T.C.
İSTANBUL ÜNİVERSİTESİ
SOSYAL BİLİMLERİ ENSTİTÜSÜ
İŞLETME ANABİLİM DALI
SAYISAL YÖNTEMLER BİLİM DALI

DOKTORA TEZİ

ÜNİVERSİTE SINAV ÇİZELGELEME PROBLEMLERİNİN YAPAY ZEKA TEKNİKLERİ İLE ÇÖZÜMÜ

YAKUP ÇELİKBİLEK
2502130302

TEZ DANIŞMANI
DOÇ. DR. TUNÇHAN CURA

İstanbul – 2016

T.C.
İSTANBUL ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ

DOKTORA
TEZ ONAYI

ÖĞRENCİNİN;

Adı ve Soyadı	:	Yakup ÇELİKBİLEK	**Numarası**	:	2502130302
Anabilim Dalı / Anasanat Dalı / Programı	:	**İşletme Anabilim Dalı Sayısal Yöntemler Bilim Dalı**	**Danışmanı**	:	**DOÇ.DR. TUNÇHAN CURA**
Tez Savunma Tarihi	:	16.06.2016	**Saati**	:	11.00
Tez Başlığı	:	"Üniversite Sınav Çizelgeleme Problemlerinin Yapay Zeka Teknikleriyle Çözümü"			

TEZ SAVUNMA SINAVI, İÜ Lisansüstü Eğitim-Öğretim Yönetmeliği'nin **50. Maddesi** uyarınca yapılmış, sorulan sorulara alınan cevaplar sonunda adayın tezinin **KABULÜNE** OYBİRLİĞİ ~~/ OYÇOKLUĞUYLA~~ karar verilmiştir.

JÜRİ ÜYESİ	İMZA	KANAATİ (KABUL / RED / DÜZELTME)
1- PROF. DR. MEHPARE TİMÖR		Kabul
2- PROF.DR. Ş. ALP BARAY		Kabul
3 PROF. DR. MELTEM ÖZTURAN		Kabul
4- PROF. DR.SELİM ZAİM		Kabul
5- DOÇ. DR. TUNÇHAN CURA		Kabul

YEDEK JÜRİ ÜYESİ	İMZA	KANAATİ (KABUL / RED / DÜZELTME)
1- PROF. DR. RAUF NİŞEL		
2- DOÇ. DR. SEDA TOLUN		

Bu tez çalışması İstanbul Üniversitesi Öğretim Üyesi Yetiştirme Programı (ÖYP) Koordinatörlüğünün 096.2014–DR–35/25–10 numaralı projesi ile desteklenmiştir.

.

ÖZ

ÜNİVERSİTE SINAV ÇİZELGELEME PROBLEMLERİNİN YAPAY ZEKA TEKNİKLERİ İLE ÇÖZÜMÜ

Yakup ÇELİKBİLEK

Çizelgeleme, sistemler için her zaman önemli bir konu olmuştur. Teknolojinin gelişimine kadar manuel olarak gerçekleştirilen çizelgeleme işlemleri, günümüzde bilgi teknolojilerinin de gelişimiyle birlikte artan bir önem kazanmıştır. Bilgi teknolojileri kullanılarak gerçekleştirilen çizelgeleme işlemlerinin, manuel olarak gerçekleştirilen çizelgeleme işlemlerine göre sistemlerin etkinliğini yüksek miktarda arttırmakta olduğu literatürde yer alan çalışmalarda gözlemlenmektedir. Bu tez çalışmasında da çizelgeleme problemlerinden biri olan üniversite sınav çizelgeleme problemlerinin çözümü yapay zeka teknikleri kullanılarak gerçekleştirilmiştir.

Çalışmanın amacı, üniversite sınav çizelgeleme problemlerinin hem hızlı hem de etkin bir şekilde çözümlerinin gerçekleştirilmesidir. Bu amaçla, çalışmanın ilk bölümünde uzman sistemler incelenmiştir. Daha sonra, ikinci kısımda en sık kullanılan sezgisel yöntemler incelenmiştir. Üçüncü kısımda, tezin de temelini oluşturan üniversite sınav çizelgeleme problemi ve literatür taramasına yer verilmiştir. Önerilmekte olan üniversite sınav çizelgeleme yöntemi de dördüncü bölümde ayrıntılı olarak aktarılmıştır. Son bölümde de, önerilmekte olan üniversite sınav çizelgeleme yöntemi kullanılarak gerçekleştirilmiş olan Toronto Üniversitesi Kıyaslama Veri Seti uygulamalarına ait sonuçlar iki başlık altında aktarılmıştır.

Anahtar Kelimeler: Sınav Çizelgeleme, Üniversite Sınav Çizelgeleme, Yapay Zeka, Uzman Sistem, Genetik Algoritma.

ABSTRACT

SOLVING THE UNIVERSITY EXAM TIMETABLING PROBLEMS WITH ARTIFICIAL INTELLIGENCE TECHNIQUES

Yakup ÇELİKBİLEK

Timetabling is always an important issue for systems. Timetabling process performed manually until the development of technology has gained increasing importance with the development of information technologies nowadays. It is observed in the studies in the literature that timetabling process performed by using information technologies increases the effectiveness of systems greatly according to timetabling process performed manually. In this thesis, one of the timetabling problems, university exam timetabling problems were performed by using artificial intelligence techniques.

The aim of the study is to be performed of the solutions of the university exam timetabling problems both quickly and effectively. With this aim, expert systems were analyzed in the first section of the study. Afterwards, in the second section, most frequently applied heuristic methods were analyzed. The third section was given to literature review and university exam timetabling which is formed the basis of the thesis. Proposed university exam timetabling method was described in detailed in the fourth section. In the last section, Toronto University Benchmark data set application results which is obtained by using the proposed university exam timetabling method were reported under two headings.

Keywords: Exam Timetabling, University Exam Timetabling, Artificial Intelligence, Expert System, Genetic Algorithm.

ÖNSÖZ

İnsanoğlunun tarih boyunca karşılaştığı en önemli problemlerden birisi karar verme problemidir. Az sayıda değişkene ya da az sayıda kısıta sahip problemlerin çözümü çoğunlukla, deterministik yöntemler kullanılarak rahatlıkla elde edilebilirken değişken sayısının fazla sayıda olması ve aynı zamanda kısıtların da artması problemlerin çözümlerini zorlaştırmaktadır. Çok sayıda değişkene sahip problemlerin çözüm uzayının konveks bir küme olması durumunda da çözüm yine deterministik yöntemlerle elde edilebilmektedir. Fakat problemde yer alan değişken ve kısıt sayısının fazla olmasıyla birlikte aynı zamanda çözüm uzayının da konkav bir bölge olması diğer bir ifadeyle çok fazla sayıda yerel optimum değerine sahip olması durumunda en iyi çözüm deterministik yöntemlerle tespit edilemeyecek durumlarda olabilmektedir.

Sınav çizelgeleme problemleri de bu şekilde hem sonsuz çözüm kombinasyonuna hem de çok fazla sayıda yerel optimum değerlerine sahip olan problemlerdendir. Literatürde sınav çizelgeleme problemleri için geliştirilmiş farklı sezgisel yöntemler ve sezgisel yöntem uyarlamaları bulunmasına karşın bir uzman sistem uygulamasına rastlanmamıştır. Uzman sistem yapısı incelendiği zaman, optimum çözüme ulaşmada etkin sonuçlar elde etme yeteneği görülmektedir. Bu durumlardan dolayı da, bu tez çalışmasında uzman sistem tabanlı olarak yapay zeka teknikleriyle üniversite sınav çizelgeleme problemlerinin en iyi çözümünün elde edilmesi üzerine çalışılmıştır.

İstanbul Üniversitesi İşletme Fakültesi bünyesinde görev yaptığım süre boyunca hayatıma değer katmış olan başta Yıldız Yılmaz, Bilgen Onurlu Kürkçü, Şirin İlhan ve Murat Gülmez olmak üzere tüm akademik ve idari personele teşekkür ederim.

ÖYP bünyesindeki tüm resmi prosedürleri ve işlemlerimi sorunsuz bir şekilde gerçekleştiren başta Sevilay Gündoğdu olmak üzere tüm İstanbul Üniversitesi ÖYP Koordinatörlüğü çalışanlarına ve İstanbul Üniversitesi Sosyal Bilimler Enstitüsü çalışanlarından Fatma Akbaş ve Engin Garip'e teşekkür ederim.

Bu tez çalışmasının gerçekleştirilmesi sırasında bilgi, deneyim ve tecrübelerini paylaşarak ufkumun genişlemesini sağlayan, tezimin şekillenmesinde büyük katkıları olan ve her zaman arkamda durarak üç sene gibi bir zamanda doktora eğitimimi tamamlamama destek olan tez danışmanım sayın Doç. Dr. Tunçhan Cura'ya teşekkürü bir borç bilirim. Ayrıca, gerek doktora ders ve yeterlilik dönemim gerekse de tez yazım sürecim boyunca desteğini benden esirgemeyen arkadaşım Arş. Grv. Muhlis Özdemir'e teşekkür ederim.

Özellikle doktora sürecim boyunca tüm stresli ve tahammülsüz zamanlarımda bile bana destek olan ve her zaman moralimin yüksek olmasını sağlayan arkadaşlarım Berna Çalışkan'a, Merve Öztürk'e ve benim için her zaman kardeşten bile öte olmuş kıymetli dostum Deniz Özkum'a teşekkürlerimi borç bilirim.

Bana her zaman inanan, bugüne kadar desteklerini benden hiçbir zaman esirgemeyen, zor zamanlarımda hep yanımda olan ve bugünlere gelmemde şüphesiz en büyük katkısı olan değerli annem Yurdagül ÇELİKBİLEK'e sonsuz teşekkür ederim.

Son olarak; isimlerini sayamadığım ama bir yerinden hayatıma dokunmuş ve değer katmış olan herkese teşekkür eder, saygılarımı sunarım.

Haziran, 2016 Yakup ÇELİKBİLEK

İÇİNDEKİLER

Sayfa No

ÖZ ..... **iv**
ABSTRACT ..... **v**
ÖNSÖZ ..... **vi**
İÇİNDEKİLER ..... **viii**
ŞEKİLLER LİSTESİ ..... **xi**
TABLOLAR LİSTESİ ..... **xiii**
SİMGELER VE KISALTMALAR LİSTESİ ..... **xiv**
GİRİŞ **1**

BİRİNCİ BÖLÜM
UZMAN SİSTEMLER

1.1. UZMAN SİSTEMLERİN TEMEL BİLEŞENLERİ 5
1.2. UZMAN SİSTEMLERİN UYGULAMA ALANLARI 7
1.3. UZMAN SİSTEMLERİN AVANTAJLARI 8
1.4. UZMAN SİSTEMLERİN DEZAVANTAJLARI 11

İKİNCİ BÖLÜM
SEZGİSEL YÖNTEMLER

2.1. GENETİK ALGORİTMALAR 16
2.1.1. Genetik Algoritmalarda Kullanılan Yapılar ve Tanımlar 18
2.1.2. Seçim Teknikleri 19
2.1.2.1. Rastgele Seçim Tekniği 20
2.1.2.2. Rulet Tekerleği Seçim Tekniği 20
2.1.2.3. Turnuva Seçim Tekniği 21
2.1.2.4. Diğer Seçim Teknikleri 22

2.1.3. Çaprazlama Teknikleri 22
2.1.3.1. Tek Nokta Çaprazlama 22
2.1.3.2. Çift Nokta Çaprazlama 23
2.1.3.3. Uniform Çaprazlama 24
2.2. KARINCA KOLONİSİ OPTİMİZASYONU 24
2.2.1. Karınca Kolonisi Optimizasyonunda Kullanılan Yapılar ve Tanımlar 26
2.3. YAPAY ARI KOLONİSİ 28
2.3.1. Yapay Arı Kolonisinde Kullanılan Yapılar ve Tanımlar 29
2.3.2. Yapay Arı Kolonisinde Arıların Davranışları 31
2.4. TABU ARAMA ALGORİTMASI 32
2.4.1. Tabu Arama Algoritmasında Kullanılan Yapılar ve Tanımlar 35
2.4.2. Tabu Arama Algoritmasında Kullanılan Stratejiler 37
2.4.2.1. Aday Liste Stratejileri 37
2.4.2.2. Yoğunlaştırma Stratejileri 38
2.4.2.3. Çeşitlendirme Stratejileri 38
2.4.3. Tabu Arama Algoritmasında Kullanılan Hafıza Yapıları 39
2.5. TAVLAMA BENZETİMİ 40
2.5.1. Tavlama Benzetiminde Kullanılan Yapılar ve Tanımlar 42
2.6. PARÇACIK SÜRÜ OPTİMİZASYONU 43
2.6.1. Parçacık Sürü Optimizasyonunda Kullanılan Yapılar ve Tanımlar 45
2.7. DİĞER SEZGİSEL YÖNTEMLER 47

ÜÇÜNCÜ BÖLÜM
ÜNİVERSİTE SINAV ÇİZELGELEME

3.1. ÜNİVERSİTE SINAV ÇİZELGELEME PROBLEMİ VE BİLEŞENLERİ 51
3.1.1. Üniversite Sınav Çizelgeleme Problemi 51
3.1.2. Üniversite Sınav Çizelgeleme Probleminin Bileşenleri 52
3.1.2.1. Üniversite Sınav Çizelgeleme Problemlerinde Yer Alan Kısıtlar 56
3.2. LİTERATÜR TARAMASI 58
3.2.1. Toronto Üniversitesi Kıyaslama Veri Seti Kullanılan Çalışmalar 58
3.2.2. Nottingham Üniversitesi Kıyaslama Veri Seti Kullanılan Çalışmalar 80
3.2.3. Melbourne Üniversitesi Kıyaslama Veri Seti Kullanılan Çalışmalar 83

3.2.4. ITC2007 Veri Seti Kullanılan Çalışmalar 84
3.2.5. Diğer Gerçek Hayat Problemleri Kullanılan Çalışmalar 87

DÖRDÜNCÜ BÖLÜM
ÖNERİLEN ÜNİVERSİTE SINAV ÇİZELGELEME YÖNTEMİ

4.1. ÖNERİLEN UZMAN SİSTEM YAKLAŞIMI 93
4.1.1. Önerilen Uzman Sistemde Kullanılan Yapılar 97
4.1.2. Önerilen Uzman Sistem 98
4.2. ÖNERİLEN GENETİK ALGORİTMA YAKLAŞIMI 104
4.2.1. Önerilen Genetik Algoritmada Kullanılan Yapılar 108
4.2.2. Önerilen Genetik Algoritma 109

BEŞİNCİ BÖLÜM
UYGULAMA

5.1. SINAV KUŞAĞI SAYISININ AZALTILMASI 113
5.2. CEZA FONKSİYONU MİNİMİZASYONU 115

TARTIŞMA VE SONUÇ 118
KAYNAKLAR 124
ÖZGEÇMİŞ 138

ŞEKİLLER LİSTESİ

Sayfa No

Şekil 1.1: Uzman Sistemlerin Temel Bileşenleri ve Akış Şeması (Castillo v.d., 2012: 10). 5

Şekil 2.1: NP Problem Kümeleri. 14

Şekil 2.2: Yerel Minimuma Sahip Örnek Fonksiyon Grafiği. 15

Şekil 2.3: Genetik Algoritma Çözüm Yapısı. 17

Şekil 2.4: Tek Nokta Çaprazlama Örneği. 23

Şekil 2.5: Çift Nokta Çaprazlama Örneği. 23

Şekil 2.6: Uniform Çaprazlama Örneği. 24

Şekil 2.7: Karıncaların Yiyeceğe Ulaşmada Takip Ettikleri Yol (Colorni v.d., 1991: 135)...25

Şekil 2.8: Karınca Kolonisi Optimizasyonu Çözüm Yapısı. 26

Şekil 2.9: Yapay Arı Kolonisi Çözüm Yapısı. 28

Şekil 2.10: Arıların Yiyecek Arama Çevrimi (Akay, 2009: 57).....32

Şekil 2.11: Tabu Arama Algoritması Çözüm Yapısı. 34

Şekil 2.12: Tavlama Benzetimi Çözüm Yapısı. 41

Şekil 2.13: Parçacık Sürü Optimizasyonu Çözüm Yapısı. 44

Şekil 4.1: Uzman Sistem Çözümü Akış Şeması. 96

Şekil 4.2: Derslerin Öğrenci Listelerinin Oluşturulması. 98

Şekil 4.3: Derslerin Listesi ve Derslerin Öğrenci Listeleri Örneği.....99

Şekil 4.4: Öğrencileri Çakışan Dersler Listelerinin Oluşturulması. 99

Şekil 4.5: Öğrencisi Çakışan Dersler Listeleri Örneği. 100

Şekil 4.6: Kuşağa Koyulabilecek Dersler Listeleri Oluşturma. 101

Şekil 4.7 : Kuşağa Koyulabilecek Dersler Listeleri Örneği. 101

Şekil 4.8: Uzman Sistem Kod Taslağı. 102

Şekil 4.9: Örneğe Ait 1. İterasyon Sonuçları. 103

Şekil 4.10: Örneğe Ait 2. İterasyon Sonuçları. 103

Şekil 4.11: Örneğe Ait 3. İterasyon Sonuçları. 104

Şekil 4.12: Genetik Algoritma Çözümü Akış Şeması. 107

Şekil 4.13: Çift Nokta Çaprazlama Tekniği ile Yeni Çözümler Üretme Örneği. 109

Şekil 4.14: İkili Yer Değiştirme İşlemi ile Çözümlerin İyileştirilmesi Örneği. 110

