



**T.C.
DÜZCE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**AMELİYAT ODASI ÇİZELGELEMENİN GENETİK ALGORİTMA
İLE GERÇEKLEŞTİRİLMESİ**

TUNAHAN TİMUÇİN

**YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI**

**DANIŞMAN
DR. ÖĞR. ÜYESİ SERDAR BİROĞUL**

DÜZCE, 2018

T.C.
DÜZCE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**AMELİYAT ODASI ÇİZELGELEMENİN GENETİK ALGORİTMA
İLE GERÇEKLEŞTİRİLMESİ**

Tunahan TİMUÇİN tarafından hazırlanan tez çalışması aşağıdaki jüri tarafından Düzce Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Tez Danışmanı

Dr. Öğr. Üyesi Serdar BİROĞUL

Düzce Üniversitesi

Jüri Üyeleri

Dr. Öğr. Üyesi Serdar BİROĞUL

Düzce Üniversitesi

Doç. Dr. Aydın ÇETİN

Gazi Üniversitesi

Dr. Öğr. Üyesi Bahar TERCAN

Düzce Üniversitesi

Tez Savunma Tarihi: 13/12/2018

BEYAN

Bu tez çalışmasının kendi çalışmam olduğunu, tezin planlanmasından yazımına kadar bütün aşamalarda etik dışı davranışımın olmadığını, bu tezdeki bütün bilgileri akademik ve etik kurallar içinde elde ettiğimi, bu tez çalışmasıyla elde edilmeyen bütün bilgi ve yorumlara kaynak gösterdiğimi ve bu kaynakları da kaynaklar listesine aldığımı, yine bu tezin çalışılması ve yazımı sırasında patent ve telif haklarını ihlal edici bir davranışımın olmadığını beyan ederim.

13 Aralık 2018

(İmza)

Tunahan TİMUÇİN

TEŐEKKÖR

Yüksek Lisans öğrenimimde ve bu tezin hazırlanmasında gösterdiği her türlü destek ve yardımdan dolayı çok değerli hocam Dr. Öğr. Üyesi Serdar Biroğul'a en içten dileklerle teşekkür ederim.

Bu çalışma boyunca yardımlarını ve desteklerini esirgemeyen sevgili aileme ve çalışma arkadaşlarıma sonsuz teşekkürlerimi sunarım.

13 Aralık 2018

Tunahan TİMUÇİN

İÇİNDEKİLER

Sayfa No

ŞEKİL LİSTESİ	vii
ÇİZELGE LİSTESİ	ix
KISALTMALAR.....	x
ÖZET	xi
ABSTRACT	xii
1. GİRİŞ.....	1
2. PROBLEM TANIMI.....	4
2.1. AMELİYAT ODASI KARAR SEVİYELERİ VE ÇİZELGELEME TÜRLERİ	5
3. AMELİYAT ODASI ÇİZELGELEME VE OPTİMİZASYONU.	7
4. GENETİK ALGORİTMA	11
4.1. TEMEL GENETİK ALGORİTMA	12
4.1.1. Parametrelerin Kodlanması.....	13
4.1.1.1. Binary (İkili) Kodlama	13
4.1.1.2. Permutasyon Kodlama.....	14
4.1.1.3. Ağaç Kodlama.....	14
4.1.2. Başlangıç Popülasyonunun Oluşturulması	15
4.2. GENETİK ALGORİTMA NESİL DÖNGÜSÜNÜN OLUŞTURULMASI... ..	15
4.3. GENETİK ALGORİTMADA TEMEL KAVRAMLAR.....	18
4.3.1. Gen.....	18
4.3.2. Kromozom	19
4.3.3. Popülasyon	19
4.4. SEÇİM MEKANİZMALARI	20
4.4.1. Rulet Tekerleği Seçim Yöntemi	20
4.4.2. Turnuva Seçim Yöntemi.....	21
4.4.3. Rank Seçim Yöntemi.....	22
4.4.4. Sabit Durum Seçimi	22
4.4.5. Elitist Seçim Yöntemi.....	22
4.5. GENETİK OPERATÖRLER.....	22
4.5.1. Çaprazlama Operatörü	22
4.5.1.1. Tek Noktalı Çaprazlama	23
4.5.1.2. İki Noktalı Çaprazlama.....	23
4.5.1.3. Çok Noktalı Çaprazlama.....	24
4.5.2. Mutasyon Operatörü	24
4.5.2.1. Komşu Mutasyon.....	25
4.5.2.2. Keyfi Mutasyon	25
4.5.2.3. Ötelemeli Mutasyon	25

4.5.3. Tamir Operatörü.....	26
4.6. GA'NIN KISITLI OPTİMİZASYON PROBLEMLERİNDE KULLANIMI	27
4.7. BASİT BİR GA UYGULAMA ÖRNEĞİ.....	28
5. GA'NIN AMELİYAT ODASI ÇİZELGELEMeye	
UYARLANMASI.....	32
5.1. UYGULAMA PARAMETRE KODLAMA.....	32
5.1.1. Değer Kodlama.....	32
5.2. YAZILIM PROGRAMININ İNCELENMESİ	34
5.2.1. Program Kısıtları	35
5.2.1.1. Sert Kısıtlar.....	36
5.2.1.2. Yumuşak Kısıtlar.....	36
5.2.2. Program Algoritma Akışı.....	37
5.3. DENEYSEL SONUÇLAR.....	38
5.3.1. Doktor Sayısının Problem Çözümüne Etkisi.....	38
5.3.2. Ameliyathane Sayısının Problem Çözümüne Etkisi	42
5.3.3. Rezervasyon Sayısının Problem Çözümüne Etkisi	47
5.4. TAMİR OPERATÖRÜNÜN GENETİK ALGORİTMA	
UYGULANMASINA ETKİSİ.....	52
6. SONUÇ ve DEĞERLENDİRME	54
7. KAYNAKLAR	57
ÖZGEÇMİŞ	60

ŞEKİL LİSTESİ

	<u>Sayfa No</u>
Şekil 2.1. Ameliyat odası karar seviyeleri.	6
Şekil 4.1. Nümerik kodlama.	14
Şekil 4.2. Alfa-nümerik kodlama.	15
Şekil 4.3. İkili kodlama örneği.	15
Şekil 4.4. Genetik algoritma akış diyagramı.	18
Şekil 4.5. Gen yapısı.	19
Şekil 4.6. Kromozom yapısı.	19
Şekil 4.7. Popülasyon yapısı.	20
Şekil 4.8. Bireylerin uygunluk durumuna göre rulet tekerleği üzerine dağılımı.	21
Şekil 4.9. Turnuva seçim yöntemi.	21
Şekil 4.10. Tek noktalı çaprazlama.	23
Şekil 4.11. İki noktalı çaprazlama.	24
Şekil 4.12. Çok noktalı çaprazlama.	24
Şekil 4.13. Komşu mutasyon.	25
Şekil 4.14. Keyfi mutasyon.	25
Şekil 4.15. Ötelemeli mutasyon.	26
Şekil 4.16. Tamir operatörünün uygulanmama durumu.	26
Şekil 4.17. Tamir operatörünün uygulanma durumu.	27
Şekil 4.18. Kromozomların rulet tekerleği üzerine dağılımı.	29
Şekil 5.1. Kromozom yapısı.	32
Şekil 5.2. Uygulama başlangıç ekranı.	34
Şekil 5.3. Rezervasyon ekranı.	35
Şekil 5.4. Uygulama kayıt ve düzenleme ekranı.	35
Şekil 5.5. Kısıt durumunun incelenmesi.	37
Şekil 5.6. Algoritma akış şeması.	37
Şekil 5.7. Ceza değeri.	38
Şekil 5.8. Uygunluk değeri.	39
Şekil 5.9. Ceza değeri.	39
Şekil 5.10. Uygunluk değeri.	40
Şekil 5.11. Ceza değeri.	40
Şekil 5.12. Uygunluk değeri.	41
Şekil 5.13. Problem çözüm kromozomu (ameliyathane 1).	42
Şekil 5.14. Problem çözüm kromozomu (ameliyathane 1).	42
Şekil 5.15. Problem çözüm kromozomu (ameliyathane 2).	42
Şekil 5.16. Problem çözüm kromozomu (ameliyathane 2).	42
Şekil 5.17. Ceza değeri.	43
Şekil 5.18. Uygunluk değeri.	44
Şekil 5.19. Ceza değeri.	44
Şekil 5.20. Uygunluk değeri.	45
Şekil 5.21. Ceza değeri.	45
Şekil 5.22. Ceza değeri.	46
Şekil 5.23. Problem çözüm kromozomu (1 ameliyathane).	46

Şekil 5.24. Problem çözüm kromozomu (1 ameliyathane).....	46
Şekil 5.25. Ceza değeri.	47
Şekil 5.26. Uygunluk değeri.	48
Şekil 5.27. Çözüm kromozomu ameliyathane 1 (ilk 6 saat).....	48
Şekil 5.28. Çözüm kromozomu ameliyathane 1 (son 6 saat).....	49
Şekil 5.29. Çözüm kromozomu ameliyathane 2 (ilk 6 saat).....	49
Şekil 5.30. Çözüm kromozomu ameliyathane 2 (son 6 saat).....	49
Şekil 5.31. Ceza değeri.	49
Şekil 5.32. Uygunluk değeri.	50
Şekil 5.33. Çözüm kromozomu ameliyathane 1 (ilk 6 saat).....	51
Şekil 5.34. Çözüm kromozomu ameliyathane 1 (son 6 saat).....	51
Şekil 5.35. Çözüm kromozomu ameliyathane 2 (ilk 6 saat).....	51
Şekil 5.36. Çözüm kromozomu ameliyathane 2 (son 6 saat).....	51
Şekil 5.37. Çözüm kromozomu ameliyathane 3 (ilk 6 saat).....	51
Şekil 5.38. Tamir operatörünün uygulanmama durumu.	52
Şekil 5.39 Tamir operatörünün uygulanmama durumunda genlerin anlamı.	52
Şekil 5.40. Tamir operatörünün uygulanması sonrası görünüm.	53
Şekil 5.41. Tamir operatörünün uygulanması durumunda genlerin anlamı.	53

ÇİZELGE LİSTESİ

	<u>Sayfa No</u>
Çizelge 4.1. Örnek popülasyon havuzu (nesil 0).....	29
Çizelge 4.2. Çaprazlama öncesi ve sonrası durum.	30
Çizelge 4.3. Mutasyon öncesi ve sonrası durum.	31
Çizelge 4.4. Çözümün bulunması durumu (nesil 1).	31
Çizelge 5.1. Ameliyat süreleri ve ameliyathane hazırlık süreleri.	33
Çizelge 5.2. Doktor sayısının 16 olması durumu.....	38
Çizelge 5.3. Doktor sayısının 12 olması durumu.....	39
Çizelge 5.4. Doktor sayısının 8 olması durumu.....	40
Çizelge 5.5. Ameliyathane sayısının 3 olma durumu	43
Çizelge 5.6. Ameliyathane sayısının 2 olma durumu	44
Çizelge 5.7. Ameliyathane sayısının 1 olma durumu.	45
Çizelge 5.8. Rezervasyon sayısı 3 olduğundaki durum.	47
Çizelge 5.9. Rezervasyon sayısı 8 olduğundaki durum.	49

KISALTMALAR

GA	Genetik Algoritma
KDS	Karar Destek Sistemi
N	Popülasyon Büyüklüğü
n	Kromozom Sayısı
NP	Non-Polynomial (Polinom Olmayan)
OR	Operating Room (Ameliyat Odası)



ÖZET

AMELİYAT ODASI ÇİZELGELEMENİN GENETİK ALGORİTMA İLE GERÇEKLEŞTİRİLMESİ

Tunahan TİMUÇİN
Düzce Üniversitesi

Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı
Yüksek Lisans Tezi

Danışman: Dr. Öğr. Üyesi Serdar BİROĞUL
Ocak 2018, 59 sayfa

Bu tezde, hastanelerin en önemli birimlerinden birisi olan Ameliyat Odalarının en verimli şekilde kullanılabilmesi problemi ele alınmıştır. Bu problem tipi NP-Hard tipi problem olarak adlandırılmaktadır. NP-Hard tipi problem tanım olarak, çok sayıda kısıt içeren karmaşık problemler için kullanılmaktadır. NP-Hard tipi problemin, polinom değerlerden oluşmaması çözümünü karmaşıktır. Bu tip problemlerin çözümü geleneksel matematiğe dayalı yöntemlerle olmamakta ve sayısal analiz yöntemleri kullanılarak ise sağlıklı sonuçlar elde edilememektedir. Karmaşıklık seviyesi fazla olan ve çok sayıda kısıt içeren NP-Hard tipi problemlerin çözümü için Genetik Algoritma (GA), tabu arama, benzetimli tavlama gibi sezgisel ve meta-sezgisel algoritmalar ortaya çıkmıştır. Ameliyat odası çizelgeleme problemi ise bu tezde en önemli meta-sezgisel algoritmalarından birisi olan genetik algoritma ile çözülmüştür. Program kodlanırken görselliği de sağlamak amacıyla C# programlama dili tercih edilmiştir. Ayrıca tamir operatörünün bu tip problemlerde genetik algoritmanın bir operatörü olarak kullanılmasının etkisi ve önemi de incelenmiştir.

Anahtar sözcükler: Ameliyat odası çizelgeleme, Genetik algoritma, Kısıtlı optimizasyon, Tamir operatörü.

ABSTRACT

IMPLEMENTATION OF OPERATING ROOM SCHEDULING WITH GENETIC ALGORITHM

Tunahan TİMUÇİN

Düzce University

Graduate School of Natural and Applied Sciences, Department of Computer
Engineering

Master's Thesis

Supervisor: Assist. Prof. Dr. Serdar BİROĞUL

January 2018, 59 pages

In this thesis, the problem of the most efficient use of the Operating Rooms (ORs) which one of the most important departments of hospitals, was tackled. This type of problem is defined as NP-Hard. Complex problems involving multiple constraints are defined as NP-Hard type problems. As the NP-Hard type problem does not consist of polynomial values, the solution of such problems becomes complicated. Such problems cannot be solved by traditional methods based on mathematics. In addition, healthy results cannot be obtained by using numerical analysis methods. For the solution of NP-Hard type problems which have high level of complexity and many constraints, heuristic and meta-heuristic algorithms such as Genetic Algorithm (GA), tabu search, simulated annealing have emerged. In this thesis, the operating room scheduling problem is solved by the genetic algorithm, which is one of the most important meta-heuristic algorithms. C# programming language is preferred to provide visuality when coding the program. Furthermore, the effect and the importance of using the repair operator as an operator of the genetic algorithm in these types of problems were also investigated.

Keywords: Operating room scheduling, Genetic algorithm, Constrained optimization, Repair operator.

1. GİRİŞ

Günümüz gelişen teknolojisi, değişen ekonomisi, geniş bir yelpazeye yayılan tüketici ihtiyaçları gibi nedenlere bağlı olarak, işletmelerin de bu şartlara ayak uydurmaları ve gelir gider dengesini sağlamaları gerekmektedir. İşletmeler, gelişmelere ayak uydurduğu sürece varlıklarını devam ettirebilmektedirler. İşletmenin devamlılığı için en önemli aşama işletmenin gelir-gider ve tasarruf dengesini planlama ve düzenleyip uygulamaya geçirme aşamasıdır. Bunun için en küçüğünden en büyüğüne tüm işletme, kurum ve kuruluşlar bu planlamaya büyük önem vermektedirler ve kaynaklarını bu yöne doğru kaydırmaktadırlar.

Değişen dünya dengesinin ve artan dünya nüfusunun en önemli işletmelerinden birisi de hastanelerdir. Hastaneler ve diğer sağlık kuruluşları, günümüzde artan dünya nüfusu, büyüyen yaş ortalaması ve bunlara bağlı olarak hastanelerde ortaya çıkan ve gittikçe artan hastaların bekleme listesindeki artışla birlikte, mümkün olan en düşük maliyetle en kaliteli hizmeti verme konusunda ciddi bir baskı yaşamaktadırlar. Hastanelerde en çok kaynak ayrılan bölüm ise ameliyat odalarıdır. Dünya nüfusunun artış hızına bağlı olarak yıllık yapılan ameliyat sayısı da giderek artmaktadır.

Türkiye’de 2002 yılından 2014 yılına kadar yapılan hesaplamalarda yıllık yapılan ameliyat sayısı yaklaşık 8 kat artmış durumdadır [1]. Yapılan hesaplamalar her 6 kişiden 1’inin küçük veya büyük ameliyat geçirdiği sonucunu göstermektedir.

Bu durumda, hastaneler, toplam maliyetin yaklaşık %10 unu oluşturan ameliyat vb. cerrahi müdahalelerde, yaşanabilecek hasta sayısı fazlalığı durumunda hem hastaların bekleme süresini azaltmak, hem teknik ekip ve ekipman sayılarını doğru tespit edip maliyeti en aza indirmek adına çeşitli yollara başvurumaktadırlar. Bu yollardan birisi de yazılım programları yardımıyla çizelgeleme yöntemine başvurarak bu sorunu çözmektir. Teknolojinin ve elektronik araçların bu kadar geliştiği bu yüzyılda, sorunların çözümü için yazılım programlarını kullanılarak, insan faktöründen kaynaklanabilecek hatalar en aza indirgenirken ameliyat odası çizelgelemenin de önemi artmaktadır.

Çizelgeleme, bir ürünün üretilmesi ya da bir hizmetin sunulması için gerekli tüm faaliyet ve aşamaların istenilen düzeyde zamanlanmasıdır. Başlangıç, bitiş zamanları ve tamamlanma zamanları gibi kritik tarihlerin belirlenmesi, hangi işlemlerin ne zaman yapılacağına bir sıra haline getirilmesi, yeni yapılacak iş için hangi alanların ne zaman hazır olacağı gibi planların belirlenmesi işidir. Ek olarak çizelgeleme, taahhüt edilen işlerin istenildiği zamanda teslimini sağlamak için kullanılan bir araçtır. Çizelgeleme sayesinde; ürünlerin veya hizmetin temin edilme sürelerini en aza indirmek, esnekliği artırmak, mevcut olan kapasiteyi en uygun ve etkili şekilde kullanmak ve müşteri taleplerine en hızlı şekilde yanıt verebilmek mümkün olmaktadır.

İleriye doğru çizelgeleme ve geriye doğru çizelgeleme olmak üzere 2 çeşit çizelgeleme yapılabilmektedir. İleriye doğru çizelgelemede, problem için gereklilikler belli olduğundan itibaren en kısa sürede işe başlamak üzere çizelge hazırlanır. Bu yöntemde çizelge ve son teslim tarihi uyumlanabilmektedir. Genellikle müşteri talepleri ile başlayan ve en kısa sürede teslimi gereken işlerde kullanımı olmaktadır. Geriye doğru çizelgelemede ise, son teslim tarihinden geriye doğru bir çizelge planı oluşturulur ve bu plana göre tüm faaliyetlerin zaman aralıkları belirlenir. Çizelgeleme için gerekenler başlangıçta olmayabilmektedir. Bu 2 yöntemle beraber, geri ve ileri çizelgeleme beraber de uygulanabilmektedir.

Çizelgeleme yaparken önemli olan çizelgelemenin kalitesidir. Bir çizelgelemenin kalitesi, uygulandığı problemin sayısal olarak çözümündeki başarısı ve bu sonuçların gerçek hayatta uygulanmasındaki başarısıyla mümkündür. Çizelgelemenin kalitesini belirleyen iş öncelikleri, teslim tarihleri, öncelik kuralları gibi birçok kısıt bulunmaktadır.

Çizelgeleme yapılabilecek çok çeşitli optimizasyon problemleri bulunmaktadır. Bunlardan bazıları; ameliyat odası çizelgeleme, atölye çizelgeleme, ders çizelgeleme, hemşire çizelgeleme, hızlı tren çizelgeleme, araç ve sürücü çizelgeleme vb. birçok problem için çizelgeleme işlemi uygulanabilmektedir. Tüm bu çizelgeleme problemleri programlar yardımıyla çözülmeye çalışılmaktadır.

Ameliyat odası çizelgeleme problemi için birçok program yapılmış ve gerçeğe yaklaşılmak istenmiş olsa da program yapılırken birçok şartı yok saymaktadırlar. Bu yok sayılan şartlar nedeniyle yapılan programlar gerçek hayatla uyumlanmamakta ve uyarlanamamaktadır.

Bu soruna çözüm üretmek için ise birçok yeni algoritmalar ve yeni yaklaşımlar uygulanmış ve tam çözüm olmasa da en yakın çözüm bulunmaya çalışılmıştır.

Literatürde eniyileme problemlerinin bazıları klasik matematiksel yöntemler ve algoritmalarla çözülebilmektedir. Problem boyutunun artması ve problemin karmaşıklaşması klasik yöntemlerle çözümü imkânsız hale getirmektedir. Bu yüzden, bu karmaşık problemlerin çözümü için çeşitli sezgisel ve meta-sezgisel algoritmalara yönelimler başlamıştır. Meta-sezgisel, arama uzayının yüksek kaliteli çözümlerini kapsayan bölgelerinde aramaları gerçekleştirmek için probleme özgü sezgisellere rehberlik etmek amacıyla tasarlanan genel amaçlı sezgisel yöntemdir [2]. Meta-sezgiseller basit yerel arama algoritmalarından gelişmiş ve karmaşık öğrenme işlemlerine kadar geniş bir yelpazeyi kapsarlar. Bu algoritmalar; genetik algoritma, benzetimli tavlama, tabu arama, karınca kolonisi, değişken komşu arama gibi algoritmalar [3].



2. PROBLEM TANIMI

Sağlık sektörü, gelişmişte olan ve gelişmiş tüm ülkelerde günümüzün en önemli endüstrilerden birisi haline gelmektedir. Dünyanın birçok ülkesinde yeni hastalık türleri ortaya çıkmaktadır ve dünya nüfusunun artmasıyla birlikte bu hastalıklardan etkilenen insan sayısı da önemli ölçüde artmaktadır.

