



**FATİH SULTAN MEHMET VAKIF ÜNİVERSİTESİ  
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ  
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI  
BİLGİSAYAR MÜHENDİSLİĞİ PROGRAMI**

**EN KAPSAMLI TARAMA ALANI İÇİN RADAR  
YERLEŐTİRME PROBLEMİ**

**YÜKSEK LİSANS TEZİ**

**EMRAH YULİAN TEMA**

**İSTANBUL, 2022**



**FATİH SULTAN MEHMET VAKIF ÜNİVERSİTESİ  
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ  
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI  
BİLGİSAYAR MÜHENDİSLİĞİ PROGRAMI**

**EN KAPSAMLI TARAMA ALANI İÇİN RADAR  
YERLEŞTİRME PROBLEMİ**

**YÜKSEK LİSANS TEZİ**

**EMRAH YULIAN TEMA  
(200221001)**

**Danışman  
(Dr. Öğr. Üyesi Shaaban A.I. Sahmoud)  
İkinci Danışman  
(Dr. Öğr. Üyesi Berna Kiraz)**

**İSTANBUL, 2022**

20/07/2022

LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ MÜDÜRLÜĞÜNE

Bilgisayar Mühendisliği Anabilim Dalı Bilgisayar Mühendisliği Tezli Yüksek Lisans programı 200221001 numaralı Emrah Yulian TEMA'nın hazırladığı "En Kapsamlı Tarama Alanı İçin Radar Yerleştirme Problemi" konulu Yüksek Lisans tezi ile ilgili Tez Savunma Sınavı, 20/07/2022 Çarşamba günü saat 13:00'da yapılmış, sorulara alınan cevaplar sonunda adayın tezinin **Kabulüne Oy Birliği** ile karar verilmiştir.

**Düzeltilme verilmesi halinde:**

Adı geçen öğrencinin Tez Savunma Sınavı .../.../20... tarihinde, saat ...:.... da yapılacaktır.

**Tez adı değişikliği yapılması halinde:** Tez adının .....

.....  
şeklinde değiştirilmesi uygundur.

|    | Jüri Üyesi                                     | Karar |
|----|--|-------|
| 1. | Dr.Öğr. Üyesi Shaaban A.I. SAHMOUD (Danışman)  | KABUL |
| 2. | Prof. Dr. Ayşe Şima UYAR                       | KABUL |
| 3. | Dr. Öğr. Üyesi Hüsamettin OSMANOĞLU            | KABUL |
| 4. | (İkinci Danışman) * Dr. Öğr. Üyesi Berna KİRAZ |       |

\*2. Danışman varsa doldurulması gerekmektedir.

## **ETİK BİLDİRİM**

Bu tezin yazılmasında bilimsel ahlak kurallarına uyulduğunu, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, tezin herhangi bir kısmının bağlı olduğum üniversite veya bir başka üniversitedeki başka bir çalışma olarak sunulmadığını beyan ederim.

Emrah Y. Tema

## TEŐEKKÜR

Bu tezin yazılmasında büyük katkıları olan danışmanlarım Dr. Öğr. Üyesi Berna KİRAZ ve Dr. Öğr. Üyesi. Shaaban A. I. SAHMOUD'a, çalışmama destek olan ve tez konumu bulmamda yardımcı olan Tolga BAŐTÜRK'e, takıldığım konularda hep yardımcı olan çalışma arkadaşım Furkan ÖNEL'e ve yüksek lisans ve tez zamanımda her zaman destek olan eşim Merve TEMA'ya çok teşekkür ederim.

Emrah Y. Tema

# EN KAPSAMLI TARAMA ALANI İÇİN RADAR YERLEŐTİRME PROBLEMİ

**Emrah Y. Tema**

## ÖZET

Hava sahası gözlemi, ülkelerin hava sahası güvenliđi ve kontrolü için önemli bir konudur. Kullanılan radarların sayısı ve kapsama oranı ise bu durumda dikkat edilen konulardan başlıcalarıdır. Belirli bir bölgede en az sayıda radar ile en yüksek kapsama oranını sağlamak, hem bölgedeki güvenliđi ve hava sahası emniyetini etkili bir şekilde sağlayabilmek, hem de bunu en düşük maliyet ile gerçekleőtirmek bu tezin ana konusudur. Radar yerleőtirmeleri yapılırken belirli yöntemlere dayanılarak yapılması, verimliliđi arttırarak daha kontrollü bir şekilde kurulumların yapılmasını sağlayabilmektedir, bunun için istenilen bölgede en etkin kapsama oranı için radarların nerelere yerleőtirilebileceđinin önceden bulunup öyle kurulması önemli bir konu olmaktadır. Bu tez kapsamında radarların nerelere yerleőtirilecekleri ve bu yerleőtirmeler ile bölgenin ne kadarının kapsanabileceđi, çok amaçlı bir optimizasyon problemi olarak ele alınmış ve bu yönde çalışmalar gerçekleőtirilmiştir. Çalışmalar kapsamında DTED-1 harita yükseklik verileri kullanılarak radarların yerleőtirileceđi alan oluşturulmuş ve bu alana çok amaçlı metasezgisel algoritmalar kullanılarak radar yerleőtirmeleri gerçekleőtirilmiştir. Kullanılan algoritmalar arasında NSGA-II algoritması bu çalışmada en iyi sonuçları veren algoritma olarak gözlemlenmiştir ve bu algoritmanın radar yerleőtirme ihtimalinin hücre yüksekliklerine bağlanmasıyla en yüksek kapsama oranları elde edilmiştir. Az sayıda radarla %98'in üzerinde kapsama oranları elde edilebilmiştir ve radar sayısı arttıkça bu oran %99'ları geçmiştir.

Radar yerleőtirme problemi, Hava sahası gözlemi, Radar kapsamaları, Çok amaçlı metasezgisel algoritmalar, Yerleőtirme optimizasyonu.

# **THE PROBLEM OF PLACING RADARS FOR THE MOST EXTENSIVE SCANNING AREA**

**Emrah Y. Tema**

## **ABSTRACT**

Airspace surveillance is an important issue for countries' airspace security and control. The number of radars used and the coverage rate are the main issues to be considered in this case. The main subject of this thesis is to provide the highest coverage rate with the least number of radars in a certain region, to provide the security and airspace security in the region effectively, and to do this with the lowest cost. Making radar placements based on certain methods can increase efficiency and enable more controlled installations. For this, it is an important issue to find where the radars can be placed in advance for the most effective coverage rate in the desired region and set up. Within the scope of this thesis, where the radars will be placed and how much of the region can be covered by these settlements are considered as a multi-objective optimization problem and studies have been carried out in this direction. Within the scope of the studies, the area where the radars will be placed was created using the DTED-1 map height data, and radar placements were carried out using multi-objective metaheuristic algorithms in this area. Among the algorithms used, the NSGA-II algorithm was observed as the algorithm that gave the best results and the highest coverage ratios has been obtained by making radar placement probabilities dependent on cell heights. With a small number of radars, coverage rates above 98% could be achieved, and as the number of radars increased, this rate has exceeded 99%.

Radar placement problem, Airspace surveillance, Radar coverages, Multi-objective metaheuristic algorithms, Placement optimization.

## ÖNSÖZ

Bu tezin amacı, belirli bir bölge için en az sayıda radarla en yüksek kapsamın elde edilmesidir. Ülkelerin hava sahası güvenlikleri söz konusu olduğunda kullanılan radarların en etkili şekilde gökyüzünü tarayabilmesi önemli bir konudur ve bunun etkili bir şekilde yapılması gerekmektedir. Bu sayede güvenlik artarken maliyetler azaltılabilmektedir. Tez kapsamında problem çok amaçlı bir optimzasyon problemine dönüştürülmüş olup radarların ana çalışma prensibi dikkate alınarak ve sinyal işleme konularına girilmeden problemin çözülmesi hedeflenmiştir.

Temmuz 2022

Emrah Y. Tema



## İÇİNDEKİLER

|   |      |
|---|------|
| ÖZET.....   | v    |
| ABSTRACT.....   | vi   |
| ÖNSÖZ.....  | vii  |
| SEMBOLLER .....   | x    |
| TABLO LİSTESİ .....   | xi   |
| ŞEKİL LİSTESİ.....  | xiii |
| KISALTMALAR .....   | xiv  |
| GİRİŞ .....   | 1    |
| BİRİNCİ BÖLÜM.....  | 4    |
| 1. TEMEL KAVRAMLAR VE LİTERATÜR ÇALIŞMALARI .....             | 4    |
| 1.1. RADAR YERLEŞTİRME PROBLEMİ .....                         | 4    |
| 1.2. METASEZGİSEL YÖNTEMLER.....                              | 5    |
| 1.3. ÇOK AMAÇLI BENZETİMLİ TAVLAMA ALGORİTMASI .....          | 5    |
| 1.4. BELLEK TABANLI ÇOK AMAÇLI GENETİK ALGORİTMA.....         | 6    |
| 1.5. BASTIRILMAMIŞ SIRALAMALI GENETİK ALGORİTMA 2 .....       | 6    |
| 1.6. LİTERATÜR TARAMASI.....                                  | 6    |
| İKİNCİ BÖLÜM .....  | 12   |
| 2. PROBLEMİN TANIMLANMASI .....                               | 12   |
| 2.1. HARİTA VERİLERİNİN İŞLENMESİ.....                        | 12   |
| 2.1.1. Mekik Radar Topografya Misyonu (SRTM) Nedir?.....      | 12   |
| 2.1.2. İnterferometri Yöntemi ile Yükseklik Hesaplanması..... | 13   |
| 2.1.3. Dijital Harita Yükseklik Verileri (DTED).....          | 13   |
| 2.1.4. Yükseklik Harita Verileri Ne İçin Kullanılabilir?..... | 15   |
| 2.1.5. Geospatial Veri Soyutlama Kütüphanesi .....            | 15   |
| 2.1.6. Çalışma Bölgesinin Oluşturulması .....                 | 15   |
| 2.2. RADARLAR VE KAPSAMA HESAPLAMALARI .....                  | 18   |
| 2.2.1. En-Route Radarı Çalışma Prensipleri.....               | 18   |

|   |           |
|---|-----------|
| 2.2.2. Radarın Modellenmesi ve Tarama Hesaplamalarında Kullanılan Formüller ..... | 19        |
| 2.2.3. Harita Üzerinde Radar Tarama İşlemleri.....                                | 24        |
| 2.2.4. Amaç Fonksiyonları.....  | 35        |
| <b>ÜÇÜNCÜ BÖLÜM</b> .....   | <b>36</b> |
| <b>3. METODOLOJİ</b> .....  | <b>36</b> |
| 3.1. ALGORİTMALARDA KULLANILAN YÖNTEMLER.....                                     | 36        |
| 3.1.1. Çözüm Gösterimleri .....   | 36        |
| 3.1.2. Uygunluk Fonksiyonları.....  | 37        |
| 3.1.3. Çözüm Dominantlığı Kontrolü .....  | 38        |
| 3.1.4. Genetik Değişim .....  | 39        |
| 3.1.5. Mutasyon .....   | 39        |
| 3.1.6. Hiper Hacim Performans Metriği .....                                       | 40        |
| 3.2. KULLANILAN ALGORİTMALAR .....  | 41        |
| 3.2.1. Çok Amaçlı Benzetimli Tavlama Algoritması.....                             | 41        |
| 3.2.2. Bellek Tabanlı Çok Amaçlı Genetik Algoritma .....                          | 43        |
| 3.2.3. Bastırılmamış Sıralamalı Genetik Algoritma 2.....                          | 45        |
| 3.3. NSGA-II İÇİN UYGULANAN YÖNTEMLER.....  | 47        |
| 3.3.1. Harita Çözünürlüğünün Düşürülmesi .....                                    | 48        |
| 3.3.2. Radar Yerleştirme İhtimalinin Hücre Bazlı Olarak Güncellenmesi .           | 48        |
| 3.3.3. Genetik Değişim ve Mutasyon Değişiklikleri .....                           | 50        |
| <b>DÖRDÜNCÜ BÖLÜM</b> .....   | <b>51</b> |
| <b>4. DENEYSEL ÇALIŞMALAR</b> .....   | <b>51</b> |
| 4.1. PARAMETRE ATAMALARI .....  | 51        |
| 4.2. BULGULAR VE TARTIŞMA .....   | 52        |
| <b>SONUÇ</b> .....  | <b>61</b> |
| <b>KAYNAKÇA</b> .....   | <b>62</b> |

## SEMBOLLER

|           |                                     |
|-----------|-------------------------------------|
| $\alpha$  | : Radyan Formatında Açı             |
| $w$       | : Ağırlık                           |
| $p$       | : İhtimal                           |
| $\phi$    | : Enlem Bilgisi                     |
| $\lambda$ | : Boylam Bilgisi                    |
| $\theta$  | : Kuzeyden Saat Yönünde Açı Bilgisi |
| $R$       | : Dünyanın Yarıçap Bilgisi          |
| $\Delta$  | : Açısal Mesafe                     |

## TABLO LİSTESİ

### Sayfa

|   |    |
|---|----|
| <b>Tablo 4.1:</b> MOSA algoritmasının çalışma parametreleri.....  | 51 |
| <b>Tablo 4.2:</b> MBMO-GA çalışma parametreleri.....  | 51 |
| <b>Tablo 4.3:</b> NSGA-II için çalışma parametreleri.....   | 51 |
| <b>Tablo 4.4:</b> MOSA, MBMO-GA ve NSGA-II algoritmalarının çözüm kümelerine ait hiper hacim oranı istatistikleri.....  | 52 |
| <b>Tablo 4.5:</b> En yüksek hiper hacim oranına sahip çözüm kümelerinin bulduğu radar sayıları ve kapsama oranları (kapsama oranları % olarak verilmiştir).....       | 53 |
| <b>Tablo 4.6:</b> MOSA, MBMO-GA ve NSGA-II algoritmalarının 10000, 15000 ve 20000'inci çözümlerde elde ettikleri hiper hacim oranı istatistiksel değerlendirmesi..... | 54 |
| <b>Tablo 4.7:</b> NSGA-II algoritmasının farklı harita çözünürlüklerinde elde ettiği çözüm kümelerine ait hiper hacim oranları.....                                   | 56 |
| <b>Tablo 4.8:</b> En iyi hiper hacim oranlarına ait çözüm kümelerinin radar sayısı ve kapsama oranı bilgileri.....  | 56 |
| <b>Tablo 4.9:</b> NSGA-II'nin farklı haritalarda çalıştırılması sonucunda elde edilen hiper hacim oranlarının karşılaştırılması.....                                  | 57 |
| <b>Tablo 4.10:</b> En iyi hiper hacim oranlarına ait çözüm kümelerinin radar sayısı ve kapsama oranı karşılaştırmaları.....   | 57 |
| <b>Tablo 4.11:</b> MA-NSGA-II ve YA-NSGA-II algoritmalarının hiper hacim oranı istatistiklerinin NSGA-II ile karşılaştırılması.....                                   | 58 |
| <b>Tablo 4.12:</b> : En iyi hiper hacim oranlı çözüm kümelerindeki radar sayısı ve kapsama oranlarının karşılaştırılması.....   | 58 |
| <b>Tablo 4.13:</b> NSGA-II, MA-NSGA-II ve YA-NSGA-II'nin 100, 150, 200, 250 ve 300 iterasyonda elde ettikleri hiper hacim oranı istatistiksel değerlendirmesi.....    | 58 |
| <b>Tablo 4.14:</b> NSGA-II, UC-YA-NSGA-II ve UCD-YA-NSGA-II hiper hacim oranı analizleri ve karşılaştırmaları.....  | 59 |

**Tablo 4.15:** En iyi hiper hacim oranlarına ait çözüm kümelerinin radar sayısı ve kapsama oranı karşılaştırmaları.....59

**Tablo 4.16:** NSGA-II, UC-YA-NSGA-II ve UCD-YA-NSGA-II'nin 100, 150, 200, 250 ve 300 iterasyonda elde ettikleri hiper hacim oranı istatistiksel değerlendirmesi..60

## ŞEKİL LİSTESİ

### Sayfa

|  |    |
|--|----|
| <b>Şekil 2.1:</b> Çalışma için seçilen bölgenin çözünürlüğü düşürülmüş matrisinin görselleştirilmesi ve yanında gerçek harita görüntüsü [Url-4].....   | 17 |
| <b>Şekil 2.2:</b> Çalışmada kullanılan 2 boyutlu haritanın temsili bir gösterimi.....  | 18 |
| <b>Şekil 2.3:</b> En-Route radarın tarama prensibi.....  | 20 |
| <b>Şekil 2.4:</b> Açısal hesaplamalara yardımcı görsel.....  | 21 |
| <b>Şekil 2.5:</b> Radar dönüşleri arasında yapılması gereken açı değişimi denklemini ifade eden görsel.....  | 22 |
| <b>Şekil 2.6:</b> En yüksek ve en düşük kapsama oranlarını veren radarlar ile deniz kıyısına yerleştirilmiş bir radarın harita üzerindeki tek bir satırda görebildiği en küçük ve en büyük yükseklikler..... | 33 |
| <b>Şekil 2.7:</b> En yüksek ve en düşük kapsama oranlarını veren radarlar ile deniz kıyısına yerleştirilmiş bir radarın konumları ve kapsamlarıyla ilgili görselleştirmeler.....                             | 35 |
| <b>Şekil 4.1:</b> Çalıştırılan algoritmaların hiper hacim oranı ve iterasyon sayısı yakınsama grafikleri. (A) MOSA, (B) MBMO-GA ve (C) NSGA-II.....  | 52 |
| <b>Şekil 4.2:</b> En yüksek hiper hacim oranına sahip çözüm kümelerinin bulunduğu radar sayıları ve kapsama oranları (kapsama oranları % olarak verilmiştir).....  | 53 |
| <b>Şekil 4.3:</b> NSGA-II'nin 4 radara kadar bulunduğu çözümlerdeki radarların harita üzerine yerleştirilmesi.....   | 54 |
| <b>Şekil 4.4:</b> Varsayılan harita ile çözünürlükleri düşürülmüş haritalar.....   | 55 |
| <b>Şekil 4.5:</b> Yeni haritanın iki boyutlu matris görüntüsü ve gerçek görüntüsü [Url-4]...56   | 56 |

## KISALTMALAR

|                |   |
|----------------|---|
| GA             | Genetik Algoritma                                 |
| MAX            | Maksimum  |
| MA-NSGA-II     | Merkez Ağırlıklı NSGA-II                          |
| MBMO-GA        | Bellek Tabanlı Çok Amaçlı Genetik Algoritma       |
| MOSA           | Çok Amaçlı Benzetimli Tavlama Algoritması         |
| NSGA-II        | Bastırılmamış Sıralamalı Genetik Algoritma 2      |
| ORT            | Ortalama  |
| POF            | Pareto Optimal Cephe                              |
| POS            | Pareto Optimal Kümesi                             |
| SA             | Benzetimli Tavlama Algoritması                    |
| UC-YA-NSGA-II  | Tekdüze Genetik Değişimli YA-NSGA-II              |
| UCD-YA-NSGA-II | Tekdüze Genetik Değişimli ve Düşürmeli YA-NSGA-II |
| VAR            | Varyans   |
| YA-NSGA-II     | Yükseklik Ağırlıklı NSGA-II                       |

## GİRİŞ

Günümüzde hava sahası gözlemi ve kontrolü, ülkelerin hava sahası güvenlikleri için vazgeçilmez bir unsurdur ve gittikçe önemli bir hale gelmektedir [1]. Genellikle gözlem ve kontrol işlemlerinde bir takım radarlar ve sistemler kullanılmakta ve bir takım başarımlar hedeflenmektedir. Bunlar belirli bir yükseklikten itibaren en yüksek kapsama oranını sağlamak, en az radar kullanmak, belirli bir bölgeyi birden fazla radarın tarayıp kesişim alanının yüksek tutulması, radarların maliyetinin daha az olacağı yerlere kurulması gibi başarımlardır. Bu çalışma kapsamında en temel başarımlar hedeflerinden ikisi ele alınmaktadır. Birinci başarımlar indirgenmiş olarak belirli bir yükseklik dikkate alınmadan en yüksek kapsama oranını sağlamak, ikincisi ise birinci başarımları elde ederken en az sayıda radar kullanmak. En yüksek kapsama oranını en az sayıda radar kullanarak gerçekleştirmek bu çalışmadaki hedef olarak belirlenmiştir. En yüksek kapsama oranı ile hava sahası kontrolü daha iyi gerçekleştirilebilir ve gereğinden fazla radar kullanımının gereksiz maliyetinden kurtulmak önemli bir konu olduğu için en uygun radar konumları [2] ve sayılarının tespit edilmesi gerekmektedir.

Belirli bir bölgedeki [3] en yüksek kapsama oranını en az sayıda radarla elde edebilmek için bu problemi iki amaçlı bir optimizasyon problemine dönüştürülebilmektedir. Bunun için taranacak bölgeyi hücelere bölerek [4], belirli konumlara yerleştirilen radarların kaç adet hücreyi kapsadığı ve bunun alanın yüzde kaçını kapsadığı tespit edilebilmektedir [5]. Bu çalışmada hücreler dikdörtgen biçiminde ve üç boyutlu olarak değerlendirilmiştir ve kapsama alanına giren hücrelerin hacmi ile temsil edilen alan, kapsanmış olarak değerlendirilmiştir. Tabii ki radarlar tarafından kapsanan hücrelerin temsil ettiği alanın hacmi ne kadar küçükse, çözünürlük de o kadar yüksek olacaktır ve çözünürlük ne kadar yüksekse elde edilen sonuçlardaki hata payı da bir o kadar az olacaktır.



Problemin çözüleceği alan bu çalışmada üç boyutlu olarak ele alınmış ve radarların yerleşimleri ile kapsadıkları alanlara ait hesaplamalar, hücre hacimleri dikkate alınarak üç boyutlu olarak gerçekleştirilmiştir. Öncelikle tarama işlemi gerçekleştirilecek bir bölge seçilmiş olup, bölge tabanı iki boyutlu olarak ele alınıp hepsi aynı büyüklüğe sahip olan hücrelere bölünmüştür. Bu hücreler üzerine radarlar yerleştirilerek, diğer hücrelerin ne kadarlık kısımlarını kapsadıkları belirlenmiş ve ardından her bir hücrenin üzerine havasahasını oluşturmak amacıyla hücreler eklenerek bu hücrelerin kaç tanesinin kapsanabildiği hesaplanmıştır. Böylece üç boyutlu tarama işlemi gerçekleştirilebilmiştir.

Problem bu çalışmada iki amaçlı bir optimizasyon problemi olarak ele alınmış olup, amaç fonksiyonları hedeflenen başarımların tespitine göre belirlenmiştir. Bunlardan birisi problem için üretilen çözümlerde kaç adet radar olduğunu tespit ederken, diğeri bu çözümlerdeki radarların konumlarını ele alarak bölgenin yüzde kaçının bu radarlar tarafından kapsandığını hesaplamaktadır. Bu iki amaç fonksiyonuna göre çözümlerin kalitesi belirlenmektedir. Bu çözümlerin üretiminde çok boyutlulukla başa çıkabilmek için bir takım metasezgisel algoritmalar kullanılmış ve ürettikleri çözümlere göre performansları karşılaştırılmıştır. Bu algoritmalar; Çok Amaçlı Benzetimli Tavlama Algoritması (MOSA), Bellek Tabanlı Çok Amaçlı Genetik Algoritma (MBMO-GA) ve Bastırılmamış Sıralamalı Genetik Algoritma 2 (NSGA-II)'dir

Çalışma sonucunda elde edilen sayısal sonuçlara göre kullanılan algoritma ve yöntemlerin, belirlenen kısıtlar çerçevesinde oluşturulan alana etkili bir şekilde yerleştirilebildikleri ve yüksek kapsama oranı sağlanabildiği gözlemlenmiştir. En iyi kapsama oranlarının elde edildiği algoritmanın Yükseklik Ağırlıklı NSGA-II (YA-NSGA) olduğu gözlemlenmiştir. Bu algoritma hakkında detaylı bilgiler üçüncü ve dördüncü bölümlerde verilmektedir.

Bu tezin ilk bölümünde çok amaçlı metasezgisel algoritmalar ve problemin kendisinden bahsedilmiş olup literatür taraması yapılmıştır. İkinci bölümde problemin tanımlanması, harita verileri ve bu verilerin işlenmesi, radarlar hakkında bilgiler ve bu çalışmada radarların nasıl yerleştirilip kapsama alanlarının hesap edildiği ile problemin nasıl çözüldüğü yer almaktadır. Üçüncü bölümde problemin çok amaçlı

metasezgisel algoritmalar ile nasıl ele alındığı ve dördüncü bölümde de elde edilen sonuçlar ilgili bilgiler ve sayısal analizler yer almaktadır.