Şekil 4.15: Ters Çevirme İşlemi ile Çözümlerin İyileştirilmesi Örneği. 110

Şekil 4.16: Genetik Algoritma Yaklaşımı Kod Taslağı. 111

TABLOLAR LİSTESİ

Sayfa No

Tablo 1.1: Uzman Sistemlerin Uygulama Alanları. 8

Tablo 3.1: Toronto Üniversitesi Kıyaslama Veri Seti (Carter v.d., 1996: 376). 59

Tablo 3.2: Carter v.d. (1996) TÜKVS Kuşak Sayısı Minimizasyonu Uygulama Sonuçları..60

Tablo 3.3: Carter v.d. (1996) TÜKVS Ceza Puanı Minimizasyonu Uygulama Sonuçları. ...61

Tablo 3.4: Literatürde Yer Alan TÜKVS Sonuçları–I. 75

Tablo 3.5: Literatürde Yer Alan TÜKVS Sonuçları–II. 76

Tablo 3.6: Literatürde Yer Alan TÜKVS Sonuçları–III. 77

Tablo 3.7: Literatürde Yer Alan TÜKVS Sonuçları–IV. 78

Tablo 3.8: Literatürde Yer Alan TÜKVS Sonuçları–V. 79

Tablo 3.9: Nottingham Üniversitesi Kıyaslama Veri Seti (Burke v.d., 1996). 80

Tablo 3.10: Burke v.d. (1996) NÜKVS Uygulama Sonuçları. 81

Tablo 3.11: Melbourne Üniversitesi Kıyaslama Veri Seti. 83

Tablo 3.12: İkinci Uluslararası Çizelgeleme Müsabakası Veri Seti (ITC2007). 85

Tablo 5.1: TÜKVS için Sınav Kuşağı Sayısının Azaltılması Uygulama Sonuçları. 114

Tablo 5.2: TÜKVS için Ceza Puanı Minimizasyonu Uygulama Sonuçları. 116

SİMGELER VE KISALTMALAR LİSTESİ

Simgeler	Açıklamalar
$\Longrightarrow$	: İse
$\wedge$	: Ve
$\vee$	: Veya
$\in$	: Elemanı
$\notin$	: Elemanı değil
%	: Yüzde
$=$	: Eşit
$\neq$	: Eşit değil
$==$	: Eşitlik koşulu
$\leq$	: Küçük eşit
$\geq$	: Büyük eşit
$\boldsymbol{\lvert x \rvert}$	: x'in mutlak değeri
$\sum$	: Toplam sembolü
$\prod$	: Çarpım sembolü
$\boldsymbol{u_i}$	: *i.* bireyin uygunluk değeri
$\boldsymbol{p_i}$	: *i.* bireyin seçim olasılığı
$\boldsymbol{F_c}$	: Ceza puanı fonksiyonu
M	: Toplam öğrenci sayısı
N	: Toplam sınav sayısı
P	: Toplam sınav kuşağı sayısı
C	: Sınavlar arası çakışan öğrenci sayıları matrisi
T	: Sınavların bulunduğu kuşakları gösteren vektör
$\boldsymbol{prox(i,\ j)}$	: *i* ve *j* sınavları arasındaki ceza katsayısı
$\boldsymbol{c_{ij}}$	: Hem *i.* hem de *j.* sınava girmesi gereken öğrenci sayısı
$\boldsymbol{[c_{ij}]_{N\times N}}$	: c_{ij} elemanlarından oluşan $N \times N$ boyutlu matris gösterimi
$\boldsymbol{t_i}$	: *i.* sınavın bulunduğu sınav kuşağı
$\boldsymbol{[t_i]_N}$	: t_i elemanlarından oluşan N boyutlu vektör gösterimi
$\boldsymbol{P_{ij}}$	: Hem *i.* hem de *j.* kuşakta sınava girmesi gereken öğrenci sayısı
$\boldsymbol{P_c}$	: Sınav kuşakları arası çakışan öğrenci sayıları matrisi

Kısaltmalar Açıklamalar

ACO : Karınca Kolonisi Optimizasyonu (Ant Colony Optimization)
CAR : Carleton University, Ottawa
CP : Ceza Puanı
DGA : Diferansiyel Gelişim Algoritması (Differential Evolution Algorithm)
EAR : Earl Haig Collegiate Institute, Toronto
GA : Genetik Algoritmalar (Genetic Algorithm)
GB : Gigabyte
GHz : Gigahertz
GPU : Graphics Processing Unit (Grafik İşleme Ünitesi)
HA : Harmoni Arama (Harmony Search)
HEC : Ecole des Hautes Etudes Commerciales, Montreal
ITC2007 : International Timetabling Competition 2007
KFU : King Fahd University of Petroleum and Minerals, Dharan
KS : Kuşak Sayısı
LD : Largest Degree (En Büyük Derece)
LE : Largest Enrollment (En Büyük Kayıt)
LSE : London School of Economics
LWD : Largest Weighted Degree (En Büyük Ağırlıklandırılmış Derece)
MEL : Melbourne Üniversitesi
MÜKVS : Melbourne Üniversitesi Kıyaslama Veri Seti
NP : Non-deterministic Polynomial-time
NÜKVS : Nottingham Üniversitesi Kıyaslama Veri Seti
ÖYP : Öğretim Üyesi Yetiştirme Programı
PATAT : The Practice and Theory of Automated Timetabling
PUR : Purdue University, Indiana
RAM : Random Access Memory (Rastgele Erişimli Bellek)
RO : Random Ordering (Rastgele Sıralama)
RRT : Record–to–Record Travel
RS : Rassal Sayı
RYE : Ryerson University, Toronto
SD : Saturation Degree (Doygunluk Derecesi)
Sn : Saniye
STA : St. Andrew's Junior High School, Toronto
TRE : Trent University, Peterborough, Ontario
TÜKVS : Toronto Üniversitesi Kıyaslama Veri Seti (Toronto Benchmark)
UTA : Faculty of Arts and Sciences, University of Toronto
UTCC : University of the Thai Chamber of Commerce
UTE : Faculty of Engineering, University of Toronto
ÜSÇP : Üniversite Sınav Çizelgeleme Problem(ler)i
YAA : Yerçekimsel Arama Algoritması (Gravitational Search Algorithm)
YBS : Yapay Bağışıklık Sistemi (Artificial Immune System)
YOR : York Mills Collegiate Institute, Toronto
YSA : Yapay Sinir Ağları (Artificial Neural Networks)

GİRİŞ

Konkav bir çözüm uzayına ve çok sayıda yerel optimum değerine sahip olan problem çeşitlerinden biri olan çizelgeleme problemleri, bilgisayar teknolojilerinin gelişimine kadar manuel olarak ya da kısıtlı geleneksel yöntemlerle çözülmekteydi. Elde edilen bu çözümler de çoğu zaman tatminkar sonuçlar üretememekteydi. Bilgisayar teknolojilerinin gelişimiyle birlikte çizelgeleme problemlerinin çözümü için de çok çeşitli yöntemler geliştirilmiştir ve bu çözüm yöntemleri her geçen gün farklılaşarak artmaktadır. Literatürde, çizelgeleme problemlerinin çözümü için yaygın olarak sezgisel yöntemler kullanılmaktadır.

Sezgisel yöntemlerin gelişimi de bilgisayarların gelişimiyle birlikte üssel olarak hızlanmıştır. Bu gelişimin temelinde bilgisayarların, insanların yapamayacağı kadar büyük boyutlu çözüm uzaylarının ve hesaplamaların bulunduğu problemlerde kolaylıkla sonuçlar elde edebilmesi ve hesaplama zamanlarını hızlandırması bulunmaktadır. Bilgisayarlar her ne kadar hızlı hesaplamalar yapabilse ve çözüm zamanlarını kısaltsa da mevcut sezgisel yöntemler problemlerin optimum çözümlerini bulma konusunda garanti verememektedir. Çünkü, sezgisel yöntemlerin çözüm süreçlerinde rastlantısal seçimlerin etkin olarak kullanılmasından dolayı optimum çözümün garanti edilmesi mümkün olmamaktadır. Bundan dolayı da, elde edilen çözümler için genellikle "bilinen en iyi çözüm" ifadesi kullanılmaktadır. Araştırmacılar tarafından geliştirilen sezgisel yöntemlerin en iyi çözüme ulaşma durumu genellikle diğer yöntemlerden elde edilen sonuçlarla ya da bilinen en iyi sonuçlarla karşılaştırılarak yapılmaktadır.

Literatürde, geliştirilmiş olan çok çeşitli sezgisel yöntemler bulunmaktadır. Geliştirilmiş olan bu sezgisel yöntemlerin neredeyse hepsinde hesaplama süreçleri doğadan esinlenilerek gerçekleştirilmiştir. Örneğin, en bilinen sezgisel yöntemlerden biri olan genetik algoritmaların geliştirilmesinde, evrim süreçlerinden ve genetik biliminden esinlenilmiştir. Doğal süreçler ve doğada bulunan canlıların davranışları

insanlar tarafından gözlemlenmiş ve bu süreçlerin belirli prensiplere dayanarak en iyiyi gerçekleştirme eğiliminde olduğu tespit edilmiştir. Bundan dolayı, sezgisel yöntemlerde yoğun olarak doğadan faydalanılmaktadır. Herhangi bir görme duyusu bulunmayan karıncalar, yuvaları ile yiyecek kaynağı arasındaki en kısa yolu salgıladıkları feromon salgısı sayesinde kolaylıkla bulabilmektedirler. Karıncalarda olduğu gibi tüm hayvanlarda da farklı durumlar için davranışsal olarak en iyiyi bulmayı sağlayan bir özellik bulunmaktadır. Benzer durum doğal olaylar ya da evrim süreçleri için de geçerlidir. Evrim teorisinde, zayıf olan bireyler ya da özellikler elenip yok olurlar ve güçlü bireyler hayatta kalarak genlerini gelecek nesillere aktarırlar. Bu şekilde devam eden süreç, sürekli olarak, mevcut koşullar için en uyumlu ve en iyi bireyleri otomatik olarak ortamda tutar. Gözlemlenebilecek ve modellenebilecek bu kadar çok sistem olmasından dolayı da literatürde yer alan ve geliştirilmiş olan çok çeşitli sezgisel yöntemler bulunmaktadır.

Literatürde, bu sezgisel yöntemlerin çoğu, en az bir çizelgeleme probleminin çözümü için kullanılmıştır. Sınav çizelgeleme problemlerinde de diğer problemlerde olduğu gibi en sık başvurulan yöntem, genetik algoritmalar yöntemidir. Ana hatlarıyla genetik algoritmalar yapısı aynı olsa da farklı algoritmalarda kullanılan farklı yaklaşımlarla genetik algoritma çözümleri daha da iyileştirilebilmektedir. Bundan dolayı genetik algoritmalar sürekli çalışılan bir alan olarak güncelliğini korumaktadır.

Mevcut sezgisel yöntemler kullanılarak üniversite sınav çizelgeleme problemlerinin çözümleri üzerine çalışılmış olmasına karşın, literatürde, geliştirilmiş olan bir uzman sistem uygulamasının bulunmadığı görülmüştür. Bundan dolayı ve bu tez çalışmasında da kullanılacağından dolayı ilk bölümde uzman sistemler ele alınmıştır. Çalışmanın ikinci bölümünde en sık kullanılan sezgisel yöntemler aktarılmıştır. Bu bölümde ayrıca, tez çalışmasında önerilen uzman sistemle elde edilen sonuçların iyileştirilmesi için uygulamaların ikinci kısımda genetik algoritmalar kullanılacağından dolayı, genetik algoritmalar daha ayrıntılı bir biçimde incelenmiştir. Üçüncü bölümde, sınav çizelgeleme problemlerinin çözümü için literatürde yer alan çalışmalara ait bir literatür taraması verilerek, üniversite sınav

çizelgeleme problemi ve bileşenlerine ait detaylı bir anlatım gerçekleştirilmiştir. Dördüncü bölümde, önerilmekte olan üniversite sınav çizelgeleme problemi çözüm yaklaşımı detaylı olarak aktarılmaktadır. Önerilen sınav çizelgeleme algoritması iki kısımdan oluşmaktadır. İlk kısımda, uzman sistem kullanılarak sınav kuşaklarına atanacak olan ders grupları belirlenmiş ve ikinci kısımda da genetik algoritmalar kullanılarak bu sınav kuşaklarının optimum şekilde düzenlenmesi gerçekleştirilmiştir. Tez çalışmasının beşinci ve son bölümünde de önerilmekte olan üniversite sınav çizelgeleme algoritmasının literatürde yer alan test problemlerine uygulanması ve bu uygulamalardan elde edilen sonuçlara yer verilmektedir. Sonuç bölümünde de tez çalışması boyunca edinilen bilgilerle ve uygulamalardan elde edilen sonuçlarla, önerilen yöntemin etkinliği yorumlanarak değerlendirilmiştir.

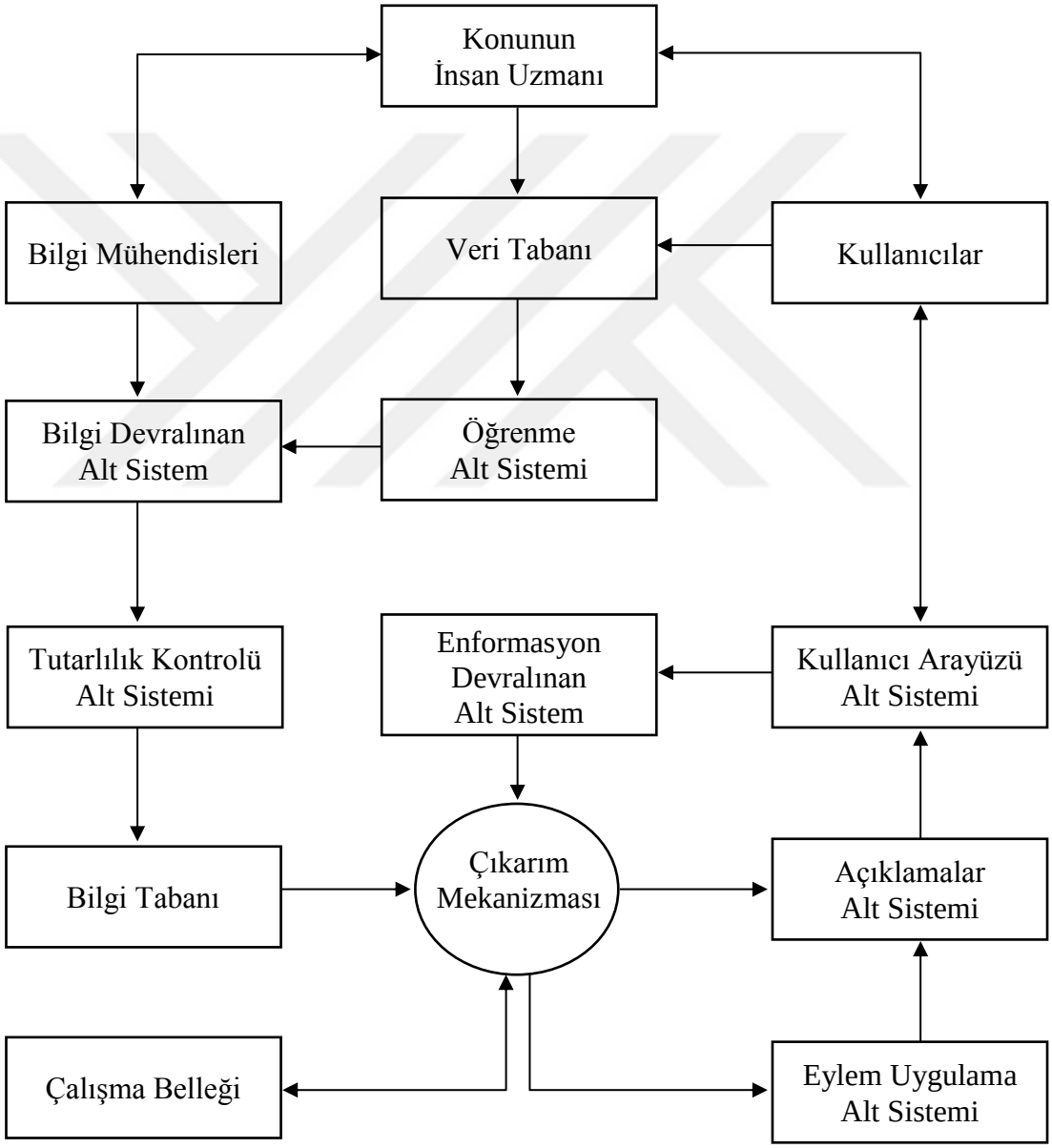
1. UZMAN SİSTEMLER

Uzman Sistemler, belirli bir uzmanlık gerektiren problemlerin çözümünde uzman görüşlerine ve kurallara dayalı olarak oluşturulmuş, insan davranışlarını taklit eden ve bir nevi insan gibi düşünebilen sistemlerdir. Uzman sistemlerin en önemli özelliği, uzman görüşlerinden elde edilmiş kurallara dayalı olarak oluşturulmuş programlama ile bu kuralların dışına çıkmadan problemin çözümüne gitmesidir. Bilgisayar yardımı ile programlanmış bir uzman sistemin en önemli avantajı, insandan daha hızlı muhakeme yeteneğine sahip olduğundan dolayı problemlerin çözümüne insandan daha hızlı sürede ulaşabiliyor olmasıdır. "Başka bir ifadeyle, karar verici her konunun uzmanı olamaz, olsa bile insanoğlunun hata yapabilme ve yavaş işlem yapma zafiyetleri söz konusudur." (Cura, 2009: 213) Uzman sistemler, özellikle insanların her konuda uzman olamayacağı ve karmaşık durumlar içeren problemlerin çözümünde büyük avantajlar sağlamaktadır.