Günümüzde kaynakların sınırlı olması, insanların alım gücünün yetemeyeceği düzeyde fiyatlara sahip ilaç ve medikal malzemeler hastaneleri de zor durumda bırakmakta ve bütçe kısıtlamasına yönlendirmektedir. Hastaneler bu durumu aşmak için kendi birimlerinde revizyonlara gitmektedir. Hastaneler, öncelikle en çok kaynak ayrılan bölümlerinden birisi olan ameliyathanelerin düzenlenmesi ve doğru kullanılmasını sağlamak amacıyla çeşitli yollara başvurmuşlardır. Bu noktada, ameliyathanelerin boş zamanlarını azaltmak, ameliyathaneleri doğru planlamayla aynı sürede daha etkin kullanmak adına yazılım programlarından faydalanmak istemektedirler.

Bu bölümde, bu tezde ele alınan Ameliyat Odası Çizelgeleme probleminin temelleri sunulmuştur. Sunulan bu problemin çözülmesi, günümüzün en önemli işletmelerinden birisi olan hastanelerde, toplam maliyetin en aza indirilerek, seçmeli hastalar için bekleme süresini en aza indirecek bir ameliyat programı yapılmasına yardımcı olabilecek bir program çözüm modülü gerçekleştirilmiştir.

Ameliyat Odası Çizelgeleme problemlerinin, kendine özgü unsurları ve çözülmesi gereken koşullarının iyi anlaşılması gerekmektedir. Bu sayede, ameliyathaneler için gerekli tüm özellikler anlaşılıp bu özelliklere uygun çözüm yöntemleri uygulanmalıdır.

Alınacak kararların acil hasta girişlerine ve mevcut kaynaklarla ilgili kısıtlamalara uygun olması çözümü daha sağlıklı hale getirme konusunda başarıyı artıracaktır. Burada belirtilen kısıtlamalardan öncelikli olanlar; ameliyat odalarının giriş-çıkış zamanları, ameliyat öncesinde ve sonrasında yatakların kapasite durumları (çizelgelemede dikkate alınma durumuna göre önemli olmaktadır), ameliyat sırasında kullanılacak ekipman durumu gösterilmektedir. Bir hastanede ameliyatlar çeşitli zaman aralıklarında planlanabilmektedirler. Bu zaman aralıkları günlük, haftalık (pazartesten cumaya), ya da orta vadede bir plan hazırlanabilmektedir. Hastaneler kendi kapasite ve çalışan eleman

durumlarına göre ameliyatları zaman periyotlarına bölmek durumundadırlar. Aksi halde zaman sınırlarının aşılması ve fazla mesai kavramının oluşması zaman zaman zor durumda kalmalarına neden olabilmektedir.

Hastanelere 2 tür hasta gelmektedir; randevu sistemi vasıtasıyla gelenler ve acil durum olarak gelen hastalar. Seçmeli hastalar yani randevu talep eden hastalar belirli olduğundan bu tür hastalar için rahat bir şekilde zaman planlanması yapılabilmektedir. Ancak acil gelen hastalar için geldikten sonra durumuna göre o anda tedavisinin yapılması gerektiğinden planlanması yapılamamaktadır. Bu tezde, acil hastaların durumu, planlamasının yapılması mümkün olmamasından dolayı, acil hastaların farklı bir hastaneye yönlendirilmesi veya acil durumlar için sürekli hazır tutulan ve bu tez çalışmasında çizelgelemeye dâhil edilmeyen başka bir ameliyat odasına yönlendirildikleri kurgusuyla hareket edilmiştir ve bir kısıtlama veya kıstas olarak dikkate alınmamıştır.

Bu tezde, planlamaya dâhil olacak hastaların sayısı ve hangi ameliyatı olacakları bellidir ve bu noktada belirsiz bir durum yoktur. Her ameliyatın süresi farklılık gösterebildiğinden ve ameliyat süreleri de ameliyat sırasında yaşanabilecek ekstra durumlara göre değişebileceğinden dolayı, çizelgeleme yapılabilmesi için ameliyat saatleri, ameliyat odasının sonraki ameliyat için hazır hale getirilme süreleri ortalama ameliyat süreleri dikkate alınarak sabitlenmiştir. Ameliyathaneler, ameliyat özelliklerine göre farklı özelliklere ve ekipmanlara sahip olabilmektedirler. Ancak, bu tezde, ameliyathanelerin bu şekilde kısıtlanmasının boş zamanı artırabileceği, daha sonradan dâhil edilebilecek acil hastaları olumsuz etkileyebileceği gibi özellikler dikkate alınarak, tüm ameliyat odalarının aynı ve belirlenen ameliyat çeşitlerinde ortak olarak kullanılacak ekipmanlarla donanmış olduğu kabul edilmektedir.

2.1. AMELİYAT ODASI KARAR SEVİYELERİ VE ÇİZELGELEME TÜRLERİ

Ameliyathanelerin planlaması problemi çoğu çalışmada Şekil 2.1'deki gibi 3 farklı ve ilgili alt probleme ayrılmaktadır.

(i) Stratejik Seviye; toplam gelirleri en üst düzeye çıkarmak ya da toplam maliyetleri en aza indirmek ya da mevcut ameliyathanelerin zamanının farklı cerrahlar üzerinde nasıl bölünmesi gerektiğinin belirlendiği seviyedir.

(ii) Taktiksel Seviye; ameliyat odası kullanımını en etkin ve etkili şekilde kullanmak için çizelge zamanına göre (haftalık, günlük vs.) ameliyatların cerrahlara atandığı seviyedir.

(iii) Operasyonel Seviye; Her ameliyatın bir başlangıç, bir bitiş, bir bekleme süresi olmaktadır. Operasyonel karar seviyesinde bu hastalar seçilerek listeler belirlenmektedir.



Şekil 2.1. Ameliyat odası karar seviyeleri.

Ameliyat odası çizelgeleme problemini ele alan birçok inceleme makalesi bulunmaktadır [4]–[6]. 2013 yılında yapılan bir çalışmada 3 ameliyat odası çizelgeleme problemi incelenmiştir. Ek olarak, bu problemler için bütünleşmiş bir araştırma çerçevesi ve planlama yöntemi sunulmuştur [7].

Ameliyat odalarının planlanmasında, blok çizelgeleme, modifiyeli blok çizelgeleme ve açık çizelgeleme olmak üzere 3 stratejiden birisi kullanılabilir [8].

Blok çizelgeleme; ameliyat odası kaynaklarının bir cerrah ya da cerrah grubuna belirli gün ve saatlerde ayırma işlemidir. Zaman bloklarında cerrahlar düzenlenmiştir ve atanan bu cerrahların yerlerine ayrılan gün ve saatte başka bir cerrah atanmamaktadır. Blok çizelgelemenin aksine, açık çizelgeleme; ameliyat odası uygun olduğunda, uygun bir cerrahın araya atanmasına izin vermektedir. Blok çizelgelemeden önemli bir farkı; açık çizelgelemede cerrahlar bir ameliyat için herhangi bir iş gününü seçebilmektedirler. Bu nedenle bu açık çizelgelemeye “Herhangi bir İşgünü” (any workday) stratejisi de denmektedir. Bu sayede blok çizelgelemeden daha esnek hale gelmektedir. Modifiye edilmiş blok çizelgeleme ise; blok çizelgeleme ve açık çizelgelemenin iyi yönlerinin birleştirilerek daha esnek ve gelişmiş bir yapı elde edilmek üzere geliştirilmiştir. Kullanılmadığında engellenen ameliyat odası zamanı diğer doktorlar tarafından kullanılabilirliğinden, daha efektif bir kullanımın yolunu açmaktadır.

3. AMELİYAT ODASI ÇİZELGELEME VE OPTİMİZASYONU

Ameliyat odalarının, hastanelerin en önemli ve en çok kaynak ayrılan birimlerinden birisi olduğundan bahsetmiştik. Bu bölümde, ameliyat odalarının çizelgeleme problemlerine literatürde nasıl çözümler getirildiği, problemin çizelgenmesi dışında hangi parametrelerin bu problem için önemli olduğu ile ilgili literatürün incelenmesi yer almaktadır.

Marques ve ark. ameliyathane mesleğini en üst düzeye çıkarmak, ameliyathanelerin en verimli şekilde kullanımını sağlamak, hastane masraflarını en aza indirmek ve kârı arttırmak amacıyla haftalık cerrahi ve seçmeli ameliyatların ayırma zamanlaması için bir tam sayılı doğrusal programlama modeli önermişlerdir [9]. Onlar bu makalelerinde, aynı Guido ve ark. [10] gibi her cerrah için günlük ve haftalık çalışma çizelgesi sınırlaması kısıtını uygulamışlardır ve cerrahların bu saatin üzerinde çalışmamalarını bir kısıt olarak uygulayıp çalışmalarını buna göre oluşturmuşlardır. Doğrusal programlama modelini öneren aynı yazarlar daha sonra, genetik algoritmalara dayanan bir çözüm yaklaşımı geliştirmişler, ancak çözülmüş örnekler hakkında hiçbir ayrıntı bildirilmemişlerdir [11].

Khambhammettu ve Persson, 3 aşamaya ayırdıkları ameliyat odası çizelgelenmelerinde (Cerrahi öncesi, cerrahi sırası ve cerrahi sonrası), karmaşık tamsayılı lineer programlama yöntemiyle bir simülasyon geliştirmişler ve sonuçları Friedman testiyle analiz etmişlerdir ve önerilen modeli değerlendirmek için yerel bir hastaneden toplanan gerçek verileri kullanmışlardır [12]. Molina-Pariente ve ark. haftalık bir ameliyat odası çizelgeleme yaparak 17 meta-sezgisel algoritma ile kıyaslamasını yapmaya çalışmışlar ve en verimli algoritmayı sunmaya çalışmışlardır. Bunu yaparken daha önce sadece 1 kez denenen klinik önceliklendirme yöntemini uygulamışlar ve bekleme süresini en aza indirmeye çalışmışlardır [13]. Guido ve Conforti, hastane ameliyathane takımlarını verimli bir şekilde planlamayı ve yönetmeyi amaçlayan çok amaçlı bir tamsayılı doğrusal programlama modeli önermiş ve genetik algoritmaya dayalı hibrit bir algoritma geliştirerek ameliyat sırasındaki taktiksel ve operasyonel aşamaları birleştirmeyi amaçlamışlar [14]. Xiang ve ark. ameliyat planlaması sorununu, rolleri, uzmanlıkları,

niteliği ve bulunabilirliği gibi durumları gerçek hayatta hemşire listesi kısıtlamaları ile bütünleştirmişlerdir. Bir matematiksel model ve karınca kolonisi algoritmasını bu probleme uygulamış ve bu probleme hemşirelerin uygunluk durumunu dâhil etmişlerdir [15]. Latorre-Núñez ve ark. ameliyat odası planlamasında ilk kez ameliyathaneleri, ameliyat sonrası anestezi düzelmesini, ameliyatın gerektirdiği kaynakları ve acil ameliyatlara ele almışlar. Matematiksel model öneren Bouguerra ve ark. gibi küçük boyutlu problemler için bir tamsayı doğrusal programlama ve daha büyük boyutlu örnekler için ise bir genetik algoritma ve yapıcı-buluşsal tabanlı bir sezgisel algoritma geliştirmişlerdir [16], [17]. Saadouli ve ark. Ameliyat odası planlamasını ameliyat odası ve kurtarma yatakları olmak üzere 2 kaynak üzerinde değerlendirmişlerdir. Problemin çözümü için, gün içi işlemlerin seçimleri için bir sırt çantası modeli, ameliyathanelerin zamanlanmasını en verimli halde gerçekleştirmek için ise bir karışık tamsayı programlama modeli kullanmışlardır [18]. Xiang ve ark. esnek iş ofisi planlama problemi ve ameliyat odası problemi arasındaki benzerlikleri gözlemleyerek, bu tür zamanlama problemlerini etkin bir şekilde çözmek için karınca kolonisi algoritmasını geliştirmişlerdir [19]. Aringhieri ve ark. sabit sayıda ameliyathane ve ameliyat sonrası yatağı paylaşan çeşitli özellikleri içeren bir ameliyat bölümü verildiğinde, müşterek ameliyat odasını (OR) planlamayı ve gelişmiş çizelgeleme problemini incelemişlerdir. Ameliyat odası ve özellikle hafta sonu konaklaması olan karar seviyelerini aynı anda planlamayı amaçlamışlardır. Problemin NP-Hard olduğunu kanıtlamak üzere bir 0-1 doğrusal programlama formülasyonu ve daha sonrasında problemin çözümü için iki seviyeli (Tabu ve Greedy) bir meta-sezgisel algoritma geliştirilmiştir [20]. Razmi ve ark. ise daha çok hastanelerin benzersiz elemanlarını (C-Arm Görüntüleme Ekipmanı) planlamaya odaklanmışlar. Bu çalışmada, benzersiz bir ekipman kısıtlaması altında Ameliyathane (OR) planlaması için bir model açıklanmaktadır. İlk aşamada, seçkin ameliyatlara benzersiz bir ekipman kullanmanın belirsizliği altında programlama, İkinci aşamada, acil ameliyatlara ve üçüncü seviyede cerrahlar için bu eşsiz ekipmanı kullanırken bir katsayı faktörü getirilmiştir [21]. Wang ve ark. standart çizelgeleme kısıtlamalarına ek olarak, gerçek yaşamda karşılaşılan insan ve maddi kaynaklara ilişkin pek çok ilave kısıtlama da dikkate alınmalıdır fikrini ortaya atmış ve karışık tamsayı ve kısıtlı programlama tekniklerini kullanarak iki model geliştirmişlerdir [22].

Riise ve ark. kendi geliştirdikleri “Uyarlamalı yapı ve iyileştirme” arama algoritmasını uygulamışlar ve günlük, haftalık ve hasta kabul şeklinde üç farklı durumda

uygulamışlardır [23]. Dios ve ark. bir bekleme listesindeki hastaların ameliyatlarına tarihlerin ve ameliyathanelerin atanması ile ilgili kararları desteklemek için bir dizi optimizasyon prosedürünü (hem kesin hem de yaklaşık) içeren ve şu anda İspanya'nın en büyük hastanelerinden birinde, birkaç cerrahi birimde kullanımda olan cerrahi planlama için bir Karar Destek Sistemi (KDS) sunmuşlardır [24]. Murray ve ark. ise eğitim amaçlı ameliyat odası simülasyonu geliştirmişlerdir [25]. Standart convex-hull ve büyük-M tekniklerini tamsayı ya da karmaşık-tamsayı lineer programlamaya uygulayan Castro ve Marques ise bekleme listesinden öncelikli hastaların seçimi üzerine kısa vadeli bir program üzerine odaklanmışlardır [26]. Abedini ve ark. geleneksel olarak ameliyat odası blok zamanlaması için önemli bir yöntem olan bin-packing modelinde başlangıç dizisini oluşturmak için çok adımlı bir yaklaşım ve öncelikli bir süreleme kuralı önermişlerdir [27]. Bu çalışmalar dışında problemin farklı yönünü ele alan çalışmalar da vardır. Örneğin, Beroule ve ark. daha çok ameliyat odası ekipmanlarının sterilizasyonunu ön planda tutmuşlardır [28]. Bununla beraber Alamedaa ve Macario çalışmalarında hastane yöneticisinin rolünü ön plana çıkarmış ve bu alanda çalışma yelpazesini genişletmişlerdir [29]. Veen-Berkx ve ark. tüm elektif OR'lerde OR kullanımı açısından acil cerrahi müdahale kapasitesinin poliçe sonuçlarını değerlendirmişlerdir. Amaç yine diğerleri gibi daha efektif, az ameliyat iptali olan, zaman aşımının oldukça az olduğu bir plan oluşturmaktır [30]. Riet ve Demeulemeester, makalelerinde hem seçmeli hem de seçmeli olmayan hasta kategorilerinin dâhil olduğu OR planlaması hakkındaki literatürü gözden geçirmektedirler [31]. Siqueira ve ark. periyodik cerrahi tahsisi tasarlayan bir optimizasyon problemi sunmuşlardır. Tüm cerrahi operasyonların zamanında yapılacağı şekilde, uzun süreli hasta girişlerini ve salınımlarını dengeleme bakış açısıyla periyodik cerrahi tahsisat planı ve kurtarma yatağı kullanım planını tasarlayan bir optimizasyon yöntemi sunmuşlardır [32]. Roshanaei ve ark. ise Benders ayrıştırma yaklaşımı, IP (Internet Protokolü) modelini bir tam sayı konumuna ve sırt çantasına dayalı ayırma ana soruna ve birden fazla paketleme IP alt soruna dönüştürür. LBBD'lerin IP + Gurobi'den(Gurobi Optimizer (Gurobi, Inc.) v5.63) 10-100 kat daha hızlı olduğunu ve en uygun çözümleri bulmada daha başarılı olduklarını da göstermişlerdir [33]. Landaa ve ark. monte carlo simülasyonu ile birlikte mahalle arama tekniklerinin potansiyelini kullanan bir hibrit iki fazlı optimizasyon algoritması geliştirilmişlerdir. Her alt problem için bir tamsayılı doğrusal stokastik formülasyon verilmiş ve cerrahi sürelerin doğasında olan belirsizliğini hesaba katarak, ortak ilerleme ve tahsis çizelgeleme problemini çözmek için etkili bir algoritmik çerçeve sağlamayı amaçlamışlardır [34]. Xiao ve ark. ise, Sağlık

hizmetlerine yönelik artan talebin bir sonucu olarak, Çin'in büyük şehir hastaneleri, aşırı derecede karmaşık bir hal almış ve hassas ve karmaşık ameliyathane planlama problemleri ile karşılaşmıştır. Aynı zamanda bekleme listelerinde patlama yaşanması nedeniyle iptallerin ve iptallere bağlı maliyetlerin olumsuz etkilenmesinin önüne geçmek amacıyla, örnek ortalama yaklaşımı kullanılarak çözüm elde etmeye çalışmışlardır. Örnek ortalama yaklaşım yöntemleri ve karşılık gelen optimizasyon teknikleri kullanılarak üç aşamalı hasta kayıt problemini çözmüşlerdir [35]. Jebali ve Diabat, yoğun bakım ünitesi yataklarının değerlendirilmesi, ameliyattan sonra hastanın yoğun bakım yatağı gerektirdiği durumlar dâhil ameliyathane planlaması için özellikle önemli olduğunu savunmuşlar ve modeli çözmek için özellikli bir Örnek Ortalama Yaklaşım algoritması geliştirmişlerdir [36].

Günümüzün NP-Hard tipi olan çizelgeleme problemleri; araştırmacıları hızlı, kolay ve gerçeğe en yakın çözümü veren yöntem ve algoritma arayışına itmiştir. Bu noktada evrimsel yaklaşımlar içinde kabul edilen genetik algoritma ve hibritleri önemli bir çözüm algoritma yapısı olarak ön plana çıkmıştır. Özellikle sert eniyileme teknikleri yerine daha yumuşak eniyileme ve evrimsel algoritmaların kullanımı önem kazanmıştır.



4. GENETİK ALGORİTMA

Genetik Algoritma, kısa tanım olarak parametre kodlama esasına dayanan ve rastsal arama teknikleriyle çözüme ulaşmayı hedefleyen bir eniyileme algoritması olarak tanımlanmaktadır. Bir veri kümesi içerisindeki en özel ve en iyi çözümü bulmak için kullanılır.

Genetik algoritma, doğadaki canlıların üreme şekilleri dikkate alınarak geliştirilmiş ve kodlanarak algoritma halini almıştır. Canlılardaki gibi güçlü ve iyi olanın öne çıkıp yaşamını devam ettirmesi, güçsüz olanın ise zamanla yok olması prensibiyle hareket etmektedir. Bir genetik algoritma, bir problem için geçerli olan tüm olası çözümleri değerlendirmez, çözüme daha hızlı bir şekilde gidebilmek için problemin sadece gerekli parçalarını değerlendirir. Bu nedenle GA günümüz küçük ve karmaşık olmayan problemler ve aynı zamanda büyük ve karmaşık problemler için uygun bir algoritma olmaktadır.

Genetik algoritmadan ilk defa Bagley (1967) bahsetmiştir. Bu durum genetik algoritmanın başlangıcı kabul edilse de genetik algoritmanın asıl evrimi, John H. Holland'ın, makine önerisinden sonra gerçekleşti. Holland, bir makineye evrim sürecinin başarılı bir şekilde nasıl aktarılacağına önerisinde bulunmuştur. Holland, genetik algoritmanın prensiplerinin yapay zekâ problemlerine nasıl uygulanacağını ise "Adaptation in natural and artificial system" isimli kitabında açıklamıştır [37]. Holland bu çalışmada GA'yı soyut bir evrim modeli olarak ele almıştır. Bunu yaparken Holland, çözümlerden yeni ve en iyi bireyleri elde etmek için, evrim modelinin, bilgisayarları öğrenme, yeniden üretim, değiştirme (mutasyon, çaprazlama) ve uyarlama becerilerini içeren mekanik yapısını kopyalamayı amaçlamıştır. 1989 yılında Hollandalı bir öğrenci olan Goldberg "Genetic algorithms in Search Optimization and Machine Learning" isimli kitabında gaz boru hattı denetimine genetik algoritma uygulayarak akademik yönden GA'yı bir adım öne taşımıştır [38].

Günümüzde GA aşağıdaki alanlarda ve bu alanların alt alanlarında gerek teorik gerekse pratik olmak üzere aktif olarak kullanılmaktadır.

- Optimizasyon problemleri

- Çizelgeleme problemleri
- Rotalama problemleri
- Matematik problemleri
- Mekanik öğrenme
- Otomatik programlama ve bilgi sistemleri
- Oyun programlama
- Finans ve pazarlama
- Popülasyon genetiği [39].

4.1. TEMEL GENETİK ALGORİTMA

Genetik algoritma kodlanırken algoritmanın temel adımlarının ve bunun yanında varsa probleme özgü olarak geliştirilen genetik operatörlerin bir döngü içerisinde uygulanması gerekmektedir. Kodlamaya başlamadan önce tüm adımların, kriterlerin, hangi operatörlerin kullanılacağı gibi tüm planın önceden yapılması algoritmanın kalitesini arttırmaktadır. Genetik algoritma genel adımları ve probleme özgü olarak geliştirilebilecek bazı metot ve operatörlerin neler olduğu aşağıda adım adım gösterilmiştir.