## BİRİNCİ BÖLÜM

### 1. TEMEL KAVRAMLAR VE LİTERATÜR ÇALIŞMALARI

#### 1.1. RADAR YERLEŞTİRME PROBLEMİ

Hava sahasının güvenli yönetimi ve savunma amaçlı hava sahası kontrolü ülkeler için önemli bir konudur. Bu amaçla hava sahası gözlem radarları kullanılmaktadır [6]. Gözlem radarlarının nerelere yerleştirileceği, yerleştirildikleri yerdeki görüş açılarının ne kadar engellendiği, kuruldukları yerlerdeki bakım maliyetleri ve kaç adet yerleştirilirse hedeflenen bölgede istenilen kapsamanın sağlanabileceği araştırmalara konu olmaktadır. Radarların sinyal gücünün, kalitesinin ve çalışmadaki doğruluğunun artırılmasına yönelik çalışmalar olduğu gibi, kapsama oranını ve yerleşim maliyetlerinin hesaba katıldığı çalışmalar mevcuttur. Örneğin bir radarı bir şehir merkezine koymanın bakım maliyeti düşüktür ancak dağ başına koymanın bakım maliyeti yüksektir. Bir radar iki dağın arasına koyulduğunda kapsama oranı bir hayli düşebiliyorken, bir dağın tepesine koyulduğunda bu oran yükselmektedir ama aynı şekilde daha alçak yerleri gözlemlemedeki performansı da radarın tipine göre değişkenlik gösterip düşebilmektedir. Bütün bunlar ve bölgedeki engeller hesaba katıldığında radar yerleşimlerinin düzgün yapılabilecek bir kapsama oranının sağlanması çok önemli bir hal almaktadır.

Radar yerleştirme problemi genel itibarıyla belirli bir hava sahasını gözleme amacıyla belirli bir bölgeye radarların yerleştirilmesi olayını kapsamaktadır. Radarların yerleştirilmesindeki amaç hava sahası gözlemidir ve bunun en yüksek seviyede olması beklenmektedir ancak bunu tek bir radar ile yapmak çoğu zaman yeterli değildir. Birden çok radar kullanmak gerektiğinde ise radarların birbirleriyle etkileşimlerinin nasıl olacağı, aynı bölgeyi gözlemleyerek kesişim yaratacak radarların gerekip gerekmediği, radarların nerelere yerleştirileceği ve kaç adet olacağı gibi soruların cevaplanması gerekmektedir. Böyle sorulara cevap aramak için bir takım yöntemler uygulanmış ve bir takım çalışmalar gerçekleştirilmiştir.

## 1.2. METASEZGİSEL YÖNTEMLER

Metasezgisel yöntemler önemli birer arama ve optimizasyon yöntemleridir. Bu yöntemler en iyinin ya da en iyiye yakın çözümlerin bulunmasında arama uzayını araştırmayı hedeflerler. Genellikle iki türde ele alınabilirler. Bunlardan biri tek bir çözüm kullanarak lokal arama yapanlar ve popülasyon tabanlı rasgele arama yapan yöntemlerdir. Yüksek karmaşıklıkta bir çok problemin çözümünde kullanılabilirler [7]. Bu çalışmada metasezgisel algoritmalarından Çok Amaçlı Benzetimli Tavlama Algoritması (MOSA), MOSA'daki Pareto Optimal Cephe (POF) mantığı ve iki amaç fonksiyonu eklenerek klasik Genetik Algoritma'nın (GA) güncellenmiş hali olan Bellek Tabanlı Çok Amaçlı Genetik Algoritma (MBMO-GA) ve Bastırılmamış Sıralamalı Genetik Algoritma 2 (NSGA-II)'dir.

## 1.3. ÇOK AMAÇLI BENZETİMLİ TAVLAMA ALGORİTMASI

Benzetimli Tavlama Algoritması (SA), sürekli ve ayrık optimizasyon problemlerini ele almak için kullanılan bir meta-sezgisel yerel arama algoritmasıdır. Algoritmanın temel özelliği, global bir optimum bulma umuduyla amaç fonksiyon değerini kötüleştiren hareketlere izin vererek yerel optimumdan kaçmaya çalışmasıdır [8]. Benzetimli tavlama algoritması, karar vericiye geniş bir baskılanmayan çözümler seti sunarak birden çok amacı olan bir forma sahip olabilmektedir. Yöntem buluşsal olduğu için sadece optimal olmayan çözümler bulunabilir ve yineleme sayısı ne kadar fazla olursa baskılanamayan çözümler bulma şansı da o kadar yüksek olur [9].

Çok Amaçlı Benzetimli Tavlama Algoritması (MOSA), SA'nın çok amaçlı optimizasyon için genişletilmiş bir halidir. Uygulanma alanını çözümlerken baskılanamayan çözümleri bellekte tutması sayesinde optimal çözüme yakın çözümleri saklayabilmektedir [10]. Bu çalışmada, iki adet hedefimizin olmasından dolayı radar yerleştirme amacıyla MOSA kullanılmıştır. Bu algoritmanın yerleştirme problemlerinde sıklıkla kullanıldığı bilinmektedir ve bu yüzden radar yerleşimlerini düzgün yapacağı varsayılarak seçilmiştir. Yerleştirme problemlerinde sıkça kullanılan ve iyi bir sonuç veren bir algoritma olma özelliğine sahiptir.

#### 1.4. BELLEK TABANLI ÇOK AMAÇLI GENETİK ALGORİTMA

Klasik Genetik Algoritma (GA), canlıların çiftleşmesi sonucu genlerindeki özellikleri yeni nesile aktarmaları becerisinden esinlenerek oluşturulmuş bir algoritmadır. Mantığında güçlü olan bireylerin bir sonraki nesli oluşturmada bir takım faktörler karşısında daha avantajlı olmaları durumunu yansıtmaktadır. Daha güçlü bireylerin oluşturacağı daha güçlü yeni jenerasyonun yaşama şansı daha yüksektir sürekli iyiye giden bir döngüye gireceği varsayılmaktadır [7]. Bu çalışmada birden fazla amaç olmasından ötürü algoritma çok amaçlı bir hale getirilmiştir ve aynı zamanda da MOSA'da olduğu gibi her jenerasyonda tespit edilen baskılanamayan çözümler Pareto Optimal Cephe (POF) özelliğinde bir bellekte tutularak optimal çözüme en yakın bireyler saklanmıştır. Bu değişikliklerle birlikte bu çalışmada güncellenmiş bu algoritmadan Bellek Tabanlı Çok Amaçlı Genetik Algoritma (MBMO-GA) olarak bahsedilmektedir.

#### 1.5. BASTIRILMAMIŞ SIRALAMALI GENETİK ALGORİTMA 2

Bastırılmamış Sıralamalı Genetik Algoritma 2 (NSGA-II), literatürde çok amaçlı problemlerin çözümünde etkili ve yaygın olarak kullanılan bir algoritmadır. Tek seferde bir ön bilgi gerektirmeden bir dizi baskın optimal çözümleri eder ve her bir çözüm için tekrar etmek zorunda kalmadan diğer çözümlere göre baskın sıralama yaklaşımını ve kalabalık mesafeyi hesaplayarak işlem ve hesaplama maliyetini en aza indirir [11][12]. Temel olarak elitizm ve çeşitlilik özellikleriyle ön plana çıkmakta ve yerleştirme problemlerinde iyi bir şekilde çalışmaktadır.

#### 1.6. LİTERATÜR TARAMASI

Sun B. ve diğerleri, [13] yaptıkları çalışmada dağıtılmış bir Çoklu Girişli Çoklu Çıkışlı (MIMO) radar ağının hedef çözülebilirliği açısından MIMO radarının çeşitlilik kapasitesini maksimize etmek için bir örnekleme zamanında antenlerin bir alt kümesini seçme problemini, bilgi geometrik bakış açısından ele almışlardır. Belirli bir hedef konumu kapsama durumlarına göre en yüksek kapsamayı en az maliyetle sağlayacak bir radar kümesini, seçtikleri kapsamlı alt küme arama yöntemleriyle elde etmeye çalışmışlar ve olumlu sonuçlar elde etmişlerdir.

Mirchandani ve diğeri, [6] yaptıkları çalışmada, radar sinyallerinin geçemeyeceği engellere sahip bir hedef yüzeyi kaplayacak gözetleme radarlarının yerini belirleme problemini ele almaktadır. Çalışmada, daha önceden ele alınmış iki problem değerlendirmişlerdir. İlk problem belirli bir alanın radarlar ile tam olarak kapsanabilmesi. İkinci problem n adet radar ve konumlarını belirleyerek belirli bir alan için en yüksek kapsama oranını elde etmektir. Bu iki problemi çözmek için Lagrange buluşsal yöntemi ve fethet ve böl ölçeklemeli iki aşamalı bir prosedür geliştirmişlerdir. Geliştirilen bu yöntemler ile bu iki problem çözebilmişlerdir.

Xie ve diğeri, [14] yaptıkları çalışmada bistatik radar çiftleri tarafından ortak olarak kapsanması gereken alanlarda kapsama performansını arttırmak için K-kapsama performansını kullanarak bir radar yerleştirme optimizasyonunu M-merkez modeli ve radar kesişim alanlarını formüle ederek gerçekleştirmişlerdir. Optimizasyonda çok boyutluluk sorunuyla başa çıkmak için iterative yöntemler olan K-sıralı Voronoi algoritmasını ve bir komşu arama yöntemi önermişlerdir. En optimal yerleştirmede %99.8'lik bir kapsama elde etmeyi başarmışlardır.

Y. Yang ve diğeri, [5] yaptıkları çalışmada kriter olarak radarların kapsama oranlarını ele alarak bunu arttırmayı hedeflemişlerdir. Problem yüksek hesaplama ve çok boyutluluk gibi zorlayıcı etkenlere sahip olduğundan iki bölümlü bir çözüm önermişlerdir. Bunlar amaç fonksiyonunu basitleştirmek için düşük karmaşıklık maliyetine sahip bir yöntem ve yüksek boyutlulukla başa çıkmak için parçacık sürü optimizasyonudur. Simülasyon sonuçları ile önerdikleri yaklaşımın hesaplama maliyetindeki avantajlarını doğrulamışlardır. Aynı zamanda çalışmada MIMO radarlarının yerleştirilmesinde optimal konumların tespiti için verimli bir çözüm yer almaktadır.

Abhinav Jauhri ve diğeri, [15] çalışmalarında belirli bir alana belirli radarları içeren bir sistemin oluşturulmasında zamandan tasarruf sağlamak ve sistemin performansını optimize etmeyi, radarların nerelere yerleştirileceğini bularak arttırmayı amaçlamışlardır. Bunun için bir dizi stokastik algoritmayı incelemişler ve simüle edilmiş tavlama ve evrimsel stratejinin, optimal yerleşimleri bulmada etkili olduklarını gözlemlemişlerdir.

S. Pillana ve diğeri [16] çalışmalarında kontrol panosu bileşen yerleşiminin çok amaçlı optimizasyonu için Pareto Benzetimli Tavlama Algoritmasına dayalı bir yaklaşım sunmuşlardır. Yaptıkları optimizasyonda bileşenler arası kablo uzunluğunu kısıltma ve kabindeki sıcaklığı düşürmeyi hedeflemişlerdir. Çalışma sonucunda optimale yakın sonuçlar elde ederek pano bileşenlerinin hedeflere uygun düzgün bir şekilde yerleştirilebildiğini göstermişlerdir.

Ghaffarinasab, ve diğeri, [17] çalışmalarında lojistik ve tedarik zinciri yönetiminde ortaya çıkan çok amaçlı LRP'yi ele almışlardır. Modelleme yapabilmek adına amaç fonksiyonları için matematiksel programlama formülasyonları oluşturmuş ve sunmuşlardır. Sistem maliyetlerini düşürmek ve müşteriye yapılan teslimatların sürelerini azaltmayı hedefledikleri bu çalışmada Benzetimli Tavlama tabanlı bir yöntem önermişler ve uygulamışlardır.

Vahid Hajipour ve diğeri, [18] yaptıkları çalışmalarında klasik kuyruk sistemini kullanan çok amaçlı ve çok katmanlı bir tesis yerleşim tahsisi modeli önermişlerdir. Tesis sayısını ve her tesisteki katman sayısını optimize etmeyi amaçladıkları bu çalışmada toplam seyahat ve bekleme sürelerini en aza indirme, tesis kurulum maliyetlerinin azaltılması, ve tesislerin boşa kalma sürelerinin azaltılması olarak üç amaç fonksiyonu ile çalışmışlardır. Çok amaçlı titreşim sönümlenme optimizasyonuna ve çok amaçlı uyum arama algoritmasına dayalı çok amaçlı bir meta sezgisel yaklaşım sunmuşlardır. Sunulan bu yaklaşımın etkisini ölçmek amacıyla MOSA ve NSGA-II ile karşılaştırma yapmışlardır. Belirledikleri bir takım kriterlere göre, özellikle de işlemci kaynak tüketimi göze alındığında önerdikleri yaklaşımın büyük problemlerde daha iyi çalıştığını göstermişlerdir.

İbrahim Demir ve diğeri, [19] yaptıkları çalışmada klasik hub yerleştirme probleminin bir türevi olan çok amaçlı kapasiteye sahip çoklu tahsisli hub yerleştirme problemini ele almışlardır. Bu problemi NSGA-II tabanlı ve MOSA tabanlı iki meta hüristik yaklaşım önermişlerdir. MOSA tabanlı yaklaşım için beş farklı komşuluk operatörü ve NSGA-II için probleme özel iki mutasyon operatörü geliştirmişlerdir. Algoritmaların davranışlarını ve performanslarını belirleyebilmek adına iki hazır veri seti kullanmışlar ve performans metriği olarak hiper hacim kullanmışlardır. Elde ettikleri sonuçlarda her iki algoritmanın da uygulanabilir çözümler bulunduğunu ancak

büyük ölçekteki problemlerde NSGA-II'nin, küçük çaplı problemlerde MOSA'nın daha iyi sonuçlar elde edebildiğini gözlemlemişlerdir.

Diogo R.M. Fernandes ve diğerleri, [20] yaptıkları çalışmada yük taşımacılığında sıklıkla karşılaşılan bir problem olan bir ürünün ara depolardan geçerek taşınmasının maliyetlerini, önerdikleri bir Genetik Algoritma (GA) ile en aza indirmeye ve bu sayede acentelerin talep ve kapasite kısıtlamalarını karşılamaya çalışmışlardır. Önerdikleri Genetik Algoritma'nın büyük ölçekli örneklerin çoğunda iyi çözümler üretebildiği gözlemlenirken en kötü durumlarda optimal çözümden %1.3 kadar uzak çözümler bulunduğu tespit edilmiştir.

Abdelhalim Hiassat ve diğerleri [21] yaptıkları çalışmada bozulabilir ürünler için bir konum-envanter-yönlendirme modelini geliştirdikleri bir genetik algoritma ile ele almışlardır. Model gerekli depoların sayısını ve yerini, her perakendecideki envanter seviyesini ve her aracın kat ettiği rotaları belirleyebilmektedir. Geliştirdikleri yaklaşım bu model için makul bir sürede yüksek kaliteli ve optimal çözüme yakın çözümler verebilmektedir.

Kim ve diğerleri, [22] çalışmalarında sosyal afetlerin neden olduğu tehditleri gözleme amacıyla kullanılan kızılötesi kameralar ve diğer gözetim ekipmanlarının optimum konumunu aramak için Genetik Algoritma (GA) kullanmışlardır. Çalışmalarında sensör özellikleri, Dijital Yükseklik Modeli (DEM) ve VITD haritalarını kullanmışlardır. Algoritma için oluşturulan bireylerin kromozomları, sensörlerin piksel konumları kullanılarak oluşturmuş ve gözetim bölgesindeki tespit olasılığıyla ilgili bir uygunluk fonksiyonu türetmişlerdir. 5 sensör ve 10 sensör için sonuçlar elde ederek karşılaştırmışlar ve elde ettikleri sonuçları literatürdeki bilgilerle karşılaştırdıklarında sensör sayısı arttıkça tespit edebilme özelliğinin doğrusal olarak artmadığını kendi problemleri için de doğrulamışlardır.

Rabbani ve diğerleri, [23] çalışmalarında atık toplama sorunu ekonomik ve toplumsal amaç işlevleri açısından ele alınmış olup depoları ve arıtma tesislerini bulmak ve müşterilere hizmet verme rotaları tasarlamak için NSGA-II kullanmışlardır. Yeni bir matematiksel model önerip hibrit bir NSGA-II oluşturmuşlardır. Çalışma sonucunda önerdikleri hibrit NSGA-II'nin tüm ayrıştırma yöntemlerinden daha iyi



performans sergilediğini ama çalışma süresinin bu yöntemlerden daha uzun olduğunu gözlemlemişlerdir.

Ebrahimi Zade ve diğerleri, [24] çalışmalarında tekli ve çoklu tahsisli hub maksimum kaplama problemleri için doğrusal olmayan çok amaçlı formüller ve doğrusallaştırılmış versiyonlar önermişlerdir. İki amaç fonksiyonu belirleyerek mutasyon ve çaprazlamanın sonuçlarına bağlı yeni bir operatör kullanan NSGA-II'nin bir versiyonunu geliştirmişlerdir. Geliştirilmiş olan NSGA-II'nin geleneksel yöntemlere göre daha iyi sonuçlar verebildiğini gözlemlemişlerdir.

Guo ve diğerleri, [25] çalışmalarında bir lojistik haritalama fonksiyonunu NSGA-II çalışma sürecine dahil ederek çok amaçlı optimizasyon problemlerini ele almak için genetik süreç tarafından üretilen çözümlere kaos ekleyerek yeni bir yaklaşım sunmuşlardır. Karmaşık fonksiyon optimizasyonunu çözebilen ve hesaplama verimliliği yüksek olan Kaos Optimizasyon Algoritması (COA) ile birlikte kullandıkları NSGA-II'yi Kaotik-NSGA-II olarak adlandırıp normal NSGA-II ile karşılaştırmışlardır. Karşılaştırma sonucunda önerilen bu algoritmanın global optimizasyonu aramada iyi bir performansa sahip olduğunu gözlemlemişlerdir.

Mingjie Song ve Dongmei Chen, [26] çalışmalarında Çok Amaçlı Arazi Tahsisi (MOLA) problemini yama tabanlı, kenar büyütme ve azaltma, komşuluk ve kısıtlama yönlendirme kurallarını entegre ederek çözmek için geliştirilen bir NSGA-II önermişlerdir. Klasik NSGA-II ile yapılan karşılaştırmalarda bu önerilen algoritmanın optimal çözümlere daha yakın çözümler ürettiği gözlemlenmiştir.

Yahui W. ve diğerleri, [27] çalışmalarında çok amaçlı esnek atölye çizgeleme problemi çalışmasına ve geleneksel Genetik Algoritma'nın (GA) öğrenilmesine dayalı olarak optimize etmek için bir NSGA önermişlerdir. Optimizasyon sürecinde bir alt operatör seçmek için baskın olmayan hızlı sıralama operatörü ve rekabet operatörü kullanmış ve bu sayede algoritmanın hesaplama verimliliğini ve optimize etme yeteneğini geliştirmişlerdir. Elde edilen simülasyon sonuçlarını başka algoritmalarla karşılaştırarak doğrulamışlardır. Geliştirilen algoritmanın optimal çözümü daha hızlı elde edebileceğini ve daha etkili olduğunu gözlemlemişlerdir.

E. Yakıcı, [28] çalışmasında toplam radar kapsamını ve kritik radar kapsamını maksimize etme amacıyla operasyon bölgesinde deniz platformlarının konumlandırılmasını sorununu çözmek için NSGA-II ve SMS-EMODA algoritmalarını karşılaştırmıştır. NSGA-II'nin performansının hem yakınsama hem de çeşitlilik açısından daha iyi olduğu sonucuna varmıştır.

X. Liu ve diğerleri, [29] çalışmalarında NSGA-II'nin çok radarlı ortak planlama görevlerinin özelliklerini inceleyerek çok radarlı ortak görev planlamasını hızla tamamlayabileceğini göstermişlerdir. Elde ettikleri sonuçlarda tek tip veri dağıtımı, yüksek yakınsama, yüksek sağlamlık ve daha iyi optimizasyon etkisi elde edilebildiğini gözlemlemişlerdir.

## İKİNCİ BÖLÜM

### 2. PROBLEMİN TANIMLANMASI

Problem, üretilen çözümlerin kalitelerinin amaç fonksiyonlarının çıktıklarıyla yeteri kadar doğru bir şekilde ölçülmesi ve üç boyutlu ortamdan ötürü yapılan maliyetli kapsama hesaplamalarından ötürü zorlayıcı bir problemdir. Bu çalışmada ele alındığı açıdan ve hedeflenen başarımlar doğrultusunda değerlendirildiğinde optimizasyon problemi olarak çözümlenebilmektedir. Bu doğrultuda problemi çözmeye başlamadan önce uygun ortamın ve gerekli hesaplamaların doğru tanımlanmış olması gerekir. Bu çalışmada, problemin çözümünde gerçek yükseklik haritası verileri gerekli işlemler yapılarak kullanılmıştır. Harita, tanımlı boyuttaki hücrelere bölünerek ve her hücrenin temsil ettiği bölgedeki en yüksek nokta, ilgili hücrenin yüksekliği olarak kabul edilerek bölge oluşturulabilmekte ve ardından tarama hesaplamaları esnasında üçüncü boyut eklenerek hava sahası oluşturulabilmektedir. Radarların yerleştirileceği konumlar bu hücreler ve yükseklik bilgileri kullanılarak belirlenebilmekte ve radarların tarama alanlarındaki engel yükseklikleri tespit etmek kolaylaşmaktadır.

#### 2.1. HARİTA VERİLERİNİN İŞLENMESİ

##### 2.1.1. Mekik Radar Topografya Misyonu (SRTM) Nedir?

Mekik Radar Topografya Misyonu (SRTM), Ulusal Jeo-uzamsal İstihbarat Ajansı (NGA) ile Ulusal Havacılık ve Uzay Dairesi (NASA) arasında ortak bir uluslararası projedir. Bu projenin amacı, Dünya'nın kara yüzeyinin %80'i (60° kuzey ve 56° güney enlemleri arasındaki tüm kara alanları) için dijital topografik veriler üretmektir. Üretilen veriler enlem/boylam ızgarası yapısında sistematik olarak tutulmaktadır. Bu çalışmalarda elde edilen topoğrafik yükseklik verilerinin doğruluk oranı %90'dır. Bu radar sistemi, Dünya yüzeyinin yaklaşık %80'nin şimdiye kadar oluşturulmuş en doğru ve eksiksiz topografik haritasını çıkaracak verileri toplamaktadır. Yükseklik bilgisi interferometri yöntemi ve X-Band Synthetic

Aperture Radar (X-SAR) radar sinyalleri gönderilerek yapılmaktadır. Mekik ölçümleri sırasında mekik günde 16 kez Dünya etrafını 7.5 km/sn hızla dönerek 176 turda ölçümlerini gerçekleştirmektedir [Url-1]. Bu ölçümler ile toplanarak oluşturulan yükseklik harita verileri Dijital Harita Yükseklik Verileri (DTED) olarak adlandırılmaktadır.

### **2.1.2. İnterferometri Yöntemi ile Yükseklik Hesaplanması**

İnterferometri, iki grup radar sinyalinin birleştirilmesiyle oluşturulan girişim desenlerinin inceleme tekniğidir. SRTM, interferometri tekniğinde radardan yeryüzüne sinyaller gönderilir. Gönderilen bu sinyalin yeryüzüne çarpıp döndüğünde radar üzerindeki bir anten ve radara bir direk ile yanal olarak yaklaşık 60 metre uzatılmış ikinci bir anten yardımıyla iki veri seti olarak toplanır. Ana anten ve dıştan eklenmeli anten arasındaki mesafe sabit olduğundan, tek fark iki antenin yeryüzüne olan mesafesidir. İki anten arasındaki mesafe ve yansıyan radar dalgası sinyallerindeki farklılıklar hakkındaki bilgiler kullanılarak, Dünya yüzeyinin doğru yüksekliği hesaplanabilir. Bu mesafe farkı kullanılarak arazi yükseklik değeri hesaplanmaktadır [Url-1].

### **2.1.3. Dijital Harita Yükseklik Verileri (DTED)**

DTED, Ulusal Havacılık ve Uzay Dairesi (NASA) ve Ulusal Jeo-uzamsal İstihbarat Ajansı (NGA) tarafından Mekik Radar Topografya Misyonu (SRTM) kapsamında yeryüzü yükseklik bilgisinin elde edilmesi için yapılan çalışmalar sonucu olarak elde edilen yükseklik verilerine ait veri formatıdır. Bu misyon kapsamında ilk kez küresel arazi yükseklik verileri toplanılmaya çalışılmıştır. SRTM misyonu kapsamında elde edilen topografik arazi yükseklik verilerinin gerekli alanlarda aynı yöntem ve tekniklerle çözümlenip kullanılabilmesi için tanımlanan spesifikasyonlar DTED veri formatı olarak sunulmaktadır. Bu tez kapsamında yükseklik harita verilerinden yararlanıp taranacak bölgeyi oluşturmak amacıyla Almanya Havacılık ve Uzay Merkezi (DLR) tarafından sağlanan SRTM/X-SAR DTED verisi kullanılmıştır.