Farklı problemlerin çözümü için geliştirilmiş olan uzman sistemlerin yapısı birbirinden farklı olsa da uzman sistemlerin tasarımında genel olarak aynı adımlar takip edilmektedir. Uzman sistem geliştirme süreci, mevcut ihtiyaçların belirlenmesi, bilgi edinimi, uzman sistem bileşenlerinin oluşturulması, sonuçların uygulanması ve bakım ve gözden geçirme prosedürlerinin formüle edilmesi adımlarını içermektedir (Abraham, 2005: 911). Uzman bilgisi ve görüşünü gerektiren bir problemin çözümü için öncelikle probleme ait ihtiyaçların ve elde edilmek istenen sonuçların belirlenmesi gerekmektedir. Problem ve probleme ait gereksinimler belirlendikten sonra ise uzman sistem tasarlanır ve çözüm süreci için bilgiler elde edilir. Son olarak, elde edinilen bilgiler, gereksinimler ve uzman görüşleri doğrultusunda uzman sistem oluşturulur. Uzman sistemin oluşturulmasından sonra sistem çalıştırılarak elde edilen sonuçlar uygulanır ve değerlendirilir. Uzman sistem, sonuçların değerlendirilmesi ve ortaya çıkan yeni gereksinimler ile birlikte gözden geçirilerek düzenlenir ve bu şekilde sisteme devam edilir.

1.1. UZMAN SİSTEMLERİN TEMEL BİLEŞENLERİ

Uzman sistemler ya da kural tabanlı sistemler, çıkarım yapılabilen ve kurallarla oluşturulabilen tüm sistemlere entegre edilebilir ve geliştirilebilirler. Çok fazla alana entegre edilebilir olmasıyla birlikte, uzman sistemlerin oluşturulmasında belli temel bileşenler bulunur. Uzman sistemlerin temel bileşenleri aşağıda yer alan Şekil 1.1'de gösterilmektedir.



Şekil 1.1: Uzman Sistemlerin Temel Bileşenleri ve Akış Şeması (Castillo v.d., 2012: 10).

Uzman sistemlerin temel bileşenleri ve bu bileşenlerin temel işleyiş ve akışları genel olarak Şekil 1.1'de gösterildiği şekilde gerçekleşmektedir. Şekilde gösterilmekte olan akış şeması, farklı problemler için geliştirilmiş olan farklı uzman sistemlerde değişiklik gösterebilmektedir. Bu kısımda, uzman sistemlerin temel bileşenleri ve sistem içindeki görevleri incelenecektir.

İnsan Uzmanlar: İnsan uzmanlar, uzman sistemlerin en önemli parçalarından biridir. Uzman sistemde kullanılan kurallar, bilgiler ve çıkarımlar, insan uzmanlardan elde edilmektedir. İnsan uzmanlardan elde edilen bilgiler, bilgi mühendisleri, veri tabanı ve kullanıcılara iletilir. İnsan uzmanlardan alınan bilgiler son olarak, bilgi tabanı vasıtasıyla çıkarım mekanizmasına iletilirler.

Kullanıcılar ve Kullanıcı Arayüzü: Uzman sistemlerde, kullanıcılar direkt olarak sistemle bağlantı içerisinde bulunmamaktadır. Sistemin kullanıcılarının, çıkarım mekanizmasına ve çıkarım mekanizmasından elde edilen bilgilere erişimi için kullanıcı arayüzü kullanılmaktadır. Kullanıcı arayüzü, sistemde bulunan ve standart kullanıcıların algılayamayacağı dilleri ve sistem çıktılarını, kullanıcının algılayabileceği şekilde dönüştürerek kullanıcılara aktaran bir mekanizmadır.

Bilgi Tabanı: Bilgi tabanları, sistemde konunun uzmanlarıyla da iletişimde olan bilgi mühendisleri tarafından oluşturulurlar. Bilgi mühendisleri, insan uzman bilgilerini, kurallara ve stratejilere dönüştürerek bilgi tabanlarında saklarlar. Uzman sistemler de bilgi tabanları yardımıyla çıkarımlar yaparak, elde ettiği sonuçları kullanıcılara iletirler. Sisteme yeni kurallar ve bilgilerin eklenebilmesi, gerekli güncellemelerin gerçekleştirilebilmesi için bilgi tabanları esnek olarak tasarlanmalıdırlar.

Veri Tabanı: Veri tabanları, uzman sistemlerde bilgi tabanlarının desteklenmesi amacıyla kullanılmaktadır. Veri tabanları dinamik yapılardır ve sistem içerisinde oluşan bilgilerin ve sonuçların depolanması için kullanılırlar. Veri tabanları, diğer veri tabanları veya programlarla bağlantılı olarak çalıştırılarak, daha etkin ve verimli bir sistem elde edilebilir.

Çıkarım Mekanizması: Çıkarım mekanizmasının amacı, bilgi tabanında bulunan mevcut bilgileri kullanarak problemlerin çözümlerini elde etmektir. Çıkarım mekanizmalarında tek bir çıkarım yöntemi kullanılmamaktadır. Literatürde farklı problemlerin çözümleri için geliştirilmiş çok çeşitli çıkarım yöntemleri bulunmaktadır. En sık kullanılan çıkarım yöntemleri; sezgisel, deneysel, benzetim, tümden gelim, tüme varım vb.

Burada incelenmiş olan uzman sistemlere ait beş bileşenden farklı olarak, problemlerin çeşitlerine göre farklı bileşenler eklenerek, uzman sistemler özelleştirilebilir ve geliştirilebilir. Sağlık sektörü için geliştirilecek olan bir uzman sistem ile bankacılık ve finans alanında geliştirilecek bir uzman sistemin gereklilikleri ve çıktıları birbirinden farklı olacaktır. Uzman sistemler kural tabanlı sistemler olduklarından ve amaçlara göre geliştirildiklerinden dolayı, uzman sistemlerin asıl gereklilikleri ve tasarımı, konunun insan uzmanlarıyla ve kullanıcılarla görüşerek belirlenebilir.

1.2. UZMAN SİSTEMLERİN UYGULAMA ALANLARI

Bilgisayarlar günlük hayatımıza girdiğinden beri yaşamımızı oldukça kolaylaştırmaktadır. Bundan dolayı, bilgisayar kullanımının yaygınlaşmasıyla birlikte bilgisayar destekli uygulamalar da her geçen gün artmaktadır. Sistemlerde bulunan uzun ve maliyetli süreçler, bilgisayar destekli tasarımların geliştirilmesiyle birlikte ortadan kaldırılmakta ve yerlerine daha kısa süren ve daha az maliyetli tasarımlar getirilebilmektedir. Bilgisayarların kullanımının bu şekilde yaygınlaşmasıyla birlikte, uzman sistemlerin kullanımı da bir gereklilik haline gelmiş ve günümüzde oldukça artmıştır.

Bilgisayarlar ya da bilgisayar destekli tasarımlar, günümüz sistemlerinin neredeyse tamamında aktif olarak rol almaktadır. Bilgisayarların ve bilgi teknolojilerinin kullanılabilirliğinin artmasıyla birlikte, sistemlerin uzayan süreçlerini ve maliyetlerini indirgeyebilmek için oluşturulmuş olan uzman sistemlerden faydalanılmaktadır. Uzman sistemler, günümüz sistemlerinin içerisinde, küçük ya da büyük parçalar halinde de olsa yaygın kullanım görmektedir. Aşağıda yer alan Tablo

1.1'de, literatürde uzman sistemlerin en yaygın olarak kullanıldığı sektörler, alanlar ve uygulamalar (Liao, 2004; Dasic, 2011) verilmektedir.

Tablo 1.1: Uzman Sistemlerin Uygulama Alanları.

Adli Sistemler	Karar Destek Sistemleri
Askeri Uygulamalar	Kaynak Kullanımı
Atık Yönetimi	Kimya Sanayi
Bankacılık ve Finans	Maden Endüstrisi
Bilgi Sistemleri ve Yönetimi	Risk Değerlendirme
Bilgisayar Destekli Tasarımlar	Robotik Sistemler
Çizelgeleme	Sağlık Sektörü
Eğitim Sektörü	Sistem Geliştirme
Eğitim Simülatörleri	Stratejik Planlama
Ekolojik Sistemler	Süreç Yönetimi
Enerji Sistemleri	Tarım Sektörü
Eş Zamanlı Sistemler	Telekomünikasyon
Güvenlik Sistemleri	Ulaştırma Sistemleri
Hata Tanılama	Üretim Planlama
Havacılık Sektörü	Yazılım Değerlendirme
İnsan Kaynakları	Yerbilimi
İnşaat Sektörü	Yorumlama Mekanizmaları

1.3. UZMAN SİSTEMLERİN AVANTAJLARI

Uzman sistemler, kullanıcılarına, standart bir uzman yardımıyla elde edilen bir problem çözümüne nazaran çok büyük avantajlar sağlamaktadır. Sağlamış olduğu avantajlar çeşitli yönlerden değerlendirilebilmektedir ve bu kısımda, uzman sistemlerin sağlamış olduğu avantajlar ayrıntılı olarak değerlendirilecektir.

Veri Saklama: Uzman sistemlerde bir veri tabanı oluşturularak bilgiler bu veri tabanında saklanabilir. Veri tabanına büyük miktarlarda veriler eklenerek bu veriler güncel tutulabilir ve aynı zamanda sıradan bir insandan daha fazla bilgi

saklanabilmektedir (Kaleem, 2011: 98). Büyük boyutlardaki veriler, bir insanın algısına nazaran veri tabanlarına hızlı bir şekilde yüklenerek kullanılabilecektir. Veri tabanları sayesinde de uzman sistem etkin ve verimli bir şekilde çalıştırılabilecektir.

Verimlilik: Uzman sistemler kuruluş aşamasında yüksek maliyetler ortaya çıkarmasına karşın, uzun vadede değerlendirildiğinde; personel maliyetlerini düşürdüğü ve verimliliği arttırdığı görülmektedir. Uzman sistemlerin bakımı ve kurulumu pahalı olmasına karşın, çalıştırılması pahalı değildir (Tomar ve Saxena, 2011: 25.3). Uzman sistemlerin ortaya çıkardığı maliyet, kullanım süresine yayılarak, personel ve diğer maliyetler üzerinde sağlamış olduğu avantajlarla birlikte değerlendirildiği taktirde, uzman sistemin verimlilik ve maliyet avantajı ortaya çıkacaktır.

Uyarlanabilirlik ve Çoğaltılabilirlik: Geliştirilmiş olan bir uzman sistem, ortaya çıkan yeni gereksinimlerle birlikte düzenlenebilir ya da bir firmanın farklı işleri için, küçük değişikliklerle, farklı sistemler için de uyarlanabilir. Benzer şekilde, geliştirilmiş olan bir uzman sistem, farklı sistemlere de entegre edilerek, birden çok sistem içinde çalıştırılabilmektedir. Uzman sistemlerin bu avantajı, personel gereksinimi ve personel maliyetlerini de azaltmaktadır.

Tutarlılık: Bilgisayar tabanlı olarak oluşturulmuş olan bir uzman sistemde, görelilikler ve kişisel yargılar olmayacağından dolayı, insan bir uzmana nazaran daha eşit değerlendirmeler gerçekleştirecektir. Uzman Sistemler, insan uzmanlara göre daha tutarlı bir sistem geliştirme imkanına sahiptir (Abraham, 2005: 910). İnsan uzmanlar, kişisel yargılamalar, güncel kişisel durumlar ya da sisteme dahil olan birimlerle alakalı ek bilgiler neticesinde, kendi kişisel görüşlerindeki yanlılıkları içererek, aynı durumlar için dahi olsa farklı sonuçlar üretebilmekte ve duygusal değerlendirmeler gerçekleştirebilmektedir. Uzman sistemlerde bu kişisellikler bulunmayacağından dolayı, tamamen tutarlı bir sistem geliştirilmesi mümkündür.

Riski Azaltması: İnsanların çalışabilmesi için belirli koşullar gerekmektedir. İnsan uzmanlar, insan sağlığına zararlı ya da insan haklarına aykırı ortamlarda çalıştırılamaz. Fakat uzman sistemler, insan için tehlikeli, zararlı ya da tatsız

ortamlarda çalışabilir (Goodarzi, 2008: 304). Uzman sistemler, insan uzmanları riske atmaması ve insan uzmanların çalışamayacağı ortamlarda da çalıştırılabilmesi sayesinde hem verimliliği arttıracak, hem maliyetleri düşürecek, hem de çalıştırılan insan uzmanların sağlığının riske atılmasını engelleyecektir.

Kalıcılık: İnsan uzmanların emekli olabilmesine, ayrılabilmesine ya da ölebilmesine karşın, uzman sistemler kalıcıdır (Gonciarz, 2014: 276). Firmalar, personeline yatırım yaparak, uzmanlık edinmeleri için gerekli eğitimlere ve programlara gönderirler. Fakat yetiştirilmiş olan uzman personelin işten ayrılması, ölmesi ya da emekli olması durumunda, yapılmış olan yatırım da personelle birlikte gidecektir. Ayrılan uzman personel yerine yeni bir uzman personel alınacak ya da mevcut personellerden bir tanesi için yeniden yatırım yapılacaktır. Tüm bu maliyetler, kalıcı bir uzman sistem yatırımı yapılarak azaltılabilecektir.

Hızlı Tepki Süresi: İnsan uzmanların tüm bilgileri değerlendirmesi, karşılaştırması ve tüm kurallara uygun sonuçlar elde etmesi zaman alırken, uzman sistemlerin bu işlemleri gerçekleştirmesi çok daha hızlı bir şekilde meydana gelmektedir. Özellikle birden fazla uzmanın ya da personelin birlikte çalışması gereken durumlarda, hızlı tepki süresi daha avantajlı bir hale gelmektedir.

Uzmanlık Tabanı: İnsan uzmanların bilgi ve deneyim edinme yeteneği sınırlıdır. Bir insanın her konuda uzman olması ve değerlendirmeler yaparak sonuçlar elde etmesi imkansızdır. Oysa bir uzman sisteme, bir insan uzmandan çok daha fazla uzmanlık alanı entegre edilerek, tamamına tek bir uzman sistemle erişim sağlanabilecektir.

Güvenilirlik: Uzman sistemlerin insan uzmanlara göre daha tutarlı olması dolayısıyla elde edilen sonuçlar da, sonuçları kullanacak olan bireyler tarafından daha güvenilir olarak karşılanacaktır. Güvenilir sonuçlar riski, yargılanabilirliği ve sorgulanabilirliği azaltarak süreçleri de kısaltacaktır.

İstenilen Bilgilere Erişim: Uzman sistemlerin çalışması sırasında, ara değerlendirmelere erişilebilir ya da sonuçlar istenilen formatta elde edilebilir. Aynı zamanda uzman sistemlerin çalışması sırasında, gerekli durumlarda sisteme müdahale edilerek bilgiler düzenlenebilmektedir.

Öğretme: Geliştirilmiş olan uzman sisteme eklenecek gerekli bilgilerle birlikte, uzmanlık gerektiren konu hakkında bilgisi olmayan kullanıcılara da, açıklamaları ve nedenleri ile birlikte konuyu öğretebilmesi sağlanacaktır.

1.4. UZMAN SİSTEMLERİN DEZAVANTAJLARI

Uzman sistemlerin avantajları olduğu kadar dezavantajları da vardır. Bu dezavantajlar, uzman sistemlerin kullanımını kısıtlayabilmektedir. Bazı alanlarda ise uzman sistemler, aşağıda belirtilen dezavantajlardan dolayı kullanılamamaktadır. Uzman sistemlerin en belirgin dezavantajları aşağıda ayrıntılı olarak incelenmektedir.

Amaç: Bir uzman sistem belirli bir amaç için tasarlanır ve farklı bir amaç için kullanılamaz (Joy ve Nkwachukwu, 2011: 246). Uzman sistemler kural tabanlı olarak tasarlandıklarından dolayı, belirli bilgileri değerlendirerek, yalnızca öğretilmiş olan kurallara göre değerlendirmeler yapabilirler. Öğretilmemiş olan kurallara dayalı farklı değerlendirmelerde bulunamazlar. Fakat insan uzmanlar, bilgileri yorumlayarak, çeşitli değerlendirmeler ve amaçlara uygun bilgiler üretebilmektedir.

Maliyet: Uzman sistemler, yüksek tasarım maliyetleri içerecek ve gerçekleşmeleri için uzmanlaşmış bir teknoloji, uzun çalışma zamanları ve özellikle büyük boyutlu verilerle çalışırken güvenilirliğinin test edilmesi için uzun periyotlar ile yüksek derecede özelleşmiş bir personel gerekecektir (Elena v.d., 2012). Uzman sistemin gerçekleştirmesinin istendiği tüm problem çözümlerine ait kural tabanlarının oluşturulması, verilerin işlenmesi ve tasarlanmış olan uzman sistemin doğru sonuçlar elde edip etmediğinin test edilmesi, hayli uzun zaman alacaktır. Bu zaman boyunca çalıştırılan personel(ler)in ve kullanılan teknolojilerin, işletmelere ekstra maliyetleri olacaktır. Özellikle komplike sistemler için bu süreçlerin uzaması ya da maliyetlerin artması, uzman sistemlerin tercih edilmemesine neden olabilmektedir.