1. Arama uzayındaki olabilecek tüm çözümler dizi olarak kodlanır.
2. Probleme göre değişebilmek ile beraber genellikle rastgele bir çözüm kümesi oluşturulur ve başlangıç popülasyonu olarak kabul edilir.
3. Her bir dizi için uygunluk değerleri hesaplanır. Bu değerler her dizinin (kromozomun) kalitesini veya sağlık durumunu göstermektedir.
4. Diziler (kromozomlar) belirli bir olasılık değerine göre rastgele olarak seçilip eşleşme havuzuna gönderilir.
5. Eşleşmesi yapılan kromozomlara çaprazlama operatörü uygulanır.
6. Seçilen kromozomlara mutasyon operatörü uygulanır.
7. Kaybolan genleri ve gereği gibi olmayan genler için probleme uygunsa tamir operatörü uygulanır.

8. Oluşan bu yeni nesil bireyler ile yeni popülasyon oluşturulur.
9. Oluşturulan yeni popülasyon ile 3. Adıma dönülür.
10. Bu işlemler ister belirlenen nesil sayısına ulaştığında, isterse başka bir sonlandırma kriteri uygulanarak sona erdirilip, amaç fonksiyonuna göre en iyi birey çözüm olarak kabul edilir.

Genetik algoritmada genel olarak 4 operatör kullanılmaktadır.

- I. Parametreleri kodlamak için kullanılan operatör
- II. Üreme Operatörü
- III. Çaprazlama Operatörü
- IV. Değişim (Mutasyon) Operatörü

Bu 4 operatöre ek olarak bir de bu tez çalışmasında gerçekleştirilen ameliyat odası çizelgeleme gibi kısıtlı optimizasyon problemlerinde uygulanması gereken bir operatör olarak tamir operatörü bulunmaktadır. Bireyler (Kromozomlar) genetik algoritma operatörleri olan çaprazlama ve mutasyona uğradıktan sonra oluşan yeni kromozomlarda kaybolan veya doğal yapısı bozulan genler ortaya çıkmaktadır. Çözüm uzayından uzaklaşmaya neden olan bu sorunun giderilebilmesi ancak; her bir nesil tamamlandığında bireylerin bir de tamir (düzeltme) operatöründen geçirilmesi ile mümkün olmaktadır. Bu nedenle bu tezde diğer genetik operatörlere ek olarak bir de tamir operatörü kullanılmaktadır.

4.1.1. Parametrelerin Kodlanması

Bir probleme genetik algoritma uygulanacaksa, o problemin parametrelerinin birbiriyle aynı değer tipinde kodlanmış bitler dizisi olmaları gerekir. Daha sonra bu parametre tipi genetik algoritmaya tanıtılarak işleme girmesi sağlanmaktadır. Bu sayede dizilerden her biri probleme ait çözüm uzayında rastgele bir noktayı simgelemektedir (15.30, “Genetik”, 01100011,1453 vs.). Parametreler kodlanırken en önemli noktalarda birisi, probleme uygun kodlama yöntemini oluşturmaktır. Bu bölümde önemli kodlama yöntemlerinden olan ikili, permütasyon, ağaç, nümerik ve alfa-nümerik kodlama yöntemlerine değinilmiştir.

4.1.1.1. Binary (İkili) Kodlama

İkili kodlamada, her kromozom 0 ve 1 dizilerinden oluşmaktadır. Bu dizilerin her biti,

çözüm uzayının belirli bir karakteristiğini temsil eder ya da tüm dizi bir karakteri temsil eder. Parametre kodlamada en sık kullanılan yöntemlerden birisidir.

İkili kodlama örnekleri;

{0 1 0 1 1 0},

{1 1 0 0 1 1 1 0 1},

{1 0 0 0 0 0 0 1 1 1}

4.1.1.2. *Permütasyon Kodlama*

Permütasyon kodlama genellikle düzenleme problemlerinde kullanılmaktadır. Özellikle gezgin satıcı ve çizelgeleme problemleri için kullanımı uygundur. Bu kodlama türünde her bir kromozom, belirli bir sırada sayılar takip edilerek oluşturulmaktadır.

Permütasyon kodlamaya örnek olarak;

Kromozom 1 → 1 5 3 2 6 4 7 9 8

Kromozom 2 → 8 5 6 7 2 3 1 4 9

Daha çok sıralama problemlerinde kullanılmaktadır. İçerisinde gerçek sırayı tutan bazı kromozomlarda, kromozomların tutarlılığı için çaprazlama ve mutasyonlarda düzeltmeler yapılması gerekmektedir. Bu durumlarda permütasyon kodlama tercih edilmektedir.

4.1.1.3. *Ağaç Kodlama*

Ağaç kodlamada her kromozom, fonksiyonların, programlama dilindeki komutların ya da diğer nesnelere birinin ağacı durumundadır. Ağaç kodlama, ağaç şeklinde kodlanabilecek herhangi bir yapı için veya evrimleşen programlar için uygundur.

Parametreler harfler ile (alfa kodlar), sayılar ile (nümerik kodlar) veya hem harf hem sayı ile (alfa-nümerik kodlar) kodlanabilmektedir. Şekil 4.1, nümerik kodlamaya örnek olarak gösterilmektedir.

Nümerik Kodlama : Nümerik Kodlamada Sözcükler sayı ile ifade edilir;

Genel Cerrahi	= 100,
Pediyatri	=101,
Kulak Burun Boğaz	=102,
Göz Hastalıkları	=103,
Endoskopi ve Radyoloji	=104,
Ortopedi	=105

Şekil 4.1. Nümerik kodlama.

Şekil 4.2, sözcük ve sayıların birlikte kullanımı ile oluşturulan kodlama yöntemi olan alfa-nümerik kodlamayı ifade etmektedir.

Alfa-Nümerik Kodlama : Alfa-Nümerik Kodlamada Sözcükler ve sayılar bir arada kullanılır;	
Genel Cerrahi	= A-100,
Pediatri	=A-101,
Kulak Burun Boğaz	=A-102,
Göz Hastalıkları	=B-101,
Endoskopi ve Radyoloji	=B-102,
Ortopedi	=B-103

Şekil 4.2. Alfa-nümerik kodlama.

4.1.2. Başlangıç Popülasyonunun Oluşturulması

Olası tüm çözümlerin kodlandığı bir çözüm grubu oluşturulur. Bu gruba popülasyon denir. Popülasyonu oluştururken; popülasyonu oluşturan kromozomlar probleme özgü kodlama teknikleri ile kodlanmaktadır. Bu yöntemlerden bazıları ikili (binary) kodlama, tamsayı kodlama, metin kodlama vb. kodlama teknikleridir. Popülasyon oluşturmaya örnek verilecek olursa; Popülasyon oluşturulurken örneğin; Şekil 4.3'te gösterildiği gibi, sadece 0 ve 1 bitlerinin oluşturduğu ikili (binary) kodlamada, rastsal sayı üreticisi ile rastgele bir sayı üretilip 0.5 ten küçük veya büyük olma durumuna göre 0 veya 1 atanarak popülasyon oluşturulabilmektedir.

1. Kromozom
1 1 1 0 1 1 0 0 0 0 1 1 1 0 0 0 0
2. Kromozom
1 1 1 1 1 1 1 1 0 1 1 0 1 1 1 0 0
3. Kromozom
0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 1

Şekil 4.3. İkili kodlama örneği.

4.2. GENETİK ALGORİTMA NESİL DÖNGÜSÜNÜN OLUŞTURULMASI

Başlangıç popülasyonu oluşturulduktan sonra, döngünün başlatılabilmesi için ilk adım her bir kromozomun uygunluk değerlerinin veya diğer bir anlamıyla her bir kromozomun kalitesinin hesaplanması adıdır. Çözümü aranılan her problem için bir kalite değeri olacağı düşünülürse, “her probleme ait uygunluk fonksiyonu da olmaktadır” sonucu elde edilebilir. Bireyin kalitesi ne kadar iyiyse uygunluk değeri de o kadar yüksektir. Doğal

seçilimden hareketle uygunluk değeri yüksek olan bireyin sonraki nesilde devam etme ve temsil edilme oranı da yüksek olmaktadır. Uygunluk değeri farklı kıstaslarla hesaplanabilmektedir. Örneğin; her bir bireye uymadığı kısıt karşısında belirli bir ceza değeri verilir ve bu durumda ceza değeri en düşük olan bireyin uygunluk değeri de en yüksek çıkması gerekmektedir.

İkinci adım, çoğalma işlemidir. Çoğalma işlemi, bireyleri seçme, seçilen bireyleri bir eşleştirme havuzuna kopyalama ve bu bireyleri çiftler halinde gruplandırma işlemlerini içermektedir. Diziler belirlenen amaca göre havuza kopyalanmaktadır. Böylece kaliteli ve sağlıklı bireylerin sonraki nesillerde temsil edilme oranı artmaktadır. Bu yöntemin asıl amacı uygunluğu çok daha yüksek olan adaylara çoğalma fırsatı vermek ve nesli iyileştirmektir. Bir bireyin seçilme şansı uygunluk fonksiyonundan elde edilen bireye ait değer ile belirlenir. Genetik algoritmada kullanılan çeşitli seçim yöntemleri vardır. Bunlardan bazıları; deterministik, rulet tekerleği (roulett wheel), rastgele, turnuva. Bu tezde rulet tekerleği seçim yöntemi uygulanmıştır.

Üçüncü adımda eşleştirilen kromozomlara çaprazlama operatörü uygulanır. Çaprazlama operatörü, daha önce oluşturulmuş olan gen havuzundan uygunluk fonksiyonu değerlerine göre, tercih edilen seçim yöntemi vasıtasıyla seçilen iyi bireylerden yeni ve daha iyi bireyler oluşturmak amacıyla uygulanan operatördür. Çaprazlama operatörü genetik algoritmanın performansını etkileyen baş etmenlerden birisidir.

Bu tezde, rulet tekerleği yöntemiyle seçilen iki birey %50-%90 arası değişen çaprazlama oranlarıyla ayrı ayrı denenmiştir. Çaprazlama ve mutasyon operatörleri uygulanırken oranların iyi belirlenmesi gerekir. Düşük bir oran seçilmesi gende kaybolmalar meydana getirebileceği gibi çok yüksek seçilmesi durumunda ise çok fazla değişim olması ve çözümden uzaklaşılmasına neden olabilmektedir. Değişim oranının iyi seçilmesinin yanında çaprazlama yönteminin de iyi seçilmesi gereklidir. Çaprazlama yöntemi olarak uygulanan bazı yöntemler; Tek noktalı çaprazlama, iki noktalı çaprazlama ve çok noktalı (uniform) çaprazlama olarak gösterilebilir. Bu tezde, probleme en uygun olarak çok noktalı çaprazlama yöntemi tercih edilmiştir.

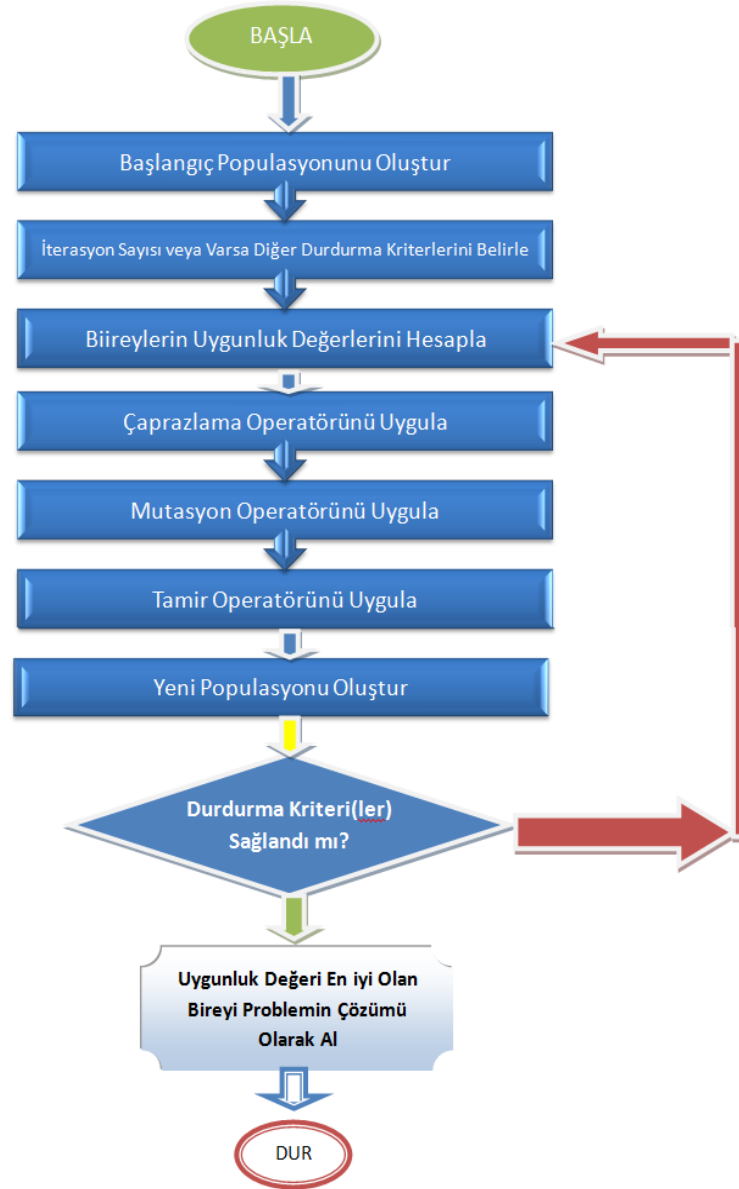
Çaprazlama operatörü mevcut olan gen havuzunda bireyler arasında gen potansiyelini araştırmak amacıyla kullanılmaktadır. Popülasyonun çözüme ulaşmak için gereken kodlanmış tüm bilgiyi içermemesi durumunda mevcut bulunan kromozomlardan yeni kromozomlar üretme yeteneğine sahip yeni teknikler, yeni operatörler gerekir. Bu

noktada, dördüncü adım olarak mutasyon operatörü devreye girmektedir.

Çaprazlamada olduğu gibi mutasyonda da tek noktalı, iki noktalı ve çok noktalı (uniform) mutasyon uygulama yöntemleri bulunmaktadır. Bu tezde, probleme daha uygun olarak çok noktalı (uniform) mutasyon tercih edilmiştir. En verimli mutasyon oranını tespit etmek adına ise algoritmaya %0.1 ile %0.15 arası farklı mutasyon oranları denenmiştir.

GA'da çaprazlama ve mutasyon operatörlerinin uygulanmasından sonra genlerde bozulmalar, kaybolan, çözümden ve çözüm uzayından uzaklaşmalar meydana gelmektedir. Bu bozulan veya kaybolan genlerin yeni nesil oluşturulmadan yerlerine konulması gerekmektedir. Bu noktada, tamir (düzeltme) operatörü devreye girmektedir. Bu tezde, diğer genetik operatörlere ek olarak tamir operatörü de uygulanmıştır.

Tüm operatörler uygulandıktan sonra, yeni neslin oluşturulması adımı gerçekleştirilir. Yeni nesil, çaprazlama, mutasyon ve tamir operatörleri uygulandıktan sonra oluşmakta olan yeni nesil bireylerin oluşması işlemidir. Yeni nesil oluştuktan sonra bireyler yine uygunluk değerinin bulunması basamağına dönüp aynı döngüyü devam ettirmektedirler. Bu durumda döngü bir sonlandırıcıya ihtiyaç duymaktadır. Döngüyü sonlandırmak için çeşitli yollar söz konusudur. Programın kodlanma durumuna göre iterasyon sayısı, belirli bir uygunluk değerine ulaşılması, ceza değerinin istenilen seviyeye ulaşması gibi durdurma koşulları uygulanabilmektedir. Şekil 4.4'te genetik algoritmanın akış diyagramı verilmiştir.



Şekil 4.4. Genetik algoritma akış diyagramı.

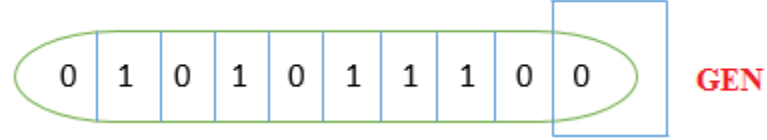
4.3. GENETİK ALGORİTMADA TEMEL KAVRAMLAR

Genetik algoritmanın doğru uygulanabilmesi için, genetik algoritmayı oluşturan temel kavramların iyi bilinmesi gerekir. Bu bölümde bu kavramlar ve nasıl oluştuğu anlatılmaktadır.

4.3.1. Gen

Gen bir popülasyonda kendi başına değeri olan, bilgi taşıyan en küçük birimdir. Birleşerek kromozomları oluşturmaktadırlar. Genlerin sahip olabileceği bilgi programı

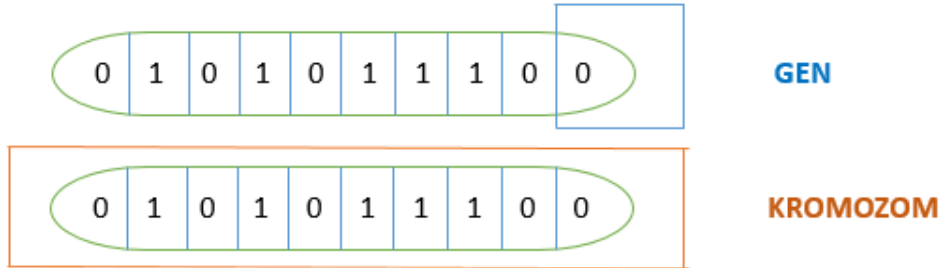
kodlayan kişiye ve probleme göre deęişim gösterebilmektedir. Bu bilgi metin içerebileceęi gibi, ikili kodlama, tamsayı kodlama, onaltılık tabanda kodlama, virgüllü kodlama gibi birçok şekilde kodlanabilmektedir. Bu tezde, genler tamsayı deęer kodlama yöntemiyle oluşturulmuştur. Şekil 4.5, programda yer alan bir kromozom parçasını göstermektedir. Kromozom, genlerden oluşmaktadır. Şekil 4.5'te hücrelerde yer alan her bir “0” ve “1” bilgisi geni ifade etmektedir.



Şekil 4.5. Gen yapısı.

4.3.2. Kromozom

Kromozomlar algoritmadaki en önemli bölümdür. Bu nedenle iyi oluşturulması önemlidir. Kromozomlar, bir ya da daha çok genin bir araya gelmesiyle oluşmaktadır. Bir kromozom, bir birey olarak da ifade edilebilmektedir. Genlerin bir araya gelip kromozomları oluşturduğu gibi, kromozomlar da bir araya gelerek popülasyonu oluşturmaktadırlar. Şekil 4.6, ikili sistemde kodlanmış bir kromozom yapısını ve gen ile farkını göstermektedir.

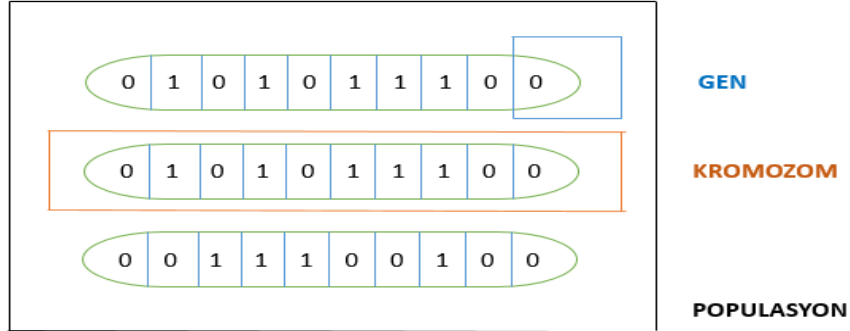


Şekil 4.6. Kromozom yapısı.

4.3.3. Popülasyon

Çözüm bilgilerine sahip genlerin oluşturduğu kromozomların birleşmesiyle oluşan yığınlardır. Popülasyondaki kromozom sayısı sabittir ve programı yapan kişi tarafından belirlenmektedir. Popülasyon sayısı çözüme ulaşma süresini de etkilemektedir. Gereğinden fazla seçilen kromozom sayısı çözüme ulaşmayı engelleyebileceęi gibi çok az seçilen kromozom sayısı da yeterli çeşitlilik sağlanamayacağından çözümsüzlük oluşturabilmektedir. Bu yüzden program yapılırken kromozom sayısı iyi belirlenmelidir.

Bu tezde, kromozom sayısı 2-50 arası seçilebilen kromozom sayıları üzerinden denenmiştir. Şekil 4.7, ikili kodlanmış genlerin birleşimi sonucu oluşan kromozomları ve kromozomların birleşerek oluşturduğu bir popülasyonu ifade etmektedir.



Şekil 4.7. Popülasyon yapısı.

4.4. SEÇİM MEKANİZMALARI

Yeni popülasyon seçilirken havuzdaki hangi bireylerin eşleneceği seçim fonksiyonuyla belirlenmektedir. Bu seçimlerde uygunluk değeri kullanılmaktadır. 5 çeşit seçim mekanizması öne çıkmaktadır.

4.4.1. Rulet Tekerleği Seçim Yöntemi

Bu seçimde, rulet tekerleği de denen bir çark üzerinden hareket edilir. Bu çark n adet parçaya bölünmüştür. Buradaki n değeri kromozom yani birey sayısını ifade etmektedir. Her bir bireyin uygunluk değeri bulunur. Bireyler uygunluk değerlerine göre [0-1] aralığında değerleri bulunur ve orantılı olarak rulet tekerleğine yerleştirilir. Örneğin; 4 tane birey olsun.

