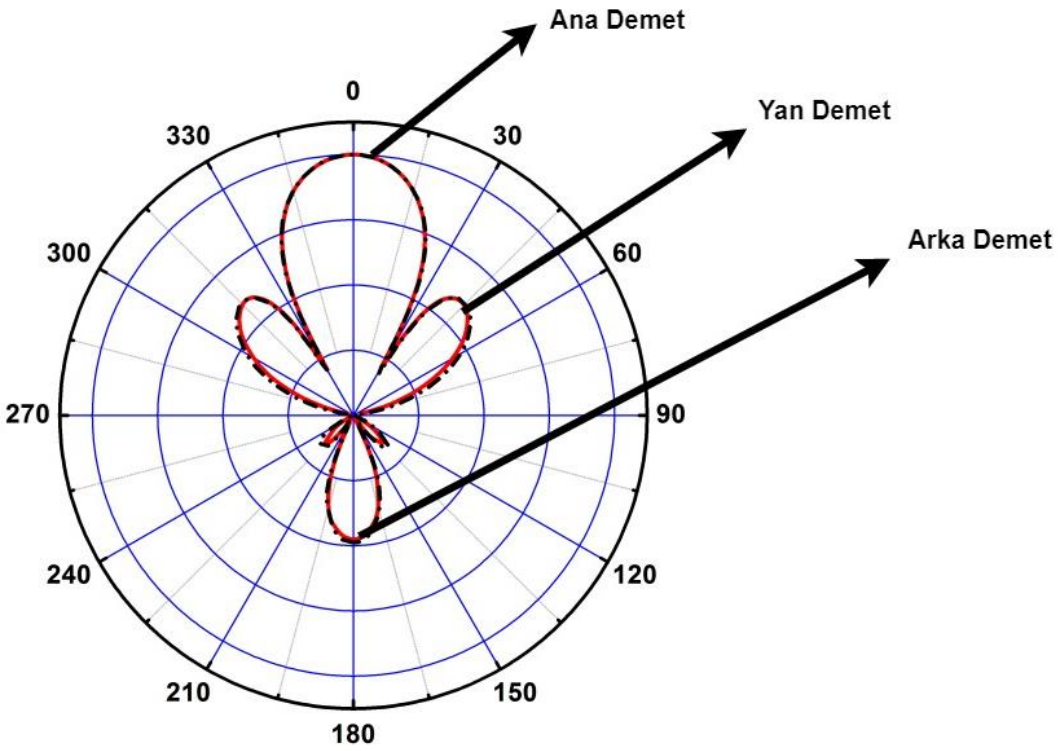
Anten dizileri radar, sonar, 5G ve savunma sanayi gibi kablosuz haberleşme alanlarında yaygın olarak kullanılmaktadır. Bu gelişen teknoloji IoT ve IIoT gibi kablosuz haberleşme teknolojilerinin ortaya çıkarılmasına vesile olmuştur. Kablosuz haberleşme teknolojisindeki yenilikler ve artan kullanıcı sayısı beraberinde daha kaliteli iletişim ve yüksek bant genişliği ihtiyacı ortaya çıkmıştır (Pisarov ve Mester, 2020). Bu gereksinimi karşılamak için mevcut antenler yetersiz kalmaktadır. Birden çok antenden meydana gelen anten dizileri istenilen ihtiyacı karşılamak için yeterli verimliliğe ulaşabilme kapasitesine sahiptir. Şekil 1.20'de anten dizisine bir örnek verilmiştir:



Şekil 1.20. *Anten dizisi gösterimi* (https://cdn.pixabay.com/photo/2017/03/05/19/53/radio-2119592_960_720.jpg Erişim Tarihi: 26.05.2022)

Anten dizilerinin ışıma diyagramlarında oluşan ana demet, arka demet ve yan demetler vardır. Anten dizilerinin yan demetleri bastırarak daha verimli hale getirilmektedir. Ancak yan demetleri bastırırken -3 dB'deki ışın genişliği olan yarı demet güç genişliğini (Half Power Beam Width-HPBW) sabit tutmak oldukça zordur. Bu zorlu optimizasyon problemini çözebilmek için yer, faz, genlik ve elemanlar arası mesafe gibi optimize edilmesi gereken birçok parametre vardır (Durmus ve ark., 2021). Fazla parametreye sahip olduğundan anten dizilerini optimize etmek hesaplamalı elektromanyetik alanında oldukça

zor bir optimizasyon problemidir. Bir anten dizisine ait ışıma diyagramında oluşan demetler Şekil 1.21'de verilmiştir.



Şekil 1.21. *Anten dizisinin ışıma diyagramı*

Anten dizilerini daha verimli hale getirmek için klasik optimizasyon metotları yetersiz kalmaktadır. Daha verimli sistemler elde etmek için yenilikçi, hızlı, yerel optimum tuzaklarına takılmama yeteneğine sahip ve birden çok arama ajanına sahip YZ optimizasyon algoritmalarına ihtiyaç vardır. Literatürde metasezgisel YZ optimizasyon metotlarının anten dizilerinin tasarlanması için kullanılması oldukça yaygındır (Chatterjee ve ark., 2012; Chen ve Wang, 2012; Durmus ve ark., 2021; Elsaidy ve ark., 2012; Feng ve Hu, 2020; Ghosh ve Das, 2011; Kenane ve ark., 2021; Lemaitre-Auger ve ark., 2013; Liang ve ark., 2020; G. X. Liu ve ark., 2021; Mandal ve ark., 2011; Ravipudi, 2020; Salgotra ve ark., 2020; Sharaqa ve Dib, 2012; Sharma ve ark., 2022; H. Singh ve ark., 2022; Singh ve Salgotra, 2018; Su ve ark., 2021; Zheng, Liu, Sun, Liang, ve ark., 2020). Optimizasyon işlemleri sayesinde anten dizilerinin verimliliği artırılarak daha kaliteli iletişim yapılması

mümkün kılınmakta, elektromanyetik kirlilik önlenmekte ve daha kazançlı anten sistemleri elde edilmektedir.

1.4.Tez Motivasyonu

Bu tezin amacı, doğrusal (LAA), dairesel (CAA) ve eş merkezli dairesel (CCAA) geometrilere sahip farklı anten dizileri tasarlayarak, kablosuz haberleşme sistemlerinin daha verimli ve etkin hale getirilmesini sağlamaktır. Farklı optimizasyon algoritmaları ile elde edilen sonuçlar karşılaştırılarak literatüre anten dizi sentezi için yeni algoritmalar önerilmesi hedeflenmiştir.

1.5. Tez Düzenlemesi

Tezin ilk bölümü, kablosuz haberleşme teknolojisi, antenler ve metasezgisel YZ optimizasyon algoritmaları hakkında bilgileri içermektedir. Tezin sonraki bölümleri, anten dizi teorisi, anten dizilerinin problem formülasyonları, anten dizi tasarımları, simülasyon sonuçları, gelecekteki çalışmalar ve tezin sonucunu içermektedir. Tezin teorik kısmından sonra sonuçlar karşılaştırmalı olarak sunulmuştur. Optimize edilince elde edilen genlik ve yer değerleri, bu değerlere karşılık gelen yarı demet güç genişliği (Half Power Beam Width-HPBW) ve yan demet seviyelerinin (Sidelobe Levels-SLLs) değerleri ve elemanlar arası mesafe değerlerinin verileri tablolar ile birlikte sunulmuştur. Daha sonra gelecekte yapılabilecek çalışmalar için önerilerde bulunulup tezin sonucuna ulaşılacaktır.

1.6. Literatür Özeti

Antenler çevreden gelen elektromanyetik dalgaları toplayan ve iletim hattı içinde yayılmasını sağlayan ya da çevreye elektromanyetik dalga yayılımı yapan yapılardır (Graf, 1999). İki ya da daha fazla antenin değişik geometrik yapılarla birleştirilmesiyle oluşturulan anten dizileri ile daha etkin bir kablosuz haberleşme sistemi tasarlanabilmektedir. Anten dizileri doğrusal, dairesel, eliptik gibi farklı geometrik yapılara sahip olabilir. Farklı uygulama sahalarında farklı anten tipleri ve geometrileri kullanılmaktadır. Kablosuz

haberleşme teknolojilerindeki gelişmelerle birlikte günümüzde daha az enerji harcayan, yönelticiliği ve kazancı daha yüksek olan anten sistemlerinin tasarlanması oldukça önemli hale gelmiştir. Bir anten sisteminin performansı anten tasarımına sıkı sıkıya bağlıdır (Agha ve ark., 2021). Anten tasarımında genellikle genlik, yer ve faz gibi temel üç parametre optimize edilir. Bu parametreler kullanılarak istenen ışıma diyagramına sahip anten sistemleri tasarlanabilmektedir (Durmus, 2020). Anten dizi tasarımıyla ilgili son yıllardaki çalışmalar incelendiğinde genellikle YZ optimizasyon algoritmalarının kullanıldığı görülmektedir. Önemli bir elektromanyetik problem olan doğrusal anten dizi (LAA) sentezi üzerine birçok çalışma yapılmıştır. LAA kullanılmasının temel sebebi kolayca uygulanabilir olmasıdır. LAA'nın genlik kontrolü ile ışıma diyagramında istenilen doğrultularda sıfırlar üretmesi için bitki gelişim simülasyon algoritması kullanılmıştır (Guney ve ark., 2009). N. Dib, düşük SLL'ye sahip doğrusal dizi modellerinin tasarımı için simbiyotik organizma arama yöntemini önermiştir (Dib, 2016). Biyocoğrafya tabanlı optimizasyon yöntemi, LAA ve eliptik anten dizilerini sentezlenmesi için kullanılmıştır (Sharaqa ve Dib, 2014). Khodier ve Al-Aqeel, optimum ışıma diyagramları elde etmek için LAA ve CAA elemanlarının genliğini, fazını ve konumunu PSO ile belirlemiştir (Khodier ve Al-Aqeel, 2009). Taguchi'nin optimizasyon yöntemi ve kendinden uyarlamalı diferansiyel evrim algoritmaları, SLL'yi bastırmış ve dizi elemanlarının farklı parametrelerini kontrol ederek istenilen doğrultularda sıfırlar üretmişlerdir (Dib ve ark., 2010). Das ve ark. yan demetleri bastırılmış ışıma diyagramlarına sahip LAA tasarlamak için güve alevi optimizasyonunu kullanmışlardır (Das ve ark., 2018). LAA'larda SLL'yi bastırmak ve derin sıfırlar üretmek için koşucu-kök algoritmasını kullanılmıştır (Subhashini, 2020). Işıma diyagramında minimum SLL elde etmek ve dizi elemanlarının genliklerini optimize etmek için geliştirilmiş çiçek tozlaşma algoritmasını önerilmiştir (Salgotra ve ark., 2020). Almagboul ve ark. SLL'yi bastırmak ve LAA'ların ışıma diyagramında sıfırlar üretmek için atom arama optimizasyon algoritmasını uygulamıştır (Almagboul ve ark., 2019).

Haberleşme sistemlerindeki farklı ihtiyaçlar doğrultusunda dairesel anten dizileri gibi karmaşık geometrilere sahip anten dizilerinin optimum tasarımı özellikle tüm yönlerde ışıma ihtiyacı olan uygulamalarda önemli bir gereklilik haline gelmiştir. Her yöne

yönlendirilebilmesi nedeniyle CAA'lar oldukça yaygın bir şekilde literatürde bulunmaktadır. Panduro ve ark. yan demetleri bastırmak ve eş aralıklı olmayan CAA'ları tasarlamak için genetik algoritmayı önermiştir (Panduro ve ark., 2006). PSO, minimum SLL elde etmek için eş aralıklı olmayan CAA'ları sentezlemek için kullanılmıştır (Najjar ve Khodier, 2008). Rattan ve ark. izotropik elemanlı CAA'larının optimizasyonu için benzetilmiş tavlama yöntemini önermişlerdir (Rattan ve ark., 2009). Khodier, CAA'ların tasarımında istenilen doğrultuda sıfırlar üretilerek ve üretilmeden SLL'yi bastırmak için guguk kuşu arama algoritmasını uyguladılar (Khodier, 2020). CAA elemanlarının genliği ve konumu biyocoğrafya tabanlı optimizasyon metodu ile optimize ederek, ışıma diyagramlarında SLL değeri minimize edilmiştir (Sharaqa ve Dib, 2014). Sharaqa ve Dib, SLL'yi bastırmak için anten dizilerinin genliğini ve konumlarını optimize ederek CAA ve CCAA tasarlamada ateş böceği algoritmasını uygulamışlardır (Sharaqa ve Dib, 2014). Sabit HPBW ve düşük SLL ile CAA ve CCAA tasarlamada arayıcı optimizasyon algoritmasının paralel uygulaması önerilmiştir (Guney ve Basbug, 2014). Ram ve ark. dizi elemanlarının genlik ve elemanlar arası mesafelerini optimize ederek CAA ve CCAA sentezinde bir kedi sürüsü optimizasyonu önermişlerdir (Ram ve ark., 2015). TO yöntemi ile optimum CAA tasarlamak için elde edilen ışıma diyagramlarında SLL bastırılmıştır (Babayigit ve Senyigit, 2017). LAA, CAA ve rastgele anten dizilerinin SLL'sini bastırmak amacıyla geliştirilmiş tavuk sürüsü optimizasyonu önerilmiştir (Liang ve ark., 2020). Zheng ve ark. dar HPBW ve bastırılmış SLL değerlerine sahip ışıma diyagramları olan LAA'ları ve CAA'ları tasarlamada rastgele mutasyon, Lévy uçuş algoritması ve gelişmiş istilacı ot optimizasyonu metotlarını kullanmışlardır (Zheng, Liu, Sun, Zhang, ve ark., 2020). LAA ve CAA sentezinde literatürdeki en güncel çalışma olarak Durmuş ve Kurban, denge optimizasyon algoritmasını kullanarak ışıma diyagramlarının yan demetlerini bastırmışlardır (Durmus ve Kurban, 2021).

Bu anten dizilerinden eşmerkezli dairesel ile çalışılmasının temel nedeni, 0 dereceden 360 dereceye kadar istenilen azimut açısı ile sinyalleri istediğimiz yönde iletebilmesidir (Swathi ve ark., 2021). Ayrıca, elemanlar arası aralığın değiştirilmesi gerektiğinde, halkaların yarıçapları değiştirilerek ve her halka için elemanlar arasındaki mesafe değiştirilerek elemanlar arası boşluk değiştirilebilir. Bu kolaylıklardan dolayı CCAA tasarımı oldukça

yaygın bir şekilde yapılmaktadır (Albagory, 2021; Das ve ark., 2019; Dutta ve Mahanti, 2021; Dutta ve ark., 2021; Khodier, 2020; Mohan ve ark., 2016; Rahaman ve ark., 2019, 2020; Ram ve ark., 2015; Satish ve ark., 2015; Sun ve ark., 2018; Swathi ve ark., 2021; Zhang ve Zhang, 2017).

Das A. ve diğerleri tarafından yapılan bir çalışmada, gri kurt optimizasyonu kullanılarak LAA ve CCAA tasarımları için optimal bir uzak alan ışıma diyagramı sentezi gerçekleştirilir (Das ve ark., 2019). Simbiyotik organizma arama algoritması kullanılarak optimum derecede azaltılmış yan demet seviyesine sahip CCAA'ların tasarımı Dib tarafından incelenmiştir (Dib, 2018). Ravipudi tarafından Rao Algoritmaları kullanılarak Düşük yan demet seviyesi ile doğrusal, dairesel ve eşmerkezli anten dizilerinin sentezi yapılmıştır (Ravipudi, 2020). Durmus A. ve Kurban R., Politika Optimize Edici'yi kullanarak yan demetleri bastırarak eşmerkezli anten dizilerinin optimal sentezini gerçekleştirmiştir (Durmus ve Kurban, 2021). Sahoo P.K. ve Mandal D., CCAA'nın yan demet seviyesini en aza indirmek için standart parçacık sürüsü optimizasyonu ve çılgınlığa dayalı parçacık sürüsü optimizasyonu (CRPSO) tekniklerini kullanarak karşılaştırmalı bir çalışma yürütmüşler ve CRPSO'nun daha iyi sonuç bulduğunu veriler ile sunmuşlardır (Sahoo ve Mandal, 2018). CCAA'nın yan demet seviyesi Dutta K.P. ve diğerleri tarafından evrimsel optimizasyon teknikleri kullanılarak minimize edilmiştir (Dutta ve Mahanti, 2020). Sun G. ve arkadaşları geliştirilmiş ayrık guguk kuşu arama algoritması (Bibiks ve ark., 2018) ve istilacı ot algoritmasını kullanarak büyük ölçekli eşmerkezli dairesel anten dizilerinin yan demet seviyelerini bastırmıştır (Sun ve ark., 2018). Jamunaa d. ve arkadaşları simbiyotik organizma araması optimizasyon tekniğini kullanarak tek tip genlik dağılımına sahip, yalnızca fazlı yeniden yapılandırılabilir CCAA tasarlamışlardır (Jamunaa ve ark., 2020). Kısaca bahsedilen tüm bu çalışmalarda, ağırlıklı olarak izotropik anten adı verilen eş yönlü ve noktasal ışıma yaptığı kabul edilen varsayımsal anten elemanları kullanılmaktadır.

İKİNCİ BÖLÜM
ANTEN TEORİSİ VE PROBLEM FORMÜLASYONU

2. Anten Teorisi

Elektromanyetik sinyalleri belirli bir düzene göre yayan ya da bu yayılan sinyalleri toplayan cihazlar anten olarak tanımlanır. Bu antenler belirli bir yönden veya her yönden gelen sinyalleri toplamak üzere tasarlanmasının yanı sıra bu sinyalleri belirli bir yöne veya her yöne yaymak için de tasarlanabilir. Yönün ayarlanabildiği gibi belirli frekans aralığında yayın yapması sağlanabilir veya istenmeyen frekanslardan gelen elektromanyetik dalgaların toplanması önlenebilmektedir.

Antenler bilindiği üzere sadece kablosuz haberleşme cihazlarında değil en küçük sensörlerden dünyanın çevresindeki büyük uydulara kadar birçok teknolojide kullanılmaktadır. Bu sayede geniş kullanım alanına ve çok çeşide sahiptir. Nerdeyse her teknolojide ihtiyaç duyulan antenler üzerine birçok çalışma yapılmıştır ve yapılmaya devam etmektedir (Awad ve Abdelazeez, 2018; Gahgouh ve ark., 2022; Lee, 2003; Lucyszyn ve Pranonsatit, 2013; Muttair ve ark., 2022; Rabinovich ve Alexandrov, 2012; Sultan ve ark., 2022).

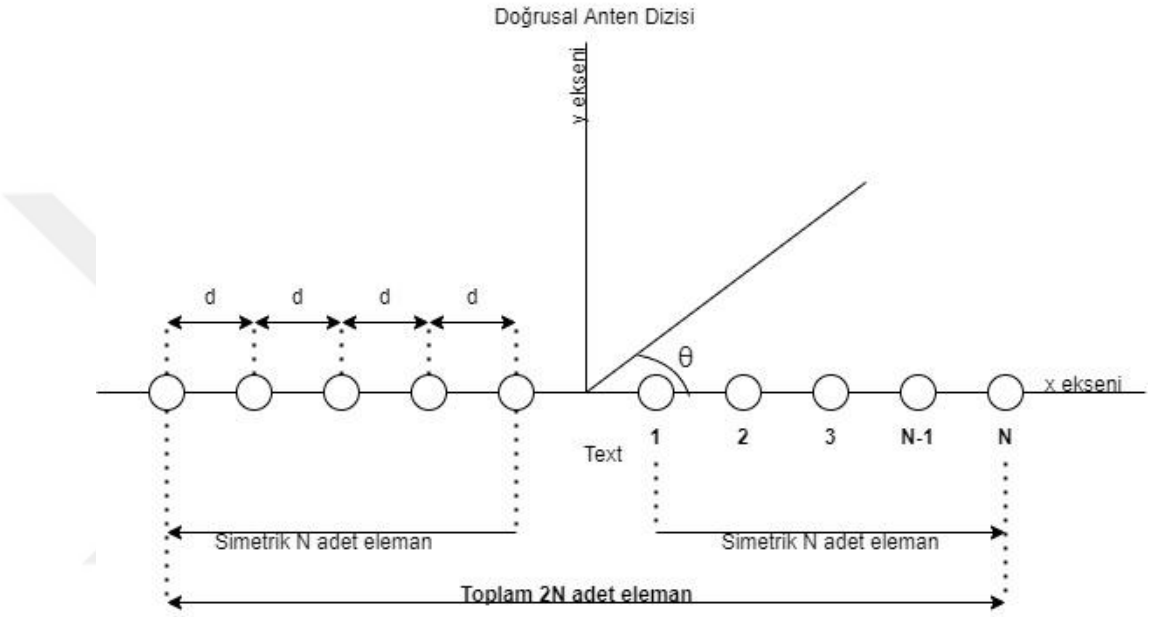
2.1. Problem Formülasyonu

Anten dizilerinin geometri özelliklerine göre farklı türleri vardır. Bu bölümde üzerinde çalışma yapılan anten dizilerinin problem formülasyonları hakkında detaylı bilgi verilecektir.

2.1.1. Doğrusal Anten Dizisi

LAA ile ilgili literatürde yapılan birçok çalışma vardır. LAA'nın bu kadar yaygın olmasının temel sebebi anten dizi problemleri içinde diğer problemlere göre daha kolay

uygulanabilmesi ve esnek olmasıdır. *2N* elemanlı x ekseni etrafında simetrik olarak yer alan LAA'nın yapısı Şekil 2.1'de gösterilmektedir.



Şekil 2.1. *2N elemana sahip doğrusal anten dizisinin yapısı*

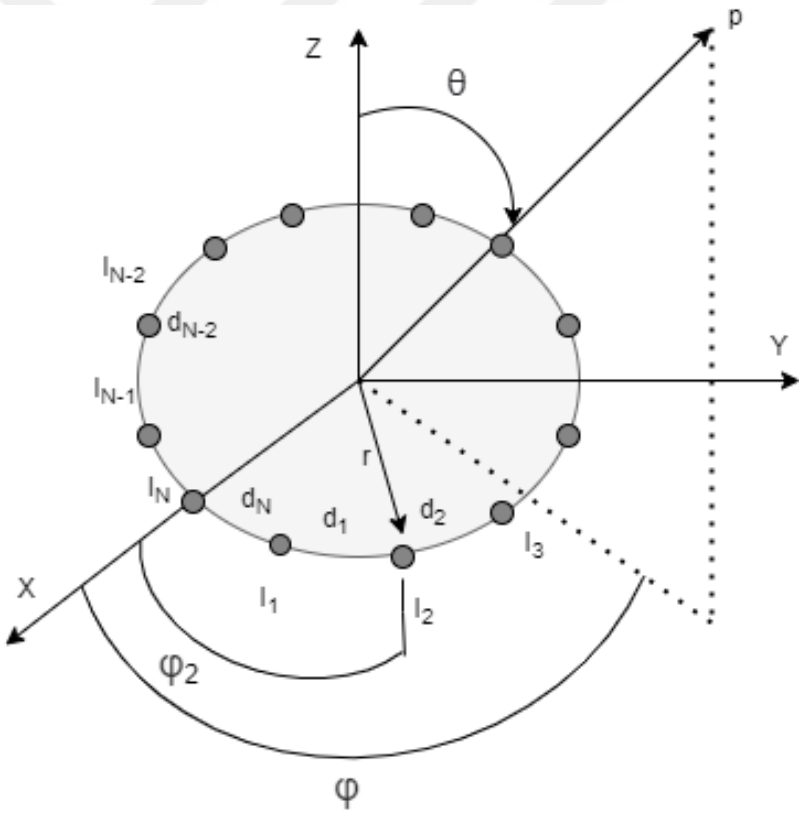
Anten dizi elemanları orijin noktası etrafında simetrik olarak yerleştirildiği için Dizi Faktörü (DF) ifadesi aşağıdaki gibi gösterilir:

$$DF = 2 * \sum_{n=1}^{N} I_n e^{(jkd_n cos\theta)} \tag{2.1}$$

Burada *n*, dizideki elemanın sırasını temsil eder. *N* ise eleman sayısını temsil etmektedir. d_n elemanlar arasındaki mesafeyi gösterir. I_n dizinin genliğini simgeler. θ antenin açısını tarif etmektedir. *k* dalga sayısını belirtir ve $2\frac{\pi}{\lambda}$ ile sunulur. Şekil 2.1'den görüleceği üzere bu yapıda toplam *2N* eleman x eksenine simetrik olarak dizildiğinden dolayı anten dizisinin sentezi yapılırken yarısı yani *N* adet elemanın parametrelerini optimize etmek için yeterli olacaktır.