Öğrenme: İnsan uzmanlar gelişmeleri, yenilikleri ve yeni bilgileri takip edebilmektedir. Onlara mevcut uzmanlıklarının üzerine her bir bilgiyi öğrenmesi için ekstradan talimat ya da eğitim verilmesi çoğunlukla gerekmemektedir. Oysa oluşturulmuş olan bir uzman sistem, gelişmeleri ve yeni çıkan uzman sistem

kurallarını otomatik olarak öğrenemeyeceğinden dolayı, dışarıdan bir düzeltme ya da güncelleme ile uzman sistemin yenilenmesi gerekecektir. İnsan uzmanlar değişen çevreye otomatik olarak adapte olabilirler, uzman sistemler ise kesinlikle açıkça güncellenmelidirler (Tomar ve Saxena, 2011: 25.3).

Hata Yapabilmesi: Uzman sistemler her ne kadar insan uzmanlara göre daha hatasız olarak çalışsa da, uzman sistemlerde tanımlanmamış durumlar ya da beklenmeyen sonuçlar gibi kendisine hiç öğretilmemiş, bilinmeyen durumlar karşısında ya da başka sebeplerle hatalı sonuçlar üretebilmektedir. İnsan uzmanlar bu tip durumlarda kişisel tecrübelerine dayanarak tercihler yaparak durumları kotarabilirken, uzman sistemler tamamen yanlış kararlar alabilmektedirler. Bu hatalar da biraz maliyetli olabilmektedir (Gonciarz, 2014: 277).

Uzmanlığın Tamamen Aktarılamaması: İnsan uzmanların uzmanlıklarının tamamen bir uzman sisteme aktarılması çok zordur. Bazı uzman sistem uygulamaları içinse imkansızdır. Bu aktarılamama durumu, ifadelendirememe ya da tanımlayamama gibi sebeplerden kaynaklanabildiği gibi, insan uzmanların işsiz kalma kaygısı, makineleşme korkusu ve benzeri çeşitli faktörlerden de kaynaklanabilmektedir.

Kişisel Değerlendirme: İnsan uzmanlar, tanımlanmamış olan durumlar, kurallar ya da beklenmedik olaylar karşısında kişisel tecrübelerine dayanarak sonuçlar elde eder ve kararlar verebilirken, uzman sistemler yalnızca öğretilmiş olan kurallarla çalıştıklarından dolayı, kendi bilgi tabanlarının dışına çıkamazlar. Uzman sistemlere, tamamen beklenmedik olaylar karşısında insan uzmanlar gibi çıkarım yapmanın ve kendi tecrübelerine dayalı sonuçlar elde etmenin öğretilmesi henüz gerçekleştirilememiştir. Gerçekleştirilebilmiş olan kısmı da, tahmin edilen ya da tahmin edilebilen durumları içermektedir ki bu da uzman sisteme kural olarak girileceğinden dolayı bu başlığın dışına çıkmaktadır.

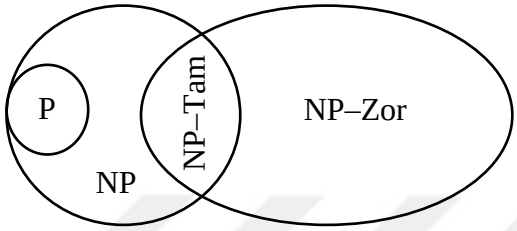
2. SEZGİSEL YÖNTEMLER

Çözümlerinde sayısal yöntemlerin kullanıldığı problemler çok çeşitlidirler. Tek değişkenli ve belirli kısıtlar içeren basit bir matematik probleminin çözümü çok kolaylıkla bulunabilirlerken, değişken sayısı arttıkça, her değişkenin problemin çözümü üzerindeki etkilerinin değerlendirilmesi gerekmektedir. Sonlu çözüm uzayına sahip problemlerde de benzer şekilde çözüm kolaylıkla elde edilebilmektedir. Fakat problemlerde yer alan değişken sayısı arttıkça ve bu değişkenlerden elde edilen çözüm kombinasyonlarının sonsuz olmasıyla birlikte, problemlerin optimum çözümünü elde etmek oldukça zorlaşmaktadır.

Sayısal yöntemlerde, kurulan modeller ile kesin olarak optimum çözüm değerleri bulunabildiği gibi çok sayıda değişkene ve sonsuz çözüm uzayına sahip, optimal çözüm değerine ulaşılamayan problemler de yer almaktadır. Bu tip problemlerde, genellikle sonsuz sayıda çözüm ve çok sayıda yerel minimum veya maksimum sonuç veren değişken değerleri kombinasyonları bulunduğundan dolayı optimum sonucun elde edilmesi zorlaşmaktadır. Bu tip problemlerde optimum çözümün elde edilmesinin zor olmasının en büyük nedeni, geliştirilmiş olan çözüm yöntemlerinin bu yerel minimumlara veya yerel maksimumlara takılarak, global minimumları yada global maksimumları bulamamasından kaynaklanmaktadır. Optimum çözümü elde etmenin zor olduğu bu tip problemler için literatürde geliştirilmiş çok sayıda sezgisel yöntem bulunmaktadır. Türkçede "sezgisel" olarak kullanılan kelime, İngilizce literatürde "heuristic" olarak kullanılmaktadır. Anlamı incelendiğinde ise; "keşfe yarayan, anlamaya vesile olan, deneye dayalı, deneme yanılma yolunu kullanan" şeklinde olduğu görülmektedir.

Sezgisel yöntemlerle çözümlerinin arandığı, çözümü zor olan bu problemler; literatürde NP–Zor ve NP-Tam (NP–Hard ve NP-Complete) problemler olarak geçmektedir. NP kısaltmasının açılımı da; "Non-deterministic Polynomial-time" şeklindedir. Yani, deterministik olmayan polinom zamanlı problemler olarak Türkçe

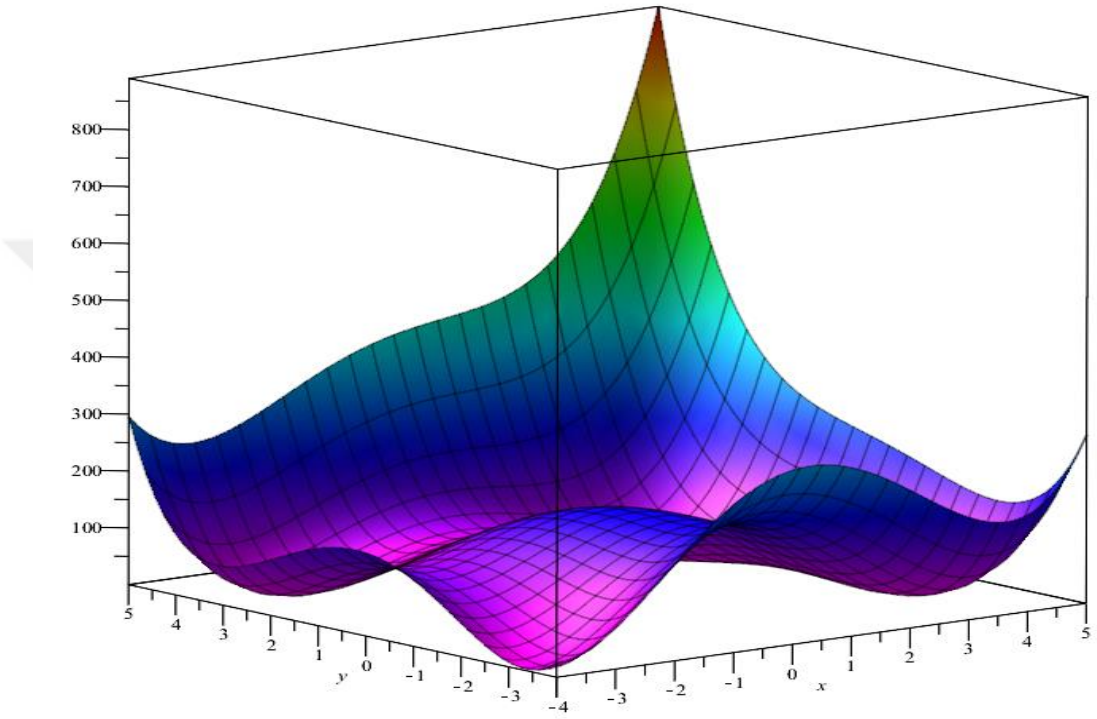
de kullanılabilmektedir. Bu problemler kümelerine ve birbirleri ile olan ilişkilerine ait genel şema aşağıda verilmiş olan Şekil 2.1'de görülmektedir. Şemada da görülebileceği gibi; hem NP hem de NP–Zor olan problem tipleri, NP–Tam problemler olarak sınıflandırılmaktadır ve diğerlerine göre çözümleri en zor olan problem tipleri olarak da bilinmektedir.



Şekil 2.1: NP Problem Kümeleri.

Çözüm uzayı geniş ve çözümü zor olan NP sınıfı problemlerin çözülmesine yardımcı olan sezgisel yöntemlerin çözüme giden yaklaşımları da kelime anlamı ile uyuşmaktadır. Literatürde yer alan sezgisel yöntemler, genellikle rastgele bir başlangıç çözümü kullanarak çözümlerine başlamakta ve çözümün devam eden iterasyonlarında sürekli olarak bu çözümü iyileştirerek, en iyi çözüme gitmeyi hedeflemektedirler. Bu başlangıç ve bitiş adımları arasında sonuca giderken kullanılan yöntemler farklılık göstermektedir. Bu yöntemlerde sürekli olarak çözümün iyileştirilmesi ve kötü olan çözümlerin elenmesi, yöntemlerin yerel minimum ya da maksimum değerlerde takılmasına yol açacaktır. Bu yerel minimum ya da yerel maksimum değerlere takılmanın engellenmesi için zaman zaman elde edilen kötü çözümlere de izin verilmektedir. Burada açıklanan ifadeler aşağıda yer alan Şekil 2.2 ile daha net bir biçimde ifade edilebilecektir. Şekilde üç boyutlu bir fonksiyonun grafik çiziminden bir bölüm gösterilmektedir. Minimum noktanın arandığı bu problemde, rastgele çözüme başlanan noktanın $(x, y) = (5, -5)$ olduğunu farz edelim. Bu nokta etrafında bulunan yakın noktalar aranarak, her seferinde bulunan en iyi noktanın saklandığı bir algoritma olduğunu düşünürsek; çözüm $(x, y) = (3, -3)$ bölgesinde bir noktada son bulacak, yakında başka daha iyi nokta bulamayacağından dolayı optimum çözüm olarak sonuçlandırılacaktır. Oysa $(x, y) = (-4, -3)$ civarındaki nokta daha minimum bir çözüm değeri verebilecektir. Bu şekilde çok değişkenli, sonsuz çözüme sahip ve çok fazla yerel minimum ya da

yerel maksimum değerine sahip problemlerin çözümünde, bu tip yerele takılma sorunları aşılmaya çalışılmaktadır. Elde edilen kötü çözümlerin saklanmasına da izin verilmesi ya da literatürde yer alan geliştirilmiş farklı çözüm teknikleriyle bu sorun aşılmaya çalışılmaktadır.



Şekil 2.2: Yerel Minimuma Sahip Örnek Fonksiyon Grafiği.

Problemlerin çözümü için sezgisel yöntemlerin geliştirilmesinde, genel olarak doğadan ve doğal olaylardan esinlenilmektedir. Doğada gerçekleşen olaylar, mükemmel veya mükemmele yakın düzeyde seçilimsel olarak devam etmekte ve en iyi olmayan bir durum giderek optimum duruma erişebilmektedir. Örneğin, Darwin'den bu yana kullanılan evrim teorisi ve genetik çalışmalarından yola çıkılarak, "Genetik Algoritmalar" çözüm yöntemi geliştirilmiştir. Karıncaların doğada yiyecek ararlarken kullandıkları feromon salgısı ile yiyeceğe ulaşmada kat edilen yolun en kısa hale getirilmesinden esinlenilerek, "Karınca Kolonisi Optimizasyonu" yöntemi geliştirilmiştir. Bunlara benzer şekilde diğer sezgisel yöntemler de, doğal olayların gerçekleşme mantıklarını taklit ederek çözüme ulaşmayı hedeflemektedirler. Bu yöntemler tek başlarına kullanılabildikleri gibi, farklı model kombinasyonları ile de çözüm uygulamaları literatürde yer almaktadır.

Özellikle bilgisayar teknolojilerinin de gelişmesiyle birlikte, en iyi çözümü elde etmenin zorlaştığı NP sınıfı problemlerde, bilgisayarlardan faydalanılmaktadır. NP sınıfı problemlerin çözümlerinde bilgisayarlardan faydalanılması ve çözüm yöntemlerinin bilgisayarların gelişimi ile paralel olarak ilerlemesi, bilgisayarların insanlardan daha hızlı hesaplamalar yapabilmesi ve karmaşık hesaplamaları doğru algoritmalarla hatasız bir şekilde gerçekleştirerek çözüme, bilgisayarların performansına da bağlı olarak olabildiğince hızlı ve doğru bir şekilde yaklaşabilmesidir.

Literatürde, çok fazla sayıda sezgisel yöntem bulunmasına karşın, tanıtılmış olan sezgisel yöntemlerin tamamı kullanım görmemektedir. Kullanımda kabul görmüş ve sıklıkla başvurulan sezgisel yöntemler ise belirli sayıdadır. Çalışmanın bu kısmında; NP sınıfı problemlerin çözümleri için geliştirilmiş olan yöntemlerden literatürde yer alan çalışmalarda sıklıkla başvurulan yöntemler ve çözüm algoritmaları kısaca aktarılmaya çalışılacaktır.

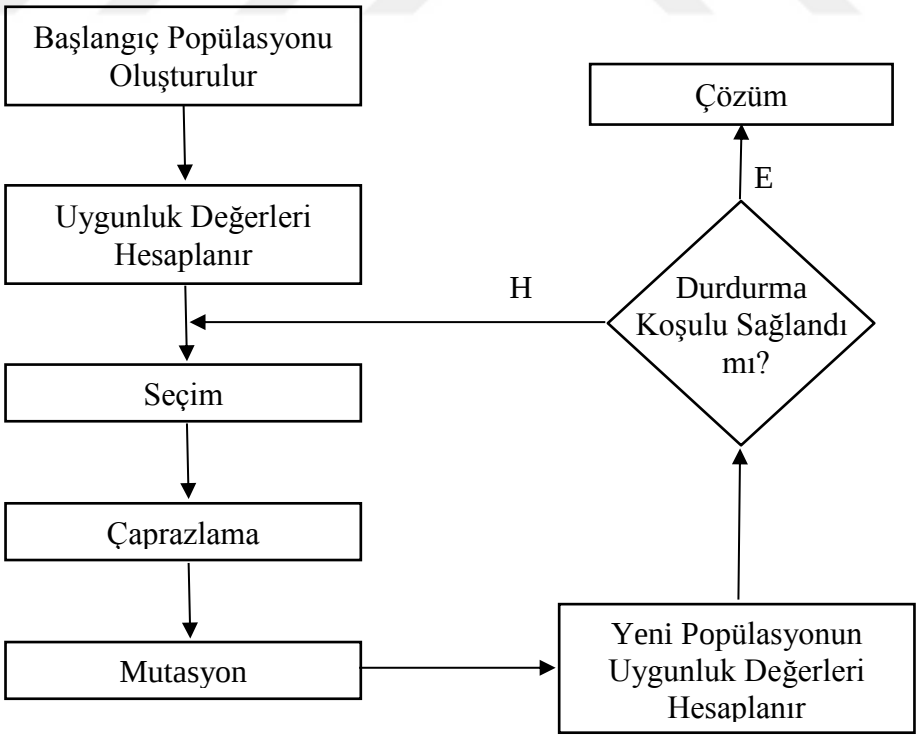
2.1. GENETİK ALGORİTMALAR

Genetik algoritma (Genetic Algorithm, GA) tabanlı çözümlerin temeli ilk olarak 1975 yılında John Holland (Holland, 1975) tarafından tanıtılmıştır. Yöntem, genetik biliminden faydalanarak problemleri modellemektedir. Problemlerde yer alan değişkenler ve çözümler, genler ve kromozomlar olarak ifade edilirler ve çözümlere, genetikte olduğu gibi her bir nesilde çaprazlamalarla daha da iyileştirilerek devam edilir.