Uygunluk Değerleri;

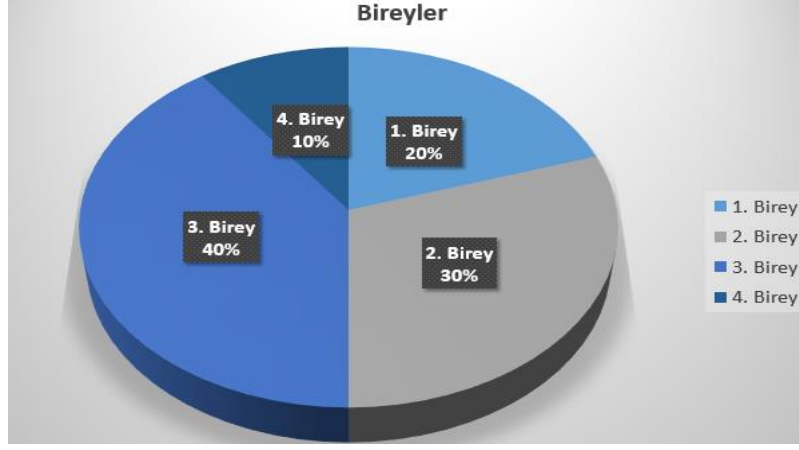
1.Birey = 0.2

2.Birey = 0.3

3.Birey = 0.4

4.Birey = 0.1

olarak hesaplandığı varsayılarak; rulet tekerleğine; Şekil 4.8'deki gibi dağılırlar.

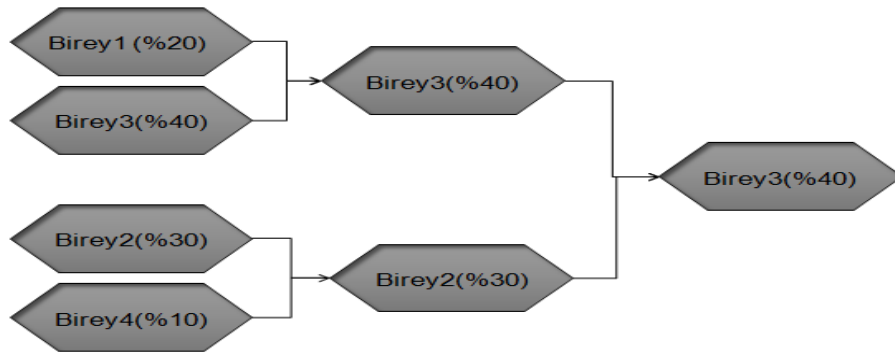


Şekil 4.8. Bireylerin uygunluk durumuna göre rulet tekerleği üzerine dağılımı.

Seçim yapılabilmesi için çarkın çevrilmesi gerekmektedir. Her bir seçim için çark bir defa daha çevrilir. Bu çarka göre, 3.birey daha sağlıklı bireydir ve çark döndürüldüğünde seçilme olasılığı %40'tır. Sağlıklı bireylerin seçilme şansını artırarak ileriki nesillerin iyileştirilmesi sağlanmaktadır. Rulet tekerleği yöntemi en çok kullanılan yöntem olarak dikkat çekmektedir. Ancak, bu metot her zaman en kullanışlı metot olmayabilir. Bu yöntemde, sonraki nesillerde beklenen kopya sayısı ve oluşan arasında belirgin bir fark olabilmektedir.

4.4.2. Turnuva Seçim Yöntemi

Bu yöntem, dünya genelinde uygulanan turnuva yöntemiyle aynı mekanizmaya sahiptir. Kromozom havuzundan iki kromozom alınır ve uygunluk değeri fazla olan alınır ve diğeri havuza geri bırakılır. Sonra tekrar iki kromozom seçilir. Bu işlem popülasyon yeterli düzeye ulaşınca kadar devam eder. Turnuva seçim yöntemi temsili Şekil 4.9'da verilmiştir.



Şekil 4.9. Turnuva seçim yöntemi.

4.4.3. Rank Seçim Yöntemi

Rank seçiminde önce havuz içerisinde bulunan popülasyon sıralanır ve daha sonra her kromozom uygunluk değerini bu sıralamadan elde eder. En kötü kromozom 1 değerini alırken, en kötü 2. kromozom 2 değerini alarak devam eder. En iyi kromozom ise en iyi değeri yani N (popülasyon büyüklüğü) değerini alır.

4.4.4. Sabit Durum Seçimi

Bu seçim yöntemi diğerleri gibi özel bir en iyiyi seçme yöntemi değildir. Bu yöntemdeki temel durum, popülasyondaki kromozomların çoğunun bir sonraki nesle aktarılmasıdır. Bu yöntemin çalışma şekline bakılırsa; her yeni kuşakta en iyi uygunluk değerine sahip kromozomlar yeni nesli oluşturmak için seçilirken, belirli bir uygunluk değerinin altında kalan kromozomlar ise elenmektedir. Elenen bu kromozomların yerine yeni oluşan yavrular havuza dâhil olmaktadır. Bu şekilde çeşitlilik de sağlanmaktadır.

4.4.5. Elitist Seçim Yöntemi

Bu seçim yöntemi, tezde kullanılan yöntemlerden birisidir. Elitist seçim yöntemi, nesillerde iyileşmenin devam etmesi veya iyileşme olmaması durumunda neslin geriye gitmemesi için kullanılan bir yöntemdir. Elitist seçim yönteminde uygunluk durumu hesaplanan kromozomlardan en iyi uygunluğa sahip olan kromozom doğrudan sonraki nesle aktarılır. Problemin karmaşıklığına göre ve popülasyon büyüklüğüne göre sonraki nesle aktarılacak kromozom sayısı da artabilmektedir.

4.5. GENETİK OPERATÖRLER

Genetik algortmada çözüm havuzu incelenirken önemli olan çeşitliliğin sağlanmasıdır. GA bir seviyeden sonra iyileşme sağlamıyorsa, probleme uygun çaprazlama ve mutasyon oranlarının belirlenmesi gerekir. Bu bölümde, genetik operatörler incelenerek uygun koşulların nasıl oluşturulduğu belirtilmektedir.

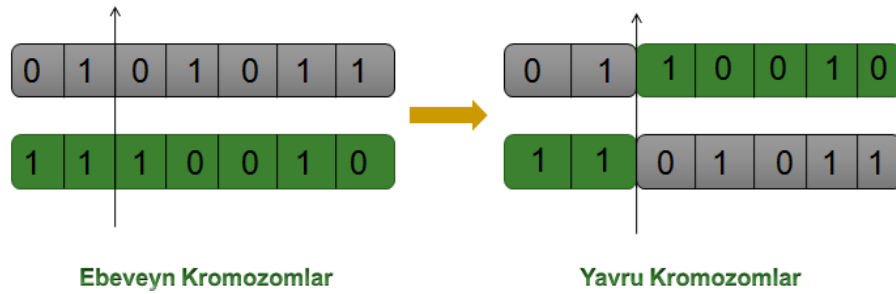
4.5.1. Çaprazlama Operatörü

Çaprazlama operatörü, mevcut havuzdaki popülasyondan alınan iki bireyden gen değişimi (bilgi değişimi) yoluyla yeni nesil için iki yeni yavru birey üretmede kullanılmaktadır. Çaprazlamaya gelen bireylerde uygunluk değeri yüksek olan bireylerin seçilme olasılığı yüksek olduğundan, çaprazlamadaki asıl amaç sağlıklı bireylerin

çaprazlanarak daha sağlıklı bireylerin ihtimalini artırmaktır. Kötü gene sahip bireyler genellikle eleneceğinden dolayı iyileşme devam etmektedir. Çaprazlama sıklığı programcı tarafından belirlenmektedir. Bu oranın iyi belirlenmesi gerekmektedir. Yüksek belirlenen oranda çok hızlı ve fazla değişim olacağından sağlıklı bireylerde hızlı bozulmalar olabilmektedir. Düşük seçilen oranda ise değişim çok az olacağından, yeni ve farklı bireyler oluşma ihtimali azalmakta ve çözüme ulaşma güçleşmektedir. En ideal çaprazlama oranı literatürde %50 - %90 olarak kabul edilmektedir. Bu tezde de %50 ve %90 arası değişen oranlarla denenmiştir. Daha önce uygulanmış olan ve kullanılan üç çeşit çaprazlama yöntemi bulunmaktadır.

4.5.1.1. Tek Noktalı Çaprazlama

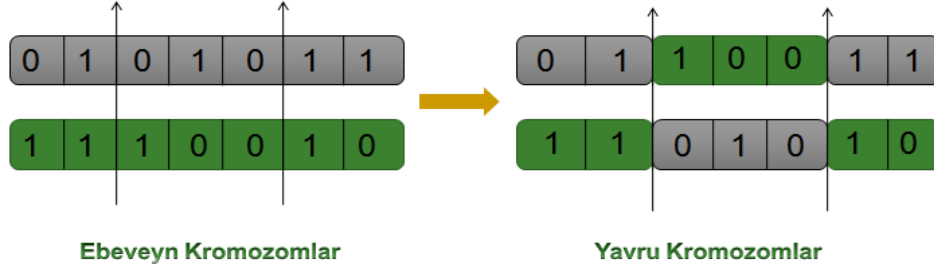
Bu çaprazlama yönteminde, popülasyon havuzundan iki kromozom çaprazlanmak üzere seçilir. Daha sonra 1 ile N arasında tek bir nokta seçilir. Bu noktanın sağ veya sol tarafı belirlenen çaprazlama oranına uygunsa 2 kromozom arasında gen takası gerçekleşir. Tek noktalı çaprazlamanın temsili gösterimi Şekil 4.10'da gösterilmektedir.



Şekil 4.10. Tek noktalı çaprazlama.

4.5.1.2. İki Noktalı Çaprazlama

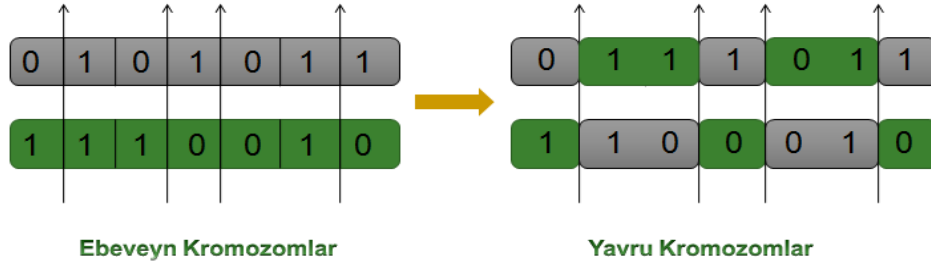
Bu çaprazlama yönteminde de, popülasyon havuzundan iki kromozom çaprazlanmak üzere seçilir. Daha sonra 1 ile N arasında iki nokta seçilir. Bu noktanın kodlamaya göre değişebilmekle birlikte belirlenen parçaları arasında, belirlenen çaprazlama oranına uygunsa gen takası gerçekleşir. İki noktalı çaprazlamanın temsili gösterimi Şekil 4.11'de gösterilmektedir.



Şekil 4.11. İki noktalı çaprazlama.

4.5.1.3. Çok Noktalı Çaprazlama

Bu çaprazlama yönteminde de, popülasyon havuzundan iki kromozom çaprazlanmak üzere seçilir. Daha sonra 1 ile N arasında istenilen sayıda nokta seçilir. Bu noktalar kodlamaya göre değişebilmekle birlikte belirlenen parçaları arasında, belirlenen çaprazlama oranına uygunsa gen takası gerçekleşir. Çok noktalı çaprazlamanın temsili gösterimi Şekil 4.12'de gösterilmektedir.



Şekil 4.12. Çok noktalı çaprazlama.

Bu tezde, literatürde geçen ve sıkça kullanılan yukarıda anlatılan çaprazlama yöntemlerinden problemin çözümüne uygun olan çok noktalı çaprazlama kullanılmıştır.

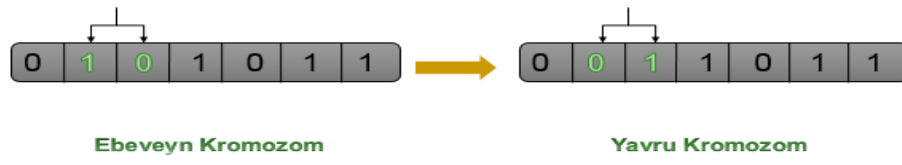
4.5.2. Mutasyon Operatörü

Genetik algoritma uygulandığı sırada, kromozomlarda eniyileme yapıldığı için nesil ilerledikçe gitgide birbirine benzeyen diziler oluşmaya başlamaktadır. Bu durum çeşitliliği de azaltmaktadır. İyileşmenin sürdürülebilir olması yeni bir operatör yardımıyla giderilebilmektedir. Bu noktada mutasyon operatörü kullanılmaktadır. çaprazlama operatöründe, iki farklı kromozom arasında gen bilgisi aktarımı yapılırken, mutasyon operatöründe, aynı kromozom üzerinde iki veya daha fazla gen bilgisi aktarımı gerçekleşir. Bu durum çeşitliliği sağlayacağı gibi daha sağlıklı bireyler ortaya çıkma ihtimalini de tekrar oluşturmaktadır. Mutasyon operatörü kromozomu değiştirdiği için ve çözümsüzlük durumunda devreye gireceği için oranı düşük tutulmalıdır. Literatürde, en

ideal mutasyon oranının %0.5 ile %0.15 arası olduğu kabul edilmektedir. Bu seviyelerin altında veya üzerinde verilen değerlerde çözüm havuzunda uzaklaşma söz konusu olduğundan tavsiye edilmemektedir. Mutasyon yapılırken de çeşitli yöntemler uygulanmaktadır. Mutasyon çeşitlerine, komşu mutasyon, keyfi mutasyon, kaydırmalı mutasyon örnek verilebilir.

4.5.2.1. Komşu Mutasyon

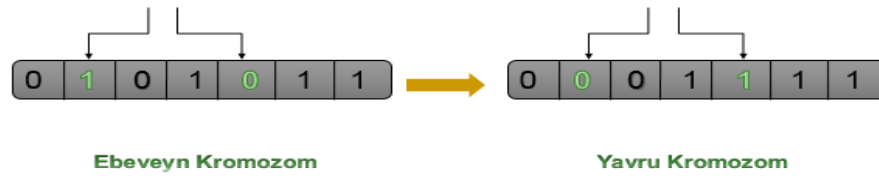
Komşu mutasyon yönteminde bir kromozom üzerindeki herhangi iki komşu gen bilgisi birbirleriyle takas edilerek çeşitlilik sağlamaya çalışılmaktadır. Şekil 4.13, bu durumu temsili olarak ifade etmektedir.



Şekil 4.13. Komşu mutasyon.

4.5.2.2. Keyfi Mutasyon

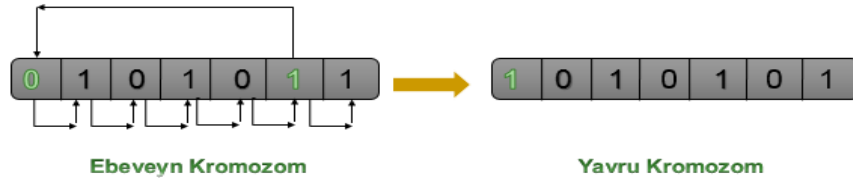
Keyfi mutasyon yönteminde, aynı kromozom üzerinde rastgele seçilen iki gen bilgisi takas edilmektedir. Diğer genlerin yeri değişmez. Bu sayede çeşitlilik sağlanmış olmaktadır. Bu durum temsili olarak Şekil 4.14'te gösterilmektedir.



Şekil 4.14. Keyfi mutasyon.

4.5.2.3. Ötelemeli Mutasyon

Ötelemeli mutasyon yönteminde, rastgele seçilen bir gen, aynı kromozom üzerinde herhangi farklı bir noktaya eklenir ve o noktada bulunan gen ve devamında bulunan genler sağa veya sola birer ötelenir. Şekil 4.15, bu yöntemi temsili olarak ifade etmektedir.



Şekil 4.15. Ötelemeli mutasyon.

4.5.3. Tamir Operatörü

Bu tez çalışmasında konu olan problem gibi kısıtlı optimizasyon problemlerinde, popülasyon havuzundaki bireyler (kromozomlar) genetik algoritma operatörleri olan çaprazlama ve mutasyona uğradıktan sonra oluşan yeni bireylerde gen bilgileri kaybolmakta veya olması gerekenden farklı dizilime doğru gitmektedirler. Bu nedenle genetik algoritma çözüm uzayından uzaklaşmakta ve uygulamayı da çözümden oldukça uzaklaştırmaktadır. Bu sorun, her bir nesil tamamlandığında bireylerin bir de Tamir (düzeltme) operatöründen geçirilmesini zorunlu kılmaktadır. Bu nedenle bu tezde ek olarak tamir operatörü de kullanılmaktadır.

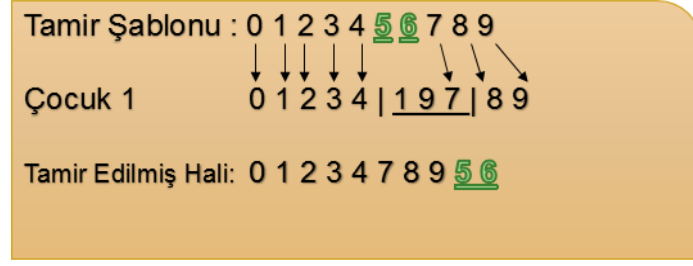
Tamir operatörü, kaybolan ya da olağan dizilimini kaybetmiş kromozomların eksik olan gen bilgilerini yerine koymakta ya da oluşumu bozulmuş genleri düzeltmektedir. Bu nedenle tamir operatörüne, düzeltme operatörü de denmektedir. Örneğin; kodlanacak parametreler [0-9] arası tamsayılar olsun ve bu tamsayılardan oluşan iki adet kromozom oluşturulduğunu varsayalım. Şekil 4.16, bu durumda çaprazlama sonucu kaybolan genleri temsil etmektedir.

Birey 1	0 1 2 3 4 <u>5 6 7</u> 8 9
Birey 2	7 3 4 2 0 <u>1 9 7</u> 3 4
Çocuk 1	0 1 2 3 4 <u>1 9 7</u> 8 9
Çocuk 2	7 3 4 2 0 <u>5 6 7</u> 3 4

Şekil 4.16. Tamir operatörünün uygulanmama durumu.

Çaprazlama operatörü uygulandıktan sonra 1. çocuğa genler aktarılırken 5 ve 6 genleri kaybolmaktadır. Bu genlerin yerine koyulması, fazlalık genlerin kaldırılması ve sağlıklı iyileşmenin devam edebilmesi için tamir operatörünün uygulanması gerekmektedir.

Şekil 4.17, tamir operatörünün uygulanmış halini temsil etmektedir.



Şekil 4.17. Tamir operatörünün uygulanma durumu.

4.6. GA'NIN KISITLI OPTİMİZASYON PROBLEMLERİNDE KULLANIMI

Optimizasyon, bir sistemde var olan işgücü, hammaddeler, kapasite, ekipman, para gibi tüm kaynakların en iyi en verimli şekilde kullanılarak maliyetin en az, kârın en çok, kapasite kullanımının en yüksek hale getirilmesi gibi amaçlara ulaşmayı sağlayan bir teknolojidir denebilir. Eniyileme yani en uygun şekilde sokma problemlerinde kısıtlar problemi zorlaştırmaktadır. Bu durum yeni bir alan olan kısıtlı optimizasyon problemlerini, bu problemler de yeni kısıtlı optimizasyon tekniklerinin ortaya çıkmasını sağlamıştır.

Kısıtlı optimizasyon problemlerine örnek olarak; ameliyat odası çizelgeleme, hemşire çizelgeleme, ders programı çizelgeleme gibi çeşitli örnekler verilebilir. Kısıtlı optimizasyon problemlerine genetik algoritma uygulanırken, çaprazlama, mutasyon gibi klasik GA operatörleri uygulandıktan sonra kısıtları sağlamayan dizilerin ortaya çıkması problemi zorlaştırmaktadır.

Bu tür problemlerde, sorunların aşılması için çeşitli teknikler ortaya konulmaktadır. Bunlardan birisi ceza fonksiyonu vasıtasıyla çözümün ilerletilmesidir. Bu yöntemde, belirlenen kısıtlara uymayan genler veya kromozomlar, uymadığı kısıt başına cezalandırılır (cezalar kısıt türüne göre değişiklik gösterebilmektedir). Daha sonra bu cezalar değerlendirme parametresi olarak kendini gösterir.

Uygunluk fonksiyonu, popülasyondaki bir elemanın aranan sonuca yakınlığının ölçüsünü verecek fonksiyon olarak tanımlanabilir. Örneğin; N elemanlı bir dizi olsun ve $f(x)$ fonksiyonunun maksimum değeri aranıyor olsun.

Bu fonksiyonda i . elemanın uygunluğu;

$$\bar{f} = \sum_{i=1}^N f_i \quad (4.1)$$

olmak üzere,

$$\frac{f_i}{\bar{f}} \quad (4.2)$$

olarak hesaplanmaktadır.

Kısıtlı optimizasyon problemlerinde ise tüm kısıtlara göre ceza puanları hesaplanır ve daha sonra uygunluk fonksiyonu hesaplanır.

Ceza puanları toplamı(tüm kısıtlar),

$$\sum_{k=0}^t \sum_{i=0}^n \sum_{j=0}^l P_k * C_{ij} \quad (4.3)$$

olmak üzere,

$$f = \frac{1}{1 + \left(\sum_{k=0}^t \sum_{i=0}^n \sum_{j=0}^l P_k * C_{ij} \right)} \quad (4.4)$$

olarak elde edilir.

*n = kromozom sayısı l = gen sayısı, t = kısıt adedi, C_{ij}= i. kromozomdaki ceza puanına sahip j. gen, P_k= k. kısıta ait ceza puanı.

Denklem (4.1) ve Denklem (4.2), bir kromozom içerisinde yer alan genlerin ceza puanlarını hesaplamak için kullanılırken, Denklem (4.3) ve Denklem (4.4), popülasyonda yer alan tüm kromozomların ceza puanlarını hesaplama için kullanılmaktadır.

Yöntemlerden bir diğeri ise kaybolan genlerin ilk nesil şablon olarak alınarak düzenlenmesi ve bunun için yeni bir algoritma düzenlenmesidir. Bu yöntemde, eksilmesi, gereğinden fazla hale gelmesi gibi nedenlerle değişen gen havuzu özel bir algoritma geliştirilerek düzeltilir.

