



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

**GENETİK ALGORİTMA TABANLI KONFERANS OTURUM
ÇİZELGELEME PROGRAMININ GELİŞTİRİLMESİ**

ERCAN ATAGÜN

**YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI**

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

DÜZCE, 2020

T.C.
DÜZCE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

GENETİK ALGORİTMA TABANLI KONFERANS OTURUM
ÇİZELGELEME PROGRAMININ GELİŞTİRİLMESİ

Ercan ATAGÜN 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

Prof. Dr. Uğur GÜVENÇ

Düzce Üniversitesi

Doç. Dr. Serhat DUMAN

Bandırma Onyedli Eylül Üniversitesi

Tez Savunma Tarihi: 03/07/2020

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.

03 Temmuz 2020

Ercan Atagün



TEŐEKKÜR

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

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

3 Temmuz 2020

Ercan Atagün

İÇİNDEKİLER

Sayfa No

ŞEKİL LİSTESİ.....	vii
ÇİZELGE LİSTESİ.....	ix
KISALTMALAR.....	x
ÖZET	xi
ABSTRACT	xii
1. GİRİŞ.....	1
2. OPTİMİZASYON PROBLEMLERİ VE ÇİZELGELEME	3
2.1. OPTİMİZASYON PROBLEMLERİ.....	3
2.2. ÇİZELGELEME PROBLEMLERİ.....	5
2.3. KONFERANS ÇİZELGELEME	6
3. SEZGİSEL VE METASEZGİSEL ALGORİTMA	12
3.1. SEZGİSEL ALGORİTMA	12
3.2. METASEZGİSEL ALGORİTMALAR.....	18
3.2.1. Karınca Kolonisi (Ant Colony) Algoritması.....	23
3.2.2. Parçacık Sürüsü Optimizasyonu Algoritması	25
3.2.3. Tavlama Benzetimi (Simulated Annealing) Algoritması.....	26
3.2.4. Yapay Arı Kolonisi (Artificial Bee Colony) Algoritması.....	28
4. GENETİK ALGORİTMA VE EPİGENETİK ALGORİTMA ...	30
4.1. GENETİK ALGORİTMA TEMEL KAVRAMLAR	31
4.1.1. Parametre Kodlama.....	31
4.1.1.1. İkili (Binary) Kodlama	31
4.1.1.2. Permütasyon Kodlama.....	31
4.1.1.3. Değer Kodlama.....	32
4.1.1.4. Ağaç Kodlama.....	32
4.1.2. Gen.....	33
4.1.3. Kromozom	33
4.1.4. Popülasyon	33
4.2. GENETİK OPERATÖRLER.....	34
4.2.1. Çaprazlama.....	34
4.2.1.1. Tek Noktalı Çaprazlama	34
4.2.1.2. İki Noktalı Çaprazlama.....	35
4.2.1.3. Çok Noktalı Çaprazlama.....	35
4.2.1.4. Uniform Çaprazlama	36
4.2.1.5. Sıraya Dayalı Çaprazlama.....	37
4.2.1.6. Kısmi Zamanlı Çaprazlama	37
4.2.2. Mutasyon.....	38
4.2.2.1. Yer Değiştirme Mutasyonu.....	38
4.2.2.2. Ekleme Mutasyonu	38
4.2.2.3. Ters Çevirme Mutasyonu	39
4.2.2.4. Karşılıklı Yer Değiştirme Mutasyonu.....	39
4.2.2.5. Komşu Mutasyon.....	39
4.3. SEÇİM OPERATÖRÜ	40

4.3.1. Turnuva Seçim Mekanizması.....	40
4.3.2. Sıralı Seçim Mekanizması	40
4.3.3. Denge Durum Mekanizması.....	41
4.3.4. Rank Seçim Mekanizması	41
4.3.5. Rulet Çemberi.....	41
4.4. ELİTİZM.....	42
4.5. TAMİR OPERATÖRÜ	42
4.6. EPİGENETİK ALGORİTMA.....	43
4.7. EPİGENETİK ALGORİTMA İLE İLGİLİ ÇALIŞMALAR.....	48
4.8. EPİGENETİK OPERATÖRLER	48
4.8.1. Epiçaprazlama Operatörü.....	48
4.8.2. Epideğişim (Epimutasyon) Operatörü	49
4.8.3. Epigenetik Faktör Listesi	50
5. UYGULAMA	52
5.1. EPİGENETİK ALGORİTMANIN PROBLEME UYGULANMASI	57
6. DENEYSEL SONUÇLAR	62
6.1. REZERVASYON İŞLEMİNİN PROBLEM ÇÖZÜMÜNE ETKİSİ.....	76
6.2. TAMİR OPERATÖRÜNÜN ÇÖZÜME ETKİSİ	79
6.3. YAZILIM PROGRAMININ İNCELENMESİ	80
7. SONUÇLAR VE ÖNERİLER.....	85
8. KAYNAKLAR	88
ÖZGEÇMİŞ	96

ŞEKİL LİSTESİ

	<u>Sayfa No</u>
Şekil 4.1. İkili kodlama yapısı.	31
Şekil 4.2. Permütasyon kodlama yapısı.	32
Şekil 4.3. Değer kodlama yapısı.	32
Şekil 4.4. Gen yapısı.	33
Şekil 4.5. Kromozom yapısı.	33
Şekil 4.6. Tek noktalı çaprazlama.	35
Şekil 4.7. İki noktalı çaprazlama.	35
Şekil 4.8. Çok noktalı çaprazlama.	36
Şekil 4.9. Uniform çaprazlama.	36
Şekil 4.10. Sıraya dayalı çaprazlama.	37
Şekil 4.11. Kısmi zamanlı çaprazlama.	37
Şekil 4.12. Yer değiştirme mutasyonu.	38
Şekil 4.13. Ekleme mutasyonu.	39
Şekil 4.14. Ters çevirme mutasyonu.	39
Şekil 4.15. Karşılıklı yer değiştirme mutasyonu.	39
Şekil 4.16. Komşu mutasyonu.	40
Şekil 4.17. Tamir operatörünün gerekliliği.	43
Şekil 4.18. Epigenetik mekanizma yapısı.	47
Şekil 4.19. Kromatin yapısı.	47
Şekil 4.20. Epiçaprazlama.	49
Şekil 4.21. Epimutasyon.	49
Şekil 4.22. Epigenetik faktör listesi.	50
Şekil 4.23. Epigenetik algoritma akış şeması.	50
Şekil 4.24. Epigenetik algoritma sözde kodu.	51
Şekil 5.1. Kullanıcı giriş ekranı.	52
Şekil 5.2. Makale kayıt ekranı.	53
Şekil 5.3. Genetik algoritma sözde kodu.	54
Şekil 5.11. Epimutasyon.	60
Şekil 5.12. Epimutasyon sonrası kromozom dizilimi.	61
Şekil 6.1. 200 bildiri, 2 gün, 6 oturum, 6 salon genetik algoritma.	64
Şekil 6.2. 300 bildiri, 2 gün, 6 oturum, 6 salon genetik algoritma.	65
Şekil 6.3. 200 bildiri, 3 gün, 4 oturum, 6 salon genetik algoritma.	66
Şekil 6.4. 300 bildiri, 3 gün, 4 oturum, 6 salon genetik algoritma.	66
Şekil 6.5. 200 bildiri, 3 gün, 6 oturum, 5 salon genetik algoritma.	67
Şekil 6.6. 300 bildiri, 3 gün, 6 oturum, 5 salon genetik algoritma.	67
Şekil 6.7. 200 bildiri, 3 gün, 6 oturum, 6 salon genetik algoritma.	68
Şekil 6.8. 300 bildiri, 3 gün, 6 oturum, 6 salon genetik algoritma.	68
Şekil 6.9. 200 bildiri, 2 gün, 6 oturum, 6 salon epigenetik algoritma.	70
Şekil 6.10. 300 bildiri, 2 gün, 6 oturum, 6 salon epigenetik algoritma.	71
Şekil 6.11. 200 bildiri, 3 gün, 4 oturum, 6 salon epigenetik algoritma.	72
Şekil 6.12. 300 bildiri, 3 gün, 4 oturum, 6 salon epigenetik algoritma.	72
Şekil 6.13. 200 bildiri, 3 gün, 6 oturum, 5 salon epigenetik algoritma.	73
Şekil 6.14. 300 bildiri, 3 gün, 6 oturum, 5 salon epigenetik algoritma.	73
Şekil 6.15. 200 bildiri, 3 gün, 6 oturum, 6 salon epigenetik algoritma.	74
Şekil 6.16. 300 bildiri, 3 gün, 6 oturum, 6 salon epigenetik algoritma.	75
Şekil 6.17. Rezervasyon kayıt ekranı.	76

Şekil 6.18. 200 bildiri, 2 gün, 4 oturum, 5 salon genetik algoritma.	77
Şekil 6.19. 200 bildiri, 2 gün, 4 oturum, 5 salon genetik algoritma.	78
Şekil 6.20. 200 bildiri, 2 gün, 4 oturum, 5 salon epigenetik algoritma.	78
Şekil 6.21. 200 bildiri, 2 gün, 4 oturum, 5 salon epigenetik algoritma.	79
Şekil 6.22. Tamir operatörünün etkisi.	80
Şekil 6.23. Çizelgeleme giriş bilgilerinin yüklenmesi.	81
Şekil 6.24. Çizelgeleme giriş bilgilerinin yüklenmesi.	82
Şekil 6.25. Gen bilgilerinin açıklanması.	83
Şekil 6.26. Program çıktısı olan konferans kitapçığı.	83



ÇİZELGE LİSTESİ

	<u>Sayfa No</u>
Çizelge 4.1. Kromozomların uygunluk değerleri.	43
Çizelge 6.1. 200 bildirinin genetik algoritma ile çizelgelenmesi.	63
Çizelge 6.2. 300 bildirinin genetik algoritma ile çizelgelenmesi.	64
Çizelge 6.3. 200 bildirinin epigenetik algoritma ile çizelgelenmesi.....	69
Çizelge 6.4. 300 bildirinin epigenetik algoritma ile çizelgelenmesi.....	70
Çizelge 6.5. 200 bildirinin genetik algoritma ile rezervasyonun incelenmesi.....	77
Çizelge 6.6. 200 bildirinin epigenetik algoritma ile rezervasyonun incelenmesi.....	78



KISALTMALAR

DNA

PSO

RNA

Deoksiribo Nükleik Asit

Parçacık Sürüsü Optimizasyonu

Ribo Nükleik Asit



ÖZET

GENETİK ALGORİTMA TABANLI KONFERANS OTURUM ÇİZELGELEME PROGRAMININ GELİŞTİRİLMESİ

Ercan ATAGÜN
Düzce Üniversitesi
Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı
Yüksek Lisans Tezi
Danışman: Dr. Öğr. Üyesi Serdar BİROĞUL
Temmuz 2020, 95 sayfa

Akademik konferanslar bilimsel gelişmelerin sunulduğu, tartışıldığı bir bilimsel faaliyettir. Akademik konferanslar birden çok bilim alanı çalışmasına imkan sağladığından dolayı multidisipliner niteliktedir. Multidisipliner konferanslar, doğa bilimi, sosyal bilimler, sağlık ve sanat bilimleri gibi farklı disiplinlerde çalışmaların sunumuna izin vermektedir. Multidisipliner konferansları organize etmek için günler, salonlar ve oturumlar belirlemek ve tüm oturumları ilgili ana disiplin alanına göre sunumları belirlemek gerekmektedir. Bir oturumdaki tüm bildirimlerin aynı alandaki çalışmalardan oluşması günümüz konferans dinleyicisinin en önemli beklentilerinden birisidir. Dinleyici bu gereksinimi sayesinde ilgi alanı olan bildiriye dinlemek için katıldığı bir salonda ilgisinin olmadığı çalışmaları dinlemek durumunda kalmayacaktır. Konferans sırasında aynı oturumda farklı disiplinlerden sunumların olması konferans katılımcıları için büyük bir sorundur. Bu tez ile multidisipliner konferansların çizelgeleme probleminin Genetik Algoritma ve Epigenetik Algoritma yaklaşımı ile çözümü gerçekleştirilmiştir. Genetik Algoritmanın ve Epigenetik Algoritmanın temel kavramları verilmiş ve konferans çizelgeleme şemaları ve çizelgelemede dikkate alınacak unsurlar belirtilmiştir. Bu tez için iki farklı multidisipliner konferans veri seti kullanılmıştır. Genetik Algoritma ve Epigenetik Algoritma ile C# dilinde farklı günler, farklı oturumlar ve farklı odaların bazı kısıtlamaları altında bir uygulama geliştirilmiştir. Çalışma sonucunda Genetik Algoritma ve Epigenetik Algoritma ile elde edilen çizelgelerin sonuçları ve grafikleri verilmiştir. Epigenetik Algoritmanın Genetik Algoritmayla kıyasla daha başarılı sonuçlar verdiği gözlenmiştir.

Anahtar sözcükler: Çizelgeleme, Genetik algoritma, Epigenetik algoritma.

ABSTRACT

DEVELOPMENT OF THE GENETIC ALGORITHM BASED CONFERENCE SESSION SCHEDULE PROGRAM

Ercan ATAGÜN

Düzce University

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

Master's Thesis

Supervisor: Assist. Prof. Dr. Serdar BİROĞUL

July 2020, 95 pages

Academic conferences are a scientific activity where scientific developments are presented and discussed. Academic conferences are multidisciplinary because they allow for more than one science field study. Multidisciplinary conferences allow the presentation of studies in different disciplines such as natural science, social sciences, health, and art sciences. To organize multidisciplinary conferences, it is necessary to set days, halls and sessions, and presentations of all sessions according to the relevant main discipline. One of the most important expectations of the conference listener is that all the papers in a session are composed of studies in the same area. Thanks to this requirement, the participants will not have to listen to studies that are not of their interest in a strict hall but only to papers of their interest. Having presentations from different disciplines in the same session during the conference is a big problem for conference participants. With this thesis, the solution to the scheduling problem of multidisciplinary conferences has been realized with the Genetic Algorithm and Epigenetic Algorithm approach. The basic concepts of the Genetic Algorithm and Epigenetic Algorithm are given and conference scheduling schemes and the elements to be considered in scheduling are specified. Two different multidisciplinary conference data sets were used for this thesis. With the Genetic Algorithm and Epigenetic Algorithm, an application has been developed in C# language under different days, sessions, and some restrictions on different rooms. As a result of the study, the charts obtained with the Genetic Algorithm and Epigenetic Algorithm are given. The Epigenetic Algorithm has been observed to give more successful results compared to the Genetic Algorithm.

Keywords: Scheduling, Genetic algorithm, Epigenetic algorithm.

1. GİRİŞ

Son yıllarda teknolojinin her alanda gelişimine paralel olarak bilimsel faaliyetler artış göstermektedir. Akademik konferanslar bilimsel faaliyetlerin en temel parçasıdır. Akademik konferanslar ile bilimsel gelişmeler paylaşılarak, tartışılarak ve sunular yapılarak bilimsel gelişmelere yön verilmektedir. Bu durum akademik konferansların sayılarında artışa neden olurken aynı zamanda niteliğindeki artışları da zorunlu kılmıştır.

Akademik konferanslar benzer alanlarda bilimsel faaliyet gösteren bilim insanlarının bir araya gelerek bir organizasyon komitesi kurması ile başlar. Daha sonra bu organizasyon komitesi belirli bilimsel konuları belirleyerek bir konferans düzenlemek üzere karar vermektedir. Ardından konferansın tarihleri belirlenmektedir. Konferans için artık ilan duyurulur ve çalışma kabul edilmeye başlanmaktadır. Çalışmalar toplandıktan sonra bu çalışmaların hangi gün, hangi sırayla, hangi salonda sunulacağına karar verilmelidir. Bu işlemler için nitelikli bir çizelgelemeye ihtiyaç duyulmaktadır.

Çizelgeleme, bir amaç gözetilerek sınırlı sayıdaki kaynakların bir zaman aralığı içindeki işlere atanmasıdır. Belli zaman aralığı, bir problemdeki başlangıç ve bitiş süreleri ifade etmektedir. Belli koşullar problem çözümü için pek çok sayıdaki çözüm önerisinden yalnızca ihtiyaç duyulan çözümü tanımlamaktadır. Çizelgeleme yardımıyla kurum ya da organizasyon ürün geliştirirken veya hizmet sağlarken iç kaynakların en verimli kullanılmasını sağlamaktadır. Çizelgeleme, literatürde ve endüstride kendine çok fazla uygulama alanı bulmaktadır. Atölye çizelgeleme, ders programı çizelgeleme, iş akışı çizelgeleme, işlemci çizelgeleme, hemşire çizelgeleme, konferans çizelgeleme bazı çizelgeleme alanlarıdır.

Konferans çizelgeleme, konferansta bulunan bildirilerin sunumlarının belirli bir tarih, saat ve yer bilgisi sağlanarak sıralanması işlemidir. Konferans çizelgeleme probleminde gün, oturum ve salonların nasıl tahsis edeceği belirlenmelidir. Ayrıca bu tez kapsamında ele alınan aynı bilimsel konuların bir arada olması günümüzde artan konferanslardaki gerçek bir ihtiyaç olarak doğmuştur. Aynı oturumda yer alan çalışmaların birbirine yakın konulardan olması katılımcı memnuniyetini artırırken, konferansın ana

unsurlarından olan bilgi paylaşımını da en uygun düzeye çıkarmaktadır.

Kalabalık katılımcı sayısı, uzman kişilerin katılımları, yapılan etkinliklerin harcamaları, topluma ve bilim dünyasına sunduğu katkılar göz önüne alındığında konferansların çizelgelenmesi önemli bir sorun haline gelmiştir. Konferans çizelgelemede temel amaç konferans salonlarından etkili bir biçimde yararlanmak, dengeli ve konu odaklı çizelgelemenin yapılmasını sağlamaktır. Konferanslar için çizelgelemenin otomasyon ile gerçekleştirilmesi, konuşmacı yazarların oturumları çakışmadan ilgili oturumların konu bütünlükleri sağlanarak tüm oturumlar için istenilen koşullara uyacak biçimde sıralamanın yapılmasıdır. El yordamıyla hazırlanan çizelgelenmeler ve bunlarda herhangi bir değişiklik yapılması oldukça zor olacaktır. Yapılan her değişikliğin çizelgelemenin başka bir tarafında bir başka değişikliğe yol açması kontrol sürecini tekrar başlatacaktır. Tüm bu problemler birlikte muhakeme edildiğinde konferans çizelgelemenin bilgisayar yazılımı kullanılarak yapılması yönetilebilir, sürdürülebilir, daha az maliyetli ve daha esnek yapıda olacaktır.

Bu tez çalışması, tez konusu hakkındaki genel bilgiler ve tezin planlanmasının özet olarak sunulduğu giriş bölümü dâhil yedi bölümden meydana gelmektedir. Çalışmanın ikinci bölümünde, optimizasyon problemleri ve çizelgeleme kavramı, özellikleri ve son zamanlarda konferans çizelgeleme problemlerine ait literatür çalışmalarına yönelik derinlemesine incelemeler bulunmaktadır. Çalışmanın üçüncü bölümünde sezgisel algoritmalar ve metasezgisel algoritmalar ile ilgili genel bilgilerin yanı sıra bu algoritmalarından en çok kullanılanlar seçilerek açıklamaları yapılmıştır. Dördüncü bölümde ise, çalışmada uygulanan algoritmalar olan Genetik Algoritma ve Epigenetik Algoritma açıklanmıştır. Çalışmanın beşinci bölümünde ise konferans çizelgeleme probleminin önerilen ve bu tez kapsamında geliştirilen program belirlenen stratejilerle birlikte detaylarıyla açıklanmıştır. Ayrıca Genetik Algoritmanın ve Epigenetik Algoritmanın problem çözümü üzerindeki etkinlikleri ortaya konmuştur. Çalışmanın altıncı bölümünde ise konferans çizelgeleme gibi gerçek hayat probleminin yapısı ve bu problemin çözümüne yönelik gelişmeler tüm detaylarıyla açıklanmıştır ve sonuçlarına tablo ve grafiklerle yer verilmiştir. Epigenetik Algoritmanın Genetik Algoritmaya göre üstün noktalarının sonuçları aktarılmıştır.

2. OPTİMİZASYON PROBLEMLERİ VE ÇİZELGELEME

2.1. OPTİMİZASYON PROBLEMLERİ

Optimizasyon genel olarak verilen kısıtlar altında problemin en iyi çözümünün elde edilmesi işlemidir. Bu işlemler matematiksel kurallar yardımıyla gerçekleştirilir. Problemdeki en iyi tanımlaması göreceli bir kavram olarak ortaya çıkmaktadır. Optimizasyonda amaç sermaye, maliyet, harcanan çaba, teknoloji ve malzemenin minimize edilmesi ve kar maksimizasyonu yapmaktır. Bundan dolayı optimizasyonda hedeflenen amacın minimum ya da maksimum yapan şartları bulmaktır [1]. Belirlenen kısıtlamaları sağlayacak biçimde bilinmeyen parametre değerlerinin bulunmasına ilişkin herhangi bir problem optimizasyon problemidir. Optimizasyon işlemlerinde ilk olarak karar değişkenleri, karar parametreleri veya tasarım parametreleri olarak adlandırılan parametre setinin tanımlanması yapılır. Ardından bu parametreler bağlı olarak minimizasyon yapılacak bir maliyet fonksiyonu ya da maksimizasyon yapılacak kar fonksiyonu ve sınırlamaları içeren kısıt fonksiyonu tanımlaması yapılır. Maliyet fonksiyonu daha iyi çözüm temsil eden parametre değerlerinin kullanılması için daha düşük sayısal değer üretir. Benzer biçimde kar fonksiyonu ise çözümü temsil eden daha büyük bir sayısal değer üretir. Kısıt fonksiyonu ise parametrelerin alamayacağı değerleri tanımlamakta kullanılır. Bu fonksiyon eşitlik şeklinde veya eşitsizlik şeklinde tanımlanabilmektedir [2]. Problem için kısıtları sağlayan bütün mümkün çözümlerin bulunduğu bölge araştırma yapılabilecek uygun çözüm bölgesi olarak adlandırılmaktadır. Optimum çözüm ise en küçük olanı bulma probleminde; en düşük maliyet fonksiyonu değerine sahip çözüm olurken, en büyük olanı bulma probleminde; en büyük kar fonksiyonu değerine sahip çözüm olur. Optimum çözüm mümkün çözümlerin en iyisi olan çözüm olarak seçilmektedir [3].

Optimizasyon problemlerinin temelinde bulunan $f(x)$ amaç fonksiyonundaki x 'e bağlı olarak yapılan minimizasyon ve maksimizasyon işlemlerinde x 'in sınırlaması söz konusu ise sınırlamalı optimizasyon, herhangi bir sınırlama söz konusu değilse sınırlamasız optimizasyon denmektedir. Ayrık değerlerin yer aldığı problem türleri ayrık optimizasyon olarak adlandırılmaktadır. Ayrık optimizasyon problemlerinde en

uygun çözümleri tanımlayacak şekilde düzenlemeler yapılmaktadır. Parametrelerin, ayrık değişkenlerin sürekli değer aldığı problem türleri sürekli optimizasyonlardır. Amaç fonksiyonu ve kısıtlarla ilgili fonksiyon lineer ise doğrusal optimizasyon problemi olarak adlandırılmaktadır. Eğer bu amaç fonksiyonlarından ya da kısıt fonksiyonlarından bir tanesi lineer fonksiyon değilse doğrusal olmayan optimizasyon olarak adlandırılmaktadır [2].

Optimum çözümün nasıl olacağı problemin ilk başlarında belirlenemeyeceği, nasıl en iyi çözüme ulaşım sağlanacağı tam olarak öngörülmeyeceği, çözüm uzayının boyutunun büyük olduğundan dolayı tüm çözüm uzayının aranmasının kabul edilemez olduğu problemlere optimizasyon problemleri denir. Bu problemler deterministik yöntemlerden biri ile makul bir zaman diliminde çözüme kavuşturulamazlar [4]. Bu tarz problemlerde kabul edilebilir çözümler sunmak için başvurulan yöntemler sezgisel algoritmalarıdır.

Klasik optimizasyonda fonksiyonun maksimum ya da minimum değere ulaşması sürecini içermektedir. Bu işlem için gerek ve yeter şart fonksiyonun sürekli ve türevlenebilir olmasıdır. Sürekli olmayan ya da türev alınamayan fonksiyonlar klasik optimizasyon çözüm için yetersiz kalmaktadır. Tez kapsamında ele alınan problem sürekli fonksiyon ile ifade edilemediğinden dolayı klasik optimizasyon ile çözülmesi mümkün değildir. Tez kapsamında metasezgisel algoritmalar seçilerek optimizasyon probleminde çözüm aranmıştır.

Optimizasyon problemlerinin sınıflandırıldığı gibi bu problemlere üretilen optimizasyon metotları da bazı kriterlere göre sınıflandırılmaktadır. En genel sınıflandırılması klasik optimizasyon teknikleri ve modern sezgisel metotlar olarak iki gruba ayrılabilir. Klasik optimizasyon metotları kesin çözüm sunarken, sezgisel metotlar yaklaşık optimal çözüm sunmaktadır [5]. Sezgisel metotlar ise herhangi bir amaca ulaşmak veya bir sonuç bulmak için olasılık prensiplerine göre etkin olanı saptamaktır. Ayrıca tanımlanan kısıt ve kriterlere göre işlem yaparak kesin çözüme yakınsaması ispat edilmemesine karşın yakınsama özelliğine sahip tekniklerdir. Bir başka optimizasyon metotları sınıflandırma biçimi doğrudan metotlar ve dolaylı metotlar olarak adlandırılmaktadır. Doğrudan metotlar araştırma metotları, dolaylı metotlar ise optimal kritere dayalı metotlar olarak adlandırılmaktadır. Dolaylı optimizasyon metotları gerek ve yeter şartları belirleyip ardından bu şartları bölgesel minimum adayları için çözüm üretir. Doğrudan metotlar bir başlangıç çözümü ile başlamaktadır. Ardından bu başlangıç çözümü yüksek olasılıkla optimum çözümün gerekliliklerini yerine getirmeyeceğinden işlemler iteratif

olarak geliştirilir. Bu tür yaklaşımlarda çözüm uzayı araştırılır ve çözüm sağlanıncaya kadar iteratif olarak devam eder. Nonlineer optimizasyon metotları lineer olarak tanımlanamayan problemlerde kullanılmaktadır. Bu nedenlerden dolayı mühendislik uygulamalarında optimum tasarım için sistematik yaklaşımlar geliştirilmektedir. Bu yaklaşımlarda bir başlangıç çözümü alınıp iteratif olarak optimallik şartı sağlanıncaya kadar geliştirme yapılır [2].

Sezgisel optimizasyon, çok fazla karmaşıklığa sahip sistemlerin planlanmasını ve yönetilebilirliğinin sağlanması için optimum karar vermede kullanılabilen yapay zekâ yöntemlerini barındıran modern tekniklerdir [6]. Sezgisel optimizasyon çoğunlukla yöneylem araştırmalarının konularında yer almaktadır. Sezgisel optimizasyon bir matematiksel model oluşturulmasını gerekli kılar ve bu modele göre oluşturulan formüllere göre en kısa zamanda en iyi çözümü bulmayı hedeflemektedir. Doğrusal olmayan problemlerin ortaya çıkma sebebi bazı problemlerde analitik metotlarla gösterilmesinin zor olması ve bazen de imkansız olmasıdır. Analitik metotlarda gerek şartla yazılır ve bölgesel minimum noktalar için çözülür [7]. Nonlineer problemler üretim, hizmet ve planlama gibi gerçek dünya problemlerinde sıkça kullanılmaktadır.

2.2. ÇİZELGELEME PROBLEMLERİ

Çizelgeleme, teslim süresi belirlenmiş olan işleri hangi sırayla gerçekleştirileceği ve kaynakları hangi sırayla kullanacağına yönelik karar verme süreçlerini kapsayan problem türüdür [8]. Çizelgeleme problemi problemin kendi özgün yapısına bağlı olarak farklılık gösterebilmektedir. Bu yüzden çözüm teknikleri probleme özgü olarak geliştirilmektedir [9]. Çizelgeleme probleminde optimum çözümün bulunabileceği bir süre öngörülemiyorsa ve polinom fonksiyon olarak belirtilemiyorsa bu tür problemler NP-hard (Non-deterministic Polynomial hard) olarak tanımlanmaktadır. Bu tür problemlerde yaklaşık çözüm bulunmaktadır [10]. Çizelgeleme problemlerinde görülen bazı ortak sorunlar şu şekilde sıralanmaktadır [11]. Problemin büyüklüğü, Problemin karmaşıklığı, uygun çözüme ulaşmanın belirsizliği, uygun çözüme ulaşmadaki güçlüklerdir.

Kaynakların az olduğu ve belli bir süre boyunca işlerin tahsis edilmesine ilişkin hedeflerin optimizasyonu gerçekleştiren karar alma sürecine çizelgeleme adı verilmektedir [10]. Bazı çizelgeleme türleri, hemşire çizelgeleme [12], ders çizelgeleme

[13], iş akışı çizelgelme [14], network çizelgeleme [15], su dağıtım şebekesi çizelgeleme [16], atölye tipi çizelgeleme [17], hastane çizelgeleme [18].

Bilgisayar teknolojisi ve yapay zekâ tekniklerinin gelişimine paralel olarak, çizelgeleme (scheduling), akış kontrol (flow control), yönlendirme (routing) gibi gerçek hayat problemlerinin çözümüne yönelik uygulamaların ve akademik çalışmaların son yıllarda giderek yaygınlaştığı gözlenmektedir. Genetik Algoritma, yapay zekâda optimizasyon tabanlı algoritmalarından biridir. Genetik Algoritma yapay zekâ tanımlarında geçen anlama ve geçmiş deneyimleme kavramlarını barındırmaktadır. Deneme yanılma, parametre yapısı, kodlama yapısı, genetik operatörleri işlevsel olarak zeka kavramını icra etmektedirler. Günümüzdeki bazı çizelgeleme problemleri hala el yordamıyla gerçekleştirilmektedir. Konferans çizelgeleme gibi parametre sayısı çok olan, veri miktarı arttıkça kalite değerlendirilmesi yapılamayan problemler için yapay zekâyı ihtiyaç doğmaktadır. Sınıflandırma, çizelgeleme gibi boyutları büyük olan optimizasyon problemlerinde evrimsel hesaplama teknikleri kullanılmaktadır [11]. Canlıların evrimsel süreçlerinde üreme ve mutasyon sayesinde farklılık kazanılmaktadır. İki birey genetik bakımdan farklı olmasına karşın kromozomların yer değişmesi ile birlikte yeni bireyler oluşmaktadır. Geçmiş nesillere kıyaslandığında güçlü bireylerin ortama uyum sağlayıp yaşamını sürdürdüğü gözlenirken, uyum sağlamayan bireylerin yaşamını yitirdiği gözlenmektedir. Yaşayarak soyunu devam ettirme kavramı doğal seleksiyon olarak adlandırılmaktadır [11]. Evrimsel algoritmalar bilgisayar ortamına aktarılırken en güçlü olanı belirleme, üreme, mutasyon gibi süreçlerin ele alındığı bir çözüm ile geliştirilmektedir. Evrimsel algoritmalar rastgele süreçlerle çözüm arayan bir yapıdadır. Bu algoritmalar öğrenen yapısı ile rastgele üretilen çözümlerden farklılaşmaktadır. Ayrıca Evrimsel Algoritmalar belirsizlik durumlarını da kontrol etmektedir. Genetik Algoritmalar, Evrimsel Algoritma türlerinden biridir. Genetik Algoritma stokastik metotlar barındırmasından dolayı genellikle optimum çözüme yakın çözüm üretmektedir. Genetik Algoritma karmaşıklığı çok, kısıt sayısı fazla, amaç fonksiyonu kolay oluşturulamayan, kesin çözüm yöntemi açıkça belirtilemeyen yapay zekâ optimizasyonlarında tercih edilmektedir. Çizelgeleme gibi karmaşık problemlerin optimum çözümlerine yönelik yapılan araştırmalar halen devam etmektedir.

2.3. KONFERANS ÇİZELGELEME

Konferans, Türk Dil Kurumu sözlüğüne göre, bir konuda topluluğa bilgi verme amacı

taşıyan konuşma ve uluslararası bir sorunun çözümlenmesi için yapılan toplantı anlamları taşımaktadır [19]. Uluslararası toplantıların bilgi edinme, tartışma, sorun çözme gibi amaçlarla planlanan katılım odaklı, planlı ve bir kuruluş önderliğinde bir konu hakkında tartışma, bilgi vermek, bildiri sunmak ve fikir alıp verme işlemlerinin gerçekleştiği etkinliklerdir [20]. Konferansların amaçlarının belirli olması hangi problemlere cevap arandığının bilinmesi oldukça önemlidir. Bu durum katılımcıların konferansta neler olabileceğini öngörüp konferansa katılım başarısını artırmaktadır [21], [22]. Konferanslar, iki yönlü iletişim kurulması sayesinde katılmalı yönetim biçiminin en somut örneklerinden birisi olmaktadır. Yüz yüze etkileşim olduğundan dolayı konferansın iyi organize edilmesi çift taraflı fayda sağlayacağından çok yararlı sonuçlar elde edilebilecektir [23]. Bir konferansa katılmanın en önemli kazanım hedefleri ağ oluşturma ve aynı alanlardaki meslektaşların buluşmasıdır [24].

Verimsiz yapılan çizelgeleme politikaları kaynakların kullanımı noktasındaki en büyük zaaflık olmakla birlikte maliyetlerin artmasına da öncülük etmektedir. Bu yüzden çizelgeleme çalışmaları konferansları organize eden kurullar için bir gereksinim haline gelmiştir. Konferans salonlarının artması ile eş zamanlı pek çok bildiri yapılabilmektedir. Bu durum hem konuşmacıların hem dinleyicilerin alışkanlıklarını değiştirdiğini göstermektedir.

Türkiye’de düzenlenen konferans sayısı her geçen gün artış göstermektedir. 2017 yılında Türkiye’de 396 akademik konferans gerçekleştirilirken, 2018 yılında 1166, 2019 yılında ise 1613 tane akademik konferans gerçekleşmiştir [25]. Türkiye’de gerçekleştirilen konferanslara sağlanan katılım dikkat çekici düzeylere ulaşmaktadır. TÜRSAB’ın 2013-2018 yıllarını kapsayan raporuna göre 2018 yılında 1902089 ziyaretçi Türkiye’ye konferans amaçlı gelmiştir ve 2018 yılında 39566327 ziyaretçiden 1902089 ünü oluşturmaktadır ve bir ziyaretçi 647 dolar harcamıştır [26].

Katılımcı sayılarında görülen artışlar, alanında uzman kişilerin katılımı, düzenlenen etkinliklerin ve sunulan hizmetlerin harcamaları, konferansın topluma ve bilim dünyasına sunduğu kazanımlar göz önüne alındığında konferansların verimli bir biçimde gerçekleştirilmesi önem kazanmaktadır. Bu bakımdan konferans çizelgeleme problemi önemli bir problem olarak ortaya çıkmıştır.

Akademik konferanslar genellikle çeşitli sunumları kapsar. Oturumlar iki dinlenme arasında mola verilmeden gerçekleştirilen sunumları ifade etmektedir. Bir oturumda

genellikle 5 sunum yapılmaktadır. Bir oturum bitince yaklaşık 10 dakika ile 30 dakika arasında değişen dinlenme molası verilmektedir.

Bildiri, belli bir konu hakkında üretilen özgün sonuçları olan, bilimsel bir dille yazılan ve genellikle konuya ilgi duyan katılımcıların oluşturduğu etkinliklerde sunmak üzere yapılan çalışmalardır. Akademik bildiriler bilimsel açıdan önemli bir paya sahiptir. Bildirilerin sunulduğu ortamlarda konuya ilgi duyan katılımcıların yanı sıra o konunun uzmanları da katılabilmektedir. Bu çalışmalar yüksek lisans ve doktora programındaki öğrencilerin eğitimi açısından oldukça yararlı olmaktadır [22]. Tebliğ, Bakanlar Kurulu 14.05.2018 tarihinde 2018/ 11834 kararına gereğince “Akademik Teşvik Ödeneği Yönetmeliği” düzenlemiş ve tebliğ tanımını hakemli uluslararası bilimsel konferansta, sempozyumda veya kongrede sözlü olarak sunulan ve bunların kitabında yayımlanan tam bildiri olarak belirlemiştir [27].

Konferanslar için sunum çizelgelemenin otomasyon ile gerçekleştirilmesi, konuşmacı yazarların oturumları çakışmadan ilgili oturumların konu bütünlükleri sağlanarak tüm oturumlar için istenilen koşullara uyacak biçimde ataması yapılır. El yordamıyla hazırlanan çizelgeler ayrıca çok zor olacaktır ve değişiklik yapılması oldukça zordur. Yapılan her değişikliğin çizelgelemenin başka bir tarafında bir başka değişikliğe yol açması kontrol sürecini tekrar başlatacaktır. Tüm bu problemler birlikte muhakeme edildiğinde konferans çizelgelemenin bilgisayar yazılımı kullanılarak yapılması yönetilebilir, sürdürülebilir, daha az maliyetli ve daha esnek yapıda olması sağlanır.

Çizelgelemenin iyi yapılması istenmeyen durumların tamamından kaçınılmasını gerektirir. Kısıt, çizelgeleme probleminde istenmeyen durumlara verilen isimdir. Kısıtların artması problemin çözümünün giderek zorlaşmasına neden olur.

Konferans çizelgelemesi gibi zaman çizelgeleme problem türlerinde belirli sayıda amaçlar sert ya da yumuşak olmak üzere sınıflandırılmaktadır. Sert kısıtlar üretilecek olan çizelgelemede mutlaka uyulması gereken kurallara dikkat edildiğini ifade etmektedir. Yumuşak kısıtlar ise ihmal edilebilecek durumları kapsayan uyulmaması büyük problemler doğurmayan kısıtlardır.

Bu tez çalışmasında konferans çizelgeleme sürecinin gerçek uygulama sahasını tam yansıtabilmesi için bir takım kısıtlar ortaya konmuştur. Kısıtlar sert kısıtlar ve yumuşak kısıtlar olmak üzere iki bölüme ayrılmıştır.