DTED [30] verileri .dt uzantılı DTED dosyalarında byte olarak tutulmaktadır. Genel olarak her bir DTED dosyası ardışık enlem ve boylamların arasında kalan alana ait arazi yükseklik bilgilerini kapsamaktadır. Ardışık enlem ve boylam arasında kalan

alan belirli çözünürlükteki yükseklik hücrelerinden oluşan bir ızgara sisteminden oluşur. Her bir yükseklik hücresi, o yükseklik hücresinin kapsadığı alana ait en yüksek noktanın yükseklik bilgisini içerir. DTED verisi farklı çözünürlüklerde olabilir. DTED verileri sınıflandırılırken yükseklik hücrelerinin çözünürlük bilgisine göre sınıflandırılırlar. Her bir yükseklik hücresine ait veri çözünürlüğü ve DTED veri seviyesi arasındaki ilişki aşağıdaki gibidir:

- Yükseklik Hücresi 30 metre(yatay) x 30 metre(dikey), DTED-2.
- Yükseklik Hücresi 90 metre(yatay) x 90metre(dikey), DTED-1.
- Yükseklik Hücresi 900 metre(yatay) x 900 metre(dikey), DTED-0.

Bu tez kapsamında kullanılan veri DTED-1 verisidir ve verideki her bir hücre  $90m^2$ 'lik bir alandaki en yüksek noktayı temsil etmektedir. Çalışma kapsamında bu hücrelerin çözünürlüğü düşürülerek  $2700m^2$ 'lik hücrelere dönüştürülmüşlerdir.

Yatay konum, Dünya Jeodezi Sistemine (WGS) göre her bir DTED dosyasının güneybatı köşesi orjin kabul edilerek hesaplanmaktadır. Yükselti, güneyden kuzeye doğru DTED dosyasında tanımlanmış olan enlem ve boylam aralık değerlerine göre hesaplanan yükseklik hücresine konumlandırılır. Her bir DTED dosyasına ait ızgara sisteminin orjin noktası güneybatı köşesi kabul edilir ve arazi yükseltilerine ait yükseklik değeri güneyden kuzeye ve batıdan doğuya olacak şekilde ızgara sisteminde tutulur. Arazi yükseklik bilgisi metre cinsinden ifade edilir.

DTED-1 verisine göre kabul edilen varsayımlar şöyledir:

- Yüksekliklerin konumları, bir matris içindeki satır ve sütunların kesişimleri(yükseklik hücreleri) ile tanımlanır.
- Yükseklik değeri işaretli ikilik tamsayı sistemine 2 bayt olarak tutulur.
- Negatif değerlere tamlama yapılmaz.
- Yükseklik değeri işaretli 2 bayt ile ifade edildiğinden  $\pm 32.767$  metre arasında teorik bir yükseklik aralığındadır.
- Arazi yükseklik değerleri +9.000 metreyi veya -12.000 metreyi geçmez.
- Boş olan değerler -32,767 değerine sahiptir.

Bir DTED dosyası genel olarak şu bilgileri içerir:

- Metadata bilgisi
- Veri seti bilgisi
- Doğruluk bilgisi
- Yükseklik bilgisi

#### **2.1.4. Yükseklik Harita Verileri Ne İçin Kullanılabilir?**

SRTM kapsamında elde edilen DTED verileri askeri, sivil ve bilimsel kullanıcı topluluklarının ihtiyaçlarını karşılamak için kullanılabilir. Bunun yanısıra diğer kullanım alanları arasında su drenajı modellemesi, uçuş simülasyonları, baz istasyonları için pozisyon tespiti, navigasyon güvenliği, hava trafik kontrol sistemleri ve diğer havacılık alanları yer almaktadır.

Arazinin şekli ve yüksekliği hakkında kesin bilgi gerektiren hemen hemen tüm çalışmalar için bu verilerin kullanılması fayda sağlayabilir. Sel kontrolü, erezyon risk tespiti, ağaçlandırma, yanardağ izleme, deprem araştırması ve buzul hareketi gibi çalışmalar içinde önemli bir veri kaynağıdır.

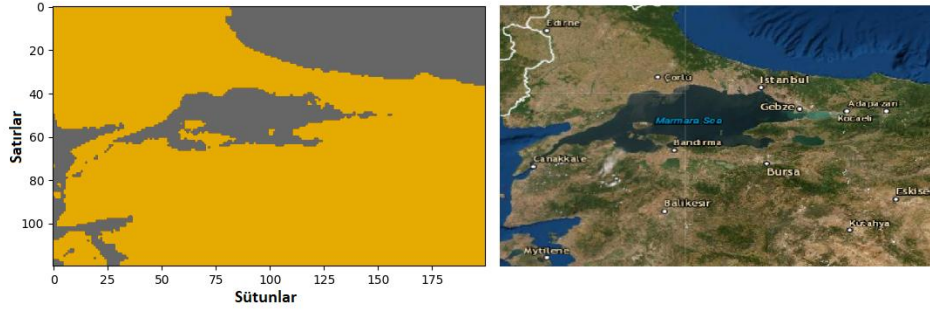
#### **2.1.5. Geospatial Veri Soyutlama Kütüphanesi**

Geospatial Veri Soyutlama Kütüphanesi (GDAL), raster ve vektör coğrafi veri formatlarını okumak ve yazmak için kullanılan açık kaynaklı bir kütüphanedir [Url-5]. Son derece yaygın bir kütüphanedir ve bir coğrafi bilgi içeren çalışmada kullanılmaktadır. Bir çok veri formatını desteklemektedir. Bu tez kapsamında bu kütüphane DTED-1 verilerini okumak ve veriyle çalışmada kullanılacak yapıları desteklemek amacıyla kullanılmıştır.

#### **2.1.6. Çalışma Bölgesinin Oluşturulması**

Bu çalışma kapsamında ilk olarak hangi bölge üzerinde çalışılacağına karar verilmiştir. Karar verilmiş olan bölge 41-39 enlemleri ve 26-30 boylamları arasındaki kalan bölgedir. Bölge dikdörtgen şeklindedir ve bir kenarı 3 enlem uzunluğunda iken diğer kenarı 5 boylam uzunluğundadır. Bu bölge çoğunlukla Türkiye'nin Marmara Bölgesi'ni kapsamaktadır. Bölgenin bir kısmında deniz bulunduğu ve

yükseklikler deniz seviyesi ve üzerinden ölçüldüğünden en küçük yükseklik 0 metre olarak ele alınmıştır. Bunun dışında en küçük yüksekliğe sahip kara parçasının 1 metre, en büyük yüksekliğe sahip kara parçası ise 2535 metre olduğu gözlemlenmiştir. Bu bölgenin seçilmesindeki amaç bölgenin çok çeşitliliğe sahip olmasıdır. Bölgede düzlemsel bölgeler, kıyı şeritleri, göller, denizler, adalar, tepeler ve dağlar bulunmaktadır. Bir radarın karşılaşılabileceği kolaylıklara ve zorluklara sahip ideal bir bölgedir. Bu bölge DTED-1 verisi olarak okunduktan sonra 90 metre kenar uzunluğuna sahip karesel hücreler güncellenmiş olup 2700 metre kenar uzunluğuna sahip karesel hücrelere çevrilmişlerdir. Böylece harita yükseklik çözünürlüğü düşmüş ve her bir hücre temsil ettiği  $2700m^2$ 'lik alandaki en yüksek noktanın yükseklik bilgisini tutmaktadır. Çözünürlüğü bu şekilde düşürmek hesaplamalarda kolaylıklar sağlanmıştır. Yükseklik haritası 2700 metre kenar uzunluğuna sahip hücrelere sahip olacak şekilde güncellendiğinde 120x200 boyutunda bir matris elde edilmiştir. Bu bölgede 0 metre olan yüksekliklere, yani denizlerin ve göllerin üzerine radar yerleştirilemeyeceği kısıtı en önemli kısıtımızdır, yani ilgili bölgede kara parçası olan her yere radar yerleştirilebileceği varsayılmaktadır. Yerleştirilecek olan noktanın ne olduğu, üzerinde ne bulunduğu, askeri alan, özel mülkiyet veya yasaklı alan olup olmadığı ve radar yerleşimi için müsait olup olmadığı ve bunun gibi unsurlar değerlendirilmemiştir. Sadece kara parçası olmasının yeterli olup, radar yerleşimi için uygun olduğu varsayılmıştır. Bölgenin matrisi analiz edildiğinde radar yerleştirilebilecek 180102 nokta olduğu tespit edilmiştir, kalan noktalar radar yerleştirilemeyecek alanlardır. Bu bölgenin daha iyi gözlemlenebilmesi açısından gerçek harita resmi [Url-4] ve bu haritanın 2 boyutlu matrisinin gösterimi Şekil 2.1'de verilmiştir. Çözünürlük düşerken hücreler birleştiğinden yer yer görsel birleşimler olmuş bulunmaktadır, örneğin İstanbul boğazının görünmemesi gibi.

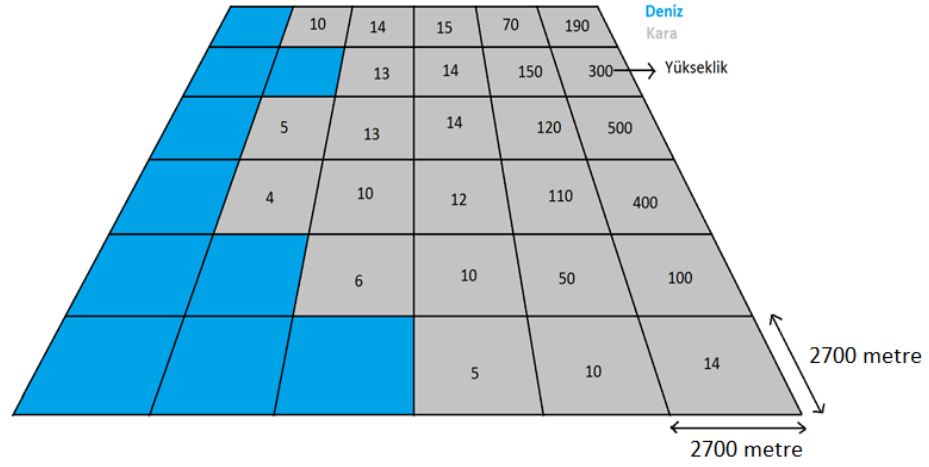


**Şekil 2.1:** Çalışma için seçilen bölgenin çözünürlüğü düşürülmüş matrisinin görselleştirilmesi ve yanında gerçek harita görüntüsü [Url-4].

Bahsedildiği üzere problemin çözümünde kullanılan hava sahası genel olarak bir kenar uzunluğu 2700 metre olan, kapsadığı alan  $2700m^2$  kare şeklindeki hücelere bölünmüştür. Radar tarama hesaplamaları sırasında her bir hücrenin deniz seviyesinden itibaren üzerine her birinin yüksekliği 1000 metre olan 20 hücre eklenmiş ve hesaplamalar sırasında bölgenin dikdörtgen bir prizmaymış gibi 3 boyutlu halinin taranması gerçekleştirilmiştir. Kısaca haritadaki tek bir hücre  $2700m^2$  taban alanına ve 1000 metre yüksekliğe sahip bir dikdörtgen prizmadır ve bu şekilde gökyüzüne doğru 20 hücre olarak uzadığı kabul edilerek işlemler gerçekleştirilmiştir.

Bahsedilen bu işlemlerin yapılabilmesi için öncelikle GDAL kütüphanesi kullanılarak çalışmada kullanılacak enlem ve boylamlar arasında kalan haritanın DTED-1 formatındaki dosyaları programa yüklenmiş olup 90 metre olan hücre kenarları 2700 metreye çıkarılarak çözünürlük düşürülmüştür ve çalışmada gerekli veri özellikleri alınacak şekilde ayrıştırma işlemi yapılarak belleğe yüklenmiştir. Bu işlemler sadece verinin yüklenmesi ve belleğe alınması için yapılmıştır. Geri kalan işlemler bir sonraki bölümde anlatılmıştır. Belleğe taşınan 2 boyutlu haritanın temsili bir gösterimi Şekil 2.2’de verilmiştir. Havasahası ekleme işlemi, radar kapsama hesaplamalarında gerçekleştirilmektedir.





Şekil 2.2: Çalışmada kullanılan 2 boyutlu haritanın temsili bir gösterimi.

## 2.2. RADARLAR VE KAPSAMA HESAPLAMALARI

Radarlar, gönderdikleri elektromanyetik enerjinin geri yansımalarını kullanarak belirli cisimleri tespit eden elektronik ekipmanlardır. Radarlar bazı durumlarda tespit ettikleri cisimlerin yönünü, mesafesini, yüksekliğini ve hızını ölçebilme kabiliyetine sahip olabilirler. Radarların kullandığı elektromanyetik enerji frekansı karanlıktan etkilenmemez, sis ve bulutlara nüfuz eder. Genellikle radar sistemleri gözle takibi yapılamayan uçakların, gemilerin, cisimlerin ve bir takım engellerin konumunun belirlenmesinde aktif olarak kullanılmaktadırlar [Url-2].

Radarların elektronik çalışma prensipleri, ses dalgası yansıma prensipleri ile benzerdir. Bir kişi, bir yöne doğru bağırdığında sesi eğer yankı yapıyorsa, ses o yönde bir şeye çarparak geri dönmüş demektir. Eğer ses hızını biliyorsak, sesin döndüğü süreyi hesaplayarak yankıyı oluşturan engelin mesafesini hesap etmek mümkün olmaktadır. Radarlar, elektromanyetik enerji darbelerini de bu mantık çerçevesinde kullanmaktadır. Radarlar bir radyo frekansı yayar ve bunun küçük bir kısmı radara geri döner ve bu dönen kısım yankı olarak adlandırılır. Radarlar, yankıyı yansıtan şeyin yönünü ve mesafesini bu yankıyı kullanarak hesaplayabilirler [Url-2].

### 2.2.1. En-Route Radarı Çalışma Prensibi

En-Route radarı, kontrol edilecek hava sahasındaki uçuşların kontrolü için geliştirilmiş bir gözlem radarıdır. Radarın tam bir Türkçe adı olmasa da seyir ya da gözetim radarı olarak tanımlanabilir. Genellikle havaalanı kontrol bölgesi dışında

kalan hava sahasındaki uçuşların trafik kontrolü için idealdir. Bu radarların menzilleri 450 kilometreye kadar çıkabilmektedir ve hava sahasının 2 boyutlu olarak elde ettiği görüntüsünü 3 boyutlu hale getirebilmek için bir İkincil Gözlem Radarı (SSR) ile birlikte çalışır. Genellikle ayarlanan menzillerine erişebilmeleri ve en iyi performansı gösterebilmeleri için yüksek yerlere ve yüksek kulelere kurulumları gerçekleştirilir. Örneğin Almanya'da, ülkenin kuzeybatısı ile kuzey denizini izlemek için her birinin menzili 270 kilometre olan 6 adet En-Route radar kullanılmaktadır [Url-2].

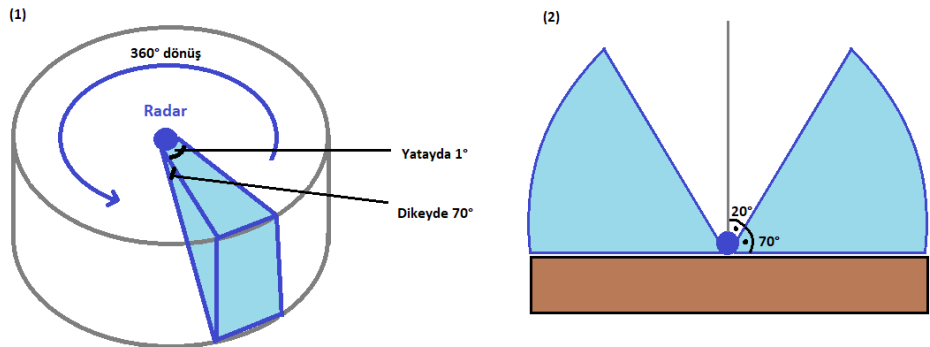
Bu çalışmada kullanılacak olan radar tipi En-Route radardır ve sinyal işleme konularına girilmediği için radarın değişken özellikleri dikkate alınmamakta, sadece genel radar çalışma prensibi dikkate alınarak radarın yazılımsal modellenmesi yapılmaktadır. Bu radar için dikkate alınan ve çalışmada kullanılan özellikler kapsama mesafesi ve kendi etrafında dönerek alan tarama özelliğidir. Kullanılacak radar modelinin tek tipte bir radar olacağı, sadece kendi etrafında dönerek etrafı tarayacağı, tarama mesafesinin sabit olduğu ve tarama işlemlerinde radarın görüş açısının sabit olduğu, aşağı yukarı hareketlerin olmayacağı varsayılmaktadır. Tarama sinyallerinin ulaşabileceği en uzak mesafenin de hep aynı olduğu, sinyal dalgaları çalışma prensibi ve özelliklerine göre değişmeyeceği varsayılmaktadır. Oluşturulan radar modelinin kapsama mesafesi 400 kilometre olarak belirlenmiştir. Bu kısıtlar çerçevesinde işlemler gerçekleştirilmektedir.

### **2.2.2. Radarın Modellenmesi ve Tarama Hesaplamalarında Kullanılan Formüller**

Radarın modellenmesinde En-Route radar örnek alınmıştır. Şekil 2.3'te radarın tarama prensipleri açıklanmaktadır. Radarın kendi etrafında dönüp 360° tarama özelliği bu tez kapsamındaki çalışmanın vazgeçilmez parçasıdır. En-Route radarların görüş açısı yatayda 1°'lik bir görüşe sahiptir, dikeyde ise 0°'nin yere paralel olduğu varsayılarak, yukarı doğru 70° bir görüş açısı vardır. Şekil 2.3'te 1 numaralı radar tarama örneğinde bu özellik görselleştirilmiştir. Şekilde 2 numaralı radar tarama örneğinde 2 boyutlu gösterimi mevcuttur. Çalışmada kullanılan radarın görüş açısının direkt yer yüzüne paralel olacağı varsayılmıştır ve koyulduğu noktadan 70° yukarısını tarayabilmektedir. Yatayda bir görüş genişliğine sahip olmamakla beraber 70° taramalarla kendi etrafında dönerek tarama işlemi gerçekleştirebilmektedir. En

yukarısını ise  $20^\circ$  görememektedir. Kendi etrafında döndüğü için üst kısmında sessizlik konisi denilen bir boşluk oluşmaktadır ve toplamda  $40^\circ$  olarak tanımlanmaktadır. Kısaca radarın modellenmesinde Şekil 2.3'e göre belirlenen kriterler şu şekildedir:

- Radar kendi etrafında  $360^\circ$  dönerek etrafı tarayabilir.
- Kapsama menzili dahilinde olmak üzere yere paralel olarak tarama yaptığı varsayıldığında yukarı doğru  $70^\circ$ 'ye kadar tarama yapabilir.
- Sessizlik konisini gözlemleyemez.
- Radarın görüş yönü aşağı ve yukarı doğru hareket ettirilmeyektir.
- Radarın yerleştirildiği konum yüksek bile olsa yere paralel bir tarama gerçekleştirecek, aşağı tarafları göremeyecek, en düşük derecesi yere paralel olmak üzere  $0^\circ$  olarak kabul edilecektir.
- Radar saat yönünde hareket ettirilerek tarama işlemleri gerçekleştirilecektir.
- Çok büyük olmayan bölgelerde çalışıldığı için taramalarda dünyanın elips yapısı dikkate alınmayacak, sadece enlem ve boylam ile ilgili hesaplamalarda dikkate alınacaktır.

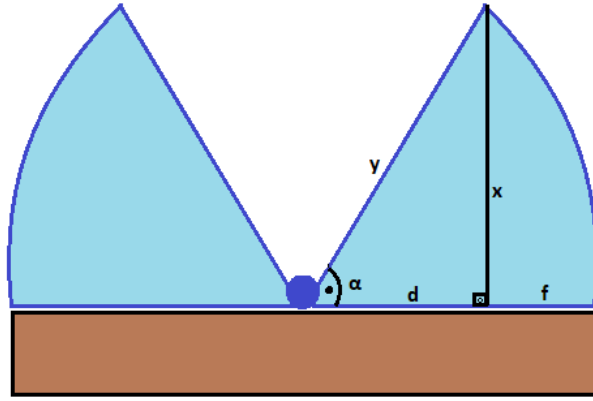


Şekil 2.3: En-Route radarın tarama prensibi.

Radar tarama hesaplamaları yapılırken radar yere paralel olacak şekilde  $0^\circ$ 'den  $70^\circ$ 'ye kadar tarama yapabilecektir. Yazılımsal olarak bunun modellenmesi yapılırken radarın iki adet ışın gönderdiği ve bu ışınların biri radarın göreceği en alt nokta olan

0° ile, ikincisinin de radarın göreceği en üst nokta olan 70° ile göndereceği belirlenmiştir. Bu ışınlar radar menzili kadar gidecek ve menzil bittiğinde sonlanacaklardır. Bu iki ışının sonlandığı noktalar arasındaki mesafe, radarın dikey olarak görebildiği mesafedir. Denklem 2.1’de menzile ve açığa dayalı bu yükseklik hesaplamalarında kullanılan formül, Şekil 2.4’te ise bu formülün kullanım şekline yardımcı çizim verilmiştir.

$$\begin{aligned}
 \alpha &= \text{radyan türünde açı} \\
 d &= \text{radar alt ışın menzili} \\
 y &= \text{radar üst ışın menzili} \\
 x &= \tan(\alpha) \cdot d \\
 y &= \sqrt{x^2 + d^2}
 \end{aligned}
 \tag{2.1}$$



Şekil 2.4: Açısız hesaplamalara yardımcı görsel.

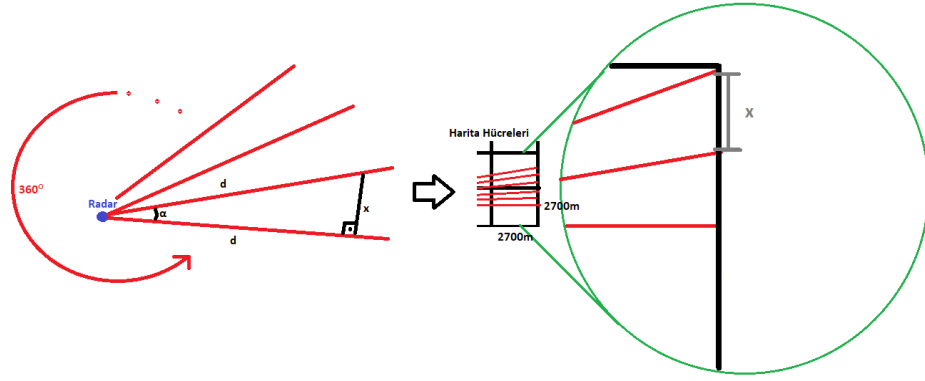
Radarı kendi etrafında 360° döndürülerek belirli mesafeler için gönderdiği ışınlar arasında kalan yüksekliğin, o noktalardaki yükseklik boyunca kapsama yapılan kapsama olduğu bilinmektedir. Ancak 400 kilometre büyük bir mesafedir ve radarı kendi etrafında döndürme hassasiyetinin ne olacağı önemli bir konudur. Radarı kendi etrafında 360° döndürme işlemi adım adım ve her seferinde küçük döndürmelerle yapılmalıdır ancak her seferinde yapılacak dönüş açısının ne olacağı önceden tespit edilmelidir. Denklem 2.2’de bu hesaplamaların nasıl yapılacağına formülü verilmiştir ve Şekil 2.5’de denklemin kullanılabilmesi için için görselleştirmeye yer verilmiştir. Amaç iki dönüş hareketi arasında alt ışınlar ile taranan en uzak noktalardaki mesafenin en kötü ihtimalle harita hücrelerinin bir kenar uzunluğundan küçük olmasıdır. Bu

durumda kabul edilen kısıtlar çerçevesinde  $\alpha$  ne olursa  $x$  2700'den küçük olur sorusuna cevap aranmalıdır.

$$\alpha = \text{radyan türünden açı} \quad (2.2)$$

$$d = \text{radar alt ışın menzili}$$

$$x = \tan(\alpha) \cdot d$$



Şekil 2.5: Radar dönüşleri arasında yapılması gereken açı değişimi denklemini ifade eden görsel.