4.7. BASİT BİR GA UYGULAMA ÖRNEĞİ

GA'nın işleyişini anlatmak için örnek olarak, $f(x) = x^2 - 2x - 1$ fonksiyonu [0-31] aralığında en yüksek değeri alması istenmektedir. İlk adım olarak değişken olan x değerinin kodlanması gerekmektedir. Bu örnekte x değerini [00000-11111] aralığında olmak üzere ikili kodlama olarak kodlayabiliriz. Daha sonra rastgele olarak bir başlangıç popülasyonu oluşturulması gerekir. Örnek Çizelge 4.1'de görüldüğü gibi 4 farklı çözüm

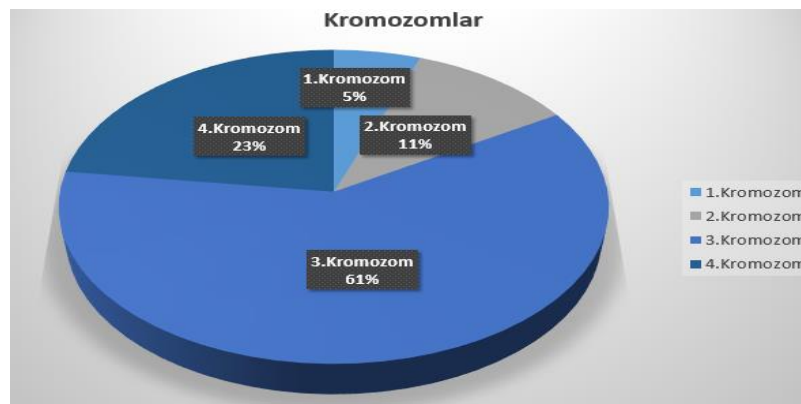
içermektedir.

Çizelge 4.1. Örnek popülasyon havuzu (nesil 0).

Kromozom_No	Kromozom_String	Sayısal Değeri	Uygunluk	Uygunluk Oranı(%)
1	01001	9	62	5,6
2	01100	12	119	10,8
3	11011	27	674	60,7
4	10001	17	254	22,9
Toplam			1109	100.0

Başlangıç popülasyonu rastgele bir sayı üretici aracılığıyla üretilmelidir. Popülasyonda oluşturulan kromozom string değerleri x değişkeninin aldığı değerlerdir. Problemden bu x değerleri kullanılarak $f(x)$ fonksiyonunun değeri hesaplanmakta ve değerlendirilmektedir. Kromozomların ayrı ayrı uygunluk değerleri hesaplanmış ve toplam uygunluk değeri 1109 olarak bulunmuştur. Seçim işleminin kolay uygulanabilmesi açısından yüzdesel oran olarak hesaplanmış ve son sütuna eklenmiştir. Her kromozomun uygunluk oranı vardır ve bu oran hangi kromozomun çözüme daha yakın olduğunu gösterebilir. Bu işlemlerden sonra eşleşme havuzuna gönderilecek kromozom çiftlerin seçilmesi için bir seçim mekanizması kullanılmalıdır.

Bu örnekte roulett wheel yani rulet tekerleği seçim mekanizması uygulanacaktır. Bu durumda kromozomlar uygunluk değerlerine göre rulet tekerleğine Şekil 4.18'deki gibi dağılmaktadırlar.



Şekil 4.18. Kromozomların rulet tekerleği üzerine dağılımı.

Rulet tekerleği mekanizmasında çiftlerin seçimi tekerleğin çevrilmesiyle gerçekleşir. Bu durumda %40 bir dilime sahip olan 3.kromozomun seçilme şansı daha yüksek olmaktadır.

Böylece ileriki nesillere uygunluk değeri yüksek olan bireylerin aktarımı ve çözüme daha hızlı yaklaşılması sağlanmış olmaktadır. Bu örnekte, 2. kromozomun 1 kopyası, 4. kromozomun 1 kopyası ve 3. kromozomun 2 kopyası seçilir. Uygunluk değeri en düşük olan 1. kromozom ise eşleşme için seçilmez. Genetik algoritma işlemlerin devam ettirilmesi için genetik operatörler kullanılmaktadır. Bunların en genel kullanılanları çaprazlama (seçilen 2 kromozomdan gen bilgisi değişimi yoluyla 2 yeni kromozom üretme işlemi) ve mutasyon (1 kromozom üzerindeki 2 ayrı genin değişimi işlemi) olarak bilinmektedir.

Çaprazlama tek noktalı, iki noktalı ve çok noktalı gibi farklı yöntemlerle yapılabilmektedir. Bu örnekte çok noktalı çaprazlama kullanılmıştır. Çaprazlama işleminin gerçekleşmesi çaprazlama oranının sağlanıp sağlanmadığının belirlenmesi ile olur. Örneğin çaprazlama oranı %70 olduğu düşünülürse, çok noktalı çaprazlamada her bir gen için ayrı ayrı 0 ile 100 arasında rastgele sayı üretilir. 70'den küçük gelen değerler için çaprazlama yapılır. Diğer durumlarda değişim yapılmayarak yeni yavrular elde edilir. Her bir kromozom beş gene sahip olduğundan beş rastgele sayı üretilir. Bu sayılar; 91, 83, 49, 77, 80 olduğu varsayılırsa; 3. Gende değişim olurken 1, 2, 4 ve 5, genlerde değişim olmayacaktır. Çaprazlama sonrası durum Çizelge 4.2'de verilmektedir.

Çizelge 4.2. Çaprazlama öncesi ve sonrası durum.

Kromozom_No	Çaprazlama Öncesi	Çaprazlama Sonrası
2	0 1 1 0 0	11000
3	1 1 0 1 1	11111
3	1 1 0 1 1	11011
4	1 0 0 0 1	10001

Çaprazlama işlemi yapıldıktan sonra mutasyon işlemi yapılır. Mutasyon oranı %0.01 ve %0.15 oranları arası uygulanmalıdır. Örnekte mutasyon oranı da sağlanmış olduğu varsayılarak, 2. kromozomun 3. geni mutasyona uğratılmıştır ve Çizelge 4.3, bu durumu göstermektedir.

Çizelge 4.3. Mutasyon öncesi ve sonrası durum.

Kromozom_No	Mutasyon Öncesi	Mutasyon Sonrası
2	11 0 00	11100
3	11111	11111
3	11011	11011
4	10001	10001

Çaprazlama ve mutasyon işlemlerinden sonra varsa ekstra uygulanması gereken operatörler uygulanır ve yeni bir nesil oluşur. Bu işlemler istenilen sonucu verinceye kadar devam eder. Bu örnekte nesil 1’de optimum çözüme ulaşması gösterilmiştir. Nesil 1, Çizelge 4.4’te gösterilmiştir ve 3. kromozom en iyi çözüm olup algoritma sona erer.

Çizelge 4.4. Çözümün bulunması durumu (nesil 1).

Kromozom_No	Kromozom_String	Sayısal Değeri	Uygunluk	Uygunluk Oranı(%)
1	11100	28	727	28,40
2	11111	31	898	35,25
3	11011	27	674	26,40
4	10001	17	254	9,95
Toplam				100.0

Sonlanma ölçütü, istenilen uygunluk değerine ulaşılan bir kromozomun ortaya çıkması durumunda, belirlenen iterasyon sayısına ulaşıldığında, ceza fonksiyonu yürütülen uygulamalarda belirlenen en düşük ceza puanına inildiğinde gibi faktörler olabilmektedir. Bu örnekte ise istenilen en yüksek çözüme ulaşan bir kromozom ortaya çıktığından dolayı program sonlanmalıdır.

GA uygulanırken karşılaşılabilecek en ciddi problem sonuçların kalitesi ile ilgilidir. Sonuçların en iyi duruma gelip gelmediğini anlamının bir diğer yolu da farklı çaprazlama ve mutasyon oranlarında algoritmayı uygulayıp elde edilen sonuçların karşılaştırılması şeklinde olabilmektedir. Bir diğer durum ise popülasyon boyutu ile ilgilidir. Sonuçların kalitesi, popülasyon boyutu artırıldığında da aynı sonuca ulaşabilmelidir.

Bunlarla birlikte farklı çaprazlama ve mutasyon yöntemleri uygulanarak sonuçların iyileştirebileceği gibi algoritmaya tamir operatörü gibi yeni operatörler ya da çözüme yaklaştıracak yeni metotlar da algoritmanın kalitesini artırabilmektedir.

5. GA'NIN AMELİYAT ODASI ÇİZELGELEMeye UYARLANMASI

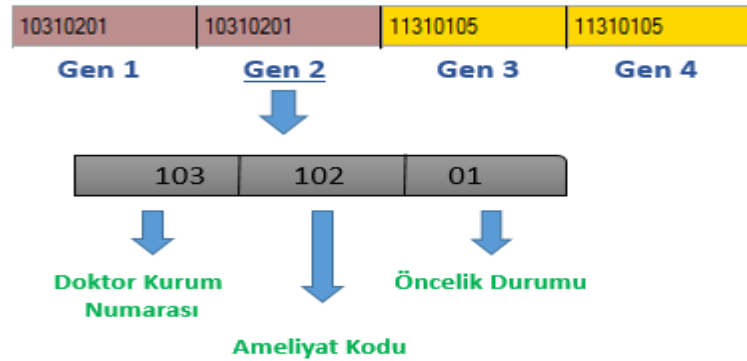
Genetik algoritma parametreler ile değil, parametrelerin kodlanmış hali ile uğraşmaktadır. Parametreler kodlanırken ise problem için çözümü en iyi ifade eden kodlama tekniği kullanılmalıdır. Günümüze kadar ulaşan birçok parametre kodlama tekniği bulunmaktadır. Bunlardan bazıları; ikili kodlama, permütasyon kodlama, değer kodlama, ağaç kodlama vb.

5.1. UYGULAMA PARAMETRE KODLAMA

5.1.1. Değer Kodlama

Reel sayılar gibi sayıların karmaşık olduğu problemler için ikili kodlama gibi yöntemler uyumsuz olacağından dolayı, bu tip problemler için değer kodlama yöntemi kullanılmaktadır. Değer kodlama da kendi içerisinde; tamsayı kodlama, kesir kodlama vb. alt dallara ayrılmaktadır. Bu tezde, ameliyat odası çizelgeleme problemine en uygun parametre kodlama yöntemi olarak değer kodlamanın alt dallarından olan tamsayı kodlama kullanılmıştır.

Tamsayı kodlamada, genler veya kromozomlar tamsayılarla ifade edilmektedir. Tezde kullanılan tamsayı kodlama yöntemi Şekil 5.1 'de gösterilmektedir.



Şekil 5.1. Kromozom yapısı.

Popülasyon kromozomlarından, kromozomlar da genlerden oluşmaktadır. Şekil 5.1, tezde oluşturulan rastgele bir kromozomu göstermektedir. Kromozomdaki her bir hücreye gen

denilmektedir. Şekil 5.1'de dört adet gen görüntülenmiştir. 1 gen 8 adet tamsayıdan oluşmaktadır. Bu 8 adet tamsayının ilk 3 hanesi; hastanede bulunan doktorların doktor kurum numaralarını temsil etmektedir. İkinci 3 hanesi ise; hastanede yapılan ameliyatlara çeşitlerinin belirtmektedir. Bu tezde 6 adet ameliyat çeşidine yer verilmiştir. Bunlar; pediatri, endoskopi ve radyoloji, genel cerrahi, göz hastalıkları, kulak burun boğaz ve ortopedidir. Ameliyathanede gerçekleştirilecek ameliyat süreleri; ameliyat çeşidine göre, o an gerçekleşebilecek komplikasyonlara göre, hastanın ameliyata vereceği tepkiye göre ve birçok faktöre göre değişebileceğinden dolayı, ameliyat çeşitlerine, ortalama ameliyat sürelerine göre belirlenen değerler atanmıştır. Ameliyat sürelerine ek olarak; ameliyathanenin bir sonraki ameliyat için geçecek süresi de hesaplanarak toplam ameliyat süresi elde edilmiştir. Hesaplanan ve tezde kullanılan bu değerler Çizelge 5.1'de de gösterilmektedir.

Çizelge 5.1. Ameliyat süreleri ve ameliyathane hazırlık süreleri.

Ameliyat Adı	Ameliyat Süresi	Ameliyathane Hazırlanma Süresi
Pediatri	t	t
Endoskopi ve Radyoloji	t	t
Genel Cerrahi	6t	2t
Göz Hastalıkları	2t	t
Kulak Burun Boğaz	2t	t
Ortopedi	4t	2t

Gen yapısında yer alan 8 tamsayının son 2 hanesi ise, kullanılan 6 ameliyatın öncelik durumunu göstermektedir. Öncelik durumu geliştirilen algoritmanın belirlenen ve uyulması gereken kısıtlarından birisidir. Öncelik sıralaması en yüksek olan ameliyatın son 2 hanesi 06, en düşük olan ameliyatın ise 01'dir.

Ameliyat çeşitlerinin öncelik durumları şu şekilde belirlenmiştir;

Genel Cerrahi → 06

Endoskopi ve Radyoloji → 05

Ortopedi → 04

Göz hastalıkları → 03

Kulak-Burun-Boğaz → 02

Pediatri → 01

Ameliyathane rezervasyonu için kullanılan öncelikli olarak genel cerrahi ameliyatı, daha sonra endoskopi ve radyoloji ve bu sırayla yerleşmeleri gerekmektedir. Bu sıraya uymayan her bir gen cezalandırılarak algoritmanın en uygun sonucu alması sağlanmaya çalışılmaktadır.

5.2. YAZILIM PROGRAMININ İNCELENMESİ

Bu tezde, ameliyathane rezervasyonu için çözümlenmesi gereken problemi çözmek için genetik algoritma kullanılmış, algoritmanın kodlanması ise C# programlama dili kullanılarak gerçekleştirilmiştir. Uygulama başlangıç ekranı Şekil 5.2'de verilmiştir.



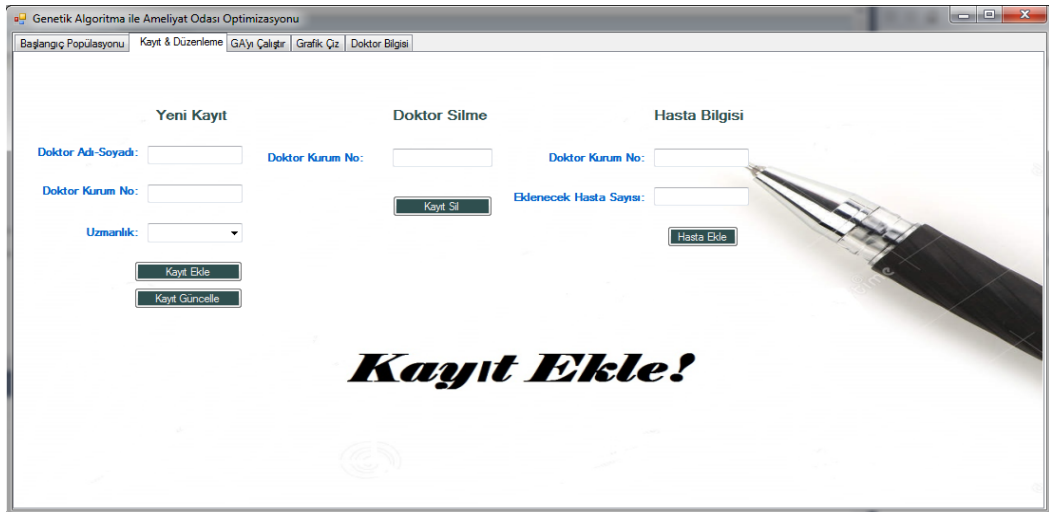
Şekil 5.2. Uygulama başlangıç ekranı.

Programı çalıştırmadan önce, ameliyathane sayısı, popülasyon havuzuna dâhil olacak başlangıç kromozom sayısı, çaprazlama operatörü için kullanılacak çaprazlama oranı ve mutasyon operatörü için uygulanacak mutasyon oranı değerleri belirlenerek algoritmayı çalıştır butonu vasıtasıyla uygulama çalıştırılmaktadır. Ameliyathane sayısı 1 ile 3 arasında, kromozom sayısı 2 ile 50 arasında, çaprazlama oranı %50 ile %90 arasında, mutasyon oranı ise %0.01 ile %0.15 arasında seçilebilmektedir. Farklı kombinasyonlar kullanılarak en iyi çözümü verecek değerlerin bulunması amaçlanmıştır. Programda, bu parametrelere ek olarak bir de rezervasyon kısmı bulunmaktadır. Şekil 5.3'te rezervasyon ekranından bir görüntü yer almaktadır.



Şekil 5.3. Rezervasyon ekranı.

Bu kısımda önceden belirlenebilecek rezervasyonlar kaydedilmektedir. Rezervasyonu yapacak doktor, hangi gün ve saatte ameliyatını gerçekleştireceğini girerek rezervasyonunu kaydetmektedir. Bu durumda iterasyon sayısı artıp popülasyonlar değişse bile rezervasyon yapılan gün ve saate başka bir ameliyat yerleşmemektedir. Rezervasyon bölümü algoritma kısıt parametrelerinden birisidir. Ayrıca bu geliştirilen bu paket programda, Şekil 5.4'te gösterilen “Kayıt & Düzenleme” sekmesinden rahat bir şekilde, hastaneye yeni gelecek doktorların sisteme eklenmesi, doktorun alanının eklenmesi ve bu alana göre doktora istenilen sayıda hasta atanabilmesi işlemlerini yapabilmekle birlikte, hastaneden ayrılan doktorların da sistemden düşürülmesi işlemleri gerçekleştirilebilmektedir.



Şekil 5.4. Uygulama kayıt ve düzenleme ekranı.

5.2.1. Program Kısıtları

Ameliyat odası çizelgeleme problemi gibi kısıt sayısı fazla olan eniyileme problemlerinde karşılaşılabilecek en büyük zorluk, kısıtları sağlamayan yapıların en aza indirgenmeye

çalışılmasıdır. Bu zorlukları aşmak için ceza fonksiyonu yöntemi, yeni bir operatör geliştirme, yeni bir algoritma geliştirme gibi çeşitli yöntemler denenmektedir. Bu tezde, bu yöntemlerden birisi olan ve genetik algoritmanın temel birimlerinden olan ceza fonksiyonu yöntemi kullanılmıştır. Kısıtlara uymayan her bir gen, önceden belirlenen sert ve yumuşak kısıtların süzgecinden geçirilmiş, her bir kısıt için farklı ve uygun derecede ceza puanları belirlenmiş ve bu kısıtlara uymayan genler kısıt ceza katsayısı ile cezalandırılmıştır. Amaç ise genetik algoritma kullanılarak ceza puanını en aza indirgeyerek en iyi çözüme ulaşmaktır.

5.2.1.1. Sert Kısıtlar

- I. Ameliyatlar yerleşirken öncelik sıralamasına uymalıdır. (Genlerin son 2 hanesi öncelik durumunu göstermektedir. (Örneğin; 06>05 olduğundan 06 olan önceliklidir ve ameliyatın önce yapılması gerekir)).
- II. Bir doktor aynı saatte birden fazla ameliyathanede bulunamaz.
- III. Bir Ameliyathanede, yalnızca bir hastanın ameliyatı yapılabilir.
- IV. Ameliyatlar yarıda kesilemez. Blok halinde kalmalıdır.
- V. Ameliyat saati, gün sonu ameliyat bitim saati olan 20.00'ı aşamaz.
- VI. Rezervasyon yapılan saate rezervasyon yapan doktor dışında herhangi bir doktor atanamaz.
- VII. Bir doktora belirlenen hasta sayısından fazla hasta atanamaz.

5.2.1.2. Yumuşak Kısıtlar

Ameliyat odası ile ilgili yumuşak parametrelerden birisi de boş vakitlerin olabildiğince azaltılmasıdır. Algoritmada ceza puanı düşük tutularak boş zamanların azaltılması yumuşak bir kriter olarak uygulanmıştır.

Örneğin Şekil 5.5'te program çalışırken kromozomdan bir parça gösterilmektedir. Gende yer alan 8 haneli sayının ilk 3 hanesinin doktor kurum kodunu, sonraki 3 hanesinin ameliyat kodunu, son 2 hanesinin ise ameliyat önceliğini gösterdiği belirtilmişti. Bu noktada, çarşamba gününde yer alan "10410201" geninin önceliği son 2 hanesi olan "01" dir. Bu ameliyat bittikten sonra başlayan "10010006" ameliyatının ise öncelik değeri "06" dir. Önceliği fazla olan "06" sonra geldiğinden dolayı bu durum öncelik kısıtını ihlal etmektedir (01<06). Bu gen öncelik kısıtına atanan ceza katsayısıyla cezalandırılacaktır.

	08:00	08:30	09:00	09:30	10:00	10:30
P.tesi		10210105	10210105	10210105	10210105	10210105
Salı		10110006	10110006	10110006	10110006	10110006
Çarş.	10310201	10410201	10410201	10010006	10010006	10010006

Şekil 5.5. Kısıt durumunun incelenmesi.

Ayrıca pazartesi günü saat 08.00 ve salı günü saat 08.00 boş zaman olarak görünmektedir. Yumuşak kısıtlardan birisi olan ve ceza katsayısı daha düşük olan boş zamanlar da olabildiğinde azaltılmaya çalışılmıştır. Ameliyat saatleri günlük 12 saat olarak kabul edilmiş ve Ameliyat Saatleri 08.00-20.00 arası olarak belirlenmiştir. Sert kısıt maddelerinden V. maddede belirtildiği gibi mesai saatlerinin dışında ameliyat atanmaması da problem kısıtlarından birisidir.

5.2.2. Program Algoritma Akışı

Program kodlanırken C# programlama dilinin kullanıldığından bahsedilmiştir. Bu bölümde algoritma geliştirilme aşamasından bahsedilmiştir. Uygulama akış şeması aşağıda Şekil 5.6'da verilmektedir.

```

1  Genetik Algoritma
2  {
3      Başlangıç popülasyonunu oluştur
4      Kriterleri belirle
5      while (!(iterasyon sonu mu) veya !(iyileşme durdu mu))
6      {
7          Kromozomları uygunluk fonksiyonuna göre değerlendir
8          Kromozomları eşleştirme havuzuna gönder
9          Elitizm ile en iyi kromozomu al
10         Kromozomlara çaprazlama operatörünü uygula
11         Kromozomlara değişim(mutasyon) operatörünü uygula
12         Popülasyondaki kromozomlara tamir(düzeltilme) operatörünü uygula
13         Her bir kromozomun uygunluk değerine göre Roulett tekerleği yöntemini uygula
14         Yeni popülasyonu bir önceki popülasyondan oluştur
15     }
16     En iyi kromozomu sonuç olarak al
17 }
18

```

Şekil 5.6. Algoritma akış şeması

Program kodlanırken, öncelikle başlangıç popülasyonu rastsal olarak tamsayı parametre kodlama yöntemiyle oluşturulmuştur. Daha sonra algoritmanın işleyişini oluşturacak en önemli kısımlarından birisi olan problem kısıtları belirlenmiştir. Çözüm istenilen noktaya gelinceye kadar döngü içerisinde bulunan; her bir kromozom için uygunluk değeri hesaplanması, kromozomların eşleşme havuzuna gönderilmesi, en iyi kromozomun elite edilerek yeni nesle doğrudan eklenmesi, rulet tekerleği yöntemiyle seçilen kromozomların çaprazlama, mutasyon ve tamir operatörlerinden geçirilmesi ve yeni popülasyonun oluşturulması işlemi sürdürülmüştür.