2.1.2. Dairesel Anten Dizisi

Bir daire üzerinde düzgün olmayan bir şekilde konumlandırılmış N adet izotropik CAA elemanları Şekil 2.2'de gösterilmektedir. CAA'lar 0'dan 360 dereceye kadar istenilen genişlikte ayarlanabilir ve istenilen yöne doğru doğrultulabilir. Bu yüzden oldukça kullanışlı bir anten tipidir. Bahsedilen özelliklerinden dolayı literatürde oldukça çok çalışma bulunmaktadır. *x-y* düzleminde konumlanan dairenin yarıçapı *r*'dir ve bu yapıdaki tüm elemanlar benzer ışıma özellikleri sergiler.



Şekil 2.2. *CAA'nın geometrik yapısı*

CAA'nın DF'si aşağıdaki gibi hesaplanır:

$$DF(\theta,\varphi) = \sum_{n=1}^{N} I_n \, e^{(j[ka\,sin\theta\cos(\varphi-\varphi_n)+\,a_n])} \tag{2.2}$$

$$ka = {}^{2\pi a}/_{\lambda} = \sum_{i=1}^{n} d_i \tag{2.3}$$

burada λ işlemin dalga boyudur. θ ve φ sırasıyla yükseklik ve azimut açılarıdır. n'inci elemanın x–y düzlemindeki açısal konumu *n* ile temsil edilir. Bu çalışmada gösterildiği gibi, *DF*'nin x–y düzleminde olduğu varsayılır. Aşağıdaki değişiklikler yapılınca son haline ulaşılır.

$$\theta = \frac{\pi}{2}, \qquad a_n = ka\cos(\varphi_0 - \varphi_n) \tag{2.4}$$

Sonuç olarak *DF* aşağıdaki gibi yeniden yazılabilir:

$$DF(\theta, \varphi) = \sum_{n=1}^{N} I_n \exp(jka[\cos(\varphi - \varphi_n) - \cos(\varphi_0 - \varphi_n)]) \tag{2.5}$$

burada d_n, n'inci ve n+1'inci elemanlar arası boşluktur. I_n, CAA'nın n'inci elemanının geçerli genlik değeridir. φ_0 ise ana dalga yönünün 0'a ayarlanmış durumunu temsil etmektedir.

2.1.3. Eşmerkezli dairesel anten dizisi

CCAA'daki elemanların, yarıçapların ve halkaların sayısı değişebilir. Şekil 2.3, r_m yarıçaplı *M* halkalarında düzenlenen özdeş elemanlardan oluşan bir CCAA'nın geometrisini göstermektedir. Her CCAA halkası, m = 1, 2, ... *M* olmak üzere N_m öğelerinden oluşur. Merkezdeki tek bir CCAA elemanının eşdeğer DF'si şu şekilde ifade edilir:

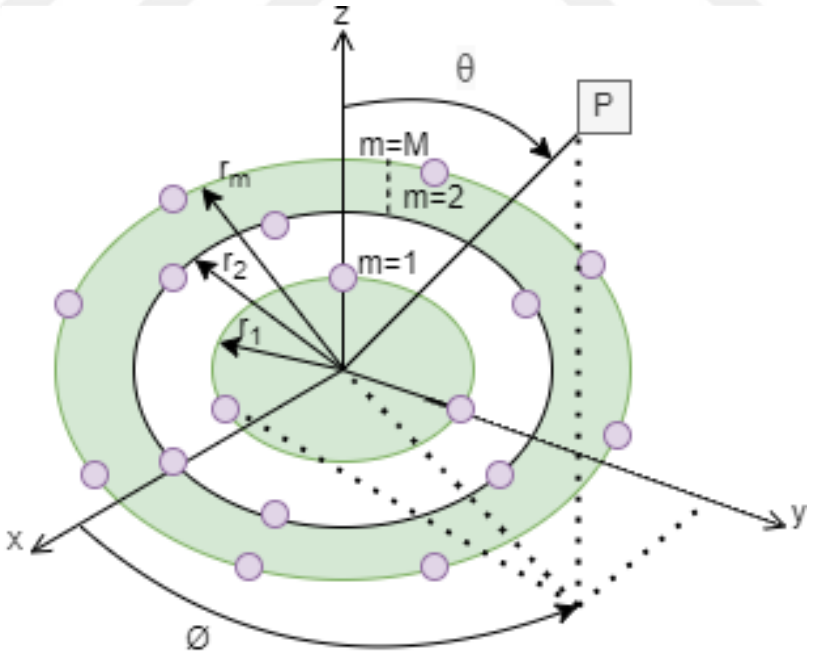
$$AF(\theta, \emptyset) = I_{merkez} + \sum_{m=1}^{M} \sum_{n=1}^{N_m} I_{mn} e^{[j(kr_m \sin(\theta)(\cos(\emptyset - \emptyset_{mn}) + a_{mn})]} \tag{2.6}$$

ayrıca,

$$\emptyset_{mn} = \frac{2\pi(n-1)}{N_m} \tag{2.7}$$

$$a_{mn} = -krsin(\theta_0)\cos(\emptyset_0 - \emptyset_{mn}) \tag{2.8}$$

I_{merkez}, λ, r_m sırasıyla merkez elemanın uyarımını, dalga boyunu ve m'inci halkanın yarıçapını temsil eder. $k = \frac{2\pi}{\lambda}$ dalga numarasını gösterir. m'inci halkanın n'inci elemanı, I_{mn}'lik bir uyarma genliğine sahiptir. $\emptyset_{mn}$ elemandan elemana açısal ayrımdır. m. halkadaki n. elemanın fazı a_{mn} ile gösterilir. Yükseklik ve azimut açıları sırasıyla $\emptyset$ ve θ ile temsil edilir. m. halka boyunca yayılan CCAA elemanlarının sayısı N_m ile gösterilir. $\emptyset_0$ ve θ_0, sırasıyla 0 derecedeki yükseklik ve azimut açılarıdır. Şekil 2.3'te eşmerkezli dairesel anten dizisinin yapısı gösterilmektedir.



Şekil 2.3. *Eşmerkezli dairesel anten dizisinin yapısı*

2.2. YAPAY ZEKA OPTİMİZASYON ALGORİTMALARININ ANTEN DİZİ SENTEZİNE UYGULANMASI

Uzun mesafelerde haberleşen ve aynı frekans bandında çalışan diğer iletişim sistemlerinden etkilenmeyen anten dizisinin sentezlenmesi, anten dizi tasarımının en önemli özelliğidir (Najjar ve Khodier, 2008). Bunu yapmak için, anten dizileri, daha düşük SLL ve daha dar HPBW değerlerine sahip ışıma diyagramlarına sahip olacak şekilde tasarlanmalıdır. Bu işlemleri yapmak için metasezgisel optimizasyon metotlarına tanımlanan bir fonksiyon vardır. Bu fonksiyona maliyet fonksiyonu denilir. En iyi SLL'lere sahip anten dizilerini elde etmek için kullanılacak maliyet fonksiyonu aşağıdaki gibi formüle edilmiştir.

$$MF = F_{SLL} \cdot S_{SLL} + F_{HPBW} \cdot D_{HPBW} \tag{2.9}$$

burada F_{SLL} ve F_{HPBW} sırasıyla SLL'nin ve HPBW'nin ağırlık faktörüdür. S_{SLL} ve D_{HPBW}, sırasıyla SLL'yi azaltmak ve HPBW değerlerini daraltmak için kullanılmıştır. S_{SLL} aşağıdaki gibi hesaplanabilir,

$$S_{SLL} = \int_{-\pi}^{\varphi_{n1}} C_{SLL}(\theta).d\theta + \int_{\varphi_{n2}}^{\pi} C_{SLL}(\theta)\,.d\theta \tag{2.10}$$

Ana kirişin her iki tarafındaki ilk sıfırlardaki iki açı θ_{n1} ve θ_{n2}'dir. Aşağıda $C_{SLL}(\theta)$'a ait bir örnek verilmiştir.

$$C_{SLL}(\theta) = \begin{cases} SLL_{elde\ edilen} - SLL_{istenen,} & SLL_{elde\ edilen} > SLL_{istenen,} \\ 0 & , Diğer\ durumlarda \end{cases} \tag{2.11}$$

burada $SLL_{istenen,}$ ulaşılmak istenen SLL değerini temsil ederken $SLL_{elde\ edilen}$ ise elde edilmiş olan SLL değerini tanımlar. D_{HPBW} değeri aşağıdaki gibi hesaplanır:

$$D_{HPBW} = \begin{cases} D_{elde\ edilen} - D_{istenen}, D_{elde\ edilen} > D_{istenen} \\ 0 \qquad\qquad\qquad\qquad , Diğer\ durumlarda.. \end{cases} \tag{2.12}$$

burada $D_{elde\ edilen}$ve $D_{istenen}$ sırasıyla metasezgisel algoritmalar ile elde edilen HPBW ve ulaşılmak istenen HPBW değerleridir.

ÜÇÜNCÜ BÖLÜM
SONUÇLAR

3. YAPILAN OPTİMİZASYON ÇALIŞMALARI VE ELDE EDİLEN SİMÜLASYON SONUÇLARI

3.1. LAA Sentezine PSO, ABC, HBA ve CSA Metotlarının Uygulanması

Bu çalışmada 10, 16, 24 ve 32 elemanlı LAA'lar optimize edilmiş ve 4 farklı metasezgisel optimizasyon metoduyla karşılaştırılmıştır. İlk iki algoritma, literatürde oldukça yaygın olarak kullanılan sürü tabanlı metasezgisel algoritmalar PSO ve ABC'dir. Diğer iki algoritma ise literatüre yeni kazandırılmış metasezgisel algoritmalar olan HBA ve CSA'dır. Bu yeni optimizasyon metotlarının LAA sentezindeki performansını test etmek için farklı sayıda elemana sahip LAA'lar incelenmiştir. Tüm simülasyonlar sabit HPBW değerleri ile minimum SLL değerlerine ulaşan ışıma diyagramları elde etme amacı ile yapılmıştır.

PSO, ABC, HBA ve CSA yöntemlerinin CPU zamanı ve SLL değerleri açısından istatistiksel karşılaştırmaları yapılmıştır. Bu istatistiksel analiz için tüm metotlar rastgele 30 kez çalıştırılmıştır. Tüm simülasyon çalışmaları i5 işlemciye ve 8GB RAM'e sahip kişisel bir bilgisayarda yapılmıştır. Bu çalışmaların sonucunda en iyi değerler ve ortalama değerler SLL ve CPU zamanı cinsinden Tablo 3.1'de gösterilmektedir.

Tablo 3.1. *PSO, ABC, HBA ve CSA metotları ile elde edilen SLL ve CPU zamanının istatistiksel değerleri*

	Metot	**PSO**	**ABC**	**HBA**	**CSA**
10 Eleman	**SLL_{Best}(dB)**	-21.355	-21.065	-21.430	-21.430
	SLL_{Mean} (dB)	-19.441	-20.247	-21.211	-20.809
	$CPUT_{min}$(sn)	6.758	10.728	2.331	2.154
	$CPUT_{Mean}$(sn)	7.149	11.148	2.448	2.232
	$CPUT_{Max}$(sn)	7.807	12.053	2.753	2.580
16 Eleman	**SLL_{Best}(dB)**	-24.579	-22.799	-24.826	-23.458
	SLL_{Mean}(dB)	-22.194	-22.240	-23.644	-22.643
	$CPUT_{Min}$(sn)	7.548	11.998	3.133	3.028
	$CPUT_{Mean}$(sn)	7.797	12.098	3.178	3.048
	$CPUT_{Max}$(sn)	8.133	12.566	3.218	3.101
24 Eleman	**SLL_{Best}(dB)**	-26.173	-24.086	-26.515	-26.099
	SLL_{Mean}(dB)	-23.782	-23.173	-25.872	-24.065
	$CPUT_{Min}$(sn)	8.875	13.811	4.293	4.211
	$CPUT_{Mean}$(sn)	9.009	13.909	4.378	4.2686
	$CPUT_{Max}$(sn)	9.114	14.057	4.435	4.3231
32 Eleman	**SLL_{Best}(dB)**	-25.232	-23.743	-25.423	-25.114
	SLL_{Mean}(dB)	-23.592	-22.989	-24.147	-23.166
	$CPUT_{Min(}$sn)	10.024	15.718	5.496	5.4139
	$CPUT_{Mean}$(sn)	10.201	15.804	5.591	5.495
	$CPUT_{Max}$(sn)	10.299	15.909	5.852	5.569

PSO, ABC, HBA ve CSA metotlarının SLL ve CPU zamanı açısından karşılaştırması Tablo 3.1'de verilmiştir. Tablo 3.1'de gösterildiği gibi, CSA yönteminin diğer optimizasyon tekniklerinden daha hızlı optimal sonuca ulaştığı görülmüştür. SLL'lerinin bastırılması için incelendiğinde HBA yöntemiyle elde edilen sonuçlar genellikle PSO, ABC ve CSA'dan daha iyidir.

10 elemanlı LAA için en iyi SLL değeri, HBA algoritması ile elde edilmiştir. HBA yöntemi, 16 elemanlı LAA için -24.826 dB'lik en iyi SLL değerini bulmuştur. Bu sonuca yaklaşık 3.2 saniyede ulaşmıştır. 24 elemanlı LAA'da HBA yöntemi en iyi SLL değerini elde etmiştir. HBA yöntemi en iyi SLL değerine ulaşmak için 4.29 saniye işlem süresi harcamıştır. Son olarak 32 elemanlı LAA'da HBA yöntemi SLL değerini 5.49 saniyede diğer üç optimizasyon yönteminden daha iyi bulmuştur. Genel olarak YZ optimizasyon

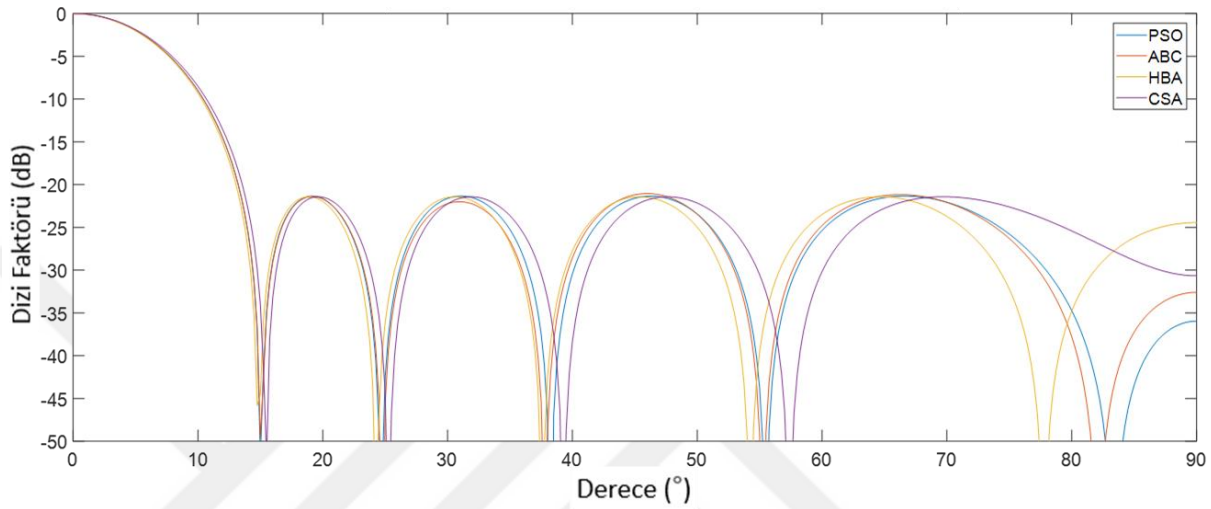
algoritmalarına baktığımızda diğer algoritmalara göre en hızlı yöntem CSA olmuştur. En iyi SLL değerine ulaşan metot HBA olmuştur. Ayrıca HBA ve CSA yöntemleri, PSO ve ABC yöntemlerine göre çok daha hızlıdır ve daha iyi değerler bulmuştur. PSO, ABC, HBA ve CSA ile elde edilen LAA'nın konum değerleri Tablo 3.2'de verilmiştir.

Tablo 3.2. *PSO, ABC, HBA ve CSA metotları ile elde edilen LAA elemanlarının konumları*

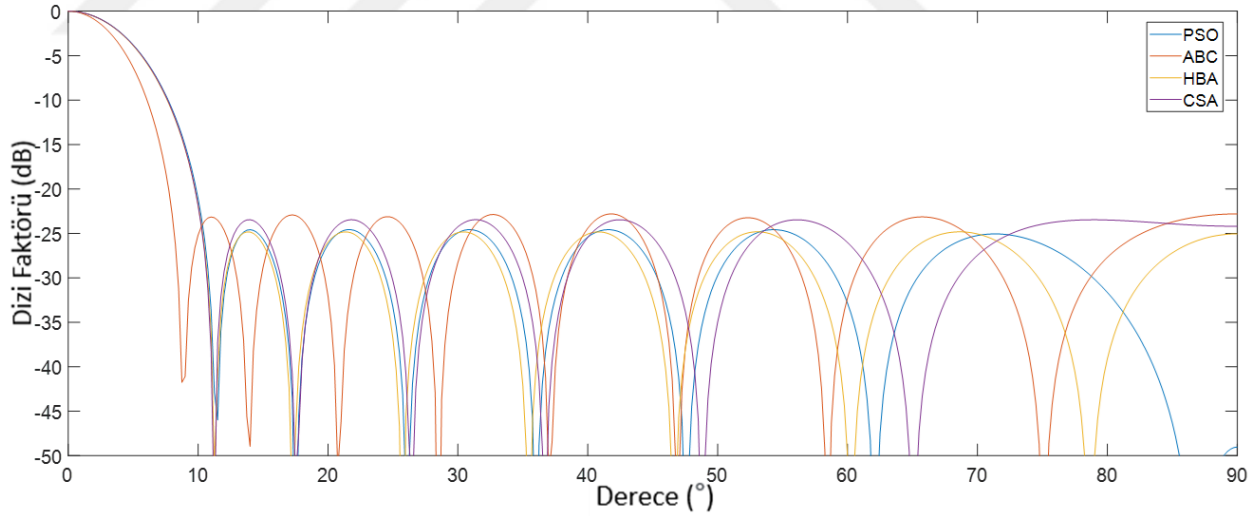
	Metot	**LAA Elemanlarının Pozisyonları [d_1 d_2....d_n]**
10 Eleman	**PSO**	[0.3287 0.4080 0.9853 1.3782 2.0463]
	ABC	[0.3707 0.3707 0.9882 1.3716 2.0491]
	HBA	[0.3736 0.3736 1.0010 1.3945 2.0753]
	CSA	[0.3589 0.3610 0.9644 1.3435 1.9994]
16 Eleman	**PSO**	[0.1437 0.5472 0.6130 1.2200 1.3155 1.8745 2.3196 3.0091]
	ABC	[0.2375 0.5203 0.9902 1.3452 1.8401 2.3203 2.9808 3.7430]
	HBA	[0.3390 0.3390 0.6981 1.2578 1.2932 1.8997 2.3360 3.0464]
	CSA	[0.1072 0.4870 0.7291 1.1064 1.4357 1.8527 2.3572 2.9865]
24 Eleman	**PSO**	[0.1417 0.4794 0.7390 1.0727 1.4707 1.6627 2.1890 2.4516 2.9830 3.4487 4.1356 4.9066]
	ABC	[0.0241 0.5402 0.7810 1.1629 1.5228 1.8772 2.3196 2.7078 3.1661 3.7285 4.3909 5.1390]
	HBA	[0.3094 0.3094 0.7306 1.0910 1.2924 1.7086 2.0190 2.4000 2.8399 3.3109 3.9545 4.7147]
	CSA	[0.2040 0.4344 0.8208 1.1062 1.4964 1.7814 2.2445 2.5812 3.0981 3.5940 4.3027 5.09809]
32 Eleman	**PSO**	[0.1886 0.5935 0.9212 1.3643 1.6757 2.1449 2.5301 2.9816 3.3957 3.8569 4.4497 4.9975 5.5445 6.3796 7.2335 8.0756]
	ABC	[0.1480 0.6483 1.0024 1.4041 1.8720 2.3341 2.7026 3.2151 3.6673 4.2105 4.7801 5.3231 6.0406 6.8581 7.7682 8.4942]
	HBA	[0.0160 0.7334 0.7960 1.4105 1.6247 2.1738 2.5253 2.9774 3.4047 3.8513 4.4427 4.9725 5.5481 6.3888 7.2280 8.0959]
	CSA	[0.1536 0.6081 0.9402 1.3590 1.7417 2.1169 2.5645 3.0009 3.4808 3.9266 4.4680 4.9916 5.6481 6.4891 7.2552 8.1626]

Sayısal sonuçlardan da görülebileceği gibi, önerilen HBA ve CSA optimizasyon metotları, LAA'ların SLL'sini bastırmada çok başarılı olmuştur. Tablo 3.2'de verilen konum değerleri, PSO, ABC, HBA ve CSA olmak üzere dört farklı metasezgisel metotla elde edilen optimum ışınım modelinin verileridir. Tablo 3.2'de verilen 10, 16, 24 ve 32 elemanlı LAA'ların konum değerleri ile elde edilen karşılaştırmalı ışıma diyagramları Şekil 3.1'de

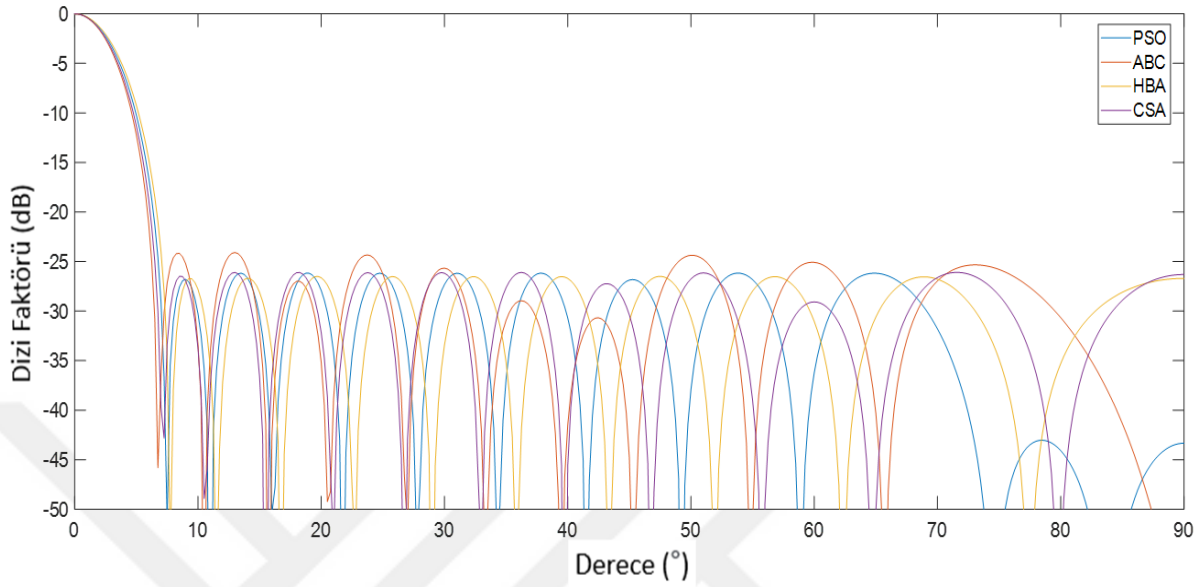
gösterilmiştir. Şekil 3.2, 30 rastgele çalışma ile elde edilen farklı sayıda elemana sahip LAA'ların yakınsama eğrilerini göstermektedir.