Yapılan analizlere göre  $\alpha$  0.38 olarak seçildiğinde radarın her bir dönüşü  $0.38^\circ$  olarak gerçekleşecektir. Bu durumda iki dönüş hareketi sırasında 400 kilometre menzilli alt ışınların en uç noktaları arasında yaklaşık olarak 2653 metre olacaktır ve bu da 2700 metre olarak ayarlanan harita hücre kenarlarından düşüktür. Bu sayede iki dönüş hareketi esnasında en uç ışın noktalarının arasında tespit edilmekten kaçan bir hücre olmayacağı varsayılmıştır. Bu durum için düşünülen kısıta göre eğer bir hücre uç noktalara yeteri kadar yakınsa, hücre alanının yarısından fazlasının zaten gözlem noktasına gireceğidir. Eğer gözlem uç noktaları hücrenin merkezine yeteri kadar yakın değilse zaten kapsanmasının gerekli olmadığı varsayılmıştır.

Kendi etrafında belirlenmiş bir açısal değere göre dönen ve döndükçe harita hücrelerini tarayabilecek bir radarın matris üzerine yerleştirip belirlenmiş bir açıya göre ileriye tespit etmesini sağlamak için başka formüllere de ihtiyaç vardır. Bunun için yapılan yazılımsal modellemede radar, iki boyutlu haritada herhangi bir hücreye yerleştirildiğinde her bir dönüşünde ilk olarak bir gözlem ışını göndermektedir. Bu

ışının üzerinden geçtiği hücreler sırasıyla tespit edilerek radarın her bir hücre için alt ve üst ışınlarını gönderip belirlenmiş hücrelerin hangilerinin hangi yüksekliğe kadar gözlemlenebildikleri bulunmaktadır. Bulunduktan sonra her bir hücrenin gözlemlenebildiği en küçük ve en büyük yükseklikler arasındaki mesafe bilgileri ile radarın kapsama haritası 3 boyutlu olarak oluşturulup hangi hücreleri kapsadığıyla ilgili bilgiler, ilgili radara özel bir dosyaya daha sonra metasezgisel algoritmalarca kullanılmak üzere kaydedilmektedir. Algoritmaların amaç fonksiyonları doğrultusunda sürekli olarak radarların kapsama hesaplamalarının yapılması oldukça yüksek maliyetli bir işlem olduğu için metasezgisel algoritmalar çalıştırılırken her seferinde ve eğer denk gelirse aynı radarlar için tarama hesaplamalarının tekrar ve tekrar yapılmaması için bu dosyalar kullanılmakta ve direkt olarak radarın kapsadığı hücreler bulunmaktadır. Bu yöntem, maliyetli işlem tekrarlarının önüne geçmekte ve algoritma çalışırılık zamanlarını çok büyük bir ölçüde düşürmektedir.

Radarı 2 boyutlu harita üzerinde herhangi bir hücreye yerleştirip, kendi etrafında saat yönünde döndürüp haritayı tarama işlemini yapmasını sağlamak için bahsedilen ilk sıralı hücre tespit ışını gönderimi belirli formüller ile yapılabilmektedir. Direkt olarak radarın gördüğü hücreleri tespit etmek zor bir iş olduğundan öncelikle radarın üzerinde bulunduğu hücrenin enlem ve boylam bilgisi kullanılarak, belirlenen açı bilgisiyle belirlenen mesafe kadar ilerisinin enlem boylam bilgisinin elde edilmesi gerekmektedir. Elde edilen enlem boylam bilgilerinin hangi hücreye ait olduğu tespit edilerek radarın gördüğü ve tespit ışını dahilinde birbiri ardına gelen hücreler sıralı bir şekilde tespit edilebilmekte ve radarın ilgili açı üzerindeki hücreleri görebildiği tespit edilebilmektedir. Bundan sonra yapılması gereken şey, hücrelerin üzerindeki hava sahasının hangi yükseklikler arasında kapsanabildiği ve eğer varsa birbiri ardına gelen hücrelerin birbirlerini engellemeleri durumunun tespit edilerek radar taramasının doğru açısal değişimlerle yapılmasıdır. Verilen bir enlem ve boylamdan başlayarak belirli bir açı ile belirli bir mesafe sonrasındaki noktanın enlem ve boylam bilgisi [Url-6] Denklem 2.3 ile bulunabilmektedir.

$$\varphi_1 = \text{başlangıç noktası enlem bilgisi} \quad (2.3)$$

$$\lambda_1 = \text{başlangıç noktası boylam bilgisi}$$

$$\varphi_2 = \text{bir sonraki nokta enlem bilgisi}$$

$$\lambda_2 = \text{bir sonraki nokta boylam bilgisi}$$

$$\theta = \text{kuzeyden saat yönündeki açı}$$

$$R = \text{Dünya'nın yarı çap bilgisi}$$

$$D = \text{noktalar arası mesafe}$$

$$\delta = \text{açısal mesafe}$$

$$\delta = D/R$$

$$\varphi_2 = \text{asin}(\sin\varphi_1 \cdot \cos\delta + \cos\varphi_1 \cdot \sin\delta \cdot \cos\theta)$$

$$\lambda_2 = \text{atan2}(\sin\theta \cdot \sin\delta \cdot \cos\varphi_1, \cos\delta - \sin\varphi_1 \cdot \sin\varphi_2)$$

### 2.2.3. Harita Üzerinde Radar Tarama İşlemleri

Radar tarama işlemlerinin bütün adımları Algoritma 2.1'de verilmiştir. İlgili tarama adımları metasezgisel algoritmaların çalıştırılmasından bağımsız olarak gerçekleştirilmekte ve bu algoritmaların kullanacağı radar kapsama hesaplamalarını her bir radar için yaparak, radarın gördüğü hücrelerdeki en düşük ve en büyük yükseklikler arasında kalan hava sahası hücrelerini dosyalara kaydetmektedir.

Kapsama hesaplamaları adımlarında ilk olarak gerekli parametre atamaları yapılmakta, ardından enlem ve boylam bilgileri ile gerekli olan harita verileri, HaritayıYükle fonksiyonu ile bir veri yapısına yüklenmektedir. Bu veri yapısı aracılığı ile harita kullanım kolaylıkları elde edilebilmektedir. Bu veri yapısından hesaplamalarda lazım olan bazı bilgiler alınarak değişkenlere atanmaktadır. Sonrasında haritadaki her bir satır ve her bir sütunu gezebilmek için döngüler başlatılmaktadır. Döngüdeki ilgili sütunun hücre bilgileri bir fonksiyon aracılığı ile radarNoktası veri yapısına yüklenmektedir. Bu veri yapısı da hücre kullanım kolaylığı sağlayacak bir noktadır ve içerisinde işaret ettiği hücrenin satır, sütun, enlem, boylam, yükseklik gibi bir takım bilgiler barındırır. Aynı zamanda boş değişkenlere de sahiptir, bu değişkenler ileriki adımlarda kullanım kolaylığı için gerekli olmaktadır. Radar tarama hesaplamalarında radarın hangi hücrelerin ne kadarını kapsadığıyla ilgili

bilgileri tutmak amacıyla çok boyutlu boş bir liste oluşturulmakta ve radarın kendi etrafında dönerek tarama hareketlerini gerçekleştirmesi işlemlerini sağlaması amacıyla 0°'den 360°'ye kadar dönüşü temsil edecek olan bir döngü başlatılmaktadır. Asıl hesaplamalar bu döngünün içerisinde sıralı bir şekilde gerçekleşmektedir. Dönüş döngüsünün ilk bölümünde boş bir nokta listesi oluşturulduktan sonra radarın her bir dönüşündeki dönüş açısını ifade eden  $d$  sayacı radyan formatına ve tespit edilecek noktaların arasında olması beklenen mesafe bilgisi metreden kilometre cinsine dönüştürülmektedir. Ardından radarın görebileceği varsayılan nokta sayısı kadar bir döngüye girilmekte ve bu döngünün her bir adımında mesafe bilgisi, tespit edilecek noktanın sıra bilgisi ile ilişkili bir şekilde artmakta ve radardan `NoktaTespitEt` fonksiyonu ile radardan hesaplanan mesafe kadar uzaklıktaki noktanın bilgileri Denklem 2.3 kullanılarak bulunmaktadır. Bu fonksiyon, radarın bulunduğu hücrenin enlem, boylam bilgilerini, dünyanın yarıçap bilgisini (6371.01), mesafeyi ve radarın baktığı yöndeki açıyı (radyan formatında  $d$  değişkenini) ilgili denklem ile işleyerek, verilen mesafe kadar ileride ve belirtilen yöndeki noktanın enlem ve boylam bilgilerini bulmakta, ardından bu enlem ve boylam bilgisine sahip hücreyi tespit ederek nokta veri yapısı olarak döndürmektedir. Radardan, baktığı yönün ilgili mesafesinin sonundaki nokta tespit edildikten sonra noktanın harita sınırları içinde olup olmadığı kontrol edilmektedir. Bulunan nokta, harita sınırları dışında olduğu takdirde fonksiyon ilgili enlem ve boylam bilgilerine sahip hücreyi bulamayacak ve boş bir veri yapısı döndürecektir. Bu durumda nokta veri yapısının bilgileri kontrol edilerek harita sınırları içinde olup olmadığı kontrol edilebilmekte ve harita sınırları içerisindeyse daha sonra yeniden işlenmek üzere `noktaListesi` listesine eklenmektedir. Eğer nokta, harita sınırları içerisinde değilse, ilgili açı ile görülebilecek diğer noktaların da artan mesafede olduğu bilindiğinden, onların da harita sınırları içerisinde olamayacağı göz önüne alınarak ilgili döngü sonlandırılmaktadır. İkinci bölümde, ilgili yönde tespit edilen noktaları içeren `noktaListesi` listesinden tekrar eden noktalar varsa silinmektedir. Tespit edilecek noktalar arasında artan mesafe, birinci bölümde bir hücre kenar uzunluğunun yarısı kadar olduğu için, aynı noktanın tekrar etmesi mümkün olmaktadır, bu sebeple de tekrar eden noktaların sadece bir tanesi ile hesaplama yapılması yeterlidir. Bu bölümde yapılan hesaplamalar, tespit edilmiş olan noktaların radara olan uzaklıkları hesaplanarak, radarın bu noktaları hangi



yükseklikler arasında tarayabildiği bilgisinin bulunmasıdır. Bunu yapabilmek için öncelikle noktaListesi'ndeki her bir noktayı gezecek bir döngü oluşturulup ilk olarak noktanın temsil ettiği hücrenin satır ve sütun numaraları kullanılarak noktanın radara olan birim uzunluğu bulunur. Ardından bu birim uzunluk, hücrelerin kenar uzunluğu ile çarpılarak radar ile nokta arasındaki gerçek uzaklık metre cinsinden bulunur. Eğer noktanın radara olan uzaklığı, radar kapsama menzili sınırları içindeyse nokta veri yapısına, radara olan uzaklığı bilgisi eklenmektedir. Eğer noktanın radara olan uzaklığı, radar kapsama mesafesi dışındaysa, noktaların sıralı olduğu bilindiğinden, bu noktanın ve ardındaki diğer noktaların radar tarafından kapsanamayacağı sonucuna varılır ve noktaListesi'ndeki ilgili nokta da dahil olmak üzere geri kalan bütün noktalar silinerek hesaplamaları yapılmamaktadır. Üçüncü bölümde, radarın kapsayabildiği ve radara olan uzaklıkları metre cinsinden bilinen sıralı noktalara sahip noktaListesi ile kapsama hesaplamaları Denklem 2.1 kullanılarak gerçekleştirilmektedir. İlk olarak radarın bulunduğu hücre yüksekliği, kullanılmak üzere bir değişkene atılmaktadır, ardından radarın alt ışını olarak kabul edilen, görebileceği en alt noktanın tespiti için altAçı bilgisi tanımlanmaktadır. İlgili denklemdeki  $d$  ışınının, başlangıçta direkt olarak yere paralel olarak gönderileceği varsayıldığından açısı  $0^\circ$  kabul edilmekte ve altAçı ile temsil edilmektedir. Yine aynı denklemdeki  $y$  ışını da, görülebilecek olan en yüksek açı olarak kabul edilen  $70^\circ$ , üstAçı değişkenine tanımlanmaktadır. Atamalardan sonra radarın kapsadığı noktaları barındıran noktaListesi'ndeki her bir elemanı gezecek bir döngüye girilir. Döngüde ilk olarak radarın göndereceği ve daha önceki bölümlerde alt ışın olarak isimlendirilen, radarın ilgili hücre üzerinde görebileceği en düşük yükseklik hesaplanır. Bu görülen yüksekliğin, ilgili hücrenin yüksekliğinden düşük olması durumunda hücredeki yükseklik bu ışına engel oluşturmakta ve radar alt ışını bu hücrenin ardını görememektedir. Bu sebeple altAçı bilgisi, alt ışının ilgili hücrenin yüksekliğinden daha yüksek ilk noktayı görene kadar her adımda  $0.5^\circ$  arttırılarak hesaplamalar tekrarlanmaktadır. Bunun sonunda radarın, ilgili hücre üzerinde görebileceği en küçük yükseklik bulunmaktadır. Noktaların sıralı olduğu ve ardı sıra geldikleri bilindiğinden, herhangi bir engelleme gerçekleştiğinde altAçı bilgisi güncellendiğinden kayıplar en aza indirilmekte ve güncellenen açıyla bundan sonraki döngü iterasyonlarında sıralı noktalar doğru hesaplamalarla kontrol edilmeye devam edilebilmektedir. Radarın ilgili nokta üzerinde gördüğü en küçük yükseklik tespit

edildikten sonra bu sefer de görebileceği en büyük yükseklik hesaplamaları yapılmaktadır. En büyük yükseklik hesabı yapılırken Denklem 2.1'deki  $y$  ışını olarak adlandırılan bilgi ve üstAçı bilgisi kullanılmaktadır. Bunun yanı sıra,  $y$  radarın gönderebileceği en üst ışın olarak kabul edildiğinden bunun da radar kapsama menziline aşmaması beklenmektedir. Yine aynı denklemdaki bilgiler kullanılarak  $y$  uzunluğu hesaplanmakta ve radarın kapsama menziline aşması durumunda üstAçı bilgisi her seferinde  $0.5^\circ$  düşürülerek en büyük yükseklik ve  $y$  hesaplamaları tekrar edilmektedir. Bu sayede  $y$ , hiç bir zaman radar menziline uzun olamayacaktır. Nokta ile radar arasındaki mesafe arttıkça,  $y$  bir süre sonra yükselişten düşüşe geçecek ve böyle devam edecektir. Yine üstAçı bilgisi güncellenerek, her bir sıralı nokta için güncel hali kullanılmaya devam edecektir. En büyük yükseklik bulduktan sonra üstAçı'nın altAçı bilgisine eşit veya daha düşük olup olmadığı kontrol edilmekte, küçük olması durumunda hesaplama döngüsünden çıkılmaktadır. Yine aynı kontrol radarın gördüğü en büyük yüksekliğin, ilgili hücrenin yüksekliğinden küçük olup olmadığıdır. Eğer küçükse radarın en üst taraması da engellenmiş olmaktadır, bu durumda yine hesaplama döngüsü sonlandırılır. Eğer hesaplama döngüsü sonlandırılmamışsa radarın ilgili nokta üzerinde gördüğü en küçük ve en büyük yükseklik bilgileri, noktanın veri yapısına bilgi olarak eklenmektedir. Dördüncü bölümde, yine noktaListesi'ndeki her bir nokta gezilerek, noktaların radar tarafından kapsanıp kapsanmadığı, en küçük ve en büyük yükseklik atamaları kontrol edilerek yapılmaktadır. Eğer bu atamalar yoksa radarın bu noktayı göremediği varsayılmaktadır ve nokta hesaba katılmamaktadır. Eğer nokta kapsanmışsa, geçici olarak oluşturulmuş olan radarınGördüğüYükseklikler listesine, hücrenin satır ve sütun numarası kullanılarak hücrenin hangi noktadan hangi noktaya kadar kapsandığı, en küçük yükseklik ve en büyük yükseklik bilgilerinin kaydedilmesi aracılığıyla saklanmaktadır. Radar, saat yönünde döndüğünden her bir hücre için en güncel yükseklik bilgileri, o hücre için hesaplanmış en son yükseklik bilgileri olmaktadır. Radar çok küçük bir açıyla döndüğünden aynı hücre için hesaplamalar birden fazla kere olabilmektedir. Bir hücre, radarın bir dönüşünde başka bir hücre tarafından engellenmiş ve hiç kapsanmamış olabilir ama bir sonraki dönüşünde de radarın bakış açısından kaynaklı olarak engel teşkil etmeyen başka bir hücrenin ardından gelebilir ve engellenmeyebilir. Aynı şekilde ilk hesaplamada hiç engellenmeyen bir hücre de

son hesaplamada engellenmiş olabilmektedir. Hesaplamalar bu kısıtlar çerçevesinde doğru kabul edilmektedir. En sonunda radar kendi etrafında 360° dönüşünü tamamladıysa, geçici olarak oluşturulan yükseklik listesindeki her bir hücre bilgisi, ilgili radar için oluşturulan bir dosyaya yazılır. Hesaplamalar dosyaya direkt yazılmamakla birlikte öncelikle basit işlemlerden geçmektedirler. Her bir hücre için, hücrenin gökyüzüne doğru 1000 metre kenar yüksekliğine sahip 20 hava hücresine bölündüğü varsayılmakta ve ilgili hücrenin kapsandığı en küçük yükseklik ve en büyük yükseklik bilgileri kullanılarak, bu 20 hava hücresinden kaçının kapsanabildiği hesaplanmaktadır. Bununla ilgili örnekler ve kısıtlar şu şekildedir: Hücre tamamen havadan oluşuyorsa ve radar dikeyde ilgili hücrenin yarısından fazlasını görüyorsa hücre kapsanmış kabul edilir, aksi halde kapsanmamış kabul edilir. Örneğin bir radarın gördüğü en düşük yükseklik 1450 metre ve gördüğü en yüksek yükseklik 19550 metre olsun. Bu radar 1450 metreden 2000 metreye kadar ikinci hava hücresinin yarısından fazlasını kapsadığı için (550 metre) ikinci hücreyi kapsadığı kabul edilir, aynı şekilde 19000 metreden 19550 metreye kadar yirminci hava hücresinin yarısından fazlasını (550 metre) kapsadığı için yirminci hücreyi kapsadığı kabul edilir. Asıl duruma bakıldığında radar ikinci ve yirminci hücreler de dahil olmak üzere aralarında kalan hücreleri kapsamaktadır, yani 19 hava hücresini kapsamıştır. Eğer radarın gördüğü en düşük yükseklik 1550 metre ve gördüğü en yüksek yükseklik 19450 metre olsaydı, ikinci ve yirminci hücrelerin yarısından azını gördüğü için onları kapsamadığı kabul edilerek üçüncü ve ondokuzuncu hücre de dahil olmak üzere 17 hücreyi kapsayabilecekti. Eğer ilgili hücrelerde kara parçası varsa kısıtlar şu şekilde kabul edilmektedir: Eğer kara parçası yüksekliği dikeyde hücrenin yarısından az ise hücre radarlar tarafından kapsanabilecek bir hava hücresi olarak kabul edilip bir önceki kısıtlara göre işlem görmektedir. Eğer kara parçası yüksekliği dikeyde hücre yüksekliğinin yarısını aşıyorsa bu hücre kara parçası olarak kabul edilerek radarlar tarafından kapsanamamaktadır.

Verilen bu kısıtlar çerçevesinde her bir harita hücresi için, hava sahasını temsil etmesi amacıyla hücrenin yukarısına yerleştirilen 20 hava hücresinin kaç tanesinin kapsanabilen hücre olduğu ve radarlar tarafından hangilerinin kapsanabildiği hesaplanmaktadır. Alan kapsama hesaplamaları yapılırken sadece radarlar tarafından

kapsanabilme özelliğine sahip hücreler hesaba katılmakta, kapsanma özelliği bulunmayan hücreler kapsama oranları hesaplanırken dikkate alınmamaktadır, örneğin bir radarın kapsadığı hücrelerin sayısı bulunduktan sonra alandaki toplam kapsanabilme özelliğine sahip hücre sayısına bölünerek kapsama oranı bulunabilmektedir, bölme işleminde kapsanma özelliği olmayan ve kara parçası olarak kabul edilen hücreler kullanılmamaktadır. Bütün bu kısıtlar doğrultusunda yapılan hesaplamalar ile radarın kapsama dosyasına kapsadığı her bir hücre için ilgili bilgiler  $(x,y,z)$  olarak yazılmaktadır. Verilen formatta x hücre satır numarasını, y hücre sütun numarası ve z yukarı doğru kaçınıcı hücrenin kapsadığı bilgisini içermektedir. Buradan da anlaşılacağı üzere tek bir harita hücresi için 20'ye kadar varabilen kapsanma bilgileri kaydedilebilmektedir. Bu sayede de haritanın üç boyutlu modellenmesi için gerekli kayıtlar ilgili radar için oluşturulmuş olmaktadır.

Algoritmalar çalışırken radarların kapsama oranları hesaplandığında bu dosyalar kullanılmaktadır. İlgili çözümde kaç radar varsa, her bir radarın kapsadığı hücre bilgileri bu dosyalardan alınmakta ve küme birleşimi oluşturulmaktadır. Bu sayede ilgili çözümdeki radarların, ilgili haritadaki hava sahasında kaç adet hücreyi kapsadığı belirlenebilmekte ve bu hücre sayısı toplam hücre sayısına bölündüğünde toplam kapsama oranı elde edilebilmektedir.

```

hücreKenarUzunluğu = 2700
radarMenzili = 400000 // metre cinsinden radar menzili
radarınGöreceğiNoktaSayısı = radarMenzili / (hücreKenarUzunluğu / 2)
haritaVerisi = HaritayıYükle()
satırSayısı = haritaVerisi.satırSayısı
sütunSayısı = haritaVerisi.sütunSayısı
haritaYükseklikMatrisi = haritaVerisi.yükseklikler

i = 0'dan satırSayısı'na kadar dön:
| j = 0'dan sütunSayısı'na kadar dön:
| | radarNoktası = hücreBilgileriniYükle(i, j)
| | radarınGördüğüYükseklikler = []
| | d = 0.0
| | d <= 360.0 olduğu sürece dön:
| | | // Bölüm 1
| | | noktaListesi = []
| | | radyanD = radyanaDönüştür(d)
| | | kmCinsindenMesafe = (hücreKenarUzunluğu / 2) / 1000
| | | n = 1'den (radarınGöreceğiNoktaSayısı + 1)'e kadar dön:
| | | | mesafe = kmCinsindenMesafe * n
| | | | nokta = radardanNoktaTespitEt(radarNoktası, radyanD, mesafe)
| | | | Eğer nokta harita sınırları içindeyse:
| | | | | noktaListesi.ekle(nokta)
| | | | Değilse:
| | | | | döngüden çık
| | | | d = d + 0.38
| | |
| | | // Bölüm 2
| | | tekrarEdenNoktalarıSil(noktaListesi)
| | | noktaListesi'ndeki her bir eleman (nokta) için dön:
| | | | satırUzaklığı = nokta.satır - radarNoktası.satır
| | | | sütunUzaklığı = nokta.sütun - radarNoktası.sütun
| | | | radaraOlanUzaklık =  $\sqrt{\text{satırUzaklığı}^2 + \text{sütunUzaklığı}^2}$ 
| | | | radaraOlanUzaklık = radaraOlanUzaklık * hücreKenarUzunluğu
| | | | Eğer radaraOlanUzaklık <= radarMenzili ise:
| | | | | nokta.radaraOlanUzaklık = radaraOlanUzaklık
| | | | Değilse:

```

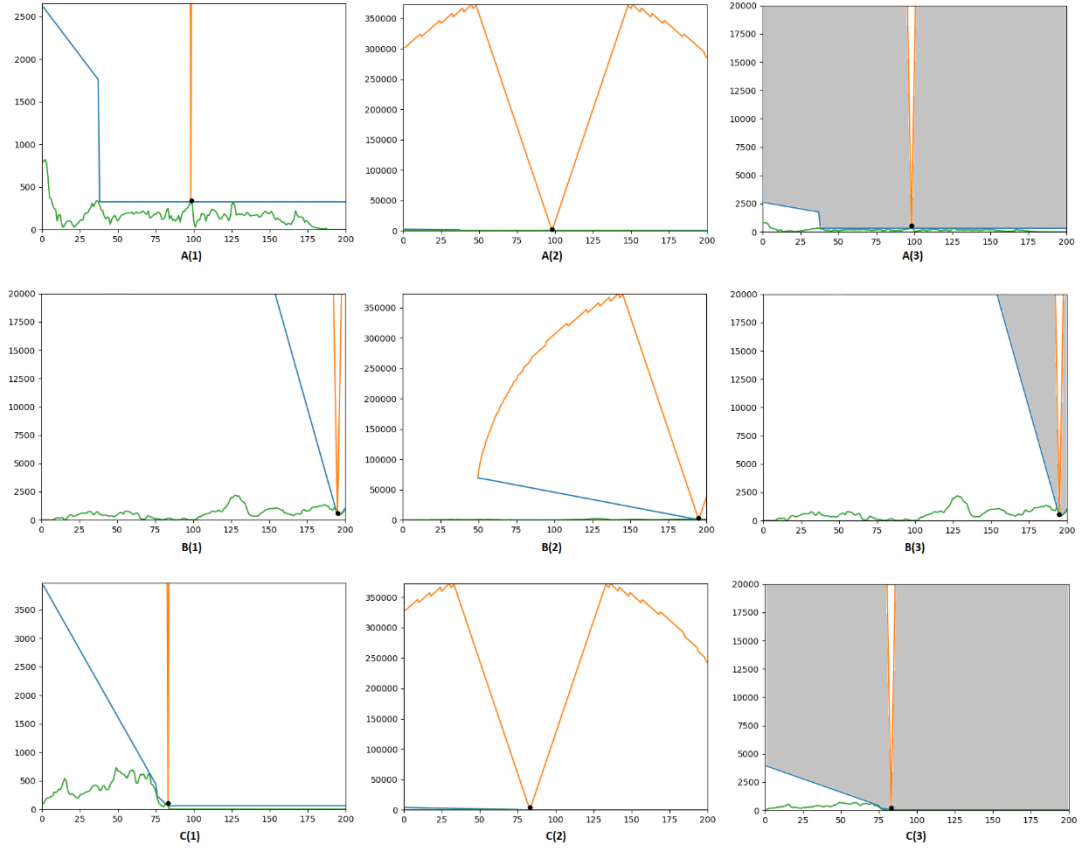
```

| | | | | kalanElemanlarıSil(noktaListesi, nokta)
| | | | | döngüden çık
| | |
| | | // Bölüm 3
| | | radarYüksekligi = radarNoktası.hücreYüksekligi
| | | altAçı = 0.0
| | | üstAçı = 70.0
| | | noktaListesi'ndeki her bir eleman (nokta) için dön:
| | | | enKüçükYükseklik = enKüçükYükseklikBul(
| | | | | radarYüksekligi, altAçı, nokta.radaraOlanUzaklık)
| | | | enKüçükYükseklik < nokta.hücreYüksekligi olduğu sürece dön:
| | | | | altAçı = altAçı + 0.5
| | | | | enKüçükYükseklik = enKüçükYükseklikBul(
| | | | | | radarYüksekligi, altAçı, nokta.radaraOlanUzaklık)
| | | | | enBüyükYükseklik = enBüyükYükseklikBul(
| | | | | | radarYüksekligi, üstAçı, nokta.radaraOlanUzaklık)
| | | | | yUzunluğu =  $\sqrt{\textit{nokta}.\textit{radaraOlanUzaklık}^2 + \textit{enBüyükYükseklik}^2}$ 
| | | | | yUzunluğu >= radarMenzili olduğu sürece dön:
| | | | | | üstAçı = üstAçı - 0.5
| | | | | | enBüyükYükseklik = enBüyükYükseklikBul(
| | | | | | | radarYüksekligi, üstAçı, nokta.radaraOlanUzaklık)
| | | | | | yUzunluğu =  $\sqrt{\textit{nokta}.\textit{radaraOlanUzaklık}^2 + \textit{enBüyükYükseklik}^2}$ 
| | | | | Eğer üstAçı <= altAçı ise:
| | | | | | döngüden çık
| | | | | Eğer enBüyükYükseklik < nokta.hücreYüksekligi ise:
| | | | | | döngüden çık
| | | | | | nokta.enKüçükYükseklik = enKüçükYükseklik
| | | | | | nokta.enBüyükYükseklik = enBüyükYükseklik
| | | | |
| | | | | // Bölüm 4
| | | | | noktaListesi'ndeki her bir eleman (nokta) için dön:
| | | | | | Eğer nokta radar tarafından kapsanmışsa:
| | | | | | | radarınGördüğüYükseklikler[nokta.satır][nokta.sütun] =
| | | | | | | | [nokta.enKüçükYükseklik, nokta.enBüyükYükseklik ]
| | | | | | radarKapsamasınıKaydet(radarNoktası, radarınGördüğüYükseklikler)

```

**Algoritma 2.1:** Radar tarama hesaplamaları için algoritma çalışma adımları.

Radar tarama hesaplamaları sonucunda elde edilen radar kapsama oranı dosyaları kullanılarak radar kapsama oranları metasezgisel algoritma çalışma adımlarında uygunluk fonksiyonu hesaplamaları ile bulunmakta ve uygunluk fonksiyonu çıktısı olarak değerlendirilmektedir. Bütün bu taramalar sonucunda elde edilen bilgilere göre her bir radarın tekil olarak kapsama oranı bulunabilmektedir. Bu haritada tek başına en yüksek kapsama oranını veren radar, %94.72'lik bir kapsama oranına, en düşük kapsama oranını veren radar ise %8.61'lik bir kapsama oranına sahiptir. Şekil 2.6'da bu iki radarın, haritanın tek bir satırındaki yüksekliklere olan kapsama oranları 2 boyutlu olarak gösterilmektedir. Şekilde A görüntüleri, en yüksek kapsamayı sağlayan radara ait bilgileri sunmaktadır. A(1), radarın haritadaki 34. satır için kapsadığı en düşük yükseklikleri, A(2) ise radarın haritadaki 34. satır için kapsadığı en büyük yükseklikleri vermektedir. Radarın bulunduğu sütun numarası 97'dir. Şekilde B görüntüleri, en düşük kapsama oranına sahip radara ait bilgileri sunmaktadır. B(1), radarın haritadaki 75. satır için kapsadığı en düşük yükseklikleri, B(2) ise radarın haritadaki 75. satır için kapsadığı en büyük yükseklikleri vermektedir. Radarın bulunduğu sütun numarası 194'tür. Şekilde C görüntüleri, kıyıya koyulan bir radarın deniz üzerinde kapsadığı en küçük yükseklikleri daha iyi görüntüleyebilmek adına ilgili radar ile ilgili bilgileri vermektedir. C(1), radarın haritadaki 4. satır için kapsadığı en düşük yükseklikleri, C(2) ise radarın haritadaki 4. satır için kapsadığı en büyük yükseklikleri vermektedir. Radarın bulunduğu sütun numarası 82'dir. Şekildeki 1, 2 ve 3 numaralı görüntülerde turuncu çizgiler, radarların en büyük kapsama yüksekliklerini, mavi çizgiler en küçük kapsama yüksekliklerini, yeşil çizgiler ilgili satırdaki yeryüzü yüksekliklerini vermektedir. Siyah noktalar radarları temsil etmektedirler. Numarası 3 olan görüntülerde gri alanlar radarlar tarafından kapsanan bölgeleri temsil etmektedir. Dikkat edilmesi gereken önemli bir konu ise ilgili şeklin 2 boyutlu görüntüler içerdiğiidir. Farklı satırlar ve yönler için bu kapsama görüntüleri değişebilmektedir.

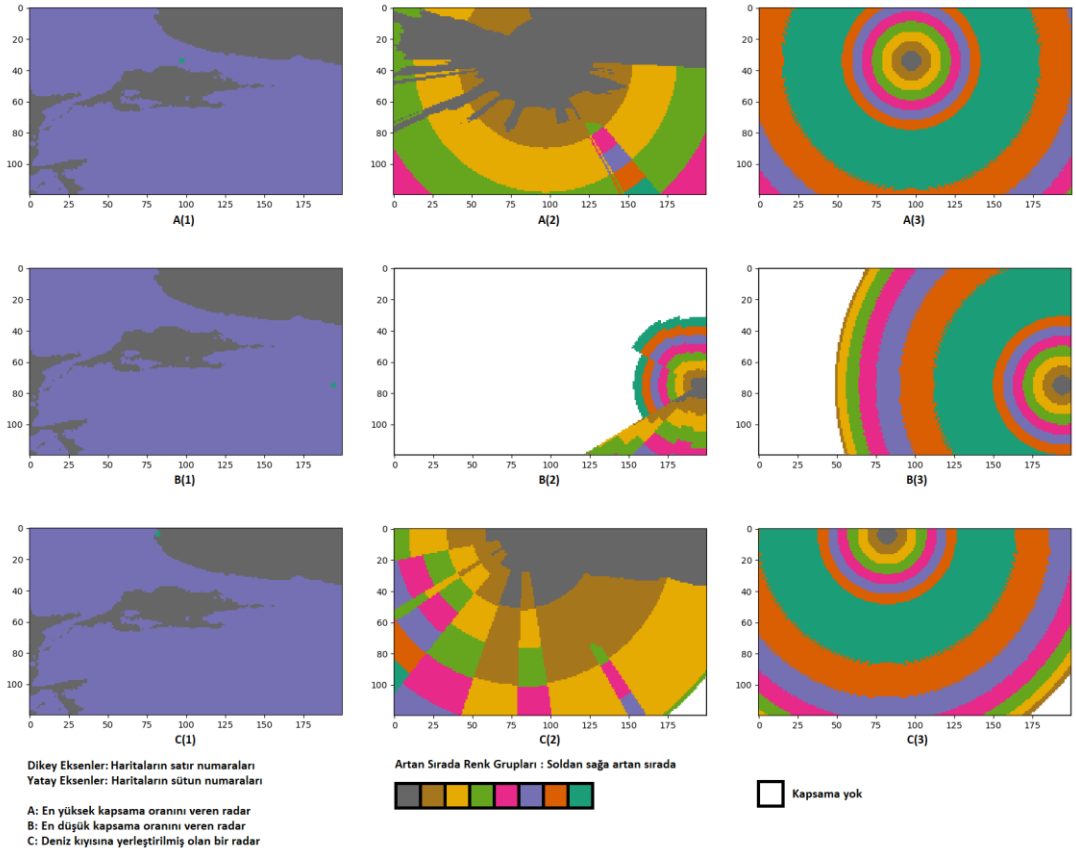


**Şekil 2.6:** En yüksek ve en düşük kapsama oranlarını veren radarlar ile deniz kıyısına yerleştirilmiş bir radarın harita üzerindeki tek bir satırda görebildiği en küçük ve en büyük yükseklikler.

Yukarıda Şekil 2.6, bahsi geçen radarların tek bir satır için gösterimlerini içermektedir. Radarların gördüğü en küçük ve en büyük yüksekliklerin, haritanın yukarisından bakıldığındaki görüntüleri Şekil 2.7’de yer almaktadır. Radarlar değişken sayıda yükseklik gördükleri için, görüntüler oluşturulurken yükseklikler gruplandırılmış ve ona göre renklendirilmiştir, bundan ötürü değişik renkler görebilmekteyiz. Görüntülerde beyaz renkteki alanların kapsama bölgesi dışında oldukları, koyu gri renkli alanların gruplanmış en düşük yükseklikler ve nil yeşili alanların gruplanmış en yüksek yükseklikler olduğu kabul edilmektedir. Renkler arasındaki ilişki, ilgili şekilde küçükten büyüğe doğru artan sırada verilmiştir. Bu renk sıralaması A, B ve C görüntüleri için geçerlidir ancak her bir görüntüde renkli alanların ifade ettiği yükseklikler değişebilmektedir. Örneğin bir görüntüde gri alanlardaki en büyük yükseklik değeri 1000 iken, başka bir görüntüde 2000 olabilir, burada önemli olan şey yükseklik değişimlerinin incelenmesidir. Şekil 2.7’de A görüntüleri en yüksek kapsama oranını veren radar ile ilgili bilgileri, B görüntüleri en düşük kapsama



oranını veren radar ile ilgili bilgileri, C görüntüleri de deniz kıyısına yerleştirilmiş bir radar ile ilgili bilgileri vermektedir. Numaralandırmalarda 1 numaralar radarın haritada hangi konumda olduğu bilgisini içermektedir ve konumlar turkuaz renkli nokta şeklinde verilmiştir. Radarların gördüğü en küçük yüksekliğin 20000 metre ile sınırlandırılmış hali ise 2 numaralı görüntülerde verilmiştir ve en önemli görüntü de budur. Bu görüntülerde kapsanan en küçük yükseklikler verildiği ve 20000 metreyi geçmediği için radarın aslında etkin olduğu en önemli alanlar vurgulanmaktadır ancak yine de tüm detayları içermemektedir. Radarların gördüğü en büyük yükseklikler ise 3 numaralı görüntülerde verilmekte, hesaplamalarda kullandığımız en büyük yüksekliklerin tümünü ve çok daha fazlasını içermektedir. Şekil 2.6'da görülebildiği üzere bu radarlar gökyüzüne doğru 350000 metre tarayabilmekte ancak biz bunun sadece 20000 metreye kadarını bu tez çalışmaları kapsamında kullanmaktayız. Bu sebeple 3 numaralı görüntüler 20000 metreden çok daha yüksek kapsamaları da göstermektedir. Bu 3 numaralı görüntülerdeki amaç, radarın etkin bir şekilde nasıl bir kapsama yaptığının görselleştirilmesidir.



**Şekil 2.7:** En yüksek ve en düşük kapsama oranlarını veren radarlar ile deniz kıyısına yerleştirilmiş bir radarın konumları ve kapsamlarıyla ilgili görselleştirmeler.

#### 2.2.4. Amaç Fonksiyonları

Bu tez kapsamında metasezgisel algoritmalar tarafından problemin çözümü için kullanılan iki amaç fonksiyonu vardır. Bu amaç fonksiyonlarından biri çözümlerdeki radar sayısını hesap ederken diğeri çözümdeki radarların yerleştirildikleri konumlara göre üç boyutlu alandaki hücrelerin yüzde kaçını kapsadıklarının bulmaktır. Kaç adet radarın olduğunun bulunması işlemi gayet kolaydır; 0 ve 1'lerden oluşan çözümlerdeki 1'lerin sayısı tespit edilerek radar sayısı bulunabilmektedir. Radarların kapsama oranlarının bulunmasında ise öncelikle bir çözümdeki radarlar tespit edilir, radarların üç boyutlu haritada hangi hücreleri kapsadıkları belirlenir. Bireydeki veya çözümdeki bütün radarların kapsadığı hücreler bulununca birleşimleri alınır ve toplam kapsanan hücre sayısı elde edilir. Bulunan bu hücre sayısı, üç boyutlu alanın radarlar tarafından kapsanma özelliğine sahip toplam hücre sayısına bölününce toplam kapsama oranı bulunmaktır.

## ÜÇÜNCÜ BÖLÜM

### 3. METODOLOJİ

Bu tezde ele alınan en az sayıda radar ile en yüksek kapsama oranını elde etme problemi MOSA, MBMO-GA ve NSGA-II algoritmaları ile farklı yöntemler denenerek ele alınmıştır. Denenen yöntemler arasında farklı genetik değişim ve mutasyonlar kullanılmakla birlikte radarların hücelere yerleşim ihtimallerini de barındıran bir takım yöntemler denenmiştir.

#### 3.1. ALGORİTMALARDA KULLANILAN YÖNTEMLER

##### 3.1.1. Çözüm Gösterimleri

Her bir algoritmadaki bireylerin ve çözümlerin temsili aynı şekilde gerçekleştirilmiştir. Temsili çözüm ve bireyler, tek bir tanımlamayla çözüm olarak adlandırılmıştır. Çözümler, tamamen işlem yapılacak alan haritasını temsil edecekleri için çözümlerin boyutu haritada radar yerleştirmeye uygun hücre sayısı kadar olmaktadır ve bu sayı 18102'dir. Buradan da anlaşılacağı üzere her bir çözümün her bir elemanı, harita verisindeki her bir hücreyi temsil etmekte ve radar olup olmama durumunu sayısal değerler ile gösterebilmektedir. Elemanların alabileceği değerler 0 ve 1 olarak değişebilmekte, bu da çözümümüzü ikili bir çözüm yapmaktadır. Çözümdeki herhangi bir elemanın 1 olması, elemanın temsil ettiği hücreye radar konulduğunu göstermekte, 0 olması da radar koyulmadığını göstermekte ve işlemler buna göre gerçekleştirilmektedir. Bütün çözümler ilk defa oluşturulurken çözümün hangi elemanlarına 1 sayısının koyulacağı tek bir kritere bağlanmış ve her bir algoritma için uygulanmıştır, uygulanmayan algoritmalar ve yöntemlerde bu durum belirtilmiştir. Bu uygulama adımları Algoritma 3.1'de verilmiştir.

```

çözüm = [0,0,0,...,n] // her bir elemanı 0 olan boş bir çözüm
0'dan n/2'ye kadar dön:
| radarYerleştirmeİhtimali = 0-1 arası rasgele sayı üret
| Eğer radarYerleştirmeİhtimali < 0.0015 ise:
| | rasgeleİndis = 0-n arası rasgele sayı üret
| | çözüm[rasgeleİndis] = 1
çözüm'ü kaydet

```

**Algoritma 3.1:** Çözüm oluşturma adımları.

Bu çözüm üretme kriteri standart olarak belirlenmiş ve bütün algoritma çalıştırmalarında aksi belirtilmedikçe çözüm üretmek için kullanılmıştır.

### 3.1.2. Uygunluk Fonksiyonları

Bütün algoritmalarda kullanılan uygunluk fonksiyonları aynıdır. Bu uygunluk fonksiyonlarından biri, çözümde kaç adet radar olduğunu tespit etmektedir ve bunu yaparken çözümdeki 1'lerin sayısını saymaktadır. Radar sayısının en az olması istenildiğinden bu bir minimizasyon fonksiyonudur, yani ne kadar düşük bir çıktı verirse o kadar iyi olduğu anlamına gelmektedir. Fonksiyon 0-n (n çözüm boyutudur) arasında bir sonuç verebilmektedir. Bu amaç fonksiyonuna ait çalışma adımları Algoritma 3.2'de verilmiştir.

```

çözüm = [] // 0 ve 1'lerden oluşan bir liste
radarSayısı = 0
çözüm'deki her bir elemanı (eleman) gez:
| Eğer eleman'ın değeri 1 ise:
| | radarSayısı += 1
radarSayısı'nı amaç fonksiyonu hesaplaması olarak kaydet

```

**Algoritma 3.2:** Radar sayısını hesaplayan uygunluk fonksiyonunun çalışma adımları.

İkinci uygunluk fonksiyonu ise hangi hücrelerde 1 olduğunu tespit eder ve o hücrelere yerleştirilmiş radarların kayıtlı kapsama dosyalarından kapsadıkları hücreleri tespit ederek çözümdeki bütün radarlar için alan kapsamalarının birleşimini alır. Ardından birleşim sonucundaki hücre sayısını alana bölerek kapsama oranını tespit eder. Buradaki amaç radarların en yüksek kapsama oranını bulmak olduğundan bu bir maksimizasyon fonksiyonudur, yani fonksiyonun çıktısı ne kadar yüksekse o

kadar iyidir. Fonksiyon 0-100 arasında bir sonuç verebilmektedir. Bu uygunluk fonksiyonuna ait çalışma adımları Algoritma 3.3'te verilmiştir.

```
çözüm = [] // 0 ve 1'lerden oluşan bir liste
haritadakiToplamHücreSayısı = tam sayı
kapsananHücreler = [] // boş liste
çözüm'deki her bir elemanı (eleman) gez:
| Eğer eleman'ın değeri 1 ise:
| | radarınKapsadığıHücreler = radarDosyasıOku(eleman) // her bir elemanı (x,y,z) olan liste
| | radarınKapsadığıHücreler'deki her bir kapsanan hücreyi (hücre) gez:
| | | Eğer hücre kapsananHücreler'de yoksa:
| | | | kapsananHücreler.ekle(hücre)
kapsamaOranı = (kapsananHücreler.uzunluk / haritadakiToplamHücreSayısı) * 100
kapsamaOranı'nı amaç fonksiyonu hesaplaması olarak kaydet
```

**Algoritma 3.3:** Çözümdeki radarların kapsama oranını hesaplayan uygunluk fonksiyonunun çalışma adımları.

Bu adımlarda geçen radarDosyasıOku() fonksiyonu verilen elemanın satır ve sütun bilgilerini bularak hangi radarı temsil ettiğini tespit ederek kayıtlı o radar dosyasını okumaktadır. Bu radar dosyası, radarın kapsadığı bütün hücrelerin bilgilerini içermektedir. Fonksiyonda okunan dosya içeriği, üç boyutlu bir veri listesine dönüştürülmektedir. Bu veri listesinin ilk boyutu, temsili haritada kapsanan hücrenin satır bilgisini (x), ikinci boyutu kapsanan hücrenin sütun bilgisini (y), ve üçüncü boyutu da kapsanan hücrenin yukarı doğru kaçınıcı hücre olduğu (z) bilgisini verir. Kısaca algoritma adımlarında bunlar x, y ve z olarak adlandırılmıştır.

### 3.1.3. Çözüm Dominantlığı Kontrolü

Algoritmalarda gerekli yerlerde hangi çözümün bir diğerinden ya da hangi çözümün diğer çözümlerden daha iyi olduğu belirlenirken eğer çözümün uygunluğu tek bir amaç fonksiyonuyla belirleniyorsa, bunların karşılaştırması yapılır. Eğer çözüm uygunluğu iki amaç fonksiyonuna göre belirleniyorsa, çözümler arasında dominantlık kontrolü yapılır. Bir çözüm başka bir çözümü şu durumlarda domine etmektedir:

- Eğer çözümün amaç fonksiyonları çıktılarının ikisi de diğer çözümden daha iyiyse.

- Eğer çözümün amaç fonksiyonları çıktılarında biri diğer çözümün aynı amaç fonksiyonu çıktısı ile aynı veya daha iyi ve diğer amaç fonksiyonunun çıktısı daha iyiyse.

Bu şartlardan birinin sağlanması, ilgili çözümün diğer çözümü domine ettiğini göstermektedir. Eğer bu şartlar sağlanmıyorsa, ilgili çözümün karşılaştırıldığı çözümü domine etmediği bilinir. Bu tez kapsamında problem iki amaçlı bir optimizasyon problemi olduğundan dominantlık karşılaştırması yukarıdaki kriterlere göre yapılmaktadır.

#### **3.1.4. Genetik Değişim**

Problemin çözümünde kullanılan MOSA dışındaki algoritmaların hepsinde aynı genetik değişim yöntemi kullanılmıştır. MOSA’da genetik değişim olmadığı için kullanılmamıştır. Algoritmalarda kullanılan genetik değişim yöntemi bir noktalı genetik değişimdir. Bu genetik değişimi uygulayabilmek için öncelikle iki ebeveyn seçimi yapılmaktadır. Ebeveyn seçiminde turnuva seçimi kullanılmıştır ve her bir ebeveyn seçiminde var olan popülasyon içerisinde rasgele 3 birey belirlenerek aralarındaki dominasyon ilişkisine bakılarak en iyi olan ebeveyn olarak seçilmiştir. Turnuvalar ile iki birey ebeveyn olarak belirlendikten sonra, bireylerin eleman sayısı sınırları içerisinde rasgele bir k noktası belirlenir. İlk belirlenen bireyin k noktasına kadarki elemanları ve ikinci belirlenen bireyin k noktasından sonraki elemanları sıraları bozulmadan alınarak birleştirilir ve yeni bir birey oluşturulur. Bu yöntem standart genetik değişim olarak belirlenmiştir.

#### **3.1.5. Mutasyon**

Problemin çözümünde çalıştırılan ve mutasyon özelliğini kullanan algoritmaların hepsinde standart olarak aynı mutasyon kullanılmıştır. Sadece MOSA’da kullanılmamıştır. Kullanılmış olan yöntem bit çevirme mutasyonudur. Verilen bir birey üzerinde gerçekleşen değişim adımları Algoritma 3.4’te verilmektedir.

```

çözüm = [] // mutasyona uğrayacak n elemanlı çözüm
mutasyonİhtimali = 1.0 / Çözüm.uzunluk
çözüm'deki her bir elemanı (eleman) gez:
| rasgeleSayı = 0-1 arası rasgele sayı
| Eğer rasgeleSayı < mutasyonİhtimali:
| | Eğer eleman == 0 ise:
| | | eleman = 1
| | Değilse:
| | | eleman = 0

```

**Algoritma 3.4:** Mutasyon gerçekleştirme adımları.

Adımları verilen mutasyonda verilen çözümdeki her bir eleman teker teker gezilmekte ve her bir eleman için 0-1 arasında bir rasgele sayı belirlenmektedir. Eğer bu sayı, belirlenmiş olan mutasyon ihtimalinden küçükse ilgili elemanın değeri tersine çevrilmektedir.