5.3. DENEYSEL SONUÇLAR

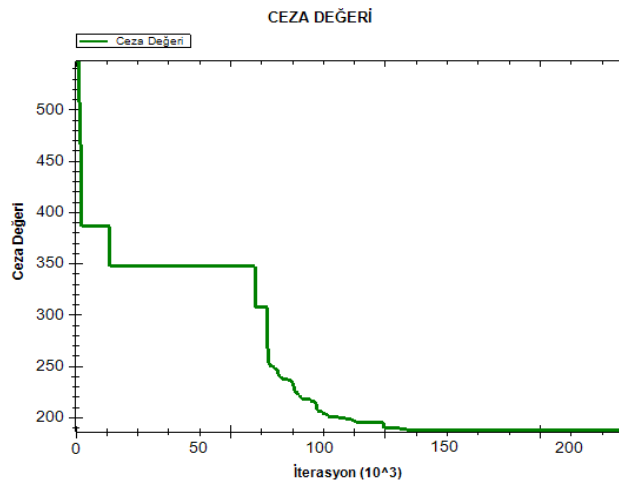
Bu bölümde algoritmanın uygulanması için geliştirilen programın farklı ameliyathane sayısının problemin çözümüne etkisi, farklı doktor sayısının çözüme etkisi, farklı çaprazlama ve mutasyon oranlarının çözüme etkisi veya rezervasyon sayısının çözüme etkisi gibi parametrelerin grafik üzerinde gösterimine ve ortaya çıkan çizelgeleme yapısının gösterimine yer verilmiştir.

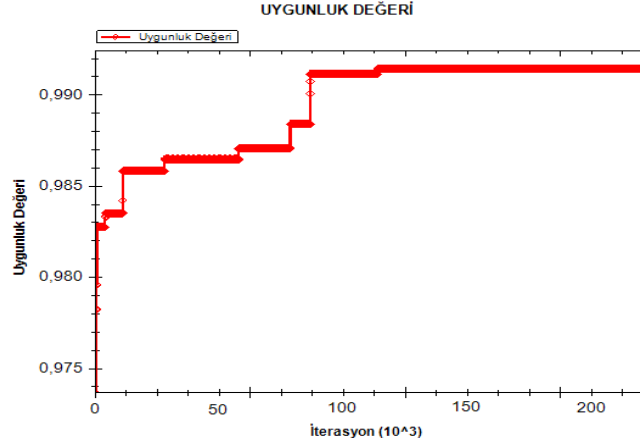
5.3.1. Doktor Sayısının Problem Çözümüne Etkisi

Farklı sayılarda belirlenen doktor sayısının, problemin çözümünde nasıl değişiklikler ortaya çıkardığı grafikler ve çizelgeler yardımıyla gösterilmiştir. Ameliyathane sayısının 3, popülasyon miktarının 40, çaprazlama oranının %85, mutasyon oranının ise %0.15 sabit değerlerine ayarlandığı bir durumda doktor sayısındaki değişimin kromozomların ceza ve uygunluk değerindeki değişimi gösterilmektedir.

Çizelge 5.2. Doktor sayısının 16 olması durumu.

Ameliyathane Sayısı	Popülasyon Miktarı	Çaprazlama Oranı (1/100)	Mutasyon Oranı (1/1000)	Doktor Sayısı
3	40	85	15	16



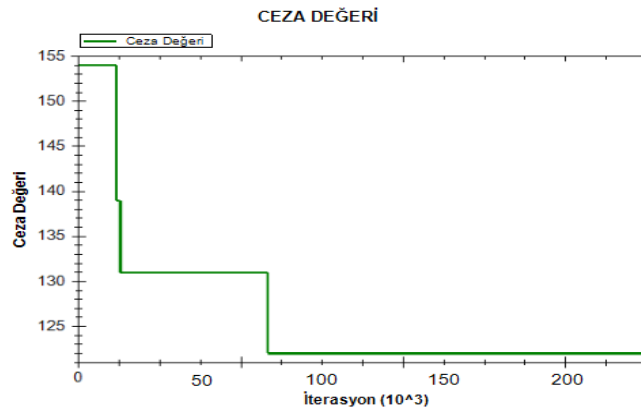


Şekil 5.8. Uygunluk değeri.

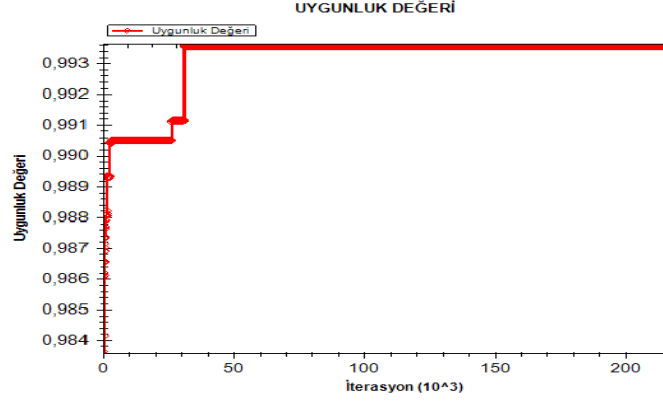
Şekil 5.8'de yer alan uygunluk değeri grafiğinde amaç değerini 1.0'a ulaşmasıdır. Bu durum algoritmanın tam olarak çözüme ulaştığı durumdur. Aynı şekilde ceza değerinin de 0'a kadar indirilebilmesi amaçlanmaktadır. Algoritma çalıştırıldığında durdurulması için kriterlerin olması gerekir. Bunlardan bazıları uygunluk değerinin 1.0'a ulaşması, ceza değerinin 0 olması veya istenilen sayıda iterasyonda durdurulabilmektedir. Bu deneylerde iterasyon sayısı durdurma kriteri olarak belirlenmiş ve 220 bin iterasyona ulaştığında durdurulmak koşuluyla sonuçlar elde edilmiştir. Şekil 5.7 ve Şekil 5.8'de gösterildiği gibi 16 doktor ile yapılan deneyde; Ceza değeri 188 değerine inmiş; uygunluk değeri ise 0.9920 değerine yükselmiştir.

Çizelge 5.3. Doktor sayısının 12 olması durumu.

Ameliyathane Sayısı	Popülasyon Miktarı	Çaprazlama Oranı (1/100)	Mutasyon Oranı (1/1000)	Doktor Sayısı
3	40	85	15	12



Şekil 5.9. Ceza değeri.

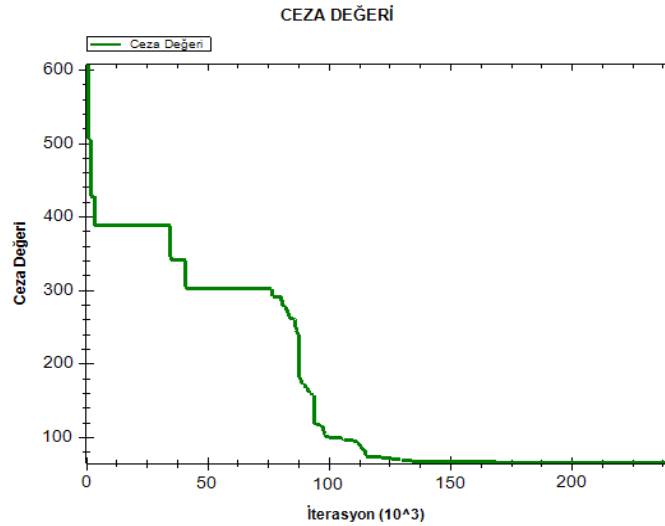


Şekil 5.10. Uygunluk değeri.

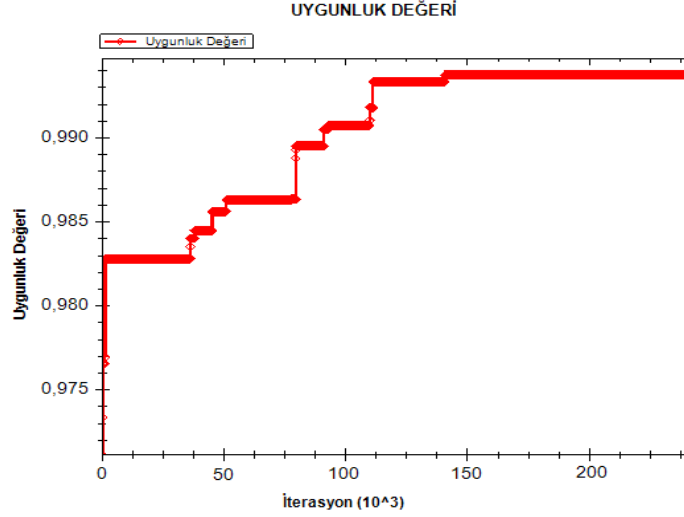
Şekil 5.9 ve Şekil 5.10'da sonuçlar göstermektedir ki doktor sayısı azaldıkça çözüme gitme durumu aynı iterasyon sayısında daha ileridedir. Ceza değeri; 122 değerine inmiş, uygunluk değeri ise 0.9936 değerine yükselmiştir. Her deneyde başarılı sonuçlar alınsa da işlem basitleştikçe çözüme daha kısa iterasyonda gidebilmektedir.

Çizelge 5.4. Doktor sayısının 8 olması durumu.

Ameliyathane Sayısı	Popülasyon Miktarı	Çaprazlama Oranı (1/100)	Mutasyon Oranı (1/1000)	Doktor Sayısı
3	40	85	15	8



Şekil 5.11. Ceza değeri.



Şekil 5.12. Uygunluk değeri.

Bu deneyler gösteriyor ki farklı doktor sayısı verildiğinde problem yine çözüme doğru ilerlemektedir. Ancak algoritmanın verdiği tepki; doktor sayısı arttıkça işlem karmaşıklaştığından ve bununla birlikte kısıtlara yakalanan gen sayısı arttığından dolayı daha ileriki nesillerde aynı çözüme ulaşabilmektedir. Şekil 5.11 ve Şekil 5.12'de gösterildiği gibi; Ceza değeri 74 değerine inmiş; Uygunluk değeri ise 0.994 değerine yükselmiştir.

Şekil 5.13 ve Şekil 5.14, bu bölümde belirtilen ameliyathane sayısının 3, popülasyon miktarının 40, çaprazlama oranının %85, mutasyon oranının ise %0.15 olduğu durumda 220 bin iterasyon sonucunda oluşan popülasyondan en iyi çözümün ameliyathane 1 için oluşan sonuçlarını göstermektedir.

Şekil 5.15 ve Şekil 5.16 ise aynı şartlar altındaki ameliyathane 2 için oluşan sonuçlarını göstermektedir.

Doktor sayısı 8 iken çalıştırılan algoritma, kısıtlardan dolayı oluşacak cezaları azaltma eğiliminde olduğundan dolayı, kısıtlardan birisi olan boşlukları da azaltma yolunu seçmektedir. Boşluklar azalırken, ameliyatlar da yerleşebildiği kadar haftanın ilk günlerine ve aynı zamanda yerleşebildiği kadar az ameliyathaneye yerleşmeye çalışmaktadırlar.

8 doktor ve bu 8 doktora belirli sayıda atanan hasta ile çalıştırılan program sonucuna göre çözüm kromozomu 2 ameliyathaneye yerleşmiştir. 3. Ameliyathane ise boş ve sonraki hastalar için kullanıma hazır halde tutulmaya devam etmektedir.

	08:00	08:30	09:00	09:30	10:00	10:30	11:00	11:30	12:00	12:30	13:00	13:30
P.tesi		10210105	10210105	10210105	10210105	10210105	10210105	10510304	10510304	10510304	10510304	10510304
Salı		10110006	10110006	10110006	10110006	10110006	10110006	10110006	10110006	10710402	10710402	10710402
Çarş.	10310201	10410201	10410201	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10110006
Perş.	10510304	10510304	10510304	10510304	10510304	10510304	10210105	10210105	10310201	10310201		
Cuma	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10010006				10010006

Şekil 5.13. Problem çözüm kromozomu (ameliyathane 1).

	14:00	14:30	15:00	15:30	16:00	16:30	17:00	17:30	18:00	18:30	19:00	19:30
P.tesi	10510304	10510304	10510304	10510304	10510304	10510304	10510304	10210105	10210105			
Salı	10410201	10410201	10310201	10310201	10210105	10210105	10710402	10710402	10710402	10410201	10410201	10310201
Çarş.	10110006	10110006	10110006	10110006	10110006	10110006	10110006				10210105	10210105
Perş.	10310201	10310201			10310201	10310201						
Cuma	10010006	10010006	10010006	10010006	10010006	10010006	10010006					

Şekil 5.14. Problem çözüm kromozomu (ameliyathane 1).

	08:00	08:30	09:00	09:30	10:00	10:30	11:00	11:30	12:00	12:30	13:00	13:30
P.tesi	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10010006				
Salı		10510304	10510304	10510304	10510304	10510304	10510304					10510304
Çarş.			10610503	10610503	10610503					10610503	10610503	10610503
Perş.												
Cuma												

Şekil 5.15. Problem çözüm kromozomu (ameliyathane 2).

	14:00	14:30	15:00	15:30	16:00	16:30	17:00	17:30	18:00	18:30	19:00	19:30
P.tesi	10110006	10110006	10110006	10110006	10110006	10110006	10110006	10110006				
Salı	10510304	10510304	10510304	10510304	10510304	10510304	10510304	10510304	10510304	10510304	10510304	
Çarş.	10610503	10610503	10610503	10610503	10610503	10610503	10610503	10610503	10610503			
Perş.												
Cuma												

Şekil 5.16. Problem çözüm kromozomu (ameliyathane 2).

5.3.2. Ameliyathane Sayısının Problem Çözümüne Etkisi

Ameliyat odası çizelgeleme yapılırken dikkate alınması gereken önemli konulardan birisi de ameliyat odası sayısının ne kadar olacağıdır. Hastanelerde ameliyat odaları çeşitlendirilebilmektedir. Örneğin aynı hastanede bir ameliyathane sadece ortopedi ameliyatları için gereken malzemeler bulunurken bir diğer ameliyat odası sadece göz ameliyatı için gereken malzemeleri içermektedir. Ancak bu durum ameliyat odalarının efektif kullanımının önüne geçebilmektedir. Hastaneye çok sayıda ortopedi ameliyatı olacak hasta geldiğinde bekleme listesi artarken, göz için az kişi gelmesi boş bir ameliyat vakitleri oluşturmaktadır. Bu yüzden bu tezde her ameliyat odasının aynı malzemelerle

ve tüm ameliyat tipleri için gereken malzemeler bulundurduğu kabul edilmiştir. Bu bölümde, ameliyat odası sayısı 1 ile 3 arasında değişen değerler verilerek aynı koşullarda yapılan deneyler gösterilmektedir.

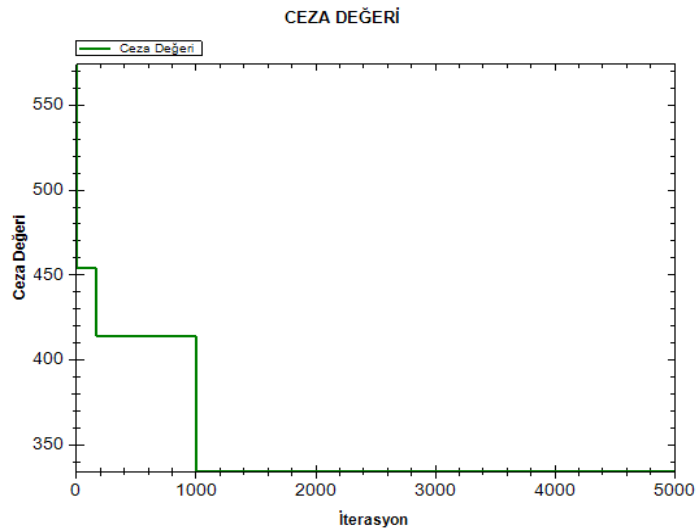
Deneyler yapılırken başlangıç popülasyonunun üretimi rastgele olduğundan dolayı ilk ceza puanı ve uygunluk değeri her program çalıştırıldığında değişebilmektedir.

Programın kısıtlarından birisi de boşluk durumu olduğundan dolayı, boşluklar azaltılmaya çalışılmaktadır. Bu da iterasyon ilerledikçe genlerin zamanla hafta başına doğru veya kromozom dizisinin ilk değerlerine doğru bir ilerleme oluşturmaktadır.

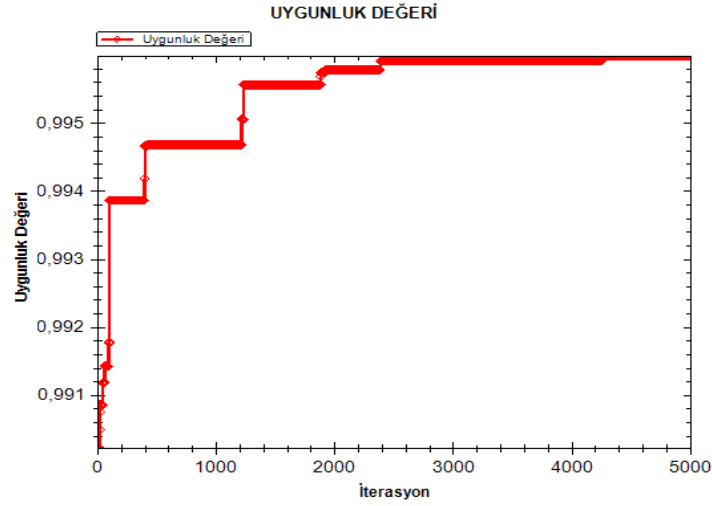
Bu gibi durumlar ile birlikte; sonuçlar değerlendirilirken; algoritmanın başlangıç popülasyonunda meydana gelen kısıtlara uygun olmayan genleri düzeltmede sağladığı başarı önem kazanmaktadır.

Çizelge 5.5. Ameliyathane sayısının 3 olma durumu

Ameliyathane Sayısı	Popülasyon Miktarı	Çaprazlama Oranı (1/100)	Mutasyon Oranı (1/1000)	Doktor Sayısı
3	40	85	10	Sabit(8)



Şekil 5.17. Ceza değeri.

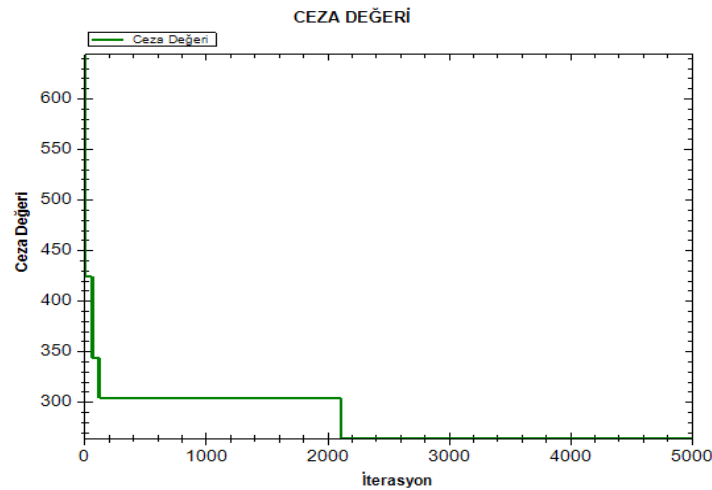


Şekil 5.18. Uygunluk değeri.

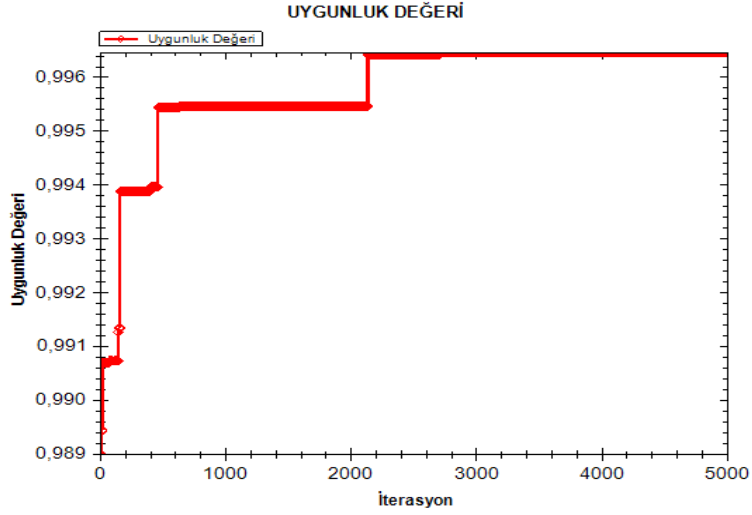
Şekil 5.17 ve Şekil 5.18'de gösterildiği gibi en uygun değerler olduğu elde edilen; popülasyon miktarının 40, çaprazlama oranının %85, mutasyon oranının %0.1 olarak belirlendiği, doktor sayısının ise sabit tutulduğu durumda ameliyathane sayısının 3 olduğu durumda, en iyi kromozomun uygunluk ve ceza değerinin değişimi verilmiştir. Deney sabit 5000. İterasyon sonucunda elde edilmiştir. Şekil 5.17 ve 5.18 'de gösterildiği gibi; 3 Ameliyathane ile yapılan deneyde; Ceza değeri 335 değerine düşmüş, Uygunluk değeri ise 0.996 değerine yükselmiştir.