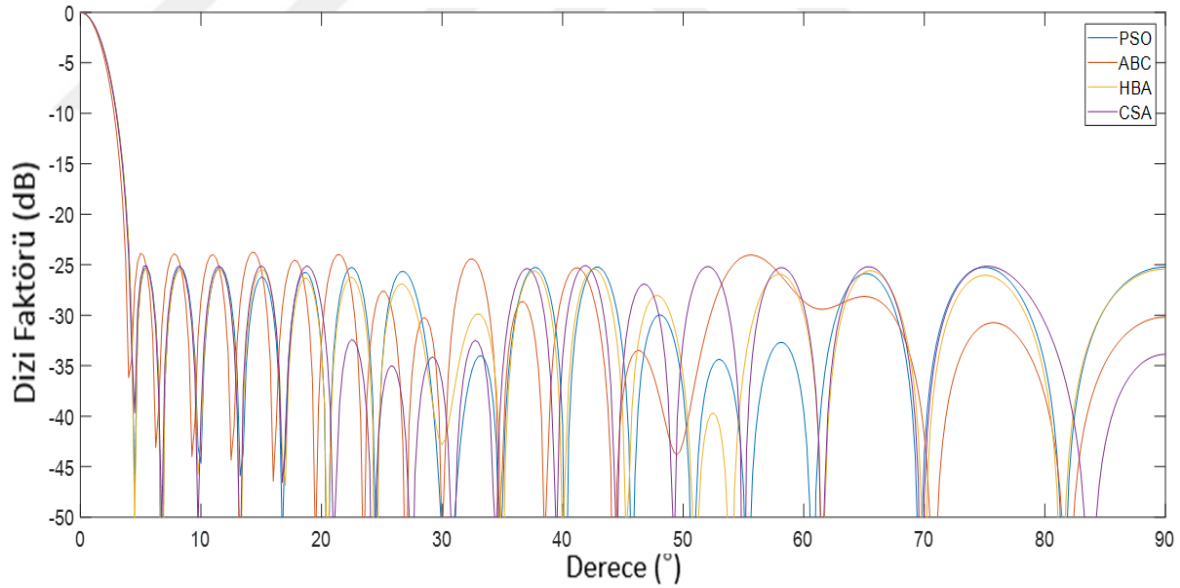
(a)



(b)

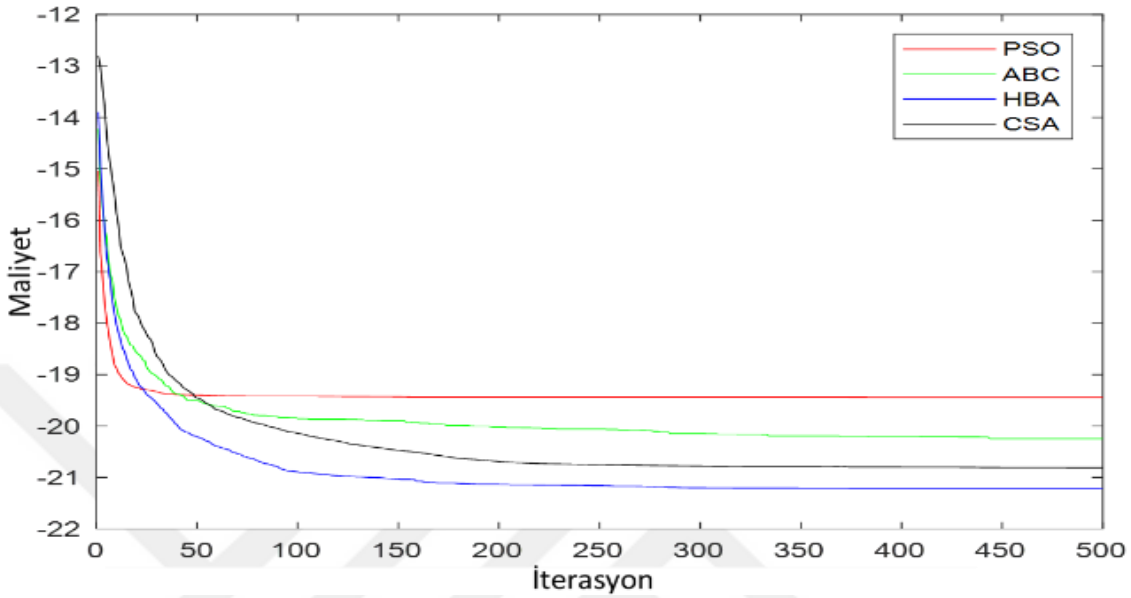


(c)

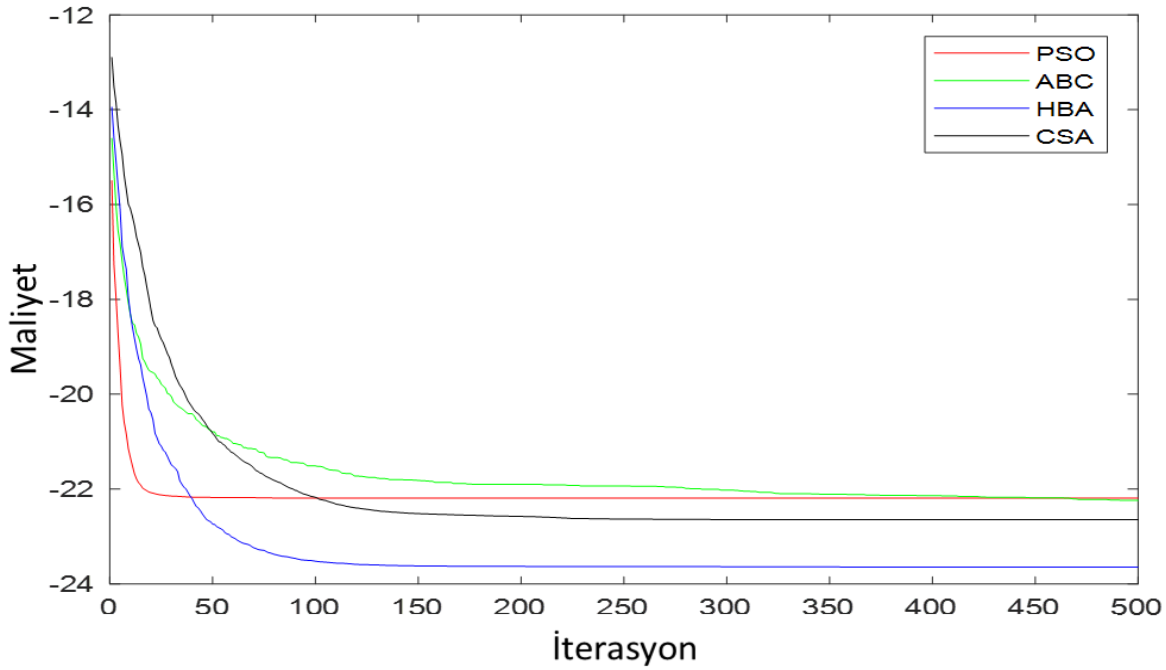


(d)

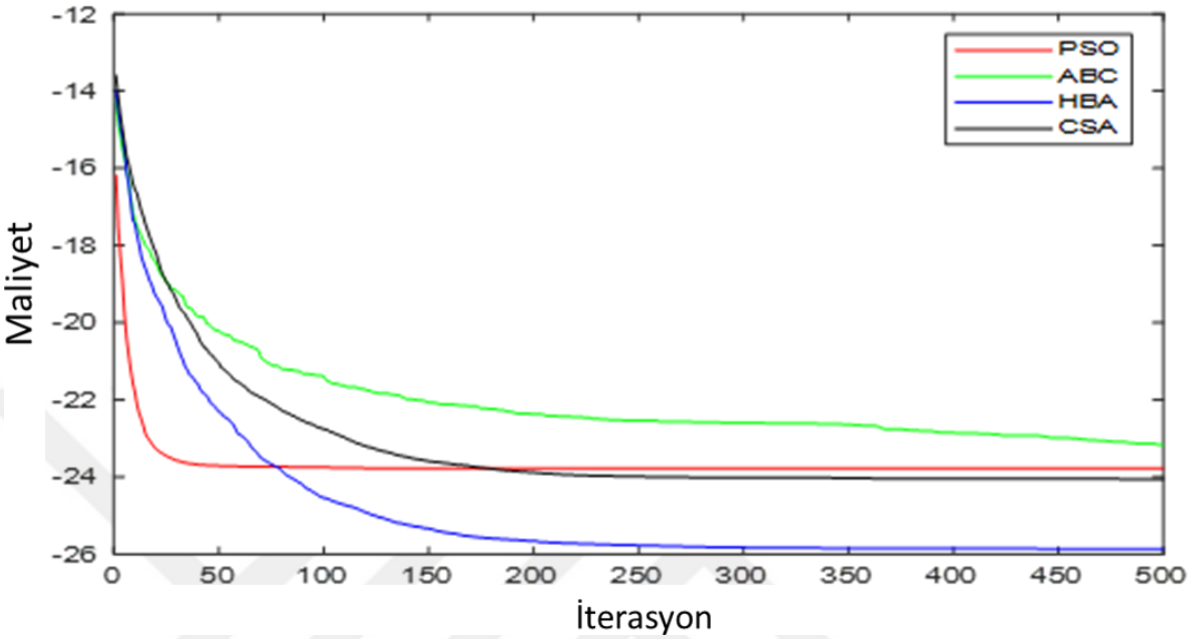
Şekil 3.1. a) 10 elemanlı, *b) 16 elemanlı, c) 24 elemanlı, d) 32 elemanlı LAA için PSO, ABC, HBA ve CSA ile elde edilen ışıma diyagramları*


PSO
ABC
HBA
CSA
Maliyet
İterasyon
-12
-13
-14
-15
-16
-17
-18
-19
-20
-21
-22
0
50
100
150
200
250
300
350
400
450
500

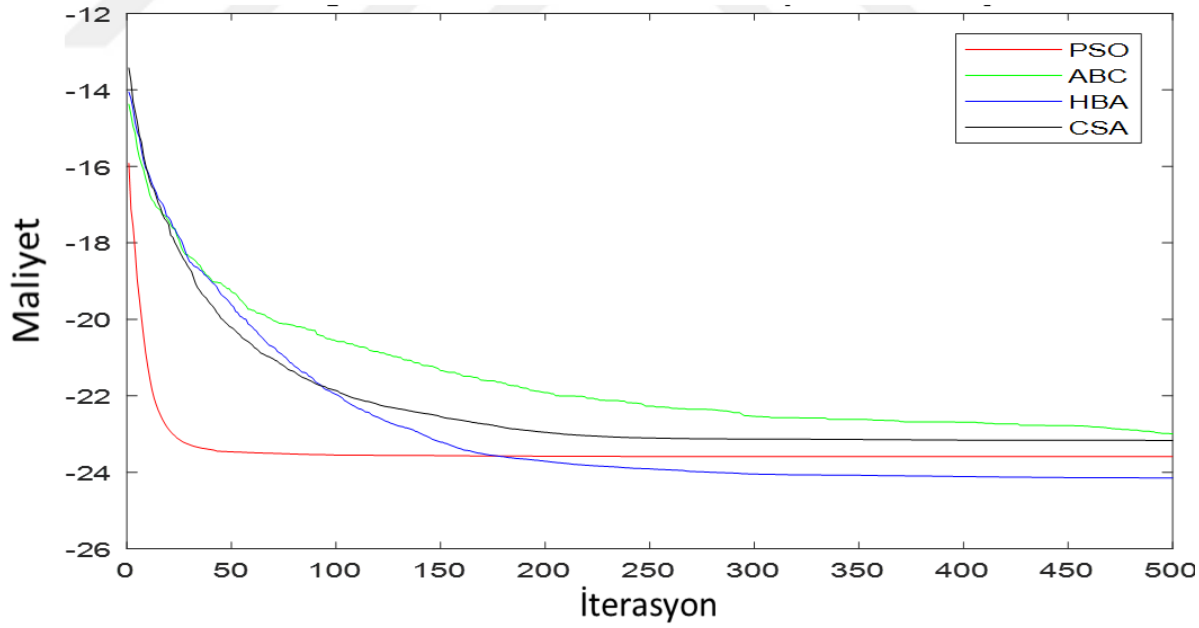
(a)


PSO
ABC
HBA
CSA
Maliyet
İterasyon
-12
-14
-16
-18
-20
-22
-24
0
50
100
150
200
250
300
350
400
450
500

(b)



(c)



(d)

Şekil 3.2. a) 10 elemanlı, *b) 16 elemanlı, c) 24 elemanlı, d) 32 elemanlı LAA için PSO, ABC, HBA ve CSA ile elde edilen yakınsama eğrisi*

Şekil 3.2'ye göre HBA ile elde edilen yakınsama eğrileri diğer 3 metottan daha iyidir. Şekil 3.2'de görüldüğü gibi HBA ve CSA yöntemleri, PSO ve ABC optimizasyon yöntemlerinden daha hızlı optimum sonuca ulaşmıştır.

Bu bölümde, LAA'ların SLL'leri, yalnızca elemanların konumlarını kontrol ederek dört metasezgisel metot olan PSO, ABC, CSA ve HBA ile bastırılmıştır. Önerilen HBA ve CSA metotlarının performansları ABC ve PSO ile karşılaştırılmıştır. Sonuçlar, HBA ve CSA metotlarının ABC ve PSO'dan daha iyi SLL değerleri elde ettiğini göstermektedir. Ayrıca, HBA ve CSA, CPU hesaplama sürelerine kıyasla PSO'dan ~2x ve ABC'den ~3x daha hızlı performans gösterir. Bu sonuçlar önerilen metotların farklı anten tasarımları için de kullanılabileceğini ortaya koymuştur.

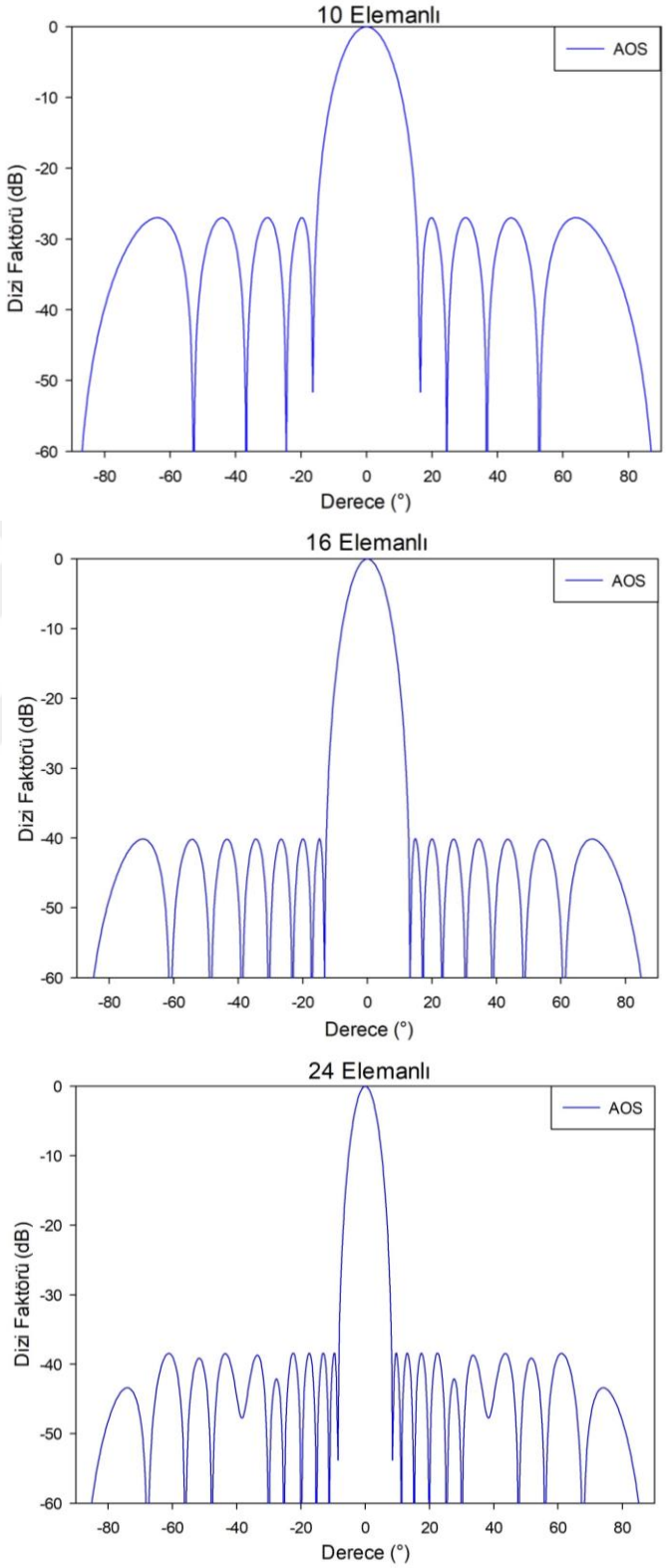
3.2. LAA Sentezine AOS Metodunun Uygulanması

Bu çalışmada AOS metodu ile 10, 16 ve 24 elemanlı LAA'nın tasarımı yapılmıştır. Bu tasarımı yaparken hedefimiz HPBW'i minimumda tutarak SLL'yi bastırmaktır. AOS optimizasyon metodu ile LAA'nın sadece genliklerinin optimizasyonu yapılmıştır. Bu işlemde popülasyon 50 ve sonlandırma kriteri 1000 iterasyon olarak belirlenmiştir. Yapılan simülasyonlarda MATLAB yazılımı kullanılmıştır. Bütün simülasyonlar 2.5GHz CPU hızına sahip i5 işlemcili bir kişisel bilgisayar yardımı ile yapılmıştır. 10, 16 ve 24 elemanlı LAA'ların AOS metoduyla elde edilen SLL ve genliklerin değerleri Tablo 3.3'te verilmiştir.

Tablo 3.3. *AOS ile elde edilen SLL ve genlik değerleri*

	$SLL_{Enİyi}$ (dB)	**Genlik Değerleri [I_1 I_2 I_3…I_n]**
10 Elemanlı	-26.95	[1.0000 0.8905 0.7002 0.4742 0.3314]
16 Elemanlı	-40.15	[1.0000 0.9351 0.8157 0.6604 0.4913 0.3306 0.1951 0.1123]
24 Elemanlı	-38.40	[1.0000 0.9396 0.9074 0.8420 0.7317 0.6247 0.5411 0.4021 0.3125 0.2421 0.1509 0.1134]

Bu tabloda elde edilen sonuçlara göre 10 elemanlı LAA'da elde edilen en iyi SLL değeri -26.95 (dB)'dir. 16 elemanlı dizide elde edilen en iyi SLL değeri -40.15 (dB)'dir. 24 elemanlı LAA'da ise en iyi SLL -38.4 (dB) olarak elde edilmiştir. Tablo 3.3'de AOS optimizasyon metoduyla elde edilen genlik verileri kullanarak tasarladığımız antenlerin ışıma diyagramları Şekil 3.3'de gösterilmiştir.



Şekil 3.3. *AOS ile elde edilen 10, 16 ve 24 elemana sahip LAA'ların ışıma diyagramı gösterilmektedir*

Literatüre yeni kazandırılmış olan AOS metoduyla 10, 16 ve 24 elemana sahip LAA'ların sentezi başarılı bir şekilde gerçekleştirilmiştir. Bu optimizasyon sürecinde LAA'ların sadece genlikleri optimize edilmiş ve istenilen ışıma diyagramları elde edilmiştir. Ayrıca bu çalışma sonucunda ulaşılan değerler AOS'un farklı geometrideki anten dizilerinin sentezinde de kullanılabileceği sonucunu ortaya çıkarmıştır.

3.3. LAA Sentezine AOA, CryStAl ve LA Metotlarının Uygulanması

Bu çalışma 3 farklı eleman sayısına sahip LAA'ya uygulanmıştır. LAA'nın eleman sayıları 10, 16 ve 24 olarak belirlenmiştir.

3.3.1. 10 Elemanlı LAA Sentezine AOA, CryStAl ve LA Metotlarının Uygulanması

İlk örnekte eleman sayısı 10 olan simetrik LAA için genlik değerleri AOA CryStAl ve LA kullanılarak optimum olarak belirlenmiştir. Tablo 3.4, AOA, CryStAl ve LA metotlarının SLL ve HPBW açısından istatistiksel olarak karşılaştırmasını göstermektedir.

Tablo 3.4. *10 elemanlı LAA için karşılaştırmalı sonuçlar*

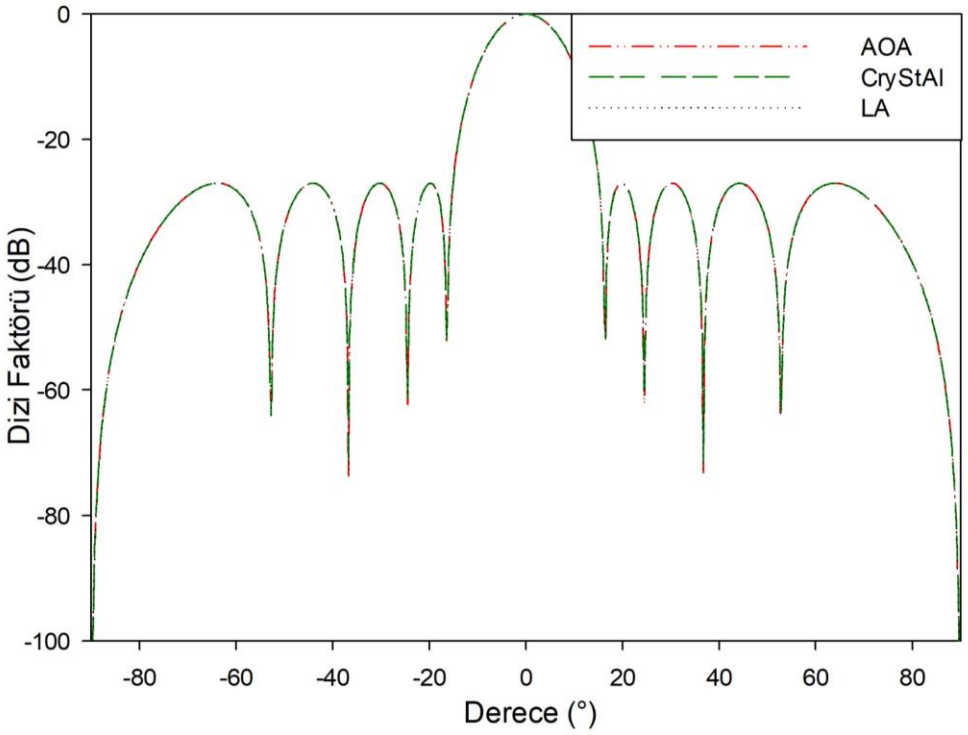
	SLL_{min}	$SLL_{ortan.}$	SLL_{max}	SLL_{std}	$HPBW_{ort}$	$HPBW_{med}$	$HPBW_{max}$	$HPBW_{std}$
AOA	-26.9772	-26.9768	-26.9677	0.0019	12.5000	12.5000	12.5000	$222x10^{-9}$
CryStAl	-26.9772	-26.9772	-26.9772	0.0000	12.5000	12.5000	12.5000	$87x10^{-9}$
LA	-26.9755	-26.9610	-26.9171	0.0146	12.4983	12.4998	12.5000	$4x10^{-5}$

CryStAl, Tablo 3.4'te verilen değerlere göre en iyi SLL'yi elde etmiştir. HPBW değerleri incelendiğinde, algoritmalar arasında gözle görülür bir fark yoktur. CPU zaman değerleri standart sapmaya göre incelendiğinde en kararlı algoritma 0.778 saniye ile LA'dır. En iyi SLL değeri sonuçlarının genlik değerleri Tablo 3.5'te verilmiştir.

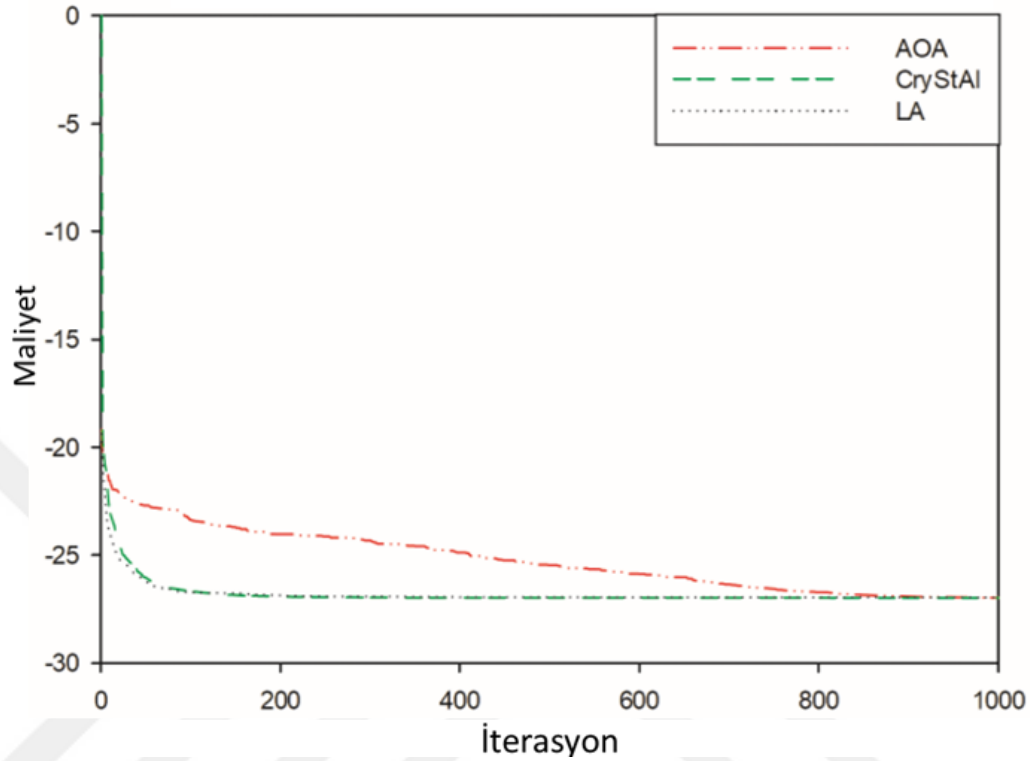
Tablo 3.5. *10 elemanlı LAA'ya ait genlik değerleri*

	Genlik Değerleri [I_1 I_2 I_3…I_n]
AOA	[0.2169 0.1932 0.1518 0.1028 0.0719]
CryStAl	[0.6342 0.5649 0.4440 0.3007 0.2102]
LA	[0.8340 0.7430 0.5840 0.3955 0.2765]

Her algoritma ortanca değeri elde ederken bir CPU zamanı harcamıştır. En az CPU süresini tüketen algoritma 13.073 saniye ile AOA'dır. Tablo 3.5'te verilen genlik değerleri kullanılarak bu değerlerin oluşturduğu ışıma diyagramları Şekil 3.4'te gösterilmiştir. Şekil 3.5'te 10 elemanlı LAA için yakınsama eğrisi gösterilmektedir. En hızlı yaklaşma eğrisine sahip algoritma LA algoritması olmuştur. AOA'nın 10 elemanlı LAA'nın iyi bir yakınsama eğrisine sahip olmadığı Şekil 3.5'te açıkça görülmektedir.