Sert kısıtlar şu şekilde belirlenmiştir. Bir yazar aynı anda birden fazla oturumda yer

alamaz. Bir oturumda yer alan tüm bildiriler aynı birincil alana ait olmalı. Bir oturumda yer alan tüm bildiriler aynı ikincil alana ait olmalı. Yumuşak kısıtlar şu şekilde belirlenmiştir. Bir oturumda boş alanlar bulunmamalı. Bir oturumda yer alan tüm bildiriler aynı üçüncül alana ait olmalıdır.

Türkiye’de bilime ve bilimsel faaliyetlere verilen önem giderek artmaktadır. Bu bağlamda bilimsel faaliyetlerin önemli bir parçası olan akademik konferans düzenlemeye ilgi artmıştır. Akademik konferanslar gelişmiş ve gelişmekte olan toplumların refah ve kalkınmışlık seviyelerinde vazgeçilmez bir unsur olarak yer almaktadır.

Çizelgeleme problemleri çok sayıda konferanslarda çalışma konusu olarak ele alınmaktadır. Uluslararası Otomatik Planlama ve Çizelgeleme Konferansı (ICAPS), ilk olarak çizelgeleme temalı konferans olarak ortaya çıkmıştır ve çok sayıda yeni algoritmaların tartışılmasına imkan sunmuştur [28].

Bilimsel etkinlik olan konferansların başlıca amacının bilgiyi üretmek ve bilgi paylaşımının gerçekleşmesini sağlamak olduğu ve bilgi paylaşımı esnasında araştırma sonuçları ve deneyimler paylaşarak dinleyici ve konuşmacılar dahil tüm katılımcılarının bilgi kazanmalarını sağlanmaktadır. Çalışmasında konferans katılımcılarının konferans programı, sunumu yapılan bildiriler, konaklama imkanları, katılım ücretleri, önemli tarihler gibi unsurlara yoğunlaştığını gözlemiştir [29].

Konferansların düzenlenmesi için birden fazla iş parçacıklarına ihtiyaç duyulmaktadır. Andlauer ve arkadaşları çalışmasında bilimsel, örgütsel ve insani beceri gereksinimlerine ihtiyaç duyulduğunu belirtmektedir. Konferansın boyutunun ve süresinin ilgili komiteler oluşturularak iş parçacıklarının tanımlanmasının sağlanması gerekmektedir. Bilim kurulu oluşturularak konferansın teması, programlar, oturumlar, oturum başkanları tanımlaması yapılmalıdır [30]. Nicholls çalışmasında küçük ve orta büyüklükteki konferansların oturum başkanı tarafından planlanmasını sağlayan bir sezgisel tarama önermiştir. Çalışmada sezgisel tarama yöntemi ile yazarların çakışmasının önüne geçilmiştir. Ayrıca katılımcıların bazı tercihleri dikkate alınarak rezervasyon sistemi uygulanmıştır [31].

Potthpff ve Brams tam sayı programlama konferans çizelgeleme problemini ele almıştır. 2005 ve 2006 yılındaki Kamu Seçim Topluluğu ve New Orleans’ta gerçekleşen konferansları belirlenen zaman dilimlerine atanmasını sağlamışlardır. Bu çalışmada

gönderilen tüm bildiriler ilk önce konularına ayrılmıştır. Konularına ayrılan bu bildiriler 3'erli veya 4'erli gruplara ayrılarak belirlenen kısıtlara uyacak biçimde tam sayı programlama yardımı ile gerçekleştirilmiştir [32]. Bildiri sunumu yapanların birden fazla sunum yapmalarına izin verilmemesi önerilmiştir. Ayrıca birden fazla bildiri olduğunda bunun için ödenecek ücretin daha fazla alınması ile bu durum caydırıcı bir şekilde önlenmesi tavsiye edilmiştir. Ancak bu tez çalışması hazırlanırken elde edilen veriler incelendiğinde ve 2018 ve 2019 yılına ait konferanslar incelendiğinde bu durumun mümkün olmadığı sonucuna varılmıştır. Bir konferans boyunca bir kişiye yalnızca bir bildirim sunma önerisi henüz kabul görmemiştir [32]. Katılımcıların rezervasyonlar talep etmesi çizelgeleme probleminin çözümünü daraltmaktadır. Bu durumda kullanıcılar ilk gün, son gün, son oturum gibi belirli zaman dilimlerinde sunum yapmak isteyebilmektedir. Bu durum özel sebep gerekmedikçe optimum çözümden kaçınmaya sebep olmasından dolayı ek ücret ile cezalandırılabilir [32].

Zulkipli ve arkadaşları konferans çizelgeleme problemleri için belirlenen zaman dilimlerine sunumu yapılacak bildirilerin atanmasını sağlamıştır. Hedef Programlama (Goal Programming) modeli adını verdikleri yöntemle konferans kapasitesini ele alarak bildirilerin yerleşmesini sağlamıştır. Önerilen model ile kısıtlama kapasitesi ana faktör alınarak çözüm üretilmiştir [33]. Eglese ve Rand çalışmasında konferans çizelgeleme için tavlama algoritması içeren bir sezgisel yöntem önermiştir. Konferans çizelgelemesi yapılmadan önce konferansa katılanlardan bir anket yardımıyla katılmak istedikleri oturumları seçmeleri beklenmiştir. Konferans düzenleme kurulu tarafından katılımcıların katılmak istedikleri oturumları mümkün kılan bir çizelgeleme problemine dönüştürerek Tavlama Benzetimi Algoritması (Simulated Annealing) ile sezgisel bir çözüm sağlanmıştır [34].

İbrahim ve arkadaşları çalışmasında konferans çizelgeleme problemine üç paralel oturum gerçekleşebilecek bir kombinatoriyal tasarım önermişlerdir. Bu tasarım temelinde alt kümelerin içinde alt küme içeren çözüm önerilerinin yer alması olarak tanımlanmıştır [35]. Sampson ve Weiss çalışmasında konferansa katılımı artırmaya yönelik çizelgeleme önermişlerdir. Doğrusal programlama (Linear Programming) ile kısıtların belirlenmediğini vurgulayarak ve katılımcı tercihlerinin dikkate alarak konferans düzenleme kuruluna yardımcı bir model önermişlerdir [36].

Sampson çalışmasında tercih odaklı konferans çizelgeleme problemini ele almıştır. Katılımcıların memnuniyetinin incelendiği bu çalışmada optimize edilmeyen

oturumların kullanıcılar tarafından memnuniyetinin azaldığı gözlemlenmiştir. Konferans oturumlarının aynı konulardan seçilerek hazırlanan oturumlarda memnuniyetin arttığını belirtmiştir [37].

Thompson çalışmasında eş zamanlı oturumları planlayan bir bilgisayar programı önermiştir. Katılımcıların bazı oturumlara ilgi gösterirken bazılarında göstermediği saptandığından, katılımcı sayısı eş zamanlı oturumlarla düzenleneceği öngörülmüştür. Çalışmasında düzenlemek istediği konferansı zamanlama karmaşıklığını dikkate alarak sezgisel tabanlı ve özel bir bilgisayar algoritması adını verdiği algoritma önermiştir [38]. Bhardwaj çalışmasında konferans düzenleme kurulunun, bildiri sunumu yapacak yazarların ve katılımcıların tercihlerini birlikte ele alarak konferans çizelgeleme yaklaşımını savunmuştur [39]. Konferanstaki aynı oturumda bulunan bildirilerin benzer konularda olması gerektiğini vurgulayan bir başka çalışma Tanaka ve arkadaşları tarafından yapılmıştır. Tanaka ve arkadaşları konferansta sunumu yapılacak bildirideki anahtar kelimeleri uyumlu oturumlarda yer almasını hedefleyen bir çalışma önermiştir. Çalışmasında anahtar kelimeleri temsil eden bir özellik vektörü oluşturarak bireylerin sunumlarının uygun zaman ve konulara atanmasını Özdüzenleyici Harita Algoritması (Self Organizing Map) ile gerçekleştirmiştir [40]. Aynı oturumda benzer konuların bulunmasını sağlayacak Gruplandırılmış Genetik Algoritma ile bu durumu sağladıklarını vurgulamıştır. Ancak çalışmada optimize edilen diğer kısıtlar paylaşılmamıştır [41].

Konferans sunumlarının ve oturumlarının çizelgelenmesini içeren bir çözüm önerisi Tam Sayılı Programlama modeli ile gerçekleştirilmiştir. Çalışma ile birlikte oturumların konu başlıklarının oluşması sağlanmakta ve sunumların bu konu başlıklarınca yerleştirilmesi sağlanmaktadır. Çalışmada bir konuşmacının eş zamanlı birden fazla oturumda yer almaması ve bir gün süresince aynı zaman dilimindeki oturumların olduğu sunum sayılarının minimize edilmesi gibi iki temel kısıt göz önüne alınmıştır [42]. Stidsen ve arkadaşları, dünyanın en büyük Yöneyim Araştırma Konferansları arasında yer alan EURO-k konferanslarının çizelgelenmesini gerçekleştirmişlerdir. Çizelgeleme için ilk olarak toplanan çalışmalar benzer konular bir arada olacak biçimde konularına göre kategoriye ayrılır. Bu işlemden sonra konferansın düzenleneceği binalar ve salonlar belirlenir ve her oturumları belirli salonlarda olacak biçimde atama işlemi yapılır. Çalışmada önerilen model Karışık Tam Sayılı Programlama (Mixed Integer Programming) ile gerçekleştirilmiştir [43].

3. SEZGİSEL VE METASEZGİSEL ALGORİTMA

Anlama eylemini gerektiren işlerde robotlar dâhil bilgisayar sistemlerinin kullanılması yapay zekâ olarak tanımlanabilmektedir [44]. Yapay zekâ çözüm teknikleri, insan davranışlarını taklit ederek çözüm sunan tekniklerdir. Yapay zekâ, geçmiş bulguları kullandığından, performansı öğrenip uyarladığından, ileriye dönük planlama yapabildiğinden dolayı optimizasyon problemlerine konvansiyonel tekniklerden daha iyi, çok daha hızlı ve çok daha doğru çözüm sunmaktadır [45]. Yapay zekâ teknikleri temel alınarak optimizasyon problemlerinde uygulama geliştirme literatürde sıkça rastalanmaktadır. Sezgisel ve metasezgisel algoritmalar yapay zekâ teknikleri arasında yer almaktadır.

3.1. SEZGİSEL ALGORİTMA

Sezgisel optimizasyon, çok fazla karmaşıklığa sahip sistemlerin planlanmasını ve yönetilebilirliğinin sağlanması için optimum karar vermede kullanılabilen yapay zekâ yöntemlerini barındıran modern tekniklerdir [6]. Sezgisel optimizasyon çoğunlukla yöneylem araştırmalarının konularında yer almaktadır. Sezgisel optimizasyon bir matematiksel model oluşturulmasını gerekli kılar ve bu modele göre oluşturulan formüllere göre en kısa zamanda en iyi çözümü bulmayı hedeflemektedir. Doğrusal olmayan problemlerin ortaya çıkma sebebi bazı problemlerde analitik metotlarla gösterilmesinin zor olması ve bazen de imkansız olmasıdır. Analitik metotlarda gerek şartlar belirlendikten sonra bölgesel minimum noktalar için çözüm aranır [7]. Bir probleme ait amaç fonksiyonu doğrusal fonksiyon değilse bu problem Nonlinear problemdir. Nonlinear problemde çözüm kümesinin bulunduğu uygun alanın sınırları doğrusal olmayan sınırlar ile belirlenmiştir. Nonlinear problemler üretim, hizmet ve planlama gibi gerçek dünya problemlerinde sıkça kullanılmaktadır. Bu problemlere ihtiyaç duyulma sebepleri şu şekilde sıralanabilir [6]. Kısıt sayısı ve tasarım değişkenlerinin çok olması denklemdeki değişkenlerin sayısını artırmaktadır. Problem boyutu küçük olmasına karşın problemin gerek şartları tasarımın yapısını yüksek seviye linear yapmaktadır. Güncel mühendislik problemlerinde fonksiyonlar ile tasarım

parametreleri arasındaki bağlantı açık bir şekilde kurulamamaktadır.

Sezgisel algoritma Yunan dilinde ‘yeni metot bulmak’, ‘problemi çözmeye’ gibi anlamlar taşıyan ‘heuristic’ kelimesine dayanmaktadır. Heuristic terimi Türkçe dilinde ‘sezgisel’, ‘bulgusal’, ‘buluşsal’ gibi terimlerle kullanılmaktadır. Sezgisel algoritma ile klasik algoritma kullanılarak çözümü bulunamayan ya da çözüm maliyeti oldukça yüksek olan problemlerde tercih edilmektedir [46].

Bir probleme ait tüm olasılıkların denenmesi ile en iyi çözüme ulaşılabilmektedir. Ancak boyutu büyük problem çözümlerinde tüm kombinasyonların denenmesi ve en iyi çözümün bulunması zaman maliyeti açısından mümkün olmamaktadır. En iyi çözüme erişmek inanılması güç zaman gerektirmektedir. Bu nedenle en iyi çözüme yakın çözümleri sunan ve bunu hızlı bir biçimde yapan sezgisel algoritmalar geliştirilmiştir [47].

Sezgisel algoritmalar, optimizasyon problemlerinde en iyiye yakın çözüm sunan algoritmalar [46]. Sezgisel algoritmaların bir başka tercih sebebi ise çözümü doğrusal olarak tanımlanamayan ve fonksiyonel olarak quadratic özellik taşıyan fonksiyonlarda kullanılmaktadır. Literatürde bu algoritmaların çokça tercih edilmesinin başlıca sebepleri şunlardır [46]. Anlaşılması ve kavranılması kolay algoritmalar. En iyi çözümü kesin olarak vermezler fakat küçük sapma miktarları ile en iyiye yakın çözüm verirler. Çözüme ulaşmak için basit ve kısa yöntemler kullanırlar. Küçük problem türlerinde en iyi çözümü verebilirler. Problemlerde bir formül üretilmediği durumlarda ve kesin çözümü bulunmayan problemlerde kullanılmaktadırlar. Doğrusal olmayan ve ikinci dereceden fonksiyonla ifade edilen problemlerde çözüm bulabilirler.

Sezgisel algoritmalar farklı alanlardaki birçok problem türüne uyarlanabilmektedir [48]. Sezgisel algoritmaların hesaplama maliyetleri klasik algoritmalara göre daha azdır [49].

Klasik sezgisel algoritmalar meta sezgisellere göre daha dar kapsamlı (yerel) arama yaptıklarından ayrı olarak incelenirler [46].

Klasik optimizasyon algoritmaları genellikle esnek çözüm üretmede ve genel çözümler üretmede eksik kalmaktadır. Bununla birlikte klasik optimizasyon algoritmaları çok uzun süren çözüme yakınsama süreleri ve yerel (bölgesel) minimumlara takılma eğilimleri göstermektedir. Klasik optimizasyon algoritmalarının bu tür dezavantajları araştırmacıları sezgisel ve metasezgisel algoritmaları kullanmaya sevk etmiştir. Sezgisel algoritmalar bir çözüm üretmek için genellikle bir başlangıç noktasına veya noktalar

kümesine ihtiyaç duymaktadır [50].

Sezgisel algoritmalarından bazıları şunlardır [48], [51]: Fisher ve Jaikumar algoritması, 2-Opt algoritması, 3-Opt algoritması, Or-opt algoritması, K-Opt algoritması, Lin-Kernighan algoritması, Sweep algoritması, önce rota sonra küme yöntemi, önce küme sonra rota yöntemi, geliştirilmiş petal algoritması, Christofides algoritması.

Son yıllarda yapılan çalışmalarda sezgisel algoritmalarından çoğunlukla metasezgisel algoritmalar tercih edilmektedir. Metasezgisel algoritmalar, problem türüne göre değişime ve gelişime uğrayarak klasik sezgisel algoritmalara göre daha kolay çözüme ulaşmaktadır. Böylece metasezgisel algoritmalar çok daha verimli çözümler üretebilmektedir [46]. Klasik matematiksel optimizasyon algoritmaları genellikle optimum çözüme ulaşmayı garanti etmektedirler. Ancak sezgisel algoritmalar küçük sapma miktarları ile optimum çözüme yakın çözümleri bulmaktadır. Klasik matematiksel optimizasyon algoritmaları genellikle Tam sayılı programlama ve Doğrusal programlama olarak tanımlanabilen problemleri çözmek için geliştirilmiştir. Sezgisel algoritmalar ise doğrusal olmayan ve ikinci dereceden olduğu problemleri çözmek için geliştirilmiştir. Klasik optimizasyon algoritmaları genellikle formül geliştirilebilen problemlerde kullanılmaktadır. Ancak sezgisel algoritmalar kesin çözümü bulunmayan ve formül üretilmesi karmaşık olan problemlerde kullanılmaktadır. Klasik matematiksel optimizasyon algoritmaları yoğun hesaplama maliyetleri gerektirirken sezgisel algoritmaların hesaplama maliyetleri daha düşüktür.

Klasik matematiksel optimizasyon algoritmalarını hatalı sonuç verme ihtimali bulunmazken sezgisel algoritmalar kimi durumlarda hatalı sonuçlar üretebilmektedir.

Klasik matematiksel optimizasyon algoritmalarında esnetebilme işlemi sert kurallar ile birlikte yapılabilmektedir. Ancak sezgisel algoritmalar çok daha kolay bir biçimde esnetilerek farklı problem türlerine uyarlanabilmektedir.

Klasik matematiksel optimizasyon algoritmalarında probleme ait olan kısıtlarda herhangi bir kısıta ilişkin ihmal yapılması mümkün değildir. Ancak sezgisel algoritmalarda bazı kısıtlara ilişkin düzenleme yapılabilir, bazı kısıtlar atılabilir.

Klasik matematiksel optimizasyon algoritmaları kavram bakımından karmaşık bir yapıya sahip olabilmektedir. Ancak sezgisel algoritmalar kavram bakımından çok basit ve anlaşılabilir niteliktedir.

Klasik matematiksel optimizasyon algoritmaları uzun çaba gerektiren matematik

çalışmaların bir neticesi olarak geliştirilmiştir. Ancak sezgisel algoritmalar doğal yaşamdan veya hayatın içinden ilham alınarak geliştirilmiştir [46].

Sezgisel algoritmalar bir sistemde bulunan matematik modellerinin tam olarak bilinemediği durumlarda bir amaç fonksiyonu ile parametre üretilmesini sağlamaktadır. Sezgisel algoritmalarındaki bu parametreler çözüm uzayında geliştirilen algoritmanın temel ilkelerine göre arama yapmaktadır. Tabu Arama Algoritmasında uygulanmış olan parametre vektörlerinin tekrar denenmemesi, Yapay Arı Kolonisi Algoritmasında arıların hareket stratejileri, Karınca Kolonisi Algoritmasında en kısa yolun belirlenmesinde çözüm uzayları hep algoritmaların temel ilkelerine göre çözüm uzayında arama yapmalarına birer örnektir. Bu tip problemlerde çoğunlukla amaç fonksiyonu için hata (ceza) fonksiyonları kullanılır. Belirlenen cezaya minimizasyon sağlamak için çözüm uzayında arama yapılır [52].

Mühendislik problemlerinde veya endüstrideki diğer problemlerde kesin çözümü bulma işlemi aşırı fazla zaman alan veya kesin çözüm hakkında net bir tanımlamanın yapılamadığı durumlarda sezgisel algoritmalara ihtiyaç duyulmaktadır [4].

Sezgisel algoritmalar, bir amaca yönelik alternatif çözüm adımlarını kullanarak etkili olanlara karar veren algoritmalarlardır. Bu algoritma türü çözüm uzayında optimum çözüme yakın olması ispat edilemeyen algoritmalar olarak da tanımlanmaktadır. Sezgisel algoritmalar yakınsama özelliği taşımaktadırlar ancak kesin çözümü garanti etmezken kesin çözüme yakın olan bir çözümü garanti edebilmektedir [2]. Sezgisel algoritmalar, optimizasyon problemlerinde kesin çözümü bulma işleminin klasik yöntemlerle tanımlanamadığı yapılarda kullanılmaktadır. Anlaşılabilirliğin ön planda olduğu bazı optimizasyon problemlerinde sezgisel algoritmalar karar verici olarak kullanılmaktadır. Sezgisel algoritma, tek başına ya da birden fazla algoritma ile problem çözümü olarak uygulandığında çözüm kalitesi ve hesaplama zamanı, kod basitliği ve gerçekleştirilebilirlik, esneklik, dinçlik, basitlik ve analiz edilebilirlik, etkileşimli hesaplama ve teknoloji değişimleri gibi kriterler eşliğinde değerlendirilebilmektedir [2]:

Çözüm Kalitesi ve Hesaplama Zamanı: Tüm algoritmalarının etkinliği kontrol edilirken kullanılan bir kriterdir. Bu yüzden iyi bir algoritma ayarlanabilen değişken setine sahip olmalı ve bu değişkenler kullanıcı tarafında hesaplama maliyeti ile çözüm kalitesi arasında yer almasını sağlamalıdır. Böylece bir algoritmanın hesaplama zamanı ile çözüm kalitesi ilişkisi değerlendirilebilir olmalıdır.

Kod Basitliđi ve Gerçeklenebilirlik: Kullanılan algoritmanın prensipleri yalın ve geniş ölçüde uygulanabilir olması gerekmektedir. Bu özellik sayesinde çok bilgi ile algoritmayı yeni uygulama alanlarına taşımak mümkün olacaktır.

Esneklik: Algoritma geliştirilen çözüm modelinde kısıt fonksiyonları ve amaç fonksiyonlarında yapılan deđişikliklere karşı oldukça pratik olması beklenmektedir.

Dinçlik (Robustness): Algoritma başlangıç çözüm seçimine bađlı olmadan her zaman kabul edilebilir ve kaliteli çözümleri üretme yeteneđine sahip olması beklenmektedir.

Basitlik ve Analiz Edilebilirlik: Basit algoritma kolay anlaşılacağından dolayı analiz edilmesi de kolay olacaktır. Buna karşın yapısı karmaşık olan algoritma analiz edilmesi de daha zor olacaktır. Dolayısıyla algoritmanın analiz edilmesinin basit olması beklenmektedir.

Etkileşimli Hesaplama ve Teknoloji Deđişimleri: İnsan makine etkileşimini etkin kullanarak deđerlendirme yapmak oldukça yaygın bir yöntemdir. Etkileyici bir kullanıcı arayüzü ve algoritmayı grafiksel olarak sunmak etkin bir deđerlendirme biçimi olmaktadır.

Birçok gerçek dünya problemlerinde karmaşıklik seviyesi probleme ait olan deđerşken türlerine ve deđerşken sayılarına bađlı olmaktadır. Bir problemde deđerşken sayıları arttıkça problemin çözümü ve hesaplanması giderek güçleşmektedir. Problemin boyutu arttıkça çözümün klasik optimizasyon yöntemleri ile çözmek etkisiz olmaktadır. Böyle durumlarda sezgisel ve metasezgisel algoritmalara ihtiyaç duyulmaktadır [53].

Sezgisel algoritmalar, mühendislik bilimlerinde, bilgisayar bilimlerinde, yapay zekâ ve optimizasyon problemlerinde, arama işlemlerini sezgiye ve buluşa dayandıran problemlerde tercih edilmektedir. Bu algoritmalaradaki başarı ve performans kriteri sezgisel yaklaşımın kalitesine bađlı olarak deđerşikik göstermektedir [54]. Bu bakımdan dolayı sezgisel algoritmaların arama uzayında optimum çözüme yakınsadığını ispat etmek mümkün deđildir. Ancak sezgisel algoritmalar, genellikle en iyiye yakın çözümleri kabul edilebilir bir sürede sundukları için tercih sebebi olmaktadır [4]. Sezgisel algoritmalar, deterministik olarak ya da stokastik olarak bir başlangıç çözüm kümesi oluştururlar [52].

Sezgisel algoritmaların büyük bir kısmı başlangıç olarak tamamıyla rastgele bir çözüm oluştururlar. Rastgele ifadesi bilgisayar birimlerinde tamamen ilgisiz çözümlerin bir arada olması anlamına gelmemektedir. Rastgele terimi bilgisayar bilimlerinde bir

kümedeki tüm verilerin eşit şansı olduğu anlamına gelmektedir. Sezgisel algoritmalarda ise rastgele ifadesi bir çözüm kümesindeki tüm uygun çözümlere, uygun olmayan çözümlere ve en iyi çözüme başlangıçta eşit şans tanımaktadır. Bu durum bir çok sezgisel algoritmada bu şekildedir. Bununla birlikte bazı sezgisel algoritmalar başlangıç çözümünü bazı kriterler göre oluşturabilmektedir. Sezgisel algoritma yeni değerleri üretirken, çözüm uzayındaki bir sonraki adım için çözüm noktasını üretmektedir. Ancak bazı durumlarda ilgili çözüm uzayındaki noktadan çok daha farklı bir çözüm noktası üretebilmektedir. Problem için uygun sezgisel algoritma belirlendikten sonra algoritmanın temel ilkelerine bağlı kalınarak düzenlemeler yapılmalıdır [52].

Sezgisel algoritmaları performansını ve kullanım etkinliği değerlendirebilmek için çok sayıda kritere bulunmaktadır. İlk olarak çözülmeye çalışılan problem birden fazla mümkün çözüm içeriyorsa sezgisel algoritmanın bu mümkün çözümlerin arasından en iyisini bulabilmesidir. Bu durum literatürde optimallik olarak adlandırılmaktadır. Bir başka değerlendirme kriteri ise sezgisel algoritmanın çözülmesi gereken problemdeki tüm çözümlere ulaşabilme potansiyelidir. Bu durum bütünlük olarak tanımlanmaktadır.

Bir diğer değerlendirme kriteri ise ulaşılan çözümün optimum çözüme olan yakınlık derecesidir. Bu durum aynı koşullar altında yinelenen deneylerde aynı sonucu veya aynı sonuca olan yakınlık derecesidir. Hesaplama karmaşıklığı ise sezgisel algoritmalarda problemlerin kabul edilebilir zaman dilimlerinde çözülmesini kapsamaktadır. Ancak bir sezgisel algoritma farklı özelliklerdeki problemlerde cevaba ulaşma süreleri ve zaman karmaşıklığı bakımından farklılık gösterebilmektedir. Bu değerlendirme kriterleri göz önünde bulundurulduğunda bir problem ile karşı karşıya kalındığında hangi sezgisel algoritmanın seçileceği önem kazanmaktadır.

Sezgisel algoritmalar döngüsel bir yapıya sahip olmalarından dolayı her döngü adımında başlangıçtaki çözüm adayına ya da çözüm adaylarına bağlı olarak önce bölgesel en iyi çözüm üretir. Döngü adımları iteratif olarak iyileştigiinden dolayı bu bölgesel en iyi çözümlerin global en iyi çözüme ulaşması beklenir. Ancak bu bölgesel en iyi çözümler global en iyi çözümlerden çok uzak noktalarda yer alabilmektedir. Bu problemin önüne geçmek için başlangıç çözüm adaylarının yeniden başlatılması, mevcut noktaların komşuluk bilgilerinin incelenerek bu bölgesel en iyi çözümlerden kurtulması gibi yöntemler uygulanabilmektedir [2].

3.2. METASEZGİSEL ALGORİTMALAR

Meta ismi Yunan dilinde “yüksek seviyeli” anlamına gelmektedir. “Meta” ön eki bu algoritmaların “yüksek seviyeli” sezgiseller olduğunu belirtmektedir [4].

Metasezgisel algoritmalar, zeki optimizasyon tekniklerinin modern bir kullanımı olarak ortaya çıkmıştır. Metasezgisel algoritmalar 3 önemli özellikleri bünyelerinde taşımaktadırlar. Bunlar, çözüm uzayındaki komşuluk bilgilerinin keşfedilmesi hafızanın etkin kullanımı, ulaşılan bir çözümün bir sonraki adıma aktarılması [55].

Metasezgisel algoritmalar yapısı gereği kesin çözüm verme garantisi bulunmamaktadır. Ayrıca bu algoritmalar her çalıştırılmasında aynı performans göstermesi beklenmeyen algoritmalarlardır. Ancak metasezgisel algoritmalar uygulandıkları problemleri optimize ederken genellikle kullanışlıdır. Büyük boyutlu problemlerde ve yapısı karmaşık problemlerde kesin çözüm bulmak kabul edilmesi mümkün olmayan zaman maliyeti gerektirmektedir. Metasezgisel algoritmalar böyle boyutu büyük ve yapısı karmaşık problemleri çok daha makul zaman dilimlerinde çözüme kavuştururlar. Bu yüzden büyük boyutlu ve karmaşık problemlerde en kullanışlı yöntem Metasezgisel algoritmalar [56].

Metasezgisel algoritmalar, temel sezgisel algoritmaları daha üst bir çatı altında toplayarak çözüm uzaylarının etkin bir biçimde araştırma ilkelerini amaçlayarak oluşturulmuş algoritmalar [57]. Metasezgisel algoritmalar, tek tip problemlere özgün olmamaları, bölgesel (yerel) minimum noktalarından sakınabilme özellikleri, aday çözümlerdeki çeşitlilik sağlayabilmeleri, çözüm sürelerinde makul iyileştirme yapabilmeleri nedeniyle klasik optimizasyon algoritmalarına tercih edilebilmektedir. Güncel hesaplama teknikleri ile birlikte metasezgisel algoritmaların günümüz optimizasyon problemlerinde kullanılması oldukça yaygınlaşmıştır [50].

Tek bir noktadan başlayarak çözüm arayan metasezgisel algoritmalar olmakla birlikte çok sayıda nokta ile çözüme başlayan popülasyon tabanlı metasezgisel algoritmalar da bulunmaktadır. Ancak metasezgisel algoritmaların çoğunluğu popülasyon tabanlı olmaktadır. En yaygın kullanım ise biyolojik tabanlı olan ve evrimsel ilkelerin benimsendiği metasezgisel algoritmalar [50].

Popülasyon tabanlı metasezgisel algoritmalarda başlangıçta rastgele biçimde popülasyon oluşturulur. Popülasyonda bulunan üyeler genelde çözü uzayına rastgele bir biçimde dağılmış olması beklenmektedir ve ilk değerleri almaktadır. Ardından mevcut

popülasyondan bir kısmı ya da tamamı seçilerek bu popülasyondaki bireylerin yardımıyla yeni bireyler türetilir. Yeni popülasyondaki üyelerin elde edilmesi çeşitliliğin artmasını sağlamaktadır. Ancak bu yeni nesillerin elde edilmesi pek çok algoritmaya göre farklılık içermektedir. Bir çok algoritmanın geliştirilmesi bu yeni popülasyon üyelerinin oluşması ile şekillenmiştir. Örneğin popülasyondaki yeni üyelerin oluşturulması belirli bir olasılık eşliğindeki çaprazlamayla, yeni popülasyondaki bireylerin belirli bir olasılığa göre diğer üyelere bağımsız bir şekilde üretilmesiyle, yeni popülasyondaki bireylerin önceden belirlenmiş bir kural süzgecinden geçirilmesi ile oluşturulabilmektedir. Popülasyondaki yeni üyeleri ya tamamen eski üyelerin yerine geçebilmektedir ya da eski popülasyon üyelerinin bir kısmının yerine geçebilmektedir. Bu şekilde tam bir döngü işleminin en başına dönülmesi sağlanmaktadır. Böylece iteratif döngü belirlenen bir maksimum iterasyon sayısına ulaşıncaya kadar ya da gereksinim duyulan kriterlerin sağlanması durumuna kadar sürmektedir. Optimizasyon işlemlerinin sonucunda ulaşılan sonuç optimum veya belirlenen şartları sağlayan bir çözüme yakın değerler oluşabilmektedir [50].

Metasezgisel algoritmalarda yeni bireyler elde edebilmek için işleme alınacak üyelerin seçimi, yeni popülasyon oluşturma, yeni üyelerin belirlenmesinde ihtiyaç duyulan üyeler gibi işlemlerin tamamında rastgele değerler yardımıyla işlemler sürdürülmektedir. Bunun yanında popülasyondaki bireylerin sahip olduğu özelliklerin aktarımında hafızadan da etkin bir şekilde yararlanmak mümkün olmaktadır. Optimizasyon işlemleri döngü biçimde yapıldığından her adımda metasezgisel algoritma elde edilen değerlerin hepsinin ya da bir bölümünün hafızaya kaydedilmesini sağlayabilmektedir. Hafızaya kaydı yapılan bu değerler ilerleyen adımlarda sonucun daha etkin olmasında pay sahibi olabilmektedir [50].

Sezgisel kavramı bulma ve deneme yanılma sonucunda keşif yapma anlamlarını taşımaktadır. Metasezgisel kavramı ise üst düzeyde bulmak ve keşif yapmak anlamlarındadır. Sezgisel algoritmaların birleşimi ve yönetilmesi ile birlikte metasezgisel algoritmalar meydana gelmektedir [58]. Metasezgisel, çözüm uzayında arama yapmak ve çözümdeki çeşitliliğin artmasını sağlamak ve ayrıca optimum değerleri keşfetmek, ondan faydalanmak için alt seviye sezgisel yaklaşımların birleştirilmesi sürecini kapsamaktadır [59].

Literatürde yer alan tanımlar incelendiğinde metasezgisel, problem çözümü için çözüm uzayında verimli bir arama işlemi yapmak ve bölgesel en iyilerden sakınmak için alt

seviye sezgisel algoritmalar rehberlik edilmesi olarak tanımlanmaktadır [58].

Metasezgisel algoritmalar problemlerden bağımsız bir içimde uygulanabilen algoritmalar. Metasezgisel algoritmalar problemde önceden optimum çözümün ne olduğuna ilişkin ya da optimum çözüme giden yolun ilkesel bir tanımlamasının yapılamadığı durumlarda kullanılmaktadır. Çözülmesi beklenen problem hakkında çok az sezgisel bilgi olduğu durumlarda metasezgisel algoritmaları kullanmak birer ihtiyaç olmaktadır. Metasezgisel algoritmalar ile bir çözüme ulaşıldığında bunun iyi bir çözüm olup olmadığı sınıanabilmektedir [4].

Metasezgisel algoritmalar birbirinden farklı da olsa bazı temel özellikler yönünde benzerlikler göstermektedir. Metasezgisel algoritmalar doğal yaşamdan ve insan yaşamından esinlenmiştir. Ayrıca bu algoritmalar stokastik mekanizma veya stokastik unsurlar içermektedir. Metasezgisel algoritmalar basit bölgesel arama tekniklerinden evrilerek karmaşık öğrenme tekniklerine kadar geniş bir perspektife sahip algoritmalar. Bununla birlikte bu algoritmalar problemi en doğru biçimde çözmek için bir veya birkaç parametreye gereksinim duymaktadır. Metasezgisel algoritmanın bir problem çözümünde başarıya ulaşabilmesi için çözüm uzayındaki çeşitliliklere erişebilmesi ve en iyi çözümü aramaya yoğunlaşabilmesi arasındaki güçlü dengeyi ve koordinasyonu sağlaması beklenmektedir. En iyi çözümün aranabilmesi için çözüm uzayında belli bölgelerin belirlenmesine ihtiyaç duyulmaktadır. Metasezgisel algoritmaların döngü yapılarındaki ilerleyen iterasyonların belirli bir deneyim elde etmesi sağlanmaktadır. Bu deneyimler çözüm aramadaki yoğunlaşmanın ana unsurlarından biri olmaktadır. Birçok metasezgisel algoritmanın farklılaşması ve yeni algoritmalar geliştirilmesine olanak sağlayan durum bu yoğunlaşma için belirlenen stratejilerden kaynaklanmaktadır [60].

Metasezgisel algoritmaların ortak özelliklerinden bazıları şunlardır [61]. Metasezgisel algoritmada genel hedef, çözüm uzayında etkin bir arama yapmak ve en iyi çözüme yakın çözümleri elde etme etmektir. Metasezgisel algoritmalar arama işlemleri devam ederken aramaya rehberlik yapan mekanizmalara sahiptir. Metasezgisel algoritmalar, deterministik özellik taşımayıp genelde yaklaşık çözümler sunmaktadır. Metasezgisel algoritmalar yerel arama bölgelerinden başlayarak karmaşık öğrenim stratejilerine kadar geniş bir mekanizmaya sahiptir. Metasezgisel algoritma, bölgesel en iyi noktalara takılmayı engelleyerek stratejileri bünyelerinde barındırmaktadırlar. Metasezgisel algoritma, tek tip problemlere özgün olmayıp genelde tüm kombinatoriyal problemlere

uygulanabilmektedir. Metasezgisel algoritmalar, sezgisel algoritmaların arama işlemlerinde rehberlik yaparken hafızadan yararlanmaktadır. Metasezgisel algoritmalar, çözüm uzayının optimum çözüm için en olası noktalarında arama yapmasını sağlamaktadır.

Metasezgisel algoritmalar, kombinatoriyal eniyileme problemlerinde yaygın olarak kullanılmaktadır. Metasezgisel algoritmalar pek çok farklı alanlarda kullanılmaktadır [62]. Güncel mühendislik problemleri, elektrik ve elektronik alanındaki topolojik eniyileme problemlerinde kullanılmaktadır. Telekomünikasyon tasarım problemlerinde, akışkanlar dinamiği problemlerinde kullanılmaktadır. Biyoinformatik alanındaki yapay zekâ uygulamalarında kullanılmaktadır. Hesaplamalı biyoloji problemlerinde, finans ve ekonomi problemlerinde kullanılmaktadır. Sistem tasarım ve modellemede, sinyal ve görüntü işleme problemlerinde, araç ve sürücü rotalama problemlerinde kullanılmaktadır.

Metasezgisel algoritmalar bir eniyileme problemlerine çözüm sunmaktadır. Bu algoritmalar problemin ana hedefi olan bir amaç fonksiyonunu maksimize veya minimize etmekle yükümlüdür. Metasezgisel algoritmalar, yasaklı arama algoritması, Tavlama Benzetim Algoritması gibi arama uzayında iteratif olarak eniyilemenin yapıldığı gibi veya Genetik Algoritma, Karınca Kolonisi Algoritması gibi popülasyon tabanlı da olabilmektedir [62].