Çizelge 5.6. Ameliyathane sayısının 2 olma durumu

Ameliyathane Sayısı	Popülasyon Miktarı	Çaprazlama Oranı (1/100)	Mutasyon Oranı (1/1000)	Doktor Sayısı
2	40	85	10	Sabit(8)



Şekil 5.19. Ceza değeri.

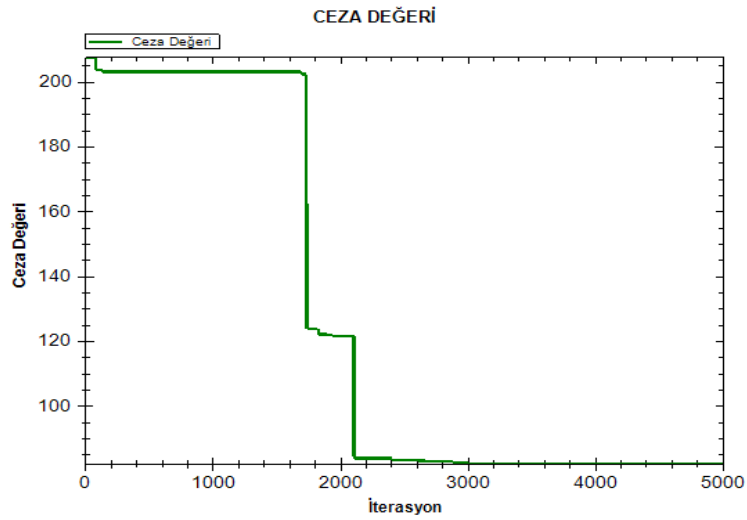


Şekil 5.20. Uygunluk değeri.

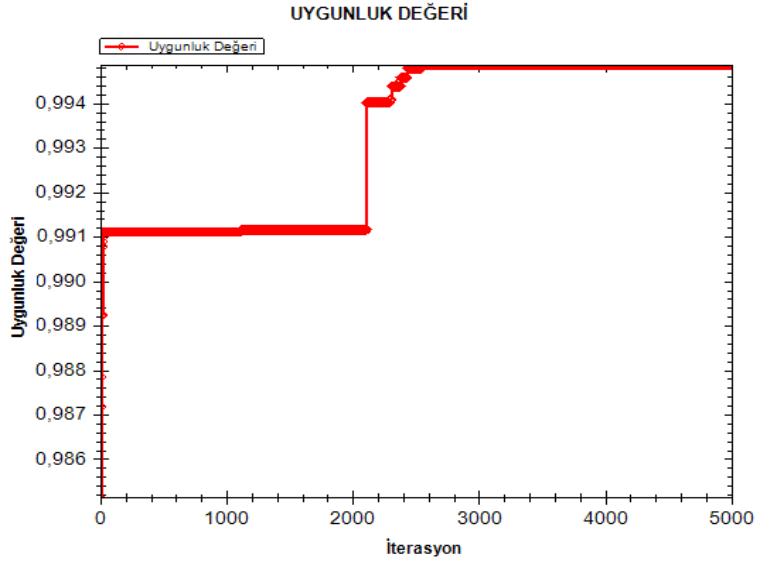
Ameliyathane sayısı arttıkça her bireye ait kromozom boyutu katlanacağından dolayı işlem karmaşıklaşmaktadır. Bu durum göz önüne alındığında elde edilen sonuçlar, işlem yükü artmasına rağmen nesil ilerledikçe problemin çözüme doğru ilerlediğini göstermektedir. Şekil 5.19, 5000 iterasyon sonunda, en iyi kromozomun ceza değerinin 262 değerine düştüğünü, Şekil 5.20 ise uygunluk değerinin; 0.9965 yükseldiğini göstermektedir.

Çizelge 5.7. Ameliyathane sayısının 1 olma durumu.

Ameliyathane Sayısı	Popülasyon Miktarı	Çaprazlama Oranı (1/100)	Mutasyon Oranı (1/1000)	Doktor Sayısı
1	40	85	10	Sabit(8)



Şekil 5.21. Ceza değeri.



Şekil 5.22. Ceza değeri.

Şekil 5.21, 1 ameliyathane için aynı şartlar altındaki ceza durumunu göstermektedir. 1 ameliyathane ile yapılan deneyde; ceza değerinin 82 değerine indiğini ve uygunluk değerinin ise 0.997 değerine yükseldiğini göstermektedir. Ceza puanının bu derece azaltılması Şekil 5.22'de gösterildiği gibi uygunluk değerine de yansımış ve 1.0 değerine çok yaklaşmıştır. Şekil 5.23 ve Şekil 5.24, popülasyon miktarının 40, çaprazlama oranının %85, mutasyon oranının %0.1 ve doktor sayısının sabit tutulduğu deneyde; ameliyathane sayısının 1 olduğu durumda elde edilen en iyi çözümü göstermektedir. Genler veya ameliyatlar 1 ameliyathaneye etkili bir biçimde yerleşmiştir.

	08:00	08:30	09:00	09:30	10:00	10:30	11:00	11:30	12:00	12:30	13:00	13:30
P.tesi	10610503	10610503	10610503	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10510304
Salı		10310201	10310201	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10010006
Çarş.	10310201	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10510304	10510304	10510304
Perş.		10310201	10310201	10110006	10110006	10110006	10110006	10110006	10110006	10110006	10110006	10110006
Cuma		10210105	10210105	10510304	10510304	10510304	10510304	10510304	10510304	10510304	10510304	10510304

Şekil 5.23. Problem çözüm kromozomu (1 ameliyathane).

	14:00	14:30	15:00	15:30	16:00	16:30	17:00	17:30	18:00	18:30	19:00	19:30
P.tesi	10510304	10510304	10510304	10510304	10510304	10710402	10710402	10710402	10710402	10710402	10710402	
Salı	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10410201	10410201	10410201	10410201	10310201
Çarş.	10510304	10510304	10510304	10510304	10510304	10510304	10510304	10510304	10510304	10410201	10410201	
Perş.	10110006	10110006	10110006	10110006	10110006	10110006	10110006	10210105	10210105	10210105	10210105	
Cuma	10510304	10510304	10510304	10610503	10610503	10610503	10610503	10610503	10610503	10310201	10310201	

Şekil 5.24. Problem çözüm kromozomu (1 ameliyathane).

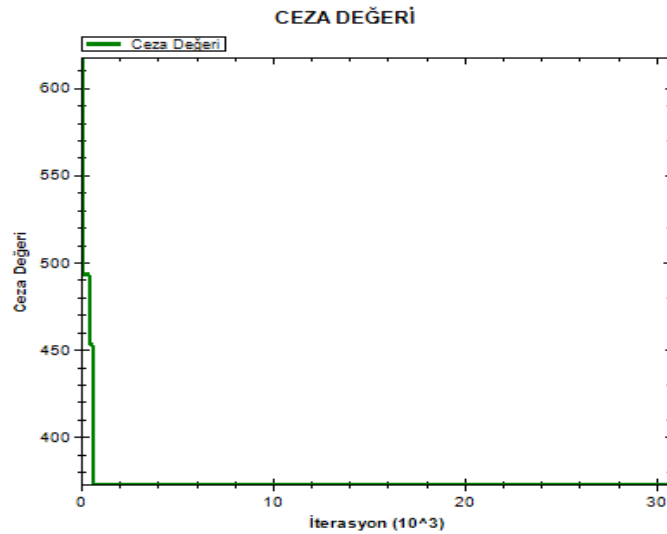
Bu 3 deney göstermektedir ki ameliyathane sayısı arttıkça kromozom boyutu büyümekte ve kromozomlar fazladan kısıtla karşılaşmaktadır. Bu durum işlem karmaşıklığını artırmaktadır ancak, ameliyathane sayısının artması aynı zamanda aynı sayıda ameliyatı daha fazla alana yerleştirme imkânı da oluşturmaktadır. Bu durum, ameliyathane sayısı arttığında ceza puanının daha yüksek başlamasına neden olmakla birlikte, ceza puanında daha fazla azaltma olanağı sağlamıştır. Şekil 5.18'de 1 ameliyat odası ile yapılan deneyde, yine 5000. İterasyon sonucunda, en iyi bireyin ceza değerinin 80-100 aralığına kadar düştüğü gözlenmektedir.

5.3.3. Rezervasyon Sayısının Problem Çözümüne Etkisi

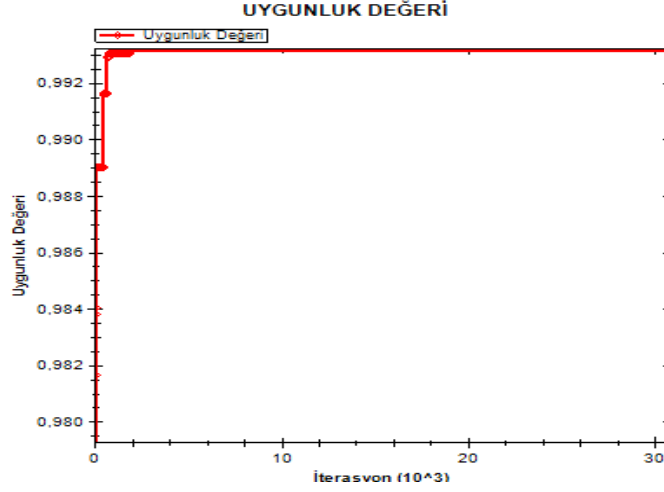
Ameliyat odası çizelgeleme problemi çözülürken, C# uygulaması gerçekleştirilen programda; kullanıcı tarafından doktorların kolaylıkla rezervasyon yapabilmesi de sağlanmıştır. Problemin kısıtlarından birisi olarak da uygulandığı üzere, rezerve edilen doktorun ameliyatının bulunduğu gün ve saate, rezervasyon iptal edilmediği sürece başka bir doktorun ameliyatı atanamamaktadır. Gerçekleştirilen bu özelliğin sayısının problem çözümüne etkisi bu bölümde gösterilmektedir.

Çizelge 5.8. Rezervasyon sayısı 3 olduğundaki durum.

Ameliyathane Sayısı	Popülasyon Miktarı	Çaprazlama Oranı (1/100)	Mutasyon Oranı (1/1000)	Rezervasyon Sayısı
3	40	85	10	3



Şekil 5.25 Ceza değeri.



Şekil 5.26. Uygunluk değeri.

İlk deneyde, 3 doktorun rezervasyon yaptırdığı durumda ve algoritma parametrelerinden ameliyathane sayısının 3, Çaprazlama oranının %85, mutasyon oranının ise %0.1 olarak belirlendiği durumdaki sonuçlar incelenmiştir. 30000 iterasyon sonucunda Şekil 5.25 ve Şekil 5.26'da gösterildiği gibi ceza değeri 370 değerine inmiş, uygunluk değeri ise 0.993 değerine kadar yükselmiştir.

Rezervasyon yapılma durumunda; diğer kromozomların yer değiştirme alanı daralacağından dolayı, ceza değerinin düşürülmesi de zorlaşacaktır. Deneye etki eden diğer faktörler ise doktor ve doktora atanan hasta sayısıdır. Bu deneyde doktor sayısı sabit ve 8 olarak kabul edilirken, doktora ise 2-8 arası değişen hasta sayıları atanmıştır.

Rezervasyon yapılan gün, saat ve doktorlar şu şekildedir;

Salı – 08.00 – 104 kodlu Doktor (Ameliyathane 1)

Salı – 14.00 – 107 kodlu Doktor (Ameliyathane 1)

Çarşamba – 08.00 – 101 kodlu Doktor (Ameliyathane 1)

Şekil 5.27, Şekil 5.28, Şekil 5.29 ve Şekil 5.30, 3 doktorun rezervasyon yaptırdığı durumda elde edilen çözüm kromozomunun 1. ve 2. ameliyathane üzerinde gösterimidir.

	08:00	08:30	09:00	09:30	10:00	10:30	11:00	11:30	12:00	12:30	13:00	13:30
P.tesi	10110006	10110006	10110006	10110006	10110006	10110006	10110006	10110006	10010006	10010006	10010006	10010006
Salı	10410201	10410201	10010006	10010006	10010006	10010006	10010006	10010006	10210105	10210105	10610503	10610503
Çarş.	10110006	10110006	10110006	10110006	10110006	10110006	10110006	10110006	10010006	10010006	10010006	10010006
Perş.	10110006	10110006	10110006	10110006	10110006	10110006	10110006	10110006	10510304	10510304	10510304	10510304
Cuma	10210105		10310201	10310201		10310201	10310201	10610503	10610503	10610503	10110006	10110006

Şekil 5.27. Çözüm kromozomu ameliyathane 1 (ilk 6 saat).

	14:00	14:30	15:00	15:30	16:00	16:30	17:00	17:30	18:00	18:30	19:00	19:30
P.tesi	10010006	10010006	10010006	10010006	10710402	10710402	10710402	10410201	10410201	10310201	10310201	
Salı	10710402	10710402	10710402	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10010006	
Çarş.	10010006	10010006	10010006	10210105	10210105	10610503	10610503	10610503		10310201	10310201	
Perş.	10510304	10510304	10710402	10710402	10710402	10310201	10310201	10210105	10210105			10210105
Cuma	10110006	10110006	10110006	10110006	10110006	10110006	10510304	10510304	10510304	10510304	10510304	10510304

Şekil 5.28. Çözüm kromozomu ameliyathane 1 (son 6 saat).

	08:00	08:30	09:00	09:30	10:00	10:30	11:00	11:30	12:00	12:30	13:00	13:30
P.tesi			10210105	10210105					10210105	10210105		
Salı			10610503	10610503	10610503		10510304	10510304	10510304	10510304	10510304	10510304
Çarş.			10510304	10510304	10510304	10510304	10510304	10510304			10510304	10510304
Perş.												
Cuma												

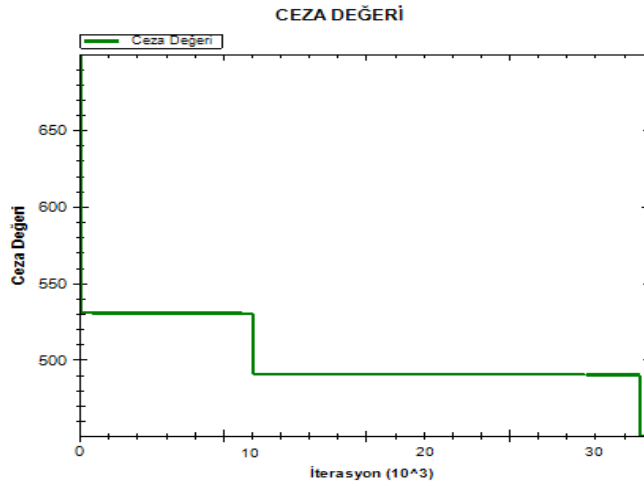
Şekil 5.29. Çözüm kromozomu ameliyathane 2 (ilk 6 saat).

	14:00	14:30	15:00	15:30	16:00	16:30	17:00	17:30	18:00	18:30	19:00	19:30
P.tesi	10510304	10510304	10510304	10510304	10510304	10510304	10610503	10610503	10610503			
Salı												
Çarş.	10510304	10510304	10510304	10510304								
Perş.												
Cuma												

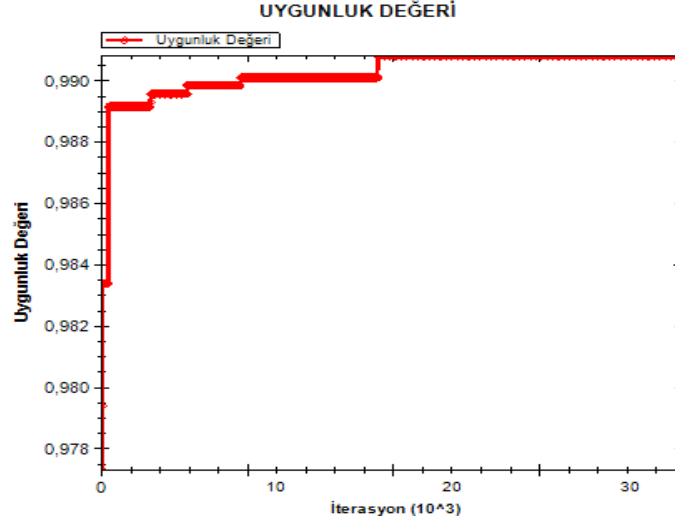
Şekil 5.30. Çözüm kromozomu ameliyathane 2 (son 6 saat).

Çizelge 5.9. Rezervasyon sayısı 8 olduğundaki durum.

Ameliyathane Sayısı	Popülasyon Miktarı	Çaprazlama Oranı (1/100)	Mutasyon Oranı (1/1000)	Rezervasyon Sayısı
3	40	85	10	8



Şekil 5.31. Ceza değeri.



Şekil 5.32. Uygunluk değeri.

İkinci deneyde ise, 8 doktorun rezervasyon yaptırdığı durumda ve algoritma parametrelerinden ameliyathane sayısının 3, çaprazlama oranının %85, mutasyon oranının ise %0.1 olarak belirlendiği durumdaki sonuçlar incelenmiştir. 30000 iterasyon sonucunda Şekil 5.31 ve Şekil 5.32'de gösterildiği gibi ceza değeri 450 değerine inmiş, uygunluk değeri ise 0.991 değerine kadar yükselmiştir. Diğer değerlerin ilk deney ile aynı tutulduğu bu deneyde ceza değerinin daha az düşmesi algoritmanın gen değişim hareketlerinin azalmasından kaynaklanmaktadır.

Rezervasyon yapılan gün, saat ve doktorlar şu şekildedir;

Salı – 08.00 – 104 kodlu Doktor (Ameliyathane 1)

Salı – 14.00 – 107 kodlu Doktor (Ameliyathane 1)

Çarşamba – 08.00 – 101 kodlu Doktor (Ameliyathane 1)

Cuma – 12.00 – 100 kodlu Doktor (Ameliyathane 1)

Pazartesi – 14.00 – 107 kodlu Doktor (Ameliyathane 2)

Salı – 09.00 – 106 kodlu Doktor (Ameliyathane 2)

Cuma – 09.30 – 105 kodlu Doktor (Ameliyathane 2)

Bu rezervasyonlar sonucu elde edilen sonuçlar ve rezervasyonların uygulanmış durumu gösterilmiştir. Şekil 5.33, Şekil 5.34, Şekil 5.35, Şekil 5.36 ve Şekil 5.37, 8 doktorun rezervasyon yaptırdığı durumda elde edilen çözüm kromozomunun ameliyathane üzerine dağılımını göstermektedir.

	08:00	08:30	09:00	09:30	10:00	10:30	11:00	11:30	12:00	12:30	13:00	13:30
P.tesi	10510304	10510304	10510304	10510304	10510304	10510304	10610503	10610503	10610503	10610503	10610503	10610503
Salı	10410201	10410201	10010006	10010006	10010006	10010006	10010006	10010006	10310201	10310201	10210105	10210105
Çarş.	10110006	10110006	10110006	10110006	10110006	10110006	10110006	10110006	10010006	10010006	10010006	10010006
Perş.	10310201	10310201	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10510304	10510304
Cuma	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10010006

Şekil 5.33. Çözüm kromozomu ameliyathane 1 (ilk 6 saat).

	14:00	14:30	15:00	15:30	16:00	16:30	17:00	17:30	18:00	18:30	19:00	19:30
P.tesi	10610503	10610503	10610503	10610503	10610503	10610503					10410201	10410201
Salı	10710402	10710402	10710402	10410201	10410201	10310201	10310201	10210105	10210105		10210105	10210105
Çarş.	10010006	10010006	10010006	10010006	10410201	10410201	10310201	10310201	10210105	10210105	10210105	10210105
Perş.	10510304	10510304	10510304	10510304	10110006	10110006	10110006	10110006	10110006	10110006	10110006	10110006
Cuma	10010006	10010006	10010006	10010006	10110006	10110006	10110006	10110006	10110006	10110006		

Şekil 5.34. Çözüm kromozomu ameliyathane 1 (son 6 saat).

	08:00	08:30	09:00	09:30	10:00	10:30	11:00	11:30	12:00	12:30	13:00	13:30
P.tesi			10110006	10110006	10110006	10110006	10110006	10110006	10110006	10110006		
Salı			10610503	10610503	10610503	10510304	10510304	10510304	10510304	10510304	10510304	
Çarş.	10510304	10510304	10510304	10510304	10510304	10510304						
Perş.												
Cuma				10510304	10510304	10510304	10510304	10510304	10510304			

Şekil 5.35. Çözüm kromozomu ameliyathane 2 (ilk 6 saat).

	14:00	14:30	15:00	15:30	16:00	16:30	17:00	17:30	18:00	18:30	19:00	19:30
P.tesi	10710402	10710402	10710402			10510304	10510304	10510304	10510304	10510304	10510304	
Salı	10510304	10510304	10510304	10510304	10510304	10510304						
Çarş.												
Perş.												
Cuma												

Şekil 5.36. Çözüm kromozomu ameliyathane 2 (son 6 saat).

	08:00	08:30	09:00	09:30	10:00	10:30	11:00	11:30	12:00	12:30	13:00	13:30
P.tesi												
Salı												
Çarş.	10610503	10610503	10610503									
Perş.												
Cuma												

Şekil 5.37. Çözüm kromozomu ameliyathane 3 (ilk 6 saat).

5.4. TAMİR OPERATÖRÜNÜN GA UYGULANMASINA ETKİSİ

Tamir operatörü, genetik algoritmanın bilinen operatörleri olan çaprazlama ve mutasyon (değişim) operatörleri uygulandıktan sonra, kromozomlarda ortaya çıkan kaybolmuş, bozulmuş ya da gereği gibi olmayıp çözüm uzayından uzaklaşmış genlerin düzeltilip tekrar çözüme doğru yol almasını sağlayan ve bu problem türüne olarak özel olarak tasarlanmış ve kodlanmış bir genetik operatördür. Bu bölümde Tamir operatörünün uygulanma ve uygulanmama durumunda nasıl bir hal aldığı gösterilmektedir. Şekil 5.38, geliştirilen paket programda çaprazlama ve mutasyon operatörlerinden sonra en iyi kromozomun bir parçasından alınmıştır. Ameliyat süreleri dikkate alındığında Pazartesi günü saat 12.00'de başlaması gereken "10010006" ameliyatı 8 bölme olması gerekmektedir. Ancak sürmesi gereken bu ameliyat, Pazartesi günü saat 12.00'da "10310201" ameliyatı ile bölünmüştür. Bunun yanı sıra Çarşamba günü saat 10.30'da başlayan ve 8 bölme devam etmesi gereken "10010006" ameliyatı da, saat 13.00'da "10110006" ameliyatı tarafından bölünmüştür. Aynı durum Perşembe günü saat 12.30'da başlayan 3 bölme sürmesi gereken "10410201" ameliyatının bölünmesi ile kendini göstermiştir. Bu durumlar bozulmaların göstergesidir ve çözümden uzaklaşmaktadır. Bu durum tamir operatörünün zorunluluğun ortaya çıkarmıştır. Şekil 5.39 ise tamsayı ile kodlanmış genlerin hangi doktorun ameliyatı olduğunu göstermektedir.