### 3.1.6. Hiper Hacim Performans Metriği

Metasezgisel algoritmalar ile bir takım problemlerin çözümü yapılırken oluşturulan çözümlerin ya da algoritmanın çalışması sonucu bulunan çözüm kümelerinin kalitesini belirlemek zor olabilmektedir. Eğer en iyi çözüm bilinmiyorsa ya da elde edilmek istenen başarımlar için belli bir sayısal ifade bulunmuyorsa, algoritmaların bulduğu çözümlerin kalitesini değerlendirmek için yardımcı metrikler kullanılmaktadır. Bu çalışmadaki problemde kapsama oranının en üst sınırı bilinmemekte ancak bu kapsamanın elde edilip edilemeyeceği bilinmemektedir. Aynı şekilde en az radar sayısının ne olabileceği bilinmemekte ama hangi kapsama oranları en az ne kadar radar gerektirir bilgisi birden fazla radar için bulunmamaktadır. Bu durumda üretilen çözüm kümelerinin kalitesini belirleyebilmek için hiper hacim performans metriği kullanılmıştır. Bu metrik ile çözüm kümelerinin belirlenmiş olan bir referans noktasına olan uzaklığı hesaplanarak çözüm kümesi için sayısal bir değer elde edilmekte, ve bu elde edilen değerler karşılaştırılarak, daha yüksek değere sahip çözüm kümelerinin daha iyi olduğu kabul edilmektedir. Yine elde edilebilecek en yüksek hiper hacim değeri bilinemediğinden, çalışma kapsamında bulgular kısmında direkt olarak bu değerleri göstermek çok etkili olmaz ve bu sebeple, bütün çalışmalar boyunca elde edilmiş en yüksek hiper hacim değeri baz alınarak, elde edilen bütün hiper hacim

değerleri bu en yüksek değere bölünerek normalizasyon yapılmıştır. Bu sayede çözüm kümelerinin hiper hacim oranı 1'e ne kadar yaklaşıyorsa o kadar iyi olduğu anlaşılmakta ve 1 olan hiper hacim oranına sahip çözüm kümesinin ise elde edilen en yüksek hiper hacim değeri olduğu bilinebilmektedir. Hiper hacim hesaplamaları yapılırken radar kapsama oranının minimizasyon olarak ele alınması için negatif hale getirilmiştir. Hiper hacimde kullanılan referans noktaları kapsama oranı için 0, radar sayısı için 11 olarak belirlenmiştir.

## 3.2. KULLANILAN ALGORİTMALAR

### 3.2.1. Çok Amaçlı Benzetimli Tavlama Algoritması

Sayısal sonuçlar elde edilirken, MOSA algoritması herhangi bir değişikliğe uğratılmadan kullanılmıştır. Algoritmanın çalışm adımları Algoritma 3.5'te verilmiştir. Algoritmanın en başında çalışma parametreleri atanmış olup boş bir çözüm tutan x veri yapısı oluşturulmuştur. Bu x veri yapısı içerisinde boş bir çözümün yanı sıra bu çözüme ait amaç fonksiyonu hesaplamalarını tutacak f1 ve f2 değişkenlerine sahiptir. Bu x veri yapısı oluşturulduktan sonra içerisindeki boş çözüm güncellenerek rasgele bir çözüm üretilmiştir. Bunun için çözümÜret metodu kullanılmıştır. Bu metod, boş bir çözümdeki her bir eleman için 0-1 arasında rasgele bir sayı üretir ve eğer bu sayı 0.0015'ten küçükse ilgili elemanın değerini 1, aksi halde 0 yapar ve çözümü x için rasgele üretmiş olur. Bu adımdan sonra amaçFonksiyonlarınıHesapla metodu ile x veri yapısının tuttuğu çözüm için radar kapsama oranı hesaplanarak f1 değişkeninde, radar sayısı hesaplanarak f2 değişkeninde saklanır. Karşılaştırmalar yapılırken f1 değeri ne kadar büyükse o kadar iyi, f2 değeri de ne kadar küçükse o kadar iyi olacak şekilde dominantlık kontrolleri yapılmaktadır. İlgili x veri yapısı oluşturulduktan sonra iterasyon sayısı kadar bir döngüye gidilmektedir. Döngü içinde ilk olarak x veri yapısındaki çözüm güncellenerek aynı özelliklere sahip bir y veri yapısı oluşturulmakta ve ardından y veri yapısında tutulan çözüm için f1 ve f2 değerleri hesaplanmaktadır. Çözüm güncelleme işlemi için çözümüGüncelle metodu kullanılmış olup bu metod da çözümdeki eleman sayısının yarısı kadar bir döngüye girerek döngünün her adımında herhangi bir elemanı işaret edecek rasgele bir indis değeri belirlemektedir. Ardından 0-1 arasında rasgele bir sayı üretmektedir ve üretilen bu sayı 0.0015'ten küçük ise indis değerinin işaret ettiği elemanın değeri 1 olarak



güncellenmekte, değilse 0 olarak güncellenmektedir. Çözüm güncelleme işleminden sonra  $y$ ,  $x$ 'i domine ediyor mu bunun kontrolü yapılmakta ve eğer  $x$ 'i domine ediyorsa  $y$  veri yapısı, en iyi çözümleri tutmak için oluşturulmuş olan bir listeye ekleme işlemi, `peListesiniGüncelle` metodu ile gerçekleştirilir. Bu metod aracılığı ile `peListesi`'ne eklenmeye çalışılan bir çözüm için çözüm zaten bu listede var mı diye kontrol edilir. Eğer yoksa çözümün, liste içerisinde domine ettiği çözümler var mı diye kontrol edilir ve varsa domine olan çözümler listeden kaldırılır ve bu yeni çözüm listedeki herhangi bir çözüm tarafından domine edilmiyorsa listeye eklenir. Listeye ekleme işleminden sonra  $y$  yeni veri yapısı olarak kabul edilerek  $x$  veri yapısına ataması yapılır. Eğer  $y$ ,  $x$ 'i domine etmemişse bu sefer algoritma adımlarında gösterildiği şekilde sıcaklık ve delta ilişkisi kullanılarak bir rasgelelik belirlenir ve  $y$ ,  $x$ 'i domine etmediği halde `peListesi`'ne eklenme şansı doğar, bu durum lokal optimumdan kaçmak için daha kötü çözümlerin değerlendirilmesi ilkesine dayanır. Son olarak eğer gerekiyorsa sıcaklık değişimi gerçekleştirilir ve iterasyon tamamlanmış olur. Bu şekilde 20000 iterasyon gerçekleştirilir ve her iterasyonun sonunda `peListesi`'ndeki çözümler, hiper hacim değerleri hesaplanmak üzere bir dosyaya kaydedilir. Daha sonra bu dosyadaki kayıtlar kullanılarak hiper hacim değişimleri ve en son çözüm listesinin hiper hacim değerleri hesaplanmaktadır.

```

peListesi [] // en iyi çözümleri tutacak olan liste
iterasyonSayısı = 20000
iterasyon = 0
sıcaklık = 100.0
sonSıcaklık = 0.001
alfaDegeri = 0.99
dereceninSabitKalacağıİterasyon = 10
x // boş çözüme sahip bir veri yapısı
x = çözümÜret(x)
amaçFonksiyonlarınıHesapla(x) // x için f1(radar sayısı) ve f2(kapsama oranı) değerlerini hesaplar
iterasyon < iterasyonSayısı olduğu sürece dön:
| y = çözümlüGüncelle(x)
| amaçFonksiyonlarınıHesapla(y)
| yDominantMı = ((y.f1 > x.f1 && y.f2 < x.f2) ||
| (y.f1 > x.f1 && y.f2 <= x.f2) ||
| (y.f1 >= x.f1 && y.f2 < x.f2))
| Eğer yDominantMı = evet ise:
| | peListesiniGüncelle(y)
| | x = y
| Değilse:
| | delta = (y.f1 - x.f1) + (y.f2 - x.f2)
| | rasgeleSayı = 0-1 arasında rasgele sayı
| | Eğer rasgeleSayı < (e-delta / sıcaklık) ise:
| | | peListesiniGüncelle(y)
| | | x = y
| Eğer sıcaklık > sonSıcaklık && iterasyon % dereceninSabitKalacağıİterasyon == 0 ise:
| | sıcaklık = sıcaklık * alfaDegeri
| iterasyon = iterasyon + 1

```

**Algoritma 3.5:** MOSA algoritması çalışma adımları.

### 3.2.2. Bellek Tabanlı Çok Amaçlı Genetik Algoritma

Klasik Genetik Algoritma (GA) kullanılırken değişikliğe uğratılmış ve çok amaçlı hale getirilmiştir. Bu algoritmaya MOSA'dakinin aynısı bir peListesi özelliği eklenmiş ve aynı mantık ve yapıyla her bir iterasyonda bulunan en iyi bireyin ve oluşturulan yeni bireyin listeye eklenmeye çalışılarak saklanması sağlanmıştır. Bu güncellenmiş

algoritma bu çalışma kapsamında MBMO-GA olarak anılmaktadır. Algoritmanın çalışma adımları Algoritma 3.6'da verilmiştir.

Algoritma çalışma adımlarında ilk olarak popülasyon oluşturulmaktadır. Popülasyon boyutu kadar çözüm, çalışmada standart olarak kabul edilen çözüm üretme yöntemi ile oluşturulup popülasyon isimli listede tutulmaktadır. Her bir çözüm, MOSA'daki x veri yapısı olarak tutulmaktadır. Buradaki veri yapısı MOSA'dakinden daha günceldir ve içerisine bir çözüm atandığı zaman f1 ve f2 değerlerini kendisi hesaplamaktadır. Yani popülasyon çözümleri oluşurken aynı zamanda f1 ve f2 değerleri de hesaplanmaktadır. Sonrasında iterasyon sayısı kadar bir döngüye girilir ve ilk olarak turnuva seçimi yapılarak iki ebeveyn çözüm belirlenir. Turnuva seçimi için kullanılan turnuvaSeçimiYap metodu iki tane ebeveyn seçmek için popülasyondan rasgele 6 çözüm seçer ve bunları 3'erli olarak karşılaştırır. Her üçlü için birbirlerini domine etme durumları göz önüne alınarak en iyi olan çözüm belirlenir. Eğer çözümler birbirlerini domine edemiyorsa ilk seçilen birey ebeveyn olarak kabul edilir. Ardından bu ebeveynler kullanılarak çalışmada bahsedilen standart genetik değişim uygulanır ve tek bir yeni çözüm oluşturulur. Ardından yine çalışmada standart olarak belirlenen mutasyon adımları gerçekleştirilerek yeni oluşan çözüm mutasyona uğratılır. Bütün popülasyondaki çözümlerin birbirlerini domine etme durumları dominasyonlarıHesapla metodu ile belirlenir ve her bir çözümün kaç adet diğer çözümü domine ettiği bulunur. En az dominasyon değerine sahip ilk çözüm en kötü çözüm olarak kabul edilerek popülasyondan atılır ve yerine yeni oluşturulan çözüm eklenir. Bu çözümün de popülasyonda kaç çözümü domine ettiğine bakılarak dominasyon değeri belirlenir ve sıralamadaki yerini alır. En fazla çözümü domine eden, yani dominasyon değeri en büyük olan ilk çözüm, popülasyondaki en iyi çözüm olarak kabul edilir ve peListesi'ne eklemek için peListesiniGüncelle metodu kullanılır. Aynı şekilde ebeveynler kullanılarak oluşturulan yeni birey de bu listeye eklenir. Bu metod, MOSA'da aynı isimdeki metod ile aynı işlemleri yaparak uygun olması halinde çözümü listeye ekleyerek saklar ve listeyi günceller. Döngüde her bir iterasyonun sonunda peListesi'ndeki çözümler, hiper hacim değerleri hesaplanmak üzere bir dosyaya kaydedilir. Daha sonra bu dosyadaki kayıtlar kullanılarak hiper hacim değişimleri ve en son çözüm listesinin hiper hacim değerleri hesaplanmaktadır.

```

peListesi [] // en iyi çözümleri tutacak olan liste
popülasyonBoyutu = 100
popülasyon = popülasyonOluştur(popülasyonBoyutu)
iterasyon = 20000
iterasyon sayısı kadar dön:
| ebeveyn1, ebeveyn2 = turnuvaSeçimiYap(popülasyon, 3)
| yeniÇözüm = genetikDeğişim(ebeveyn1, ebeveyn2)
| yeniÇözüm = mutasyon(yeniÇözüm)
| dominasyonlar = dominasyonlarıHesapla()
| enKötüÇözüm = enKötüyüBul(dominasyonlar)
| popülasyon.sil(enKötüÇözüm)
| popülasyon.ekle(yeniÇözüm)
| dominasyonlar = dominasyonlarıGüncelle(yeniÇözüm)
| enİyiÇözüm = enİyiyiBul(dominasyonlar)
| peListesiniGüncelle(enİyiÇözüm)
| peListesiniGüncelle(yeniÇözüm)

```

**Algoritma 3.6:** MBMO-GA çalışma adımları.

### 3.2.3. Bastırılmamış Sıralamalı Genetik Algoritma 2

NSGA-II çalışma adımları Algoritma 3.7’de verildiği gibidir. Algoritmanın diğer algoritmalara göre daha iyi sonuç vermesi üzerine bir sonraki bölümlerde yapılan deneylerde ve çalışmalarda yine bu algoritma adımları kullanılmış olup, değişiklikler ilgili bölümlerde açıkça belirtilmiştir.

Algoritmada ilk olarak, bu çalışmada standart kabul edilen çözüm üretme yöntemi kullanılarak çözümler oluşturulur ve önceki algoritmalarda olduğu gibi bir x veri yapısı oluşturularak çözüm bu yapıda tutulur. Çözümün yapıya ataması yapıldığı an f1 ve f2 değerleri hesaplanarak kaydedilir, ekstradan hesaplama gerek kalmamaktadır. Oluşturulan çözümler, çözümler isimli bir listede tutulur ve iterasyon sayısı kadar dönecek bir döngüye girilir. İterasyonlarda ilk olarak çözümler listesi, çözümler2 isimli bir listeye kopyalanır. Bundan sonra yeni jenerasyon üretme işlemleri başlar ve çözümler2 listesinin eleman uzunluğu çözüm sayısının iki katı olana kadar yeni çözümler üretilip bu listeye eklenir. Çözüm üretme aşamasında turnuva seçimi uygulanır. İlk olarak 3’erli 2 ebeveyn grubu rasgele oluşturulur ve çözümler

listesindeki indeks numaralarına göre düşük olan daha iyi olarak kabul edilerek karşılaştırma yapılır. Karşılaştırmalar sonucunda her iki gruptan en iyi olan belirlenerek ebeveyn olarak seçilir, eğer en iyi belirlenememişse rasgele seçilmiş olan çözümlerden ilki ebeveyn kabul edilir. Seçilen çözümler kullanılarak, çalışmada standart olarak kabul edilen genetik değişim yöntemi ile yeni bir çözüm üretilir ve bu çözüm yine bu çalışmada standart olarak kabul edilen mutasyon işlemlerinden geçerek mutasyona uğratılır, son olarak çözümler2 listesine ekleme işlemi gerçekleştirilir. Daha sonra çözümler2 listesindeki bütün çözümlerin f1 ve f2 değerleri sıraları bozulmadan iki farklı listeye kaydedilir. Bu iki liste nsa isimli fonksiyona verilir ve geri dönüş olarak bastırılmamışSıralamalıÇözümler listesi elde edilir. Bu nsa isimli fonksiyon, NSGA-II'de cepheler oluşturma amacıyla standart olarak kullanılan Bastırılmamış Sıralama Algoritması'nın çalışma adımlarını uygulayarak, çözümleri cephelere ayırır. Bu algoritma üzerinde herhangi bir değişiklik yapılmamış olup direkt varsayılan kullanılmıştır. Kendisine verilen f1 ve f2 listelerindeki değerleri kullanarak çözümler arasında dominantlık kontrolleri yaparak en dominant çözümleri ön cephelere koymaktadır. Üretilen cephelerden, en ön cephedekiler en iyi çözümlerdir ve arka sıralara ilerledikçe çözümlerin kalitesi düşmektedir. Bu algoritmadan sonra yeni bir çözüm listesi oluşturulur ve algoritmanın döndürdüğü bastırılmamışSıralamalıÇözümler cephe listesindeki her bir cepheyi gezmek için bir döngüye girilir. Her bir cephe için kalabalıklaşma mesafeleri hesaplanır. Mesafe hesaplaması, NSGA-II algoritmasının standart kalabalıklaşma hesaplaması kullanılarak yapılmaktadır. Bu hesaplamalarda öncelikle verilen f1 ve f2 listeleri kötüden iyiye doğru sıralanır. Ardından sıralı bu listelerin ilk ve son elemanları hariç bütün elemanları için kendilerinden sonra gelen değerden, kendilerinden önce gelen değer çıkarılır ve ilgili listedeki en büyük değerden en küçük değer çıkarılınca elde edilen değere bölünür. Bu bölüm sonucu oluşan sonuç, ilk ve son elemanı büyük bir değere sahip olan bir uzaklık listesine kaydedilir. Bunun ardından cephedeki her bir çözümün uzaklık listesindeki değerleri bulunur ve bu değerler toplanarak ilgili çözümün kalabalıklık mesafesi olarak kaydedilir. Bunun ardından cephedeki her bir çözüme ait kalabalıklık mesafesi bilgisi kullanılarak çözümler büyükten küçüğe doğru sıralanır. En iyi çözümler, kalabalığa en uzak çözümler olarak kabul edilmektedir, bu sebeple çözümlerin kalabalıklık mesafeleri ne kadar büyükse, kalabalıklaşmaya o

derece uzaktardır. Mesafelerine göre büyükten küçüğe sıralanan bu çözümler yeniÇözümlerListesi'ne eklenir ve ilgili döngü tamamlanıp bütün cepheler gezildikten sonra yeniÇözümlerListesi bölünerek çözüm sayısı kadar ilk çözümü sıraları bozulmadan alt liste olarak çözümler listesine atanır. Bu atamadan sonra çözümler listesi, öncekinden daha iyi çözümlere sahip olabilmektedir. Ana döngünün her bir iterasyonunda nsa kullanılarak oluşturulan bastırılmamışSıralamalıÇözümler listesinin ilk elemanı olan cephedeki çözümler, hiper hacim değerleri hesaplanmak üzere bir dosyaya kaydedilir. Daha sonra bu dosyadaki kayıtlar kullanılarak hiper hacim değişimleri ve en son çözüm listesinin hiper hacim değerleri hesaplanmaktadır.

```

iterasyon = 1000
çözümSayısı = 100
çözümler = çözümleriOluştur(çözümSayısı)
iterasyon sayısı kadar dön:
| çözümler2 = çözümler
| çözümler2.uzunluk < (çözümSayısı * 2) olduğu sürece dön:
| | çözüm1, çözüm2 = turnuvaSeçimiYap(çözümler, 3)
| | yeniÇözüm = genetikDeğişim(çözüm1, çözüm2)
| | yeniÇözüm = mutasyon(yeniÇözüm)
| | çözümler2.ekle(yeniÇözüm)
| f1Listesi = f1ListesiOluştur(çözümler2)
| f2Listesi = f2ListesiOluştur(çözümler2)
| bastırılmamışSıralamalıÇözümler = nsa(f1Listesi, f2Listesi)
| yeniÇözümListesi = []
| bastırılmamışSıralamalıÇözümler 'deki her bir eleman (cephe) için dön:
| | kalabalıklaşmaMesafeleri = kalabalıklaşmaMesafeleriniBul(f1Listesi, f2Listesi, cephe)
| | sıralıCephe = mesafelereGöreSırala(cephe, kalabalıklaşmaMesafeleri)
| | yeniÇözümListesi.hepsiniEkle(sıralıCephe)
| çözümler = yeniÇözümListesi.altListeGetir(0, ÇözümSayısı)

```

**Algoritma 3.7:** NSGA-II çalışma adımları.

### 3.3. NSGA-II İÇİN UYGULANAN YÖNTEMLER

Çalışmalar sonucu NSGA-II, diğer algoritmalara göre daha çeşitli ve daha iyi sonuçlar verdiği için NSGA-II üzerinde farklı yöntem ve teknikler uygulanarak, bunların sonuçlar üzerindeki etkilerinin gözlemlenmesi hedeflenmiştir.

### 3.3.1. Harita Çözünürlüğünün Düşürülmesi

Harita boyutu, daha önceki bölümlerde de açıklandığı üzere 120x200'lük bir matristir. Bu da 24000 hücre olduğu anlamına gelmektedir ve her bir hücre  $2700m^2$ 'lik bir alanı temsil etmektedir. Bu alanın çözünürlüğünün daha da arttırılıp hücre temsil alanlarının daha da küçültülmesi hesaplamalarda büyük ağırlaşmalara sebep olduğundan en idealinin bu olduğu düşünülmüştür ve daha yüksek çözünürlüklü bir harita üzerinde işlemler yapılmamıştır. Alanın çözünürlüğü ne kadar yüksekse elde ettiğimiz sonuçların güvenilirliği de o kadar yüksek olmakla beraber çözünürlüğü düşürmenin etkisini incelemek için farklı çözünürlükte haritalar oluşturulmuştur. Varsayılan  $2700m^2$ 'lik hücreli harita dışında her bir hücresi  $3600m^2$  alanı temsil eden 90x150 boyutunda bir harita, her bir hücresi  $4500m^2$  alanı temsil eden 72x120 boyutunda bir harita ve her bir hücresi  $5400m^2$  alanı temsil eden 60x100 boyutunda bir harita oluşturulmuştur.

### 3.3.2. Radar Yerleştirme İhtimalinin Hücre Bazlı Olarak Güncellenmesi

Çözümlere radar yerleştirirken daha yüksekteki radarların radar tipine, yerleştirilen bölgeye ve bir takım kriterlere göre daha iyi bir kapsama yapabileceği bilinmektedir. Bu sebeple kullandığımız haritadaki hücrelere radar yerleştirme ihtimalinin yükseklikle doğru orantılı bir şekilde olması durumunda daha iyi sonuç verip vermeyeceği gözlemlenmiştir. Bu sebeple hem çözümlerin üretilmesi, hem de çözümlere uygulanan mutasyonun çalışma adımları güncellenerek etkinin gözlemlenmesi hedeflenmiştir.

Çalışmada kullanılan standart çözüm üretim yönteminde kullanılan 0.0015 ihtimali güncellenerek hücre yüksekliğiyle ilişkili bir hale getirilmiş ve Denklem 3.1'teki  $p$  değeri ile değiştirilmiştir. İlgili denklemdeki  $h$  mutasyona uğrayacak hücrenin yüksekliğini ifade etmektedir.

$$h = \text{yükseklik bilgisi} \quad (3.1)$$

$$p = \frac{0.0015}{1000} * h$$

Çözümler mutasyona uğratılırken de yine 0.0015 olan ihtimal Denklem 3.2'deki  $p$  değeri ile değiştirilmiştir. İlgili denklemdeki  $h$  mutasyona uğrayacak hücrenin

yüksekliğini ve  $n$  ise çözümün uzunluğunu, yani hücre sayısını ifade etmektedir. Bu yöntemden bahsedilirken Yükseklik Ağırlıklı NSGA-II (YA-NSGA-II) ifadesi kullanılacaktır.

$$h = \text{yükseklik bilgisi} \quad (3.2)$$

$$n = \text{çözüm uzunluğu}$$

$$p = \frac{1.0}{n * 1000} * h$$

İkinci bir yöntem olarak radar yerleşimlerinde ilgili hücrenin merkeze olan uzaklığı dikkate alınmıştır. Bir radar köşelere değil de ortaya ne kadar yakın yerleştirilirse kapsama bölgesinin daha da büyüyeceği varsayılmaktadır, çünkü kapsanmak istenen bölge bellidir ve köşelere yakın koymak, istenilen bölgenin o radarca taranmasını etkilemektedir. Bunun etkisini görebilmek için radarların haritadaki her bir hücreye yerleştirilme ihtimali, radarın haritanın merkez noktasına olan uzaklığı kullanılarak güncellenmiştir.

Çalışmada standart olarak kullanılan çözüm üretiminde 0.0015 olan ihtimal Denklem 3.3'teki  $p$  değeri ile değiştirilmiştir. Aynı şekilde standart mutasyonda da 0.0015 olan ihtimal aynı şekilde güncellenmiştir. İlgili denklemde hücrelerin satır ve sütun numaraları kullanılarak aralarındaki uzaklık bulunduktan sonra 150'den çıkartılırlar. En büyük uzaklık 150'yi geçmeyeceği için ilgili hücre merkeze ne kadar yakınsa bu çıkarma işleminin sonucu o kadar büyük olacak, ne kadar uzaksa da çıkarma işleminin sonucu küçük olacaktır. Böylece hücrenin merkeze olan yakınlığı arttıkça çıkarma işlemi sonucu artacak ve o hücreye radar yerleştirme ihtimalini arttıracaktır. Bu yöntemden bahsedilirken Merkez Ağırlıklı NSGA-II (MA-NSGA-II) ifadesi kullanılacaktır.

$$rm = \text{alanın merkez hücresinin satır numarası} \quad (3.3)$$

$$cm = \text{alanın merkez hücresinin sütun numarası}$$

$$rc = \text{ilgili hücrenin satır numarası}$$

$$cc = \text{ilgili hücrenin sütun numarası}$$

$$w = 150 - \sqrt{(rm - rc)^2 + (cm - cc)^2}$$



$$p = \frac{0.0015}{1000} * w$$

Bu yöntemlerin, standart NSGA-II'ye göre çözüm kalitesi kontrolünün yanı sıra bu değişimler ile birlikte algoritmanın daha hızlı yakınsayıp yakınsamayacağı değerlendirilmiştir. Bunun için algoritmalar daha az sayıda iterasyonlarda çalıştırılarak hangisinin daha hızlı yakınsadığı değerlendirilmiştir.