Metasezgisel algoritmalar bir probleme uygulanırken amaç fonksiyonu problemin tüm kısıt ve kriterlerine ilişkin bilgiler içermektedir. Amaç fonksiyonun ulaşacağı değerlerin her biri çözüm uzayında anlamlı ve gerçek bir değeri ifade etmektedir. Bu yüzden amaç fonksiyonun uygun bir biçimde tanımlanması, metasezgisel algoritmanın çözüm uzayında bulunan ve problemin kriterlerine ve kısıtlarına uymayan bir başka ifadeyle kabul edilmeyen çözümlere erişilmesine engel olabilecektir.

Metasezgisel algoritmaların uygulandığı eniyileme problemlerinde kısıtlar problem çözümü için oldukça önemlidir. Metasezgisel algoritmalarda kısıtların tanımlanması bir fonksiyon eşitliği, fonksiyon eşitsizliği, doğrusal olan durumlar, doğrusal olmayan durumlar gibi farklı niteliklerde tanımlanabilmektedir. Ayrıca probleme ait olan bir durumun olması ya da olmaması gibi durumlar için de problem kısıtlarını tanımlamak mümkün olmaktadır. Problem kısıtları bir metasezgisel algoritmaya aktarılırken çözüme ait olan gösterileri ve aynı zamanda amaç fonksiyonunun değerlendirilmesi adımlarında

önemli olmaktadır. Bu kısıtlar değerlendirilirken farklı mekanizmalara göre kullanılmaktadır. Bunlardan bazıları; ceza mekanizmaları, reddetme mekanizmaları, onarım mekanizmalarıdır. Ceza mekanizmalarında kısıtlara aykırı olan çözümler incelenmektedir. Bu kısıtlara aykırı olan çözümlere bir ceza değeri eklenmektedir. Reddetme mekanizmalarında çözüm uzayında arama işleminde sadece uygun olan çözümler ele alınmaktadır ve kısıtlara aykırı bulunan çözümler göz ardı edilerek direkt olarak elimine edilmektedir. Onarım mekanizmasında ise kısıtlara aykırı olan çözümlerle karşılaşılması durumunda çözüm için uygun bir biçime dönüştürülmesi için bazı yöntemlere gereksinim duyulmaktadır [62].

Metasezgisel algoritmaların başarı performansının büyük bir kısmı parametrelerin amaca uygun olarak tanımlanması ve düzenlenmesine bağlı olmaktadır. Literatürde genellikle parametrelerin değerleri sabitlenmiştir. Ancak bu parametreler problem yapısına göre değişiklik gösterebilmektedir. Parametreler sabit parametreler, dinamik parametreler ve adaptif parametreler olmak üzere üç farklı kategoriye ayrılmaktadır. Sabit parametreler metasezgisel algoritmanın çalışma süresi boyunca sabit kalmaktadır. Dinamik parametreler bir başlangıç değerine sahip olduktan sonra algoritmanın çalışma süreci içinde belli katsayılarla çarpıldığından dinamik olmaktadır. Adaptif algoritmalar ise algoritma çalışma sürecinde en iyi çözümü doğru arama yapılırken çözümü daha iyi ye doğru taşıyan ya da taşımayan şartlara göre uyum sağlayan parametrelerdir [62].

Bir metasezgisel algoritma bir probleme uygulanırken arama uzayını iyi bir biçimde keşfetmesi ve güçlü, derin bir biçimde arama işlemini sürdürmesi beklenmektedir. Problemin boyutunun artması birçok problem türünde işlemleri zorlaştırırken kombinatoryal eniyileme problemlerinde arama uzayları rassal olarak büyümektedir. Bu büyüme hesaplama karmaşıklığının artması, zaman maliyetinin artması gibi sorunlara yol açmaktadır. Bu durum kombinatoryal eniyileme problemlerinde metasezgisel algoritmaların kullanımını zorunlu hale getirmektedir [62].

Klasik optimizasyon algoritmaları çoğunlukla test fonksiyonlarında çok güçlü ve sabit varsayımlar barındırmaktadır. Bu durum klasik optimizasyon algoritmalarının kullanım alanını daraltmakta ve kısıtların çok olduğu ve kısıtların karmaşık yapıda olduğu problemlerde kullanımını güçleştirmektedir. Ancak metasezgisel algoritmalar varsayım gereksinimleri oldukça az olan algoritmalarlardır. Bu durum bir problemin klasik algoritmalarla veya klasik optimizasyon algoritmaları ile çözüme kavuşamayacağı durumlarda bir başka ifadeyle son başvuru yöntemler metasezgisel algoritmalarlardır.

Bu durum her geçen gün metasezgisel algoritmalar ekosistemine katılan yeni algoritmalarla desteklenmektedir [4]. Böylece metasezgisel algoritmalar deęişen dünya şartları ve gereksinimleri karşısında deęişen ve karmaşıklık seviyesi artan güncel problemlerde, yüksek seviyede önemli problemlerde metasezgisel algoritmaların vazgeçilmez bir başvuru kaynağı olduęu ortaya koymaktadır.

Çözümün iyileşmesi için mevcut bu yöntemlerin kullanılması hala çözümün geçerlilięi iyileştirmiyorsa özellikle son yıllarda çözüm uzayının daha etkin kullanılması için farklı stratejiler geliştirilmiştir. Bunlar sezgisel algoritmaların etkin kullanımına baęlı olan metasezgisel algoritmalar ve hiper sezgisel algoritmalarıdır.

Çizelgeleme problemlerinde çoęunlukla sezgisel ve metasezgisel algoritmalar kullanılmaktadır. Bu algoritmalar en iyi çözümleri garanti etmemekte fakat kısa sürede en iyi çözüme yakın çözümleri bulabilmektedir. Çizelgeleme problemi gibi çok büyük boyutlu problemlerde tüm kombinasyonların elde edilmesi ve en iyi çözümün tespit edilmesi ve kabul edilmesi çok güç zaman dilimlerinde olacaktır. Bu yüzden çok daha kısa sürede en iyi çözüme yakın çözümlere ulaşılmasını sağlayacak metasezgisel algoritmalar uygulanmıştır. Literatürde Genetik Algoritma [63], Epigenetik Algoritma[64], Tabu Arama Algoritması [65], Karınca Kolonisi Algoritması [66],Yapay Bağışıklık Algoritması [67], Yapay Arı Kolonisi Algoritması [68], Memetik Algoritma [69], Diferansiyel Gelişim Algoritması [70], Kedi Sürüsü Optimizasyonu Algoritması [71], Kör Fare Algoritması [72], Water Waves Optimization Algoritması [73], Harris Hawks Optimization Algoritması [74] gibi çok fazla metasezgisel algoritma bulunmaktadır. Bunların her biri farklı çalışmalarda farklı problem alanlarına uygulanarak nonpoliminal olan problemlerin çözümünde başarılı oldukları görülmüştür. Aşağıdaki alt başlıklarda en çok kullanılan algoritmalardan bir kaçına kısaca değinilmiştir. Bu tez çalışmasında kullanılan Genetik Algoritma ve bu algoritmanın hibrit bir yapı çerçevesinde ortaya konmuş olan Epigenetik Algoritma bir sonraki bölümde detaylı olarak anlatılmıştır.

3.2.1. Karınca Kolonisi (Ant Colony) Algoritması

Karıncalar, doğal ekosistem içerisinde toplu halde yaşamaktadırlar. Karıncalar yaşamlarında belirli bir iş yükü paylaşımı yapmaktadır. Bu paylaşım ile yaşam boyu karşılaştıkları pek çok sorunu çözmelerini sağlamaktadır. Karıncalar görebilme yeteneęi tam olmamasına rağmen yuvalarından bir yiyecek bulunan bölgeye, yiyecek bulunan

bölgeden yuvaya olan en kısa yolu bulabilmektedir. Bu yollar arası dinamik bir biçimde bir engel veya başka bir nedenden dolayı kapanırsa karıncaların yeniden en kısa yolu bulmaları gerekmektedir. Bu durum değişen doğa koşullarına karşı onları adapte olabilmeleri yaşamlarını sürdürmek için zaruri hale getirmektedir. Karıncaların gıda ihtiyacını karşılayacağı yer ile yuvalarının arasındaki yolu tespit edebilmeleri için gereken temel kimyasal bileşen feremondur. Karıncalar yol boyunca hareket ettikleri noktalar feremon adı verilen kimyasal bileşenlerden bırakmaktadır. Bir gıda hedef noktasına ulaşabilmeleri için bırakılan bu feremon miktarı oldukça önem kazanmaktadır. Karıncaların bu feremon miktarının çok olduğu bölgeyi takip etme ihtimali oldukça yüksektir. Bununla birlikte feremon miktarının az olduğu bölgeler gidebilme ihtimali oldukça düşüktür [2].

Karıncı Kolonisi Algoritması kombinatoriyal optimizasyon problemlerinde sıkça kullanılan ve Marco Dorigo tarafından ortaya atılmış bir metasezgisel algoritmadır. Bu algoritma karıncaların yiyecek arama işlemleri temel alınarak geliştirilmiştir [76]. Karıncı Kolonisi Algoritmasının temel adımları şu şekildedir [2]:

Adım 1. Parametreleri tanımla ve hatların feromon miktarlarına başlangıç değeri ataması yap ve sayacı sıfır ataması yap.

Adım 2. Aşağıdaki adımları durdurma kriterleri sağlanıncaya kadar tekrar et.

Adım 3. Bütün karıncalar için feromon maddesine bağlı olarak yollar üret.

Adım 4. Bütün yolların uzunluklarını hesapla.

Adım 5. Yolların uzunluklarına göre bu yollarda mevcut olan feromon maddesinin miktarını güncelle.

Adım 6. Bütün yollardan en kısa olanı hafızaya yaz.

Adım 7. En kısa yolu yaz ve dur.

Başlangıçta parametreler tanımlanır ve bütün feromon değişkenleri başlangıç değeri olan Y 'nin ilk değerine eşitlenmektedir. Karıncı Kolonisi Algoritması kombinatoriyal optimizasyon problemlerinde yaygın biçimde kullanılmaktadır. Bu algoritma kombinatoriyal optimizasyon problemlerinde yüksek bağlantı seviyeli bir graf veri yapısı ile tanımlanmaktadır. $G(Y, K)$ olarak tanımlanan bir graf yapısında tepeler çözümün yapısını, kenarlar da kimyasal bileşenler arasındaki bağlantılar olacak biçimde tanımlansın. Burada Y tepelerin kümesini, K ise kenarların kümesini tanımlamış olsun.

Ulaşılan bir çözüm graf yapısı üzerinde uygun bir yürüyüş olarak belirtilmektedir. Çözüme ait her bir bileşen V_{ij} , b_{if} değerine sahip bir Z_i değişkenini tanımlamaktadır. Karıncaların optimum olarak yürüyüşlerinin optimum olan bileşenlerin kombinasyonuna ihtiyaç duymaktadır. Her çözüm unsuru V_{if} 'ye bir Y_{if} feromon miktarı karşılık gelmektedir. Bu bir hafıza biçimi olarak değerlendirilebilir. Feromon miktarı belli zaman dilimindeki karşılık geldiği çözüm bileşenine cazip miktarın olduğunu ifade etmektedir. Bütün çözüm bileşenleri V ve feromona ait olan tüm iz parametreleri Y ile ifade edilmektedir [4]. Karıncaların çözümleri oluşturmasında belli sayılarda yapay karınca, kademeli ve rastgele olarak üretilen başlangıç çözümün boş kümeden itibaren çözümü türetilir. Türetilen bu çözümler her adımda uygun bileşenin eklenmesi ile genişletilir. Seçilecek olan bileşen feromon miktarı göz önünde bulundurularak yapılacak olasılıksal bir yöntem ile belirlenmektedir.

3.2.2. Parçacık Sürüsü Optimizasyonu Algoritması

Parçacık Sürüsü Optimizasyonu (PSO), temel kurallar kullanarak hareketini sağlayan çok sayıdaki parçacıklardan ilham alınarak doğa olayı temelli metasezgisel bir algoritmadır [2]. Parçacık Sürüsü Optimizasyon Algoritmasında sürü davranışları bilgisayar ortamına aktarılmıştır. Bu algoritmanın yapısı kuş, balık gibi bir sürü halinde olan canlıların yiyecek aramalarını temel almaktadır. Sürüdeki canlıları birbirleri ile kolektif bir uyum sergilemektedirler. Sürüdeki canlıların davranışları zekâ kavramının oluşmasına neden olmaktadır. PSO algoritması, bu zekânın yardımıyla en kısa olan noktaları aramaktadır. Bu algortmada sürüde yer alan her bir parçacığın hareketini matematiksel olarak modellenmektedir. Böylece parçacıkların en iyi yolu belirleyerek en iyi çözümü elde etmeleri hedeflenmektedir. Her bir parçacıkta yer alan konum bilgisi arama uzayında bir çözümü ifade etmektedir. Parçacıkların problem üzerindeki uygun olup olmadığı uygunluk fonksiyonu yardımıyla belirlenmektedir. Parçacıklarda konum bilgisinin yanında hareket için gerekli olan hız bilgileri de yer almaktadır. Parçacıkların konum bilgileri çözüm uzayındaki hız bilgileri ile doğrudan bağlantılıdır. PSO algoritması başlangıç adımında parçacıkların çözüm uzayına rastgele dağılmaktadır. Böylece çözüm uzayında yer alan yiyecek bölgeleri çözüm uzayına dahil edilmektedir [75]. PSO algoritmasının genel adımları gösterilmiştir [2]:

Adım 1. N_x boyutlu S sürüsünü oluştur ve kontrol parametrelerine değer ataması yap.

REPEAT

Adım 2. Her bir parçacık için ($i= 1,2,3,\dots,n_s$);

Bölgesel en iyi noktayı belirle:

Eğer $f(S.x_i) < f(S.y_i)$ ise $S.y_i = S.x_i$

Global en iyi noktayı belirle:

Eğer $f(S.y_i) < f(S.y')$ ise $S.y' = S.y_i$

Adım 3. Her bir parçacık için ($i= 1,2,3,\dots,n_s$);

Hız güncellemesini yap

Pozisyon güncellemesini yap.

UNTIL (Durdurma kriteri sağlanıncaya kadar)

PSO algoritması diğer optimizasyon algoritmaları ile kıyaslandığında düzenlenmesi gereken parametre sayısı daha az olduğundan basit ve anlaşılabilir bir algoritmadır [75].

3.2.3. Tavlama Benzetimi (Simulated Annealing) Algoritması

Tavlama Benzetimi Algoritması katı halde bulunan cisimlerin ısı değişimlerini temel alan meta-sezgisel bir algoritmadır. Katı cisimlerin ısıtılma sürecinde gereken en az miktardaki enerji düzeyine ulaşma işlemi cisimler üzerindeki kristal yapıların istatistiksel bir davranış göstermesine neden olmaktadır. Tavlama Benzetimi Algoritmasında, katı bir cisim ısıtılarak sıcaklığı yükseltilmektedir. Katı cisim erimiş bir hale getirildikten sonra atomların hareketi biraz daha serbest olmaktadır. Ancak sıcaklığın tekrardan düşürülmesi serbest olan bu atom hareketliliğini kısıtlamaktadır. Sıcaklık azaldıkça atom hareketi daha az meydana gelmektedir. Cisimdeki kristal yapılar ise en az miktardaki enerjiye sahip olmaktadır. Kristallerin meydana gelmesi soğuma oranıyla doğrudan bağlantılıdır. Eriyen cisimde soğuma hızı çok daha yüksek olmaktadır. Eğer sıcaklıktaki azalma çok hızlı bir biçimde olursa kristal yapısının elde edilmesi mümkün olmayabilir. Bu yüzden kristal yapısı yerine çok daha yüksek enerji seviyesine sahip olan polikristal yapısı meydana gelebilmektedir. Günümüz problemlerinde soğutma işleminin hızlı olması elde edilecek cisimde pek çok kusur oluşmasına neden olmaktadır. Bu yüzden minimum enerji seviyesine ulaşmak için ısıtılan katı cismin denetlenmesi yapılarak sıcaklığının düşürülmeli ve olabildiğince yavaş bir şekilde sıcaklık düşürülmelidir. Tavlama kavramı yavaş bir şekilde yapılan bu soğutma işlemini ifade etmektedir [77].

Tavlama Benzetimi Algoritmasında soğutma işleminin bilinçli bir biçimde yapılmasını sağlayan mekanizma optimizasyon problemlerinde en iyi çözümün bulunmasına yardımcı olan fonksiyondur. Tavlama Benzetimi Algoritması, çözüm uzayında arama ilk başladığında, en iyi olmayan pek çok çözüm kabul edilerek başlamaktadır. Bu bakımdan bazı bölgesel en iyi çözümleri bulabilmekte ve bunlardan sakınarak global en iyi çözüme ihtiyaç duymaktadır. Bu nedenle çözüm uzayındaki komşuluk bilgilerinden yararlanmaktadır [77].

Tavlama Benzetimi, algoritmasının adımları şu şekildedir:

Adım 1. Bir başlangıç durumunun belirlenmesi, $i \in S$

Adım 2. Bir başlangıç sıcaklığı belirlenmesi: $T > 0$

Adım 3. Sıcaklık değişimini belirten sayacın sıfırlanması: $t = 0$

Adım 4. Durdurma koşulu sağlanınca DUR, durdurma koşulu sağlanmıyorsa sayacı sıfırlama işlemi: $n = 0$ ve DEVAM ET.

Adım 5. i noktasına ait bir komşu olan j 'nin rastgele olarak üretilmesi: $\Delta = f(j) - f(i)$

Eğer $\Delta < 0$ ise $i = j$, Eğer $\Delta \geq 0$ ise ve $U(0.1) < \exp(-\Delta/T)$ ise $i = j$.

Adım 6. $n = n + 1$ Eğer $n < m$ ise *Adım 5'e GİT*, Eğer $n \geq m$ ise $t = t + 1$, $T = T(t)$ ve *Adım 2'ye GİT*.

Tavlama Benzetimi Algoritması tamamen rastgele olan bir başlangıç kümesi ile başlamaktadır. Döngüsel bir yapıyla adım adım ilerlemektedir. Her adımda yerel komşuluk bilgisi yardımıyla bir sonraki adım için gerekli olan çözüm adayını üretmektedir. Sistemde enerji seviyesini ifade eden kavram amaç fonksiyonudur. Bu fonksiyon sayesinde en iyi çözümü ararken hep mevcuttan daha iyi çözümler aranmaktadır. Ancak sistemde sıcaklık artışına neden olan ve algoritma bakımından ise amaç fonksiyonun aradığı en iyi çözüm olmayan çözüm adayları da olası çözüm olarak sisteme dahil edilmektedir [77]. Mevcut olan çözüm ile yeni elde edilecek en iyi çözüm arasındaki farkı ΔW olarak ve sistemin enerji seviyesi W olarak tanımlandığında iyi olmayan bir çözümün sisteme dahil edilme ihtimali şu şekilde tanımlanmıştır:

$$P(\Delta W, T) = e^{-\frac{\Delta W}{T}} = e^{-\frac{W(n+1) - W(n)}{T}} \quad (3.1)$$

W_n , mevcut durumdaki çözüm adayının enerji seviyesini ve bir başka ifadeyle amaç fonksiyonunun olan uzaklığını ifade etmektedir. $W(n+1)$ ise mevcut çözüm adayının yeni

komşuluk bilgisinden doğan enerji seviyesini belirtmektedir. T ise bölgesel komşuluk ile ulaşılan yeni çözümün enerji seviyesini belirtmektedir. T ile gösterilen sıcaklık değeri arttığında, daha iyi olmayan yeni bir olası çözümün elde edilme ihtimali düşmektedir. Bu işlem adımının kaç kez tekrarlanacağı probleme göre değişmektedir. Ayrıca amaç fonksiyonun elde ettiği değere göre de işlem sonlandırılabilir [77].

3.2.4. Yapay Arı Kolonisi (Artificial Bee Colony) Algoritması

Yapay Arı Kolonisi Algoritması, arıların doğal yaşamından esinlenerek oluşturulmuş optimizasyon problemlerinde kullanılan metasezgisel bir algoritmadır. Arılar doğada koloni halinde yaşamaktadırlar. Doğal bir arı kolonisi tamamen modüler bir biçimde görev dağılımı yapılmış bir şekilde çalışmaktadırlar. Hangi işi hangi arı ya da arılar tarafından yürütüleceği açık ve nettir. İşleri modüler bir şekilde paylaşabilme ve kendiliğinden organize olma işlemi sürü zekasının temel özelliklerindedir. Arıların yiyecek arama için gerçekleştirdikleri yiyecek kaynakları, görevli işçi arılar ve görevsiz işçi arılar olmak üzere üç temel unsur bulunmaktadır. Yiyecek kaynakları, arıların bal yapabilmek için ulaştıkları polen, nektar gibi besin kaynaklarıdır. Yiyecek kaynağı uzaklık bilgisi, besin çeşidi, besin değeri, besinlerin konsantrasyonu gibi birçok nitelik ile incelense de temel de zengin olması gibi tek bir unsura indirgenmektedir. Görevli işçi arılar ise önceden tespit edilen yiyecek kaynaklarının yuvaya getirilmesi ile görevlidirler. Yuvaya getirilen besinlerin kaynağı ile ilgili bilgiler yine görevli arılar tarafından birbirlerine aktarılmaktadır. Görevsiz arılar ise nektar toplayabilecek olan kaynakları arama işlemi ile ilgilenmektedirler [2]. Arıların arasındaki kolektif bir bilgi paylaşımı tüm işlemlerin en temel yapısıdır. Bu bilgi paylaşımı arılarda dans alanları ve dans çeşitleri ile mümkün olmaktadır. Bir arı dans ettiğinde besin kaynağı ve besinin değeri ile ilgili bilgi vermekte ve diğer arılar ise antenlerini kullanarak bu bilgileri almaktadırlar [2].

Arı kolonilerinin yuvalarında biriktirdikleri besin kaynaklarını ve arıların kendi aralarındaki etkileşim Arı kolonisi algoritmasının en temel unsurlarındandır. Arılar yiyecek aramak için yuvalarından ilk çıktıklarında rastgele bir arama işlemi ile başlamaktadırlar. Herhangi bir yiyecek kaynağı bulunduğunda bunun diğer arılar ile paylaşılması gerekmektedir. Ayrıca bir yiyecek kaynağında yiyeceklerin azalması da sonraki gelecek arılar için önemli bir bilgidir. Bu bilgi ile arılar yeni yiyecek kaynağını arama işlemine başlayacaktır. Ayrıca yiyecek kaynağındaki bilgilerde polen, nektar, su

kaynaklarına ilişkin bilgilerin de iletilmesi gerekmektedir. Arılar yiyecek kaynakları gittiklerinde kaynağının değerine, yuvaya olan uzaklık bilgisine, yiyeceğin çeşidine, nektar ve polen gibi kimyasalların elde edilebilme kolaylığına ilişkin bilgilerin tamamını zenginlik olarak tek bir kriter ile belirterek bilgi akışını sağlamaktadır. Görevli belli olan arılar ise önceden tespit edilmiş olan yiyecek kaynaklarından elde edilen yiyeceğin yuvaya getirilmesi ile görevlidir [78]. Ayrıca yiyecek kaynağının lokasyon ve kalite bakımından da bilgilendirmesini yapmakla yükümlüdür. Görevi belli olmayan arılar yem elde edilebilecek kaynakların tespit edilmesinden sorumludurlar. Rastgele yiyecek kaynağı arayan arılar ile yuvada bekleyen arıları gözlemleyerek kaşif arılardan gelen bilgi ile yeni kaynaklara sevk edilen gözcü arılar bu kategoride bulunmaktadır [79]. Arılar arasındaki bilgi paylaşımları arıların dans etmesi ile olmaktadır. Yiyecek kaynaklarından dönen arılar yuvadaki arılara yiyecek kaynağının lokasyon bilgisini iletilmesi gerekmektedir. Enerji seviyeleri ve güneş ile olan açıdan yararlanarak yükseğe doğru uçan arılar enerji seviyesi hakkında yuvadaki arılara bilgi vermektedir. Yapay Arı Kolonisi Algoritmasında görevli arıların sayıları ve toplam besin kaynağı belirlenmektedir. Bu algoritma yiyecek kaynaklarının lokasyon bilgilerini öğrenerek olası çözümlere ulaşmakta ve nektar miktarı ile de çözümün kalitesinin değerlendirmektedir. En çok nektara sahip olan kaynağın çözüm uzayında yer alan en iyi çözüm olduğu bulunmaktadır [78]. Bu nokta minimizasyon problemi için en küçük değeri, maksimizasyon problemi için en yüksek noktayı ifade etmektedir. Yapay Arı Kolonisi Algoritmasının temel adımları şu şekildedir [2]:

Adım 1. Başlangıç yiyecek kaynaklarının üretilmesi

REPEAT

Adım 2. İşçi arıların yiyecek kaynağı bölgelerine gönderilmesi ve nektar miktarlarının belirlenmesi

Adım 3. İşçi arılardan gelen bilgilere göre olasılık değerlerinin hesaplanması

Adım 4. Gözcü arıların belirlenen olasılık değerlerine göre yiyecek kaynak bölgesini seçmesi

Adım 5. Ayrılacak kaynakların belirlenmesi ve kaşif arının üretilmesi

UNTIL (Çevrim Sayı)

4. GENETİK ALGORİTMA VE EPİGENETİK ALGORİTMA

Genetik Algoritma, John Holland tarafından ortaya atılan ve canlıların evrimsel süreçlerin modellenmesi esasına dayanan bir algoritmadır [80]. Canlıların evrimsel süreçleri, iyi olan bireylerin kendi yaşamını koruması ve kötü olan bireylerin yok olmasıdır. Genetik Algoritma, rastlantısal arama teknikleri kullanılarak çözüm üretmeye çalışan sezgisel bir algoritmadır. Parametre kodlama ilkesine dayanmaktadır. Holland'ın doktora öğrencisi olan Goldberg Genetik Algoritmayı teoriden pratiğe geçiş yapmasını sağlayarak kullanım alanının yaygınlaşmasına öncü olmuştur [81]. Genetik Algoritma, çok sayıda birey arasında arama yapmaktadır. Bu yüzden bir kontrol mekanizması bulunmaktadır. Bu kontrol mekanizması uygunluk fonksiyonu ile yapılmaktadır [82]. Genetik Algoritma doğal seleksiyon ve diğer genetik kurallara dayanan bir arama ve optimizasyon algoritmasıdır [83]. Doğal yaşamda var olan en iyi bireyin hayatta kalıp diğerlerinin yok olması esasına dayanmaktadır. En iyi bireyin belirlenmesi uygunluk fonksiyonu aracılığı ile tespit edilir. Ayrıca çaprazlama ve mutasyon gibi operatörler aracılığıyla çözüm çeşitliliği sağlanmaktadır. Genetik Algoritmanın en uygun çözüm ürettiği problemler klasik yöntemler ile çözümü mümkün olmayan problemlerdir. Genetik Algoritma, türev integral işlemlerine ihtiyaç duymaması, problemin içeriği ile ilgilenmemesi, problem yerine parametre kodları ile ilgilenmesi, sayısal yöntemlerden farklı olarak birden çok aday çözüm ile işlemlere başlaması, nondeterministik ve stokastik olması gibi özelliklerinden dolayı tercih sebebi olabilmektedir [84].

Genetik Algoritma çözülmesi zor olan ya da imkânsız olan karmaşık yapıdaki problemlerde sıkça kullanılmaktadır. Yapay zekâ çalışmaları ve otomatik programlama çalışmalarında, makine öğrenmesi ve robot hareket sisteminde, ekonomi modellerinde, sosyal sistem analizinde, doğa biliminde, bilgisayar işlemci tasarımı gibi kullanım alanları bulunmaktadır [16]. Genetik Algoritma, hesaplama gücü ve depolama yetenekleri kaynak paylaşımlarının yapıldığı ve iş paylaşımlarının paralel olarak dağıtılarak çözümlendiği Grid Computing gibi güncel problemlerde çözüm üretmek için kullanılmaktadır [85]. Genetik Algoritma kullanılarak Storm Akış Uygulamaları (Storm Streaming) gibi veri akışı sistemlerinin performansı artırılmıştır [86].

4.1. GENETİK ALGORİTMA TEMEL KAVRAMLAR

4.1.1. Parametre Kodlama

Parametre kodlama, bir probleme ait verilerin bir algoritmaya uygulanabilmesi için bilgisayar sistemine dönüştürülmesidir. Parametre kodlama işleminin Genetik Algoritmanın uygulanması sırasında çözüm uzayını en iyi tanımlayacak şekilde yapılması gerekmektedir. Kodlama ile programlayıcı kromozomların nasıl temsil edildiğini, genlerin hangi bileşenlerden oluşacağını belirler. Çözümün esnek, dinamik, hızlı sonuç verme gibi kritik özellikleri kodlamanın düzgün yapılmasına bağlıdır. Genetik Algoritmanın diğer arama tekniklerinden ayrıştığı önemli bir nokta parametrelerin kullanılması yerine parametreleri temsil eden dizilerin kullanılmasıdır. Parametre kodlama işleminde genler ayrı ayrı biçimde simgelenmektedir. Bu simgeleme işleminde bit, ağaç, sayı, liste gibi nesnelere kullanılarak gerçekleştirilir. Parametre kodlama aslında problemin çözümüne odaklanmaktadır. Parametre kodlama şemaları bir kodlamadan başka bir kodlamaya büyük ölçüde farklılık gösterebilmektedir [87]. Kodlama ile birlikte çözüm için gerekli olan değişkenlerin temsil edilmesi sağlanır.

4.1.1.1. İkili (Binary) Kodlama

Tüm kromozomlar 0 ve 1 bitlerinden oluşmaktadır [88]. Bu bitler kendi içlerinde anlamlı ifadeler olmak durumundadır ve tüm genetik süreç boyunca çözüm için anlamlı bir karşılığı olmaktadır. Genin uzunluğu kodlama yapan tarafından belirlenmektedir. Gen uzunluğu ile elde edilen veri dizisi birleştirilerek bir araya getirilir. Şekil 4.1 ile ikili kodlama yapısı gösterilmiştir.



Şekil 4.1. İkili kodlama yapısı.

4.1.1.2. Permütasyon Kodlama

İkili kodlamadan farklı olarak kromozomlar tüm rakamların kullanılabilirdiği bitlerle ifade edilmektedir. Rakamlar kromozom içerisinde ve gen içerisinde sıralı ve anlamsal bütünlüğü bozmayacak şekilde bulunmalıdır. Çaprazlama ve mutasyon işlemlerinde çözüm için anlam ifade etmeyen genler bulunabileceğinden ötürü düzenlemeleri gerekli

kılmaktadır. Permütasyon kodlama, bir kromozomda rakamların peş peşe sıralandığı ve gerçek değerlerin bulunduğu dizilerdir [88]. Şekil 4.2 ile permütasyon kodlama yapısı gösterilmiştir.

Kromozom 1

1456237869

Kromozom 2

4523617928

Şekil 4.2. Permütasyon kodlama yapısı.

4.1.1.3. Değer Kodlama

Değer kodlama, karmaşık sayıların (complex numbers) bulunduğu problemlerde kullanılmaktadır. Şekil 4.3 ile değer kodlama yapısı gösterilmiştir.

Kromozom 1

4.3 8.8 1.2 3.6

Kromozom 2

(sol), (geri), (sağ), (ileri)

Şekil 4.3. Değer kodlama yapısı.

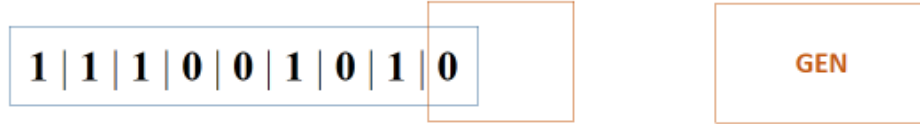
Değer kodlamada, her bir kromozom rakamlar, reel sayılar ya da probleme özgü özel karakterlerin yer aldığı değerler dizilerinden oluşmaktadır. Özelleştirilmiş bazı problem türleri için bu kodlama biçimi avantajlı olabilmektedir. Ancak bu durum probleme özgü olarak genetik operatörlerin tanımlanmasını zorunlu kılmaktadır [87].

4.1.1.4. Ağaç Kodlama

Ağaç kodlama, popülasyonda bulunan her bir kromozom bir nesnenin ağacı olacak şekilde kodlama yapılmasıdır. Ağaç yapısı düğüm ve dallara ayrılarak nesnelere arası ilişkileri ifade etmektedir. Ağaç kodlama Genetik Algoritmanın program ifadelerinin dönüştürmek amacıyla kullanılmaktadır. Her kromozom programlama dilinin komutları ya da metotları gibi öğelerden oluşur. Ağaç kodlamada kromozom objeler ve objeler arası işlemleri tanımlamakta kullanılır ve program geliştirme sürecine uygundur. Örneğin yapay zeka alanından kullanılan programlama dillerinden olan LISP ve PROLOG ağaç yapısının kullanılmaktadır [87].

4.1.2. Gen

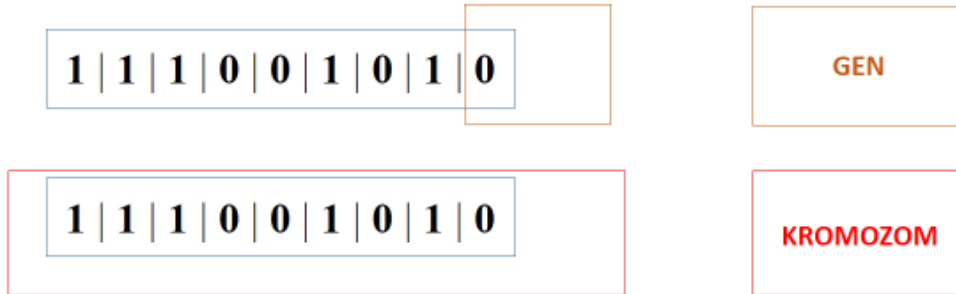
Genetik Algoritmada kendi başına bilgi taşıyan en küçük birimdir. Genler bir araya gelmesiyle kromozomlar meydana gelir. Gen yapısı tamamen programcının tanımlamasına bağlıdır. Genler problemin çözümü için doğru yapıda tanımlanmak durumundadır. Onluk tabandaki sayılardan oluştuğu gibi onaltılık tabandaki sayı sistemini de içerebilir. Şekil 4.4 ile gen yapısı gösterilmiştir.



Şekil 4.4. Gen yapısı.

4.1.3. Kromozom

Birden çok genin bir araya gelmesiyle oluşan genetik yapılarıdır. Problemin çözümüne ilişkin bilgiler barındırır. Tüm kromozomlar problem için olası çözüm kümeleridir. Bu bakımdan kromozomlar içeriği oldukça zengin oluşturulmalı ve iterasyonlu çözümlerde kontrol edilmesi gerekir. Şekil 4.5 ile kromozom yapısı gösterilmiştir.



Şekil 4.5. Kromozom yapısı.

Kromozom, problemlerde birden fazla sayıda olan bir küme biçimindedir ve olası çözümleri temsil etmektedir. Kromozomların oluşturduğu bu küme içine nüfus, nesil gibi terimler kullanılmaktadır. Nesil içindeki anlamlandırılan her bir elaman gen adını almaktadır.

4.1.4. Popülasyon

Popülasyon, kromozomların bir araya gelmesiyle oluşan kümelerdir. Popülasyondaki kromozom sayısı sabit olmakla beraber probleme özgü olarak programlayıcı tarafından belirlenir. Genetik operatörler işleme alındığında popülasyondaki kromozomlar yok

olurken yerlerine yeni oluşan kromozomların eklenmesiyle birlikte popülasyon sayısı sabitlenir.

4.2. GENETİK OPERATÖRLER

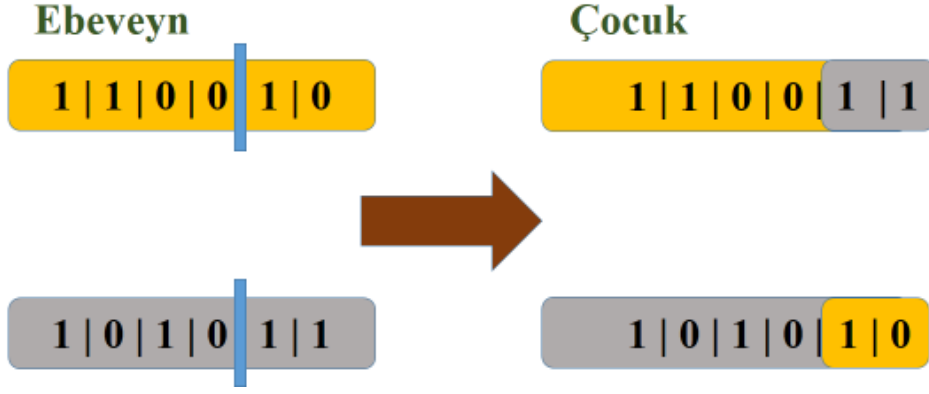
Genetik Algoritma uygulanırken bir süre sonra popülasyonlarda çeşitliliğin sağlanamadığı gözlenmektedir. Bu durumda çözüme gitmek için genetik operatörler kullanılmaktadır. Genetik operatörler ile çeşitlilik sağlanarak en iyi çözümü arama işlemi sürdürülmektedir.

4.2.1. Çaprazlama

Çaprazlama, iki adet kromozomun bir araya gelerek karşılıklı olarak gen bloklarını yer değiştirmesi işlemidir. Çaprazlama işlemi yapılmadan önce çaprazlamanın gerçekleşme olasılığı tanımlanmalıdır. Bu oran literatür incelendiğinde % 50 ile % 95 arasında değiştiği gözlenmiştir [89]. Çaprazlamada bir diğer kritik nokta ise çaprazlamanın türünün belirlenmesidir. Tek noktalı çaprazlama, çok noktalı çaprazlama, pozisyona dayalı çaprazlama, kısmi zamanlı çaprazlama gibi çaprazlama türlerinden birisi seçilir. Çaprazlanacak kromozomların seçilme kriteri de optimizasyon açısından oldukça önemlidir. Çaprazlama operatöründe olasılık çaprazlamanın gerçekleşme sıklığını düzenler. Bir popülasyon için P_c çaprazlama olasılığı olsun her popülasyonda $P_c \times X_1 \times \dots \times X_N$ adet kromozoma çaprazlama işlemi uygulanır. Olasılığın yüksek olması popülasyonun değişkenliğini çok hızlı olarak değiştirirken olasılığın düşük olması arama işleminin yavaş yapılmasına neden olur. Bu bakımından problemin yapısına bağlı olarak bu oran optimum olacak biçimde belirlenmelidir.