	10:30	11:00	11:30	12:00	12:30	13:00	13:30	14:00	14:30	15:00	15:30
P.tesi	10210105	10210105		10310201	10010006	10010006	10010006	10010006	10010006	10010006	10010006
Salı	10110006	10110006	10110006	10510304	10510304	10510304	10510304	10510304	10510304	10610503	10610503
Çarş.	10010006	10010006	10010006	10010006	10010006	10110006	10010006	10010006	10110006	10110006	10110006
Perş.	10510304	10610503	10610503	10610503	10410201	10410201	10310201	10310201	10210105	10210105	10010006
Cuma	10110006	10110006	10210105	10510304	10510304	10510304	10510304	10510304	10510304	10610503	10610503

Şekil 5.38. Tamir operatörünün uygulanmama durumu.

	10:30	11:00	11:30	12:00	12:30	13:00	13:30	14:00	14:30	15:00	15:30
P.tesi	Doktor 3	Doktor 3		Doktor 4	Doktor 1	Doktor 1	Doktor 1	Doktor 1	Doktor 1	Doktor 1	Doktor 1
Salı	Doktor 2	Doktor 2	Doktor 2	Doktor 6	Doktor 6	Doktor 6	Doktor 6	Doktor 6	Doktor 6	Doktor 7	Doktor 7
Çarş.	Doktor 1	Doktor 1	Doktor 1	Doktor 1	Doktor 1	Doktor 2	Doktor 1	Doktor 1	Doktor 2	Doktor 2	Doktor 2
Perş.	Doktor 6	Doktor 7	Doktor 7	Doktor 7	Doktor 5	Doktor 5	Doktor 4	Doktor 4	Doktor 3	Doktor 3	Doktor 1
Cuma	Doktor 2	Doktor 2	Doktor 3	Doktor 6	Doktor 6	Doktor 6	Doktor 6	Doktor 6	Doktor 6	Doktor 7	Doktor 7

Şekil 5.39. Tamir operatörünün uygulanmama durumunda genlerin anlamı.

Şekil 5.40, Tamir operatörü uygulanarak düzeltilmiş ve tekrar çözüm uzayına dâhil edilmiş olduğunu göstermektedir. Eksilen ameliyatlar tamir edilmiştir. Şekil 5.41 ise bu

düzeltilen kromozomların hangi doktora ait ameliyat olduğunu göstermektedir.

	10:30	11:00	11:30	12:00	12:30	13:00	13:30	14:00	14:30	15:00	15:30
P.tesi	10210105	10210105		10010006	10010006	10010006	10010006	10010006	10010006	10010006	10010006
Salı	10110006	10110006	10110006	10510304	10510304	10510304	10510304	10510304	10510304	10610503	10610503
Çarş.	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10010006	10110006	10110006	10110006
Perş.	10510304	10610503	10610503	10610503	10410201	10410201	10410201		10210105	10210105	
Cuma	10110006	10110006		10510304	10510304	10510304	10510304	10510304	10510304	10610503	10610503

Şekil 5.40. Tamir operatörünün uygulanması sonrası görünüm.

	10:30	11:00	11:30	12:00	12:30	13:00	13:30	14:00	14:30	15:00	15:30
P.tesi	Doktor 3	Doktor 3		Doktor 1	Doktor 1	Doktor 1	Doktor 1	Doktor 1	Doktor 1	Doktor 1	Doktor 1
Salı	Doktor 2	Doktor 2	Doktor 2	Doktor 6	Doktor 6	Doktor 6	Doktor 6	Doktor 6	Doktor 6	Doktor 7	Doktor 7
Çarş.	Doktor 1	Doktor 1	Doktor 1	Doktor 1	Doktor 1	Doktor 1	Doktor 1	Doktor 1	Doktor 2	Doktor 2	Doktor 2
Perş.	Doktor 6	Doktor 7	Doktor 7	Doktor 7	Doktor 5	Doktor 5	Doktor 5		Doktor 3	Doktor 3	
Cuma	Doktor 2	Doktor 2		Doktor 6	Doktor 6	Doktor 6	Doktor 6	Doktor 6	Doktor 6	Doktor 7	Doktor 7

Şekil 5.41. Tamir operatörünün uygulanması durumunda genlerin anlamı.

6. SONUÇ VE DEĞERLENDİRME

Ameliyat odası çizelgeleme problemi karmaşıklık seviyesine göre değerlendirilmelidir. Karmaşıklık seviyelerini belirleyen önemli kısıtlar bulunmaktadır. Bu özellikler şu şekilde sıralanabilmektedir;

Toplam kısıt sayısı,

Ameliyathane sayısı,

Doktor sayısı,

Başlangıç popülasyonunun büyüklüğü ve oluşturulması,

Çaprazlama ve mutasyon oranları,

Parametre kodlama türü,

Seçim mekanizması,

Tamir operatörünün işlevi

Ameliyat odası çizelgeleme birçok yöntemle yapılabilmektedir. Bunlar klasik matematiksel yöntemler, sayısal analiz yöntemleri ve sezgisel ve meta-sezgisel yöntemlerle yapılabilmektedir. Ancak yukarıda belirtilen özelliklerin çoğunu içeren çizelgeleme yöntemi NP-Hard problem türü olarak tanımlanmaktadır ve çözümünü klasik matematiksel yöntemler ve sayısal analiz yöntemleri ile imkânsızlaşmaktadır. Bu tür problemlerin çözümünü sağlayabilmek amacıyla sezgisel ve meta-sezgisel algoritmalar geliştirilmiştir. Bu algoritmalara; genetik algoritma, tabu arama algoritması, parçacık sürüsü optimizasyonu, karınca kolonisi algoritması gibi örnekler verilebilmektedir.

Bu tezde, ameliyat odası çizelgeleme probleminin çözümü ya da en yakın çözümün bulunabilmesi amacıyla genetik algoritma kullanılmıştır. Eniyileme algoritmalarından olan GA; kesin çözüme ulaşmaya çalışır, en iyi çözümü bulamasa bile yakın çözümler sunabilmektedir.

Literatürde, genetik algoritmanın uygulanması durumunda parametrelerin hangi aralıklarda olmasının en uygun olacağı iyi belirlenmelidir. Bu tezde, 2-50 arasında değerler ile denemeler yapılmış ve en uygun çözümün başlangıç popülasyonunun 40

olduđu durumda gerekleřtiđi gzlemlenmiřtir. Bir diđer nemli faktr ise aprazlama ve mutasyon oranlarıdır. Yapılan denemelerde aprazlama oranının %50 ile %90 arasında tutulmasının en uygun olduđu grmuřtr. Bu tezde, literatrde belirtilen aralıktaki denemeler yapılmıř, en uygun deđerin %85 olarak uygulanması grlmřtr. Bir diđer zellik olan mutasyon oranı genel olarak %0.01 ile %0.15 arasında uygulanmaktadır. Bu aralıktaki yapılan denemelerde en uygun sonu %0.10 olarak uygulandıđında alınmıřtır.

Parametre kodlama tr problem zmnde nem kazanmaktadır. Parametre kodlama trlerinden ikili kodlama, ađa kodlama, permtasyon kodlama, deđer kodlama, nmerik ve alfa-nmerik gibi trler incelenmiř ve en uygun zmn, deđer kodlamanın bir tr olan tamsayı kodlama olduđu belirlenmiř ve uygulanmıřtır. Deneyler ile test edilen doktor sayısının problemin zme etkisinin inceleme sonucuna gre; doktor sayısının artması problemi karmařık hale getirmesine rađmen en yakın zme ulařmada bařarı sađlanmıřtır. Yine ameliyathane sayısının artması da problemi kısıt sayısı arttıđından dolayı daha karmařık hale getirmekte fakat artan sre ile birlikte ameliyatların yerleřiminde kolaylık sađlamaktadır. Bu bilgiler ıřıđında zmde daha ileri nesillerde de olsa zme en yakın sonuca ulařmıřtır.

Literatrde ameliyat odası iin uygulanan seim yntemleri olarak rulet tekerleđi, turnuva yntemi, rank seim yntemi gibi seim yntemleri n plana ıkmaktadır. Bu tezde Rulet tekerleđi seim yntemi en uygun yntem olarak belirlenmiř ve elde edilen sonular da seimin dođruluđunu gstermiřtir.

Ameliyat odası gibi ok kısıtlı NP-hard izelgeleme problemlerinde genetik operatrler uygulanırken kaybolan, bozulan veya geređi gibi olmayan genlerin ortaya ıkması nedeniyle bu genlerin dzeltilmesi iin uygulanan tamir operatr, problemin zmn dođrudan etkilemektedir. Tamir operatrnn uygulanmaması durumunda zm kromozomunun dođal yapısının bozulduđu gsterilmiř ve dzeltme uygulanma zorunluluđu ortaya konmuřtur.

Ameliyat trlerinin sreleri; ameliyat sırasında yařanabilecek komplikasyonlar, doktorların kiřisel yetenekleri gibi birok nedenden tr deđiřiklik gsterebileceđinden, optimizasyonunun yapılmasının imknsızlařması sonucundan hareketle, bu ameliyatların ortalama sreleri sabit deđer olarak belirlenmiř ve probleme bu řekliyle dhil edilmiřtir. İhmal edilen bir diđer husus ise acil ameliyatlardır. Acil ameliyatlar durumu bařka bir hastaneye sevk etme yoluyla zlmřtr.

Literatürde çözüm için sezgisel ve meta-sezgisel olarak; karınca kolonisi, tabu arama, benzetimli tavlama, Karmaşık Tam sayılı Lineer Programlama gibi diğer yöntemler uygulanmakla birlikte, ameliyat odası problemi için en uygun algoritma yapısı genetik algoritma ile sağlanmaktadır.

Bu çalışmanın, ameliyat odası çizelgeleme problemi için gelecek çalışmalara ön hazırlık olması öngörülmektedir. Ayrıca çalışmanın, bazı iyileştirme, bazı metotları değiştirme veya yeni bir algoritma geliştirme gibi yöntemlerle daha da ilerletilmesi mümkündür.

Uygulama geliştirilirken kullanılan çaprazlama yöntemi ve mutasyon yöntemi çok noktalıdır. Blok çaprazlama ve mutasyon yöntemi kullanılması veya yeni bir yöntem ile denenmesi ile daha iyi sonuçlara ulaşılabilir.

C# programlama dili kullanılarak geliştirilen programda, bilinen genetik operatörlere ek olarak bir de; ameliyat odası çizelgeleme gibi kısıt sayısı fazla olan karmaşık problemler için özel olarak uygulanması gereken tamir operatörü kullanılmıştır. Programın, gelecekte, tamir operatörünün kodlanmasında yapılacak iyileştirmeler ile algoritmanın verimliliğinin artırılması düşünülmektedir.

Ayrıca, çalışma oluşturulurken popülasyon havuzundan rulet tekerleği yöntemi kullanılarak genetik operatörlerden geçirilmek üzere bireyler seçilmiştir. Rulet tekerleğinden farklı yöntemler duruma göre sonucun iyileştirilmesi için kullanılabilir. Bu yöntemler; daha önce kullanılmış olan turnuva, rank seçim yöntemi gibi yöntemler olabilmekle birlikte probleme uygun olarak geliştirilecek yeni bir seçim yöntemi, problemi kesin çözüme yaklaştırabilir. Bunun yanı sıra, çalışmada kullanılan elitizm yönteminde, sadece en iyi birey sonraki nesle aktarılmaktadır. Popülasyon boyutuna göre daha fazla sayıda en sağlıklı genlerin sonraki nesle aktarılması problemin iyileştirilmesinde denenebilecek yöntemlerden birisidir. Ameliyat odası çizelgeleme problemi NP-Hard tipi bir problemdir. Bu nedenle, sezgisel ve meta-sezgisel algoritmalar ile çözümü ya da en yakın çözümü bulunabilmektedir. Ancak, gelecekte bu problem tipine uygun yeni geliştirilecek bir algoritma ile de maksimum çözüme ulaşılabilmesi mümkün olacaktır. Bu tez çalışmasındaki yapı, gelecek çalışmalarda literatürde yeni yer almaya başlayan Epigenetik Algoritmanın uygulanmasına çalışılacaktır. Bu tez çalışmasındaki sonuçları ile Epigenetik Algoritma sonuçlarının kıyaslaması yapılarak hangisinin daha iyi sonuç vereceği incelenecektir.

7. KAYNAKLAR

- [1] Sağlık Bakanlığı. (2018, 27 Ekim). *A-b-c Grubu toplam ameliyatlara*. [Online]. Erişim: <http://rapor.saglik.gov.tr/istatistik/rapor/index.php>.
- [2] M. Dorigo and T. Stutzle, "Ant colony optimization for NP-Hard problems," in *Ant Colony Optimization*, 1st ed., Boston, MA, USA: Springer, 2004, ch.5, pp.167-181.
- [3] O. Engin ve A. Fırlalı, "Akış tipi çizelgeleme problemlerinin genetik algoritma yardımı ile çözümünde uygun çaprazlama operatörünün belirlenmesi," *Doğuş Üniversitesi Dergisi.*, c. 3, s. 2 ss. 27–35, 2002.
- [4] F. Guerriero and R. Guido, "Operational research in the management of the operating theatre: a survey," *Health Care Management Science.*, vol. 14, no. 1, pp. 89–114, 2011.
- [5] B. Cardoen, E. Demeulemeester, and J. Beliën, "Operating room planning and scheduling: A literature review," *European Journal Operational Research*, vol. 201, no. 3 pp. 921–932, 2010.
- [6] S. Brailsford and J. Vissers, "OR in healthcare: A European perspective," *European Journal Operational Research*, vol. 212, no. 2 pp. 223–234, 2011.
- [7] Z. Y. Abdelrasol, N. Harraz, and A. Eltawil, "A proposed solution framework for the operating room scheduling problems," in *World Congress on Engineering and Computer Science*, San Francisco, USA, 2013, pp. 23–25.
- [8] P. Patterson, "What makes a well-oiled scheduling system," *OR Manager*, vol. 12, no. 9, pp. 19–23, 1996.
- [9] I. Marques, M. Captivo, and M. Vaz Pato, "An integer programming approach to elective surgery scheduling," *Operations Research-Spectrum*, vol. 34, no. 2, pp. 407–27, 2012.
- [10] D. Conforti, F. Guerriero, and R. Guido, "A multi-objective block scheduling model for the management of surgical operating rooms: New solution approaches via genetic algorithms," in *Proceedings of IEEE Workshop on Health Care Management (WHCM)*, Venice, Italy, 2010, pp. 1–5.
- [11] I. Marques, M. Captivo, and M. Vaz Pato, "Planning elective surgeries in a portuguese hospital: Study of different mutation rules for a genetic heuristic," in *Lect Notes Management Science*, Netherlands, 2012, pp. 238–243.
- [12] M. Khambhammettu and P. Marie, "Analyzing a decision support system for resource planning and surgery scheduling," *Procedia Computer Science*, vol. 100, pp. 532–538, 2016.
- [13] J. M. Molina-Pariente, W. E. Hans, J. M. Framinan, and T. Gomez-Cia, "New heuristics for planning operating rooms," *Computer & Industrial Engineering*, vol. 90, pp. 429–443, 2015.

- [14] G. Rosita and C. Domenico, "A hybrid genetic approach for solving an integrated multi-objective operating room planning and scheduling problem," *Computer Operation Research*, vol. 87, pp. 270–282, 2017.
- [15] W. Xiang, J. Yin, and G. Lim, "An ant colony optimization approach for solving an operating room surgery scheduling problem," *Computer & Industrial Engineering*, vol. 85, pp. 335–345, 2015.
- [16] G. Latorre-Nunez, A. L er-Villagra, V. Marianov, C. Obreque, F. Ramis, and L. Neriz, "Scheduling operating rooms with consideration of all resources, post anesthesia beds and emergency surgeries," *Computer & Industrial Engineering*, vol. 97, pp. 248–257, 2016.
- [17] A. Bouguerra, C. Sauvey, and N. Sauer, "Mathematical model for maximizing operating rooms utilization," *IFAC-PapersOnLine*, vol. 48, no. 3, pp. 118–123, 2015.
- [18] S. Hadhemi, J. Badreddine, D. Abdelaziz, M. Lotfi, and B. Abir, "A stochastic optimization and simulation approach for scheduling operating rooms and recovery beds in an orthopedic surgery department," *Computer & Industrial Engineering*, vol. 80, pp. 72–79, 2015.
- [19] X. Wei, Y. Jiao, and L. Gino, "A short-term operating room surgery scheduling problem integrating multiple nurses roster constraints," *Artificial Intelligence in Medicine*, vol. 63, no. 2, pp. 91–106, 2015.
- [20] P. Landa, R. Aringhieri, P. Soriano, E. Tanfani, and A. Testi, "A hybrid optimization algorithm for surgeries scheduling," *Operational Research for Health Care*, vol. 8, pp. 103–114, 2016.
- [21] J. Razmi, M.S. Yousefi, and M. Barati, "A stochastic model for operating room unique equipment planning under uncertainty," *IFAC-PapersOnLine*, vol. 48, no. 3, pp. 1796–1801, 2015.
- [22] T. Wang, N. Meskens, and D. Duvivier, "Scheduling operating theatres: Mixed integer programming vs. constraint programming," *European Journal of Operational Research*, vol. 247, no. 2, pp. 401–413, 2015.
- [23] A. Riise, C. Mannino, and E. K. Burke, "Modelling and solving generalised operational surgery scheduling problems," *Computers & Operations Research*, vol. 66, pp. 1–11, 2016.
- [24] M. Dios, J. M. Molina-Pariente., V. Fernandez-Viagas, J. L. Andrade-Pineda, and J. M. Framinan, "A decision support system for operating room scheduling," *Computer & Industrial Engineering*, vol. 88, pp. 430–443, 2015.
- [25] A. W. Murray, S.T. Beaman, C. W. Kampik, and J. J. Quinlan, "Simulation in the operating room," *Best Practice & Research Clinical Anaesthesiology*, vol. 29, no. 1, pp. 41–50, 2015.
- [26] P. M. Castro and I. Marques, "Operating room scheduling with generalized disjunctive programming," *Computers & Operations Research*, vol. 64, pp. 262–273, 2015.
- [27] A. Amin, Y. Honghan, and L. Wei, "Operating room planning under surgery type and priority constraints," *Procedia Manufacturing*, vol. 5, pp. 15–25, 2016.
- [28] B. Beroule, O. Grunder, O. Barakat, O. Aujoulat, and H. Lustig, "Operating room

- scheduling including medical devices sterilization: towards a transverse logistic,” *IFAC-PapersOnLine*, vol. 49, no. 12, pp. 1146–1151, 2016.
- [29] L.E.M. Alameda and A. Macario, “Advances in operating room management, the role of operating room director,” *Revista Espanola de Anestesiologia y Reanimacion (English Ed)*, vol. 64, no. 3, pp. 121–124, 2017.
- [30] E. Veen-Berk, S. G. Elkhuisen, B. Kuijper, and G. Kazemier, “Dedicated operating room for emergency surgery generates more utilization, less overtime, and less cancellations,” *The American Journal of Surgery*, vol. 211, no. 1, pp. 122–128, 2016.
- [31] C. V. Riet and E. Demeulemeester, “Trade-offs in operating room planning for electives and emergencies: A review,” *Operational Research for Health Care*, vol. 7, pp. 52–69, 2015.
- [32] C.L. Siqueira, E. F. Arruda, L. Bahiense, G. L. Bahr, and G. R. Motta, “Long-term integrated surgery room optimization and recovery ward planning, with a case study in the Brazilian National Institute of Traumatology and Orthopedics (INTO),” *European Journal of Operational Research*, vol. 264, no. 3, pp. 870–883, 2018.
- [33] V. Roshanaei, C. Luong, D. M. Aleman, and D. Urbach, “Propagating logic-based Benders’ decomposition approaches for distributed operating room scheduling,” *European Journal of Operational Research*, vol. 257, no. 2, pp. 439–455, 2017.
- [34] R. Aringhieri, P. Landa, P. Soriano, E. Taffani and A. Testi, “A two level metaheuristic for the operating room scheduling and assignment problem,” *Computers & Operations Research*, vol. 54, pp. 21–34, 2015.
- [35] G. Xiao, W. Jaarsveld, M. Dong, and J. Klundert, “Stochastic programming analysis and solutions to schedule overcrowded operating rooms in China,” *Computers & Operations Research*, vol. 74, pp. 78–91, 2016.
- [36] A. Jebali and A. Diabat, “A chance-constrained operating room planning with elective and emergency cases under downstream capacity constraints,” *Computer & Industrial Engineering*, vol. 114, pp. 329–344, 2017.
- [37] H. John Henry, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. Oxford, England: U Michigan press, 1975.
- [38] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Boston, MA: Addison-Wesley Longman, 1989.
- [39] J. R. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. Cambridge, MA : MIT Press, 1992.

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Tunahan TİMUÇİN
Doğum Tarihi ve Yeri : 24.09.1991 / İmamoğlu-ADANA
Yabancı Dili : İngilizce
E-posta : info.tunahantimucin@gmail.com

ÖĞRENİM DURUMU

Derece	Alan	Okul/Üniversite	Mezuniyet Yılı
Y. Lisans	Bilgisayar Mühendisliği.	Düzce Üniversitesi	2018
Lisans	Bilgisayar Mühendisliği.	Ankara Üniversitesi	2015
Lise	Fen Bilimleri.	Yusuf Baysal Anadolu Lisesi	2009