Şekil 3.4. *10 elemanlı LAA'ya ait ışıma diyagramı*



Şekil 3.5. *10 elemanlı LAA'ya ait yakınsama eğrisi*

3.3.2. 16 Elemanlı LAA Sentezine AOA, CryStAl ve LA Metotlarının Uygulanması

16 Elemanla sahip LAA'ların sentezi bu bölümde gerçekleştirilmiştir. 16 elemanlı LAA'ların karşılaştırmalı sonuçları Tablo 3.6'da verilmiştir. Tablo 3.6'da verilen değerler SLL ve HPBW'nin standart sapma, ortalama, minimum ve maksimum değerleri verileriyle sunulmuştur.

Tablo 3.6. *16 elemanlı LAA için karşılaştırmalı sonuçlar*

	SLL_{min}	$SLL_{med.}$	SLL_{max}	SLL_{std}	$HPBW_{min}$	$HPBW_{med}$	$HPBW_{max}$	$HPBW_{std}$
AOA	-40.2374	-36.7346	-28.1659	3.1819	8.7221	8.9562	9.0000	0.0852
CryStAl	-40.2423	-40.2209	-40.2027	0.0094	8.9985	8.9994	8.9999	0.0004
LA	-40.2083	-39.2288	-37.5938	0.7255	8.9962	8.9989	8.9999	0.0011

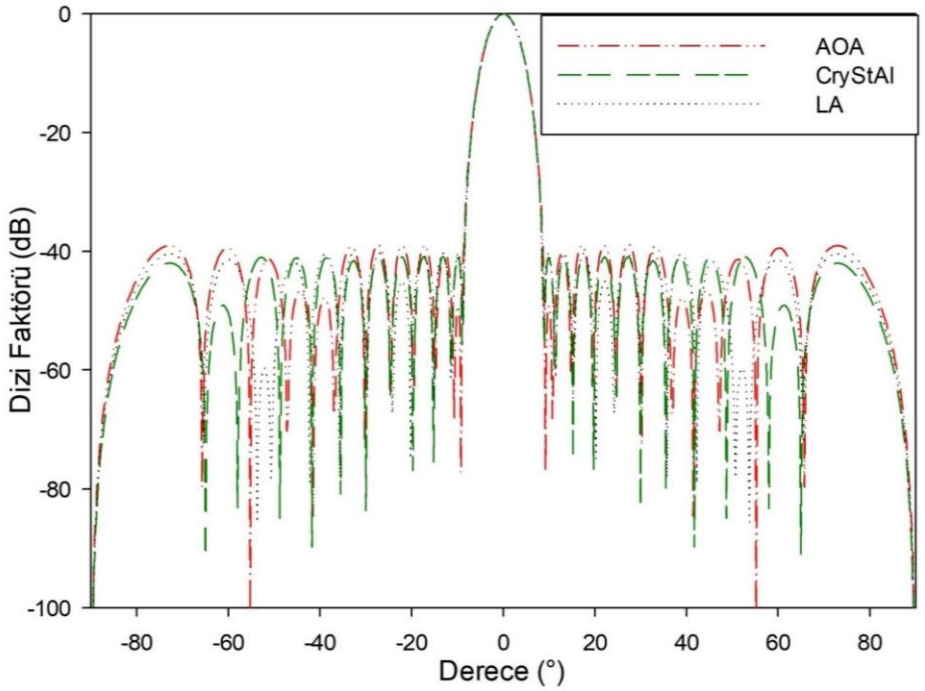
Tablo 3.6'da gösterilen değerler karşılaştırıldığında en iyi SLL değerine sahip algoritma CryStAl algoritmasıdır. Standart sapmaya göre en az saçılan algoritma CryStAl

algoritmasıdır. HPBW değerleri birbirine çok yakındır. CPU zamanı açısından en kararlı metot 0.121 saniye ile LA olmuştur. Tablo 3.7, elde edilen ışıma diyagramlarının değerlerine karşılık gelen en iyi SLL'ye ait genlik değerini göstermektedir.

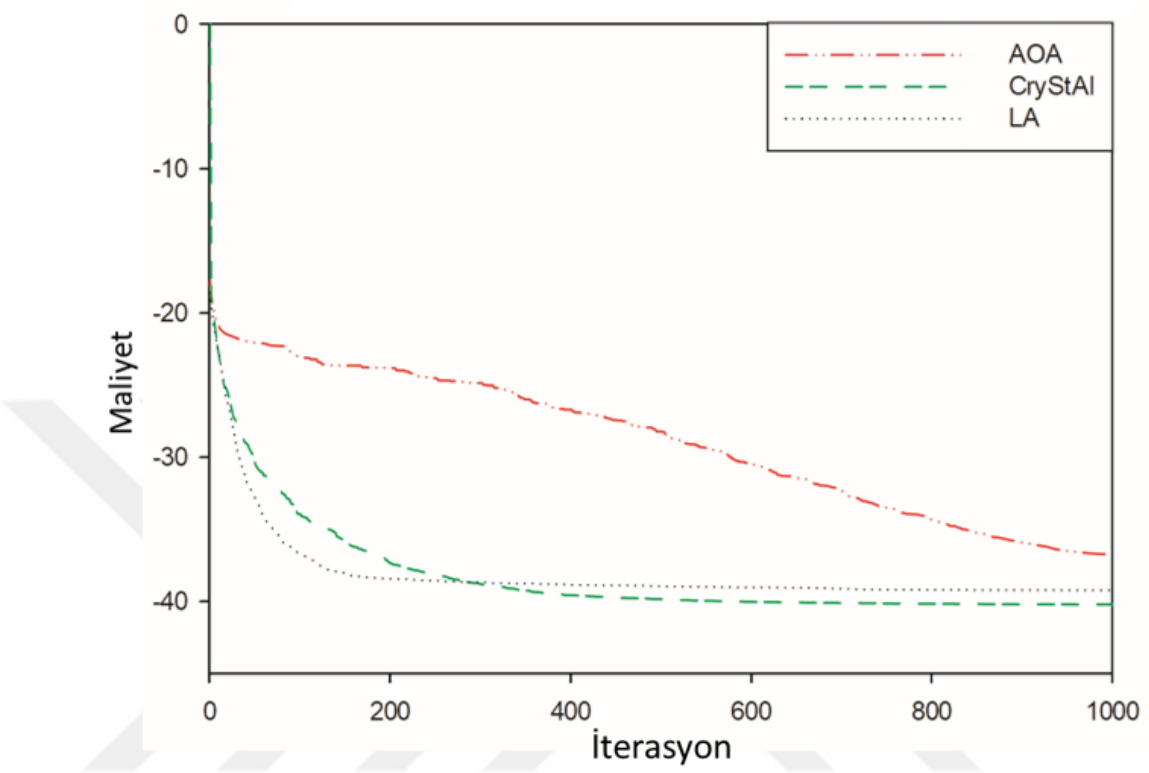
Tablo 3.7. *16 elemanlı LAA'ya ait genlik değerleri*

	Genlik Değerleri [I_1 I_2 I_3…I_n]
AOA	[0.2223 0.2078 0.1812 0.1466 0.1090 0.0732 0.0432 0.0248]
CryStAl	[0.6527 0.6102 0.5321 0.4305 0.3200 0.2151 0.1267 0.0727]
LA	[0.5826 0.5446 0.4751 0.3843 0.2856 0.1921 0.1132 0.0649]

16 elemanlı LAA'nın SLL'sini optimum olarak bastırmak için AOA metodu 16.796 saniye işlem süresi harcamıştır. CPU zaman değeriyle en hızlı algoritma AOA olmuştur. Bu genlikler kullanılarak elde edilen 2 boyutlu ışıma diyagramı Şekil 3.6'da gösterilmektedir. 16 elemanlı LAA elde edilirken algoritmaların yakınsama eğrisi Şekil 3.7'de verilmiştir.



Şekil 3.6. *16 elemanlı LAA'ya ait ışıma diyagramı*



Şekil 3.7. *16 elemanlı LAA'ya ait yakınsama eğrisi*

16 elemanlı LAA'nın yakınsama eğrisi incelendiğinde, AOA'nın yakınsama problemini çözme yeteneğinin iyi olmadığı görülmüştür. 16 elemanlı LAA'yı sentezlerken en iyi yakınsama eğrisine sahip metot CryStAl olmuştur.

3.3.3. 24 Elemanlı LAA Sentezine AOA, CryStAl ve LA Metotlarının Uygulanması

Bu bölümdeki çalışmalar için LAA'ların eleman sayısı 24 olarak belirlenmiştir. 24 elemanlı LAA için karşılaştırmalı sonuçlar Tablo 3.8'de veriler ile birlikte verilmiştir.

Tablo 3.8. *24 elemanlı LAA için karşılaştırmalı sonuçlar*

	SLL_{min}	$SLL_{med.}$	SLL_{max}	SLL_{std}	$HPBW_{min}$	$HPBW_{med.}$	$HPBW_{max}$	$HPBW_{std}$
AOA	-38.9896	-33.2765	-25.7744	3.2153	5.5013	5.8552	5.9999	0.1779
CryStAl	-40.8412	-40.0569	-36.6468	1.0083	5.9719	5.9956	6.0000	0.0070
LA	-40.2575	-38.9898	-37.2272	0.8543	5.9857	5.9973	6.0000	0.0038

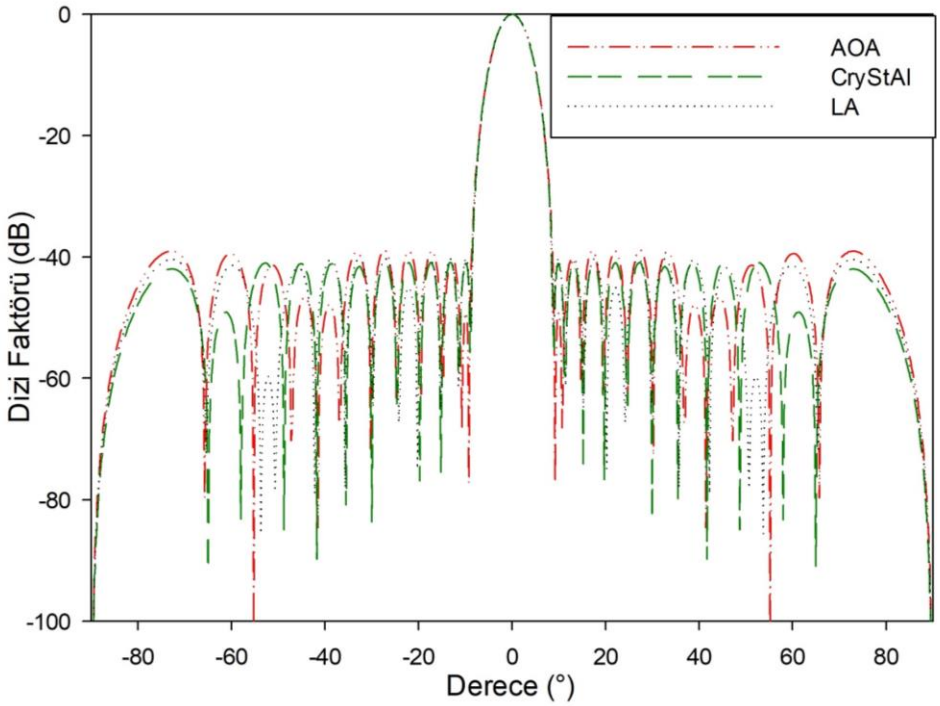
Tablo 3.8'e göre en iyi değerlere ulaşan algoritma CryStAl'dir. Ayrıca SLL'nin standart sapması açısından en iyi algoritma LA algoritmasıdır. Tüm algoritmalar için HPBW değerleri hemen hemen aynı değerlere sahipti. CPU zaman değerleri incelendiğinde en iyi

standart sapma değerine sahip algoritma 0.110 saniye ile AOA'dır. Algoritmalar tarafından en iyi SLL değerlerine ulaşılırken elde edilen genlik değerleri Tablo 3.9'da gösterilmiştir.

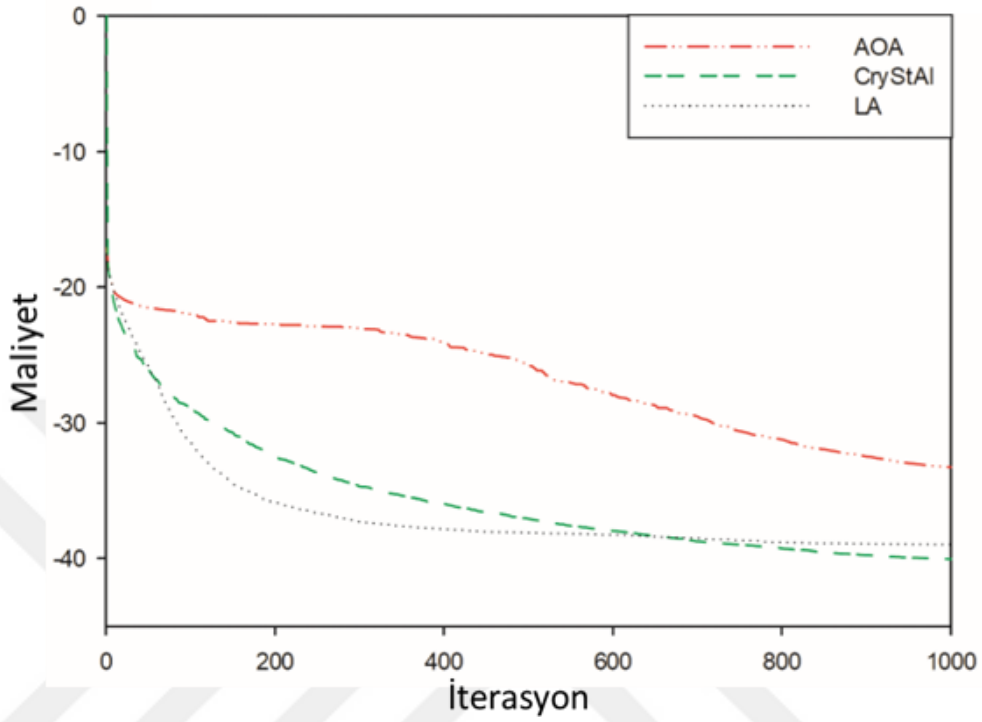
Tablo 3.9. *24 elemanlı LAA'ya ait genlik değerleri*

	Genlik Değerleri [I_1 I_2 I_3…I_n]
AOA	[0.0815 0.0787 0.0741 0.0678 0.0593 0.0499 0.0417 0.0325 0.0229 0.0181 0.0117 0.0092]
CryStAl	[0.7921 0.7728 0.7214 0.6651 0.5793 0.5014 0.4060 0.3203 0.2411 0.1652 0.1145 0.0839]
LA	[0.7990 0.7731 0.7401 0.6649 0.5888 0.5023 0.4049 0.3358 0.2385 0.1698 0.1138 0.0829]

Bu değerlerin işlenmesiyle 24 elemanlı LAA için bir ışıma diyagramı elde edilir. Ortanca CPU zamanına göre 24.120 saniye ile AOA metodu en iyi ortanca CPU değerini elde etmiştir. 24 elemanlı LAA sentezinde metotların elde ettiği ışıma diyagramı Şekil 3.8'de gösterilmektedir. Şekil 3.8'den de açıkça görüldüğü gibi en iyi ışıma diyagramını elde eden algoritma CryStAl algoritmasıdır. AOA, CryStAl ve LA metotlarının 24 elemana sahip LAA sentezi için yakınsama eğrileri Şekil 3.9'da gösterilmiştir.



Şekil 3.8. *24 elemanlı LAA'ya ait ışıma diyagramı*



Şekil 3.9. *24 elemanlı LAA'ya ait yakınsama eğrisi*

En iyi ışıma diyagramını elde eden algoritma CryStAl olmuştur. Şekil 3.9'dan görüleceği üzere optimum sonucu en hızlı bulan algoritma LA'dır.

Bu bölümde, AOA, CryStAl ve LA, 10, 16 ve 24 Elemanlı LAA'ların genlik değerlerinin optimizasyonu yapılarak optimum ışıma diyagramına sahip LAA'lar elde edilmesi amaçlanmıştır. Bütün simülasyonlarda HPBW sabit tutulurken ve SLL olabildiğince bastırılmıştır. Bu işlem sırasında HPBW değerleri 10, 16 ve 24 elemanlı LAA için sırasıyla 12.5°, 9° ve 6° olarak belirlenmiştir. Tüm LAA'ların SLL'sini en iyi bastıran algoritma CryStAl'dir. SLL'ler standart sapması açısından incelendiğinde 10, 16 ve 24 elemanlı LAA için en iyi değere ulaşan metot CryStAl'dir. Standart sapma değerleri incelendiğinde en stabil algoritma LA'dır. Tüm bölümler CPU zamanı açısından incelendiğinde en hızlı sonuca ulaşan algoritma AOA olsa bile iyi bir sonuç alma açısından yetersiz kalmıştır. Literatüre yeni kazandırılmış fizik tabanlı optimizasyon metotları, farklı anten dizisi problemlerini ve diğer hesaplamalı elektromanyetik problemlerini çözmek için gelecekteki çalışmalarda kullanılabilecek sonuçlar ortaya çıkmıştır.

3.4. CAA Sentezine AOA, AOS, CryStAl ve LA Metotlarının Uygulanması

Bu çalışmada, 8, 10 ve 12 elemanlı CAA'ların genliği ve konumu, dört farklı fizik tabanlı yeni optimizasyon metodu ile optimal olarak belirlenmiştir ve bulunan değerler istatistiksel olarak karşılaştırılmıştır. Önerilen AOA, AOS, CryStAl ve LA algoritmalarının ortak özelliği, hepsinin yeni fizik tabanlı YZ optimizasyon algoritmaları olmasıdır. Bu çalışmada kullanılan metotlar ilk kez CAA sentezine uygulanmıştır. Tüm deneylerin amacı, HPBW'i minimumda tutarken en düşük SLL'ye ulaşan algoritmayı belirlemek ve bunları karşılaştırmalı olarak sunmaktır. Karşılaştırmalı simülasyonlar, MATLAB yazılımı ve 16 GB RAM'e ve 2.8 GHz i7 işlemciye sahip bir kişisel bilgisayar kullanılarak gerçekleştirilmiştir. Tüm algoritmalar için popülasyon boyutu 100'e ve sayının maksimum iterasyon sayısı 1000 olarak belirlenmiştir.

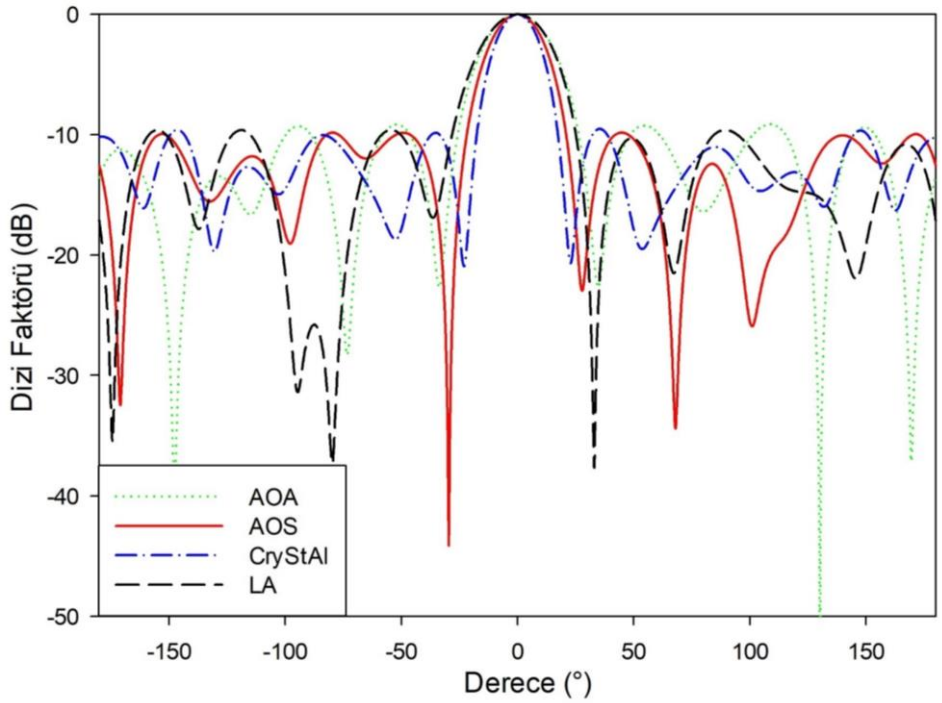
AOA, AOS, CryStAl ve LA metotları, CAA'nın optimal ışıma diyagramları tasarımı için burada uygulanmıştır. AOA, AOS, CryStAl ve LA'nın kontrol parametreleri Tablo 3.10'da gösterilmektedir. Bu parametreler karşılaştırıldığında en az dahili parametreye CryStAl metodu sahip olduğu için en kullanışlı algoritma olduğu çıkarımı yapılabilir.

Tablo 3.10. *AOA, AOS ve CryStAl metotlarının kontrol parametreleri*

Metot	**Kontrol Parametreleri**
AOA	·Nesne Sayısı= 30, C1 = 2, C2 = 6, ·C3 = 2, ·C4 = 0.5
AOS	·Fr =0.1, Ln=5
CryStAl	·Cri=30
LA	·R_c =150, ·N_p =105, ·S =1, Ref =0.2, ·M =0

3.4.1. 8 Elemanlı CAA Sentezine AOA, AOS, CryStAl ve LA Metotlarının Uygulanması

İlk durumda, optimum CAA tasarımını gerçekleştirmek için 8 elemanlı CAA'nın genlikleri ve konumları AOA, AOS, CryStAl ve LA kullanılarak belirlenmiştir. Optimizasyon işleminden sonra, 8 elemanlı CAA sentezindeki indirgenmiş SLL'lerin karşılaştırmalı ışıma diyagramları Şekil 3.10'da gösterilmektedir.



Şekil 3.10. *8 elemanlı CAA'ya ait ışıma diyagramları*

Şekil 3.10'da gösterildiği gibi, en iyi bastırılmış SLL'ye sahip ışıma diyagramı AOS metodu ile elde edilmiştir. Metasezgisel metotlar sonuca rastgele bir popülasyon oluşturarak yaklaşır. Bu nedenle, metasezgisel metotlar her çalıştırmada farklı sonuçlar elde eder. En iyi doğruluğu elde etmek için metotlar bu tez çalışmasında 30 kez çalıştırılmıştır. Daha doğru karşılaştırma için çalıştırmalardan elde edilen SLL, HPBW ve CPU Zamanı değerlerinin ortalaması ve std'si alınır. Ortalama SLL, std SLL, ortalama HPBW, std HPBW, ortalama CPU süresi ve std CPU süresi değerlerinin karşılaştırma sonuçları Tablo 3.11'de verilmiştir. Tablo 3.11'de parantez içinde verilen değerler standart sapma değerlerini belirtmektedir.