4.2.1.1. Tek Noktalı Çaprazlama

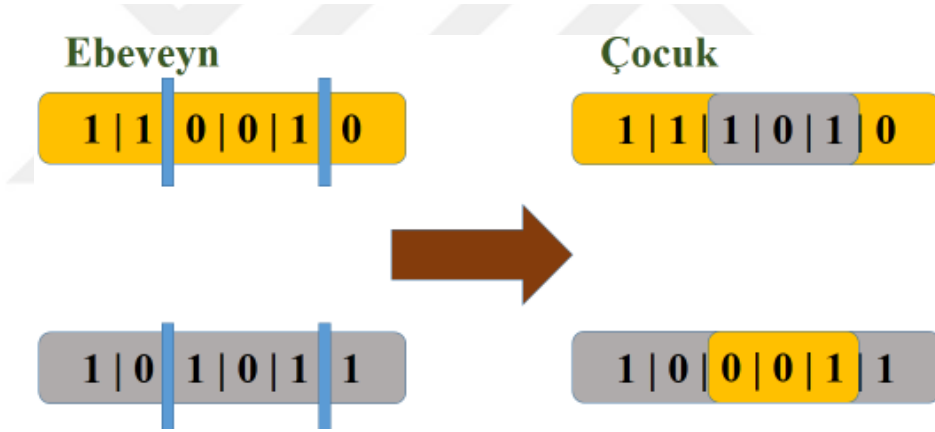
Tek noktalı çaprazlama en çok kullanılan çaprazlama türlerinden biridir. Kromozom içindeki genler bir veri dizisi içinde temsil edildiklerinde belirlenen bir nokta ile ayrılır ve uygun yer seçildiğinde yeni kromozomlar meydana gelir. Tek noktalı çaprazlama ile kromozomun baş kısmı ile son kısmı yer değiştirmiş olur. Baş kısmı veya son kısmı iyi özelliklere sahip değilse kaliteli bireyler oluşamaz [90]. Şekil 4.6 ile tek noktalı çaprazlamanın nasıl gerçekleştiği gösterilmiştir.



Şekil 4.6. Tek noktalı çaprazlama.

4.2.1.2. İki Noktalı Çaprazlama

İki noktalı çaprazlama tekniğinde kromozomlardan iki tane seçilir. Seçilen bu iki kromozom ayrı ayrı çaprazlanmak üzere seçilir. Her bir kromozom iki tane nokta ile ayrılır. Ayrılan bu genler uygun çaprazlama oranını sağlıyorsa genlerin yer değiştirmesi sağlanır [88]. Şekil 4.7 ile iki noktalı çaprazlamanın nasıl gerçekleştiği gösterilmiştir.

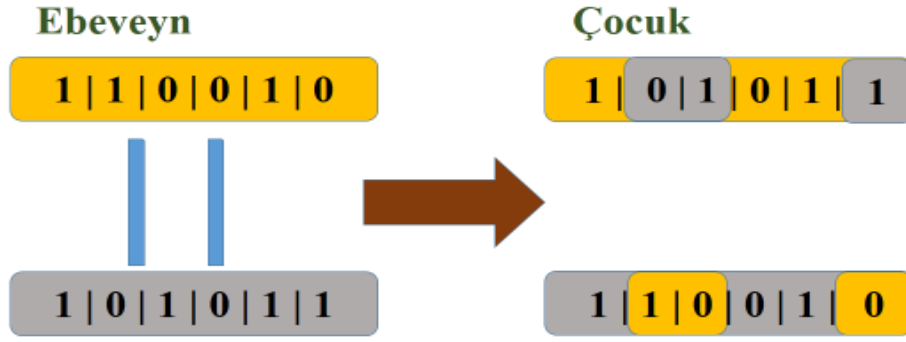


Şekil 4.7. İki noktalı çaprazlama.

Ebeveyn kromozomlar iki nokta yardımıyla üç parçaya bölünmüştür. Bu üç parçadan uygun oran sağlandığında gen bölgeleri karşılıklı olarak yer değiştirmiştir.

4.2.1.3. Çok Noktalı Çaprazlama

Çok noktalı çaprazlama popülasyon içerisinde seçilen iki kromozom N tane noktaya ayrılır ve bu n tane nokta için her bir durumda belirlenen çaprazlama oranı için mümkünse bu iki kromozom arasında gen takası sağlanır [88]. Şekil 4.8 ile çok noktalı çaprazlamanın nasıl gerçekleştiği gösterilmiştir.

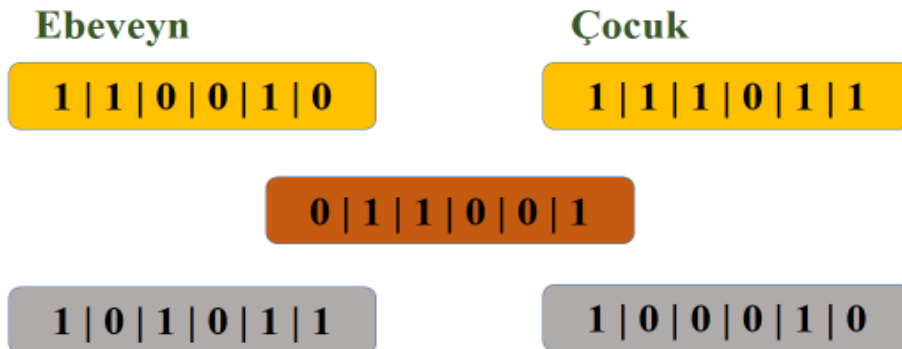


Şekil 4.8. Çok noktalı çaprazlama.

Ebeveyn kromozomların tüm gen bölgesi n tane noktaya bölünür. Ardından çaprazlama oranı şartı sağlandığından bu n tane noktanın her biri karşılıklı olarak yer değiştirmektedir. Bu tez kapsamında literatür çaprazlama türleri incelendiğinde problem türü için en uygun çaprazlama operatörü çok noktalı çaprazlama olduğu tespit edilmiş ve uygulanmıştır.

4.2.1.4. Uniform Çaprazlama

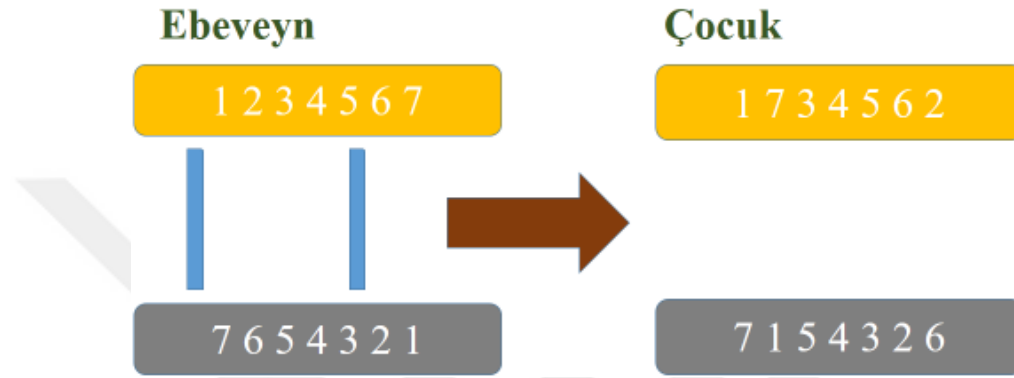
Uniform çaprazlama için kromozomların parçalanması gerekmez. Popülasyonda bulunan iki kromozom üzerinde kromozomlar ile aynı uzunluğa sahip bir maske oluşturulur. Bu maske yardımıyla yer değiştirmesi gereken genler belirlenir. Kodlama türüne göre farklılık göstermekle birlikte maske üzerinde bulunan bit 1 sonuçtaki gen birinci ebeveynden kopyalanarak oluşturulur, maske üzerindeki bit 0 ise ikinci ebeveynden kopyalanarak oluşturulur. Bu işlem her bir ana kromozom çifti için isteğe bağlı olarak yeni bir maske oluşturulur [90]. Şekil 4.9 ile uniform çaprazlamanın nasıl gerçekleştiği gösterilmiştir.



Şekil 4.9. Uniform çaprazlama.

4.2.1.5. Sıraya Dayalı Çaprazlama

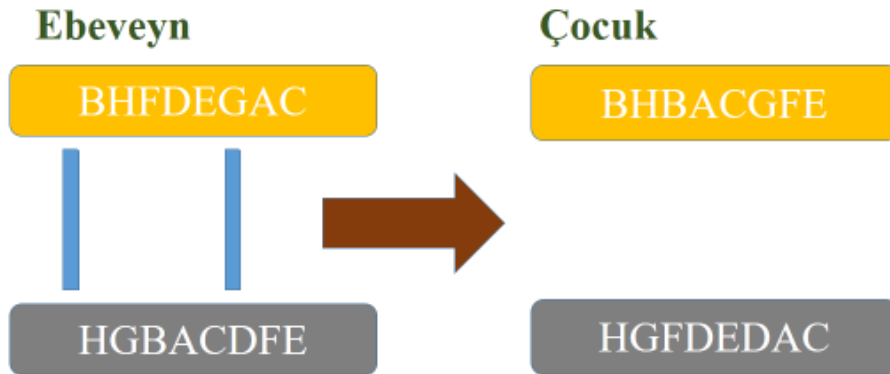
Sıraya Dayalı Çaprazlama (Order Based Crossover) türünde iki kromozom seçilir. Birinci kromozomda seçilen genlere karşılık gelen gen bölgesi yerlerini korur. İkinci kromozomun seçilen genlere ait bölgesi birinci kromozomun aynı bölgedeki genlerinin arkasına gelmektedir. Geriye kalan boş gen bölgesi ikinci kromozomdan aktarılan yeni genlerle ilk kromozomun boşta kalan genleri sıra ile doldurulmaktadır [91]. Şekil 4.10 ile sıraya dayalı çaprazlamanın nasıl gerçekleştiği gösterilmiştir.



Şekil 4.10. Sıraya dayalı çaprazlama.

4.2.1.6. Kısmi Zamanlı Çaprazlama

Goldberg tarafından ortaya atılan bu çaprazlama biçimi gezgin satıcı probleminde uygulanmıştır [81]. Kısmi zamanlı çaprazlama biçimi iki farklı kromozom üzerinde rastgele belirlenen genler arasındaki değişimdir. Bu değişimde iki kromozom arasında ortaklaşa olan bir bölgede gen grupları aynı olmalıdır [91]. Şekil 4.11 ile kısmi zamanlı çaprazlamanın nasıl gerçekleştiği gösterilmiştir.



Şekil 4.11. Kısmi zamanlı çaprazlama.

Ebeveyn kromozomlarından birinci kromozomun 3, 4 ve 5. gen bilgileri ile olana F, D, E

genleri ile bu gene karşılık gelen ikinci kromozomun B, A, C genleri yer değiştirmiştir. Benzer biçimde bu sefer iki nokta seçilen 7. ve 8. gen bilgilerine karşılık gelen birinci kromozomdaki A, C genleri ile ikinci kromozomda 7. ve 8. gen olan F, E genleri karşılıklı olarak yer değiştirmiştir.

4.2.2. Mutasyon

Genetik Algoritmanın uygulandığı problemlerde çaprazlama sonrası ilerleyen nesillerde gen çeşitliliğinde azalma olacaktır. Kromozom yapıları birbirine çok benzeyeceği için çözüm uzayında çözüme giden alan daralacak veya kaybolacaktır. Böyle bir durumdan kurtulmanın bir başka ifadeyle gen, kromozom, popülasyon çeşitliliğinin sağlanması için mutasyon işlemi uygulanmaktadır. Mutasyon işleminde aynı kromozomdaki iki ya da daha fazla gen bilgisi aktarımı gerçekleşmektedir. Mutasyon operatörü ile çeşitliliği sağlanacağı gibi daha iyi özelliklere sahip bireylerin ortaya çıkma ihtimali de tekrar mümkün olmaktadır. Mutasyon operatörü kromozomda değişiklik meydana getirdiği için ve çözümsüzlük esnasında devreye gireceği için mutasyon oranı düşük tutulmalıdır [88]. Mutasyon oranı tipik olarak % 0.1 ila % 0.5 arasında seçilmektedir [88]. Literatürde kullanılan bazı mutasyon çeşitlerinden bazıları şöyle sıralanmıştır: Ters Çevirme, Ekleme, Yer Değişikliği, Karşılıklı Değişim gibi sıralanmaktadır [92].

4.2.2.1. Yer Değiştirme Mutasyonu

Kromozomda bulunan rastgele gen, kromozom uzunluğunu geçmeyecek biçimde rastgele seçilen iki adet pozisyona yerleştirilerek yeni kromozom elde edilmesidir [93]. Şekil 4.12 ile yer değiştirme çevirme mutasyonunun nasıl gerçekleştiği gösterilmiştir.



Şekil 4.12. Yer değiştirme mutasyonu.

4.2.2.2. Ekleme Mutasyonu

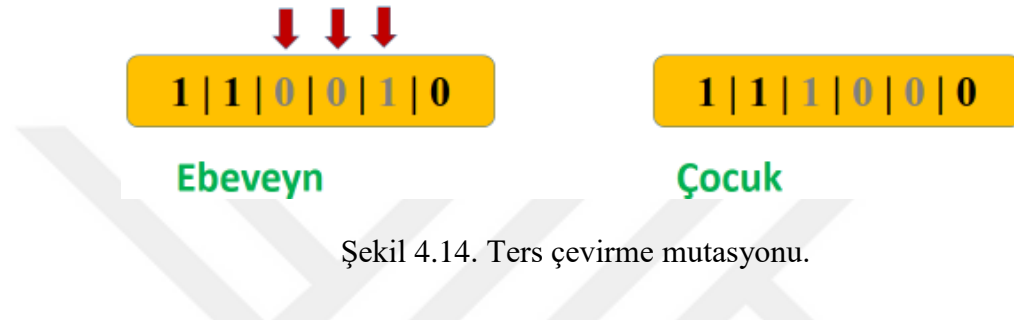
Kromozomda bulunan rastgele bir genin, bir başka genin üzerine taşınması ile elde edilir [94]. Şekil 4.13 ile ekleme mutasyonu gösterilmiştir.



Şekil 4.13. Ekleme mutasyonu.

4.2.2.3. Ters Çevirme Mutasyonu

Kromozomda bulunan rastgele iki gen pozisyonu seçilip bu elemanların tersine çevrilmesi ile elde edilir [92]. Şekil 4.14 ile ters çevirme mutasyonu gösterilmiştir.



Şekil 4.14. Ters çevirme mutasyonu.

4.2.2.4. Karşılıklı Yer Değiştirme Mutasyonu

Kromozomda bulunan rastgele iki genin pozisyon bilgileri vasıtasıyla bu iki gen üzerinde değişiklik yapılmasıdır [95]. Şekil 4.15 ile karşılıklı yer değiştirme mutasyonu gösterilmiştir.



Şekil 4.15. Karşılıklı yer değiştirme mutasyonu.

4.2.2.5. Komşu Mutasyon

Komşu mutasyon, kromozomda bulunan ve birbiri ardı sıra bulunan herhangi iki genin birbirleri ile karşılıklı yer değiştirerek genetik farklılaşma sağlamalıdır. Şekil 4.16 komşu mutasyon gösterilmiştir.



Şekil 4.16. Komşu mutasyonu.

Genetik Algoritma yalnızca seçim operatörü ve çaprazlama operatörü ile oluşsaydı yerel optimuma ulaşması daha çabuk olurdu. Popülasyonda çeşitlilik oluşturabilmek çaprazlama operatörü ile kaybolabilen iyi nitelikteki özelliklerin yeniden kazanmak için mutasyon operatörüne ihtiyaç duyulmaktadır. Mutasyon operatörü ile birlikte en iyiye ulaşmada bireyler belirli bir olasılık ile mutasyona uğratılmaktadır.

4.3. SEÇİM OPERATÖRÜ

Evrin teorisinde üreme sonrasında yeni nesiller oluşturulurken bir kısım bireyler yeni nesillere aktarılır ve bir kısım bireyler yok olur. Hangi bireylerin hayatta kalacağı hangi bireylerin yeni nesillere aktarılacağı seçim mekanizmaları ile belirlenmektedir. Bu seçim yöntemleri bazı özellikler değerlendirilerek yapılır. Bu değerlendirmelerin farklılaşması sonucu farklı seçim mekanizmaları kullanılmaktadır. Kullanılan seçim mekanizmaları Rulet Tekerleği (Çemberi), Orantılı Seçim Mekanizmaları, Sıralı Seçim Mekanizmaları, Turnuva Seçim Mekanizmaları, Denge Durum Seçim Mekanizmaları gibi sıralanmaktadır [88].

4.3.1. Turnuva Seçim Mekanizması

Turnuva Seçim Mekanizmasında, popülasyondan bir grup kromozom seçilir. Seçilen bu grup içerisinde uygunluk değerlerine göre turnuvaya dahil edilir. Turnuva sonucunda bir kazanan olana kadar kromozomlar uygunluk değerlerine göre rekabet eder. Sonuçta bir tane çaprazlamaya dahil edilecek bir kromozom seçilir. Popülasyon sayısı tamamlana kadar bu işlem sürecektir.

4.3.2. Sıralı Seçim Mekanizması

Sıralı Seçim Mekanizmasında, popülasyonda bulunan kromozomlar uygunluk değerleri nispetince en yüksek değerden en düşük değerli olana doğru sıralanır. En iyi kromozomdan başlayarak birer azalacak biçimde kromozomların kopya sayıları

belirlenir. Ardından kopya sayılarına göre yeni popülasyon oluşturulur.

4.3.3. Denge Durum Mekanizması

Denge Durum Seçim Mekanizması Denge Durum Seçim Mekanizmasında doğrusal seçim mekanizması kullanılır ve birkaç tane kromozom genetik operatörlere uğrar. Genetik operatörlere uğrayıp yeni oluşan bu kromozomları popülasyonda bulunan ve uygunluk değeri bakımından kötü olarak nitelendirilen kromozomlarla yer değişikliği yapar.

4.3.4. Rank Seçim Mekanizması

Rank Seçim Mekanizmasında ilk olarak popülasyonun sıralanması yapılır. Ardından her kromozom uygunluk değerini bu sıralamadan yararlanarak kazanır. Popülasyondaki en kötü kromozom 1 ve en iyi kromozom N olacak biçimde sıralanmaktadır.

4.3.5. Rulet Çemberi

Rulet Çemberi seçim mekanizmasında kromozomlar uygunluk değerlerine göre bir çember içine yerleştirilmektedir. Uygunluk değeri daha yüksek olan kromozom çember içerisinde daha büyük bir alana sahip olur [96]. Böylece uygunluk değeri daha yüksek olan bireyin seçilme olasılığı daha fazla olmuş olur. Çizelge 3.1 ile $f(x) = x^2$ amaç fonksiyonuna ait kromozomlar ve uygunluk değerleri gösterilmiştir. Bu popülasyonda 5 tane kromozom bulunmaktadır. Her bir kromozom bir uygunluk değerine sahiptir. Uygunluk değerleri ile uygunluk oranları hesaplanmıştır.

Çizelge 4.1. Kromozomların uygunluk değerleri.

Birey (i)	Kromozom	Uygunluk Değeri	Toplam (% 100)
1	0001	1	1.2
2	0010	4	4.8
3	0100	16	19.5
4	0101	25	30.4
5	0111	36	43.9

Problem çözümü için bu uygunluk değerleri oldukça önemlidir. Daha sonra uygunluk değeri normalizasyon yapılarak 0-100 arasında bir değere dönüştürülmüştür. Bu sayede kromozomların çözüm için uygunluğu ve bir çözümün diğerinden iyi olup olmadığı tespit edilebilecektir. Bu normalizasyon işlemleri 0-1 arasında da yapılabilmektedir.

Başlangıç popülasyonu oluşturulduğunda ilk olarak ceza fonksiyonu yardımıyla ceza değeri, uygunluk fonksiyonu yardımıyla uygunluk değeri oluşturulur. Gelecek nesillerde Rulet Tekerleğindeki paylarına göre kromozomların seçilmesi şu şekilde belirlenmektedir:

$$P_i = \frac{U_i}{\sum_{i=1}^n U_i} \quad (4.1)$$

* P_i : i. Sıradaki kromozomun seçilme olasılığı, U_i : i. sıradaki genin uygunluk değeri, n: kromozomdaki toplam gen sayısı.

4.4. ELİTİZM

Elitizm, popülasyon içerisinde yer alan kromozomların en iyisinin belirlenerek sonraki nesillere hiçbir değişiklik yapılmadan aktarılmasıdır. Genetik Algoritmanın rastgelelik özelliği gereği genetik operatörler uygulandığında daha kötü kromozomların oluşması muhtemel bir olaydır. Yeni nesillerin daha iyi çözüm üretmesi beklenirken daha kötüye gitmesi kabul edilemez bir durumdur. En iyiye doğru giderken yeni popülasyon oluştuğunda en iyi kromozom korunmalıdır.

4.5. TAMİR OPERATÖRÜ

Kısıtlı optimizasyon problemlerinde, popülasyon havuzunda bulunan kromozomlar Genetik Algoritma operatörlerinin çaprazlama ve mutasyon operatörlerine uğramaktadırlar. Çaprazlama ve mutasyon sonrası yeni oluşan kromozomlarda gen bilgileri kaybolmakta ya da olması beklenenden farklı dizilime doğru gitmektedirler. Bu durumda Genetik Algoritma çözüm uzayından sapmaktadır ve uygulamayı da çözümden oldukça uzaklaştırmaktadır [88]. Bu problem her bir neslin tamamlanmasının ardından kromozomların Tamir operatöründen geçirilmesini gerektirmektedir [97]. Bu sebeple bu tezde ek olarak tamir operatörü de kullanılmıştır [98]. Tamir operatörü ile olağan dizilimini kaybetmiş kromozomların eksik olan gen bilgileri yerine koyulmaktadır. Böylece bozulmuş gen bilgileri düzeltmektedir. Bu bakımdan tamir operatörü düzeltme operatörü olarak bilinmektedir [99]. Şekil 4.17 ile çaprazlama sonrası ortaya çıkan gen tekrarı ve gen kaybolması gösterilmiştir.

Ebeveyn 1

12 | AB | 10 | CD

Ebeveyn 2

CD | 10 | AB | 21

Çocuk 1

12 | AB | AB | CD

Çocuk 2

CD | 10 | 10 | 21

Şekil 4.17. Tamir operatörünün gerekliliği.

Bir kromozomda çaprazlama ve mutasyon sonucunda aynı genden birden fazla bulunabilir. Ancak kısıtlı optimizasyon probleminde her bir gen birbirinden bağımsız bir iş birimini, problemin bağımsız bir parçasını temsil ediyorsa bir genin tekrar etmesi problem olmaktadır. Örneğin bir işin tekrar yapılmasını gerektirebilmektedir. Benzer bir şekilde bir genin kaybolması o genle tanımlanan iş biriminin ortadan kalkmasına neden olmaktadır.

4.6. EPIGENETİK ALGORİTMA

“Epi” kelimesi bir ön ek olup “yanı sıra, ek olarak, ilaveten” gibi ifadelerle karşılık gelmektedir. Epigenetik, genetik kod yapısını oluşturan Adenin, Timin, Guanin ve Sitozin baz dizilimlerine etki etmeden bunların yanısıra oluşan farklılaşmaları ifade etmektedir [100]. Epigenetik, klasik genetik kavramının dışında kalan anlamına gelmektedir [101]. Deoksiribo Nükleik Asit (DNA) dizisinde bir değişim meydana gelmeden ancak gen düzeyinde değişimlerin oluşmasına Epigenetik denilmektedir [102]. Epigenetik mekanizmalar ile genlerin ne kadar, nerede, ne zaman çalışacağı belirlenmektedir [103]. Epigenetik, DNA yapısında değişikliğe neden olmazken genlerin çalışmasında yaşam süresi boyunca önemli derecede değişimler meydana getirmektedir [104].

Conrad Waddington, canlı gelişimi sırasında DNA diziliminden başka düzeneklerin bu farklılaşmaya neden olduğunu belirtmiştir. Yakın zamandaki çalışmalar, Epigenetiğin yalnızca gelişim esnasında değil, erişkin yaşamda da gen ifadesinin düzenlenmesinde

rol oynadığını ortaya koymuştur [105] - [107]. Bir hücreye veya bir organizmaya ait DNA dizisinde kodlanan genetik bilginin tamamına ‘genom’ adı verilirken; bir hücre veya bir organizmanın, DNA ve histonları üzerinde gerçekleşen moleküler düzeydeki kimyasal değişimler ile gen ekspresyonunun DNA dizisinden bağımsız olarak düzenlenmesine ‘epigenom’ adı verilmektedir [108].

Epigenetik değişimler dengeli ve kalıcı özelliktedir [106]. Bununla birlikte Epigenetik değişimler geri çevrilebilir nitelik taşımaktadır [107]. Epigenetiğin bu şekilde dinamik özellikte olması koşulların değişim göstermesinden kaynaklanmaktadır [109]. Ancak bu özellik çevresel yaşam koşullarının değiştirilmesi, ilaç kullanımı, terapi uygulanması gibi müdahaleler yardımıyla gen aktivitesinin kontrol edilmesini mümkün kılmaktadır. Bu dinamik özellik sayesinde Epigenetik değişimler yeni tedavi stratejisi belirlenirken çalışma temellerini oluşturmaktadır [109]. Epigenetik, kalıtılabilir özelliktedir. Bu özellik ise çevre koşullarının gen ifadesi üzerinde oluşturduğu etkiyi, bunun o birey üzerindeki oluşturduğu fayda ve zarar belirlenerek sonraki nesillere aktarılmasını mümkün kılmaktadır [110].

Epigenetik, dinamik özelliğe sahip bir mekanizmadır. Bu özellik onu değişen çevre koşullarına göre düzenlenmesini sağlamıştır. İlaç, beslenme, tedavi gibi etkilerle gen ifadesinin kontrol edilmesi mümkün olabilmektedir. Bu durumda çevresel şartların gen ifadesi üzerindeki oluşturduğu avantaj veya dezavantaj sonraki kuşaklara taşınabilmektedir [111], [112].

Epigenetik DNA diziliminde değişiklik yapılmadan gen ekspresyon programları üzerinde uzun vadeli etkileri olan çeşitli süreçleri ifade etmektedir [111].

Waddington Epigenetik kavramını ortaya attığında kavramın merkezinde evrim bulunmaktaydı. Gelişimsel mekanizmaların evrimi ile ilgilenmekteydi. Dolayısıyla epigenetik, sadece embriyoloji değil, evrim teorisini de içine alan bir disiplin olarak yer almaktadır [113].

Genomlar, bir canlıda bulunan kalıtım materyallerinin sahip olduğu tüm genetik bilgilerdir [114].

Epigenom ise Epigenetik mekanizmalarda bulunan bütün oluşumlardır. Gen transkripsiyonunu düzenleyen işlemler doğrudan genom organizasyonunun etkisi altında bulunmaktadır. Epigenom, DNA dizisi tarafından getirilmeyen ek bilgiler içermektedir. Epigenetik talimatları tamamlayan uzamsal ve fonksiyonel kısıtlamalar

tanımlamaktadır [107].

Güncel Epigenetik yalnızca tıp, tarım ve türlerin korunması için değil aynı zamanda kalıtım ve evrimin nasıl oluştuğu konusunda etkileri olduğu için önemli bir kavrama dönüşmüştür. Özellikle, DNA varyasyonlarının hücre ve organizmalarda iletilebileceği epigenetik kalıtım sistemlerinin olduğunu kabul etmek, kalıtım kavramını genişletmektedir. Epigenetik terimi yayın olarak ise, genler ve onların ürünleri arasındaki nedensel etkileşimleri inceleyen süreçlerdir. Çünkü Epigenetik kavramında Epi ve gen kavramı genetiğin genlerini dâhil etmektedir ve genin üstünde bir yapıyı araştırma merkezi olarak ortaya koymaktadır [113].

Epigenetiğin özellikleri belirtilirken kontrol mekanizmaları ve hücrede belirli bir gen aktivitesi spektrumunun kalıtımını içerdiğini bilgisi de eklenmeye başlamıştır. Klasik DNA koduna ek olarak, kalıtsal materyalin bir kısmını oluşturan ek bir bilgi katmanının üst üste binmesini sağlayan bu yapı Epigenetiğin temel işlevlerinden birinin belirginleşmesini sağlamıştır [115].

Epigenetik kalıtım ve Epigenetik bellek terimleri genellikle hücrelerin altta yatan DNA dizisini değiştirmeden alternatif gen ekspresyon programlarını ve hücre fenotipleri hücre bölünmeleri boyunca iletme kabiliyetini belirtmektedir.

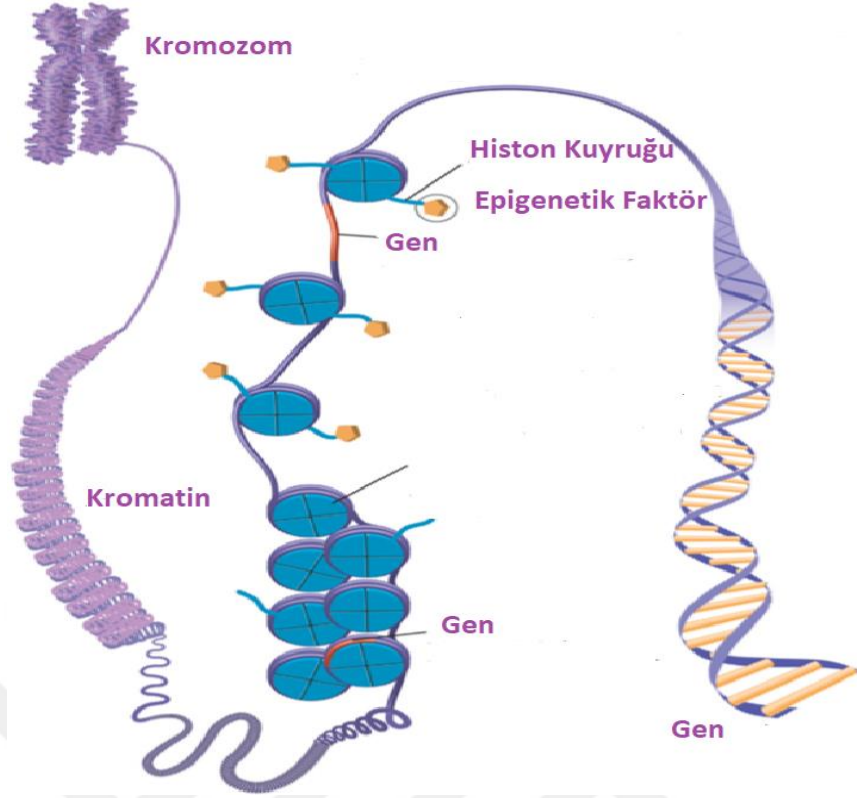
Kalıtsal epigenetik varyasyonlar evrimsel değişikliğinin temelini oluşturmaktadır. Bu, evrimin sadece değişen gen frekansları açısından değerlendirilemeyeceği aynı zamanda epigenetik varyantların sıklığının da dikkate alınması gerektiğini ortaya koymuştur. Epigenetik sistemler hücre aktivite düzenlenmesine katıldığından ve aynı zamanda kalıtım sistemleri olduğundan dolayı edinilmiş, düzenlenmiş ve uyarılmış varyasyonun kalıtımı da mümkündür. Evrim teorisinde bulunan Lamarckian bileşeni de çevrenin bir varyasyon seçici olarak yer aldığını ortaya koymaktadır. Çok hücreli organizmalarda, genetik ve epigenetik varyasyon meydana gelmeli ve bir sonraki kuşağa geçebilmek için mayotik ve embriyogenez yoluyla devam etmelidir [113].

Epigenetik etkilerin yavru bireylere bulaşabilmesi, bilim için de önemli etkilere sahiptir. Çevresel kaynaklı değişikliklerin gelecek nesil üzerindeki etkileri hakkında ilgili veriler mevcuttur [113]. Epigenetiğin hastalıkların önlenmesi ile ilgili olabilecek özellikleri de mevcuttur.

Epigenetik ekoloji için oldukça önemli stratejilere konu olabilmektedir. Organizmalar birbirleriyle ve abiyotik ortamlarıyla etkileşime girer ve bu etkileşimler yoluyla bazıları

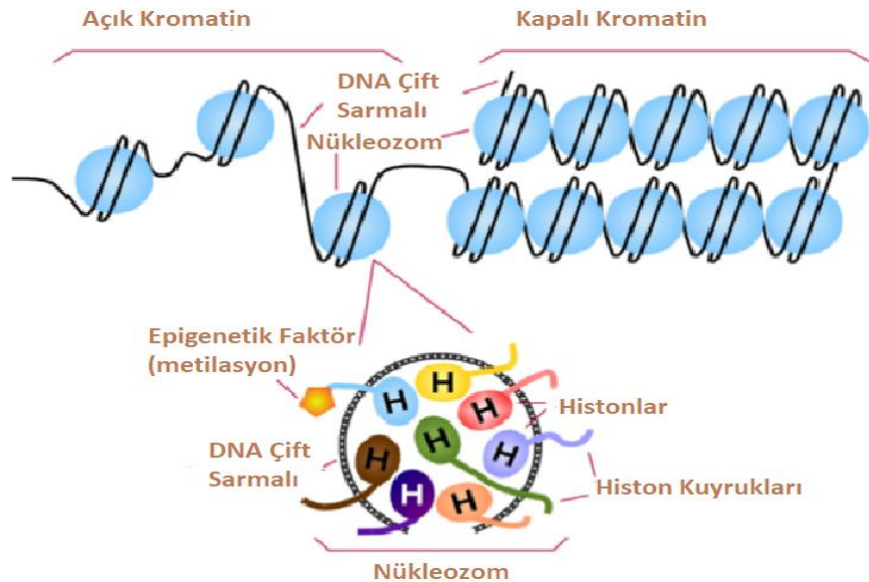
kalıtsal olan epigenetik bilgi edinirler. Doğası gereği, bu epigenetik bilgi bir kez ve tüm bireyler için oluşturulmuş bir bilgi değildir. Birçok kuşak boyunca inşa edilmiş kademeli tarihsel-gelişimsel süreçlerin sonucu olabilmektedir. Bu, bitkileri ve hayvanları doğaya daha iyi, daha ekolojik olarak aklı başında bir geleceğe geri kazandırmak için dondurma tohumlarının, embriyoların veya DNA'nın epigenetik mirasında yansıtılan koşullar da yeniden yapılandırılmadığı sürece çalışmayabileceğini göstermektedir. Toplulukların istikrarı, genellikle, türlerin etkileşen geçmişlerinden kaynaklanan bu çeşitliliğe bağlı olmaktadır [113].

Genetik Algoritma, olasılık belirtilmesi ile rastgele olarak gerçekleştirilen çaprazlama ve mutasyon operatörleri tarafından düzenlenmektedir. Genetik Algoritma son yıllarda farklı özellikler üretilmesi ve yapılan iyileştirmelere sayesinde pek çok hibrit algoritmaya temel olmuştur [116]. Epigenetik algoritma kalıtım, çaprazlama, mutasyon, seleksiyon işlemlerini barındırırken bu işlemlere ek olarak Epiçaprazlama, Epimutasyon eklenir [64]. Çevresel faktörler, ilaç kullanımı ve diğer dış faktörler Epigenetik süreçlerdeki Epigenetik faktörlerin kontrol edilmesini sağlar [116]. Epigenetik tasarımda, bir ceza puanı alan veya ceza almayan her gen için bir Epigenetik Faktör Listesi oluşturulur ve bu genin aktif olup olmadığını tanımlar (aktif gen için 1 ve aktif olmayan gen için 0 atanır). Epigenetik faktörler, tüm popülasyon ve her kromozom boyunca bir liste olarak tutulur. Bu liste, klasik genetik algoritmanın mutasyon ve çaprazlama işlemine katılacak genlerin seçilmesine yardımcı olup algoritmanın adım sayısını kolaylaştıracaktır. Şekil 4.18 ile Epigenetik unsurların DNA üzerindeki görüntüsü verilmiştir.



Şekil 4.18. Epigenetik mekanizma yapısı.

Epigenetik, DNA dizilimi değiştirmeden DNA'nın belirli bir segmentinin Ribo Nükleik Asit (RNA) üzerine kopyalandığı ilk gen ifadenmesinin adımında uzun süreli değişimlerin oluşmasını incelemektedir [119]. Gerçekleşen bu değişimler bazen bir sonraki nesilde bazen çok daha sonra gelecek olan nesillerde görülebilmektedir. Şekil 4.19 ile DNA Kromatin yapısı gösterilmiştir.



Şekil 4.19. Kromatin yapısı.

Bireysel kromozomların yanına eklenmiş biçimde bulunan ek bilgilerin, gelecek nesillere aktarıldığı ortaya konmuştur [64]. Bu ek bilgiler klasik genetik mekanizmada kullanılan rastgelelik üzerine kurulu olan çaprazlama ve mutasyonu tanımlamada yardımcı olmaktadır.