### 3.3.3. Genetik Değişim ve Mutasyon Değişiklikleri

Yapılan analizler sonucunda YA-NSGA-II'nin NSGA-II daha da iyi sonuçlar elde ettiği görülmüş olup daha iyi sonuçlar elde edilip edilemeyeceğinin analizinin yapılması adına genetik değişim ve mutasyon adımlarında bazı güncellemeler yapılmıştır. Yapılan değişikliklerden en kayda değer olanları tekdüze genetik değişim ve tekdüze genetik değişimin ardından düşürmeli mutasyon gerçekleştirilmesi olmuştur.

Tekdüze genetik değişim esasında yeni bir çözüm oluşturmak üzere iki çözüm seçildiğinde, yeni oluşacak çözüm her bir genini %50 ihtimalle bir bireyden almaktadır. Genetik değişime yansıtıldığı kadarıyla yeni oluşacak çözümün sırasıyla her bir geni için 0 ile 1 arasında rasgele bir sayı üretilmiş ve bu sayı 0.5'ten küçükse birinci çözümden, büyükse ikinci çözümden ilgili geni almıştır.

Tekdüze genetik değişime sahip algoritmanın mutasyon kısmında da değişikliğe gidilmiş ve düşürme özelliği eklenmiştir. Düşürme olayında ise mutasyona uğrayan çözümün toplam radar sayısı belirlenmekte ve bu radarların 3'te 1'i rasgele bir şekilde silinmektedir. Daha az sayıda radar hedeflenen bu yaklaşımda çözümün radarlarının 3'te 1'i rasgele olarak silinerek, daha az sayıda radara sahip çözümlerin üretilebileceği düşünülmüştür. Bu işlem için radar sayısı 3'e bölünmüş ve elde edilen sayı aşağı yuvarlanarak silinecek radar sayısı elde edilmiş ve rasgele silmeler yapılmıştır.

## DÖRDÜNCÜ BÖLÜM

### 4. DENEYSEL ÇALIŞMALAR

#### 4.1. PARAMETRE ATAMALARI

MOSA algoritması için kullanılan parametreler Tablo 4.1’de verilmiştir. Bu algoritma çalıştırıldığında diğer algoritmalar ile aynı sayıda, 25 kere çalıştırılmış ve bütün bu çalışmalarda bulunan çözüm kümelerinin hiper hacim oranları hesaplanarak en iyisinin hangisi olduğu belirlenerek bulgular kısmına eklenmiştir.

**Tablo 4.1:** MOSA algoritmasının çalışma parametreleri.

|  |       |
|--|-------|
| İlk sıcaklık derecesi                    | 100.0 |
| Son sıcaklık derecesi                    | 0.001 |
| Alfa değeri                              | 0.99  |
| Derecenin sabit kaldığı iterasyon sayısı | 10    |
| İterasyon sayısı                         | 20000 |

MBMO-GA’nın çalışma parametreleri Tablo 4.2’de verilmiştir. Bu algoritma çalıştırıldığında diğer algoritmalar ile aynı sayıda, 25 kere çalıştırılmış ve bütün bu çalışmalarda bulunan çözüm kümelerinin hiper hacim oranları hesaplanarak en iyisinin hangisi olduğu belirlenerek bulgular kısmına eklenmiştir.

**Tablo 4.2:** MBMO-GA çalışma parametreleri.

|                   |       |
|-------------------|-------|
| Popülasyon boyutu | 100   |
| İterasyon sayısı  | 20000 |

NSGA-II çalışma parametreleri Tablo 4.3’te verildiği gibidir. Bu algoritma çalıştırıldığında diğer algoritmalar ile aynı sayıda, 25 kere çalıştırılmış ve bütün bu çalışmalarda bulunan çözüm kümelerinin hiper hacim oranları hesaplanarak en iyisinin hangisi olduğu belirlenerek bulgular kısmına eklenmiştir.

**Tablo 4.3:** NSGA-II için çalışma parametreleri.

|                   |     |
|-------------------|-----|
| Popülasyon boyutu | 100 |
| İterasyon sayısı  | 200 |

Bahsedilen bu algoritmalar çalışma süreleri boyunca bu parametrelere göre çalışmaktadırlar.

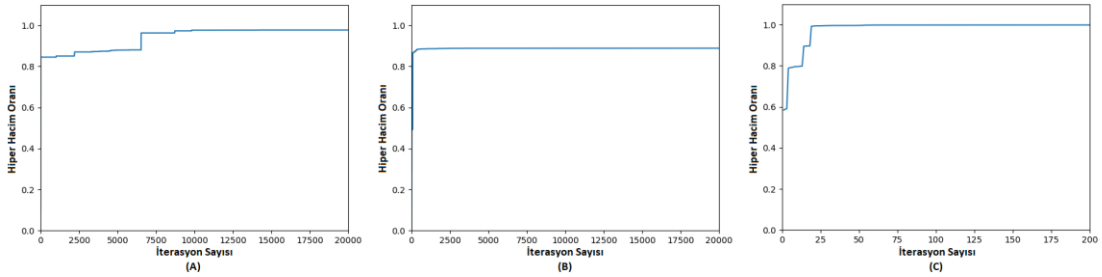
## 4.2. BULGULAR VE TARTIŞMA

MOSA, MBMO-GA ve NSGA-II algoritmalarının her biri 25 defa daha önce belirtilen parametreler ile çalıştırılmışlardır. Bu çalışmaların buldukları çözüm kümelerinin hiper hacim oranları hesaplanmıştır. Hesaplanan bu hiper hacim oranları ilgili istatistiksel bilgiler Tablo 4.4’de verilmiştir. Tablo 4.4’deki ve diğer hiper hacim gösterim tablolarındaki varyans harici değerlerin noktadan sonraki 4 elemanı alınmış ve dördüncü eleman için eğer yuvarlama yapılmıştır.

**Tablo 4.4:** MOSA, MBMO-GA ve NSGA-II algoritmalarının çözüm kümelerine ait hiper hacim oranı istatistikleri.

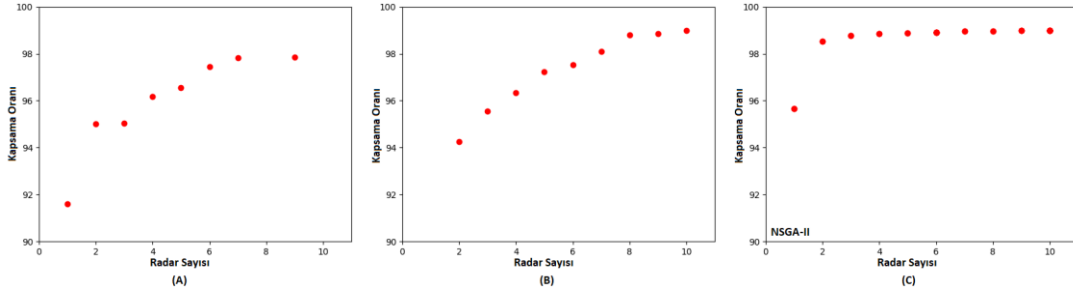
| Algoritma | Minimum       | Maksimum      | Ortalama      | Medyan        | Varyans |
|-----------|---------------|---------------|---------------|---------------|---------|
| MOSA      | <b>0.8793</b> | 0.9772        | 0.9297        | 0.9635        | 2.0E-03 |
| MBMO-GA   | 0.4911        | 0.8884        | 0.6679        | 0.6897        | 1.4E-02 |
| NSGA-II   | 0.7927        | <b>0.9987</b> | <b>0.9333</b> | <b>0.9680</b> | 4.5E-03 |

Tablo 4.4’e göre en yüksek hiper hacim oranına sahip çözümlerin çalıştıkları iterasyonlar boyunca buldukları çözümlere göre hiper hacim değişimleri Şekil 4.1’de verilmiştir. Şekilde A MOSA’yı, B MBMO-GA’yı ve C NSGA-II’yi temsil etmektedir. MOSA ve MBMO-GA’nın hiper hacim oranlarının NSGA-II’ye göre daha yetersiz kaldığı görülebilmektedir.



**Şekil 4.1:** Çalıştırılan algoritmaların hiper hacim oranı ve iterasyon sayısı yakınsama grafikleri. (A) MOSA, (B) MBMO-GA ve (C) NSGA-II.

Tablo 4.4’e göre en yüksek hiper hacim oranına sahip çözümlerin bulunduğu radar sayıları ve kapsama oranlarına ait görsel bilgiler Şekil 4.2’de, sayısal bilgiler ise Tablo 4.5’te verilmiştir. Şekil 4.2’de A MOSA’yı, B MBMO-GA’yı ve C NSGA-II’yi temsil etmektedir.



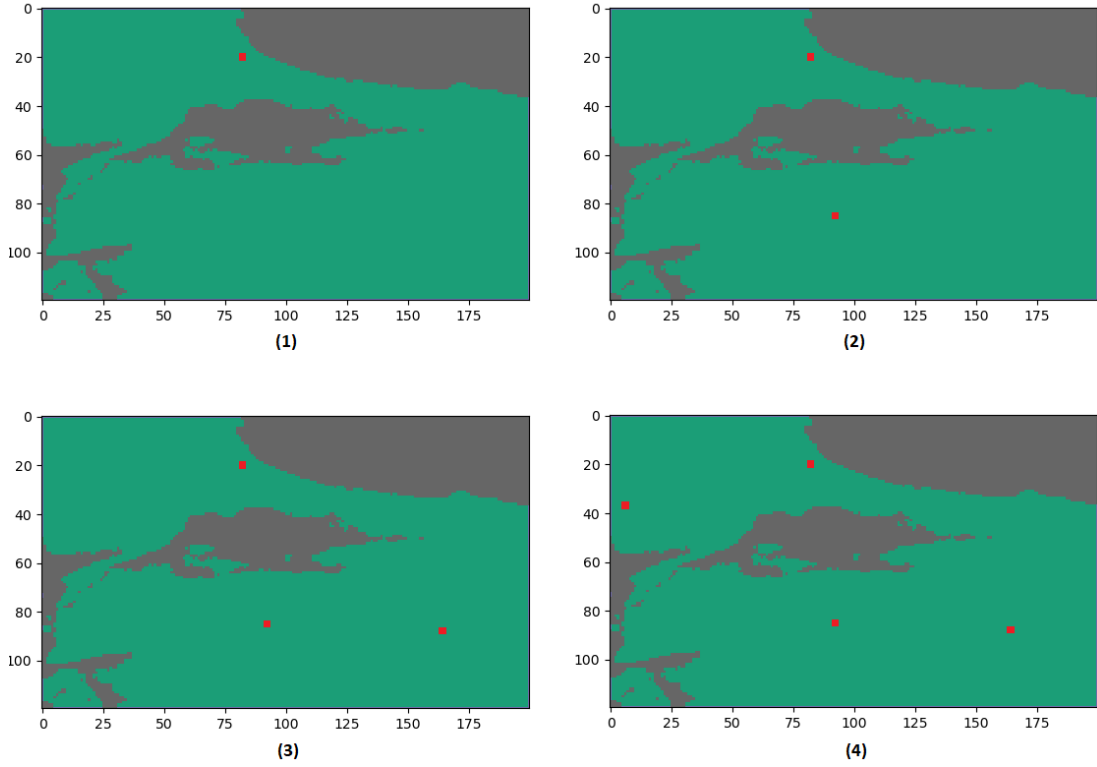
**Şekil 4.2:** En yüksek hiper hacim oranına sahip çözüm kümelerinin bulunduğu radar sayıları ve kapsama oranları (kapsama oranları % olarak verilmiştir).

Şekil 4.2, Tablo 4.5’deki bilgiler ile oluşturulmuştur.

**Tablo 4.5:** En yüksek hiper hacim oranına sahip çözüm kümelerinin bulunduğu radar sayıları ve kapsama oranları (kapsama oranları % olarak verilmiştir).

| Radar Sayısı | MOSA  | MBMO-GA | NSGA-II |
|--------------|-------|---------|---------|
| 1            | 91.59 | -       | 94.52   |
| 2            | 95.00 | 94.24   | 97.05   |
| 3            | 95.04 | 95.55   | 98.45   |
| 4            | 96.17 | 96.33   | 98.80   |
| 5            | 96.56 | 97.22   | 98.99   |
| 6            | 97.45 | 97.53   | 99.16   |
| 7            | 97.82 | 98.09   | 99.26   |
| 8            | -     | 98.80   | 99.29   |
| 9            | 97.85 | 98.84   | 99.36   |
| 10           | -     | 99.00   | 99.41   |

Tablo 4.5’e baktığımızda NSGA-II’nin bütün radar sayıları için en iyi kapsama oranlarını verdiğini görebilmekteyiz. Bu tabloya göre NSGA-II’nin 4 radara kadar bulunduğu çözümlerdeki radar yerleşimleri aşağıda Şekil 4.3’te verilmiştir. Şekildeki görüntülerin altındaki sayılar, görüntünün kaç adet radar içerdiğidir.



**Şekil 4.3:** NSGA-II'nin 4 radara kadar bulduğu çözümlerdeki radarların harita üzerine yerleştirilmesi.

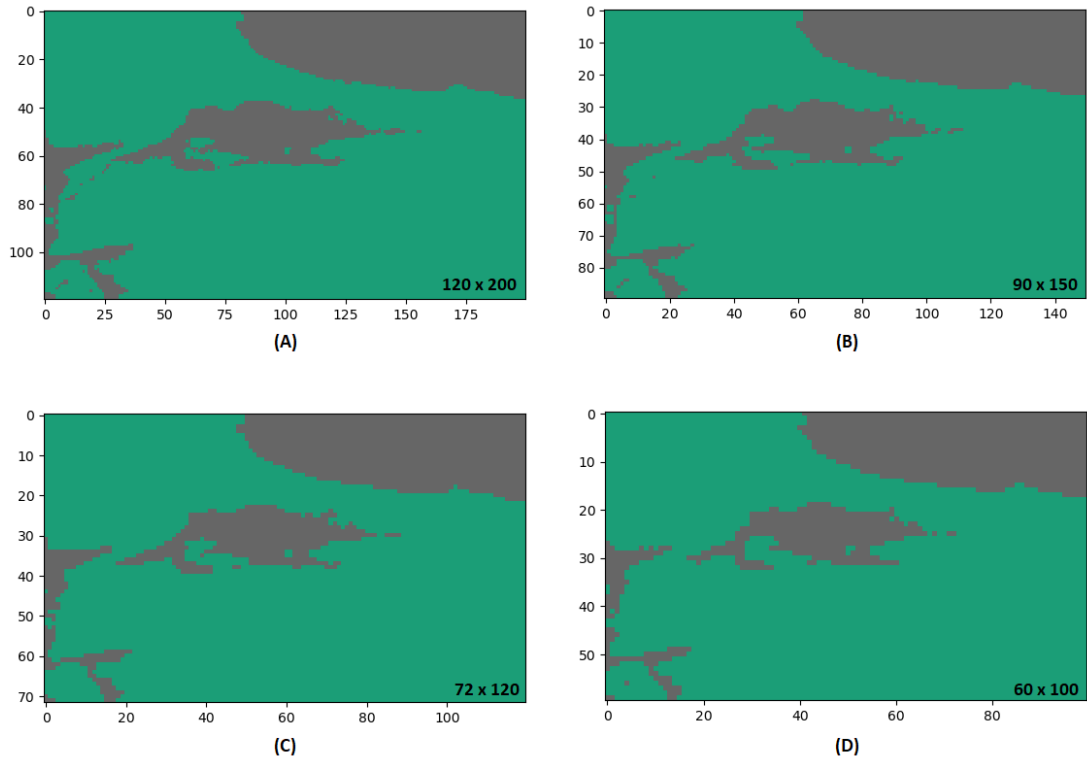
Bu tez kapsamında odaklanılan öncelikli konu en yüksek kapsama oranını elde etmek olduğundan daha yüksek hiper hacimler ve kapsama oranları daha önemlidir. Tablo 4.6'da bu üç algoritmanın ürettikleri 10000, 15000 ve 20000'inci çözümlerdeki hiper hacim oranına göre değerlendirme istatistikleri görülebilmektedir. Bu ve benzeri tablolardaki değerlendirmelerde Willcoxon Ranksum Test kullanılmıştır. Tabloda Max maksimum değeri, Ort ortalama değeri ve Var varyansı temsil etmektedir. Yine önceki tablolarda en yüksek hiper hacim oranını ve kapsama oranlarını veren NSGA-II'nin, diğer iki algoritmaya göre daha iyi çalıştığı görülebilmektedir. Bu sebeple bir sonraki yöntemler NSGA-II üzerinde denenmektedir.

**Tablo 4.6:** MOSA, MBMO-GA ve NSGA-II algoritmalarının 10000, 15000 ve 20000'inci çözümlerde elde ettikleri hiper hacim oranı istatistiksel değerlendirmesi.

| Çözüm Sayısı | MOSA   |        |         | MBMO-GA |        |         | NSGA-II       |               |         |
|--------------|--------|--------|---------|---------|--------|---------|---------------|---------------|---------|
|              | Max    | Ort    | Var     | Max     | Ort    | Var     | Max           | Ort           | Var     |
| 10000        | 0.9765 | 0.9016 | 1.6E-03 | 0.8884  | 0.6676 | 1.4E-02 | <b>0.9986</b> | <b>0.9212</b> | 4.9E-03 |
| 15000        | 0.9772 | 0.9098 | 1.8E-03 | 0.8884  | 0.6679 | 1.4E-02 | <b>0.9986</b> | <b>0.9258</b> | 5.1E-03 |
| 20000        | 0.9772 | 0.9297 | 2.0E-03 | 0.8884  | 0.6679 | 1.4E-02 | <b>0.9987</b> | <b>0.9333</b> | 4.5E-03 |

Harita boyutu, daha önceki bölümlerde de açıklandığı üzere 120x200'lük bir matristir. Bu da 24000 hücre olduğu anlamına gelmektedir ve her bir hücre  $2700m^2$ 'lik

bir alanı temsil etmektedir. Varsayılan  $2700m^2$ 'lik hücreli harita dışında her bir hücresi  $3600m^2$  alanı temsil eden  $90 \times 150$  boyutunda bir harita, her bir hücresi  $4500m^2$  alanı temsil eden  $72 \times 120$  boyutunda bir harita ve her bir hücresi  $5400m^2$  alanı temsil eden  $60 \times 100$  boyutunda bir harita oluşturulmuştur. Bu haritalar için NSGA-II algoritması çalıştırılmıştır. Bu haritaların 2 boyutlu matrisleri Şekil 4.4'te gösterilmiştir. Şekilde (A)  $120 \times 200$ 'lük haritayı, (B)  $90 \times 150$ 'lik haritayı, (C)  $72 \times 120$ 'lik haritayı ve (D)  $60 \times 100$ 'lük haritayı göstermektedir. Çözünürlük düştükçe harita hücreleri alanlarının da büyüdüğü açıkça görülebilmektedir.



**Şekil 4.4:** Varsayılan harita ile çözünürlükleri düşürülmüş haritalar.

Çözünürlükleri düşürülmüş haritalar ile gerçekleştirilen çalışmalara ait hiper hacim oranı istatistikleri Tablo 4.7'de, bu hiper hacim oranlarına göre en yüksek hiper hacmi veren çözüm kümelerine ait radar sayısı ve kapsama oranları Tablo 4.8'de verilmektedir. Bu haritalar üzerindeki çalışmalar NSGA-II ile gerçekleştirilmiştir ve 25'er kere çalıştırılmışlardır.

**Tablo 4.7:** NSGA-II algoritmasının farklı harita çözünürlüklerinde elde ettiği çözüm kümelerine ait hiper hacim oranları.

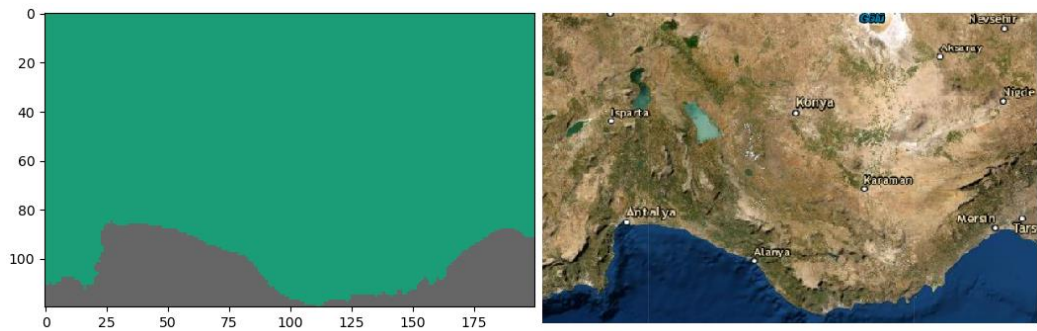
| Harita Hücre Temsil Alanı | Minimum | Maksimum | Ortalama | Medyan | Varyans |
|---------------------------|---------|----------|----------|--------|---------|
| 2700m <sup>2</sup>        | 0.7927  | 0.9987   | 0.9333   | 0.9680 | 4.5E-03 |
| 3600m <sup>2</sup>        | 0.8983  | 0.9989   | 0.9928   | 0.9968 | 3.9E-04 |
| 4500m <sup>2</sup>        | 0.9938  | 0.9999   | 0.9977   | 0.9982 | 2.5E-06 |
| 5400m <sup>2</sup>        | 0.9954  | 0.9997   | 0.9982   | 0.9984 | 1.1E-06 |

**Tablo 4.8:** En iyi hiper hacim oranlarına ait çözüm kümelerinin radar sayısı ve kapsama oranı bilgileri.

| Radar Sayısı | 2700m <sup>2</sup> | 3600m <sup>2</sup> | 4500m <sup>2</sup> | 5400m <sup>2</sup> |
|--------------|--------------------|--------------------|--------------------|--------------------|
| 1            | 94.52              | 94.56              | 94.47              | 94.49              |
| 2            | 97.05              | 97.12              | 97.43              | 97.41              |
| 3            | 98.45              | 98.39              | 98.66              | 98.51              |
| 4            | 98.80              | 98.80              | 99.01              | 98.91              |
| 5            | 98.99              | 98.98              | 99.13              | 99.08              |
| 6            | 99.16              | 99.16              | 99.23              | 99.23              |
| 7            | 99.26              | 99.29              | 99.32              | 99.33              |
| 8            | 99.29              | 99.34              | 99.38              | 99.41              |
| 9            | 99.36              | 99.40              | 99.44              | 99.47              |
| 10           | 99.41              | 99.45              | 99.49              | 99.49              |

Harita çözünürlükleri düştükçe büyük bir fark olmamakla beraber bir takım değişikliklerin olduğu görülmektedir. Çözünürlük düştükçe kapsama oranlarının artma eğiliminde olduğu ve güvenilirliğin de düştüğü anlaşılabilmektedir.

NSGA-II algoritmasının yeterince iyi sonuçlar verdiği gözlemlenmiştir, ancak bu iyi sonuçların seçilen bölgeyle ne kadar alakalı olduğu ya da bölge değişimi olması halinde algoritmanın nasıl çalışacağı incelenmiştir. Bu amaçla varsayılan harita ve hücre büyüklüklerine sahip başka bir bölgenin analizi yapılmıştır. Seçimi yapılan bölge Konya ilini ve Akdeniz bölgesinin bir bölümünü kapsayan bir bölgedir ve oluşturulan harita Yeni Harita olarak isimlendirilmiş olup iki boyutlu matris görüntüsü Şekil 4.5'te verilmiştir.



**Şekil 4.5:** Yeni haritanın iki boyutlu matris görüntüsü ve gerçek görüntüsü [Url-4].

Verilen bu harita Yeni Harita üzerinde radar yerleştirme işlemleri NSGA-II'nin 25 kere çalıştırılması ile gerçekleştirilmiş olup elde edilen çözüm kümelerine ait hiper hacim oranı istatistiklerinin varsayılan harita üzerinde çalıştırılan NSGA-II istatistikleri ile karşılaştırılması Tablo 4.9'da yapılmıştır. Aynı şekilde bu hiper hacim oranlarına ait radar sayısı ve kapsama oranları bilgilerinin karşılaştırılması da Tablo 4.10'da yapılmıştır. Bu karşılaştırma ile algoritmanın farklı haritalarda da benzer şekilde çalışıp benzer sonuçlar elde edebileceğini göstermek amaçlanmıştır.