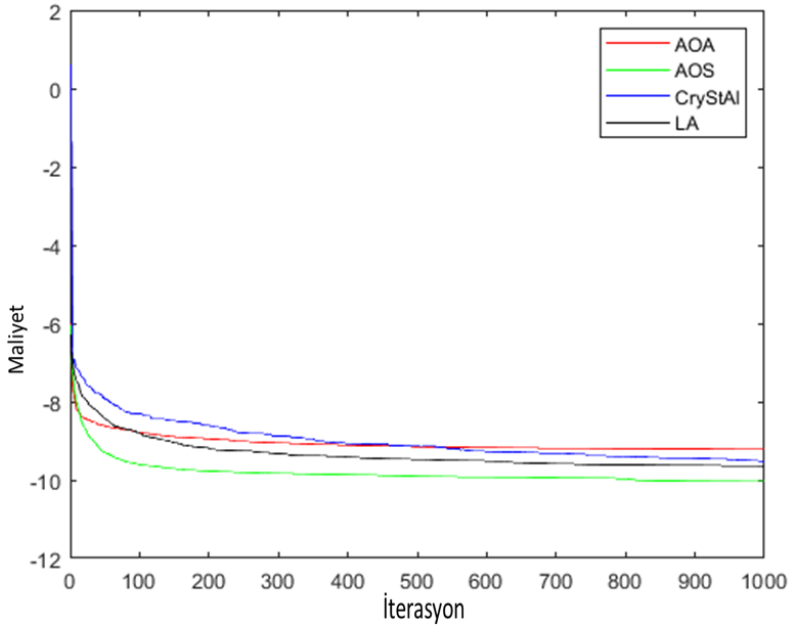
Tablo 3.11. *$SLL_{ort\&Std}$, $HPBW_{ort\&Std}$, ve CPU $Zamanı_{ort\&Std}$ 8 elemanlı CAA için 30 çalışmada elde edilen değerler*

Karşılaştırma Parametreleri	**AOA**	**AOS**	**CryStAl**	**LA**
$SLL_{ort\&(std)}$ (dB)	-9.2184 (0.5088)	**-10.0405 (0.8441)**	-9.5184 (0.3426)	-9.6412 (0.9529)
$HPBW_{ort\&(std)}$ (°)	31.3029 (0.3597)	**26.3887 (3.4967)**	27.9906 (3.0794)	26.6925 (4.2637)
CPU $Zamanı_{ort\&(std)}$ (s)	58.5184 (2.0567)	62.9165 (1.8804)	**56.0693 (1.8072)**	69.2596 (2.2566)

AOS, -10.0405 dB ile en iyi ortalama SLL'ye ulaşmıştır. En kötü SLL değeri -9.2184 dB ile AOA bulmuştur. CryStAl ve LA metotları ile ışıma diyagramlarının SLL değerleri sırasıyla -9.5184 dB ve -9.6412 dB elde edilmiştir. AOS algoritması ile elde edilen ışıma diyagramında HPBW değeri 26.3887° olarak bulunmuştur. Bu değerde HPBW açısından en iyi değerdir. 31.3029°değeriyle AOA, en kötü HPBW değerini elde etmiştir. CryStAl 27.9906° ve LA 26.6925° değerlerine ulaşmıştır. AOA 58.5184 s, AOS 62.9165 s, CryStAl 56.0693 s ve LA 69.2596 s ortalama CPU zamanı harcamış olduğu belirlendi. En hızlı algoritma CryStAl olmuştur. En iyi sonuçları bulan metot AOS olmuştur. 30 çalışma sonucunda elde edilen ortanca SLL için konum değerleri ve genlik değerleri Tablo 3.12'te gösterilmektedir. Şekil 3.11, metotların yakınsama eğrilerini göstermektedir.

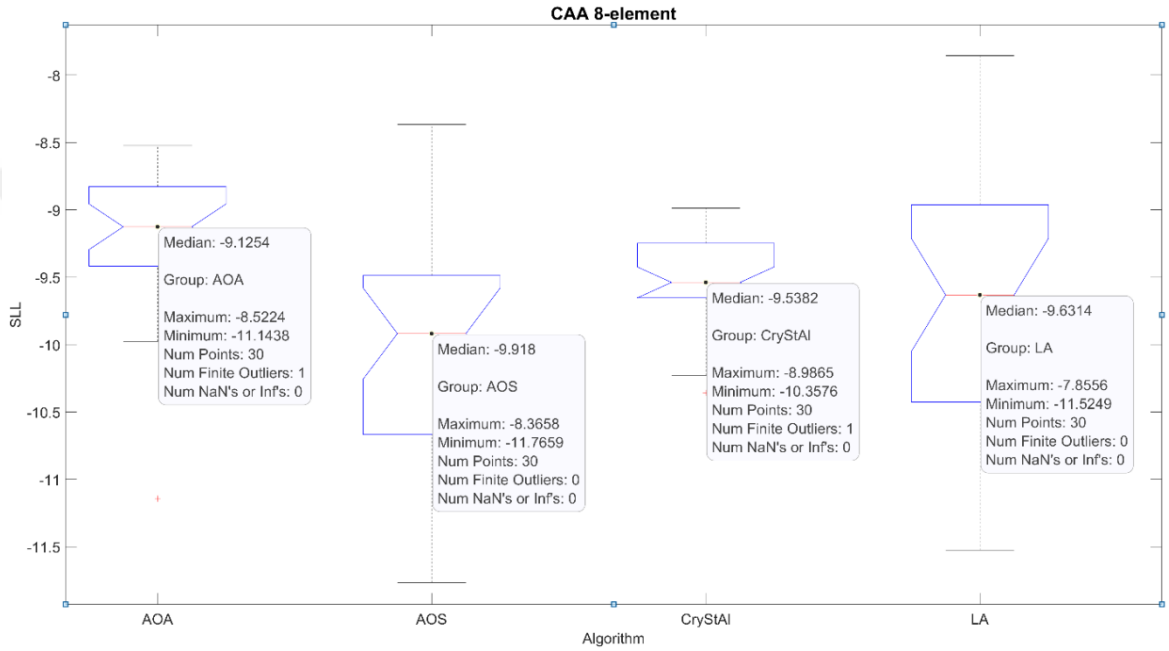
Tablo 3.12. *8 elemanlı CAA için ortalama SLL'nin konumları ve genlikleri değeri*

Metot	**$SLL_{ortanca}$ (dB)**	**Pozisyon Değerleri [d_1 d_2 d_3. …. d_N]**	**Genlik Değerleri [I_1 I_2 I_3 ……I_N]**
AOA	-9.1254	[0.5318 0.5572 0.7699 0.5730 0.5105 0.4383 0.5098 0.5422]	[0.5854 0.3305 0.6655 0.6453 0.4682 0.6028 0.5365 0.7182]
AOS	**-9.9180**	[1.0000 0.8129 0.4801 0.3122 0.5604 0.5840 0.9999 1.0000]	[0.5484 0.6743 0.2151 0.9998 0.8534 0.4116 0.7880 0.9930]
CryStAl	-9.5382	[0.6718 1.0000 1.0000 0.6607 1.0000 0.6907 0.6382 0.9915]	[0.9723 1.0000 1.0000 1.0000 1.0000 0.0330 1.0000 1.0000]
LA	-9.6314	[0.3460 0.9631 0.4860 0.1487 0.5762 0.9309 0.5926 0.4049]	[0.7284 0.7796 0.7217 0.2145 0.6794 0.6655 0.6794 0.5484]



Şekil 3.11. *8 elemanlı CAA'ya ait yakınsama eğrisi*

Şekil 3.11 AOS algoritmasının diğerlerine göre çok daha iyi yakınsama yaptığını göstermektedir. 30 bağımsız çalışma sonucunda önerilen bütün algoritmalarla elde edilen her SLL değeri kaydedilir ve ortalama ortanca değerleri bulunarak karşılaştırılma yapılır. Şekil 3.12'de kutu grafiği gösterilmiştir.

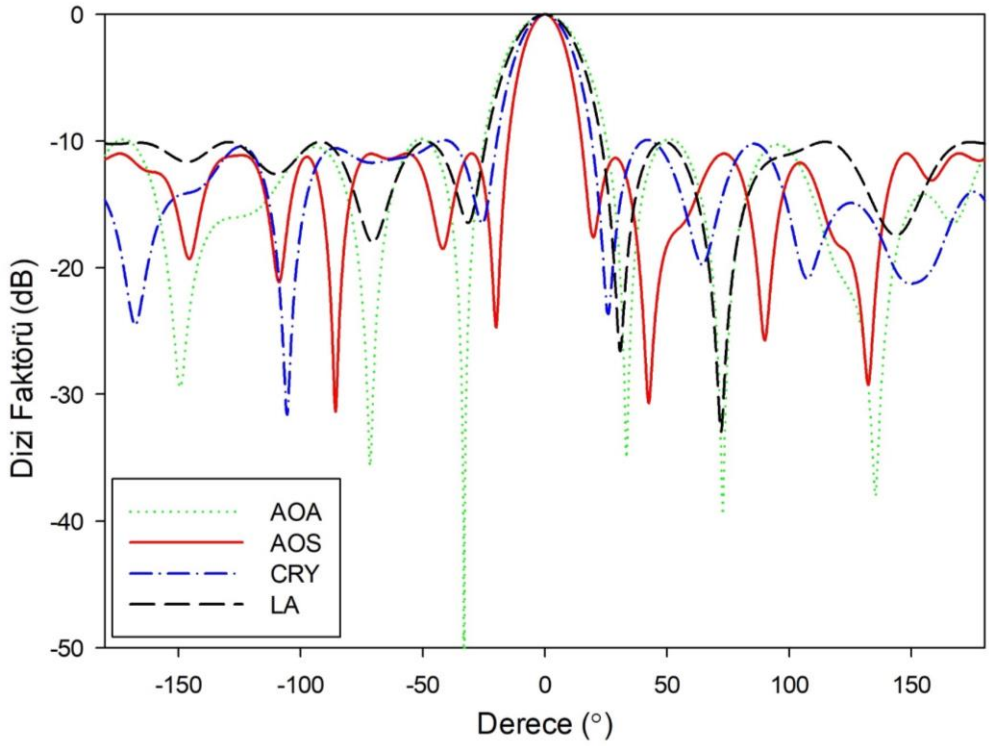


Şekil 3.12. *8 elemanlı CAA'ya ait kutu grafiği*

SLL değerlerinin kutu grafiği ile görüntülenmesi, metotların yakınsama performansının daha iyi anlaşılmasını sağlar. Şekil 3.12'de gösterildiği gibi, AOS algoritmasının maksimum SLL, minimum SLL ve ortanca SLL'si, diğer metotların kutu çizimlerine kıyasla daha iyi bir sonuç elde etmiştir.

3.4.2. 10 Elemanlı CAA Sentezine AOA, AOS, CryStAl ve LA Metotlarının Uygulanması

İkinci bölümde CAA'ya ait eleman sayısı 10 olarak belirlenmiştir. En optimum ışıma diyagramını en doğru şekilde belirlemek için tüm yöntemler 30 kez bağımsız olarak çalıştırılmıştır. AOA, AOS ve LA ile optimum şekilde belirlenen genlik ve konum değerlerine karşılık gelen ışıma diyagramları Şekil 3.13'te karşılaştırmalı olarak verilmiştir.



Şekil 3.13. *10 elemanlı CAA'ya ait ışıma diyagramı*

AOA, AOS ve LA ile elde edilen SLL ve HPBW değerleri Tablo 3.13'de karşılaştırmalı olarak verilmiştir. Şekil 3.13'te açıkça görüleceği üzere en iyi SLL değerlerine AOS metodu ile ulaşılmıştır.

Tablo 3.13. *$SLL_{ort\&Std}$, $HPBW_{ort\&Std}$, ve CPU $Zaman\imath_{ort\&Std}$ 10 elemanlı CAA için 30 çalışmada elde edilen değerler*

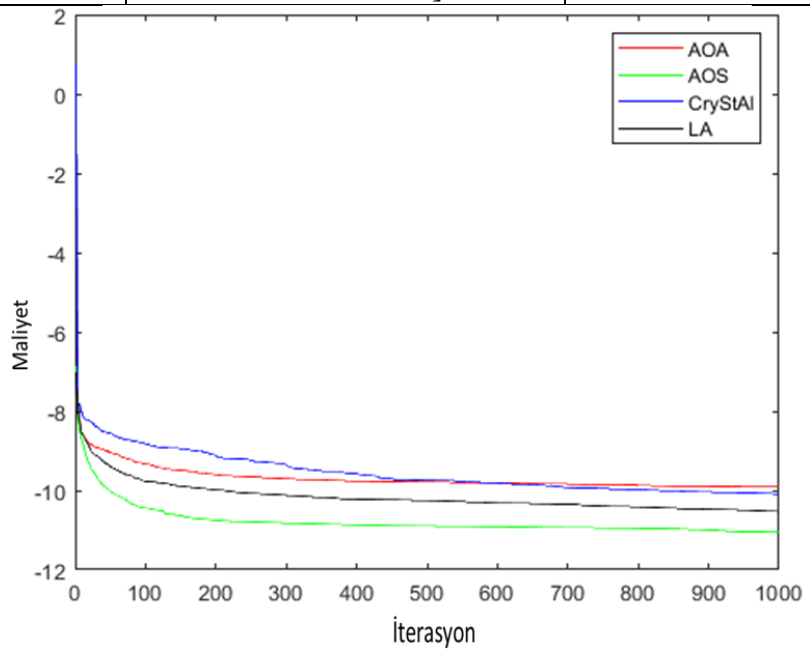
Karşılaştırma Parametreleri	**AOA**	**AOS**	**CryStAl**	**LA**
$SLL_{ort\&(std)}$ (dB)	-9.9091 (0.4004)	**-11.0466** (0.9809)	-10.0982 (0.6288)	-10.5388 (1.2327)
$HPBW_{ort\&(std)}$ (°)	28.5705 (2.1704)	22.9668 (4.0482)	**21.3735** (3.9555)	22.6892 (4.7332)
CPU $Zaman\imath_{ort\&(std)}$ (s)	68.1794 (0.4006)	71.5274 (0.2563)	**64.1678** (0.3152)	82.4133 (0.3222)

Tablo 3.13 incelendiğinde en kötü ortalama SLL değeri AOA metodu ile -9.9091 dB bulunurken, AOS -11.0466 dB ile en iyi ortalama SLL'yi bulmuştur. Ayrıca CryStAl -10.0982 dB ve LA -10.5388 dB SLL değerlerini elde etmiştir. HPBW açısından bakıldığında CryStAl 21.3735° ile en dar HPBW değerine ulaşmıştır. En kötü HPBW değerini bulan yöntem 28.5705° değeri ile AOA metodu olmuştur. AOS 22.9668° ve LA

22.6892° HPBW değerlerini elde etmiştir. Bu değerleri elde ederken ortalama harcanan CPU zamanının değerleri 64.1678 s ile CryStAl, 68.1794 s ile AOA, 71.5274 s ile AOS ve 82.4133 s ile LA'dır. Bu sonuçlar değerlendirildiğinde, CryStAl metodu diğer algoritmalardan daha hızlı olmasına rağmen, AOS diğer metotlardan çok daha iyi SLL değerlerini elde etmiştir. Şekil 3.14, maksimum iterasyon sayısına göre yakınsama eğrilerini göstermektedir. Tablo 3.14'te 10 elemanlı CAA için pozisyon ve genlik değerleri verilmiştir.

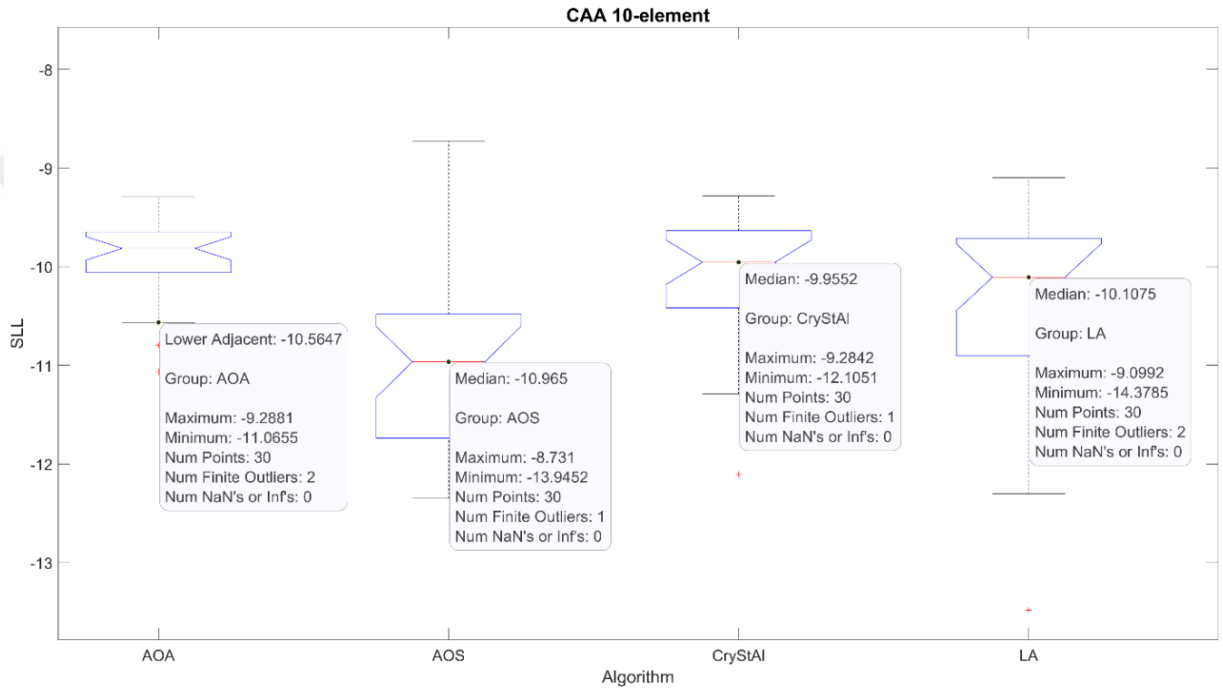
Tablo 3.14. *10 elemanlı CAA için Ortanca SLL'nin konumlarının ve genliklerinin değerleri*

Metot	**$SLL_{ortanca}$ (dB)**	**Konum Değerleri [d_1 d_2 d_3 …. d_N]**	**Genlik Değerleri [I_1 I_2 I_3 ……I_N]**
AOA	-9.8146	[0.4325 0.4815 0.4929 0.4830 0.4704 0.4736 0.4664 0.4485 0.4160 0.4158]	[0.6432 0.3705 0.4204 0.5649 0.6247 0.5472 0.4431 0.2954 0.6078 0.5292]
AOS	**-10.9650**	[0.9968 0.8483 0.5450 1.0000 0.6127 0.6188 0.9981 0.5750 0.9089 0.7961]	[0.9973 0.6688 0.7265 0.8038 0.9898 0.5779 0.1704 0.7101 0.5755 0.9975]
CryStAl	-9.9552	[0.4169 1.0000 0.3405 1.0000 0.3236 1.0000 0.2960 0.2204 0.8339 0.4872]	[0.7032 0.8449 0.9177 0.8070 0.6036 0.6115 0.4116 0.1828 0.3927 0.9989]
LA	-10.1075	[0.2436 0.2512 0.5627 0.6910 0.1923 0.8342 0.5824 0.9445 0.9697 0.3198]	[0.8062 0.2006 0.9460 0.7117 0.1972 0.9508 0.4592 0.4783 0.9944 0.7712]



Şekil 3.14. *10 elemanlı CAA'ya ait yakınsama eğrisi*

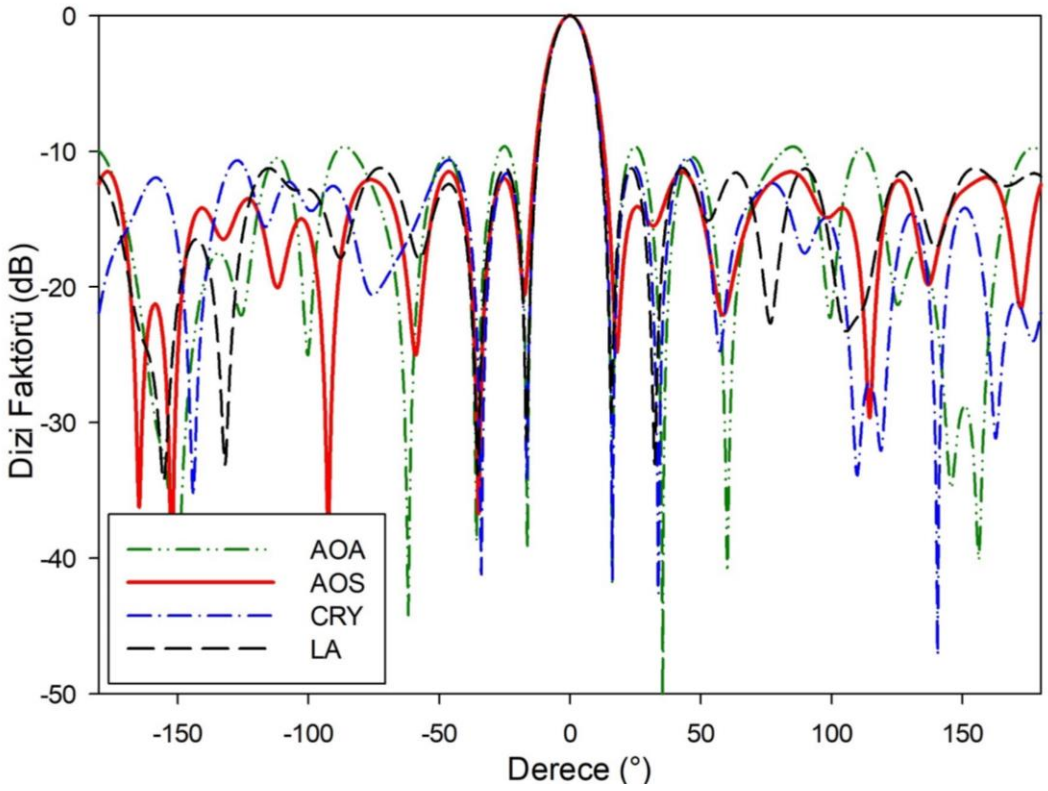
Şekil 3.14 incelendiğinde, en hızlı yakınsayan ve en iyi sonucu bulan metot AOS olmuştur. LA, diğer algoritmalardan daha hızlı yakınsamasına rağmen, AOS kadar iyi sonuçlar elde edememiştir. Şekil 3.15'te 10 elemanlı CAA'ya ait kutu grafiği verilmiştir.



Şekil 3.15. *10 elemanlı CAA'ya ait kutu grafiği*

3.4.3. 12 Elemanlı CAA Sentezine AOA, AOS, CryStAl ve LA Metotlarının Uygulanması

CAA'lar için 3. örnekte eleman sayısı 12 olarak belirlenmiştir. SLL, HPBW, CPU süresi ile birlikte elde edilen genlik ve konum değerleri 30 rastgele çalışma ile AOA, AOS, CryStAl ve LA metotları kullanılarak belirlenmiştir. Elde edilen sonuçlar karşılaştırmalı olarak tablo ve şekiller kullanılarak sunulmuştur. Önerilen 4 yeni metasezgisel optimizasyon metotları ile elde edilen genlik ve konum değerleri neticesinde ortaya çıkan ışıma diyagramları karşılaştırmalı olarak Şekil 3.16'da verilmiştir.



Şekil 3.16. *12 elemanlı CAA'ya ait ışıma diyagramları*

Tablo 3.15. $SLL_{ort\&Std}$, $HPBW_{ort\&Std}$, ve CPU $Zaman_{ort\&Std}$ 12 elemanlı CAA için 30 çalışmada elde edilen değerler

Karşılaştırma Parametreleri	**AOA**	**AOS**	**CryStAl**	**LA**
$SLL_{ort\&(std)}$ (dB)	-9.4452 (0.7111)	**-11.4868 (0.7481)**	-10.5408 (0.4019)	-11.0105 (1.0501)
$HPBW_{ort\&(std)}$ (°)	**14.8135 (0.3729)**	15.1584 (0.6869)	14.9825 (0.6161)	15.2141 (0.8947)
CPU $Zaman_{ort\&(std)}$ (s)	80.5184 (0.3053)	81.2683 (0.2588)	**73.5558 (0.2618)**	96.2438 (0.4745)

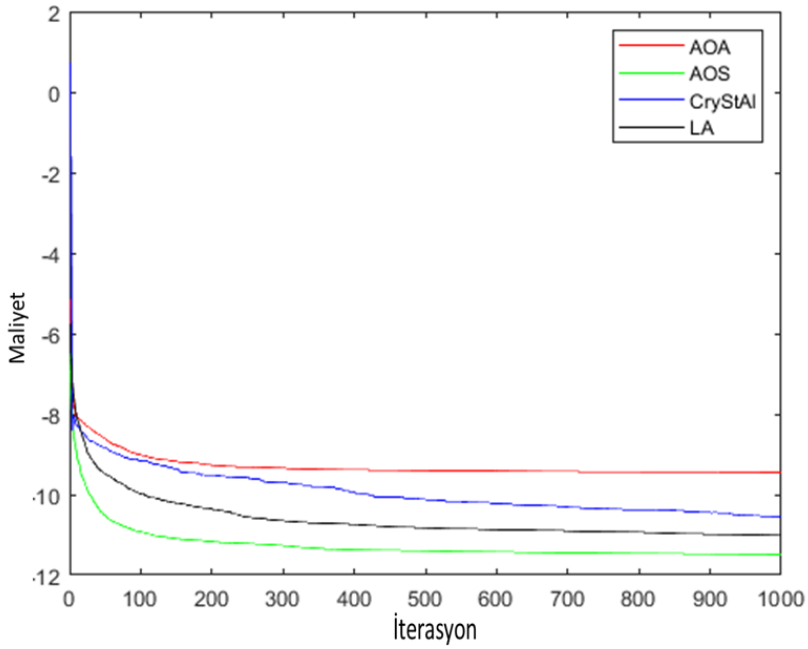
Tablo 3.15'te algoritmalar ortalama SLL, ortalama HPBW ve ortalama CPU zamanı değerleri açısından karşılaştırılmıştır. SLL sonuçlarına bakıldığında AOS algoritması -11.4868 dB değeri ile en iyi sonucu bulmuştur. AOA algoritması -9.4452 dB ile en kötü sonucu elde ederken CryStAl ve LA algoritması sırasıyla -10.5408 dB ve -11.0105 dB değerlerine ulaşmıştır. Tablo 3.15'teki HPBW değerleri incelendiğinde en iyi değer AOA

olmasına rağmen tüm metotlar birbirine yakın sonuçlar elde etmiştir. SLL değerlerini bulmak için harcadıkları CPU zamanı değerleri karşılaştırıldığında, CryStAl en hızlı ve LA en yavaş metot olarak çıkmıştır. Her bir algoritmanın ortalama değerine karşılık gelen konum ve genlik değerleri de Tablo 3.16'da verilmiştir.