4.7. EPİGENETİK ALGORİTMA İLE İLGİLİ ÇALIŞMALAR

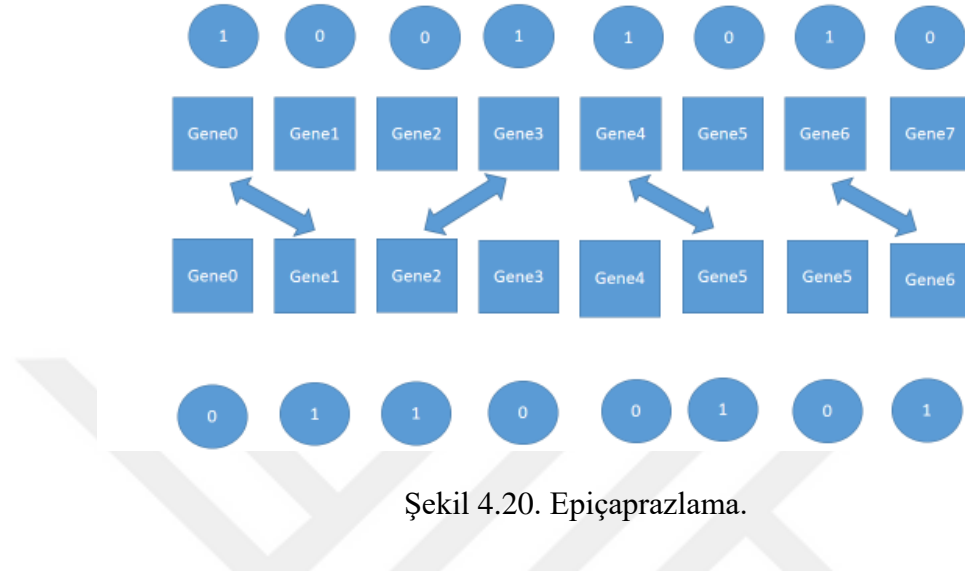
Epigenetik algoritmalar Global System for Mobile Communications (GSM) mobil frekanslarının optimizasyonu ve planlanması alanında uygulanmıştır [64]. Epigenetik, Genetik Algoritmanın rasgelelik üzerine kurulu olan çarpazlama ve mutasyon işlemlerini tanımlamada yardımcı olarak güvenlik amaçlı Saldırı Tespit Sistemi çalışmasında kullanılmıştır [116]. Çalışmada iki türlü saldırı tespit sistemi olduğu, birincisi bilinen imzalarla yapılan saldırıları tespit etmek, ikincisi ise anormal davranışları sezgisel yaklaşımlarla tespit edildiği vurgulanmıştır [116]. Epigenetik Algoritma en iyi çözümü elde etmek için sezgisel algoritmaların hibritleştirilerek optimizasyon yapmaktadır. Genetik Algoritmaya ek olarak getirilen faktörler yeni hibrit algoritmaların doğmasına neden olmuştur. Araştırmalar neticesinde gen dizisini değiştirmeden gen ekspresyonunu inceleyen epigenetik ortaya atılmasına neden olmuştur. Daha iyi bir çözüme yaklaşmak adına ve yineleme sayısını azaltmaya yardımcı olan klasik Genetik Algoritmaya Epimutasyon ve Epiçaprazlama eklenmiştir. Epigenetik, kromozom yapısının aktive edilmesine veya aktive edilmeyen genlerin düzenlenmesini kontrol eden bir mekanizmadır [116]. Epigenetik Algoritma çok boyutlu sırt çantası (Knapsack Problem) problemine çözüm geliştirilmiştir [118].

4.8. EPİGENETİK OPERATÖRLER

4.8.1. Epiçaprazlama Operatörü

Epiçaprazlama işlemi Genetik Algoritma bünyesinde bulunan Çaprazlama işlemine benzer nitelikte olsa da temel farklılıklar göstermektedir. Genetik Algoritma için gerekli olan Çaprazlama işleminde rastgelelik esas unsurdur. İki gen belirlenen çaprazlama türü ile belirlenmiş oranda gerçekleştirilmektedir. Epiçaprazlama için hangi genlerin çaprazlanacağı belli ölçülerde yönetilebilmektedir. Bu yönetilebilmenin yapılabilmesi için Epigenetik Faktör Listesi belirlenmiş olmalıdır. Epigenetik Faktör Listesi sayesinde hangi genlerin sorunlu olduğu kromozomlar boyunca bilinmektedir. Çaprazlanacak olan

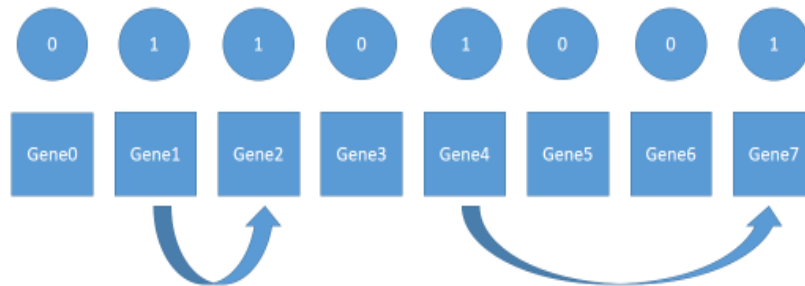
herhangi iki gen Epigenetik Faktör Listesi ile çaprazlama oranına sunulurak optimum sonuca ulaşılma adımlarını kısaltmaktadır. Epigenetik Faktör Listesi ile hangi genlerin uygunluk açısından kısıtlara aykırı olduğu bilgisi belirlenmektedir. Şekil 4.20 ile Epiçaprazlama gösterilmiştir



Şekil 4.20. Epiçaprazlama.

4.8.2. Epideğişim (Epimutasyon) Operatörü

Epideğişim (Epimutasyon) operatörü; Epideğişim işleminde Genetik Algoritmada yapılan Mutasyon işlemine benzerlik gösterse de temel farklılıklar bulunmaktadır. Genetik Algoritma bünyesinde bulunan Mutasyon aynı kromozom içinde herhangi iki genin tamamen rastgele seçilip önceden belirlenmiş olan oranda yer değiştirmesi esasına dayanmaktadır. Ancak Epimutasyon işlemi bu rastgeleliliği belir bir seviyede yönetilmesini sağlamaktadır. Şekil 4.21 ile Epimutasyon gösterilmiştir.



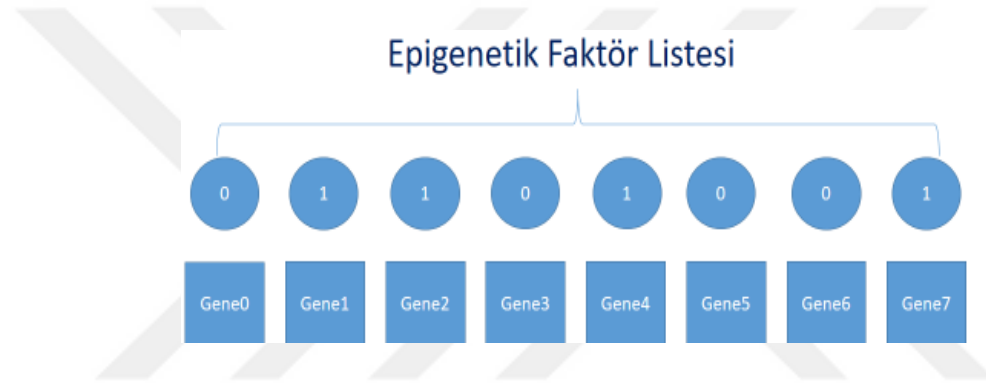
Şekil 4.21. Epimutasyon.

Epimutasyon, Epigenetik Algoritma tasarımında çaprazlama, mutasyon işlemlerinden sonra oluşan yeni bireyler ve Elitizm işlemi sonrasında elde edilen bireyler hafızada tutularak yeni kuşaklara aktarımı sağlanmaktadır. Epigenetik Algoritma tasarımında da

benzer biçimde epikalıtım yapılarak her yeni nesil için Epigenetik Faktör Listesi oluşturulmaktadır.

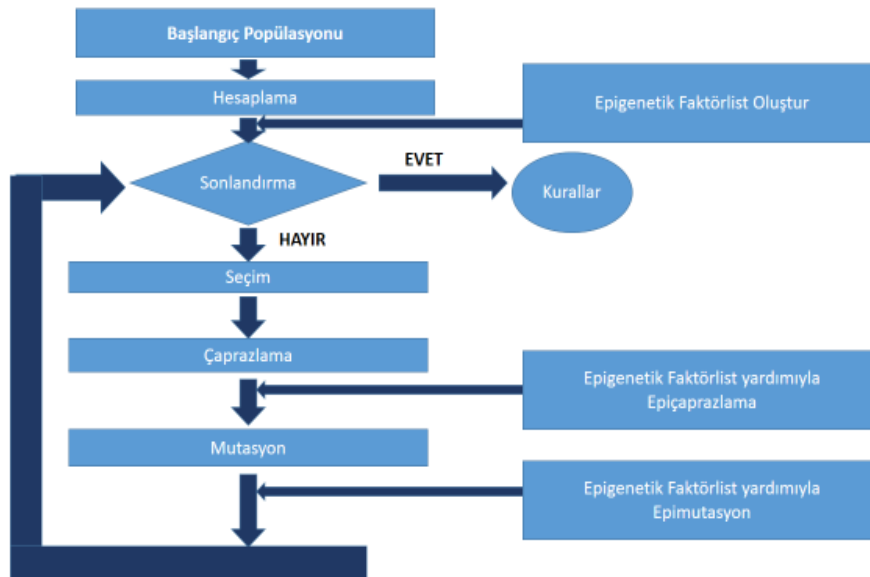
4.8.3. Epigenetik Faktör Listesi

Epigenetik Faktör Listesi, Epigenetik tasarımda bir ceza puanı alan veya ceza almayan her gen için bir Epigenetik faktör listesi oluşturulur. Oluşturulan bu Epigenetik faktör listesi her bir genin aktif olup olmayacağını belirlemektedir. Aktif olup olmamasını belirlemek için her bir gen 1 veya 0 ile belirlenir. Epigenetik faktörler, tüm popülasyon için ve tüm kromozom için bir liste oluşturulması ile meydana gelir. Bu liste ile klasik Genetik Algoritmadaki çaprazlama ve mutasyon işlemlerine katılacak olan genlerin seçiminde kolaylık sağlayacaktır. Şekil 4.22 ile Epigenetik Faktör Listesi gösterilmiştir.



Şekil 4.22. Epigenetik faktör listesi.

Epıçaprazlama ve Epıtmutasyon için '1' değerine sahip genler için geçerli olan yeni geçiş operatörüdür. Şekil 4.23 ile Epigenetik Algoritma akış şeması gösterilmiştir



Şekil 4.23. Epigenetik algoritma akış şeması.

Bu durumda aktif olan genler ile aktif olmayan genlerin ayrımı yapılmış olur. Aktif olan genleri ve aktif olmayan genleri belirledikten sonra Epigenetik Algoritmayı uygulamak için Epigenetik Faktör Listesi oluşturulur. Epigenetik Faktör Listesi hazırlandıktan sonra Epiçaprazlama ve Epimutasyon işlemleri gerçekleştirilmek üzere hazır hale getirilmiş olur. Epigenetik Algoritma sözde kodu şu şekildedir:

```
1  Epigenetik Algoritma
2  {
3      Başlangıç Popülasyonu Oluştur
4      Kriterleri Tanımla
5      Döngü
6      {
7          Uygunluk Fonksiyonuna göre kromozomları değerlendir
8          Kısıtlara uymayan genlerin epigenetik faktör listesini ayarla(Epigenetik Faktör Listesi)
9          Seçim Operatörü Uygula(Roulett Tekerleği Tekniği)
10         Kromozomları Çaprazlama için Havuza gönder.
11         Çaprazlama operatörünü uygula
12         Epiçaprazlamayı epigenetik faktör listesine göre uygula
13         Mutasyon operatörü uygula
14         Epigenetik faktör listesine göre Epimutasyon operatörünü uygula
15         Yeni popülasyon oluştur
16     }
17     Döngü (! (Son nesil mi?) veya! (durmak için diğer kriterler))
18     En iyi kromozomu seç
19 }
```

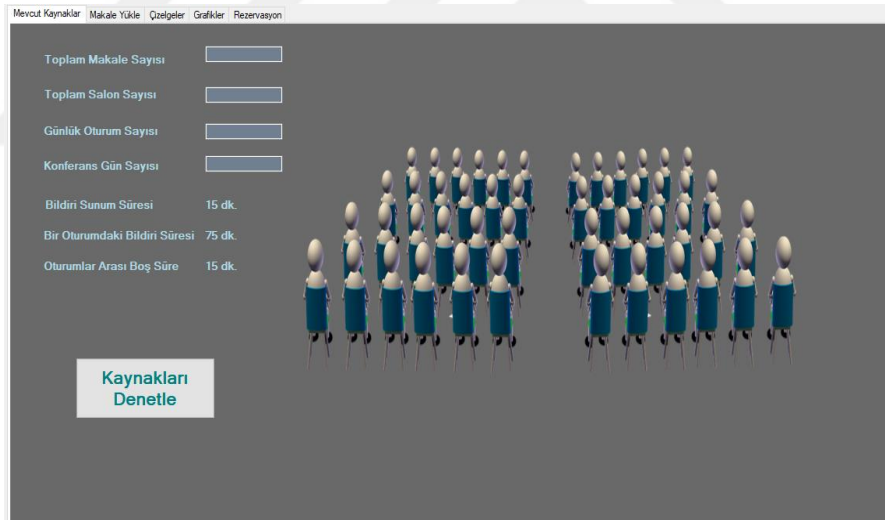
Şekil 4.24. Epigenetik algoritma sözde kodu.

Sonlandırma işlemi algoritmanın tüm amacına yönelik olarak belirlenir ve optimizasyon sağlanır. Epigenetik Algoritma kuşaklar arası uyumu temel alan bir algoritmadır. Aynı kuşaktaki bireyler arası uyuma ve bir kuşağın bir sonraki kuşakla olan uyumuna odaklanmaktadır. Kuşaklar arası uyum sayesinde algoritmanın oluşturulma amacına katkıda bulunacak biçimde önceden tanımlanmış olan faaliyetlerin optimize edilmesini sağlar. Bu optimize bazen hedeflenen değeri artırmaya bazen de azaltmaya yönelik işlemler sürecidir. Kuşaklar arasındaki uyum ise aynı popülasyondaki faaliyetlerini değiştirmeye odaklanır [119].

5. UYGULAMA

Değer kodlama tam sayı kodlama, kompleks sayılı kodlama, reel sayılı kodlama gibi türleri bulunmaktadır. Bu tez kapsamında bildirilerin en uygun ifade edilebileceği kodlama biçiminin tam sayı kodlama olduğu belirlenmiştir.

Bu tezde, konferans çizelgeleme problemi Genetik Algoritma ve Epigenetik Algoritma ile ele alınmış ve algoritmaların kodlanması Visual Studio ortamında C# dili kullanılarak geliştirilmiştir. Genetik Algoritma ve Epigenetik Algoritma kullanılarak hazırlanan program sonlandırıldığı zaman otomatik olarak en iyi kromozoma ait çizelge yapısı Word dosyasına aktarılmıştır. Bu Word dosyası bir konferansa ait bir bildiri kitapçığı olarak kullanılabilir. Geliştirilen uygulamanın başlangıç ekranı Şekil 5.1 ile gösterilmiştir.



Şekil 5.1. Kullanıcı giriş ekranı.

Bir konferans gerçekleşmeden önce gün, sayısı, oturum sayısı, salon sayısı ve toplam makale sayısı bilinmelidir. Bu bilgilerin bilinmesi ile konferans çizelgelemenin mevcut değerler çerçevesinde yapılıp yapılamayacağı belirlenir. Bu sayılardan herhangi birinin bilinmemesi, yanlış bilinmesi veya bu sayılarda yapılan herhangi bir değişiklik çizelgeleme başlamadan önce sisteme belirtilmelidir. Aksi takdirde çizelgelemenin en iyi olması mümkün olmamakla birlikte katılımcılar açısından kabul edilemez hataların doğmasına neden olacaktır. Bu bakımdan geliştirilen programa “Kaynakları Denetle”

isimli bir modül eklenerek konferans çizelgeleme için gün, oturum, salon ve makale sayılarına ilişkin bilgilerin denetlenmesi sağlanmaktadır. Kullanıcı Giriş Ekranı ile konferansta yer alan toplam makale sayısı belirtilmelidir. Ardından konferansa ait gün sayısı, salon sayısı ve oturum sayısı belirtilmelidir. “Kaynakları Denetle” butonuyla konferansın gerçekleşip gerçekleşmeyeceği bilgisi sonucu ekrana yazdırılır. Gerekli alanların doldurulmasıyla Geliştirilen programı çalıştırmadan önce bildiri sayısı, gün sayısı, oturum sayısı, salon sayısı, çaprazlama oranı ve mutasyon oranı Genetik Algoritma butonu aracılığıyla çalıştırılmaktadır. Şekil 5. 2 ile bir makalenin geliştirilen programa kaydedilmesi gösterilmiştir.

Şekil 5.2. Makale kayıt ekranı.

Makalenin adı, makalenin yazarları, makalenin sunum yapacak olan sorumlu yazarı, çalışmanın birinci disiplin alanı, ikinci disiplin alanı ve üçüncü disiplin alanı girilmektedir.

Konferans çizelgeleme problemi gibi kısıt sayısı çok olan kısıtlı eniyileme problemlerinde temel amaç kısıtların sağlanmadığı çözüm önerilerini en az sayıda bulundurmaktır. Bu durumu düzeltmek için ceza fonksiyonu, farklı operatör geliştirme, özgün bir algoritma üretimi ortaya konmaktadır. Bu tez kapsamında ceza fonksiyonu önemi ortaya konmuştur. Problem başlangıcında tanımlanan kısıtlar açısından uygun olmayan gen önceden tanımlanan sert ve yumuşak cezalarla denetlenmiştir.

Kısıtlar sert kısıtlar ve yumuşak kısıtlar olmak üzere iki bölüme ayrılmıştır. Sert kısıtlar şu şekilde belirlenmiştir. Bir yazar aynı anda birden fazla oturumda yer alamaz. Bir yazar aynı anda birden fazla oturumda yer alamaz. Bir oturumda yer alan tüm bildiriler

aynı birincil alana ait olmalı. Bir oturumda yer alan tüm bildiriler aynı ikincil alana ait olmalı. Yumuşak kısıtlar şu şekilde belirlenmiştir. Bir oturumda boş alanlar bulunmamalı. Bir oturumda yer alan tüm bildiriler aynı üçüncül alana ait olmalı.

Genetik Algoritma adımları şu şekildedir [88]:

```
1  Genetik Algoritma
2  {
3      Başlangıç Popülasyonu Oluştur
4      Kriterleri Ayarla
5      Döngü
6      {
7          Uygunluk fonksiyonuna göre kromozom değerlendir
8          Çaprazlama havuzuna kromozomları gönder
9          En iyi kromozomu seç
10         Kromozomlara çaprazlama operatörü uygula
11         Mutasyon operatörü uygula
12         Her kromozom için Tamir operatörü uygula
13         Yeni popülasyon oluştur
14     }
15     En iyi kromozomu al
16 }
```

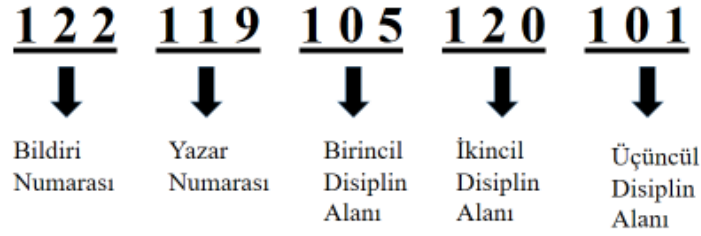
Şekil 5.3. Genetik algoritma sözde kodu.

Her genin pozisyon bilgisi bir bildiri sunumuna karşılık gelmektedir. Genin içinde yer alan bilgiler parça parça anlam ifade etmektedir. İlk 3 bit çalışmanın kimlik numarasını, ikinci 3 bit yazar adını, üçüncü 3 bit çalışmanın birincil disiplin adını, dördüncü 3 bit çalışmanın ikincil disiplin adını, ikinci 3 bit çalışmanın üçüncül disiplin adını, belirtmektedir. Bu 15 bitlik bir sayı bir bildiriye ait bilgileri barındırmaktadır. Bu oluşan bir gen tüm evrimsel süreçler boyunca tek bir adettir. Duplikasyona uğraması veya kaybolması kabul edilebilir bir durum değildir. Şekil 5.3 ile tamsayı kodlama ile üretilen gen gösterilmiştir.

1 2 2 1 1 9 1 0 5 1 2 0 1 0 1

Şekil 5.4. Gen gösterimi.

Ancak genlerin tüm biraraya gelmeleri birer anlam taşımaktadır. Örneğin, bir oturumda yer alan makale sayısı 5 olduğunda, 1. Sıradaki gen ile 5 arasındaki genler bir oturumu temsil etmektedir. Şekil 5.4 ile genin oluşumu gösterilmiştir.



Şekil 5.5. Genin oluşumu.

Genler birbirlerinden tamamen ayrı bilgi taşımaktadır. Ancak genlerin tüm biraraya gelmeleri birer anlam taşımaktadır. Örneğin, bir oturumda yer alan makale sayısı 5 olduğunda, 1. Sıradaki gen ile 5 arasındaki genler bir oturumu temsil etmektedir. Şekil 5.5 ile genin yapısı bileşenlerine ayrılmıştır.

1 2 2 | 1 1 9 | 1 0 5 | 1 2 0 | 1 0 1

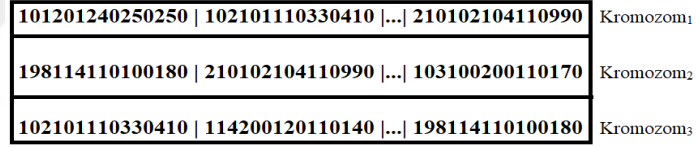
Şekil 5.6. Genin yapısı.

Gen içerisinde ilk 3 bitlik bölüm bildiri numarasını, ikinci 3 bitlik bölüm bildirinin yazarını, üçüncü 3 bitlik bölüm bildirinin birincil disiplin alanını, dördüncü 3 bitlik bölüm bildirinin ikincil disiplin alanına, beşinci 3 bitlik bölüm bildirinin üçüncül disiplin alanını belirtmektedir. Geliştirilen yazılımın yönetim panelinde kullanıcı; çalışma adı ve yazar adını yazdıktan sonra ekrana listelenen menüden çalışmanın konularını seçilmelidir. Program için hazırlanan veritabanında bilim alanları bulunmaktadır. Veritabanında kayıtlı 13 birincil disiplin alanı bulunmaktadır. Bunlar; Eğitim Bilimleri ve Öğretmen Yetiştirme Temel Alanı, Fen Bilimleri ve Matematik Temel Alanı, Filoloji Temel Alanı, Güzel Sanatlar Temel Alanı, Hukuk Temel Alanı, İlahiyat Temel Alanı, Mimarlık, Planlama ve Tasarım Temel Alanı, Mühendislik Temel Alanı, Sağlık Bilimleri Temel Alanı, Sosyal, Beşeri ve İdari Bilimler Temel Alanı, Ziraat, Orman ve Su Ürünleri Temel Alanı, Spor Bilimleri Temel Alanı, Diğer.

Birincil disiplin alanı olan Mühendislik Temel Alanına bağlı olan Atmosfer Bilimleri ve Meteoroloji Mühendisliği, Bilgisayar Bilimleri ve Mühendisliği, Biyomedikal Mühendisliği, Biyomühendislik, Çevre Bilimleri ve Mühendisliği, Deniz ve Gemi Mühendisliği, Elektrik-Elektronik Mühendisliği, Elektronik-Haberleşme Mühendisliği, Endüstri Mühendisliği, Enerji Sistemleri Mühendisliği, Gıda Bilimleri ve Mühendisliği, Harita Mühendisliği, İnşaat Mühendisliği, Kimya Mühendisliği, Maden Mühendisliği, Makine Mühendisliği, Malzeme ve Metalurji Mühendisliği, Mekatronik Mühendisliği,

Nükleer Mühendisliği, Otomotiv Mühendisliği, Petrol Mühendisliği, Tekstil Bilimleri ve Mühendisliği, Uçak-Havacılık-Uzay Mühendisliği, Yer Bilimleri ve Mühendisliği, Diğer bulunmaktadır.

13 bilimsel Alana ait olan 258 ikincil disiplin alanı bulunmaktadır. Bu birincil ve ikincil disiplin alanına ait olan 1904 tane üçüncül disiplin alanı bulunmaktadır. Seçilen konunun birincil disiplin alanı seçildikten sonra birincil alana ait olan ikincil alanlar listelenmektedir. İkincil alanlar listelendikten sonra ilgili ikincil alan seçilmelidir. İkincil alan seçildikten sonra o alana ait olan üçüncül alanlar listelenmektedir. Örneğin birincil Disiplin alanına Mühendislik, Hukuk, Sağlık Bilimleri gibi alanlardan biri seçilmelidir. Birincil disiplin alanında Mühendislik seçildiği durumda, ikincil disiplin alanına Bilgisayar Bilimleri ve Mühendisliği, Biyomedikal Mühendisliği, Elektrik-Elektronik Mühendisliği gibi ikincil disiplin alanı seçilmelidir. İkincil disiplin alanı Bilgisayar Bilimleri ve Mühendisliği seçildiğinde ona bağlı olan Bulut Bilişim, Bilgi Güvenliği ve Kriptoloji, Yapay zekâ gibi üçüncül disiplin alanı seçilmelidir. Şekil 5.6 ile Kromozom Yapısı gösterilmiştir.



Şekil 5.7. Kromozom yapısı.

Genlerin bir araya gelmesi ile oluşturulan kromozomlar çözüm için birer aday örneklerdir. Bu çözüm adayları bir araya gelerek popülasyon adındaki veri kümesini oluştururlar. Kromozomların bir araya gelerek oluşturdukları ilk popülasyon başlangıç popülasyonudur. Başlangıç popülasyonu konferansa ait tüm bildirimlerin yer aldığı kromozom adı verilen dizi yapısı ile tamamen rastgele oluşturulmuştur. Literatür örnekleri incelendiğinde problem türüne özgü farklılıklar olmakla birlikte ortalama 20 alınmaktadır. Bu çalışmada popülasyon boyutu 20 kromozom temel alınarak oluşturulmuştur. Başlangıç popülasyonu için belirlenen popülasyon boyutu tüm Genetik Algoritma boyunca sabit tutulmaktadır. Başlangıç popülasyonu Genetik Algoritma uygulamalarında sonuca doğrudan etki edebilmektedir. Bu durum başlangıç popülasyonun sonucu olumlu ya da olumsuz etkilememesi için bazı öneriler sunulmuştur. Başlangıç popülasyonu sabit tutmak bu önerilerden biridir. Ancak bu tez kapsamında çözüm uzayı oldukça geniş olduğundan bu öneri tercih edilmemiştir. Bir

diğer öneri ise çok sayıda deney yaparak farklı başlangıç popülasyonlarının oluşmasını sağlamak ve daha tutarlı sonuç elde etmektir. Bu bağlamda Genetik Algoritmanın aynı problem için tekrarlı çalıştırılmasında 10, 20, 100 gibi sayıda uygulama sayıları tavsiye edilmektedir. Bu tez çalışmasında 10 tekrarlı deney sayısı referans alınarak sonuçlar incelenmiştir.

Uygunluk Değeri kromozomların popülasyon içindeki bireylerin en iyisini seçme şeklinde hesaplanmıştır. Popülasyondaki kısıtlara uymayan kromozomlara verilen cezalar dikkate alınmıştır. Tüm kromozomların ceza değerleri belirlenmiş ve o popülasyona ait tüm kromozomların ceza değerleri toplanmıştır. Uygunluk değerinin hesaplanmasının ardından popülasyon içindeki en yüksek uygunluk değerine sahip kromozom en iyi kromozom olarak belirlenmektedir. Uygunluk fonksiyonu, popülasyon içindeki her bir elamanın ulaşılması istenen sonuca yakınlığı ifade eden fonksiyondur. Örneğin, K elemanlı bir dizi için $f(x)$ fonksiyonu ile maksimum değere ulaşılması isteniyor olsun. Bu durumda fonksiyonun i . Elemanın uygunluk değeri;

$$f' = \sum_{i=1}^K f_i \quad (5.1)$$

olarak tanımlanır ve uygunluk değeri, $\frac{f_i}{f'}$ biçiminde hesaplanmaktadır. Kısıtlı optimizasyon problem türlerinde belirlenen tüm kısıtlara özgü ceza puanları hesaplanmaktadır. Ceza puanları hesaplandıktan sonra uygunluk fonksiyonu hesaplanmaktadır. Tüm kısıtlar için ceza puanları toplamı,

$$Cp = \sum_{a=0}^x \sum_{b=0}^y \sum_{c=0}^z P_a * C_{bc} \quad (5.2)$$

Şeklinde belirlenmiştir. Buradan yararlanarak uygunluk değeri ise:

$$f = \frac{1}{1 + \sum_{a=0}^x \sum_{b=0}^y \sum_{c=0}^z P_a * C_{bc}} \quad (5.3)$$

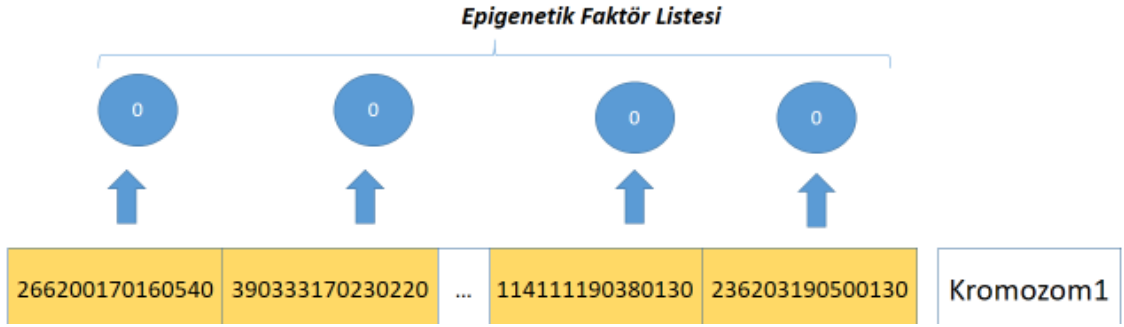
olarak hesaplanmaktadır.

*y = kromozom sayısı z = gen sayısı, x = kısıt sayısı, C_{bc} = b. Kromozomdaki ceza puanına sahip olan c. Gen, P_a = a. kısıta ait olan ceza puanı.

5.1. EPİGENETİK ALGORİTMANIN PROBLEME UYGULANMASI

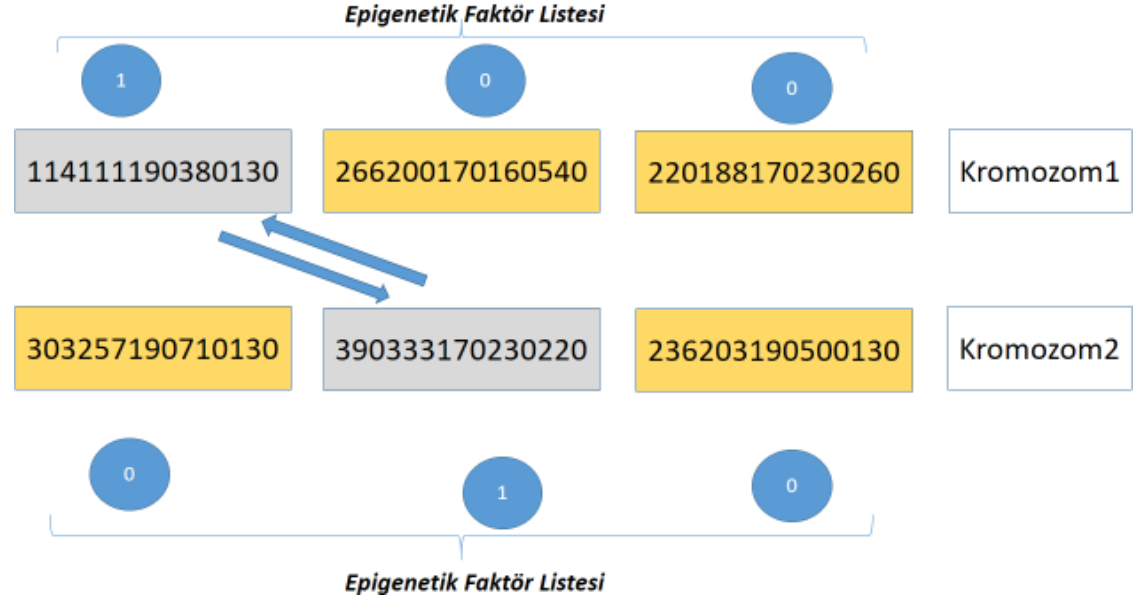
Epigenetik Algoritma gen yapısı, kromozom yapısı ve popülasyon yapısı Genetik Algoritma ile aynı şekilde uygulanmıştır. Konferansa ait bildiriler 15 bitlik bir gen oluşturulmuş ve tamamen rastgele olarak kromozomlar üretilmiştir. Bu bakımdan gen

ve kromozom nasıl üretildiği bu başlıkta tekrarlanmayacaktır. Bu başlıkta yalnızca Epigenetik Algoritmaya özgü bileşenler açıklanacaktır. Şekil 5.7 ile Epigenetik Faktör Listesi gösterilmiştir.



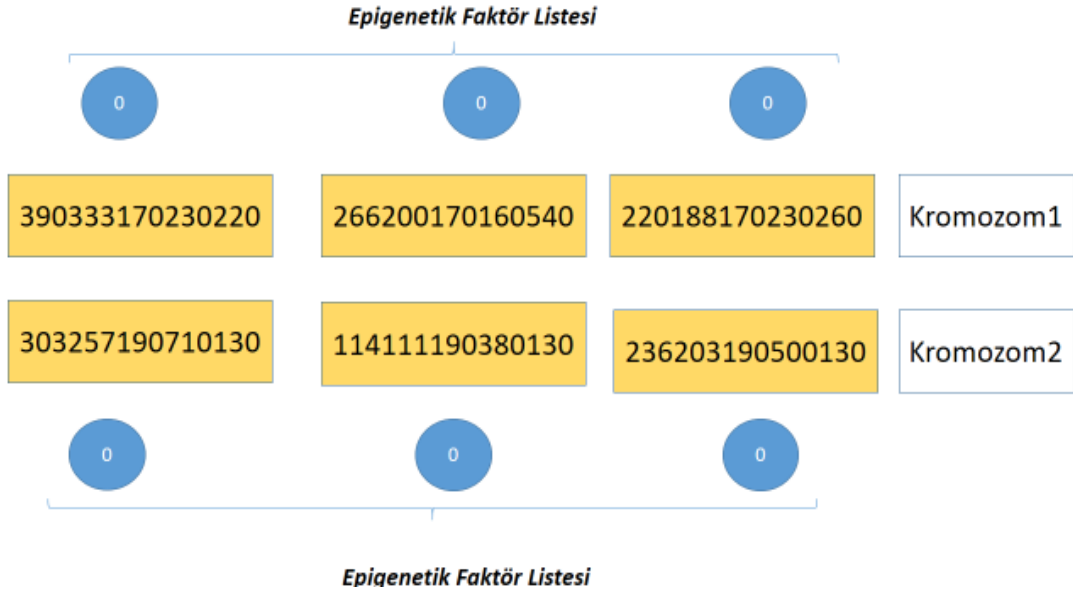
Şekil 5.8. Epigenetik faktör listesi.

Burada yer alan bir kromozomun 5 genlik bir bölümü ele alınmıştır. 5 farklı gen içinde 4 tane genin aynı birincil alan taşıdığı bir tane genin ise birincil alanın diğer dört geninden farklı olduğu görülmektedir. Bu dört gen Epifaktör Listesinde 0 olarak işaretlenirken birincil alanı farklı olan gen 1 olarak işaretlenmiştir. Bu durumda Epifaktör liste içinde problem kısıtlarına uymayan sorunlu genlerin 1, problem kısıtlarına uyan sorunsuz genlerin 0 olarak işaretlendiği belirtilmektedir. Dolayısıyla Epigenetik Faktör Listesi kromozom uzunluğunda ve kromozom sayısı kadar tanımlanmalı ve 1 veya 0 olan değerleri problem kısıtlarına göre tanımlanmalıdır. Ayrıca her nesilde kromozomlar değişeceği için Epigenetik Faktör Listesinin her iterasyonda tekrar güncellenmesi gerekecektir. Şekil 5.8 ile Epiçaprazlama gösterilmiştir.



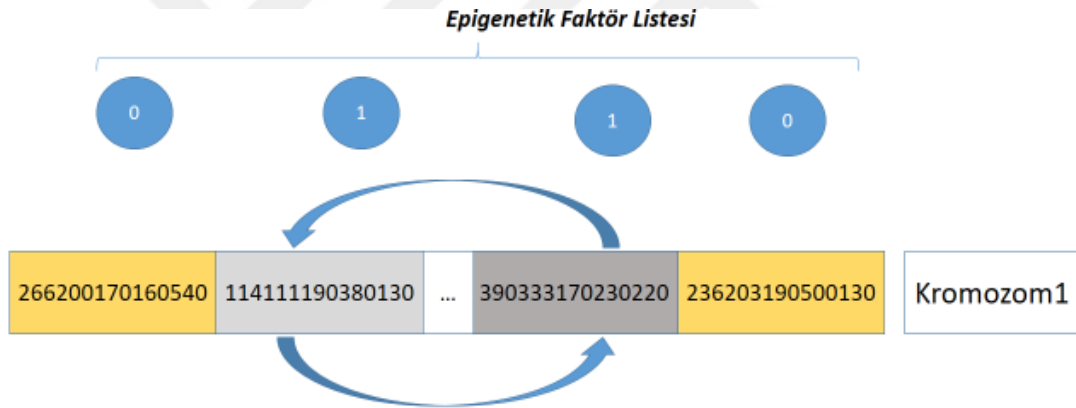
Şekil 5.9. Epiçaprazlama.

Burada iki farklı kromozom ve bu kromozomlara ait iki farklı Epigenetik Faktör Listesi gösterilmiştir. İki kromozomun Epigenetik Faktör Listesi incelendiğinde her ikisinde de bir adet gen 1 olarak işaretlenmiştir. Epiçaprazlama işleminin yapılabilmesi için iki farklı kromozom üzerindeki 1 olarak işaretlenen herhangi iki gen olarak işaretlenmiştir. Ancak bu örnek incelendiğinde birinci kromozom üzerinde bir gen, ikinci kromozom üzerinde bulunan bir gen 1 olarak işaretlenmiştir. 1 olarak işaretlenen bu bölgeler ile Epiçaprazlama işlemi gerçekleştirilmesi mümkün olacaktır. Epiçaprazlama ile 1 olarak işaretlenen bu iki gen Epiçaprazlama olasılığı sağlandığı durumda karşılıklı olarak yer değiştirecektir. Epi çaprazlama olasılığı tıpkı Genetik Algoritmadaki çaprazlama olasılığı gibi oluşturulmaktadır. Ancak Epiçaprazlama olasılığı Epigenetik Algoritmanın doğası gereği Genetik Algoritmadaki çaprazlama olasılığından çok daha düşük olarak belirlenmektedir. Şekil 5.9 ile Epiçaprazlama sonrasında kromozomların yeni dizilimleri ve yeni oluşan Epigenetik Faktör Listesi gösterilmiştir.