Genetik algoritmalar yönteminin temeli, evrim teorisinde Darwin tarafından tanıtılmış olan “adaptasyon” ve “doğal seçilim” kavramlarına dayanmaktadır. Adaptasyon, canlıların içinde bulundukları çevre koşullarına göre ayakta kalabilmesini sağlayan genetik değişikliklerdir. Darwin, türlerin kökeni kitabının giriş bölümünde adaptasyon ve doğal seçilim üzerine “Her türün bireyleri muhtemelen hayatta kalabilenlerden çok daha fazla olarak doğarlar, bunun sonucu olarak da, var olmak için sıklıkla tekrarlanan bir mücadele vardır, hayatın karmaşık ve bazen de değişen koşulları altında herhangi bir canlı, kendisi için herhangi bir şekilde faydalı biraz değişikliğe uğrasa da daha iyi bir hayatta kalma şansı olacak ve

bu şekilde doğal olarak seçilecektir." (Darwin, 1859: 5) ifadelerini kullanmıştır. Darwin'in yapmış olduğu bu tanımlamada da görülebileceği gibi, değişen koşullara ayak uydurabilen canlılar yaşamakta, fakat değişikliğe uğramayan ve kendisini daha iyiye götürmeyen bireyler ise ölmektedir. Bu ifadeleri NP sınıfı problemler için yorumlayacak olursak; her bir iterasyonda uyumu en iyi olan çözümler popülasyonda kalırken, uyumu düşük olan çözümler ise popülasyondan elenmektedir. Gerçekleşen bu doğal seçilim ve çevre koşullarına olan adaptasyonlar ile, her bir iterasyonda daha da iyileşen popülasyon değerleri, bir süre sonra en mükemmel hale gelerek istenilen çözüm değerine götürebilmektedir.

Problemlerin çeşitlerine göre çözüm yapıları farklılık gösterse de genetik algoritmaların genel çözüm yapıları benzerdir. Aşağıda gösterilen Şekil 2.3'te bir genetik algoritma çözüm yapısı, ana akış hatlarıyla gösterilmektedir. Şekilde ana hatlarıyla görülen genetik algoritma çözüm yapısının bileşenleri ayrıntılı olarak incelenmeye çalışılacaktır.



Şekil 2.3: Genetik Algoritma Çözüm Yapısı.

2.1.1. Genetik Algoritmalarda Kullanılan Yapılar ve Tanımlar

Genetik algoritmaların çözümlenmesinde kullanılan terimler de genetikte kullanılan terimlerle benzerlikler içermektedir. Bu kısımda genetik algoritmaların daha iyi anlaşılabilmesi için, genetik algoritmalarda kullanılan terimler tanımlamalarıyla birlikte incelenecektir.

Popülasyon: Problemin çözümü için oluşturulmuş olan ve her biri probleme ait bir çözümü ifade eden bireyler topluluğudur. Problemin çözümü sırasında her bir iterasyonda, popülasyonda yer alan bireyler güncellenerek en iyi popülasyon aranır.

Kromozom: Problemde yer alan tüm değişkenleri ifade etmektedir. Popülasyon, aday çözümü temsil eden *n* adet kromozoma sahiptir; her bir kromozom optimize edilen parametrelerin sayısı kadar boyutlu bir gerçek değer vektörüdür (Moradi ve Abedini, 2012: 69).

Gen: Genetikte tek bir özelliği ifade eden gen terimi, genetik algoritmalarda da kromozom diziliminde yer alan tek bir özelliği, değişkeni ifade etmektedir.

Alel: Alel ifadesi genetikte, özellikleri belirleyen genlerin alternatiflerini ifade etmektedir. Genetik algoritmalarda da bu durum, problemde yer alan değişkenlerin alabileceği farklı değerleri ifade etmektedir.

Genotip: Bir canlının genetik yapısını ifade eden terimdir. Genetik algoritmalarda da bu ifade, çözümün ikilik düzende gösterimi olarak karşılık bulmaktadır.

Fenotip: Bir canlının genetik yapısının dış görünüşündeki karşılığı ifade eder. Genetik algoritmalarda da benzer şekilde genotipin yani ikilik düzendeki gösterimin karşılığını ifade etmektedir.

Uygunluk: Biyolojik anlamda uygunluk, kromozomların üreme verimliliğinin ölçüsü olan bir kalite değeridir ve genetik algoritmalarda da uygunluk, popülasyonda yer alan bireylere üreme özellikleri tahsis etmede kullanılır ve bu sayede de optimize edilecek iyiliğin bazı ölçüleri olarak hareket edilir (Mathew, t.y.:5).

Uygunluk Fonksiyonu: Genotip değerlerinin, probleme göre oluşturulmuş olan amaç fonksiyonu üzerindeki uygunluk değerlerinin hesaplandığı fonksiyondur. Literatürde genellikle, amaç fonksiyonu maksimizasyon olan problemlerde uygunluk fonksiyonu değeri aynen alınırken, amaç fonksiyonu minimizasyon olan problemlerde ise uygunluk fonksiyonu değerinin çarpmaya göre tersi alınarak kullanılmaktadır.

Ebeveyn Seçimi: Yeni oluşturulacak bireyler, çözümler için uygunluk fonksiyonundan elde edilen değerlere göre, ebeveyn çözümleri arasından rastgele olarak yapılacak seçimleri ifade eder. Uygunluk fonksiyonuna göre yapılan tercihlerle, iyi olan sonuçların seçilme şansı daha yüksek olacaktır. Literatürde yer alan farklı seçim teknikleri ilerleyen kısımlarda incelenecektir.

Çaprazlama: Rastlantısal olarak seçilmiş olan iki ebeveynin genetik kodlarının rastlantısal olarak seçilerek yeni oluşturulacak bireylere aktarılması ve bu şekilde ebeveynlerinden farklı genetik dizilime sahip yeni bireyler elde etmeyi ifade eder. Literatürde kullanılan farklı çaprazlama teknikleri yer almaktadır. Bu teknikler de ilerleyen kısımlarda incelenecektir.

Mutasyon: Problemlerin çözümünde yer alan bireylerin genlerinden bazılarının önceden belirlenmiş olan belirli bir yüzdeye göre rastlantısal olarak değiştirilmesini, bozulmasını ifade eder. Mutasyon, genetik algoritma tabanlı çözümlerde, özellikle ilerleyen iterasyonlarda, yerel minimuma takılmayı da engellemeye olanak sağlayabildiğinden dolayı tercih edilen bir durumdur.

2.1.2. Seçim Teknikleri

Seçim işlemi ve seçim teknikleri yalnızca genetik algoritmalarda değil sezgisel yöntemlerin birçoğunda kullanılmaktadır. Seçim işleminin sıklıkla kullanılan bir işlem olmasından da dolayı, farklı problemler için geliştirilmiş genetik algoritmalarda ve diğer sezgisel yöntemlerde kullanılan çok fazla seçim tekniği bulunmaktadır. Literatürde en sık kullanılan üç seçim tekniği; rastgele seçim, rulet tekerleği seçim ve turnuva seçim tekniğidir. Bu kısımda, literatürde en sık karşılaşılan seçim teknikleri aktarılacaktır.

2.1.2.1. Rastgele Seçim Tekniği

Rastgele seçim tekniği, en primitif seçim tekniğidir. Popülasyonda yer alan bireylerin ya da diğer bir ifadeyle seçilecek olan birimlerin tamamına eşit seçilme şansı verilmektedir. Rastgele seçim tekniğinde, popülasyonda bulunan birey sayısından küçük rastgele bir sayı üretilerek seçim yapılabileceği gibi, popülasyonda bulunan birey sayısının çarpmaya göre tersi alındıktan sonra bu değer her bir birey için kümülatif olarak sıralanarak, $[0; 1)$ aralığında elde edilen rastgele sayıya göre de seçilebilmektedir. Diğer seçim tekniklerinde de kullanıldığından dolayı bu seçim işlemi, popülasyonda bulunan birey sayısı 4 olan bir problem için aşağıda gösterilmektedir.

Ebeveyn Numarası	Seçilme Olasılıklarının Kümülatif Toplamı
Ebeveyn 1	0,25
Ebeveyn 2	0,50
Ebeveyn 3	0,75
Ebeveyn 4	1,00

$RS = 0{,}56 \quad \Rightarrow \quad$ Ebeveyn 2 Seçilir

Yukarıda yer alan örnekte elde edilen rastgele sayı 0,56 olduğundan ve $[0{,}50; 0{,}75)$ aralığında Ebeveyn 3 bulunduğundan dolayı, Ebeveyn 3 seçilir. Daha önce de aktarıldığı gibi her bir bireye eşit seçilme şansı verilmek üzere, probleme ve algoritmaya göre rastgele seçim tekniği farklı şekilde tasarlanabilir.

2.1.2.2. Rulet Tekerleği Seçim Tekniği

Rulet tekerleği seçim tekniğinde, popülasyonda yer alan bireylere eşit seçilme şansı verilmemektedir. İyi olan bireylerin genetik kodlarının bir sonraki nesillere aktarılma şansını arttırmak amacıyla, bireylerin seçim olasılıkları uygunluk fonksiyonu değerlerine göre düzenlenir. N adet bireyin bulunduğu bir popülasyonda yer alan *i.* bireyin uygunluk değeri u_i olmak üzere, seçim olasılığı p_i aşağıda gösterilen formül yardımıyla hesaplanır.

$$p_i = \frac{u_i}{\sum_{j=1}^{N} u_j} \tag{1}$$

Formülde öncelikle popülasyondaki tüm bireylere ait uygunluk değerleri toplanmaktadır. Her bir bireyin uygunluk değeri bu toplamın çarpmaya göre tersiyle çarpıldığı taktirde, bireyler kendi uygunluk değerleri ölçüsünde seçim şansına sahip olacaklardır. Aşağıda yer alan örnekte, popülasyonda bulunan birey sayısı 4 olan bir problem için rulet tekerleği seçim tekniğine göre olasılıklar verilmektedir.

Ebeveyn Numarası	Uygunluk Değeri	Seçilme Olasılığı	Kümülatif Olasılıklar
Ebeveyn 1	2	0,1	0,1
Ebeveyn 2	4	0,2	0,3
Ebeveyn 3	6	0,3	0,6
Ebeveyn 4	8	0,4	1,0
Toplam	20	1,0	

Burada gösterilen örnekte, rastgele sayının 0,83 gelmesi durumunda Ebeveyn 4 seçilecektir. Seçilme olasılıklarından da görülebileceği gibi, uygunluk değeri yükseldikçe ebeveynlerin seçim olasılıkları da yükselmektedir. Rulet tekerleği seçiminin kullanımıyla yöntemin yakınsama oranı arttırılır, çünkü yeni nesil, ebeveynlerden elde edilen iyi çözümlerin çoğundan etkilenecektir (Findik v.d., 2014: 120). En iyi çözümlerin seçilme olasılıklarının yüksek olması ve çoğunluğunun bir sonraki nesillere aktarılması, yöntemin optimum çözüme yaklaşma hızını da arttıracaktır.

2.1.2.3. Turnuva Seçim Tekniği

Turnuva seçim tekniği, popülasyondan seçilmiş bazı bireyler arasında turnuvalar çalıştırılmasını içerir (Mehra v.d., 2012: 168). Bireyler arasında yapılan turnuvalarda her turnuvanın kazananı, seçilmiş olan birey olacaktır. Turnuva seçim tekniğinin kullanılması sırasında önemli olan noktalardan biri de turnuva boyutunun belirlenmesidir. Eğer yüksek sayıda turnuva boyutları seçilirse, düşük uygunluk fonksiyonu değerlerine sahip bireylerin tercih edilme olasılığı düşecek, turnuvalarda eleneceklerdir. Bu durumda da çözüm, uygunluk değeri yüksek olan çözümlerden çok daha fazla etkilenecek ve çözüme çok daha hızlı gidecektir. Fakat turnuva boyutunun düşük tutulması, uygunluk değeri düşük olan çözümlere de seçilme şansı verecek ve çözüme daha yavaş ulaşılacaktır. Çözüme daha yavaş ulaşılmasına karşın,

çözüm tekniğinin global en iyi değeri bulma ihtimali, düşük uygunluk değerine sahip bireylerin de kullanılmasıyla artacak ve bu bireylerin bulunduğu bölgelerden de gidilerek daha iyi bir arama gerçekleştirebilecektir.

2.1.2.4. Diğer Seçim Teknikleri

Buraya kadar aktarılmış olan üç adet seçim tekniği, literatürde sıklıkla başvurulan seçim teknikleridir. Bunların dışında literatürde kullanılan çok çeşitli seçim teknikleri de bulunmaktadır. Aşağıda, literatürde kullanılan diğer seçim tekniklerinden bazılarının isimleri verilmektedir (Findik v.d., 2014; Mehra v.d., 2012; Slowik, 2012).

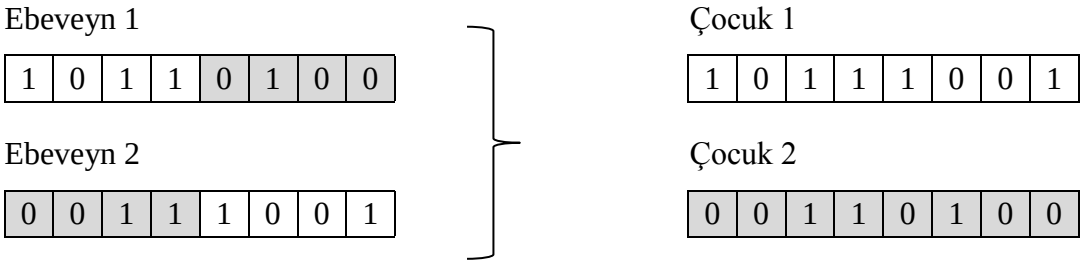
- Elitist Seçim
- Deterministik Seçim
- Kesme Seçim
- Fan Seçim
- Karışık Seçim
- Sıralamaya Dayalı Seçim

2.1.3. Çaprazlama Teknikleri

Çaprazlama, ebeveyn popülasyonundan seçilmiş olan iki bireyin genlerinin karıştırılmasıyla yeni bireyler elde etmeyi amaçlamaktadır. Ebeveyn popülasyonundan seçilen bireylerin genlerinin çaprazlanması için kullanılan üç farklı teknik bulunmaktadır; tek nokta çaprazlama, çift nokta çaprazlama ve uniform çaprazlama. Bu kısımda genetik algoritmalarda kullanılan çaprazlama çeşitleri örneklerle aktarılacaktır.

2.1.3.1. Tek Nokta Çaprazlama

Tek nokta çaprazlama tekniğinde, ebeveyn çözümlerin dizilimlerinde yer alan bir nokta seçilerek, çaprazlama işlemi bu noktadan önce ve sonra olacak şekilde gerçekleştirilir. Çaprazlamanın gerçekleştirileceği nokta, rastgele olarak seçilebileceği gibi standart olarak bir noktayla da devam edilebilmektedir. Aşağıda verilen çaprazlama örneğinde, iki farklı ebeveynden tek nokta çaprazlama tekniği kullanılarak iki farklı çocuk elde edilmektedir.

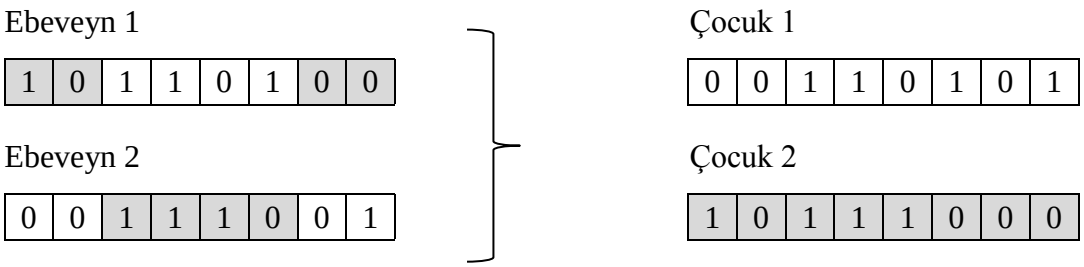


Şekil 2.4: Tek Nokta Çaprazlama Örneği.

Şekil 2.4'te gösterilen örnek çaprazlamada, Çocuk 1 için oluşturulan dizilimin ilk yarısı Ebeveyn 1'den, ikinci yarısı ise Ebeveyn 2'den alınmıştır. Çocuk 2 için oluşturulmuş olan dizilimin ise ilk yarısı Ebeveyn 2'den ikinci yarısı da Ebeveyn 1'den alınmıştır. Elde edilmiş olan çocuklardan uygunluk değeri daha iyi olan çocuk bir sonraki nesile aktarılarak çaprazlamaya diğer çocuklar için devam edilir.

2.1.3.2. Çift Nokta Çaprazlama

Çift nokta çaprazlama tekniğinde, ebeveyn çözümlerin dizilimlerinde yer alan iki nokta seçilerek, çaprazlama işlemi bu noktalar arasında gerçekleştirilir. Çaprazlamanın gerçekleştirileceği noktalar tek nokta çaprazlamada olduğu şekilde rastgele olarak seçilebileceği gibi, standart olarak iki noktayla da devam edilebilmektedir. Aşağıda verilen çaprazlama örneğinde, iki farklı ebeveynden çift nokta çaprazlama tekniği kullanılarak iki farklı çocuk elde edilmektedir.

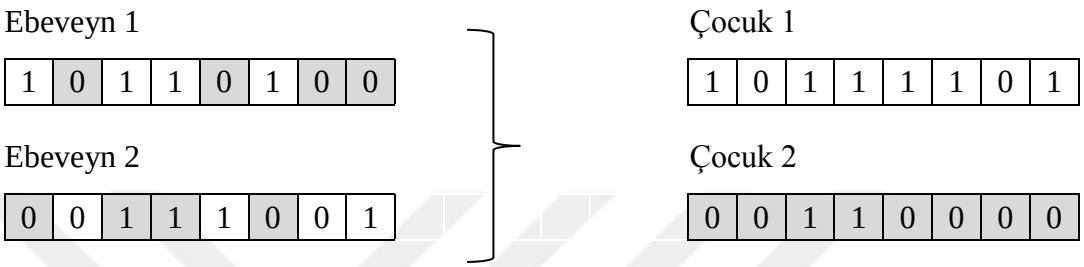


Şekil 2.5: Çift Nokta Çaprazlama Örneği.