Tablo 3.16. *12 elemanlı CAA için Ortanca SLL'nin konumlarının ve genliklerinin değerleri*

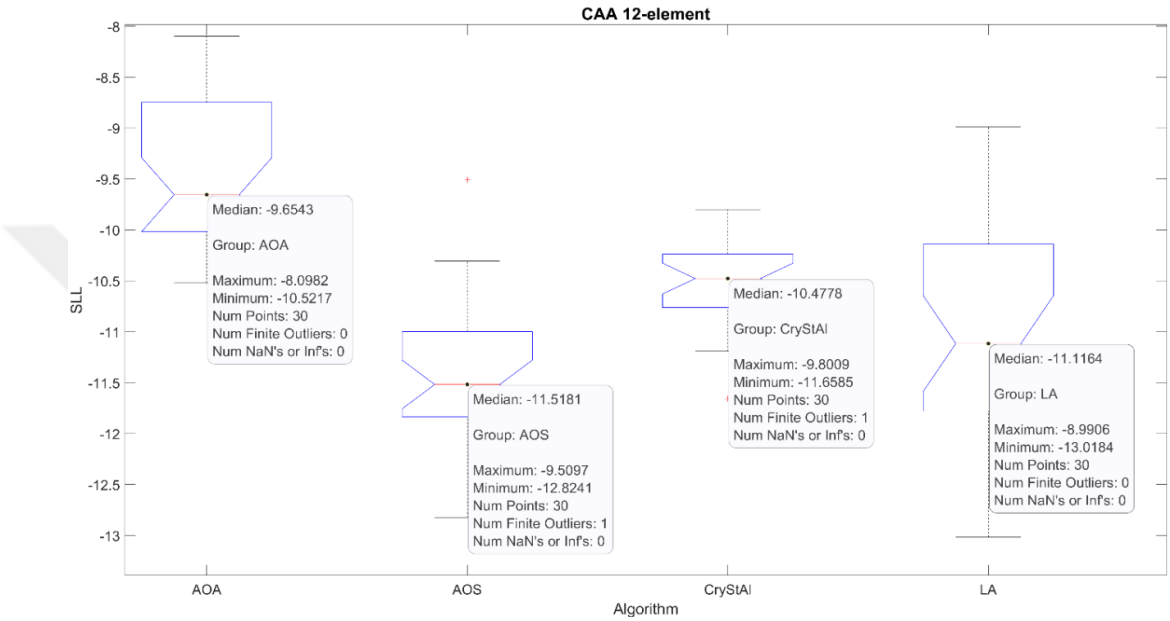
Metot	$SLL_{ortanca}$ (dB)	Pozisyon Değerleri [d_1, d_2, d_3, …. d_N]	Genlik Değerleri [I_1, I_2, I_3, ……I_N]
AOA	-9.6543	[0.6625 0.7967 0.7326 0.7301 0.8439 0.7052 0.6880 0.8373 0.7534 0.6926 0.8278 0.6644]	[0.6479 0.5056 0.5850 0.5942 0.6067 0.6279 0.5619 0.5894 0.5948 0.5489 0.6411 0.7156]
AOS	**-11.5181**	[0.7724 0.4518 0.9976 0.9981 0.9998 0.5045 0.5432 0.9998 0.8170 0.5145 0.9995 0.6495]	[0.8793 0.3588 0.7189 0.5103 0.6530 0.9998 0.8026 0.6810 0.7798 0.7972 0.9995 0.9994]
CryStAl	-10.4778	[0.7008 0.9371 0.7143 0.6301 0.8860 0.7047 0.4443 0.6727 0.9663 0.9719 0.9555 0.7352]	[0.7467 0.6611 0.7815 0.3173 0.9266 0.7705 0.7097 0.6159 0.7937 0.8991 0.8001 0.8776]
LA	-11.1164	[0.5851 0.9762 0.8477 0.8550 0.9997 0.5653 0.8774 0.8678 0.9960 0.2390 0.9996 0.6270]	[0.9911 0.5893 0.8456 0.8644 0.9605 0.8444 0.5902 0.3027 0.7645 0.8481 0.8675 0.7743]

30 çalışma sonucunda ortalama SLL değerleri karşılaştırıldığında, AOS -11.5181 dB ile en iyi ortalama SLL değerine ulaşmıştır. Şekil 3.17'de algoritmaların 12 elemanlı CAA'yı elde ederken ki yakınsama eğrilerini göstermektedir.



Şekil 3.17. *12 elemanlı CAA'ya ait yakınsama eğrisi*

Bu grafik incelendiğinde en kararlı algoritmanın CryStAl olduğu görülmektedir. AOS algoritması 12 elemanlı CAA için en iyi yakınsama eğrisini elde etmiştir. Şekil 3.18'de 12 elemana sahip CAA için kutu grafiği verilmiştir.



Şekil 3.18. *12 elemanlı CAA'ya ait kutu grafiği*

3.5. CCAA Sentezine AOS Metodunun Uygulanması

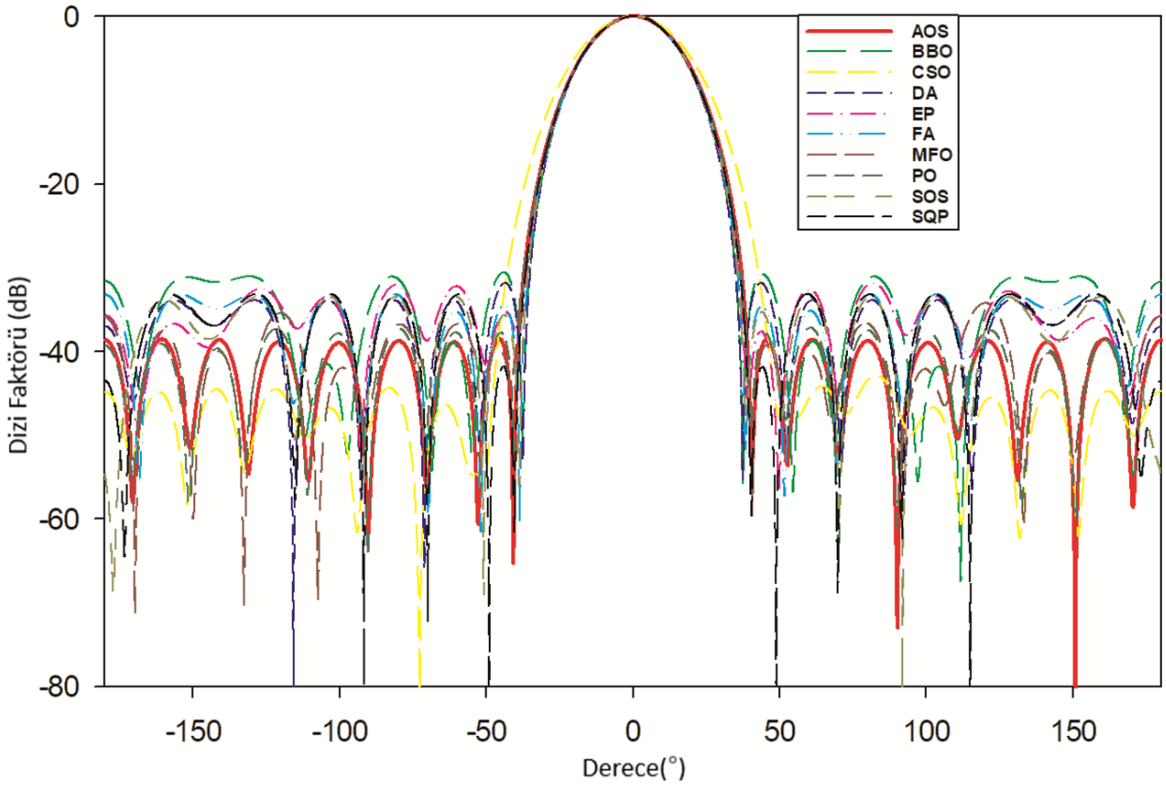
AOS metodu, en iyi SLL'yi elde etmek için üç düzgün olmayan halka ile CCAA'nın elemanlar arası boşluk ve genlik değerlerini belirlemek için kullanılır. AOS optimizasyon metodu, arama alanında en iyi sonucu almak için 2.8 GHz i7 CPU ve 32 GB RAM'e sahip bir bilgisayarda çalıştırıldı. Tüm simülasyonlar için, AOS'un popülasyon boyutu ve maksimum iterasyon sayısı sırasıyla 100 ve 1000'dir. AOS metodu, SLL'yi etkin bir şekilde bastırarak üç halkalı CCAA'ların optimal sentezi için kullanılmıştır. Simülasyonlar, simetrik üç halkalı CCAA'ların sentezi ile sınırlandırılmıştır. Literatürde yaygın olarak $n_1=4$, $n_2=6$, $n_3=8$ ve $n_1=8$, $n_2=10$, $n_3=12$ elemanlı CCAA'lar kullanıldığı için tüm simülasyon çalışmalarında referans olarak bunlar incelenmiştir. Tüm sentezlerde iki durum vardır: merkezi elemanlı olan ve olmayan üç halkalı CCAA'lar. Çalışmalarda, elemanlar arası boşluk, AOS optimizasyon metodu kullanılarak CCAA'ların her halkasında optimize edilmiştir. Bu işlem sonucunda merkezi elemanlı $n_1=4$, $n_2=6$, $n_3=8$ elemanlı CCAA'nın elemanlar arası boşluk değerleri içten dışa sırasıyla 0,5493 λ, 0,604 λ ve 0,7487 λ bulunmuştur. Merkezi elemanı olmayan $n_1=4$, $n_2=6$, $n_3=8$ elemanlı CCAA'nın elemanlar

arası boşluk değerleri içten dışa sırasıyla 0,485 λ, 0,565λ ve 0,765 λ olarak belirlenmiştir. Merkezi elemanlı $n_1=8$, $n_2=10$, $n_3=12$ elemanlı CCAA'lar için elemanlar arası boşluk değerleri 0,53 λ, 0,58 λ ve 0,73λ olarak elde edilmiştir. Son olarak, $n_1=8$, $n_2=10$, $n_3=12$ elemanlı CCAA'lar için elde edilen elemanlar arası boşluk değerleri 0,54 λ, 0,6048 λ ve 0,74 λ'dır. Literatürde mevcut çalışmalar olan BBO, CSO, DA, EP, FA, MFO, SOS, SQP ve PO ile elde edilen sonuçlar önerilen AOS metodu ile karşılaştırılmıştır ve AOS metodunun sonuçları diğer metotlardan oldukça başarılı olduğu tespit edilmiştir. AOS optimizasyon metodu ve diğer optimizasyon metotları ile elde edilen $n_1=4$, $n_2=6$, $n_3=8$ ve $n_1=8$, $n_2=10$, $n_3=12$ elemanlı merkezi elemanı olan ve olmayan CCAA'ların genlik değerleri Tablo 3.17-3.20'de verilmiştir.

Tablo 3.17. *Merkez elemanı olmayan $n_1=4$, $n_2=6$, $n_3=8$ elemanlı CCAA'ya ait elde edilen optimal genlik ve SLL değerleri*

Metot $\phi_{null2}=52^o$	**SLL (dB)**	**Genlikler**		
		$[I_{1,2,3,4}]$	$[I_{5,6,7,8,9,10}]$	$[I_{11,12,13,14,15,16,17,18}]$
AOS	**−38.50 dB**	[0.9645 0.7612 0.9587 0.7697]	[0.8210 0.8153 0.9984 0.8051 0.8076 0.9763]	[0.5802 0.9455 0.5766 0.2900 0.5745 0.9339 0.5790 0.2997]
BBO (Dib ve Sharaqa, 2015)	−30.60 dB	[0.3477 0.7950 0.3348 0.7975]	[0.5046 0.5078 1.0000 0.5171 0.5073 1.0000]	[0.5173 0.8248 0.5079 0.2251 0.5263 0.7986 0.5332 0.2260]
CSO (Ram ve ark., 2015)	−33.39 dB	[0.0703 0.5675 0.0824 0.5491]	[0.3442 0.3516 0.8304 0.3463 0.3440 0.8596]	[0.4199 0.5768 0.4140 0.1919 0.4040 0.5928 0.4149 0.2097]
DA (Babayigit, 2018)	−33.91 dB	[0.1390 0.673 0.1512 0.6732]	[0.4334 0.4284 0.9978 0.4210 0.4264 0.9999]	[0.4975 0.7080 0.4998 0.2296 0.4975 0.7399 0.4949 0.2290]
EP (Mandal ve ark., 2010)	−31.84 dB	[0.0496 0.3242 0.0283 0.3416]	[0.2114 0.1923 0.4901 0.1876 0.1994 0.5321]	[0.2555 0.3527 0.2450 0.1229 0.2294 0.3449 0.2400 0.1204]
FA (Sharaqa ve Dib, 2014)	−33.20 dB	[0.1410 0.6770 0.1215 0.7025]	[0.4349 0.4084 0.9999 0.4076 0.4305 0.9999]	[0.4789 0.7366 0.4831 0.2542 0.4790 0.7172 0.4730 0.2352]
MFO (Babayigit, 2018)	−36.84 dB	[0.8696 0.7969 0.9075 0.7528]	[0.6441 0.6758 0.8161 0.6599 0.6771 0.7870]	[0.5037 0.8125 0.4958 0.2756 0.4930 0.8334 0.4989 0.2813]
PO (Durmus ve Kurban, 2021)	−37.30 dB	[0.4972 0.5878 0.4909 0.6153]	[0.5801 0.5586 0.8983 0.5649 0.5804 0.8895]	[0.4653 0.7313 0.4620 0.2672 0.4604 0.7365 0.4655 0.2433]]
SOS (Dib, 2018)	−33.47 dB	[0.0433 0.6712 0.0748 0.6489]	[0.3915 0.4036 1.0000 0.4141 0.4037 0.9999]	[0.4866 0.6634 0.5005 0.2437 0.4804 0.6897 0.4659 0.2401]]
SQP (Dib ve Sharaqa, 2015)	−33.16 dB	[0.0310 0.6669 0.0203 0.6701]	[0.3911 0.3886 1.0000 0.3831 0.3852 0.9998]	[0.4610 0.6812 0.4627 0.2542 0.4630 0.6642 0.4614 0.2501]

CCAA tasarımının amacı, aynı frekanstaki SLL'leri bastırarak en verimli CCAA'ları elde etmektir. Tablo 3.17'de verilen sonuçlar incelendiğinde önerilen AOS optimizasyon metodunun diğer metotlara göre çok daha iyi SLL değerleri üreterek diğer tüm metotlardan üstün olduğu ortaya çıkmıştır. Elde edilen değerlerin 2 boyutlu ışıma diyagramları Şekil 3.19'da gösterilmektedir. Şekil 3.19'da açıkça görüldüğü gibi AOS algoritması ile elde edilen ışıma diyagramının SLL değeri diğer metotlara göre daha iyidir. AOS diğer metotları -35.5 dB ile geride bırakmıştır. Tablolarda gösterilen genlik katsayısı değerleri kullanılarak elde edilen ışıma diyagramları Şekil 3.19-23'te gösterilmektedir.

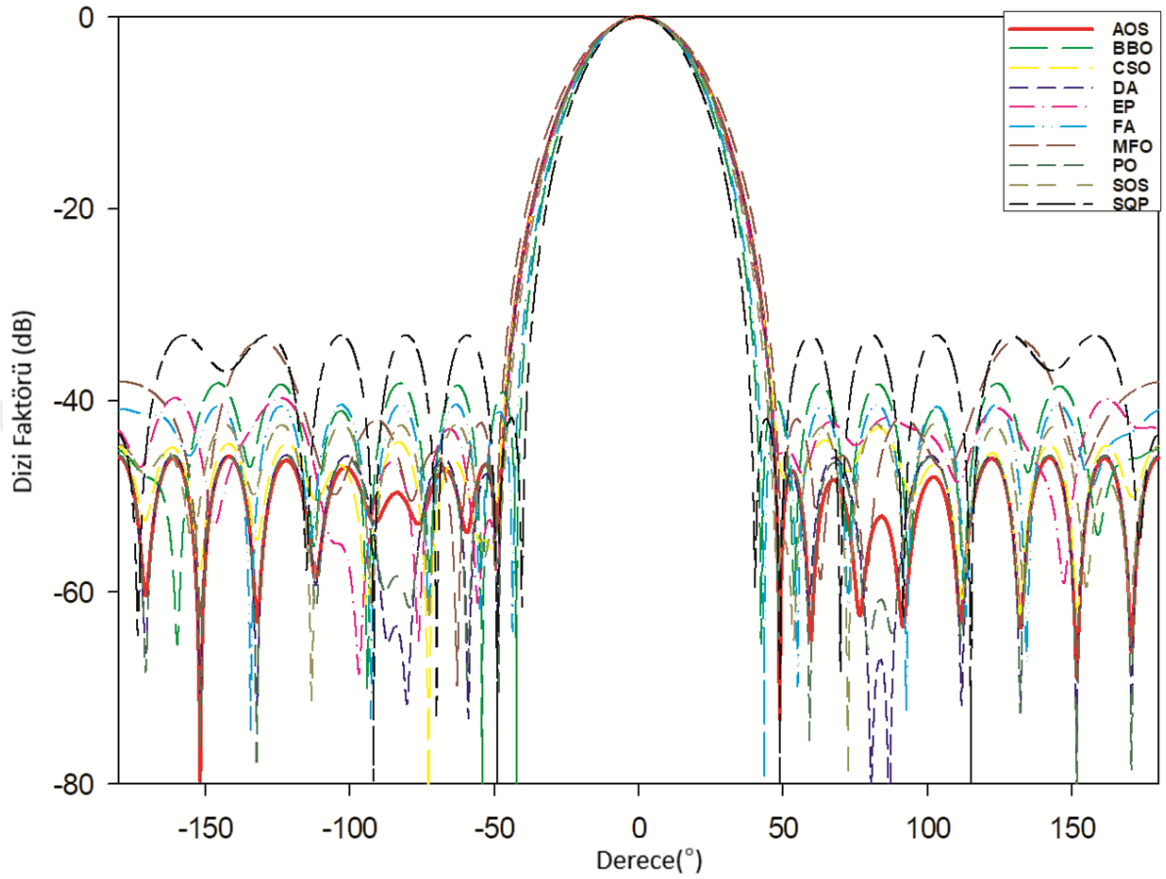


Şekil 3.19. *Merkez elemanı olmayan n_1=4, n_2=6, n_3=8 elemanlı CCAA'ya ait ışıma diyagramı*

Şekil 3.19'dan görüldüğü üzere AOS metodu SLL'yi diğer algoritmalardan daha iyi bastırmıştır. Merkezi elemanlı n_1=4, n_2=6, n_3=8 elemanlı CCAA'lar için elde edilen SLL ve optimal genlik değerleri Tablo 3.18'de verilmiştir. Bu değerler kullanılarak elde edilen ışıma diyagramı Şekil 3.20'de gösterilmektedir.

Tablo 3.18. *Merkez elemanlı n_1=4, n_2=6, n_3=8 elemanlı CCAA'ya ait elde edilen optimal genlik ve SLL değerleri*

Metot ϕ_{null2}**=46º**	**SLL (dB)**	**Genlikler**			
		[$I_{1,2,3,4}$]	**[$I_{5,6,7,8,9,10}$]**	**[$I_{11,12,13,14,15,16,17,18}$]**	**I_{merkez}**
AOS	**-45.98dB**	[0.9808 0.9123 0.9938 0.9101]	[0.6801 0.6741 0.1100 0.6801 0.6875 0.1097]	[0.2702 0.3800 0.2701 0.0611 0.2701 0.3900 0.2706 0.0600]	[0.6827]
BBO (Dib ve Sharaqa, 2015)	−38.20dB	[0.7560 0.9983 0.7491 1.0000]	[0.7319 0.7228 0.7698 0.7307 0.7288 0.7394]	[0.5440 0.6968 0.5683 0.1953 0.5568 0.6848 0.5297 0.1801]	[0.4669]
CSO (Ram ve ark., 2015)	−42.87dB	[0.4719 0.4897 0.4805 0.4830]	[0.3430 0.3528 0.1096 0.3579 0.3483 0.1139]	[0.1531 0.2177 0.1539 0.0421 0.1586 0.2257 0.1576 0.0464]	[0.3320]
DA (Babayigit, 2018)	−45.73dB	[0.9905 0.9262 0.9961 0.9292]	[0.6859 0.6826 0.1041 0.6876 0.6890 0.1064]	[0.2734 0.3897 0.2741 0.0649 0.2731 0.3918 0.2717 0.0637]	[0.6878]
EP(Mandal ve ark., 2010)	−39.73dB	[0.5477 0.5530 0.5890 0.5502]	[0.3830 0.3972 0.0999 0.4152 0.4051 0.0976]	[0.1730 0.2290 0.1734 0.0401 0.1750 0.2755 0.1717 0.0417]	[0.3770]
FA (Sharaqa ve Dib, 2014)	−40.43dB	[0.8029 0.9508 0.8087 0.9943]	[0.7228 0.7056 0.6025 0.7020 0.7262 0.5727]	[0.4732 0.6145 0.4837 0.1627 0.4748 0.6159 0.4648 0.1516]	[0.5142]
MFO (Babayigit, 2018)	−42.93dB	[0.3878 0.5541 0.3707 0.5288]	[0.8094 0.8214 0.7204 0.8183 0.8089 0.7100]	[0.3894 0.3864 0.4018 0.0041 0.4270 0.3832 0.4191 0.0019]	[0.7288]
PO (Durmus ve Kurban, 2021)	−45.75dB	[0.9861 0.9196 0.9896 0.9202]	[0.6828 0.6789 0.1046 0.6816 0.6845 0.1070]	[0.2722 0.3881 0.2726 0.0635 0.2725 0.3902 0.2716 0.0638]	[0.6802]
SOS (Dib, 2018)	−42.40dB	[0.8722 0.9998 0.9026 1.0000]	[0.7095 0.7121 0.4203 0.7249 0.7276 0.4055]	[0.3928 0.5217 0.3991 0.1313 0.3939 0.5531 0.3885 0.1287]	[0.6111]
SQP (Dib ve Sharaqa, 2015)	−42.40dB	[0.9818 0.9129 0.9831 0.9111]	[0.6798 0.6751 0.1048 0.6756 0.6800 0.1076]	[0.2709 0.3865 0.2711 0.0621 0.2718 0.3885 0.2714 0.0638]	[0.6724]

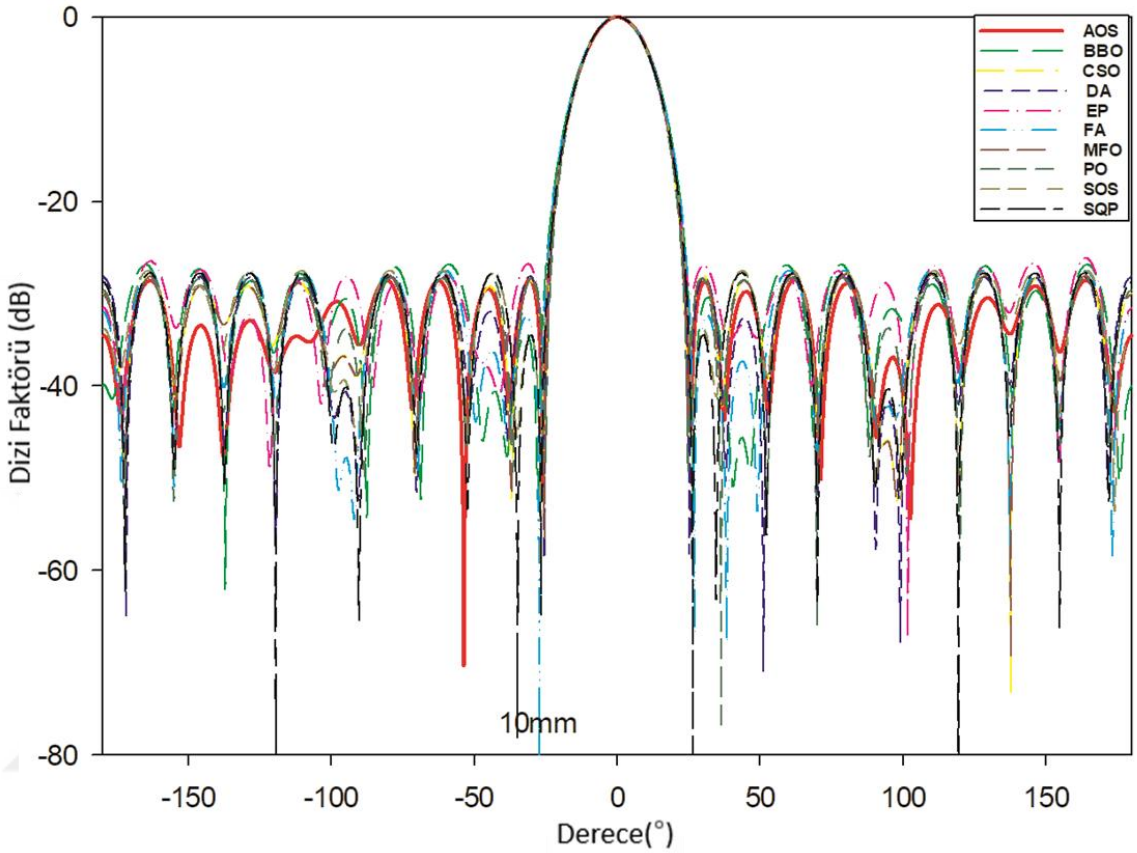


Şekil 3.20. *Merkez elemanlı n_1=4, n_2=6, n_3=8 elemanlı CCAA'ya ait ışıma diyagramı*

Önerilen algoritmanın performansını daha iyi gösterebilmek için n_1=8, n_2= 10, n_3=12 elemanlı merkezsiz CCAA'ya da uygulanmıştır. Merkezsiz n_1=8, n_2= 10, n_3=12 elemanlı CCAA için optimize edilmiş genlik değerleri Tablo 3.19'da verilmiştir. n_1=8, n_2=10, n_3=12 elemanlı merkezsiz CCAA için elde edilen ışıma diyagramları Şekil 3.21'de gösterilmiştir. Metasezgisel yöntemler karşılaştırıldığında, AOS -28.52dB ile diğer algoritmalardan daha iyi değerlere ulaşmıştır.