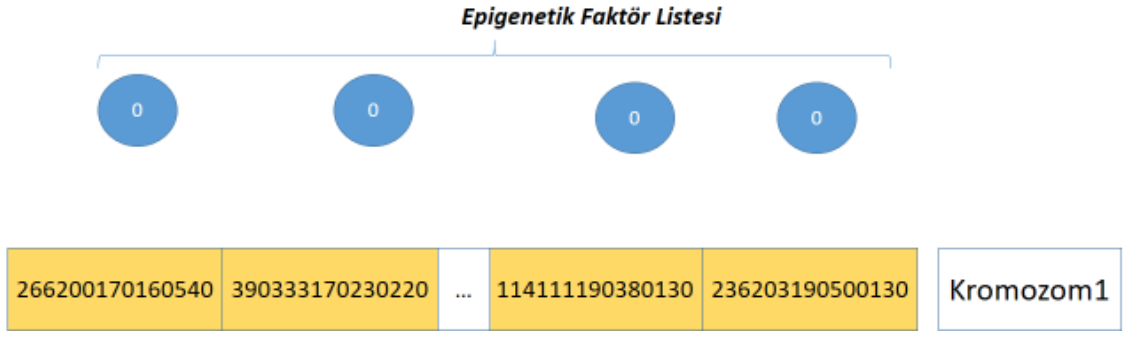
Şekil 5.10. Epiçaprazlama sonrasındaki kromozomların dizilimleri.

Şekil 5.10 ile Epimutasyon gösterilmiştir.



Şekil 5.41. Epimutasyon.

Burada bir adet kromozom gösterilmiştir. Bu kromozoma ait olan Epigenetik Faktör Listesi hemen kromozomun üst bölgesinde gösterilmiştir. Kromozom içinde problem kısıtlarına uymayan genler 1 olarak işaretlenmiştir. Epimutasyon oranı koşulu sağlandığından bu 1 olarak işaretlenen genler karşılıklı olarak yer değiştirecektir. Böylece kromozom yapısı değişmiş olacaktır. Problem kısıtlarına uymayan genler yer değiştirdiğinden dolayı kromozom daha iyi bir kromozom olma eğiliminde olacaktır. Şekil 5.11 ile Epimutasyon sonrası kromozom dizilimi ile güncellenen Epigenetik Faktör listesi gösterilmiştir.



Şekil 5.52. Epimutasyon sonrası kromozom dizilimi.

Geliştirilen program sonlandırıldığında Genetik Algoritma ve Epigenetik Algoritma ile bir konferans sunum kitapçık programı oluşturulmaktadır.



6. DENEYSEL SONUÇLAR

Tezin bu bölümde Genetik Algoritmanın ve Epigenetik Algoritmanın uygulanması amacıyla geliştirilen programın farklı gün sayısının problem çözümüne etkisi, farklı oturum sayısının çözüme etkisi, farklı salon sayısının etkisi gibi parametrelerinin grafik üzerinde gösterimine ve ortaya çıkan çizelgeleme yapısının gösterimine yer verilmiştir. Ayrıca problemin bir başka gerçeklik boyutu olan rezervasyon durumu ele alınmış ve rezervasyonun problem çözümüne olan etkisi incelenmiştir.

Bir konferans gerçekleştirilmeden önce gün, oturum ve salon sayıları ile kapasite hesaplanmaktadır. Bu kapasitenin bildiri sayısından büyük veya eşit olması beklenir. Kapasite ile bildiri sayısının eşit olması konferansta hiç boş oturum ve boş zaman olmamasını gerektirir. Ancak kapasitenin bildiri sayısından büyük olduğu durumlarda boş zamanlar oluşmaktadır. Konferans çizelgelemede boş oturumların olduğu bu duruma boş zaman denilmektedir. Boş zaman dinamik bir çizelgeleme problemi için bir gerekliliktir. Konferans çizelgeleme için kapasiteyi oluşturan kaynakların belirlenmesi çok erken aşamada olmaktadır. Bunların sayısında bir değişiklik yapmak oldukça zordur. Ancak bildiri sayısı kapasite sayısına ulaşamayabilir. Ayrıca konferans süresinin uzatılması ile de bildiri sayısının artması da mümkündür. Yazarların bazıları konferans yönetim kurulundan kabul mektubu almasına rağmen konferanstan geri çekilebilmektedir. Böyle durumlarda konferans çizelgelemenin yeniden yapılması gerekmektedir. Ancak dinamik olarak hazırlanan bir yazılımda mevcut bildiri listesine bildiri eklemek ya da mevcut listeden bildiri çıkarmak çizelgelemeyi en az düzeyde etkilemektedir. Bu çalışmada dinamik nitelikte olması için kapasite sayısı ile bildiri sayısı arasında kalan bölüm boş zaman olarak tanımlanmaktadır. Böylece çizelgelemeye başlamadan önce yapılacak olan bildiri sayısının artması ya da azalması yazılımın çalışmasını engellemeyecektir.

Bir konferans çizelgelemede bildiri sayısı ile kapasite arasındaki farkın büyük olması boş zamanın çok olması demektir. Bildiri sayısı çok oldukça boş zaman sayısı az olacaktır. Bu bakımdan bildiri sayısı sabit iken kapasite artırımı yapmak boş zaman sayısını artıracaktır. Bölüm 2.3 konferans çizelgeleme başlığında tanımlanan sert kısıtlar

gereği boş zamanların konferansın son gün olması gerekmektedir. Bildiri sayısı sabit iken gün, oturum ve salon sayılarının artırılması ile kapasite artışı yapmak boş zaman sayısını artırmaktadır. Bu durum problem çözümünü zorlaştırmaktadır. Çünkü boş zaman ilk günlerde olduğu sürece ceza değerini yükseltmektedir. Hedeflenen en iyi çözümde boş zamanların konferansın son gününde son oturumunda yer alması beklenmektedir.

Bu tez çalışması kapsamında elde edilen gerçek bir uluslararası multidisipliner bir konferans bildiri listesi kullanılmıştır. Bu liste 200 ve 300 bildiri olarak iki farklı liste olarak ayrılmıştır. Çizelge 6.1 ile 200 bildirinin Genetik Algoritma farklı gün, oturum ve salon sayıları bakımından Genetik Algoritma ile çizelgelenmesi gösterilmiştir.

Çizelge 6.1. 200 Bildirinin Genetik Algoritma ile Çizelgelenmesi.

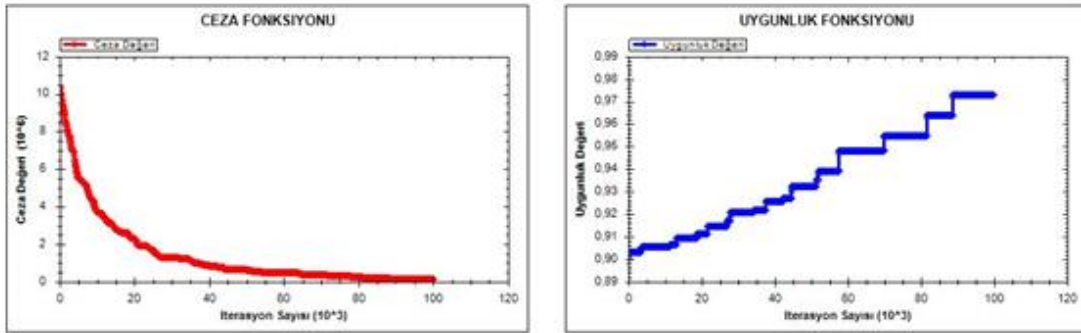
200 Bildiri						
2 Gün						
	4 Oturum			6 Oturum		
	4 Salon	5 Salon	6 Salon	4 Salon	5 Salon	6 Salon
İlk Ceza Değeri	XXX	18492.0	2522069.5	2276954.5	6166139.1	10032105.3
En İyi Kromozom Ceza Değeri	XXX	3508.5	6196.6	8699.6	10649	23787.4
Uygunluk Değeri	XXX	0.9974533	0.9961617	0.9960522	0.9960972	0.9960562
İtersyon Sayısı	XXX	100000	100000	100000	100000	100000
3 Gün						
	4 Oturum			6 Oturum		
	4 Salon	5 Salon	6 Salon	4 Salon	5 Salon	6 Salon
İlk Ceza Değeri	2344149.1	6199135.7	10420143.6	9964208.4	16901860.5	21378905.2
En İyi Kromozom Ceza Değeri	9504.6	11718.0	26416.3	25004.2	40117.4	55053.8
Uygunluk Değeri	0.9976458	0.9952727	0.9951305	0.9942096	0.9913698	0.9907918
İtersyon Sayısı	100000	100000	100000	100000	100000	100000

Çizelge 6.1 bünyesinde gösterilen XXX ifadesi konferans kapasitesinin bildiri sayısını karşılamadığı için çizelgelemenin yapılamadığını belirtmektedir. Burada en iyi ceza değerine ait olan kromozom uygunluk değerleri gösterilmiştir. Denklem 5.3 ile belirtildiği gibi en iyi kromozomun ceza değerinin içinde bulunduğu popülasyonun toplam ceza değeri belirlenebilmektedir. Çizelge 6.2 ile Genetik Algoritmanın 300 bildiri bulunan bir konferansın farklı gün, oturum ve salon sayıları eşliğinde çizelgelenmesi sonucu elde edilen ceza değerleri ve uygunluk değerleri gösterilmiştir.

Çizelge 6.2. 300 bildirinin genetik algoritma ile çizelgelenmesi.

300 Bildiri						
2 Gün						
	4 Oturum			6 Oturum		
	4 Salon	5 Salon	6 Salon	4 Salon	5 Salon	6 Salon
İlk Ceza Değeri	XXX	XXX	XXX	XXX	410495.2	5634381.2
En İyi Kromozom Ceza Değeri	XXX	XXX	XXX	XXX	16595.2	27478.1
Uygunluk Değeri	XXX	XXX	XXX	XXX	0.9890468	0.9763063
İterasyon Sayısı	XXX	XXX	XXX	XXX	100000	100000
3 Gün						
	4 Oturum			6 Oturum		
	4 Salon	5 Salon	6 Salon	4 Salon	5 Salon	6 Salon
İlk Ceza Değeri	XXX	420100.2	5946920.7	5919414.8	14995469.2	23339238.8
En İyi Kromozom Ceza Değeri	XXX	14775.4	28412.6	29152.2	46255.6	72065.8
Uygunluk Değeri	XXX	0.9849908	0.9728186	0.972302	0.971975	0.9709679
İterasyon Sayısı	XXX	100000	100000	100000	100000	100000

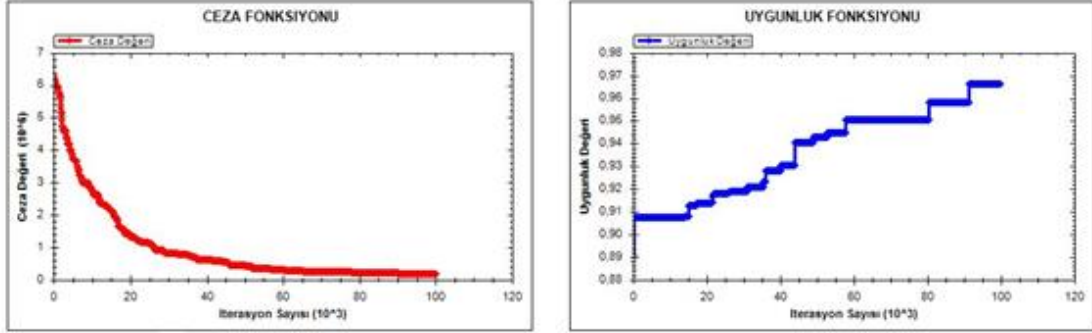
Çizelge 6.2 bünyesinde gösterilen XXX ifadesi konferans kapasitesinin bildiri sayısını karşılamadığı için çizelgelemenin yapılamadığını belirtmektedir. Şekil 6.1 ile 200 bildirili bir konferansın 2 gün, 6 oturum, 6 salon eşliğinde Genetik Algoritma ile çizelgelenmesinin sonucunda elde edilen ceza fonksiyonu ve uygunluk fonksiyonu gösterilmiştir.



Şekil 6.1. 200 bildiri, 2 gün, 6 oturum, 6 salon genetik algoritma.

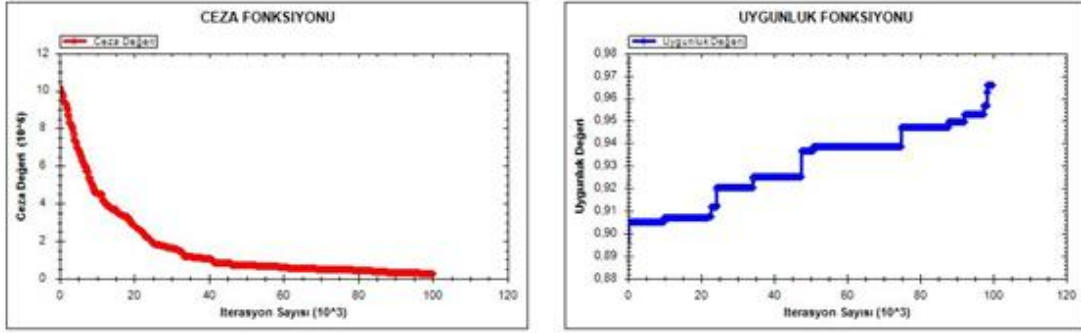
Ceza fonksiyonun 0'a ulaşması hedeflenmektedir. Uygunluk fonksiyonun amacı 1.0'a ulaşmaktır. Genetik Algoritma çalıştırıldığında durdurmak için bazı kriterler bulunmaktadır. Bunlar hedef ceza değeri, hedef uygunluk değeri ve iterasyon sayısıdır. Bu tez için yapılan deneylerde iterasyon sayısı durdurma kriteri olarak belirlenmiştir. 100 bin iterasyona ulaşıldığında algoritma sonlandırılmıştır. 100 bin iterasyon sonucu elde edilen ceza değerleri ve uygunluk değeri gösterilmiştir. Şekil 6.1 ile 200 bildiri 2

gün, 6 oturum ve 6 salon eşliğinde çizelgelenmiştir. Ceza değeri 11000000'den başlayarak logaritmik olarak azalmıştır. 100 bin iterasyon sonucu ceza değeri 23787'ye düşmüştür. Şekil 6.2 ile 300 bildirili bir konferansın 2 gün, 6 oturum, 6 salon eşliğinde Genetik Algoritma ile çizelgelenmesinin sonucunda elde edilen ceza fonksiyonu ve uygunluk fonksiyonu gösterilmiştir.



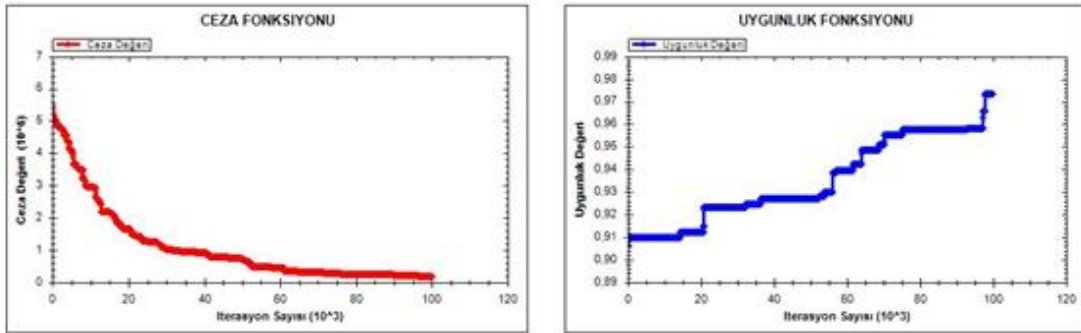
Şekil 6.2. 300 bildiri, 2 gün, 6 oturum, 6 salon genetik algoritma.

300 bildiri bulunan bu senaryoda ceza değeri 27478'e düşmüştür. Çizelge 6.1 ve Çizelge 6.2 ile belirtildiği gibi Şekil 6.1 de bulunan durum ile karşılaştırıldığında aynı iterasyon sayısında elde edilen ceza değeri 200 bildiri ile yapılan çizelgelemede daha düşük olmuştur. Ayrıca uygunluk değeri, 200 bildiri ile yapılan çizelgelemede daha yüksek olmuştur. Bu durum aynı gün, salon ve oturum sayısında bildiri sayısının artmasının çizelgeleme performansını etkilediğini ortaya koymaktadır. Şekil 6.3 ile 200 bildirili bir konferansın 3 gün, 4 oturum, 6 salon eşliğinde Genetik Algoritma ile çizelgelenmesinin sonucunda elde edilen ceza fonksiyonu ve uygunluk fonksiyonu gösterilmiştir.



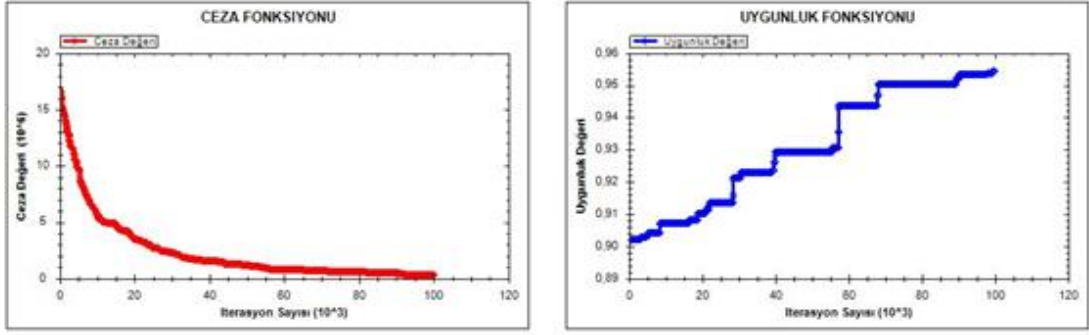
Şekil 6.3. 200 bildiri, 3 gün, 4 oturum, 6 salon genetik algoritma.

Ceza fonksiyonu 26416 ceza değerine düşmüştür. Ceza fonksiyonu 2 gün 6 oturum ve 6 salon eşliğinde çizelgelenen durum ile benzerlik göstermektedir. Çünkü bu iki durum aynı gen sayısı ile ifade edilmektedir. Bu iki durumda kromozom içinde bulunan boş zaman sayıları da eşittir. Şekil 6.4 ile 300 bildirili bir konferansın 3 gün, 4 oturum, 6 salon eşliğinde Genetik Algoritma ile çizelgelenmesinin sonucunda elde edilen ceza fonksiyonu ve uygunluk fonksiyonu gösterilmiştir.



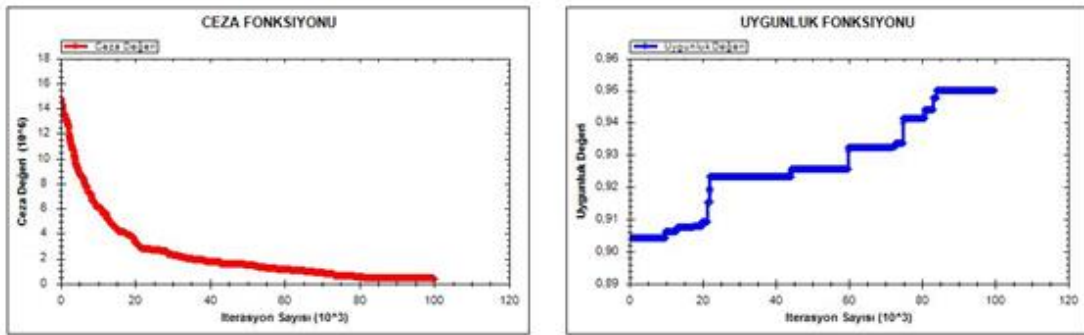
Şekil 6.4. 300 bildiri, 3 gün, 4 oturum, 6 salon genetik algoritma.

300 bildiri bulunan bu durumda elde edilen en küçük ceza değeri 28412 olmuştur. Bu durum Şekil 6.2 ile gösterilen durum ile benzerlik göstermektedir. Şekil 6.2 ile Şekil 6.4 aynı gen sayısı ile ifade edilmektedir. Bu iki durumda kromozom içinde bulunan boş zaman sayıları da eşittir. 300 bildiri bulunan bu durum 200 bildiri ile yapılan çizelgelemeden daha yüksek ceza değerine sahiptir. Bu durum bildiri sayısının artmasının çizelgeleme performansını etkilediğini ortaya koymaktadır. Şekil 6.5 ile 200 bildirili bir konferansın 3 gün, 6 oturum, 5 salon eşliğinde Genetik algoritma ile çizelgelenmesinin sonucunda elde edilen ceza fonksiyonu ve uygunluk fonksiyonu gösterilmiştir.



Şekil 6.5. 200 bildiri, 3 gün, 6 oturum, 5 salon genetik algoritma.

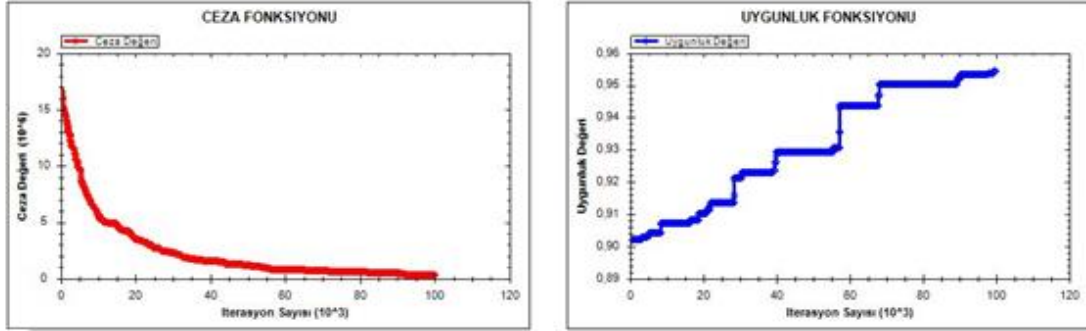
Çizelge 6.1 ile belirtildiği üzere elde edilen ceza değeri 40117 olmuştur. Elde edilen bu ceza değeri 3 gün 4 oturum 6 salon eşliğinde yapılan çizelgelemeden daha yüksektir. Bu durum çizelgelemenin kapasite arttığında elde edilen ceza değerinin daha yüksek olduğunu ortaya koymaktadır. Grafik üzerinden ve Çizelge 6.1 üzerinden anlaşılacağı üzere 200 bildiri 3 gün, 6 oturum, 5 salon ile çizelgelendiğinde ilk cez değeri 3 gün, 4 oturum 6 salon ile yapılan çizelgelemeden daha büyüktür. 3 gün 6 oturum 5 salon çözüm uzayında daha büyük bir alanda arama yaptığı için optimizasyon bakımından daha yüksek bir iyileşme göstermiştir. Ancak elde edilen en küçük ceza değeri bakımından 3 gün 4 oturum 6 salon daha iyi bir sonuç üretmiştir. Şekil 6.6 ile 300 bildirili bir konferansın 3 gün, 6 oturum, 5 salon eşliğinde Genetik Algoritma ile çizelgelenmesinin sonucunda elde edilen ceza fonksiyonu ve uygunluk fonksiyonu gösterilmiştir.



Şekil 6.6. 300 bildiri, 3 gün, 6 oturum, 5 salon genetik algoritma.

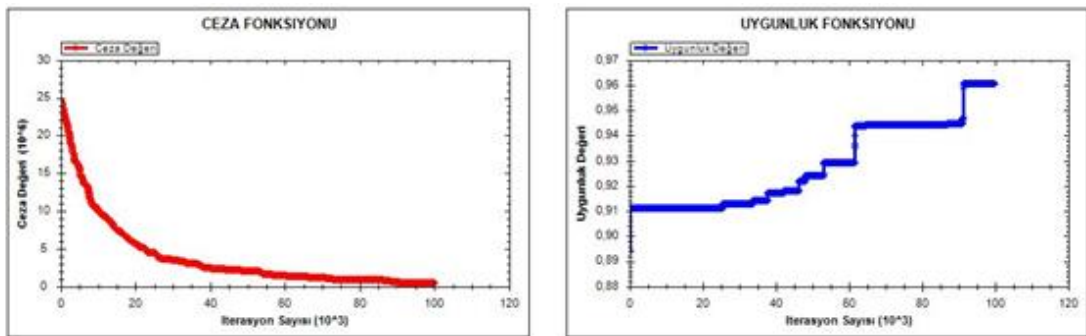
Çizelge 6.1 ile belirtildiği gibi 300 bildiri 3 gün 6 oturum 5 salon eşliğinde çizelgelendiğinde 46255 ceza değerine ulaşmaktadır. Bu durum Şekil 6.5 ile gösterilen 200 bildirin çizelgelendiği durumdan daha yüksek bir ceza değerine ulaştığını göstermektedir. Bu durum bildiri sayının artmasının çizelgeleme performansını olumsuz

etkilediğini ortaya koymaktadır. Şekil 6.7 ile 200 bildirili bir konferansın 3 gün, 6 oturum, 6 salon eşliğinde Genetik Algoritma ile çizelgelenmesinin sonucunda elde edilen ceza fonksiyonu ve uygunluk fonksiyonu gösterilmiştir.



Şekil 6.7. 200 bildiri, 3 gün, 6 oturum, 6 salon genetik algoritma.

Çizelge 6.1 ile gösterildiği gibi bu durumda elde edilen en küçük ceza değeri 55053 olmaktadır. Elde edilen bu ceza değeri 200 bildirinin 3 gün 6 oturum 5 salon ile çizelgelenildiği durumdan daha yüksek bir ceza değerine sahiptir. Bu durum kapasite artırmanın çözüm uzayındaki büyümeye neden olmasından dolayı çözümü zorlaştırmaktadır. Şekil 6.8 ile 300 bildirili bir konferansın 3 gün, 6 oturum, 6 salon eşliğinde Genetik Algoritma ile çizelgelenmesinin sonucunda elde edilen ceza fonksiyonu ve uygunluk fonksiyonu gösterilmiştir.



Şekil 6.8. 300 bildiri, 3 gün, 6 oturum, 6 salon genetik algoritma.

Çizelge 6.1 ile gösterildiği gibi bu durumda ceza değeri 72065 olarak belirlenmiştir. Bu durum 200 bildirinin 3 gün 6 oturum 6 salon eşliğinde yapılan çizelgelemeden daha büyük olduğunu göstermektedir. Bu durum bildiri sayısı arttıkça çizelgelemenin zorlaştığını ortaya koymaktadır. Ayrıca 300 bildirinin 3 gün 6 oturum 6 salon ile çizelgelenmesi, 3 gün 6 oturum 5 salon eşliğinde çizelgelenmesinden daha zor olmuştur. İlk iterasyonda üretilen ceza değeri 3 gün 6 oturum 6 salon durumunda daha fazladır.

Bu durum çözüm uzayının büyüdüğünü göstermektedir. 3 gün 6 oturum 6 salon ilk iterasyon bakımından daha büyük bir iyileşme göstermesine rağmen algoritma onunda elde edilen çözüm 3 gün 6 oturum 5 salon ile elde edilen sonucun en iyi çözüme daha yakın olduğunu ortaya koymaktadır. 300 bildirinin 3 gün 6 oturum 6 salon eşliğinde çizelgelenmesi, 300 bildirinin 2 gün 6 oturum 6 salon eşliğinde çizelgelenmesinden de zor olmaktadır. Çünkü 3 gün 6 oturum 6 salon eşliğinde yapılan çizelgelemenin çözüm uzayı daha büyüktür. Daha büyük çözüm uzayında en iyi çözüme olan yakınsama daha güç olmaktadır. Bu durum elde edilen en küçük ceza değerine de yansımaktadır. Çizelge 6.3 ile 200 bildirinin Epigenetik Algoritma ile çizelgelenmesi gösterilmiştir.

Çizelge 6.3. 200 bildirinin epigenetik algoritma ile çizelgelenmesi.

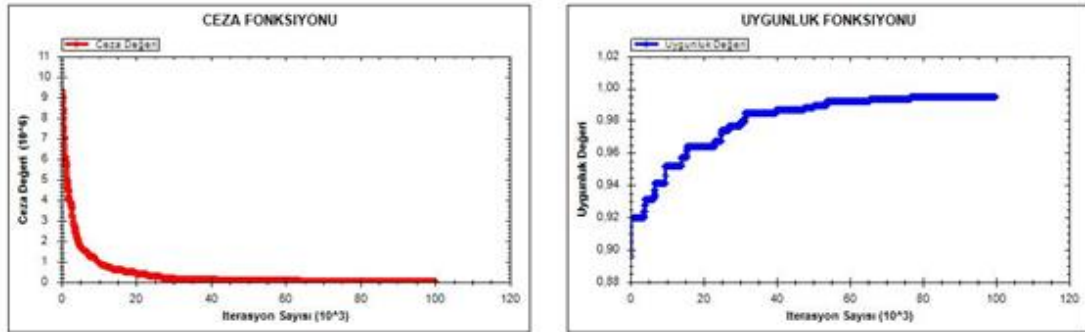
200 Bildiri						
2 Gün						
	4 Oturum			6 Oturum		
	4 Salon	5 Salon	6 Salon	4 Salon	5 Salon	6 Salon
İlk Ceza Değeri	XXX	19258.7	2412984.6	2367892	6258833.2	10111981.2
En İyi Kromozom Ceza Değeri	XXX	2770.5	5292.9	5032.7	5146.3	10395.0
Uygunluk Değeri	XXX	0.998802	0.998739	0.99805	0.996112	0.9954013
İtersyon Sayısı	XXX	100000	100000	100000	100000	100000
3 Gün						
	4 Oturum			6 Oturum		
	4 Salon	5 Salon	6 Salon	4 Salon	5 Salon	6 Salon
İlk Ceza Değeri	2317255.6	6270035.1	10162089.3	10182039	16429725.6	21960374
En İyi Kromozom Ceza Değeri	6150.5	6471.0	14726.2	13293.6	34599.5	41250.8
Uygunluk Değeri	0.99442	0.995246	0.995747	0.99496	0.994733	0.994165
İtersyon Sayısı	100000	100000	100000	100000	100000	100000

Çizelge 6.3 bünyesinde gösterilen XXX ifadesi konferans kapasitesinin bildiri sayısını karşılamadığı için çizelgelemenin yapılamadığını belirtmektedir. Burada en iyi ceza değerine ait olan kromozom uygunluk değerleri gösterilmiştir. Denklem 5.3 ile belirtildiği gibi en iyi kromozomun ceza değerinin içinde bulunduğu popülasyonun toplam ceza değeri belirlenebilmektedir. Çizelge 6.4 ile 300 bildirinin Epigenetik Algoritma ile çizelgelenmesi gösterilmiştir.

Çizelge 6.4. 300 bildirinin epigenetik algoritma ile çizelgelenmesi.

300 Bildiri						
2 Gün						
	4 Oturum			6 Oturum		
	4 Salon	5 Salon	6 Salon	4 Salon	5 Salon	6 Salon
İlk Ceza Değeri	XXX	XXX	XXX	XXX	420395.7	6121400.0
En İyi Kromozomun Ceza Değeri	XXX	XXX	XXX	XXX	8479.8	14071.9
Uygunluk Değeri	XXX	XXX	XXX	XXX	0.991165	0.9861815
İterasyon Sayısı	XXX	XXX	XXX	XXX	100000	100000
3 Gün						
	4 Oturum			6 Oturum		
	4 Salon	5 Salon	6 Salon	4 Salon	5 Salon	6 Salon
İlk Ceza Değeri	XXX	444021	5919811.5	6044143.4	14875378.5	23758770.7
En İyi Kromozomun Ceza Değeri	XXX	9371.0	15478.2	15920.6	37197.1	49812.0
Uygunluk Değeri	XXX	0.9897001	0.987693	0.984691	0.975718	0.974778
İterasyon Sayısı	XXX	100000	100000	100000	100000	100000

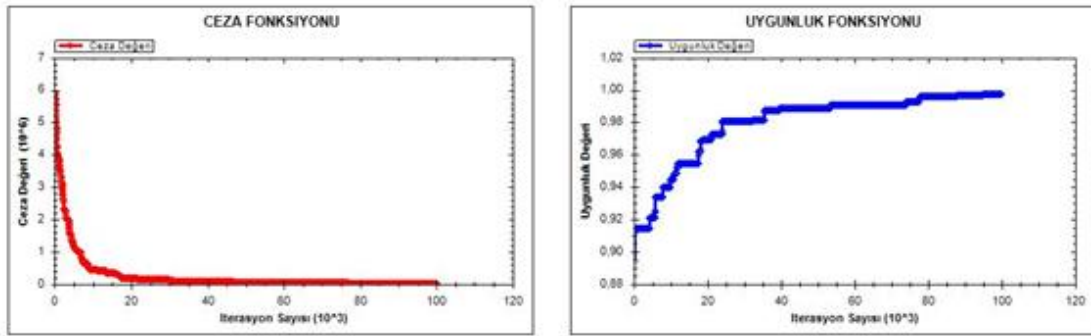
Çizelge 6.4 bünyesinde gösterilen XXX ifadesi konferans kapasitesinin bildiri sayısını karşılamadığı için çizelgelemenin yapılamadığını belirtmektedir. Şekil 6.9 ile 200 bildirili bir konferansın 2 gün, 6 oturum, 6 salon eşliğinde Epigenetik Algoritma ile çizelgelenmesinin sonucunda elde edilen ceza fonksiyonu ve uygunluk fonksiyonu gösterilmiştir.



Şekil 6.9. 200 bildiri, 2 gün, 6 oturum, 6 salon epigenetik algoritma.

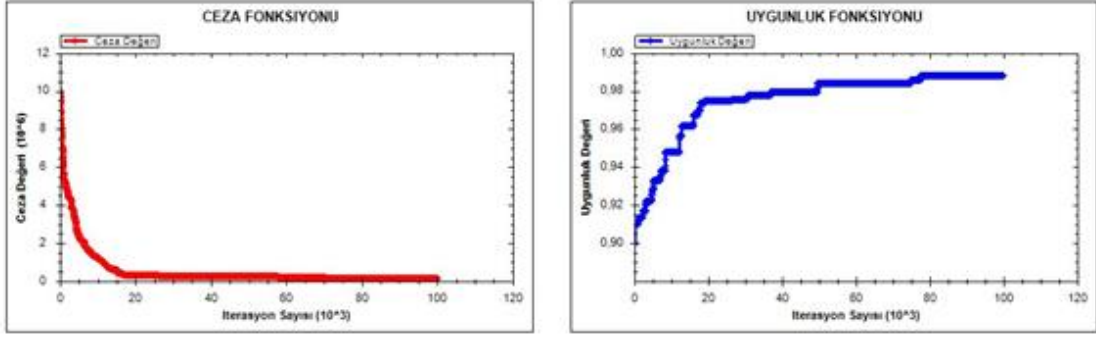
Ceza fonksiyonun 0'a ulaşması hedeflenmektedir. Uygunluk fonksiyonun amacı 1.0' a ulaşmaktır. Epigenetik Algoritma çalıştırıldığında durdurmak için bazı kriterler bulunmaktadır. Bunlar hedef ceza değeri, hedef uygunluk değeri ve iterasyon sayısıdır.

Bu tez için yapılan deneylerde iterasyon sayısı durdurma kriteri olarak belirlenmiştir. 100 bin iterasyona ulaşıldığında algoritma sonlandırılmıştır. 100 bin iterasyon sonucu elde edilen ceza değerleri ve uygunluk değeri gösterilmiştir. Şekil 6.9 ile 200 bildiri 2 gün, 6 oturum ve 6 salon eşliğinde çizelgelenmiştir. Ceza değeri yaklaşık 1000000'den başlayarak logaritmik olarak azalmıştır. 100 bin iterasyon sonucu ceza değeri Çizelge 6.1 ile belirtildiği üzere 10395'e düşmüştür. Şekil 6.10 ile 300 bildirili bir konferansın 2 gün, 6 oturum, 6 salon eşliğinde Epigenetik Algoritma ile çizelgelenmesinin sonucunda elde edilen ceza fonksiyonu ve uygunluk fonksiyonu gösterilmiştir.



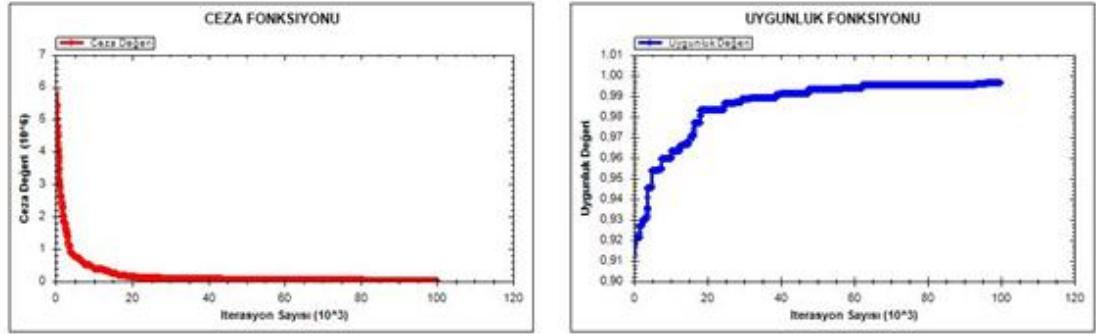
Şekil 6.10. 300 bildiri, 2 gün, 6 oturum, 6 salon epigenetik algoritma.

300 bildiri bulunan bu senaryoda ceza değeri 14071'e düşmüştür. Çizelge 6.3 ve Çizelge 6.4 ile belirtildiği gibi 300 bildiri ile yapılan çizelgeleme 200 bildiri ile yapılan çizelgelemeden daha yüksek ceza değeri elde etmiştir. Bu durum bildiri sayısındaki artışın çizelgeleme performansını olumsuz etkilediğini vurgulamaktadır. Ayrıca Şekil 6.2 ile belirtilen 300 bildirinin 2 gün 6 oturum 6 salon eşliğinde Genetik Algoritma ile yapılan çizelgelemesinde elde edilen ceza değeri aynı durumdaki Epigenetik Algoritma ile yapılan ceza değerinden daha yüksektir. Bu durum grafiklerdeki eğimden de tespit edilebilmektedir. Şekil 6.11 ile 200 bildirili bir konferansın 3 gün, 4 oturum, 6 salon eşliğinde Epigenetik Algoritma ile çizelgelenmesinin sonucunda elde edilen ceza fonksiyonu ve uygunluk fonksiyonu gösterilmiştir.