Şekil 2.5'te gösterilen örnek çaprazlamada, Çocuk 1 için oluşturulan dizilimin ilk kısmı ve son kısmı Ebeveyn 2'den, orta bölümü ise Ebeveyn 1'den alınmıştır. Çocuk 2 için oluşturulmuş olan dizilimin ise ilk kısmı ve son kısmı Ebeveyn 1'den, orta bölümü de Ebeveyn 2'den alınmıştır. Elde edilmiş olan çocuklardan uygunluk değeri daha iyi olan çocuk, bir sonraki nesile aktarılarak çaprazlamaya diğer çocuklar için devam edilir.

2.1.3.3. Uniform Çaprazlama

Uniform çaprazlama tekniğinde, çocukların gen diziliminde yer alacak genler, ebeveynlerden her seferinde rastgele olarak seçilirler. Aşağıda verilen çaprazlama örneğinde de, iki farklı ebeveynden uniform çaprazlama tekniği kullanılarak iki farklı çocuk elde edilmektedir.



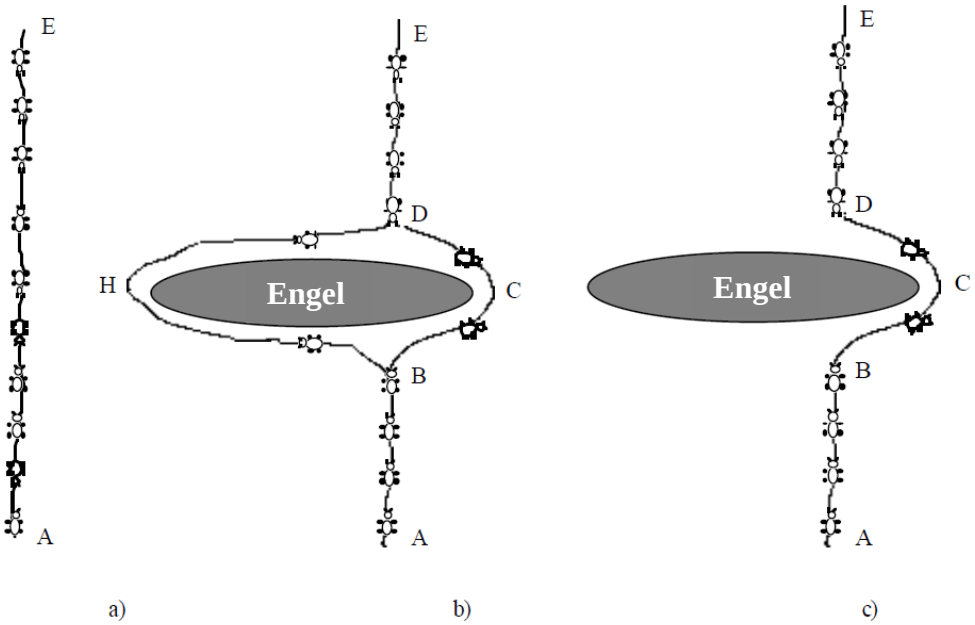
Şekil 2.6: Uniform Çaprazlama Örneği.

Şekil 2.6'da gösterilen örnek çaprazlamada, Çocuk 1 için oluşturulan dizilimde yer alan genlerin alındığı ebeveynlerin genleri beyaz ile gösterilmekte ve Çocuk 2 için oluşturulan dizilimde yer alan genlerin alındığı ebeveynlerin genleri de gri ile gösterilmektedir. Örnekte de görülebileceği gibi, çocuklara aktarılan genler ebeveynlerden karışık olarak rastgele seçilmiştir. Diğer çaprazlama tekniklerinde olduğu gibi uniform çaprazlama tekniğinde de uygunluk değeri daha iyi olan çocuk bir sonraki nesile aktarılır ve çaprazlamaya diğer çocuklar için devam edilir.

2.2. KARINCA KOLONİSİ OPTİMİZASYONU

Karınca kolonisi optimizasyonu (Ant Colony Optimization, ACO) uygulaması ilk olarak Colorni vd. tarafından 1991 (Colorni v.d., 1991) yılında tanıtılmıştır. Yöntem, karıncaların yiyecek aramaları sırasında yuvaları ile yiyecek kaynağı arasındaki en kısa yolu keşfetmeleri temeline dayanmaktadır. Bundan dolayı da yöntemin adı karınca kolonisi optimizasyonudur. Karıncalar, yuvaları ile yiyecek kaynağı arasındaki mesafede bulunan yolları görme duyuları olmadan bulabilme ve en kısa olan yola yönelme kabiliyetine sahiptirler. Karıncaların *feromon* salgısı ve bu salgının yiyecek kaynağına giden yollar üzerindeki yoğunluğuna göre, karıncalar yiyecek kaynağına gidecek oldukları yollara yönelirler.

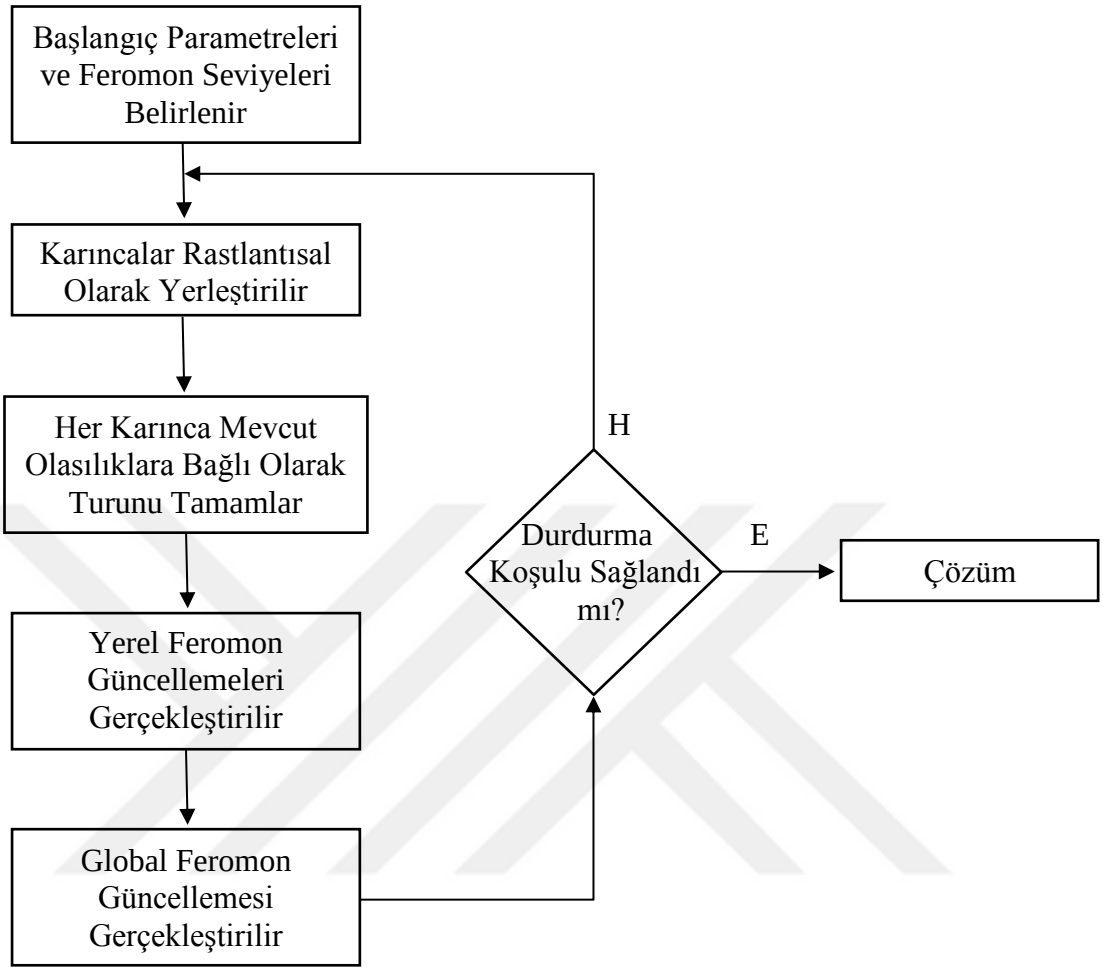
Aşağıda yer alan Şekil 2.7'de, karıncaların yiyecek kaynağına ulaşmaları sırasında takip ettikleri yolların farklı durumlara göre gösterimleri verilmektedir.



Şekil 2.7: Karıncaların Yiyeceğe Ulaşmada Takip Ettikleri Yol (Colorni v.d., 1991: 135).

Şekil 2.7'de A noktasında karıncaların yuvası, E noktasında da yiyecek kaynağı bulunsun. Şekilde bulunan ilk gösterimde, karıncalar yuva ile yiyecek arasındaki en kısa olan yolu düz bir şekilde kat etmektedirler. Daha sonra karıncaların devam ettikleri bu yol üzerine bir engel koyulduğu zaman, engel etrafında hiç feromon salgısı olmadığından dolayı, karıncalar yollarını rasgele olarak seçeceklerdir. Koyulmuş olan engel etrafından geçen karıncaların her seferinde bırakmış oldukları feromon salgılarının yoğunluklarına göre, ilerleyen zamanlarda gelen karıncalar seçimlerinde feromon salgısının yoğun olduğu bölümü seçeceklerdir. Yani feromon salgısı yoğun olan yolun karıncalar tarafından seçilme şansı artacaktır. Yeterli zaman geçtikten ve denge durumuna gelindikten sonra tüm karıncalar en kısa olan yoldan devam edeceklerdir. Uzun olan yolda bulunan feromon salgısı ise zamanla tercih edilmedikçe azalacağından dolayı, seçim olasılığı da azalacak ve karıncalar uzun olan bu yolu tercih etmeyi bırakacaklardır.

Karınca kolonisi optimizasyonu Colorni v.d. tarafından ilk tanıtıldığı zamandan bu yana farklı şekillerde geliştirilmiş ve diğer yöntemlerle de entegre olarak kullanılmıştır. Aşağıda yer alan Şekil 2.8'de, karınca kolonisi optimizasyonunun genel çözüm yapısı gösterilmektedir.



Şekil 2.8: Karınca Kolonisi Optimizasyonu Çözüm Yapısı.

2.2.1. Karınca Kolonisi Optimizasyonunda Kullanılan Yapılar ve Tanımlar

Karınca kolonisi optimizasyonu, karıncaların yiyecek aramalarını baz alan bir yöntem olduğundan dolayı, genetik algoritmalara benzer şekilde bu yöntemde kullanılan terimlerde ve yapılarda da karıncalar için olan isimlerle aynı şekilde adlandırmalar yapılmıştır. Bu kısımda da, karınca kolonisi optimizasyonunun daha iyi anlaşılabilmesi için karınca kolonisi optimizasyonunda kullanılan terimler tanımlamalarıyla birlikte incelenecektir.

Karıncalar: Karınca kolonisi optimizasyonunda kullanılan karıncalar, gerçek hayattaki karıncaların bilgisayar ortamında oluşturulmuş birer uyarlamasıdır. Bilgisayar ortamındaki bu karıncalar oluşturulurken, gerçek hayattaki karıncalardan esinlenilmiş ve daha önce geçilmiş olan şehirleri hafızada tutma gibi problem

çeşidine göre değişen özellikler de eklenebilmektedir. Oluşturulmuş olan bu yapay karıncalar da gidecekleri yolları gerçek karıncalar gibi feromon salgısına bağlı olarak tercih edecektir. Karınca kolonisi optimizasyonu için gerçek karınca davranışlarından transfer edilen üç fikir aşağıdaki gibidir (Tuker v.d., 2013: 91):

- Yüksek feromona sahip olan yol tercih edilir,
- Daha kısa yollarda bulunan feromon miktarının artış oranı yüksek olur,
- Karıncalar arasında feromon izine bağlı iletişim olur.

Feromon: Karıncaların yiyecek kaynağına ulaşmaları sırasında geçtikleri yol üzerinde bıraktıkları sıvıdır. Bu sıvı, karıncaların geçtikleri yol üzerinde zamanla buharlaşıp azalarak yok olur. Bu durumdan yola çıkılarak da karınca kolonisi optimizasyonunda problemin tipine göre feromon dağıtımı ceza puanıyla ters orantılı olarak alternatifler için hesaplanmaktadır. Feromon, engelin etrafında daha kısa olan yolda daha hızlı birikir (Parpinelli v.d., 2002: 322). Tercih edilmeyen alternatiflerde bulunan feromon miktarları da zamanla azalarak yok olur. Böylece nihai durumda, karıncalar optimum güzergah üzerinden yiyeceğe ulaşmış olurlar. Karınca kolonisi optimizasyonunun en kısa yol problemlerine uyarlanmasında da feromon bu şekilde kullanılır. Yapılan çözüm sırasında, her iterasyonda feromon değerleri güncellenirler.

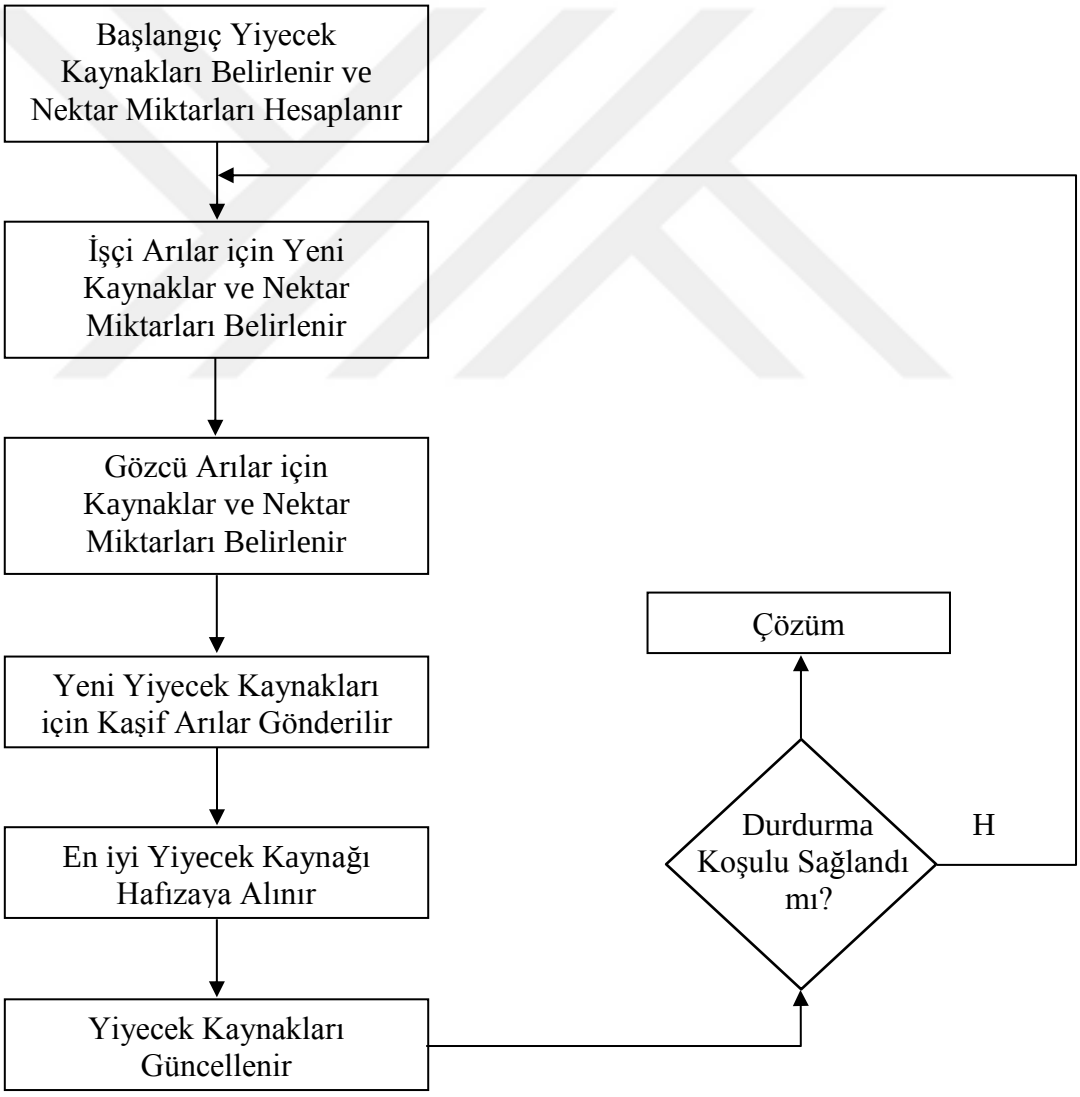
Yerel Feromon Güncellemesi: Karıncaların turlarını tamamlamaları sırasında, geçtikleri yollara ait feromon miktarlarının her karınca için arttırılmasıdır. Örneğin; A noktasında bulunan bir karıncanın etrafında gidebileceği B, C ve D noktaları ve bu noktalara ulaşımını sağlayabileceği birer yol olsun. Karıncanın rastgele olarak seçtiği noktanın C noktası olduğunu varsayarsak; karınca A–C yolundan geçişini tamamladıktan sonra bu yoldaki feromon miktarının arttırılmasına, yerel feromon güncellemesi adı verilir.