Tablo 3.19. *Merkez elemanı olmayan n_1=8, n_2=10, n_3=12 Elemanlı CCAA'ya ait elde edilen optimal genlik ve SLL değerleri*

Metot ϕ_{null2}**=28º**	**SLL (dB)**	**Genlikler**		
		$[I_{1,2,3,4,5,6,7,8}]$	$[I_{1,2,3,4,5,6,7,8,9,10}]$	$[I_{1,2,3,4,5,6,7,8,9,10,11,12}]$
AOS	**−28.52dB**	[0.7161 0.2965 0.6025 0.5299 0.5563 0.3771 0.6564 0.8384]	[0.7376 0.0946 0.1004 0.4642 0.5626 0.4963 0.0847 0.0624 0.8908 0.7158]	[0.4337 0.3880 0.9021 0.2886 0.3572 0.4325 0.3382 0.2708 0.9315 0.4496 0.3524 0.3766]
BBO (Dib ve Sharaqa, 2015)	−26.83dB	[0.6957 0.3821 0.7366 0.9184 0.8001 0.2855 0.7184 0.9670]	[0.6631 0.1436 0.2588 0.6330 0.6310 0.6184 0.2104 0.1214 0.7737 0.5634]	[0.4075 0.3080 1.0000 0.2797 0.3986 0.5384 0.3191 0.3241 0.9751 0.3741 0.3804 0.6059]
CSO (Ram ve ark., 2015)	−27.86dB	[0.6739 0.3432 0.6771 0.8039 0.7780 0.2618 0.7627 0.7271]	[0.4706 0.1789 0.1448 0.5577 0.4837 0.6122 0.0510 0.0895 0.6417 0.4643]	[0.3747 0.2329 0.9529 0.2425 0.4582 0.4318 0.3656 0.3845 0.9701 0.4396 0.2581 0.4045]
DA (Babayigit, 2018)	−28.15dB	[0.7333 0.3098 0.7275 0.8021 0.7444 0.3016 0.7500 0.7824]	[0.5110 0.1976 0.1589 0.5479 0.4847 0.6092 0.1166 0.1535 0.5699 0.4838]	[0.3963 0.2488 0.9637 0.2897 0.3742 0.4556 0.3372 0.3510 0.9950 0.3099 0.3569 0.4499]
EP (Mandal ve ark., 2010)	−26.12dB	[0.2886 0.1891 0.3336 0.5458 0.3895 0.1000 0.2866 0.2242]	[0.1378 0.1036 0.1000 0.4048 0.2686 0.3090 0.1000 0.1000 0.1696 0.1595]	[0.1183 0.1144 0.4708 0.1685 0.2090 0.2566 0.2200 0.1000 0.4229 0.1273 0.1020 0.2419]
FA (Sharaqa ve Dib, 2014)	−27.49dB	[0.7716 0.3013 0.7299 0.8924 0.7641 0.3044 0.7999 0.9354]	[0.5686 0.2124 0.1958 0.5901 0.5647 0.6322 0.1498 0.1660 0.6379 0.5444]	[0.4125 0.2457 0.9673 0.2516 0.3827 0.4854 0.3444 0.3209 0.9734 0.3290 0.3651 0.5044]
MFO (Babayigit, 2018)	−27.92dB	[0.6751 0.3437 0.6788 0.8048 0.7787 0.2633 0.7641 0.7287]	[0.4717 0.1793 0.1452 0.5590 0.4846 0.6128 0.0518 0.0904 0.6420 0.4660]	[0.3755 0.2337 0.9535 0.2428 0.4588 0.4323 0.3670 0.3858 0.9707 0.4398 0.2589 0.4046]
PO (Durmus ve Kurban, 2021)	−28.25dB	[0.7058 0.2764 0.6833 0.6505 0.6824 0.2526 0.7078 0.8076]	[0.6547 0.0861 0.1174 0.5257 0.4812 0.4693 0.1630 0.1286 0.6045 0.5284]	[0.3718 0.3901 0.9997 0.3249 0.3444 0.4798 0.3803 0.2737 0.9998 0.3444 0.4008 0.4909]
SOS (Dib, 2018)	−27.51dB	[0.7272 0.2027 0.7132 0.7764 0.6652 0.3052 0.6754 0.8210]	[0.5748 0.1225 0.0582 0.6055 0.5303 0.6102 0.0831 0.1484 0.6280 0.5918]	[0.4459 0.3050 0.9260 0.3879 0.3089 0.4550 0.3126 0.3476 0.9983 0.2882 0.4373 0.4990]
SQP (Dib ve Sharaqa, 2015)	−27.74dB	[0.6802 0.2215 0.7085 0.7993 0.7182 0.2220 0.6914 0.7489]	[0.5471 0.1255 0.1374 0.5816 0.5741 0.5904 0.1227 0.1123 0.5560 0.5608]	[0.3797 0.2825 0.9968 0.3036 0.3680 0.4846 0.3685 0.3295 0.9991 0.3096 0.3806 0.5030]



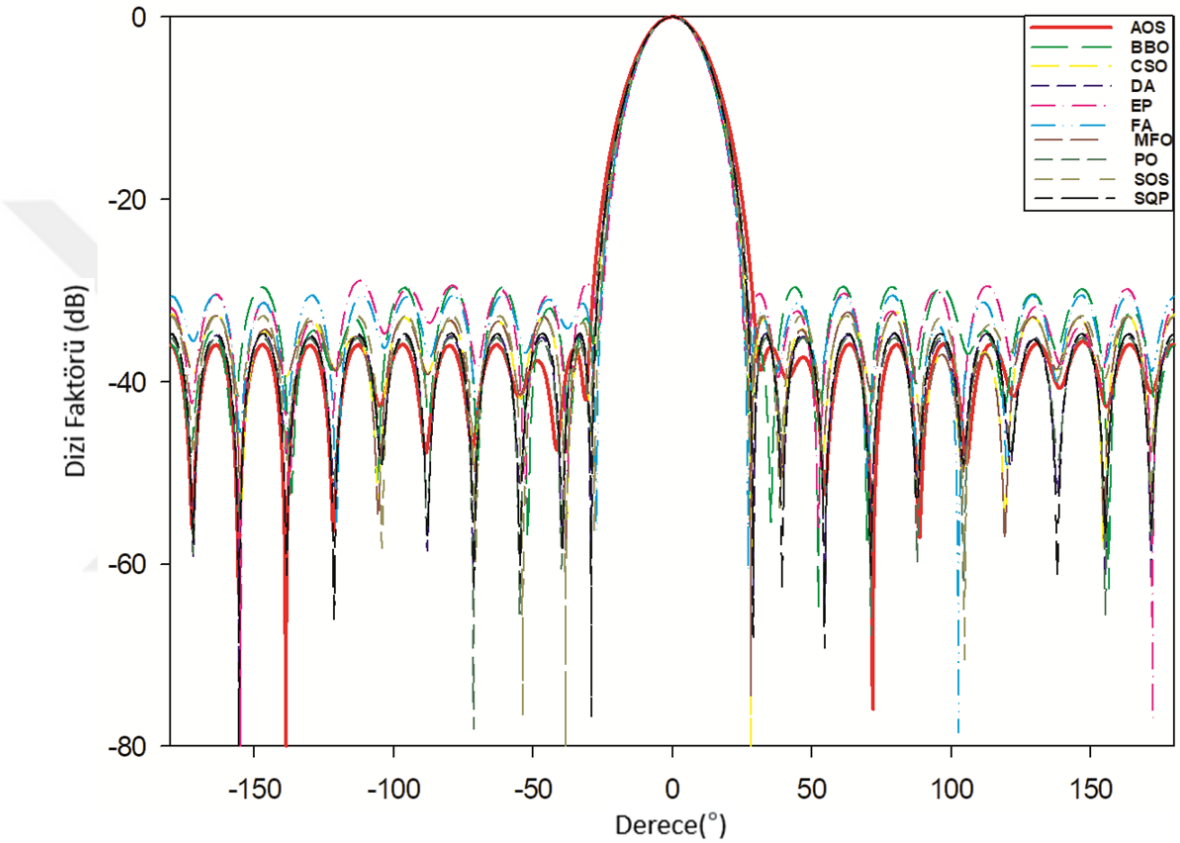
Şekil 3.21. *Merkez elemanı olmayan n_1=8, n_2=10, n_3=12 elemanlı CCAA'ya ait ışıma diyagramı*

Tablo 3.20, merkezi elemanlı n_1=8, n_2=10, n_3=12 elemanlı CCAA'lar için elde edilen optimum genlik ve SLL değerlerini göstermektedir.

Tablo 3.20. *Merkez elemanlı $n_1=8$, $n_2=10$, $n_3=12$ elemanlı CCAA'ya ait elde edilen optimal genlik ve SLL değerleri*

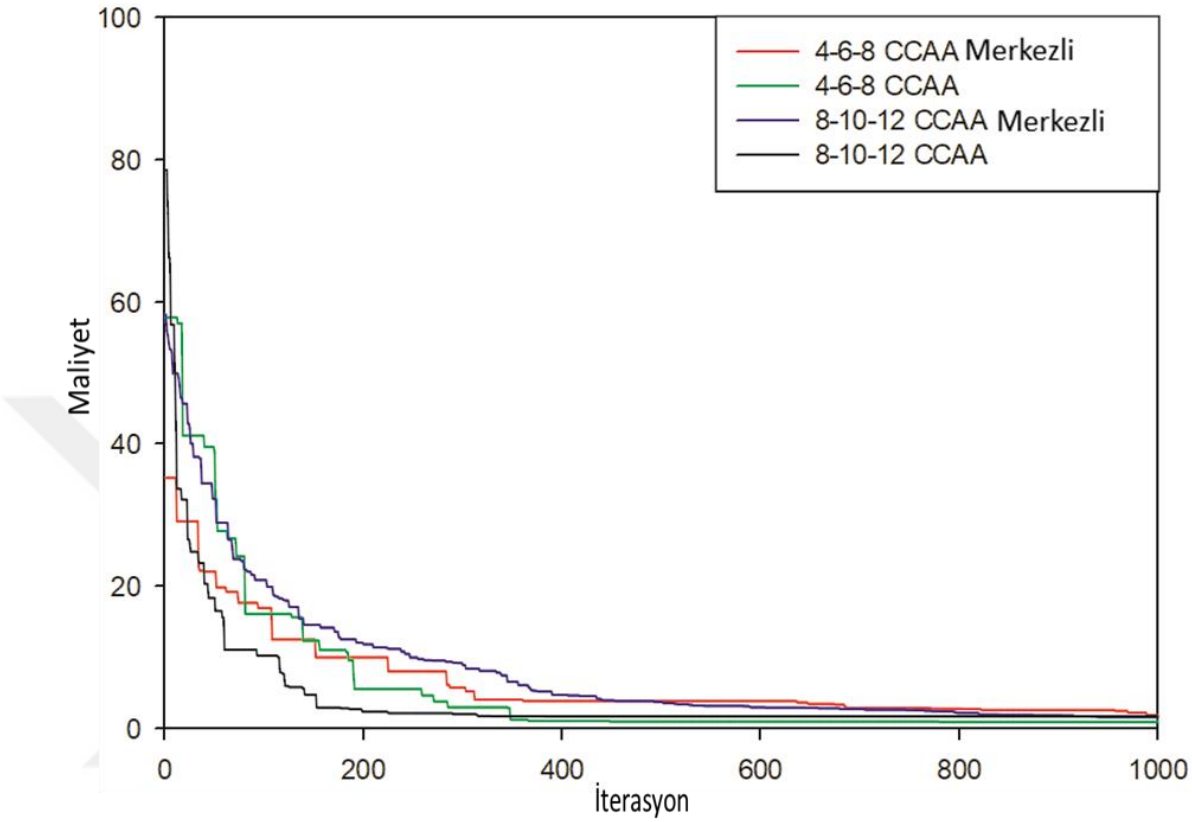
Metot $\phi_{null2}=28^o$	**SLL değerleri (dB)**	**Genlikler**			
		$[I_{1,2,3,4,5,6,7,8}]$	$[I_{1,2,3,4,5,6,7,8,9,10}]$	$[I_{1,2,3,4,5,6,7,8,9,10,11,12}]$	I_{merkez}
AOS	**−35.6dB**	[0.2470 0.9327 0.2674 0.7545 0.3727 0.8349 0.3385 0.7888]	[0.5729 0.1463 0.1571 0.5810 0.3806 0.6509 0.1258 0.1073 0.6199 0.4110]	[0.3104 0.1419 0.5621 0.2132 0.2666 0.1828 0.2158 0.3091 0.5634 0.2379 0.2616 0.2496]	[0.9945]
BBO (Dib ve Sharaqa, 2015)	−29.61dB	[0.6648 0.8292 0.6373 1.0000 0.5945 0.8948 0.5061 1.0000]	[0.8204 0.0529 0.1067 0.8818 0.5429 0.7970 0.0846 0.0000 0.8129 0.5600]	[0.4304 0.4370 0.9898 0.5142 0.3695 0.4323 0.3746 0.4201 0.9710 0.4175 0.4277 0.4745]	[0.7208]
CSO (Ram ve ark., 2015)	−32.37dB	[0.3522 0.6790 0.3002 0.5535 0.2998 0.7069 0.3734 0.7658]	[0.5586 0.0094 0.0249 0.3997 0.1955 0.3889 0.0243 0.0274 0.6129 0.2374]	[0.2335 0.2526 0.4757 0.1489 0.2343 0.1763 0.2177 0.1651 0.4915 0.3097 0.2273 0.1414]	[0.6082]
DA (Babayigit, 2018)	−34.79dB	[0.3814 1.0000 0.3930 0.9165 0.3910 0.9966 0.3776 0.8480]	[0.6043 0.0204 0.0390 0.6607 0.2502 0.6550 0.0290 0.0130 0.6100 0.2102]	[0.2706 0.2668 0.5700 0.2866 0.2978 0.1800 0.3090 0.2790 0.5585 0.2667 0.2831 0.1850]	[0.9340]
EP(Mandal ve ark., 2010)	−28.92dB	[0.2989 0.4102 0.3979 0.7325 0.3989 0.3813 0.2785 0.2628]	[0.2300 0.0187 0.0464 0.5620 0.2875 0.5240 0.0855 0.0166 0.1763 0.1283]	[0.1225 0.1932 0.5081 0.2903 0.2285 0.2227 0.2858 0.2278 0.4828 0.0957 0.1756 0.2082]	[0.2750]
FA (Sharaqa ve Dib, 2014)	−30.55dB	[0.5631 0.7591 0.4567 0.6972 0.5239 0.6171 0.6279 0.9967]	[0.7551 0.0321 0.1155 0.5299 0.4027 0.4977 0.1285 0.0531 0.7114 0.3579]	[0.3824 0.3453 0.8170 0.2995 0.2844 0.3716 0.2866 0.2705 0.7644 0.3284 0.3990 0.3466]	[0.5199]
MFO (Babayigit, 2018)	−32.88dB	[0.3534 0.6802 0.3009 0.5543 0.3003 0.7081 0.3751 0.7669]	[0.5594 0.0105 0.0256 0.4012 0.1961 0.3901 0.0252 0.0288 0.6138 0.2390]	[0.2343 0.2545 0.4765 0.1498 0.2354 0.1770 0.2182 0.1663 0.4926 0.3111 0.2288 0.1420]	[0.6090]
PO (Durmus ve Kurban, 2021)	−35.20dB	[0.3867 0.9999 0.3811 0.8751 0.3406 1.0000 0.3451 0.8653]	[0.6299 0.0305 0.0255 0.6256 0.2249 0.6182 0.0098 0.0132 0.6176 0.2168]	[0.2823 0.2679 0.5452 0.2657 0.2850 0.1753 0.2788 0.2656 0.5255 0.2669 0.2743 0.1669]	[0.9481]
SOS (Dib, 2018)	−32.77dB	[0.5128 0.9999 0.5347 0.9699 0.5246 0.9993 0.5313 0.9948]	[0.7888 0.0454 0.0978 0.6285 0.3094 0.7120 0.0442 0.0008 0.8066 0.3684]	[0.3440 0.3818 0.7780 0.2612 0.3821 0.3124 0.3594 0.3429 0.7892 0.4150 0.3454 0.2726]	[0.8628]
SQP (Dib ve Sharaqa, 2015)	−34.70dB	[0.3841 0.9919 0.4112 0.9154 0.4028 1.0000 0.3797 0.8538]	[0.6265 0.0170 0.0313 0.6672 0.2532 0.6587 0.0328 0.0192 0.6063 0.2222]	[0.2820 0.2794 0.5826 0.3003 0.2980 0.1906 0.3060 0.2900 0.5799 0.2619 0.2928 0.1924]	[0.9263]

Şekil 3.22, bir merkezi elemana sahip $n_1=8$, $n_2=10$, $n_3=12$ elemanlı CCAA'lar için Tablo 3.19'da verilen veriler işlenerek oluşturulan ışıma diyagramını göstermektedir.



Şekil 3.22. *Merkez elemanlı $n_1=8$, $n_2=10$, $n_3=12$ elemanlı CCAA'ya ait ışıma diyagramı*

Tablo 3.19 ve Şekil 3.22'deki veriler, önerilen AOS yöntemimizin bu çalışmada -35,6 dB'lik bir SLL değeri ile en iyi sonuçları elde ederek elde ettiğini göstermektedir. AOS metodu ile gerçekleştirilen optimizasyon işleminde maksimum iterasyon sayısı 1000 olarak belirlenmiştir. Şekil 3.23, AOS metodu kullanılarak üç halkalı CCAA sentezindeki yakınsama eğrisini göstermektedir.



Şekil 3.23. *Üç halkalı CCAA'ya ait yakınsama eğrisi*

Şekil 3.23'deki yakınsama eğrisi incelendiğinde merkezi elemanı olmayan $n_1=8$, $n_2=10$, $n_3=12$ elemanlı CCAA'ların daha iyi yakınsama eğrisine sahip olduğu görülmektedir.

Kısaca bu bölümde, Şekil 3.19-3.22'deki ışıma diyagramları ve Tablo 3.17-3.20'deki sonuçlar analiz edildiğinde, önerilen AOS metodu BBO, CSO, DA, EP, FA, MFO, SOS, SQP ve PO'dan daha iyi sonuçları elde ederek üstünlüğünü kanıtlamıştır. AOS metodu 18 elemanlı merkezi elemana sahip CCAA için -38.50 dB değer elde ederken merkezsiz 18 elemanlı CCAA için -45.98 dB'lik bir değer elde etmiştir. Bu sonuçların yanı sıra merkezi elemana sahip 30 elemanlı CCAA için AOS metodu -35.60 dB değerine ulaşmıştır. AOS metodu aynı zamanda merkezi elemana sahip olmayan 30 elemanlı CCAA için -28.52 dB'lik bir değer elde ederek diğer algoritmalardan daha iyi sonuca ulaşmıştır. Tablo 3.17-3.20 ve Şekil 3.19-3.22'ye bakıldığında, merkez elemanlı CCAA'lar, merkez elemanı olmayanlara göre daha iyi sonuçlar elde etmiştir. Bu nedenle, merkez elemanlı CAA'ların kullanılması, merkez elemanlı CAA'lardan daha verimli olacağı gözlemlenmiştir.

DÖRDÜNCÜ BÖLÜM
SONUÇLAR VE TARTIŞMALAR

4. Sonuçlar

Bu tez çalışmasında Yapay Arı Koloni Algoritması (ABC), Arşimet Optimizasyon Algoritması (AOA), Atomik Yörünge Arama Metodu (AOS), Kristal Yapı Algoritması (CryStAl), Kertenkele Sürü Algoritması (CSA), Bal Porsuğu Algoritması (HBA), Lichtenberg Algoritması (LA), Parçacık Sürü Optimizasyon Metodu (PSO) kullanılarak doğrusal anten dizisi (LAA), dairesel anten dizisi (CAA) ve eşmerkezli dairesel anten dizisinin sentezi gerçekleştirilmiştir PSO, ABC, HBA ve CSA metotları kullanılarak 10, 16, 24 ve 32 elemana sahip LAA'ların sentezi sadece konumları optimize edilerek yapılmıştır. Bu değerlendirmeyi yaparken 30 kere algoritmalar çalıştırılmış ve her bir çalıştırma 1000 iterasyon ile sınırlandırılmıştır. Ayrıca algoritmaların performansları karşılaştırılırken SLL ve CPU Zamanlarının maksimum, minimum ve ortalama değerleri kullanılmıştır. Bölüm 3.1'deki sonuçlar incelendiğinde literatüre yeni önerilen HBA ve CSA metotları literatürde yaygın olarak kullanılan PSO ve ABC metotlarına göre daha iyi SLL değerleri elde ederek performanslarının daha üstün olduğu ortaya çıkmıştır. Bölüm 3.2'deki çalışmada 10, 16 ve 24 elemana sahip LAA'ların sentezi AOS kullanılarak yapılmıştır. Birbirinden farklı olan tüm LAA'ların sentezinde sadece genlik değerleri optimize edilmiştir. Elde edilen LAA'ların ışıma diyagramları incelendiğinde AOS başarılı bir şekilde LAA'ların SLL'lerini bastırarak birbirinden farklı elemana sahip LAA'ların sentezini yapmıştır. Bu çalışmadaki sonuçlara bakıldığında AOS optimizasyon metodunun LAA sentezinde başarılı olduğu ve gelecek çalışmalarda farklı geometrilere sahip anten dizilerinin sentezlerinde de kullanılabileceği öne sürülmüştür. Bölüm 3.3'de yapılan çalışmada 3 farklı eleman sayısına sahip LAA'lar AOA, CryStAl ve LA metotları kullanılarak sentezlenmiştir. LAA'ların

eleman sayıları 10, 16 ve 24 olarak belirlenmiştir. Farklı elemanlara sahip LAA'ların optimum ışıma diyagramlarının elde edilmesi için sadece genlik değerleri optimize edilmiştir. Algoritmaların performansını daha iyi karşılaştırmak için tüm simülasyonlar 30 kere rastgele olarak çalıştırılmıştır. karşılaştırma kriteri olarak LAA'ların SLL ve HPBW değerlerinin ortalama, standart sapma, minimum ve maksimum değerleri belirlenmiştir. Bu kriterlere göre değerlendirildiğinde LAA sentezinde en etkin algoritma CryStAl olarak belirlenmiştir. Bölüm 3.4'de yapılan çalışmada, *8, 10 ve 12* elemanlı CAA'ların genliği ve konumu, literatüre yeni kazandırılmış fizik tabanlı yeni optimizasyon metotları olan AOA, AOS, CryStAl ve LA ile optimal olarak belirlenmiştir ve bulunan değerler istatistiksel olarak karşılaştırılmıştır. Karşılaştırmalar sonucunda birbirinden farklı tüm CAA'lar için en iyi SLL değerine AOS algoritması ile ulaşılmıştır. Bölüm 3.5'deki son çalışmada AOS optimizasyon metodu ile merkezi elemana sahip olan ve sahip olmayan $n_1=4$, $n_2=6$, $n_3=8$ ve $n_1=8$, $n_2=10$, $n_3=12$ elemanlı CCAA'ların sentezi yapılmıştır. AOS'un elde ettiği değerler literatürde daha önce BBO, CSO, DA, EP, FA, MFO, SOS, SQP ve PO metotları ile elde edilmiş olan 9 farklı çalışmanın değerleri ile karşılaştırılmıştır. Bu karşılaştırmalar sonucunda AOS metodu ile elde edilen değerler diğer tüm metotlardan daha iyi sonuca ulaşarak üstünlüğünü kanıtlamıştır. Bu tez çalışmasında sürü metotlarından HBA ve CSA fizik tabanlı YZ optimizasyon algoritmalarından AOA, AOS, CryStAl ve LA literatürde ilk kez anten dizilerinin sentezinde kullanılmıştır. Bu algoritmaların yapıları değiştirilerek performansları artırılabilir. Ayrıca önerilen YZ optimizasyon algoritmalarının, farklı anten türleri ve değişik birçok uygulamalı elektromanyetik problemlere uygulanabileceği çıkarımı yapılabilir.