Şekil 6.11. 200 bildiri, 3 gün, 4 oturum, 6 salon epigenetik algoritma.

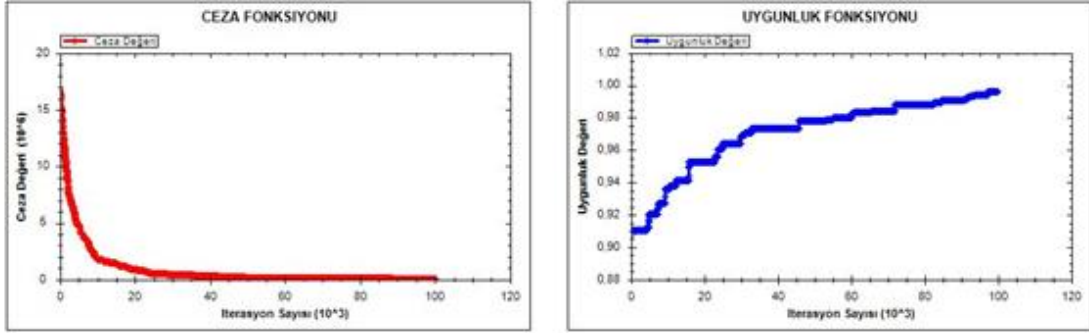
Ceza fonksiyonu 14726 ceza değerine düşmüştür. Ceza fonksiyonu 2 gün 6 oturum ve 6 salon eşliğinde çizelgelenen durum ile benzerlik göstermektedir. Çünkü bu iki durum aynı gen sayısı ile ifade edilmektedir. Bu iki durumda kromozom içinde bulunan boş zaman sayıları da eşittir. 200 bildirinin 3 gün, 4 oturum ve 6 salon eşliğinde Epigenetik Algoritma ile çizelgelenmesi Genetik Algoritma ile çizelgelenmesinden daha düşük bir ceza değerine ulaşmıştır. Şekil 6.12 ile 300 bildirili bir konferansın 3 gün, 4 oturum, 6 salon eşliğinde Epigenetik Algoritma ile çizelgelenmesinin sonucunda elde edilen ceza fonksiyonu ve uygunluk fonksiyonu gösterilmiştir.



Şekil 6.12. 300 bildiri, 3 gün, 4 oturum, 6 salon epigenetik algoritma.

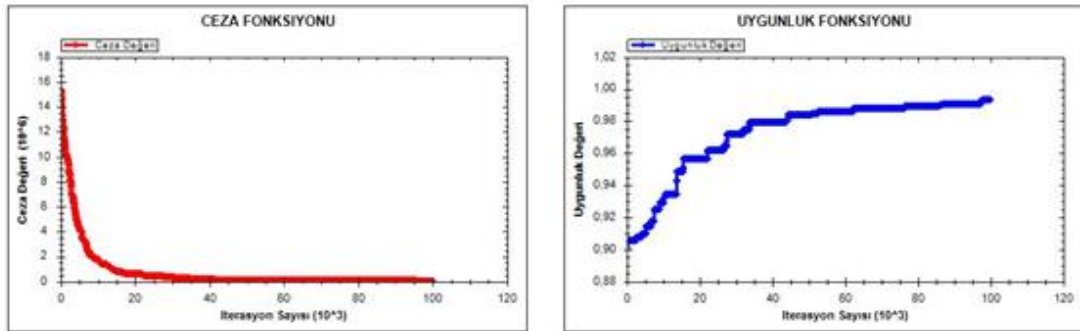
Çizelge 6.3 ile belirtildiği gibi ceza fonksiyonu 15478 ceza değerine düşmüştür. Ceza fonksiyonu 2 gün 6 oturum ve 6 salon eşliğinde çizelgelenen durum ile benzerlik göstermektedir. Çünkü bu iki durum aynı gen sayısı ile ifade edilmektedir. Bu iki durumda kromozom içinde bulunan boş zaman sayıları da eşittir. 300 bildirinin 3 gün 4 oturum 6 salon eşliğinde çizelgelenmesi 200 bildirinin 3 gün 4 oturum 6 salon ile çizelgelenmesinden daha yüksek ceza değerine sahiptir. Bu durum bildiri sayısının artmasının çizelgemeyi daha güç hale getirdiğini ortaya koymaktadır. Şekil 6.5 ile 200 bildirili bir konferansın 3 gün, 6 oturum, 5 salon eşliğinde Epigenetik Algoritma ile

çizelgelenmesinin sonucunda elde edilen ceza fonksiyonu ve uygunluk fonksiyonu gösterilmiştir.



Şekil 6.13. 200 bildiri, 3 gün, 6 oturum, 5 salon epigenetik algoritma.

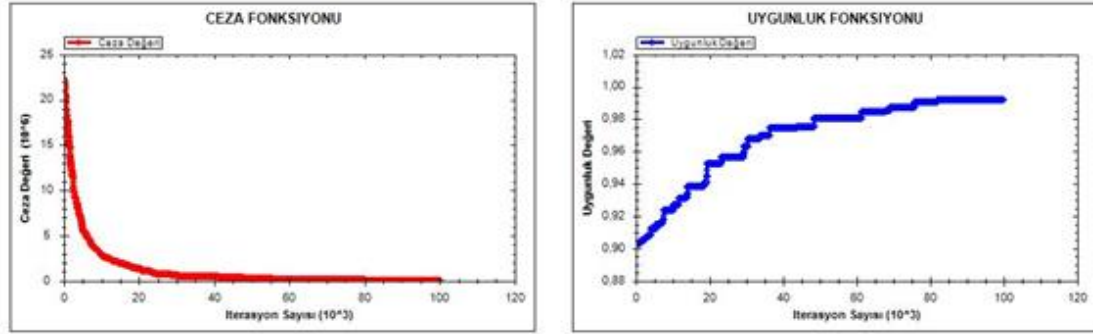
Çizelge 6.3 ile belirtildiği üzere elde edilen ceza değeri 34599 olmuştur. Elde edilen bu ceza değeri 3 gün 4 oturum 6 salon eşliğinde yapılan çizelgelemeden daha yüksektir. Bu durum çizelgelemenin kapasite arttığında elde edilen ceza değerinin daha yüksek olduğunu ortaya koymaktadır. Grafik üzerinden ve Çizelge 6.3 üzerinden anlaşılacağı üzere 200 bildiri 3 gün, 6 oturum, 5 salon ile çizelgelendiğinde ilk ceza değeri 3 gün, 4 oturum 6 salon ile yapılan çizelgelemeden daha büyüktür. 3 gün 6 oturum 5 salon çözüm uzayında daha büyük bir alanda arama yaptığı için optimizasyon bakımından daha yüksek bir iyileşme göstermiştir. Ancak elde edilen en küçük ceza değeri bakımından 3 gün 4 oturum 6 salon daha iyi bir sonuç üretmiştir. Şekil 6.14 ile 300 bildirili bir konferansın 3 gün, 6 oturum, 5 salon eşliğinde Epigenetik Algoritma ile çizelgelenmesinin sonucunda elde edilen ceza fonksiyonu ve uygunluk fonksiyonu gösterilmiştir.



Şekil 6.14. 300 bildiri, 3 gün, 6 oturum, 5 salon epigenetik algoritma.

Çizelge 6.1 ile belirtildiği gibi 300 bildiri 3 gün 6 oturum 5 salon eşliğinde

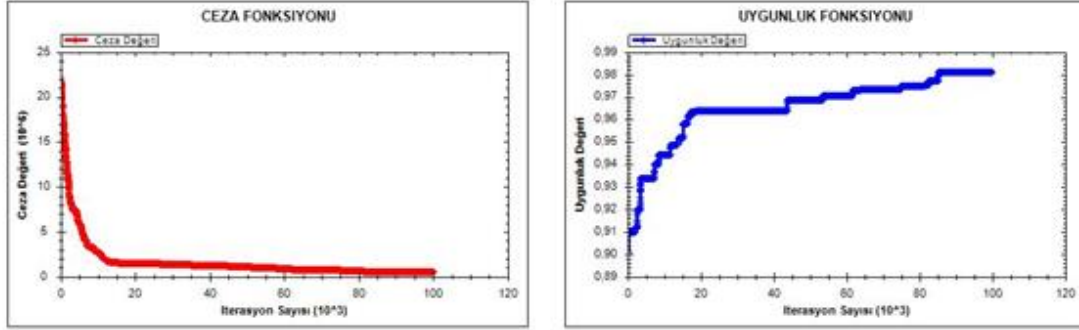
çizelgelendiğinde 37197 ceza değerine ulaşmaktadır. Bu durum Şekil 6.13 ile gösterilen 200 bildirinin çizelgelendiği durumdan daha yüksek bir ceza değerine ulaştığını göstermektedir. Bu durum bildiri sayının artmasının çizelgeleme performansını olumsuz etkilediğini ortaya koymaktadır. Şekil 6.15 ile 200 bildirili bir konferansın 3 gün, 6 oturum, 6 salon eşliğinde Epigenetik Algoritma ile çizelgelenmesinin sonucunda elde edilen ceza fonksiyonu ve uygunluk fonksiyonu gösterilmiştir.



Şekil 6.15. 200 bildiri, 3 gün, 6 oturum, 6 salon epigenetik algoritma.

Çizelge 6.3 ile gösterildiği gibi bu durumda elde edilen en küçük ceza değeri 41250 olmaktadır. Elde edilen bu ceza değeri 200 bildirinin 3 gün 6 oturum 5 salon ile çizelgelendiği durumdan daha yüksek bir ceza değerine sahiptir. Bu durum kapasite artırmanın çözüm uzayındaki büyümeye neden olmasından dolayı çözümü zorlaştırmaktadır. 200 bildirinin farklı çizelgelenmelerdeki durumu ele alındığında en yüksek ceza değerine 3 gün 6 oturum 6 salon eşliğinde çizelgelenmesindeki durum olduğu gözlenmiştir. Çünkü bu durum 200 bildiri için en büyük çözüm uzayına sahip olmaktadır. En büyük çözüm uzayında en iyi çözüme ulaşmak oldukça zor olacaktır. Ayrıca 200 bildirinin 3 gün 6 oturum 6 salon eşliğinde Epigenetik Algoritma ile çizelgelenmesi 200 bildirinin 3 gün 6 oturum 6 salon eşliğinde Genetik Algoritma ile çizelgelenmesinden daha düşük ceza değerine ulaştığı gözlenmiştir. Bu durum Epigenetik Algoritmanın daha iyi performans sergilediğini göstermektedir. 200 bildirinin 3 gün 6 oturum 6 salon ile çizelgelenmesi, 200 bildirinin 3 gün 6 oturum 5 salon ile çizelgelenmesi ve 200 bildirinin 2 gün 6 oturum 6 salon ile çizelgelenmesi durumları ile karşılaştırıldığında elde edilen en küçük ceza değerinin değiştiği gözlenmektedir. Bu üç durum ele alındığında kapasitenin daha küçük olduğu 2 gün 6 oturum 6 salon ile yapılan çizelgemenin en iyi çözüme daha yakın olduğunu ortaya koymaktadır. Bu durum kapasite kaynaklarındaki artışın çizelgeleme performansını

kademeli olarak zorlaştırdığı ortaya koymaktadır. Şekil 6.16 ile 300 bildirili bir konferansın 3 gün, 6 oturum, 6 salon eşliğinde Epigenetik Algoritma ile çizelgelenmesinin sonucunda elde edilen ceza fonksiyonu ve uygunluk fonksiyonu gösterilmiştir.

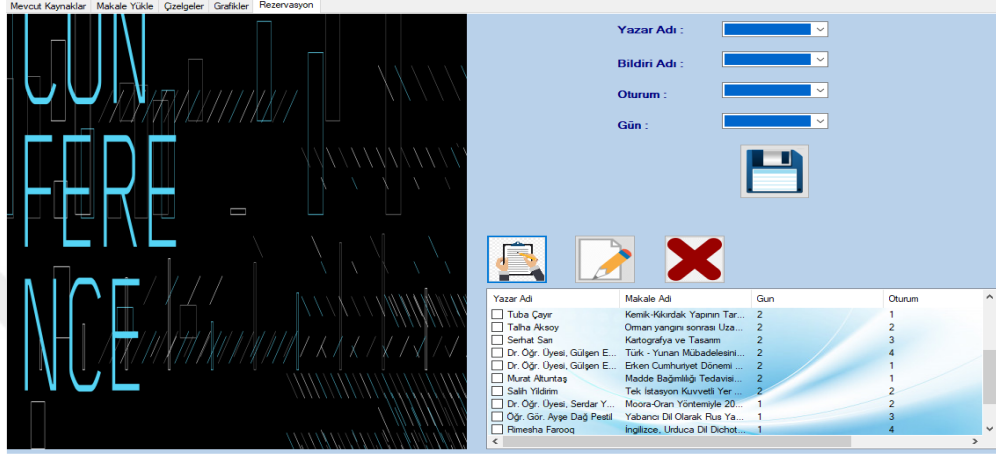


Şekil 6.16. 300 bildiri, 3 gün, 6 oturum, 6 salon epigenetik algoritma.

Çizelge 6.4 ile gösterildiği gibi bu durumda ceza değeri 49812 olarak belirlenmiştir. Bu durum 200 bildirinin 3 gün 6 oturum 6 salon eşliğinde yapılan çizelgelemeden daha büyük olduğunu göstermektedir. Bu durum bildiri sayısı arttıkça çizelgelenmenin zorlaştığını ortaya koymaktadır. Ayrıca 300 bildirinin 3 gün 6 oturum 6 salon ile çizelgelenmesi, 3 gün 6 oturum 5 salon eşliğinde çizelgelenmesinden daha zor olmuştur. İlk iterasyonda üretilen ceza değeri 3 gün 6 oturum 6 salon durumunda daha fazladır. Bu durum çözüm uzayının büyüdüğünü göstermektedir. 3 gün 6 oturum 6 salon ilk iterasyon bakımından daha büyük bir iyileşme göstermesine rağmen algoritma onunda elde edilen çözüm 3 gün 6 oturum 5 salon ile elde edilen sonucun en iyi çözüme daha yakın olduğunu ortaya koymaktadır. 300 bildirinin 3 gün 6 oturum 6 salon eşliğinde çizelgelenmesi, 300 bildirinin 2 gün 6 oturum 6 salon eşliğinde çizelgelenmesinden de zor olmaktadır. Çünkü 3 gün 6 oturum 6 salon eşliğinde yapılan çizelgelenmenin çözüm uzayı daha büyüktür. Daha büyük çözüm uzayında en iyi çözüme olan yakınsama daha güç olmaktadır. Bu durum elde edilen en küçük ceza değerine de yansımaktadır. Ayrıca 300 bildirinin 3 gün 6 oturum 6 salon ile çizelgelenmesinin 200 bildirinin 3 gün 6 oturum 6 salon ile çizelgelenmesinden daha zor olduğu görülmektedir. Bildiri sayısındaki artış çizelgeleme performansını etkilemektedir. 300 bildirinin farklı durumlardaki çizelgemelerdeki durumun incelendiğinde en yüksek ceza değerine 3 gün 6 oturum 6 salon eşliğinde olduğu belirlenmiştir. Bu durum çözüm uzayının en büyük olduğu durumdur. Aynı iterasyon sayısında daha büyük bir çözüm uzayında arama yapara en iyi çözümü bulmayı güçleştirmektedir.

6.1. REZERVASYON İŞLEMİNİN PROBLEM ÇÖZÜMÜNE ETKİSİ

Rezervasyon, bir çizelge programı yapılmaya başlamadan önce belirli yerlerin işaretlenmesidir. Konferans çizelgeleme işlemi için rezervasyon belirli gün ve saat dilimi için belirli kişilerin atanmasıdır. Şekil 6.17 ile geliştirilen programdaki Rezervasyon Kayıt Ekranı gösterilmiştir.



Şekil 6.17. Rezervasyon kayıt ekranı.

Bu bölümde çizelgeleme yapılmadan önce rezervasyonların kaydedilmesi sağlanmaktadır. Rezervasyonu yapacak olan organizatör, konferansın hangi gün, hangi oturumda gerçekleşeceğini belirterek kaydetmektedir. Geliştirilen program hangi salon olacağını belirlemektedir. Genetik Algoritmanın popülasyon sayısındaki artış, iterasyon sayısındaki artış, çaprazlama oranındaki artış ve mutasyon oranındaki artış hiçbir biçimde kaydedilen bu rezervasyonu etkilememektedir. Aynı oturuma yapılacak aynı yazara ait başka bildirinin sunumu eğer rezervasyonla yapılacaksa hemen aynı oturuma eklenir. Başka bir yazara ait aynı oturum istenirse başka bir salondaki oturuma aktarılır. Bu sayede sisteme daha önce kaydedilen rezervasyonun kaybolmaması sağlanmış olur.

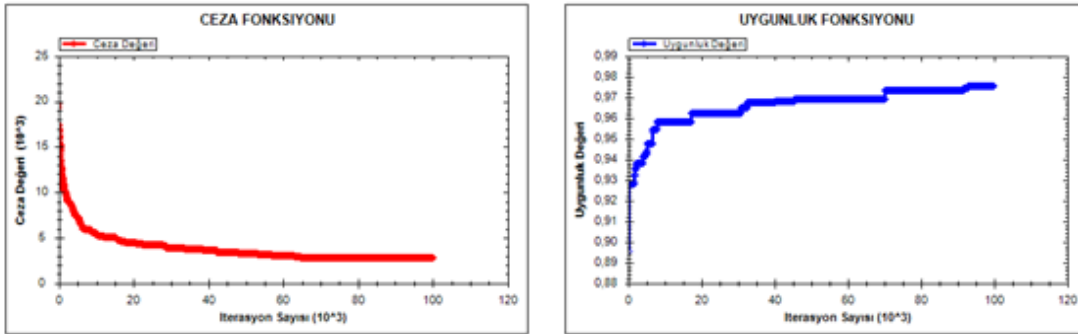
Çizelgeleme kombinatorial bir problem olduğundan dolayı çözüm uzayında rezervasyon ile belirli noktaları işaretlemek çözümün gidişatını etkilemektedir. Bu durum belirli stokastik süreçlerin olduğu algoritmalarda çözümün en iyiyi arama işleminde bazı engellere sevk etmektedir. Örneğin ilerleyen iterasyonlarda rezervasyonun olduğu bir oturumda değişiklik yapılarak oturumdaki tüm konuların aynı olması amaçlandığında rezervasyondan kaynaklı en iyi oturum sağlanamamış olur. Örneğin bir oturumda 4 tane aynı konudan çalışma var iken 1 tane farklı konudan çalışma olduğundan 1 tane çalışmanın gerekli işlemlerle yer değişmesi gerekmektedir.

Ancak bu 1 çalışma eğer rezervasyona ait bir çalışma ise bu oturumu en iyiye taşımak için diğer 4 çalışmayı değiştirmesi gerekmektedir. Çizelge 6.5. 200 bildirinin Genetik Algoritma ile rezervasyonlu ve rezervasyonsuz çizelgelenmesine ait olan değerler gösterilmiştir.

Çizelge 6.5. 200 bildirinin genetik algoritma ile rezervasyonun incelenmesi.

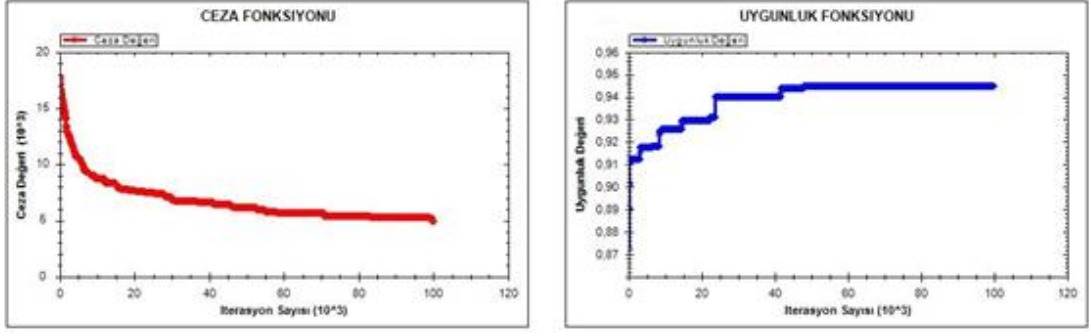
Bildiri Sayısı	200	200
Popülasyon Sayısı	10	10
Rezervasyon Sayısı	20	0
Kromozom Uzunluğu	200	200
İlk Ceza Değeri	18492.0	18517.9
En İyi Kromozom Ceza Değeri	5640.3	2955.3
Uygunluk Değeri	0.95873142	0.97926221
İterasyon Sayısı	100000	100000

Şekil 6.18 ile Genetik Algoritma ile 200 bildirinin 2 gün 4 oturum 5 salon ile çizelgelenmesi rezervasyon olmadan gerçekleştirilmiştir.



Şekil 6.18. 200 bildiri, 2 gün, 4 oturum, 5 salon genetik algoritma.

Şekil 6.19 ile Genetik Algoritma ile 200 bildirinin 2 gün 4 oturum 5 salon ile 20 tane rezervasyon işlemi yapıldıktan sonra çizelgelenmiştir.



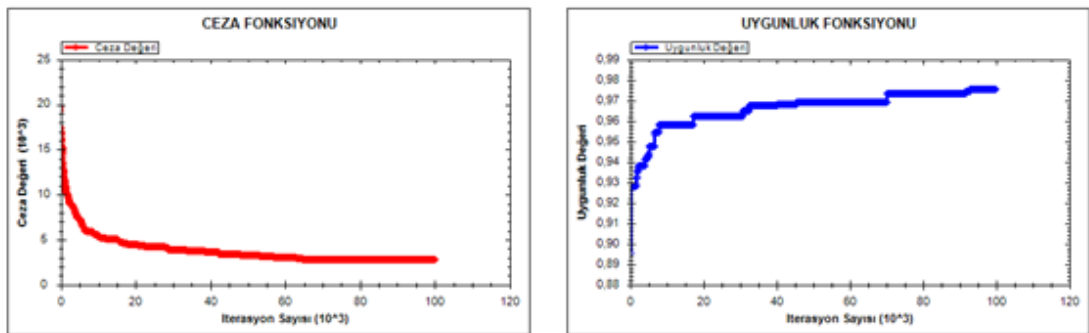
Şekil 6.19. 200 bildiri, 2 gün, 4 oturum, 5 salon genetik algoritma.

20 tane rezervasyon eklenmesi çizelgeleme işlemini güçleştirmiştir. Bu durum Çizelge 6.5 ile belirtildiği üzere rezervasyonlu durumlarda Genetik Algoritmanın rezervasyonsuz duruma göre daha yüksek ceza değerine ulaşması ile anlaşılmaktadır. Rezervasyon işlemi çizelgeleme performansını olumsuz etkilemiştir. Çizelge 6.6. 200 bildirinin Epigenetik Algoritmaya ait rezervasyonlu ve rezervasyonsuz çizelgenmesi sonucu elde edilen sonuçlar belirtilmiştir.

Çizelge 6.6. 200 bildirinin epigenetik algoritma ile rezervasyonun incelenmesi.

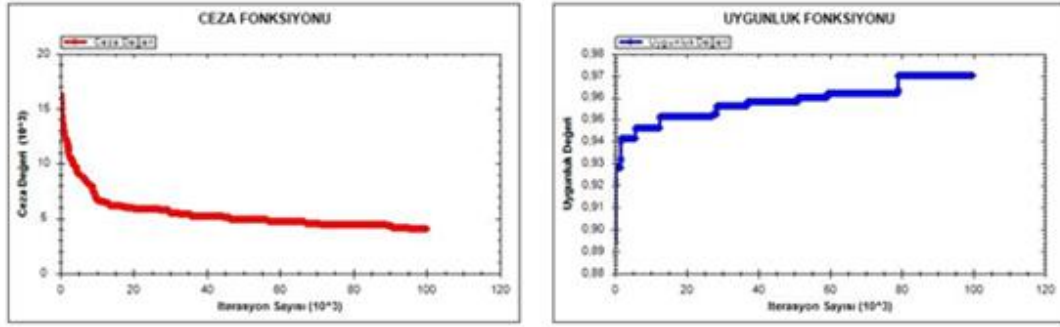
Popülasyon Sayısı	10	10
Rezervasyon Sayısı	20	0
Kromozom Uzunluğu	200	200
İlk Ceza Değeri	19258.7	19171.3
Son Ceza Değeri	4420.4	2520.1
Uygunluk Değeri	0.9721524	0.9846121
İterasyon Sayısı	100000	100000

Şekil 6.20 ile Epigenetik Algoritma ile 200 bildirinin 2 gün 4 oturum 5 salon ile çizelgenmesi rezervasyon olmadan gerçekleştirilmiştir.



Şekil 6.20. 200 bildiri, 2 gün, 4 oturum, 5 salon epigenetik algoritma.

Şekil 6.21 ile Epigenetik Algoritma ile 200 bildirinin 2 gün 4 oturum 5 salon ile çizelgelenmesi 20 tane rezervasyon işlemi yapıldıktan sonra çizelgelenmiştir.



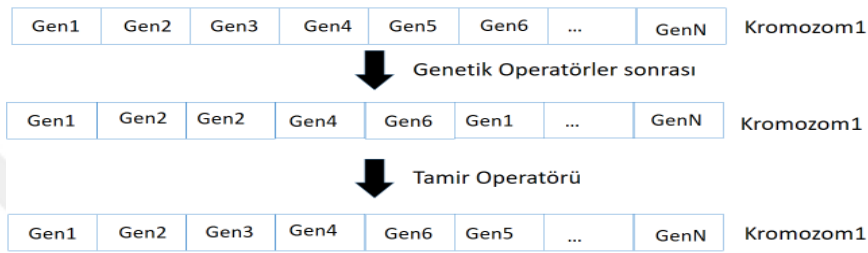
Şekil 6.21. 200 bildiri, 2 gün, 4 oturum, 5 salon epigenetik algoritma.

Çizelge 6.5 üzerinde de görüldüğü gibi 2 gün 4 oturum, 5 salon olan senaryoda algoritmalar konferansta rezervasyon olmaması durumu ile 20 tane rezervasyon olması durumu kıyaslandığında çözüme yaklaşmakta zorlandığı gözlenmektedir. Genetik Algoritma 2 gün 4 oturum, 5 salon için rezervasyon 20 iken 5640.3 ceza değerine, rezervasyon yok iken ise 2955.3 ceza değerine ulaşmıştır. Benzer biçimde Epigenetik Algoritma rezervasyon 20 tane iken 4420.4 ceza değerine, rezervasyonsuz durumda ise 2520.1 ceza değerine ulaşmıştır. Bu durum rezervasyonun çözüme olan etkisinin olumsuz olduğunu ortaya koymaktadır. Rezervasyon olmaması algoritmaları arama uzayında yalnızca problem kısıtlarına odaklanmasını sağlamaktadır. Ancak probleme rezervasyonun dahil edilmesi algoritmalara problem tanımında ifade edilen kısıtların haricinde rezervasyonların da birer kısıt olarak ele alınmasına neden olacaktır. Bu durum hedeflenen en düşük ceza değeri, hedeflenen en yüksek uygunluk değeri amaçlarına ulaşmada çözümün yönlendirilmesine neden olmuştur. Bu durumda çizelgeleme problemlerinde ve özellikle kısıtların sert ve karmaşık olduğu problemlerde çözümü olumsuz etkilediği gözlenmiştir. Hem Genetik Algoritmanın hem Epigenetik Algoritmanın rezervasyonlu çözümlerde zorlanması ve çözümün güçlenmesi bu durumu net bir biçimde ortaya koymaktadır.

6.2. TAMİR OPERATÖRÜNÜN ÇÖZÜME ETKİSİ

Konferans çizelgeleme gibi kısıtlı optimizasyon problemlerinde çaprazlama ve mutasyon operatörleri uygulandıktan sonra kromozom yapılarında bazı istenmeyen durumlar oluşmaktadır. Bu istenmeyen durumların temel sebebi gen sayıları ve gen

kaybolmaları gibi deęişimlerdir. Bu durum çözüm uzayında temsil edilmeyen noktalara doğru ilerlemeye neden olacaktır. Böyle bir durumda tamir operatörü kullanmak kaçınılmaz olmaktadır [97]. Tamir operatörü genetik operatörler uygulandıktan sonra çözüm için uygun olmayan kromozomları istenilen bilgilere göre düzenleme yaparak uygun duruma getirecektir. Bu düzenleme probleme özel olarak tasarlanmalıdır. Eğer tamir operatörü kullanılmazsa Genetik Algoritma ve Epigenetik Algoritma verimli çalışmayacak ve çözüm uzayında farklı noktalara eğilim gösterilerek en iyi çözüm elde edilemeyecektir [98]. Şekil 6. 22 ile çaprazlama ve mutasyon işlemleri sonrası kromozom durumu ve tamir operatörü gösterilmiştir.

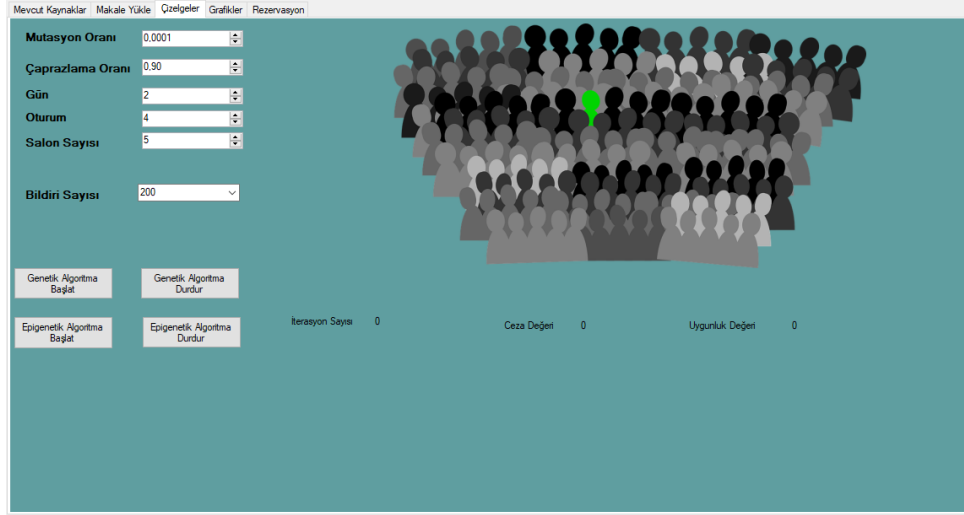


Şekil 6.22. Tamir operatörünün etkisi.

Bu örnekte görüldüğü üzere genetik operatörler sonrasında oluşan kromozom1 üzerinde bazı genler çözüm uzayını temsil etmeyecek şekilde dizilmiştir. Bu durum bazı gen bilgilerinin kaybolması ve beklenmeyen bir dizilim doğmasına neden olmuştur. Kromozom1 üzerinde gen3 ve gen5 bilgisi kaybolmuştur, gen1 ve gen2 bilgisi tekrar etmiştir. Tamir operatörü kullanılarak kaybolan gen3 ve gen5 çözüm uzayına tekrar dahil edilmektedir. Tamir operatörü ile her nesilde genetik operatörler sonucu bozulma olup olmadığı kontrol edilmelidir. Eğer bu işlem düzeltilmezse algoritma sonlandırıldığında problem çözümünü temsil etmeyen bir çözüm sunulacaktır. Bunun önüne geçilmesi için her nesilde tamir operatörü uygulanmalıdır. Bu sayede beklenmeyen dizilimlerinin önüne geçilmesi sağlanmaktadır.

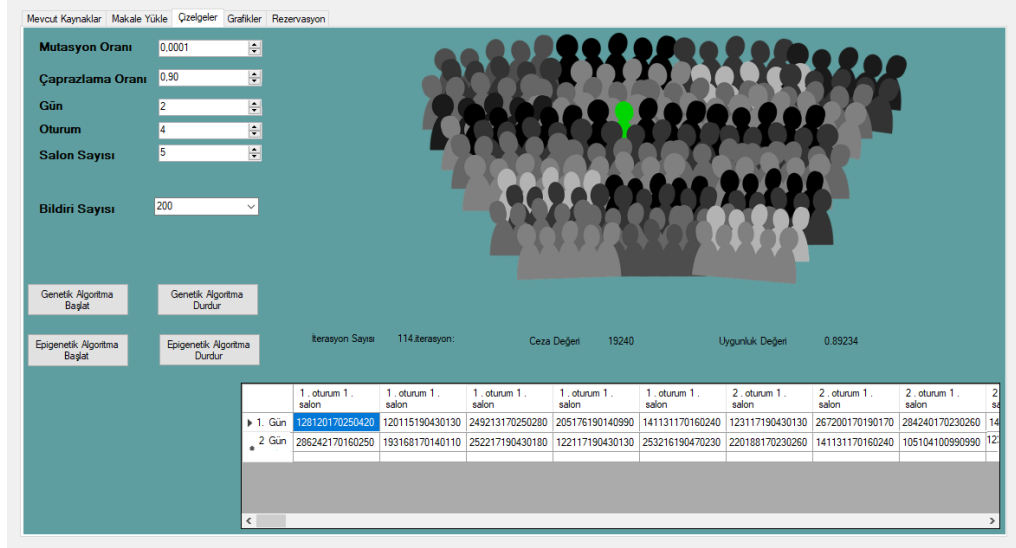
6.3. YAZILIM PROGRAMININ İNCELENMESİ

Genetik Algoritma ve Epigenetik Algoritma kullanılarak geliştirilen bu paket program yazılımında son kullanıcı Şekil 6.23 üzerinde görüldüğü gibi konferans çizelgeleme yapabilmek için karar deęişkenlerine ait gerekli bilgileri girmelidir.



Şekil 6.23. Çizelgeleme giriş bilgilerinin yüklenmesi.

Şekil 6.23 ile gösterilen giriş ekranında son kullanıcı konferans çizelgeleme probleminin en önemli karar değişkenleri arasında bulunan gün, oturum sayısı ve salon sayısını belirlemektedir. Çizelgelemenin her iki algoritma ile gerçekleştirilebilmesi için gün, oturum ve salon sayısı konferanstaki bildiri sayısından büyük ya da eşit olmak durumundadır. Aksi takdirde bu iki algoritma uygulanamayacaktır. Ayrıca çaprazlama ve mutasyon oranı da son kullanıcı tarafından dinamik bir biçimde ayarlanabilmektedir. Gerekli karar değişkenleri bilgileri girildikten sonra program Genetik Algoritma veya Epigenetik Algoritma için çalıştırılır. Veritabanına kaydedilen tüm bildiriler parametre kodlanması yardımıyla genlere dönüştürülür. Genlerin oluşturulmasının ardından kromozomlar ve daha sonra popülasyonlar oluşturulur. Böylece algoritmaların uygulanabilmesi için gerekli tanımlamalar tamamlanmış olur. Şekil 6.24 ile görüldüğü gibi program başlatılmış ve belli bir iterasyon değeri için durdurulmuştur.



Şekil 6.24. Çizelgeleme giriş bilgilerinin yüklenmesi.

Şekil 6.24 ile datagridview üzerinde gösterilen kromozomlar problemin mevcut iterasyondaki en iyi kromozomuna aittir. Algoritma sonlandırıldığında en iyi kromozom datagridview üzerinde gösterilmektedir. Bu datagridview üzerinde gösterilen ve 15 bit ile ifade edilen gen son kullanıcı için açıklayıcı olmamaktadır. Ayrıca bu datagridview üzerinde yer alan bilgiler son kullanıcının kavrayıp üzerinde karar alabileceği bir bilgi olmamaktadır. Bu bakımdan geliştirilen program ile problemin en başında belirtilen yazar isimleri, bildiri isimleri, gün, oturum ve saat bilgilerine ilişkin bir doküman hazırlanması gerekmektedir. Geliştirilen program sonlandırıldığında datagridview üzerinde en iyi kromozoma ait gen bilgileri yer almaktadır. Bu gen bilgileri son kullanıcının doğrudan işlem yapabileceği bilgi değildir. Bu bilgilerin tekrar açıklayıcı, anlamlı ve üzerinde karar alınabilir bir yapıya döndürülmesi gerekmektedir.

Şekil 6.25 ile program sonlandırıldığında elde edilen en iyi kromozom bilgilerinin anlamlı bilgilere nasıl dönüştüğü gösterilmiştir.

	1. oturum 1. salon	1. oturum 1. salon	1. oturum 1. salon	1. oturum 1. salon	1. oturum 1. salon	2. oturum 1. salon	2. oturum 1. salon	2. oturum 1. salon
1. Gün	140130170110380	195169170110380	225192170110120	273230170110380	206177170110100	123117190430130	265211190600140	104103100990990
2. Gün	288240390	283239170230130	105104100990990	159148170140210	180162170230270	299253190630160	115112190380230	295249170250470

140	130	170	110	380
Oğuzcan Uludağ	1-Veri Madenciliğinde Altman Z-Skor Finans Modeli Kullanılarak K-Means Kümeleme Metodunun Uygulanması	Mühendislik Temel Alanı	Bilgisayar Bilimleri ve Mühendisliği	Veri Madenciliği

Şekil 6.25. Gen bilgilerinin açıklanması.

Elde edilen bu bilgiler ile artık son kullanıcının faydalanabileceği bilgiler üretilmiştir. Konferansa ait tüm bildirilerin nerede ve ne zaman çözeleneceğini gösteren konferans kitapçığı oluşturulmaktadır. Şekil 6.26 ile program sonlandırıldığında ortaya çıkan konferans kitapçığından bir bölüm sunulmuştur.