Global Feromon Güncellemesi: Problemin çözümü için kullanılan tüm karıncaların turlarını tamamlamalarından sonra, en kısa yolu kullanmış olan karıncanın kullandığı güzergah ayrıca değerlendirilir. Bir iterasyonda en kısa yolu kullanan bu karıncanın geçtiği güzergahta bulunan yollara ait feromon miktarlarının ayrıca arttırılmasına da global feromon güncellemesi adı verilir.

2.3. YAPAY ARI KOLONİSİ

Yapay arı kolonisi yöntemi, ilk olarak 2005 yılında Derviş Karaboğa (Karaboğa, 2005) tarafından tanıtılmıştır. Yöntem, arıların besinlerini aramaları sırasında gerçekleştirdikleri davranışlarından yola çıkılarak geliştirilmiştir ve bu yönüyle dikkate alındığında; sürü zekası temeline dayanan bir sezgisel yöntemdir.

Gerçek hayattaki arı kolonilerinden esinlenilerek geliştirilmiş olan bu yöntemde, üç çeşit arı bulunmaktadır; işçi arı, gözcü arı, kaşif arı. Bir yapay arı kolonisi algoritmasının genel akış şeması aşağıda yer alan şekilde gösterilmektedir.



Şekil 2.9: Yapay Arı Kolonisi Çözüm Yapısı.

Yapay arı kolonisi algoritmasında öncelikle başlangıç yiyecek kaynakları belirlenir. Daha sonra bu yiyecek kaynaklarına ait nektar miktarları hesaplanır. Bu aşamadan sonra işçi arılar için yiyecek kaynağı pozisyonları belirlenerek işçi arılar gönderilirler. İşçi arılar, görevlendirilmiş olduğu belirli bir besin kaynağıyla ilişkili olan arılardır (Karaboğa v.d., 2014: 25). İşçi arıların görevlendirildiği kaynakların etrafında belirli mesafede bulunan kaynakların araştırılması ve yeni yiyecek kaynaklarının tespit edilmesi için gözcü arılar gönderilirler. Gözcü arılar daha yüksek değere sahip yiyecek kaynağı bulmuş ise işçi arı olarak değiştirilirler, aksi durumda gözcü arıların sayaç değeri bir arttırılır. Çözümde yer alan işçi arı sayısı ile gözcü arı sayısının birbirine eşit tutulması gerekmektedir. Arıların birbirleriyle olan bilgi alışverişi, ortak bilginin oluşmasında en önemli durumdur (Karaboğa ve Basturk, 2008: 688). Arılar, bu bilgi alışverişini kovan çevresinde dans ederek gerçekleştirirler. Görevli oldukları yiyecek kaynağının uzaklığı, yeri, miktarı, kalitesi gibi bilgileri farklı danslar ile kovanda bulunan diğer arılarla paylaşırlar. Kaşif arılar da rastgele bir şekilde gönderilerek, yeni yiyecek kaynaklarının konumları tespit edilir. Durdurma kriteri ya da kriterleri sağlanana kadar bu şekilde aşamalar tekrar edilir. En iyi yiyecek kaynağı her seferinde kaydedilir.

2.3.1. Yapay Arı Kolonisinde Kullanılan Yapılar ve Tanımlar

Yapay arı kolonisi de gerçek arı kolonilerinden esinlenilerek geliştirildiğinden dolayı, buraya kadar aktarılmış olan yöntemlerde olduğu gibi yapay arı kolonisinde kullanılan yapılar ve tanımlar da gerçek hayattakine benzerlikler göstermektedir. Yapay arı kolonisinin daha iyi anlaşılabilmesi için bu kısımda, yöntemde kullanılan yapılar ve tanımlar aktarılmaya çalışılacaktır.

İlk olarak yapay arı kolonisinde kullanılan arı çeşitleri incelenecektir.

İşçi Arılar: İşçi arılar, belirlenmiş ve görevlendirilmiş oldukları yiyecek kaynağında bulunan yiyeceğin toplanarak kovana getirilmesinden sorumludurlar. İşçi arılar kovana her gelişlerinde görevli oldukları yiyecek kaynağının konum ve miktar bilgilerini kovanda bulunan diğer arılara iletmekle de sorumludurlar. Gözcü arılar da bu gelen bilgilere göre daha iyi konumlandırılırlar. Görevli olduğu yiyecek kaynağındaki yiyeceği biten arılar, kaşif arı olarak görevlendirilirler.

Gözcü Arılar: Gözcü arılar, işçi arılar etrafında belirli mesafede konumlandırılarak ve kaynaklarına ait bilgilere dayanarak yiyecek kaynakları arar ve yeni kaynaklara yönelirler. Gözcü arılar, işçi arılardan daha yüksek bir değere sahip yiyecek kaynağı bulması durumunda, işçi arı olarak değiştirilirler. Aksi taktirde gözcü arılara ait sayaç değeri bir arttırılır. Gözcü arılara ait sayaç değeri belirli bir limite ulaştığı taktirde, daha iyi bir besin kaynağı bulunmadığından dolayı da yiyecek kaynağı terk edilerek gözcü arılar, kaşif arı olarak görevlendirilirler.

Kaşif Arılar: Kaşif arılar, gözcü arılarla benzer görevlere sahiptirler. Fakat, kaşif arılar, kovanda bekleyerek ve kovana gelen arıların danslarından edindiği bilgilere dayanarak yeni kaynaklara yönelirler.

Yapay arı kolonisinde görevli arıların, yiyecek kaynağının konumu ve miktarı ile alakalı bilgileri diğer arılara aktarması arıların dans etmesi ile gerçekleştirilir. Gerçek hayattaki arılarda çok çeşitli danslar olmasına karşın yapay arı kolonisinde kullanılan üç temel dans çeşidi bulunmaktadır; dairesel dans, kuyruk dansı, titreme dansı.

Dairesel Dans: Yapay arı kolonisinde kullanılan dairesel dans, gerçek arı kolonilerindeki dairesel dans ile benzerdir. Gerçek arı kolonilerinde, dairesel dansta besin kaynağının yönü hakkında bilgi ya hiç yoktur ya da çok azdır ve dansı izleyen arılar, kovanın 100 m uzağında herhangi bir yöne uçuşa geçebilirler (Korkmaz, 2013: 68). Yapay arı kolonisinde de, dairesel dansı yapan işçi arı yiyecek kaynağının kovana çok yakın olduğu bilgisini vermektedir. Bu dans ile gerçek hayatta olduğu gibi, kovanda bulunan arılara tam bilgi verilmemekte ve hatta çok az bilgi verilmektedir.

Kuyruk Dansı: Gerçek arı kolonilerinde, bal arıları kovana 100 m'den daha fazla uzakta olan yiyecek kaynaklarını kolonin diğer bireylerine bildirmek için petek üzerinde kuyruk sallama dansı yaparlar (Korkmaz, 2013: 68). Kuyruk sallama dansı, kovanın konumu ve güneşin konumu kullanılarak, yiyecek kaynağının konumu ile ilgili kovanda bulunan arılara detaylı bilgi vermek için kullanılmaktadır. Yapay arı kolonisinde de bu şekilde, kuyruk dansı ile yiyecek kaynağının uzaklığı, konumu, miktarı ve kalitesi gibi bilgiler kovandaki arılara iletilir. Tanımlamalardan da

anlaşılacağı gibi kuyruk dansı, dairesel dansa göre daha geniş bir alanı kapsamakta ve yiyecek kaynağıyla alakalı daha fazla bilgiyi kovanda bulunan arılara aktarmaktadır.

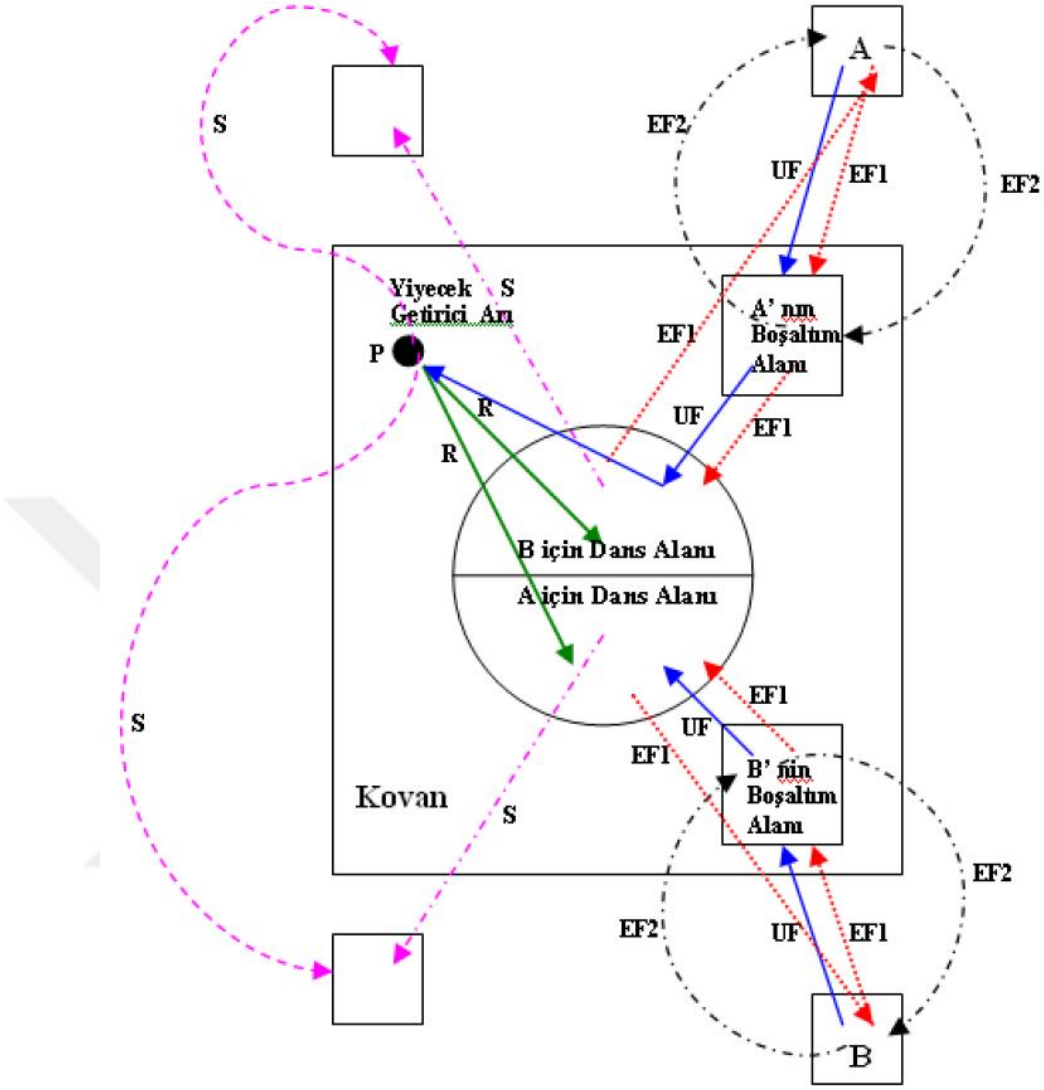
Titreme Dansı: Titreme dansı, arı kolonisinin dinamik bir sistem olduğunun en güzel örneklerinden birisidir (Özdemir, 2013: 66). Arılar, kovandaki yiyecek miktarını belirli bir seviyede tutma eğilimindedirler. Bu gibi durumlarda arılar, kovana işlenebileceğinden çok daha fazla yiyecek geldiğinde, titreme dansı ile bu yiyeceği işlemek üzere görev alacaklarını belirtirler.

Buraya kadar yapılmış olan tanımlamalar ve yapılar göz önünde bulundurulduğunda, optimum çözüme ulaşılması sırasında yapay arı kolonisinde, arılar arasındaki bilgi alış verişinin çok büyük öneme sahip olduğu anlaşılmaktadır.

2.3.2. Yapay Arı Kolonisinde Arıların Davranışları

Şekil 2.10'da, A ve B noktalarında bulunan iki yiyecek kaynağı için arıların davranışları verilmektedir. Şekilde, görevli bir arının gittiği kaynağı bırakarak gözcü arı olması durumu, UF ile gösterilmektedir. Boşaltımı yaptıktan sonra dans ederek tekrar yiyecek kaynağına yönelmesi, EF1 ile gösterilmektedir. Boşaltımı yaptıktan sonra dans etmeden tekrar yiyecek kaynağına yönelmesi durumu da, EF2 ile gösterilmektedir. s ile gözcü yada kaşif arılar tasvir edilmektedir. Bu arılar, kovanda bulunan dans alanını izledikten sonra görevli arı konumuna geçerek yiyecek kaynağına yönelebilirler.

Görevli arılar boşaltım alanından sonra dans etmeden tekrar yiyecek kaynağına yönelebildiklerinden dolayı, kovanda tam bir iletişim olduğundan söz edilemez. Fakat kovanda bulunan arılar yoğun olarak birbirleri ile iletişim içerisindedirler ve bu iletişimleri için dans alanını kullanmaktadırlar. Arıların dans alanı sayesinde birbirleri ile iletişimi, pozisyonlarının değişmesine de olanak sağlamaktadır. Arıların pozisyonlarının değişmesindeki en önemli etkenlerden biri olan dans alanının etkin kullanılmaması, kolektifliği de azaltacağından dolayı süre de uzayacaktır. O halde, yapay arı kolonisindeki en önemli noktalardan biri, arılar arasındaki bu iletişim yapısının etkin bir şekilde kurulmasıdır.



Şekil 2.10: Arıların Yiyecek Arama Çevrimi (Akay, 2009: 57).

2.4. TABU ARAMA ALGORİTMASI

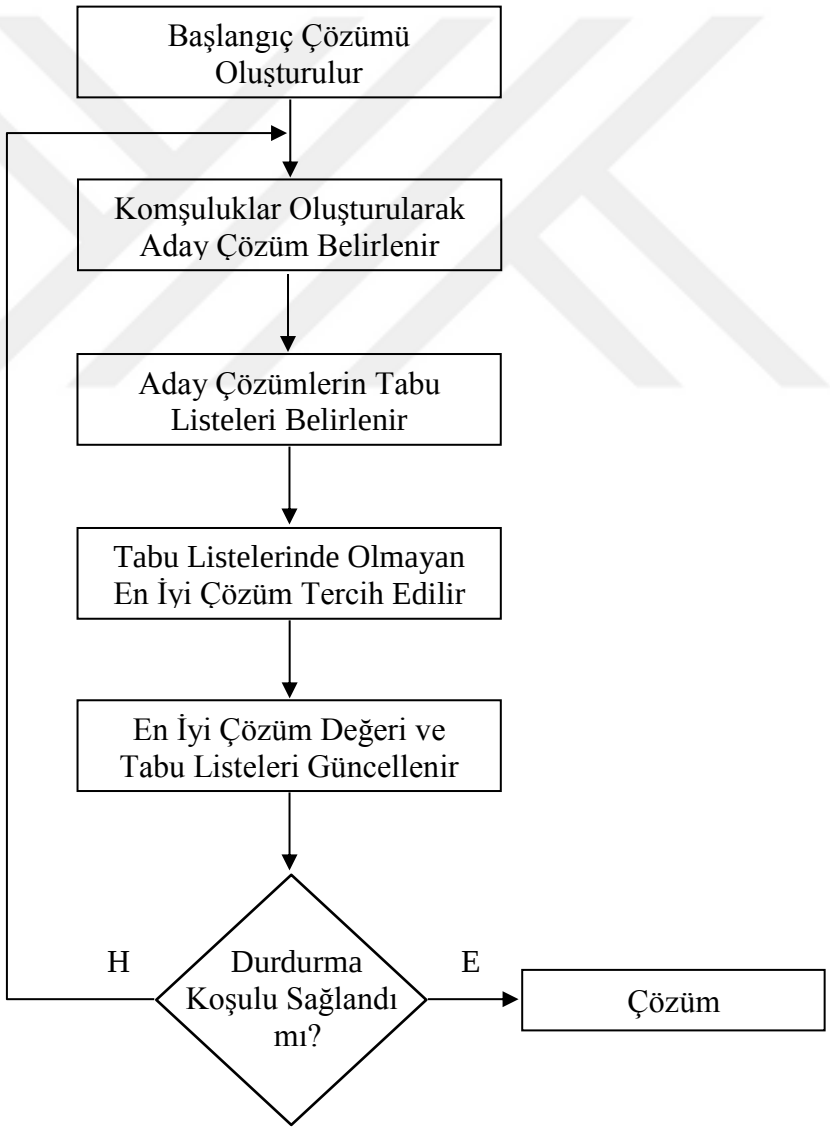
Tabu Arama yöntemi, ilk olarak 1989 yılında Fred Glover (Glover, 1989) tarafından tanıtılmıştır. Yöntem, NP sınıfı problemlerin çözümü için geliştirilmiş; bir takım yasaklar listesi oluşturma ve aramalar sırasında komşuluk hareketlerini yönlendirme temeline dayanan bir sezgisel yöntemdir. Kavramsal açıdan; tabu aramanın en basit hali olarak; tüm bu işlemlerin hedefi, hangi hareketlerin kabul edilebilir olduğunu belirlemek için bir yol olarak yerel optimumdan kaçış mesafesinin yararlı bir tahminini elde etmektir (Glover, 1989: 200).