TEZ ÇALIŞMASINDAN ÜRETİLEN ESERLER

Tez çalışmasında 1 adet uluslararası konferans ve 1 adet TR dizin makale yayınlanmak üzere kabul edildi. Ayrıca 1 adet TR dizin makale ve 2 adet SCI yayınlanmak üzere dergilere yollandı ve hakemler tarafından incelenme süreci devam etmektedir. Bu çalışmalar aşağıda listelenmiştir.

1-) Durmuş A., Yıldırım Z., & Kurban R. (2021) *Atomik Yörünge Arama Algoritması Kullanarak Doğrusal Anten Dizi Sentezi*, ICONSAD-2021.

2-) Durmuş A., Yıldırım Z. (2021) *Side Lobe Reduction of Linear Antenna Array by Position-Only Control Using Novel Metaheuristic Optimization Methods,* ESOGÜ Müh Mim Fak Derg. [Kabul Edildi]

3-) Durmuş A., Yıldırım Z. (2022) *Synthesis of Linear Antenna Arrays with Physics Based AOA, CryStAl and LA Algorithms,* Mühendislik Bilimleri ve Araştırmaları Dergisi. [Hakemde]

4-) Durmuş A., Yıldırım Z., Kurban R. & Karaköse E. (2022) *Comparative Performance of Novel Physics-Based Metaheuristic Algorithms for Circular Antenna Array Synthesis,* Advertising Engineering Software. [Hakemde]

5-) Durmuş A., Yıldırım Z., & Kurban R. (2022) *A Novel Metaheuristic Optimization Method for Synthesis Concentric Circular Antenna Arrays,* Electromagnetics Journal. [Hakemde]

KAYNAKÇA

Agha, M. H., Al-Adwany, M. A., Bayat, O., & Hamdoon, H. T. (2021). Optimization of antenna array pattern for uniformly excited rectangular array via thinning. *Journal of King Saud University-Engineering Sciences*.

Albagory, Y. (2021). An Efficient Conformal Stacked Antenna Array Design and 3D-Beamforming for UAV and Space Vehicle Communications. *Sensors*, *21*(4), 1362.

Almagboul, M. A., Shu, F., Qian, Y., Zhou, X., Wang, J., & Hu, J. (2019). 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*, *111*, 152854.

Awad, N. M., & Abdelazeez, M. K. (2018). Multislot microstrip antenna for ultra-wide band applications. *Journal of King Saud University-Engineering Sciences*, *30*(1), 38-45.

Azizi, M. (2021). Atomic orbital search: A novel metaheuristic algorithm. *Applied Mathematical Modelling*, *93*, 657-683.

Babayigit, B. (2018). Synthesis of concentric circular antenna arrays using dragonfly algorithm. *International Journal of Electronics*, *105*(5), 784-793.

Babayigit, B., & Senyigit, E. (2017). Design optimization of circular antenna arrays using Taguchi method. *Neural Computing and Applications*, *28*(6), 1443-1452.

Balaban, M., & Somov, A. (2015). Increasing the directivity of a circular antenna array. *Telecommunications and Radio Engineering*, *74*(13).

Balanis, C. A. (2015). *Antenna theory: analysis and design*. John wiley & sons.

Basbug, S. (2017). Design and synthesis of circular antenna array having elements controlled by linear actuators. 2017 10th International Conference on Electrical and Electronics Engineering (ELECO),

Bibiks, K., Hu, Y.-F., Li, J.-P., Pillai, P., & Smith, A. (2018). Improved discrete cuckoo search for the resource-constrained project scheduling problem. *Applied Soft Computing*, *69*, 493-503.

Bouchekara, H. (2021). Electric Charged Particles Optimization and its application to the optimal design of a circular antenna array. *Artificial Intelligence Review, 54*(3), 1767-1802.

Braik, M. S. (2021). Chameleon Swarm Algorithm: A bio-inspired optimizer for solving engineering design problems. *Expert Systems with Applications, 174*, 114685.

Chatterjee, A., Mahanti, G., & Chatterjee, A. (2012). Design of a fully digital controlled reconfigurable switched beam concentric ring array antenna using firefly and particle swarm optimization algorithm. *Progress in electromagnetics Research B, 36*, 113-131.

Chen, Y., & Wang, C.-F. (2012). Synthesis of reactively controlled antenna arrays using characteristic modes and DE algorithm. *IEEE Antennas and Wireless Propagation Letters, 11*, 385-388.

Das, A., Mandal, D., Ghoshal, S., & Kar, R. (2018). Concentric circular antenna array synthesis for side lobe suppression using moth flame optimization. *AEU-International Journal of Electronics and Communications, 86*, 177-184.

Das, A., Mandal, D., & Kar, R. (2019). An optimal far-field radiation pattern synthesis of time-modulated linear and concentric circular antenna array. *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields, 32*(6), e2658.

Dib, N. (2018). Design of planar concentric circular antenna arrays with reduced side lobe level using symbiotic organisms search. *Neural Computing and Applications, 30*(12), 3859-3868.

Dib, N., & Sharaqa, A. (2015). Design of non-uniform concentric circular antenna arrays with optimal sidelobe level reduction using biogeography-based optimization. *International Journal of Microwave and Wireless Technologies, 7*(2), 161-166.

Dib, N. I. (2016). Design of linear antenna arrays with low side lobes level using symbiotic organisms search. *Progress in electromagnetics Research B, 68*, 55-71.

Dib, N. I., Goudos, S. K., & Muhsen, H. (2010). Application of Taguchi's optimization method and self-adaptive differential evolution to the synthesis of linear antenna arrays. *Progress In Electromagnetics Research, 102*, 159-180.

Durmus, A. (2020). Circular antenna array synthesis using salp swarm optimization. *Balkan Journal of Electrical and Computer Engineering, 8*(4), 320-324.

Durmus, A., & Kurban, R. (2021). Optimal Synthesis of Concentric Circular Antenna Arrays Using Political Optimizer. *IETE Journal of Research*, 1-10.

Durmus, A., & Kurban, R. (2021). Optimum design of linear and circular antenna arrays using equilibrium optimization algorithm. *International Journal of Microwave and Wireless Technologies, 13*(9), 986-997.

Durmus, A., Kurban, R., & Karakose, E. (2021). A comparison of swarm-based optimization algorithms in linear antenna array synthesis. *Journal of Computational Electronics*, 1-12.

Dutta, K. P., & Mahanti, G. K. (2020). Evolutionary algorithms for optimal synthesis of thinned multiple concentric circular array antenna with constraints. *International Journal of Electronics, 107*(10), 1649-1662.

Dutta, K. P., & Mahanti, G. K. (2021). Synthesizing Uniform Concentric Circular Array Antenna for Minimized Sidelobes Using Teaching–Learning-Based Optimization Algorithm. In *Emerging Technologies in Data Mining and Information Security* (pp. 243-251). Springer.

Dutta, K. P., Sundi, S., Khalkho, S. S., Tirkey, J., & Hembrom, K. K. (2021). Minimization of Sidelobe Level of Uniformly Excited Concentric Circular Array Antenna Using Evolutionary Technique. In *Innovative Data Communication Technologies and Application* (pp. 185-194). Springer.

Elsaidy, E. I., Dessouky, M. I., Khamis, S., & Albagory, Y. (2012). Concentric circular antenna array synthesis using comprehensive learning particle swarm optimizer. *Progress In Electromagnetics Research Letters, 29*, 1-13.

Feng, W., & Hu, D. (2020). A Modified Whale Optimization Algorithm for Pattern Synthesis of Linear Antenna Array. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*.

Gahgouh, S., Saidi, I., & Gharsallah, A. (2022). Study and Analysis of a Novel Compact Cubic Antenna Design for WSN Applications. *Przeglad Elektrotechniczny, 98*(3).

Ghosh, P., & Das, S. (2011). Synthesis of thinned planar concentric circular antenna arrays---A differential evolutionary approach. *Progress in electromagnetics Research B, 29*, 63-82.

Graf, R. F. (1999). *Modern dictionary of electronics*. Newnes.

Guney, K., & Basbug, S. (2014). A parallel implementation of seeker optimization algorithm for designing circular and concentric circular antenna arrays. *Applied Soft Computing, 22*, 287-296.

Guney, K., Durmus, A., & Basbug, S. (2009). A plant growth simulation algorithm for pattern nulling of linear antenna arrays by amplitude control. *Progress in electromagnetics Research B, 17*, 69-84.

Güneş, A. (2013). Kil tabletlerden elektronik tabletlere: iletişim araçlarının tarihsel gelişim süreci. *Humanities Sciences, 8*(3), 277-300.

Hashim, F. A., Houssein, E. H., Hussain, K., Mabrouk, M. S., & Al-Atabany, W. (2022). Honey Badger Algorithm: New metaheuristic algorithm for solving optimization problems. *Mathematics and Computers in Simulation, 192*, 84-110.

Hashim, F. A., Hussain, K., Houssein, E. H., Mabrouk, M. S., & Al-Atabany, W. (2021). Archimedes optimization algorithm: a new metaheuristic algorithm for solving optimization problems. *Applied Intelligence, 51*(3), 1531-1551.

Hashim, F. A., & Hussien, A. G. (2022). Snake Optimizer: A novel meta-heuristic optimization algorithm. *Knowledge-Based Systems, 242*, 108320.

Jamunaa, D., Hasoon, F. N., & Mahanti, G. (2020). Symbiotic organisms search optimisation algorithm for synthesis of phase-only reconfigurable concentric circular antenna array with uniform amplitude distribution. *International Journal of Electronics Letters, 8*(4), 460-471.

Karaboga, D., & Basturk, B. (2008). On the performance of artificial bee colony (ABC) algorithm. *Applied Soft Computing, 8*(1), 687-697.

Kenane, E., Benmeddour, F., & Djahli, F. (2021). Nonuniform Circular Array Synthesis for Low Side Lobe Level Using Dynamic Invasive Weeds Optimization. *Progress in Electromagnetics research C, 111*.

Khalid, N., Mirzavand, R., Saghlatoon, H., Honari, M. M., Iyer, A. K., & Mousavi, P. (2021). A Battery-Less RFID Sensor Architecture with Distance Ambiguity Resolution for Smart Home IoT Applications. *IEEE Internet of Things Journal*.

Khodier, M. (2020). Optimization of circular antenna arrays using the cuckoo search algorithm. *International Journal of RF and Microwave Computer-Aided Engineering, 30*(8), e22247.

Khodier, M. M., & Al-Aqeel, M. (2009). Linear and circular array optimization: A study using particle swarm intelligence. *Progress in electromagnetics Research B, 15*, 347-373.

Lee, Y. (2003). Antenna circuit design for RFID applications. *AN710, Microchip Technology Inc*.

Lemaitre-Auger, P., Siragusa, R., Caloz, C., & Kaddour, D. (2013). Circular antenna arrays for near-field focused or multi-focused beams. 2013 International Symposium on Electromagnetic Theory,

Liang, S., Fang, Z., Sun, G., Liu, Y., Qu, G., & Zhang, Y. (2020). Sidelobe reductions of antenna arrays via an improved chicken swarm optimization approach. *IEEE Access, 8*, 37664-37683.

Liu, G. X., Qin, Q., & Zhang, Q. H. (2021). Linear Array Synthesis for Wireless Power Transmission Based on Brain Storm Optimization Algorithm. *International Journal of Antennas and Propagation, 2021*.

Liu, L., Wang, A., Sun, G., Zheng, T., & Yu, C. (2021). An improved biogeography-based optimization approach for beam pattern optimizations of linear and circular antenna arrays. *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields, 34*(6), e2910.

Lucyszyn, S., & Pranonsatit, S. (2013). RF MEMS for antenna applications. 2013 7th European Conference on Antennas and Propagation (EuCAP),

Mandal, D., Bhattacharjee, A., & Ghoshal, S. (2011). Swarm intelligence based synthesis of concentric circular antenna array for broadside radiation. 2011 Second International Conference on Emerging Applications of Information Technology,

Mandal, D., Ghoshal, S. P., & Bhattacharjee, A. K. (2010). Design of concentric circular antenna array with central element feeding using particle swarm optimization with constriction factor and inertia weight approach and evolutionary programing technique. *Journal of Infrared, Millimeter, and Terahertz Waves, 31*(6), 667-680.

Mirjalili, S., & Dong, J. S. (2020). *Multi-objective optimization using artificial intelligence techniques*. Springer.

Mohan, K., Kannadassan, D., & Zinka, S. (2016). Design and implementation of Dolph Chebyshev and Zolotarev circular antenna array. *Indian Journal of Science and Technology, 9*(36), 1-9.

Muttair, K. S., Shareef, O. A., Sabaawi, A. M. A., & Mosleh, M. F. (2022). Design of multiple-input and multiple-output antenna for modern wireless applications. *TELKOMNIKA (Telecommunication Computing Electronics and Control), 20*(1), 34-42.

Najjar, M. S. Y., & Khodier, N. D. M. (2008). Design of non-uniform circular antenna arrays using particle swarm optimization. *Journal of electrical engineering, 59*(4), 216-220.

Ojstersek, R., Brezocnik, M., & Buchmeister, B. (2020). Multi-objective optimization of production scheduling with evolutionary computation: A review. *International Journal of Industrial Engineering Computations, 11*(3), 359-376.

Oyelade, O. N., Ezugwu, A. E.-S., Mohamed, T. I., & Abualigah, L. (2022). Ebola optimization search algorithm: A new nature-inspired metaheuristic optimization algorithm. *IEEE Access, 10*, 16150-16177.

Painuly, S., Kohli, P., Matta, P., & Sharma, S. (2020). Advance applications and future challenges of 5G IoT. 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS),

Panduro, M. A., Mendez, A. L., Dominguez, R., & Romero, G. (2006). Design of non-uniform circular antenna arrays for side lobe reduction using the method of genetic algorithms. *AEU-International Journal of Electronics and Communications, 60*(10), 713-717.

Pereira, J. L. J., Francisco, M. B., Diniz, C. A., Oliver, G. A., Cunha Jr, S. S., & Gomes, G. F. (2021). Lichtenberg algorithm: A novel hybrid physics-based meta-heuristic for global optimization. *Expert Systems with Applications, 170*, 114522.

Pisarov, J., & Mester, G. (2020). The Impact of 5G Technology on Life in 21st Century. *IPSI BgD Transactions on Advanced Research (TAR), 16*(2), 11-14.

Poli, R., Kennedy, J., & Blackwell, T. (2007). Particle swarm optimization. *Swarm intelligence, 1*(1), 33-57.

Rabinovich, V., & Alexandrov, N. (2012). *Antenna arrays and automotive applications*. Springer Science & Business Media.

Rahaman, I., Hossain, M. S., Reza, M. F., & Sarkar, P. K. (2019). Linearly-arranged Concentric Circular Antenna Array Beamformer Using Tapering Technique. 2019 5th International Conference on Advances in Electrical Engineering (ICAEE),

Rahaman, I., Hossain, M. S., Reza, M. F., & Sarkar, P. K. (2020). Performane Analysis of Linearly-arranged Concentric Circular Antenna Array Using Robust ODL technique. 2020 IEEE Region 10 Symposium (TENSYMP),

Ram, G., Mandal, D., Kar, R., & Ghoshal, S. (2015). Cat swarm optimization as applied to time-modulated concentric circular antenna array: Analysis and comparison with other stochastic optimization methods. *IEEE transactions on Antennas and Propagation, 63*(9), 4180-4183.

Ram, G., Mandal, D., Kar, R., & Ghoshal, S. P. (2015). Circular and concentric circular antenna array synthesis using cat swarm optimization. *IETE Technical Review, 32*(3), 204-217.

Rattan, M., Patterh, M., & Sohi, B. (2009). Optimization of circular antenna arrays of isotropic radiators using simulated annealing. *International Journal of Microwave and Wireless Technologies, 1*(5), 441-446.

Ravipudi, J. L. (2020). Synthesis of linear, planar, and concentric circular antenna arrays using Rao algorithms. *International Journal of Applied Evolutionary Computation (IJAEC), 11*(3), 31-49.

Ray, T., & Liew, K.-M. (2003). Society and civilization: an optimization algorithm based on the simulation of social behavior. *IEEE Transactions on Evolutionary Computation, 7*(4), 386-396.

Sahoo, P. K., & Mandal, D. (2018). Swarm intelligence-based optimal design to minimize the sidelobe level of concentric circular antenna array with element failures. *IETE Journal of Research, 64*(4), 559-568.

Salgotra, R., Singh, U., Saha, S., & Nagar, A. K. (2020). Improved flower pollination algorithm for linear antenna design problems. In *Soft Computing for Problem Solving* (pp. 79-89). Springer.

Sankar, K. P., Kiong, T. S., & Paw, J. K. S. (2017). Dynamic social behavior algorithm for real-parameter optimization problems and optimization of hyper beamforming of linear antenna arrays. *Engineering Applications of Artificial Intelligence, 64*, 401-414.

Satish, T. T. R., Varma, P. K. K., & Raju, P. R. (2015). Invasive weed optimization (IWO) Algorithm for control of Nulls and sidelobes in a concentric circular antenna array (CCAA). *International Journal of Computer Applications, 126*(3).

Sharaqa, A., & Dib, N. (2014). Circular antenna array synthesis using firefly algorithm. *International Journal of RF and Microwave Computer-Aided Engineering, 24*(2), 139-146.

Sharaqa, A., & Dib, N. (2014). Design of linear and elliptical antenna arrays using biogeography based optimization. *Arabian Journal for Science and Engineering, 39*(4), 2929-2939.

Sharaqa, A. H., & Dib, N. I. (2012). On the optimal design of non-uniform concentric circular antenna arrays. Proceedings of the 2012 IEEE International Symposium on Antennas and Propagation,

Sharma, A., Sharma, A., Pandey, J. K., & Ram, M. (2022). Swarm Intelligence: Foundation, Principles, and Engineering Applications.

Singh, H., Abouhawwash, M., Mittal, N., Salgotra, R., Mahajan, S., & Pandit, A. K. (2022). Performance evaluation of Non-Uniform circular antenna array using integrated harmony search with Differential Evolution based Naked Mole Rat algorithm. *Expert Systems with Applications, 189*, 116146.

Singh, N., Houssein, E. H., Singh, S., & Dhiman, G. (2022). HSSAHHO: a novel hybrid Salp swarm-Harris hawks optimization algorithm for complex engineering problems. *Journal of Ambient Intelligence and Humanized Computing*, 1-37.

Singh, U., & Salgotra, R. (2018). Synthesis of linear antenna array using flower pollination algorithm. *Neural Computing and Applications*, *29*(2), 435-445.

Su, D., Li, L., Yang, S., & Wang, F. (2021). Adaptive Maxwell's equations derived optimization and its application in antenna array synthesis. *China Communications*, *18*(5), 263-272.

Subhashini, K. (2020). Runner-root algorithm to control sidelobe level and null depths in linear antenna arrays. *Arabian Journal for Science and Engineering*, *45*(3), 1513-1529.

Sultan, K., Ikram, M., & Nguyen-Trong, N. (2022). A Multiband Multibeam Antenna for Sub-6 GHz and mm-Wave 5G Applications. *IEEE Antennas and Wireless Propagation Letters*, *21*(6), 1278-1282.

Sun, G., Liu, Y., Li, H., Liang, S., Wang, A., & Li, B. (2018). An antenna array sidelobe level reduction approach through invasive weed optimization. *International Journal of Antennas and Propagation*, *2018*.

Swathi, A., Chakravarthy, V., & Krishna, M. V. (2021). Circular antenna array optimization using modified social group optimization algorithm. *Soft Computing*, 1-9.

Swathi, A., Chakravarthy, V., & Krishna, M. V. (2021). Circular antenna array optimization using modified social group optimization algorithm. *Soft Computing*, *25*(15), 10467-10475.

Tailor, N. K., Abdi-Jalebi, M., Gupta, V., Hu, H., Dar, M. I., Li, G., & Satapathi, S. (2020). Recent progress in morphology optimization in perovskite solar cell. *Journal of Materials Chemistry A*, *8*(41), 21356-21386.

Talatahari, S., & Azizi, M. (2020). Optimization of constrained mathematical and engineering design problems using chaos game optimization. *Computers & Industrial Engineering*, *145*, 106560.

Talatahari, S., Azizi, M., Tolouei, M., Talatahari, B., & Sareh, P. (2021). Crystal Structure Algorithm (CryStAl): A Metaheuristic Optimization Method. *IEEE Access, 9*, 71244-71261.

Trojovský, P., & Dehghani, M. (2022). Pelican Optimization Algorithm: A Novel Nature-Inspired Algorithm for Engineering Applications. *Sensors, 22*(3), 855.

Wang, Z.-J., Zhan, Z.-H., Kwong, S., Jin, H., & Zhang, J. (2020). Adaptive granularity learning distributed particle swarm optimization for large-scale optimization. *IEEE transactions on cybernetics, 51*(3), 1175-1188.

Wang, Z., Wu, Y., Mahmud, M. H., Yuan, Z., Zhao, Y., & Mantooth, H. A. (2020). Busbar design and optimization for voltage overshoot mitigation of a silicon carbide high-power three-phase T-type inverter. *IEEE Transactions on Power Electronics, 36*(1), 204-214.

Witten Jr, T. A., & Sander, L. M. (1981). Diffusion-limited aggregation, a kinetic critical phenomenon. *Physical review letters, 47*(19), 1400.

Wu, S. R., Huang, C. C., Mar, J., & Chou, J. H. (2018). A versatile pattern synthesis algorithm for circular antenna array. *International Journal of RF and Microwave Computer-Aided Engineering, 28*(5), e21254.

Zeidabadi, F. A., Dehghani, M., Trojovský, P., Hubálovský, Š., Leiva, V., & Dhiman, G. (2022). Archery algorithm: A novel stochastic optimization algorithm for solving optimization problems. *Computers, Materials and Continua, 72*(1), 399-416.

Zhang, X., & Zhang, X. (2017). Circular antenna design by adaptive position inheritance artificial bee colony algorithm. *Physical Communication, 25*, 369-375.

Zheng, T., Liu, Y., Sun, G., Liang, S., Han, J., Ju, Q., & Li, S. (2020). Joint sidelobe suppression and nulls control of large-scale linear antenna array using particle swarm optimization with global search and population mutation. *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields, 33*(4), e2710.

Zheng, T., Liu, Y., Sun, G., Zhang, L., Liang, S., Wang, A., & Zhou, X. (2020). IWORMLF: Improved invasive weed optimization with random mutation and Lévy flight for beam pattern optimizations of linear and circular antenna arrays. *IEEE Access, 8*, 19460-19478.

ÖZGEÇMİŞ

YABANCI DİL

İngilizce: 63,75 (2022/1 Yök-Dil)

ULUSLARARASI KONFERANSLAR

1-) Durmuş A., Yıldırım Z., & Kurban R. (2021) *Atomik Yörünge Arama Algoritması Kullanarak Doğrusal Anten Dizi Sentezi*, ICONSAD-2021.

2-) Karaköse E., Kurban R., Durmuş, A. L. İ., Çıtıl, F., & Yıldırım, Z. (2021) *Classification of Benign and Malignant Tumors by Machine Learning Methods using MRI Images of the Brain*, 3rd International Eurasian Conference on Science.

TR DİZİN YAYINLAR

1-) Durmuş A., Yıldırım Z. (2021) *Side Lobe Reduction of Linear Antenna Array by Position-Only Control Using Novel Metaheuristic Optimization Methods,* ESOGÜ Müh Mim Fak Derg. [Kabul Edildi]

2-) Durmuş A., Yıldırım Z. (2022) *Synthesis of Linear Antenna Arrays with Physics Based AOA, CryStAl and LA Algorithms,* Mühendislik Bilimleri ve Araştırmaları Dergisi. [Hakemde]

SCI YAYINLAR

1-) Durmuş A., Yıldırım Z., Kurban R. & Karaköse E. (2022) *Comparative Performance of Novel Physics-Based Metaheuristic Algorithms for Circular Antenna Array Synthesis*, Advertising Engineering Software. [Hakemde]

2-) Durmuş A., Yıldırım Z., & Kurban R. (2022) *A Novel Metaheuristic Optimization Method for Synthesis Concentric Circular Antenna Arrays,* Electromagnetics Journal. [Hakemde]