**Tablo 4.9:** NSGA-II'nin farklı haritalarda çalıştırılması sonucunda elde edilen hiper hacim oranlarının karşılaştırılması.

| Algoritma         | Minimum | Maksimum | Ortalama | Medyan | Varyans |
|-------------------|---------|----------|----------|--------|---------|
| Varsayılan Harita | 0.7927  | 0.9987   | 0.9333   | 0.9680 | 4.5E-03 |
| Yeni Harita       | 0.4959  | 0.9822   | 0.8456   | 0.8797 | 1.4E-02 |

**Tablo 4.10:** En iyi hiper hacim oranlarına ait çözüm kümelerinin radar sayısı ve kapsama oranı karşılaştırmaları.

| Radar Sayısı | Varsayılan Harita | Yeni Harita |
|--------------|-------------------|-------------|
| 1            | 94.52             | 92.34       |
| 2            | 97.05             | 94.62       |
| 3            | 98.45             | 95.97       |
| 4            | 98.80             | 96.93       |
| 5            | 98.99             | 97.48       |
| 6            | 99.16             | 97.76       |
| 7            | 99.26             | 98.06       |
| 8            | 99.29             | 98.20       |
| 9            | 99.36             | 98.28       |
| 10           | 99.41             | 98.43       |

Verilen tablolardaki sonuçlarla birlikte NSGA-II algoritmasının farklı haritalarda da yeteri kadar iyi sonuç verebildiği gözlemlenmiştir. Her bölgenin radar ihtiyacının farklı olabilmektedir. Bazı bölgelere 2 radar yetebilirken bazı bölgelere benzer kapsama oranı için daha fazla radar yerleştirmek gerekebilmektedir. Bu bölgenin fiziksel yapısıyla ilişkili bir durum olarak karşımıza çıkabilmektedir. Varsayılan haritamız olan ve marmara bölgesini içeren haritamızda daha az radar ile daha iyi sonuçlar elde edilebilmesinin sebebi, bölgenin ye şekilleri bakımından sade bir yapıya sahip olmasıdır. Bu yeni harita ise Akdeniz bölgesinin bir bölümü ile ve bölgenin yukarısını içermektedir. Akdeniz bölgesi engebeli ve dağlık bir bölgedir ve aynı zamanda kıyıya paralel uzanan Toros dağlarını içerir. Böyle bir bölgeye radar yerleşimleri daha zordur ve daha az radar ile kapsama oranlarının engellerden ötürü daha düşük olması normal karşılanmaktadır.



Çözümlere radar yerleştirirken radarların daha yüksek yerlere yerleştirilme ihtimalinin yükseltilip, daha alçak yerlere yerleştirilme ihtimalinin düşürülmesi de çalışmaya dahil olan bir konudur. Aynı zamanda radarların haritanın merkezine yaklaştıkça yerleşme ihtimalinin arttırılıp, merkeze uzak yerlere yerleştirilme ihtimalinin de düşürülmesi gerçekleştirilmiştir. Bu iki değişiklik ile beraber NSGA-II algoritması standart parametreler ile çalıştırılmış olup, hem normal algoritmanın, hem de bu iki özelliğe sahip algoritmanın karşılaştırılmaları yapılmıştır. Bu çalışmalara ait hiper hacim oranları Tablo 4.11’de, bu hiper hacim oranı istatistiklerine göre en iyi hiper hacim oranına ait çözüm kümelerine ait radar ve kapsama oranları ise Tablo 4.12’de verilmiştir. Bu değişiklikler ile birlikte algoritmalar 25’er kere çalıştırılmışlardır.

**Tablo 4.11:** MA-NSGA-II ve YA-NSGA-II algoritmalarının hiper hacim oranı istatistiklerinin NSGA-II ile karşılaştırılması.

| Algoritma  | Minimum       | Maksimum      | Ortalama      | Medyan        | Standart Sapma |
|------------|---------------|---------------|---------------|---------------|----------------|
| NSGA-II    | 0.7928        | 0.9988        | 0.9373        | 0.9829        | 4.6E-03        |
| MA-NSGA-II | <b>0.9926</b> | 0.9983        | <b>0.9957</b> | <b>0.9955</b> | 2.4E-06        |
| YA-NSGA-II | 0.7026        | <b>1.0000</b> | 0.9197        | 0.9010        | 6.9E-03        |

**Tablo 4.12:** En iyi hiper hacim oranlı çözüm kümelerindeki radar sayısı ve kapsama oranlarının karşılaştırılması.

| Radar Sayısı | NSGA-II      | MA-NSGA-II   | YA-NSGA-II   |
|--------------|--------------|--------------|--------------|
| 1            | <b>94.52</b> | <b>94.52</b> | <b>94.52</b> |
| 2            | 97.05        | 96.99        | <b>97.42</b> |
| 3            | 98.45        | 98.36        | <b>98.61</b> |
| 4            | 98.80        | 98.77        | <b>98.98</b> |
| 5            | 98.99        | 99.00        | <b>99.14</b> |
| 6            | 99.16        | 99.08        | <b>99.27</b> |
| 7            | 99.26        | 99.23        | <b>99.34</b> |
| 8            | 99.33        | 99.29        | <b>99.40</b> |
| 9            | 99.39        | 99.35        | <b>99.45</b> |
| 10           | 99.44        | 99.38        | <b>99.49</b> |

Tablo 4.11 ve Tablo 4.12’ye bakıldığında, haritadaki hücre yükseklikleri ve merkeze olan mesafeler, radar yerleştirme hesaplamalarında dikkate alındığında daha iyi sonuçlar alındığı gözlemlenmiştir. Özellikle bu çalışma kapsamında en yüksek hiper hacim oranını veren YA-NSGA-II olmuştur. Kapsama oranlarına baktığımızda da YA-NSGA-II’nin radar sayısı arttıkça daha iyi oranlar verebildiği görülmüştür.

Tablo 4.13’te bu üç algoritmanın 100, 150, 200, 250 ve 300’üncü iterasyonlardaki çözüm kümelerinin hiper hacim oranına göre değerlendirme istatistiklerini

görebilmekteyiz. Tablodaki istatistiksel değerlendirmeye bakıldığında MA-NSGA-II'nin daha tutarlı ve güvenilir çalıştığı gözlemlenmektedir. En yüksek hiper hacim oranı ise YA-NSGA-II'de elde edilmiştir ve bu tez kapsamında en önemli konu kapsama oranının yükseltilmesi olduğu için YA-NSGA-II'nin daha iyi olduğu kabul edilmiş ve sonraki yöntemler bu teknik üzerinde değerlendirilmiştir.

**Tablo 4.13:** NSGA-II, MA-NSGA-II ve YA-NSGA-II'nin 100, 150, 200, 250 ve 300 iterasyonda elde ettikleri hiper hacim oranı istatistiksel değerlendirmesi.

| İterasyon | NSGA-II       |        |         | MA-NSGA-II |               |         | YA-NSGA-II    |        |         |
|-----------|---------------|--------|---------|------------|---------------|---------|---------------|--------|---------|
|           | Max           | Ort    | Var     | Max        | Ort           | Var     | Max           | Ort    | Var     |
| 100       | <b>0.9986</b> | 0.9212 | 4.9E-03 | 0.9969     | <b>0.9926</b> | 5.5E-06 | 0.9959+       | 0.8986 | 7.2E-03 |
| 150       | <b>0.9986</b> | 0.9258 | 5.1E-03 | 0.9975     | <b>0.9938</b> | 3.1E-06 | 0.9960+       | 0.9029 | 6.7E-03 |
| 200       | 0.9987+       | 0.9333 | 4.5E-03 | 0.9978     | <b>0.9948</b> | 2.3E-06 | <b>0.9999</b> | 0.9120 | 7.2E-03 |
| 250       | 0.9988+       | 0.9336 | 4.5E-03 | 0.9981     | <b>0.9953</b> | 2.5E-06 | <b>1.0</b>    | 0.9156 | 7.4E-03 |
| 300       | 0.9988+       | 0.9373 | 4.6E-03 | 0.9983     | <b>0.9957</b> | 2.4E-06 | <b>1.0</b>    | 0.9197 | 6.9E-03 |

YA-NSGA-II ile iyi sonuçlar elde edilmesi bu yöntem üzerinde biraz değişiklik yapılabileceğini göstermektedir, bu yüzden algoritma tarafında genetik değişim ve mutasyon adımlarında değişiklikler yapılmıştır. Daha önceki bölümlerde bahsedilen, standart genetik değişimi tekdüze genetik değişim olarak güncellenen algoritma ile bu algoritmaya düşürme özelliğinin eklenmiş başka bir versiyonu çalıştırılarak incelenmiştir. Bu güncellemelerin, standart genetik değişim ve mutasyon kullanan YA-NSGA-II ile kıyaslanmaları adına hiper hacim oranı analizleri Tablo 4.14'de, en iyi hiper hacim oranlarına ait radar sayıları ve kapsama oranları ise Tablo 4.15'de verilmiştir.

**Tablo 4.14:** NSGA-II, UC-YA-NSGA-II ve UCD-YA-NSGA-II hiper hacim oranı analizleri ve karşılaştırmaları.

| Algoritma      | Minimum       | Maksimum      | Ortalama      | Medyan        | Varyans |
|----------------|---------------|---------------|---------------|---------------|---------|
| YA-NSGA-II     | 0.7026        | <b>1.0000</b> | 0.9197        | 0.9010        | 6.9E-03 |
| UC-YA-NSGA-II  | <b>0.9007</b> | 0.9995        | <b>0.9932</b> | <b>0.9974</b> | 3.7E-04 |
| UCD-YA-NSGA-II | <b>0.9007</b> | 0.9995        | 0.9868        | 0.9970        | 7.4E-04 |

**Tablo 4.15:** En iyi hiper hacim oranlarına ait çözüm kümelerinin radar sayısı ve kapsama oranı karşılaştırmaları.

| Radar Sayısı | YA-NSGA-II   | UC-YA-NSGA-II | UCD-YA-NSGA-II |
|--------------|--------------|---------------|----------------|
| 1            | <b>94.52</b> | <b>94.52</b>  | <b>94.52</b>   |
| 2            | <b>97.42</b> | <b>97.42</b>  | <b>97.42</b>   |
| 3            | <b>98.61</b> | 98.54         | <b>98.61</b>   |
| 4            | <b>98.98</b> | 98.79         | 98.93          |
| 5            | <b>99.14</b> | 98.99         | 99.05          |
| 6            | <b>99.27</b> | 99.18         | 99.17          |
| 7            | <b>99.34</b> | 99.31         | 99.27          |
| 8            | <b>99.40</b> | <b>99.40</b>  | 99.33          |
| 9            | 99.45        | <b>99.46</b>  | 99.38          |
| 10           | <b>99.49</b> | 99.48         | 99.40          |

Tablo 4.14 ve Tablo 4.15’de görülebildiği kadarıyla radar sayısına göre kapsama performansı ele alındığında değişim yapılmayan YA-NSGA-II’nin daha iyi sonuçlar verebildiği gözlemlenmiştir. Bu eklemelerin algoritmanın güvenilirliğini ve tutarlılığını nasıl etkilediğini incelemek için Tablo 4.16’ya bakılabilir.

**Tablo 4.16:** NSGA-II, UC-YA-NSGA-II ve UCD-YA-NSGA-II’nin 100, 150, 200, 250 ve 300 iterasyonda elde ettikleri hiper hacim oranı istatistiksel değerlendirmesi.

| İterasyon | YA-NSGA-II    |        |         | UC-YA-NSGA-II |               |         | UCD-YA-NSGA-II |         |         |
|-----------|---------------|--------|---------|---------------|---------------|---------|----------------|---------|---------|
|           | Max           | Ort    | Var     | Max           | Ort           | Var     | Max            | Ort     | Var     |
| 100       | 0.9959        | 0.8986 | 7.2E-03 | <b>0.9990</b> | <b>0.9921</b> | 3.7E-04 | 0.9988+        | 0.9855+ | 7.3E-04 |
| 150       | 0.9960        | 0.9029 | 6.7E-03 | <b>0.9991</b> | <b>0.9925</b> | 3.7E-04 | 0.9989+        | 0.9859+ | 7.3E-04 |
| 200       | <b>0.9999</b> | 0.9120 | 7.2E-03 | 0.9994        | <b>0.9927</b> | 3.7E-04 | 0.9989         | 0.9863+ | 7.4E-04 |
| 250       | <b>1.0</b>    | 0.9156 | 7.4E-03 | 0.9994        | <b>0.9930</b> | 3.7E-04 | 0.9993         | 0.9867+ | 7.4E-04 |
| 300       | <b>1.0</b>    | 0.9197 | 6.9E-03 | 0.9995        | <b>0.9932</b> | 3.7E-04 | 0.9995         | 0.9868+ | 7.4E-04 |

Tablo 4.16 değerlendirildiğinde UC-NSGA-II’nin daha tutarlı ve güvenilir olduğu, çalışabilirlik açısından daha yeterli olduğu görülmektedir. Daha düşük iterasyonlarda daha iyi hiper hacim oranı bulmasından ötürü daha hızlı çalıştığı ve daha hızlı yakınsadığı anlaşılabilmektedir.

## SONUÇ

Bu tez kapsamında, belirlenen bir bölgeye en az sayıda radar yerleştirerek en yüksek kapsama oranını sağlamak üzerine çalışılmıştır. Kullanılan yöntemler ve metasezgisel algoritmalar ile yapılan optimizasyon çalışmaları sonucunda problemin çözülebildiği görülmüştür. Problemin çözümünde kullanılan üç algoritmanın da yeteri kadar iyi sonuç verebildiği ancak NSGA-II algoritmasının diğerlerine göre çok daha iyi sonuç verebildiği görülmüştür. Bu algoritma üzerinde yapılan bir takım güncelleme işlemleri ile daha da iyi sonuçlar alınabildiği ve daha da hızlandırılabilirdiği yine sonuçlardan anlaşılabilmektedir. Bu algoritmaya uygulanan ve radar yerleştirme ihtimalinin yüksekliklerle bağlantılı olduğu yöntem ile radar sayısı başına en yüksek kapsamaların elde edilmiştir. Elde edilen sonuçlara göre %99'u geçen yüksek kapsama oranları elde edilmiştir ancak kaç radar ile ne kadarlık bir kapsama oranının en uygun çözüm olduğu ihtiyaca göre belirlenebilmektedir. Yükseklik ile bağlantılı radar yerleştirme ihtimali ile güncellenen YA-NSGA-II, problemin çözümünde etkili bir yaklaşım olmuştur.

Gelecek çalışmalarda bu tez kapsamı daha da genişletilerek gerçek ihtiyaçlara daha da yakın çözümler elde edilebilir. Amaç fonksiyonlarının artırılarak problem daha da detaylandırılabilir. İleriki çalışmalarda radarların konum maliyetleri, belirli bir seviyeden itibaren gökyüzünün tamamen kapsanması için gerekli olan radar yerleşimleri, gerekli bölgeler için bölgenin gerekli kısmının üç veya daha fazla radarın kapsama kesişimi ile taranabilmesi ve büyük çözüm boyutlarına özel teknikler ile problemin ele alınması gibi konular çalışılabilir.

## KAYNAKÇA

- [1] **W. Shi, Q. He and L. Sun**, (2022) "Stable Matching-Based Transmit Antenna Placement for MIMO Radar Detection," *2022 IEEE Radar Conference (RadarConf22)*, 2022, pp. 01-05, doi: 10.1109/RadarConf2248738.2022.9764272.
- [2] **Y. Tang and Y. Wang**, (2020) "Mobile Sensors Path Planning for Cooperative Monitoring of Different Mission Importance Areas," *2020 IEEE Radar Conference (RadarConf20)*, 2020, pp. 1-6, doi: 10.1109/RadarConf2043947.2020.9266395.
- [3] **Y. Yang et al.**, (2015) "Fast Optimal Antenna Placement for Distributed MIMO Radar with Surveillance Performance," in *IEEE Signal Processing Letters*, vol. 22, no. 11, pp. 1955-1959, Nov. 2015, doi: 10.1109/LSP.2015.2445413.
- [4] **Y. Wang, W. Yi, S. Yang, M. Mallick and L. Kong**, (2020) "Antenna Placement Algorithm for Distributed MIMO Radar with Distance Constrains," *2020 IEEE Radar Conference (RadarConf20)*, 2020, pp. 1-6, doi: 10.1109/RadarConf2043947.2020.9266613.
- [5] **C. Yang, Y. Wang and W. Yi**, (2020) "Network Architecture Optimization for Area Surveillance in Multi-static Radar Systems," *2020 IEEE Radar Conference (RadarConf20)*, 2020, pp. 1-6, doi: 10.1109/RadarConf2043947.2020.9266701.
- [6] **Mirchandani, P.B., Li, J.-Q. and Long, Y.**, (2010) "Locating a Surveillance Infrastructure in and Near Ports or on Other Planar Surfaces to Monitor Flows." *Computer-Aided Civil and Infrastructure Engineering*, 25: 89-100. <https://doi.org/10.1111/j.1467-8667.2009.00623.x>.
- [7] **Katoch, S., Chauhan, S.S. & Kumar, V.**, (2021) "A review on genetic algorithm: past, present, and future." *Multimed Tools Appl* 80, 8091–8126 (2021). <https://doi.org/10.1007/s11042-020-10139-6>.
- [8] **Henderson, D., Jacobson, S. H., & Johnson, A. W. (n.d.)**, "The Theory and Practice of Simulated Annealing." *International Series in Operations Research & Management Science*, 287–319. doi:10.1007/0-306-48056-5\_10.

- [9] **Serafini, P.** (1994). "Simulated Annealing for Multi Objective Optimization Problems." In: Tzeng, G.H., Wang, H.F., Wen, U.P., Yu, P.L. (eds) Multiple Criteria Decision Making. Springer, New York, NY. [https://doi.org/10.1007/978-1-4612-2666-6\\_29](https://doi.org/10.1007/978-1-4612-2666-6_29).
- [10] **Amine, Khalil.** (2019). "Multiobjective Simulated Annealing: Principles and Algorithm Variants." *Advances in Operations Research*. 2019. 1-13. 10.1155/2019/8134674..
- [11] **Kocatürk, Ahmet & Altunkaynak, Bulent.,** (2019). "Çok Amaçlı Optimizasyon Problemlerinin NSGA-II Yaklaşımıyla Çözümü."
- [12] **K. Deb, A. Pratap, S. Agarwal and T. Meyarivan,** (2002) "A fast and elitist multiobjective genetic algorithm: NSGA-II," in *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182-197, April 2002, doi: 10.1109/4235.996017.
- [13] **Sun B, Chen H, Yang D, Li X.,** (2014) "Antenna Selection and Placement Analysis of MIMO Radar Networks for Target Localization." *International Journal of Distributed Sensor Networks*. May 2014. doi:10.1155/2014/769404.
- [14] **Xie, R., Wan, X., Luo, K., Yi, J. and Jiang, T.,** (2019), "K -coverage based receiver placement optimisation in passive radar network." *The Journal of Engineering*, 2019: 5841-5845. <https://doi.org/10.1049/joe.2019.0157>.
- [15] **Abhinav Jauhri, Jason D. Lohn, and Derek S. Linden.,** (2014). "A comparison of antenna placement algorithms." In *Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation (GECCO Comp '14)*. Association for Computing Machinery, New York, NY, USA, 1223–1230. <https://doi.org/10.1145/2598394.2605446>.
- [16] **S. Pllana, S. Memeti and J. Kolodziej,** (2019) "Customizing Pareto Simulated Annealing for Multi-Objective Optimization of Control Cabinet Layout," 2019 22nd International Conference on Control Systems and Computer Science (CSCS), 2019, pp. 78-85, doi: 10.1109/CSCS.2019.00021..
- [17] **Ghaffarinasab, Nader & Jabalameli, Mohamad & Saboury, Ali.,** (2013) "Multi-objective capacitated location-routing problem: Modelling and a simulated annealing heuristic." *International Journal of Services and Operations Management*. 15. 140-156. 10.1504/IJSOM.2013.053642.
- [18] **Vahid Hajipour, Parviz Fattahi, Madjid Tavana, Debora Di Caprio,** (2016) Multi-objective multi-layer congested facility location-allocation

problem optimization with Pareto-based meta-heuristics *Applied Mathematical Modelling*, Volume 40, Issues 7–8, 2016, Pages 4948-4969, ISSN 0307-904X, <https://doi.org/10.1016/j.apm.2015.12.013..>

- [19] **İbrahim Demir, Berna Kiraz, Fatma Corut Ergin**, (2022) Experimental evaluation of meta-heuristics for multi-objective capacitated multiple allocation hub location problem, *Engineering Science and Technology, an International Journal*, Volume 29, 2022, 101032, ISSN 2215-0986, <https://doi.org/10.1016/j.jestch.2021.06.012..>
- [20] **Diogo R.M. Fernandes, Caroline Rocha, Daniel Aloise, Glaydston M. Ribeiro, Enilson M. Santos, Allyson Silva**, (2014) “A simple and effective genetic algorithm for the two-stage capacitated facility location problem” *Computers & Industrial Engineering*, Volume 75, 2014, Pages 200-208, ISSN 0360-8352, <https://doi.org/10.1016/j.cie.2014.05.023..>
- [21] **Abdelhalim Hiassat, Ali Diabat, Iyad Rahwan**, (2017) “A genetic algorithm approach for location-inventory-routing problem with perishable products” *Journal of Manufacturing Systems*, Volume 42, 2017, Pages 93-103, ISSN 0278-6125, <https://doi.org/10.1016/j.jmsy.2016.10.004..>
- [22] **Kim, Namhoon & Kim, Sang-Pil & Kim, Mi-Kyeong & Sohn, Hong-Gyoo.**, (2016) “A Genetic Algorithm to Solve the Optimum Location Problem for Surveillance Sensors” *Journal of the Korean Society of Surveying, Geodesy, Photogrammetry and Cartography*. 34. 547-557. [10.7848/ksgpc.2016.34.6.547..](https://doi.org/10.7848/ksgpc.2016.34.6.547..)
- [23] **Rabbani, M., Farrokhi-Asl, H. & Asgarian, B.**, “Solving a bi-objective location routing problem by a NSGA-II combined with clustering approach: application in waste collection problem.” *J Ind Eng Int* 13, 13–27 (2017). <https://doi.org/10.1007/s40092-016-0172-8>.
- [24] **Ebrahimi Zade, A., Sadegheih, A. & Lotfi, M.M.**, (2014) “A modified NSGA-II solution for a new multi-objective hub maximal covering problem under uncertain shipments.” *J Ind Eng Int* 10, 185–197 (2014). <https://doi.org/10.1007/s40092-014-0076-4>.
- [25] **Guo, Danqing & Wang, Junping & Huang, Jun & Han, Renmin & Song, Maoqiang.**, (2010) “Chaotic-NSGA-II: An effective algorithm to solve multi-objective optimization problems.” *Proceedings - 2010 International Conference on Intelligent Computing and Integrated Systems, ICISS2010*. 10.1109/ICISS.2010.5654998.
- [26] **Mingjie Song & Dongmei Chen**, (2018) “An improved knowledge-informed NSGA-II for multi-objective land allocation (MOLA)” *Geo-spatial*

Information Science, 21:4, 273-287, DOI:  
10.1080/10095020.2018.1489576.

- [27] **Yahui W, Ling S, Cai Z, Liuqiang F, Xiangjie J., (2020)** “NSGA-II algorithm and application for multi-objective flexible workshop scheduling.” *Journal of Algorithms & Computational Technology*. January 2020. doi:10.1177/1748302620942467.
- [28] **E. Yakıcı,** (2018) "A multiobjective fleet location problem solved by adaptation of evolutionary algorithms NSGA-II and SMS-EMOA", *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, c. 24, sayı. 1, ss. 94-100, Şub. 2018.
- [29] **X. Liu, C. Zhang, Y. Cao, L. Han and Z. Wu,** (2021) "Multi-Radar Cooperative Task Planning using NSGA-II Algorithm," 2021 IEEE 21st International Conference on Software Quality, Reliability and Security Companion (QRS-C), 2021, pp. 579-583, doi: 10.1109/QRS-C55045.2021.00088.
- [30] **SRTM DTED Format,** DLR(German Aerospace Center), SRTM/PD-03/11/03 Version 1.1

**Url-1** <[usgs.gov](http://usgs.gov)>, erişim tarihi 03.05.2022.

**Url-2** <[radartutorial.eu](http://radartutorial.eu)>, erişim tarihi 03.05.2022.

**Url-3** <[en.wikipedia.org](http://en.wikipedia.org)>, erişim tarihi 03.05.2022.

**Url-4** <[earthexplorer.usgs.gov](http://earthexplorer.usgs.gov)>, erişim tarihi 05.05.2022.

**Url-5** <[gdal.org](http://gdal.org)>, erişim tarihi 05.05.2022.

**Url-6** <[movable-type.co.uk](http://movable-type.co.uk)>, erişim tarihi 05.05.2022.