Gün	Saat	Salon	Bildiri Adı	Yazar
1. Gün	1. Oturum 9.00-10.00	Salon A	1-Veri Madenciliğinde Altman Z-Skor Finans Modeli Kullanılarak K-Means Kümeleme Metodunun Uygulanması	Oğuzcan Uludağ
1. Gün	1. Oturum 9.00-10.00	Salon A	2-Metin Sınıflandırma İçin Küme Yarıllarından Oluşan Bir Türkçe Veri Seti	Öz. Gör. Mehmet Fatih Karaca
1. Gün	1. Oturum 9.00-10.00	Salon A	3-Yapay İsl İşlem Algoritmalarına Dayalı Dilemlemeli Yerleşim İkili Ağacı İçin Meta-Sezgisel Bir Yöntem	Ali Ammar Hıral
1. Gün	1. Oturum 9.00-10.00	Salon A	4- Destek Vektör Makineleri ile Aritmi Sınıflandırılması	Süleyman Çetinkaya
1. Gün	1. Oturum 9.00-10.00	Salon A	5-RCS Tabanlı Robotlar İçin Çözülme ile Otonom Hedef Tespiti Yapan Android Uygulama	Muhammet Ali Yüce

Şekil 6.26. Program çıktısı olan konferans kitapçığı.

Genetik Algoritma ve Epigenetik Algoritma ile veri tabanına kaydedilen tüm bildirilerin popülasyon, kromozom, gen biçiminde ifade edilmesi ile sağlanmıştır. Bu ifadeleme işleminin ardından problem kısıtları ile Genetik Algoritma ve Epigenetik Algoritma çalıştırılmıştır. Problem kısıtları ile algoritmalar sonlandırıldıktan sonra popülasyon, kromozom, gen ifade edilmesi tekrardan son kullanıcıya hitap edecek biçimde dönüştürülmüştür. Son kullanıcı için yazar adı, bildiri adı bilgileri gün, oturum ve salon

belirlenmiş şekilde oluşturulmaktadır. Bu durum bir word dökümanı oluşturularak sağlanmıştır. Bu word dökümanı son kullanıcı için programın en önemli çıktısı olmaktadır. Word dökümanı konferans kitapçığını ifade etmektedir. Konferans kitapçığı ile hangi yazarın hangi çalışmasını hangi gün, hangi saat ve hangi salonda sunacağı bilgisi oluşturulmaktadır.



7. SONUÇLAR VE ÖNERİLER

Bu tez kapsamında, multidisipliner konferansa ait gerçek verilerden yola çıkılarak konferans çizelgeleme problemi Genetik Algoritma ve Epigenetik Algoritma ile gerçekleştirilmiştir. Konferans çizelgeleme probleminde gün, oturum ve salon sayılarına ait karar değişkenleri çizelgeleme performansını etkilemektedir. Bir konferans çizelgelemenin en iyi biçimde gerçekleştirilebilmesi için gün, oturum sayısı ve salon sayısı arasındaki sayının konferans bildiri sayısı ile arasındaki dengenin göz önüne alınarak hazırlanması gerekmektedir. Bu işlem dinamik bir çizelgeleme çözümünün önemli bir faktörüdür. Bu durum günümüz koşullarındaki konferansın ihtiyaçlarının dinamik bir biçimde değişmesine olumlu bir tepki verecektir.

Epigenetik Algoritma ile çözümü gerçekleştirilen bu tezde klasik çaprazlama ve klasik mutasyon operatörlerinin haricinde ilaveten uygulamaya yönelik olarak epiçaprazlama ve epimutasyon operatörleri Epigenetik Faktör Listesine göre uygulanmıştır. Bu sayede Epigenetik Algoritma ile daha iyi sonuçlar elde edilmiştir. Biyoloji alanından ve biyoteknoloji alanından ilham alınarak geliştirilen epigenetik yapının yazılım alanına ve özellikle optimizasyon problemlerinden kullanımı sağlanmıştır. Epigenetik yapının Genetik Algoritma ile bütünleşmesi sağlanarak Genetik Algoritmanın rastgelelik anlayışının kısmen yönetilmesini sağlamaktadır. Bu sayede daha kontrollü bir biçimde gen alışverişi sağlanarak daha iyi çözümlerin elde edilmesi sağlanmaktadır.

Tez kapsamında elde edilen deneysel sonuçlardan biri de boş zaman sayısıdır. Bir konferanstaki bildiri sayısı ile kapasitesi arttıkça boş zaman sayısı da artacaktır. Bu durumda bildiri sayısı sabit iken kapasite artırarak yapılan deneylerde ceza değerinin başlangıç değerleri daha büyük olduğu tespit edilmiştir. Kapasite artırımını çizelgelemenin daha geniş bir ölçekte yapılabilmesini sağlayabilmektedir. Ancak başlangıç ceza değeri daha yüksek olmaktadır. Bu durum uygulanan algoritmaların eniyileme için öncelik sırasını değiştirmektedir. Bildiri sayısı sabit iken kapasite arttığında hem Genetik Algoritma hem de Epigenetik Algoritma başlangıç ceza değeri daha büyük değerden başladığı için önce boş oturumları konferansın son günü son oturumuna doğru ötelemeye başlayacaktır. Ardından eş zamanlı oturumlarda aynı

yazara ait çalışma olup olmadığını, aynı oturumlarda bulunan çalışmaların birincil alt alanlarını, ikincil alt alanlarını ve üçüncül alt alanlarını aynı yapmaya başlayacaktır. Ayrıca kapasite sayının arttığı durumlarda başlangıç ceza değeri ile ulaşılan son ceza değeri kıyaslandığında en çok iyileşme kapasitenin çok olduğu durumlarda görülmektedir. Ancak bu iyileşme optimizasyon bağlamında olup son kullanıcı için avantaj sağlamamaktadır. Yazılımın son kullanıcıya en iyi hizmet verebilmesi algoritmaların elde ettiği iyileşme değerleri olmayacak, program durdurulduğunda elde edilen çözümün endüstride kullanılabilir olup olmadığı yargılanacaktır. Kapasite değişkenin parametreleri olan gün, oturum ve salon sayılarının belirlenmesi konferans çizelgeleme probleminin önemli karar değişkenlerini oluşturduğu gözlenmiştir. Bu bakımdan bu tez ile en iyi çizelge durumlarının bildiri sayısı ile kapasitenin aynı olduğu durumlarda elde edildiği gözlenmiştir.

Genetik Algoritma ve Epigenetik Algoritma problem çözümüne bir başlangıç popülasyonu ile başlamaktadır. Bu başlangıç popülasyonu rastgele olarak oluşturulmaktadır. Yapılan deneysel sonuçlarda ve elde edilen grafiklerde başlangıç popülasyonundan hemen sonraki iterasyonlarda ceza değerinin tüm senaryolarda düştüğü gözlenmiştir. Bu durum hem Genetik Algoritma hem de Epigenetik Algoritma için geçerlidir. Bu durum başlangıç popülasyonunun rastgele oluşturulduğunu ispatlamaktadır. Ayrıca bu durum çizelgeleme problemi gibi kombinatoryal problemlerin tamamen rastgele adımlarla çözüme gidilmesinin kabul edilebilir bir durum olmadığını ortaya koymaktadır.

Bu tez kapsamında öne çıkan bir başka husus rezervasyon durumudur. Konferans çizelgeleme problemlerinin gerçek bir gereksinimi olan rezervasyon sistemi çizelgeleme performansını etkilemektedir. Rezervasyon işlemi çözüm uzayında belli noktaların işaretlenerek bu noktaların genetik operatörlerle veya epigenetik operatörlerle değiştirilmesi engellenmektedir. Çizelgeleme problemi kombinatoryal bir problem olduğundan dolayı çözüm uzayında rezervasyon ile belirli noktaları işaretlemek çözümün gidişatını etkilemektedir. Bu durum belirli stokastik süreçlerle en iyiyi arayan algoritmalarda arama işlemini olumsuz etkilemektedir. Hem Genetik Algoritma hem de Epigenetik Algoritma rezervasyonlu durumda çözüme yakınsamada olumsuz etkilenmiştir.

İlerleyen iterasyonlarda rezervasyonun olduğu bir oturumda değişiklik yapılarak oturumdaki tüm konuların aynı olması amaçlandığında rezervasyondan kaynaklı en iyi

oturum sağlanamamış olur.

Bu tez kapsamında ele alınan ve geliştirilen programda tanımlanması yapılan karmaşıklık seviyeleri günümüz konferans ihtiyaçları dikkate alınarak hazırlanmıştır. Genetik Algoritma, Epigenetik Algoritma, gibi algoritmalar meta-sezgisel algoritmalarındandır. Bu tez de deney sonuçları elde edilmesi için 2 farklı konferansa ait makaleler elde edilerek veri tabanına kaydedilmiştir. Literatürdeki çalışmalarda konferans çizelgeleme problemi için sezgisel ve meta-sezgisel algoritma olan benzetimli tavlama, tam sayılı programlama, doğrusal programlama gibi başka yöntemler çözüm için önerilmiştir. Tıp alanından esinlenen Epigenetik Algoritma yapısının yazılım alanındaki optimizasyon problemlerinde klasik Genetik Algoritma yapısını tam rastgelelik anlayışından kurtardığı ortaya konmuştur. Böylelikle belirli şartlar altında kısmi kontrolün sağlandığı Epigenetik Algoritma ortaya çıkmaktadır. Kontrollü olarak yapılan gen alışverişlerinin sağlanması durumunda Epigenetik Algoritma Genetik Algoritmaya göre daha kısa sürede ve daha iyi sonuçlar vermektedir. Bu sayede daha iyi çözümün elde edilmesi sağlanmıştır. Başka araştırmacıların gelecekteki çalışmalarda Epigenetik Algoritmaya diğer metasezgisel algoritmalarla başka problemler üzerinde karşılaştırma yapabilmesi sağlanabilir. Ayrıca başka araştırmacıların Epigenetik Algoritma ile başka problemler için başarılı çözüm algoritması olup olmadığına ilişkin çalışmaların da önü açılmış olmaktadır.

8. KAYNAKLAR

- [1] F. Horasan, “Genetik algoritma ile arama motorları için web site optimizasyonu,” Yüksek lisans tezi, Bilgisayar ve Bilişim Mühendisliği, Fen Bilimleri Enstitüsü, Sakarya Üniversitesi, Sakarya, Türkiye, 2014.
- [2] D. Karaboğa, *Yapay Zeka Optimizasyon Algoritmaları*, 6. baskı, Ankara, Türkiye: Nobel Akademik Yayıncılık, 2018, böl. 1, ss. 1-212.
- [3] D. Pham ve D. Karaboğa, “Genetic algorithms, tabu search, simulated annealing and neural networks,” *Intelligent Optimisation Techniques*, 1. baskı. London, England: Springer Verlag, 2012, böl. 1, ss. 1–20.
- [4] B. T. Tezel, “Metasezgisel algoritmalar için bir işbirlikçi sistem ve uygulaması,” Yüksek lisans tezi, İstatistik, Fen Bilimleri Enstitüsü, Ege Üniversitesi, İzmir Türkiye, 2015.
- [5] B. Akay, “Nümerik optimizasyon problemlerinde yapay arı kolonisi,” Doktora tezi, Bilgisayar Mühendisliği, Fen Bilimleri Enstitüsü, Erciyes Üniversitesi, Kayseri, Türkiye, 2009.
- [6] A. B. Tafida, “Benzetilmiş tavlama algoritması kullanılarak betonarme bir köprünün toplam maliyet optimizasyonu,” Yüksek lisans tezi, İnşaat Mühendisliği, Fen Bilimleri Enstitüsü, Atılım Üniversitesi, Ankara, Türkiye, 2014.
- [7] J. J. Tsay ve J. S. Arora, “Optimum design of nonlinear structures with path dependent reponse,” *Structural Optimization*, c. 1, sayı 4, ss. 202-213, 1989.
- [8] B. Çekiç, “Ameliyathanelerin çizelgelenmesi, bir karışık tamsayılı programlama yaklaşımı,” *Verimlilik Dergisi*, c. 1, sayı 2, ss. 7–28, 2015.
- [9] E. A. Silver, D. F. Pyke ve R. Peterson, *Inventory Management and Production Planning And Scheduling*, 3. baskı, New Jersey, USA: John Wiley & Sons, 1998, böl. 1, ss. 1–265.
- [10] M. Pinedo ve K. Hadavi, “Scheduling: theory, algorithms and systems development,” *Operations Research Proceedings 1991*, 1. baskı. Berlin, Germany: Springer Verlag, 2012, ss. 35-42.
- [11] S. Paksoy ve A. Uzun, “Genetik algoritma ile kaynak kısıtlı proje çizelgeleme,” *Çukurova Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, c. 17, sayı 2, ss. 345-362, 2008.
- [12] A. A. E. Adoly, M. Gheith ve M. N. Fors, “A new formulation and solution for the nurse scheduling problem: A case study in Egypt,” *Alexandria Engineering Journal*, c. 57, sayı 4, ss. 2289-2298, 2018.
- [13] S. I. Hossain, M. A. H. Akhanda, M. I. R. Shuvoa, N. Siddiqueb ve H. Adelic, “Optimization of university course scheduling problem using particle swarm optimization with selective search,” *Expert Systems with Applications*, c. 127, sayı 1, ss. 9-24, 2019.

- [14] C. Li, J. Tang, T. Ma, X. Yang, ve Y. Luo, "Load balance based workflow job scheduling algorithm in distributed cloud," *Journal of Network and Computer Applications*, c. 152, ss. 1-15, 2020.
- [15] J. Li, "Analysis of the discrete-time geo/ g/ 1 working vacation queue and its application to network scheduling," *Computers and Industrial Engineering*, c. 65, sayı 4, ss. 594-604, 2013.
- [16] A. M. Kahraman ve D. Özdağlar, "Su dağıtım sistemlerinin genetik algoritma ile optimizasyonu," *Dokuz Eylül Üniversitesi Fen ve Mühendislik Dergisi*, c. 6, sayı 3, ss. 1-18, 2004.
- [17] İ. Gölcük, A. Baykasoğlu ve F. S. Madenoğlu, "Kril sürüsü algoritması ile atölye çizelgeleme," *Dokuz Eylül Üniversitesi Mühendislik Fakültesi Fen ve Mühendislik Dergisi*, c. 16, sayı 48, ss. 61-75, 2014.
- [18] X. Zhang, P. Yu, J. Yan, H. Hu ve N. Goureaia, "Developing an online patient appointment scheduling system based on web services architecture," *10. The Asia-Pacific Association for Medical Informatics*, Beijing, China, 2012, ss. 1-6.
- [19] Anonim, (2019, 2 Mayıs). Türk dil kurumu sözlükleri. [Online]. Erişim: <https://sozluk.gov.tr>.
- [20] B. Garda, A. Kurnaz ve H. Avcı. "Konya ili'nin kongre turizmi potansiyelinin değerlendirilmesi," *Selçuk Üniversitesi Sosyal ve Teknik Araştırmalar Dergisi*, c. 6, sayı 11, ss. 172-185, 2016.
- [21] H. H. Encan, "İlköğretim okullarında yapılan toplantıların işlevselliği," Yüksek lisans tezi, Eğitim Yönetimi ve Denetimi, Sosyal Bilimler Enstitüsü, Yeditepe Üniversitesi, İstanbul, Türkiye, 2012.
- [22] D. Şencan, "İlköğretim okulu yöneticilerinin toplantı yönetimi yeterliliklerine ilişkin öğretmen algıları," Yüksek lisans tezi, Eğitim Yönetimi ve Denetimi, Sosyal Bilimler Enstitüsü, Yeditepe Üniversitesi, İstanbul, Türkiye, 2008.
- [23] A. Taşkın, "Toplantı yönetimi ve bir uygulama," Yüksek lisans tezi, İşletme, Sosyal Bilimler Enstitüsü, Balıkesir Üniversitesi, Balıkesir, Türkiye, 2005.
- [24] D. Severt, Y. Wang, P. Chen ve D. Breit, "Examining the motivation, perceived performance, and behavioral intentions of convention attendees: evidence from a regional conference," *Tourism Management*, c. 28, sayı 2, ss. 399-408, 2007.
- [25] Anonim, (2019, 2 Mayıs). [Online]. Erişim: <http://www.kongre2018.com>.
- [26] Türkiye Seyahat Acentaları Birliği, "Türsab Kongre Turizmi Raporu", Türkiye, Rap.Tursab:12188, 2013.
- [27] Akademik teşvik ödeneğinin yürürlüğe konulmasına ilişkin kanun, *T.C. Resmi Gazete*, Sayı: 11834, 14 Mayıs 2018.
- [28] S. Zilberstein, J. Koehler ve S. Koenig, "The fourteenth international conference on automated planning and scheduling (icaps-04)," *AI Magazine*, c. 25, sayı 4, ss. 101-104, 2004.
- [29] S. Aktay, "How to organize a symposium: tracking digital footprints," *Electronic Turkish Studies*, c. 12, sayı 23, ss. 1-14, 2017.
- [30] O. Andlauer, C. Obradors-Tarrago, C. Holt ve D. Moussaoui, *Psychiatry in Practice*, 1. baskı, Oxford, England: Oxford Press, 2016, böl. 9, ss. 97-112.

- [31] M. G. Nicholls, "A small-to-medium-sized conference scheduling heuristic incorporating presenter and limited attendee preferences," *Journal of the Operational Research Society*, c. 58, sayı 3, ss. 301-308, 2007.
- [32] R. F. Potthoff, ve S. J. Brams, "Scheduling of panels by integer programming: results for the 2005 and 2006 new orleans meetings," *Public Choice*, c. 131, sayı 3, ss. 465-468, 2007.
- [33] F. Zulkipli, H. Ibrahim ve A. M. Benjamin, "Optimization capacity planning problem on conference scheduling," *Business Engineering and Industrial Applications Colloquium*, Langkawi, Malaysia, 2013, ss. 911-915.
- [34] R. W. Eglese ve G. K. Rand, "Conference seminar timetabling," *Journal of the Operational Research Society*, c. 38, sayı 7, ss. 591-598, 1987.
- [35] H. Ibrahim, R. Ramli, ve M. H. Hassan, "Combinatorial design for a conference: constructing a balanced three-parallel session schedule," *Journal of Discrete Mathematical Sciences and Cryptography*, c. 11, sayı 3, ss. 305-317, 2008.
- [36] S. E. Sampson, ve E. N. Weiss, "Designing conferences to improve resource utilization and participant satisfaction," *Journal of the Operational Research Society*, c. 47, sayı 2, ss. 297-314, 1996.
- [37] S. E. Sampson, "Practical implications of preference-based conference scheduling," *Production and Operations Management*, c. 13, sayı 4, ss. 205-215, 2004.
- [38] G. M. Thompson, "Improving conferences through session scheduling," *Cornell Hotel and Restaurant Administration Quarterly*, c. 43, sayı 3, ss. 71-76, 2002.
- [39] A. Bhardwaj, J. Kim, S. Dow, D. Karger, S. Madden, R. Miller ve H. Zhang, "Attendee-sourcing: exploring the design space of community-informed conference scheduling," 2. *Association for the Advancement of Artificial Intelligence Conference on Human Computation and Crowdsourcing*, California, USA, 2014, ss. 2-10.
- [40] M. Tanaka, Y. Mori ve A. Bargiela, , "Granulation of keywords into sessions for timetabling conferences," 3. *International Symposium on Advanced Intelligent Systems*, Tsukuba, Japan, 2002, ss. 1-5.
- [41] Y. Mori ve M. Tanaka, "A hybrid grouping genetic algorithm for timetabling of conference programs," 4. *International Conference on the Practice and Theory of Automated Timetabling*, Gent, Belgium, 2002, ss. 421-440.
- [42] E. Edis ve R. S. Edis, "Konferans çizelgeleme problemi için bir tamsayılı programlama modeli," *Celal Bayar Üniversitesi Fen Bilimleri Dergisi*, c. 9, sayı 2, ss. 55-62, 2013.
- [43] T. Stidsen, D. Pisinger ve D. Vigo, "Scheduling euro-k conferences," *European Journal of Operational Research*, c. 270, sayı 3, ss. 1138-1147, 2018.
- [44] F. Fang, M. Tambe, B. Dilkina ve A. J. Plumptre, *Artificial Intelligence and Conservation*, 3. baskı, Cambridge, England: Cambridge University Press, 2019, böl. 1, ss. 5-15.
- [45] A. Badar, B. S. Umre ve A. S. Junghare, "Study of artificial intelligence optimization techniques applied to active power loss minimization," *Journal of Electrical and Electronics Engineering*, ss. 39-45, 2014.

- [46] B. Durmuş, “Tamsayılı programlamada klasik ve greedy sezgisel algoritma sonuçlarının karşılaştırılması,” Yüksek lisans tezi, İstatistik, Fen Bilimleri Enstitüsü, Muğla Sıtkı Koçman Üniversitesi, Muğla, Türkiye, 2018.
- [47] J. F. Cordeau, M. Gendreau, G. Laporte, J. Y. Potvin, ve F. Semet, “A guide to vehicle routing heuristics,” *The Journal of the Operational Research Society*, c. 53, sayı 5, ss. 512-522, 2002.
- [48] T. Yıldırım, “Simetrik gezgin satıcı problemleri için yeni bir meta-sezgisel: kör fare algoritması,” Yüksek lisans tezi, Endüstri Mühendisliği, Fen Bilimleri Enstitüsü, Pamukkale Üniversitesi, Denizli, Türkiye, 2014.
- [49] J. Pearl, *Intelligent Search Strategies for Computer Problem Solving*, 1. baskı, California, USA: Addison-Wesley Publishing, 1984, böl. 1, ss.1-73.
- [50] S. Başbuğ, “Metasezgisel algoritmalar ile anten dizi sentezi,” Doktora tezi, Elektrik Elektronik Mühendisliği, Fen Bilimleri Enstitüsü, Erciyes Üniversitesi, Kayseri, Türkiye 2014.
- [51] T. Kesintürk, “Diferansiyel gelişim algoritması,” *İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi*, c. 5, sayı 9, ss. 85-99, 2006.
- [52] A. Ateş, “Sezgisel algoritmalarla çoklu denetçi parametrelerinin optimizasyonu,” Doktora tezi, Bilgisayar Mühendisliği, Fen Bilimleri Enstitüsü, İnönü Üniversitesi, Malatya, Türkiye, 2018.
- [53] M. Eryavuz ve C. Gencer, “Araç rotalama problemine ait bir uygulama,” *Süleyman Demirel Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, c. 6, sayı 1, ss. 139-155, 2001.
- [54] S. J. Russel ve P. Norving, *Artificial Intelligence: A Modern Approach*, 11. baskı, New Jersey, USA: Prentice Hall, 2003, böl. 5, ss. 99-130.
- [55] D. Henderson, S. H. Jacobson ve A. W. Johnson, “The theory and practice of simulated annealing,” *Handbook of Metaheuristics*, böl. 10, 2003, ss. 287–319.
- [56] M. Şirin, “Sezgisel algoritma kullanılarak en iyi yol rotalanması ve bir uygulama,” Yüksek lisans tezi, Bilgisayar Mühendisliği, Fen Bilimleri Enstitüsü, İstanbul Aydın Üniversitesi, İstanbul, Türkiye, 2018.
- [57] E. G. Talbi, *Metaheuristics: From Design To Implementation*, 3. baskı, New Jersey, USA: John Wiley & Sons, 2013, böl. 1, ss. 1–29.
- [58] B. Afşar, “Metasezgisel algoritmalarda çevrimiçi ayarlama ile uyarlanabilir parametre kontrolü,” Doktora tezi, Bilgisayar Mühendisliği, Fen Bilimleri Enstitüsü, Ege Üniversitesi, İzmir, Türkiye, 2014.
- [59] I. H. Osman, ve G. Laporte, “Metaheuristics: a bibliography,” *Annals of Operations Research*, c. 63, ss. 513–623, 1996.
- [60] M. Yetkin, “Metasezgisel algoritmaların jeodezide kullanımı,” *Geomatik Dergisi*, c.1, sayı 1, ss. 8-13, 2016.
- [61] C. Blum ve A. Roli, “Metaheuristics in combinatorial optimization: overview and conceptual comparison,” *ACM Computing Surveys*, c. 35, sayı 3, ss. 268-308, 2003.

- [62] E. Özçetin, “Açık araç rotalama problemi için metasezgisel algoritma tasarımı ve uygulaması,” Doktora tezi, Endüstri Mühendisliği, Fen Bilimleri Enstitüsü, Eskişehir Teknik Üniversitesi, Eskişehir, Türkiye, 2019.
- [63] C. Jatoth, G. R. Gangadharan, ve R. Buyya, “Optimal fitness aware cloud service composition using an adaptive genotypes evolution based genetic algorithm,” *Future Generation Computer Systems*, c. 94, sayı 185-198, 2019.
- [64] S. Biroğul, “Epigenetic algorithm for optimization: application to mobile network frequency planning,” *Arabian Journal for Science and Engineering*, c. 41, sayı 3, ss. 883-896, 2016.
- [65] F. Glover, “Tabu search—part I,” *ORSA Journal on Computing*, c. 1, sayı 3, ss. 190-206, 1989.
- [66] O. Alfarraj, A. Alzubi, ve A. Tolba, “Optimized feature selection algorithm based on fireflies with gravitational ant colony algorithm for big data predictive analytics,” *Neural Computing and Applications*, c. 31, sayı 5, ss. 1391-1403, 2019.
- [67] T. Chen, J. Shi, J. Yang, ve G. Li, “Enhancing network cluster synchronization capability based on artificial immune algorithm,” *Human-Centric Computing and Information Sciences*, c. 9, sayı 1, ss. 1-28, 2019.
- [68] H. Rao, X. Shi, A. K., Rodrigue, J. Feng, Y. Xia, M. Elhoseny, X. Yuanc ve L. Gu, “Feature selection based on artificial bee colony and gradient boosting decision tree,” *Applied Soft Computing*, c. 74, ss. 634-642, 2019.
- [69] M. Ghosh, S. Begum, R. Sarkar, D. Chakraborty ve U. Maulik, “Recursive memetic algorithm for gene selection in microarray data,” *Expert Systems with Applications*, c. 116, ss. 172-185, 2019.
- [70] P. Civicioğlu ve E. Besdok, “Bernstain-search differential evolution algorithm for numerical function optimization,” *Expert Systems with Applications*, c. 138, ss. 1-18, 2019.
- [71] A. P. Karpenko ve I. A. Leshchev, “Advanced cat swarm optimization algorithm in group robotics problem,” *Procedia Computer Science*, c. 150, ss. 95-101, 2019.
- [72] T. Yildirim, C. B. Kalayci ve Ö. Mutlu, “Gezgin satıcı problemi için yeni bir meta-sezgisel:kör fare algoritması,” *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, c. 22, sayı 1, ss. 64-70, 2016.
- [73] Y. J. Zheng, “Water wave optimization: a new nature-inspired metaheuristic,” *Computers & Operations Research*, c. 55, ss. 1-11, 2015.
- [74] S. Biroğul, “Hybrid harris hawks optimization based on differential evolution (hhode) algorithm for optimal power flow problem,” *IEEE Access*, c. 7, ss. 184468-184488, 2019.
- [75] Ö. İ. Şimşek, “Parçacık sürüsü optimizasyon yöntemi ile kesir dereceli filtre gerçekleştirilmesi,” Yüksek lisans tezi, Bilgisayar Mühendisliği, Fen Bilimleri Enstitüsü, İnönü Üniversitesi, Malatya, Türkiye, 2018.
- [76] M. Dorigo ve C. Blum, “Ant colony optimization theory: a survey,” *Theoretical Computer Science*, c. 344, ss. 243–278, 2005.

- [77] B. Ayan, "Hemşire nöbet çizelgeleme probleminin tavlama benzetimi algoritması ile çözümü," Yüksek lisans tezi, Endüstri Mühendisliği, Fen Bilimleri Enstitüsü, Necmettin Erbakan Üniversitesi, Konya, Türkiye 2019.
- [78] Y. Özcan, "Çok modlu optimizasyon problemleri için bir yapay arı kolonisi algoritması gerçekleştirimi," Yüksek lisans tezi, Bilgisayar Mühendisliği, Fen Bilimleri Enstitüsü, Kütahya Dumlupınar Üniversitesi, Kütahya, 2019.
- [79] H. O. Erkol, "Ters sarkaç sisteminin yapay arı kolonisi algoritması ile optimizasyonu," *Politeknik Dergisi*, c. 20, sayı 4, ss. 863-868, 2017.
- [80] J. H. Holland, *Adaptation in Natural and Artificial Systems*, 3. baskı, Michigan, USA: MIT Press, 1975, c. 1, ss. 1-19.
- [81] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and machine learning*, 1. baskı, Michigan, USA: Addison-Wesley, 1989, böl. 3, ss. 60-80.
- [82] D. E. Goldberg, "Genetic algorithms and walsh functions: part 1, a gentle introduction," *Complex Systems*, c. 3, sayı 2, ss. 129-152, 1989.
- [83] L. Davis, *Handbook of Genetic Algorithms*, 1. baskı, Michigan, USA: Van Nostrand Reinhold, 1991, böl. 1, ss.1-384.
- [84] S. Arslan, T. Tuncer ve A. Karcı, "Çoklu-dizi hizalama problemi için genetik algoritma," 6. *Elektrik-Elektronik-Bilgisayar Mühendisliği Sempozyumu*, Bursa, Türkiye, 2004, ss. 353-356.
- [85] M. T. Younis ve S. Yang, "Genetic algorithm for independent job scheduling in grid computing," *Mendel Soft Computing Journal*, c. 23, sayı 1, ss. 65-72, 2017.
- [86] P. Smirnov, M. Melnik ve D. Nasonov, "Performance-aware scheduling of streaming applications using genetic algorithm," *International Conference on Computational Science*, Zürich, Switzerland, 2017, ss. 2240-2249.
- [87] K. Çakar, "Genetik algoritmalar yardımıyla acil servis istasyonu yerleşiminin optimizasyonu," Doktora tezi, İşletme, Sosyal Bilimler Enstitüsü, Atatürk Üniversitesi, Erzurum, Türkiye, 2009.
- [88] T. Timuçin, "Ameliyat odası çizelgelemenin genetik algoritma ile gerçekleştirilmesi," Yüksek lisans tezi, Bilgisayar Mühendisliği, Fen Bilimleri Enstitüsü, Düzce Üniversitesi, Düzce, Türkiye, 2018.
- [89] T. Timuçin ve S. Biroğul, "Effect the number of reservations on implementation of operating room scheduling with genetic algorithm," *Lecture Notes on Data Engineering and Communications Technologies*, 1. baskı. Antalya, Türkiye, 2019, böl. 9, ss. 252-265.
- [90] P. Kora, ve P. Yadlapalli, "Crossover operators in genetic algorithms: a review," *International Journal of Computer Applications*, c. 162, sayı 10, ss. 34-36, 2017.
- [91] A. Fırlalı ve O. Engin, "Akış tipi çizelgeleme problemlerinin genetik algoritma yardımı ile çözümünde uygun çaprazlama operatörünün belirlenmesi," *Doğuş Üniversitesi Dergisi*, c. 3, sayı 2, ss. 27-35, 2011.
- [92] B., Bolat, K. O. Erol ve C. E. İmrak, "Mühendislik uygulamalarında genetik algoritma ve operatörlerin işlevleri," *Mühendislik ve Fen Bilimleri Dergisi*, c. 22, sayı 4, ss. 264-271, 2004.

- [93] M. Pulat ve İ. D. Kocakoç, “Gezgin satıcı probleminin çözümünde kullanılan genetik algoritmanın parametrelerinin incelenmesi,” *Uluslararası İktisadi Ve İdari İncelemeler Dergisi*, c. 16, sayı 1, ss. 21-36, 2017.
- [94] D. B. Fogel, “An evolutionary approach to the traveling salesman problem,” *Journal Biological Cybernetics*, c. 60, sayı 2, ss.139–144, 1988.
- [95] M. S. Emek, “İş yaşam dengesi bakışıyla hastane personelinin genetik algoritma yöntemi ile çizelgelenmesi,” Yüksek lisans tezi, Endüstri Mühendisliği, Fen Bilimleri Enstitüsü, Süleyman Demirel Üniversitesi, Isparta, Türkiye, 2016.
- [96] A. Coşkun ve N. Arıcı, “Genetik algoritma kullanılarak deriden geçiş özelliklerine göre kimyasal maddelerin molekül yapılarının bulunması,” *Politeknik Dergisi*, c. 9, sayı 4, ss. 255-261, 2006.
- [97] S. Biroğul, “Genetik algoritma yaklaşımıyla atölye çizelgeleme,” Yüksek lisans tezi, Elektrik ve Elektronik Mühendisliği, Fen Bilimleri Enstitüsü, Gazi Üniversitesi, Ankara, Türkiye, 2005.
- [98] S. Biroğul, “Gsm şebekelerinde frekans planlamasının veri füzyonu ile gerçekleştirilmesi,” Doktora tezi, Elektrik Eğitimi, Fen Bilimleri Enstitüsü, Gazi Üniversitesi, Ankara, Türkiye, 2008.
- [99] S. Biroğul, ve T. Yiğit, “Genetik algoritmada tamir operatörünün etkinliği,” 3. *Akıllı Sistemlerde Yenilikler ve Uygulamaları Konferansı*, Isparta, Türkiye, 2008, ss. 134-137.
- [100] D. L. Hartl, ve E. W. Jones, “Genetic analysis of genes and genomes,” *Neuro-Oncology*, c. 7, sayı 2, ss. 204-205, 2005.
- [101] R. Jaenisch, ve A. Bird, “Epigenetic regulation of gene expression: how the genome integrates intrinsic and environmental signals,” *Nature genetics*, c. 33, sayı 3, ss. 245-254, 2003.
- [102] A. Madlung, ve L. Comai, “The effect of stress on genome regulation and structure,” *Annals of Botany*, c. 94, sayı 4, ss. 481-495, 2004.
- [103] B. Yavuz, “Blog yoluyla öğretilen epigenetik konusunun öz düzenleme becerisi üzerine etkisi,” Yüksek lisans tezi, Biyoloji Öğretmenliği, Fen Bilimleri Enstitüsü, Gazi Üniversitesi, Ankara, Türkiye, 2014.
- [104] R. Jirtle, “Epigenome: the program for human health and disease,” *Epigenomics*, c. 1, sayı 1, ss. 13-16, 2009.
- [105] K. D. Siegmund, C. M. Connor, M. Campan, T. I. Long, D. J. Weisenberger, D. Biniszkiwicz, R. Jaenisch, P. W. Laird, ve S. Akbarian, “Dna methylation in the human cerebral cortex is dynamically regulated throughout the life span and involves differentiated neurons,” *PloS One*, c. 2, sayı 9, ss. 1-9, 2007.
- [106] T. Kouzarides, “Chromatin modifications and their function,” *Cell*, c. 128, sayı 4, ss. 693-705, 2007.
- [107] R. Metivier, R. Gallais, C. Tiffocche, C. L. Peron, R. Z. Jurkowska, R. P. Carmouche, D. Ibberson, P. Barath, F. Demay, G. Reid, V. Benes, A. Jeltsch, F. Gannon ve G. Salbert, “Cyclical dna methylation of a transcriptionally active promoter,” *Nature*, c. 452, sayı 7183, ss. 45-50, 2008.

- [108] Z. Sarıgöl, “Dinitroanilin herbisitlerin neden olduğu genotoksik etkilerin ve epigenetik değişikliklerin değerlendirilmesi,” Doktora tezi, Farmasötik, Sağlık Bilimleri Enstitüsü, Hacettepe Üniversitesi, Ankara, Türkiye, 2015.
- [109] E. Koçak ve A. Ertuğrul, “Psikiyatrik bozukluklar ve epigenetik,” *Psikiyatri Dergisi*, c. 23, sayı 2, ss. 130-140, 2012.
- [110] E. J. Richards, “Inherited epigenetic variation-revisiting soft inheritance,” *Nature Reviews Genetics*, c. 7, sayı 5, ss. 395-401, 2006.
- [111] G. P. Delcuve, M. Rastegar ve J. R. Davie, “Epigenetic control,” *Journal of Cellular Physiology*, c. 219, sayı 2, ss. 243-50, 2009.
- [112] Z. A. Kaminsky, T. Tang, S.C. Wang, C. Ptak, G. H. T. Oh, A. H C Wong, L. A. Feldcamp, C. Virtanen, J. Halfvarson, C. Tysk, A. F. McRae, P. M. Visscher, G. W. Montgomery, I. I. Gottesman, N. G Martin ve A. Petronis, “Dna methylation profiles in monozygotic and dizygotic twins,” *Nature Genetics*, c. 41, sayı 2, ss. 240-245, 2009.
- [113] E. Jablonka ve M. J. Lamb, “The changing concept of epigenetics,” *Annals of the New York Academy of Sciences*, c. 981, sayı 1, ss. 82-96, 2002.
- [114] C. Arnold, P. F. Stadler ve S. J. Prohaska, “Chromatin computation: epigenetic inheritance as a pattern reconstruction problem,” *Journal of Theoretical Biology*, c. 336, ss. 61-74, 2013.
- [115] R. Holliday, “Epigenetics: a historical overview,” *Epigenetics*, c. 1, sayı 2, ss. 76-80, 2006.
- [116] M. Ezzarii, H. Elghazi, H. El Ghazi ve T. Sadiki, “Epigenetic algorithm for performing intrusion detection system,” *International Conference on Advanced Communication Systems and Information Security*, Marrakesh, Morocco, 2016, ss. 1-6.
- [117] J. M. Berg, J. L. Tymoczko ve L. Stryer, *Biochemistry*, 8. baskı, New York, USA: W. H. Freeman, 2006, böl. 1, ss. 1-45.
- [118] D. H. Stolfi ve E. Alba, “Epigenetic algorithms: a new way of building gas based on epigenetics,” *Information Sciences*, c. 424, ss. 250-272, 2018.
- [119] S. Periyasamy, A. Gray ve P. Kille, “The epigenetic algorithm,” *IEEE World Congress on Computational Intelligence*, Hong Kong, China, 2008, ss. 3228-3236.

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Ercan ATAGÜN
Doğum Tarihi ve Yeri : 30.01.1991 / OSMANİYE
Yabancı Dili : İngilizce
E-posta : ercan-atagun@hotmail.com

ÖĞRENİM DURUMU

Derece	Alan	Okul/Üniversite	Mezuniyet Yılı
Y. Lisans	Bilgisayar Mühendisliği.	Düzce Üniversitesi	2020
Lisans	Bilgisayar Mühendisliği.	Gazi Üniversitesi	2015
Lise	Fen Bilimleri	Şht. Öğr. Orhan Gök Anadolu Lisesi	2009