Algoritma, yerel aramalarda dairesel hareketleri ve tekrarları engellemek için yasaklar listesi oluşturur ve çözümün her adımında komşuluklardan gidilerek bu yasaklar listesi güncellenir. Bu sayede yöntem, daha önceden incelenmiş olan çözümlerin tekrar tekrar ele alınmasını ve yeniden değerlendirilmesini engeller. Yöntemin önemli noktalarından biri de, komşulukların karşılıklı olarak değerlendirilmesidir. Yani komşuluklar tek yönlü olarak değerlendirilmez ve eğer iki çözüm için bir komşuluk ilişkisi varsa, bu ilişki her iki çözüm için de karşılıklı olarak kullanılmaktadır. Bundan dolayı da oluşturulmuş olan bu tabu listesi ya da listeleri sayesinde hareket tekrarları önlenebildiği gibi, aynı zamanda daha önceden yapılmış olan bir hareketin tersine dönmesi de engellenmektedir. Yani; bir çözüm algoritmasında belirli bir noktadan başlayarak, o noktanın komşuluklarından hareketle ulaştığımız bir çözümden, bir süre sonra geriye dönerek aynı hareketlerin tekrarlanması ihtimali söz konusu olabilmektedir. Tabu Arama Algoritması yöntemi ile, bu geri hareket tekrarları ve aynı çözüm bölgesinde dolaşma durumu engellemektedir. Yöntemde kullanılan Tabu Listesi, güncellenebilen bir listedir. Çözüm bölgesi değiştikçe tabu listesi de değişecektir. Tabu Arama Algoritması içerisinde kullanılan teknikler sayesinde tekrarlar önlendiğinden dolayı çözüme gitmek de hızlanacaktır. Tabu Arama Algoritmasında oluşturulmuş olan tabu listeleri sayesinde, çözümün yönelim alanı kısıtlanmakta ve çözüme gidilirken yapılan hareketler yönlendirilmektedir.

Tabu Arama Algoritması yönteminde, çözümde yer alan her değişkenin komşu çözümleri taranır ve bu komşu çözümlerden en iyi olan, eğer tabu listesinde de yer almıyorsa, çözüm listesine dahil edilir. Elde edilen yeni çözüm, mevcut çözüm listesine dahil edilirken, arama sırasında yola çıkılan çözümün yerini alır. Aynı işlemler tüm çözüm değişkenlerine de uygulandıktan sonra iterasyonun bu aşaması tamamlanır. Sonraki iterasyonda, yenilenmiş olan bu çözüm listesi ile aramalara devam edilir. Problemin çözümüne başlarken tespit edilmiş olan durdurma kriteri ya da kriterleri sağlanana kadar, bu şekilde tekrar edilen iterasyonlarla en iyi çözüme ulaşmaya çalışılır. Tabu Arama Algoritmasının bu noktadaki önemi, kısa ve uzun süreli hafızaya sahip tabu listelerinin belirlenen sürelerde hafızada tutulması sayesinde, uzun süre yerelde kalan ve tabu hareketleri bulunan bir çözüm, bir süre

sonra yerelde bulunan bu noktadan uzaklaşır ve farklı bir çözüm arayışına girer. Yöntemin bu özelliği sayesinde yerelde takılmış olan bir çözüm, neredeyse tüm hareketleri kısıtlandığından dolayı gidecek yeri kalmayarak bu yerel çözümden uzaklaşır. Bu arama yöntemi sayesinde yöntem, en nihayetinde global çözüme ulaşmayı hedefler.

Literatürde şu ana kadar geliştirilmiş olan farklı Tabu Arama Algoritması çözüm yaklaşımları bulunmaktadır. Aşağıda yer alan Şekil ‘de Tabu Arama Algoritmasının genel akış şeması gösterilmektedir.



Şekil 2.11: Tabu Arama Algoritması Çözüm Yapısı.

2.4.1. Tabu Arama Algoritmasında Kullanılan Yapılar ve Tanımlar

Tabu Arama Algoritmasında da diğer yöntemlerde olduğu gibi kendine has tanımlar ve yapılar bulunmaktadır. Bu kısımda, Tabu Arama Algoritmasının daha iyi bir şekilde anlaşılabilmesi için, yöntemde kullanılan yapılar ve tanımlar ayrıntılı bir şekilde aktarılmaya çalışılacaktır.

Hareket: Tabu Arama algoritmasının komşuluk hareketlerini esas alan bir yöntem olduğuna daha önce değinilmişti. Bu komşuluk hareketleri sırasında, komşuluk aramaları gerçekleştirilirken belirli bir kural takip edilir. Örneğin, problemin bir çözümü A noktası olsun. Bu A noktasından komşuluk arama kuralına göre B noktasına gidilir ve bu işleme hareket adı verilir.

Hareket Değeri: Çözüm sırasında gerçekleştirilen hareketlerin tamamı, çözüm üzerinde aynı değere ve etkiye sahip değildir. Bu etkilerin karşılaştırılması sırasında bir değer kullanılması gerekmektedir. Bunun için de hareket değeri olarak adlandırılan ve hareket edilen noktaların amaç fonksiyonu değerlerinin farkını ifade eden hesaplama sonucu kullanılır. Hareket ile birlikte amaç fonksiyonu değeri değişecektir ve hareketten önce ve sonra elde edilen bu amaç fonksiyonu değeri farkı hareket değeri olarak adlandırılır (Zeng ve Gao, 2015: 256).

Komşuluk Çözümü: Hareket yapısı kullanılarak mevcut bir çözümden oluşturulan yeni çözüme, komşuluk çözümü adı verilir. Yöntemde, komşuluk ilişkileri karşılıklı olarak değerlendirildiğinden dolayı, A noktasından hareketle elde edilmiş olan B noktası, A için bir komşuluk çözümken, A noktası da B noktası için bir komşuluk çözüm olacaktır.

Tabu Listesi: Tekrarların önlenmesi ve yerel optimumdan kaçmak için ziyaret edilen çözümlerin kısa süreli hafızası tutulur ve bu tabu listesi olarak adlandırılır (Ramezanian v.d., 2012: 1261). Tabu listelerinin oluşturulma amacı her ne kadar hareket tekrarlarını engellemek olarak tasarlansa da, komşuluk ilişkilerinin karşılıklı olmasından da dolayı; daha önceden gerçekleştirilmiş olan bir hareketin tersine dönmesi de tabu listeleri ile engellenmektedir. Bu yapısı ve işlevi dolayısıyla, tabu listelerinin boyutunun doğru bir şekilde ayarlanması büyük önem teşkil etmektedir.

Tabu Kısıtlaması: Gerçekleştirilecek olan hareketlerin tekrarlarının önlenmesi amacıyla tabu listesinde oluşturulmuş olan kısıtlamalardır. Tabu kısıtlamalarına göre hareketler gerçekleştirilmesi, yerel optimum bölgelerinden kurtulmayı sağlayacaktır.

Tabu Süresi: Tabu olan hareketin ya da hareketlerin hafızada tutularak yasaklı kaldığı ve tekrarlanmasının yasak olduğu süredir. Burada geçen süre ifadesi algoritmalarda gerçek anlamıyla değil, hareket adedi olarak kullanılmaktadır. Örneğin, *n* boyutlu bir tabu listesi olduğu düşünülürse; tabu listesine alınan bir hareketin listeye alınmasından sonra *n* adet hareket daha gerçekleştirildiğinde, tabu listesine alınan bu hareket listeden çıkarılacaktır. Yani diğer bir ifadeyle; tabu listesine alınan bir hareketin tekrar edilmesine, tabu listesinin boyutu kadar hareket gerçekleştirilene kadar izin verilmez.

Seçim Stratejisi: Bir çözümün etrafında çok fazla komşuluk bulunacaktır ve bu komşulukların arasından hangisinin seçileceğinin tespit edilmesi önemli bir durum haline gelmektedir. Seçim stratejisi, bir sonraki iterasyonda başlangıç çözümü olarak komşuluklardan, daha iyi bir çözümün seçilmesine dayanan yöntemdir (Zeng ve Gao, 2015: 256). Literatürde yer alan çeşitli seçim stratejileri bulunmaktadır. Bu farklı seçim stratejilerine bir sonraki kısımda ayrıntılı olarak değinilecektir.

Kalite: Literatürde, çözümler arasındaki hareketlerin değerlendirilmesi için kalite sıklıkla kullanılan bir kavramdır. Bu tanımı itibariyle kalite kavramından yalnızca Tabu Arama Algoritmasında değil, diğer yöntemlerde de faydalanılmaktadır. Bir noktanın kalitesini, baz alınan noktaya göre fonksiyon üzerindeki değişimin yüzdesel ifadesi olarak tanımlanabilir.

Elit Çözüm: Geçerli bağımsız süreç içinde en iyi sonucu temsil eden çözüme elit çözüm adı verilir (Peng v.d., 2014: 118).

Hafıza: Hafıza bileşeni, Tabu Arama Algoritmasının en önemli unsurlarından biridir. Tabu Aramanın hafıza yapıları yakın geçmiş (recency), frekans (frequency), kalite (quality) ve etkiden (influence) oluşan dört temel boyuta referans alınarak çalıştırılır (Lalla–Ruiz v.d., 2012: 1134). Bu farklı hafıza yapılarına bir sonraki kısımda ayrıntılı olarak değinilecektir.

2.4.2. Tabu Arama Algoritmasında Kullanılan Stratejiler

Tabu Arama algoritmasında, gerek algoritma geneli gerekse de farklı hafıza yapıları için geliştirilmiş çeşitli stratejiler uygulanmaktadır. Geliştirilmiş olan bu stratejiler, Glover ve Laguna tarafından 1998 yılında yayınlanmış olan "Tabu Search" kitabında (Glover ve Laguna, 1998: 61–67, 98–102) detaylı olarak anlatılmaktadır. Yöntemin tanıtıldığı günden bu yana geliştirilmiş olan çok çeşitli stratejiler bulunmaktadır, fakat bu kısımda, geliştirilmiş olan bu stratejilerden yöntemin geliştiricileri tarafından kitapta da verilmekte olan stratejiler kısaca aktarılmaya çalışılacaktır.

2.4.2.1. Aday Liste Stratejileri

Aday liste stratejileri, aday listelerinin oluşturulması için geliştirilmiş olan kuralları ifade etmektedir. Bir aday liste stratejisi, komşulukların değerlendirilmesi ve bu komşuluklardan hangisinin aday çözüm olarak tercih edileceği ya da aday çözümler içerisinde gerçekleştirilecek işlemler için geliştirilmiş stratejidir.

Aspiration Plus Strategy: Bu strateji ile çözüm işlemleri sırasında elde edilmiş olan kaliteli çözümlerin göz ardı edilmesinin önlenmesi sağlanmaya çalışılmaktadır. Aspiration Plus Strategy, arama yapısı geçmişine dayalı bir hareketin kalitesi için bir eşik değeri belirler (Glover ve Laguna, 1998: 61). Belirlenmiş olan bu eşik değerine ulaşılmasından sonra ek hareketler gerçekleştirilir ve yapılan tüm harekerlerin içerisinden en iyi olanı tercih edilir.

Elit Aday Listesi Yaklaşımı: Bu yaklaşımda, belirli sayıda bir süreç boyunca elde edilen en iyi çözümlerin bir ana listesi oluşturulur. Oluşturulmuş olan bu ana liste içerisindeki en iyi çözüm, belirlenmiş olan bir kalite eşik değerinin altına düşüldüğünde ya da belirli sayıda süreç tamamlandığında seçilerek devam edilir. Bu tanımlamadan da anlaşılacağı gibi yaklaşım, süreçlerin başlangıcında ya da devam eden süreçlerin şekillendirilmesinde kullanılır.

Ardışık Filtre Stratejisi: Ardışık filtre stratejisi başlangıçta, uygulamada tipik bir çekirdekle karşılaşacak işlemlerin ortalama sayısı için bir eşik değeri hesaplar ve sonra da diğer yoğun kullanımı olamayan çekirdekler ile düşük işlem oranları gösteren çekirdekleri en başta filtrelemeyi hedefler (Tino ve Khan, 2013: 1355).

Burada düşük işlem oranı gösteren çekirdekler ile aktarılmak istenen, ortalama eşik değerinden daha düşük değere sahip çekirdeklerdir.

Ardışık Yelpaze Aday Listesi: Ardışık yelpaze aday listesi, ağaç arama algoritmalarına benzer şekilde çalışmaktadır. Yöntemin temel mantığı, bir adımdaki hareketlerden belirli sayıda en iyi alternatifin üretilmesini ve sonrasında da oluşturulmuş olan bu ağaç yapısından her bir alternatif için bir tanesinin tercih edilmesini içerir. Her bir iterasyonda bu şekilde işlemlere devam edilerek süreç tamamlanır.

Sınırlandırılmış Değişim Aday Listesi: Bu tekniğin uygulanmasında gerçekleştirilen tüm adımlarda, iyileştirilmiş olan hiçbir çözüm değişkeni, sınırlandırılmış bir miktardan fazla değişemez. Bu şekilde değişikliklerin sınırlandırılması ile birlikte değerlendirilecek olan alternatiflerde sınırlandırılmış olacaktır. Bu durum da işlem sürelerini kısaltacak ve işlem sayılarını arttıracaktır.

2.4.2.2. Yoğunlaştırma Stratejileri

Bu stratejiler, geçmişte bulunan ve kaydedilmiş olan iyi çözümlerin hareketlerini ya da hareket kombinasyonlarını belirli kurallar altında düzenlemeye dayanır. Bu düzenlemeler sırasında iyi çözümlere ait hareketler desteklenerek en iyi çözüme ulaşma hedeflenir. Yoğunlaştırma Stratejilerinde, elit çözümler, benzer çözüm özelliklerini paylaşan ya da ortak bir bölgede bulunan hareketlerin yeniden bağlanarak birleştirilmesi üzerinedir (Ribeiro ve Resende, 2012: 205). Bu stratejilerde kullanılan komşuluk ifadesi, hareket kombinasyonlarını da göz önünde bulundurulduğundan dolayı, biraz daha geniş bir alandaki komşuluğu ifade etmektedir.

2.4.2.3. Çeşitlendirme Stratejileri

Çeşitlendirme stratejilerinde, daha önceden değerlendirilmiş olan çözümlerin dışında yer alan, yani değerlendirilmemiş olan çözümlerin değerlendirilmesini hedeflemektedir. Farklı çeşitlendirme stratejileri aşağıda aktarılmaktadır.

Değiştirici Seçim Kuralı: Bu çeşitlendirme stratejisinde, belirli bir ceza fonksiyonu ile elde edilen bir ceza puanı ve ayarlanabilir çeşitlendirme parametresi kullanılarak

hareket değeri hesaplanır. Yöntemde kullanılan hareket değeri, ceza puanının çeşitlendirme parametresi ile çarpılması sonucunda elde edilir.

Yeniden Başlama: Çözüm sırasında algoritma belirli bir frekansta durdurulur ve yeni bir başlangıç noktası oluşturularak süreç yeniden başlatılır. Bu yöntemle birlikte çözüm, daha önceden değerlendirilmemiş yeni bir noktadan başlanarak yeniden değerlendirilir. Böylece, her seferinde yeni çözüm hareket dizileri geliştirilmiş olur. Yeniden başlatma işleminde, başlangıç noktaları diğer sezgisel yöntemlerde olduğu gibi rastgele olarak seçilir, fakat burada da tabu aramada olduğu gibi başlangıç noktaları belirli kurallarla sınırlandırılarak rastgele seçilirler. Yani, çözüm uzayında bulunan her noktanın seçim şansı eşit olmayacaktır.

2.4.3. Tabu Arama Algoritmasında Kullanılan Hafıza Yapıları

Tabu Arama Algoritmasının, hafızaya dayalı ve bu hafıza ile birlikte hareketlerin tabu olarak kaydedilmesi sayesinde en iyi çözüme ulaşmayı hedefleyen bir yöntem olduğundan daha önce bahsedilmişti. Bundan dolayı hafıza yapılarının oluşturulması, tanımlanması ve kullanılması doğru bir şekilde gerçekleştirilmelidir. Kısa süreli hafıza ve uzun süreli hafıza olmak üzere, Tabu Arama Algoritmasında kullanılan en temel iki farklı hafıza yapısı bulunmaktadır.

Kısa Süreli Hafıza: Kısa süreli hafıza yapısında, yalnızca yakın geçmişte gerçekleştirilen hareketler tutulur. Gerçekleştirilmiş olan hareketlerin hem kendisi hem de tersi kaydedilerek yapılmış olan bir hareketin tersine dönmesi de bu hafıza sayesinde engellenmiş olacaktır.

Uzun Süreli Hafıza: Uzun süreli hafıza yapısında ise gerçekleştirilen ilgili tüm hareketler tutulur. Gerçekleştirilen tüm iyi çözüm değerlerinin hafızada tutulması sayesinde ise yerel optimumda takılmaların engellenmesine olanak sağlanmaktadır.

Kısa süreli hafıza, yerel optimuma yakın olmak için keşfe olanak sağlar ve tabu listesi tarafından uygulanan uzun süreli hafıza da, yerel optimum civarından kaçmaya yardımcı olur (Xie v.d., 2015: 2806). Burada aktarılan iki temel hafıza yapısı dışında literatürde yer alan çeşitli hafıza yapıları da mevcuttur. Problem çeşidine ya da önerilen çözüm algoritmasına göre farklı hafıza yapıları da geliştirilebilmektedir.

2.5. TAVLAMA BENZETİMİ

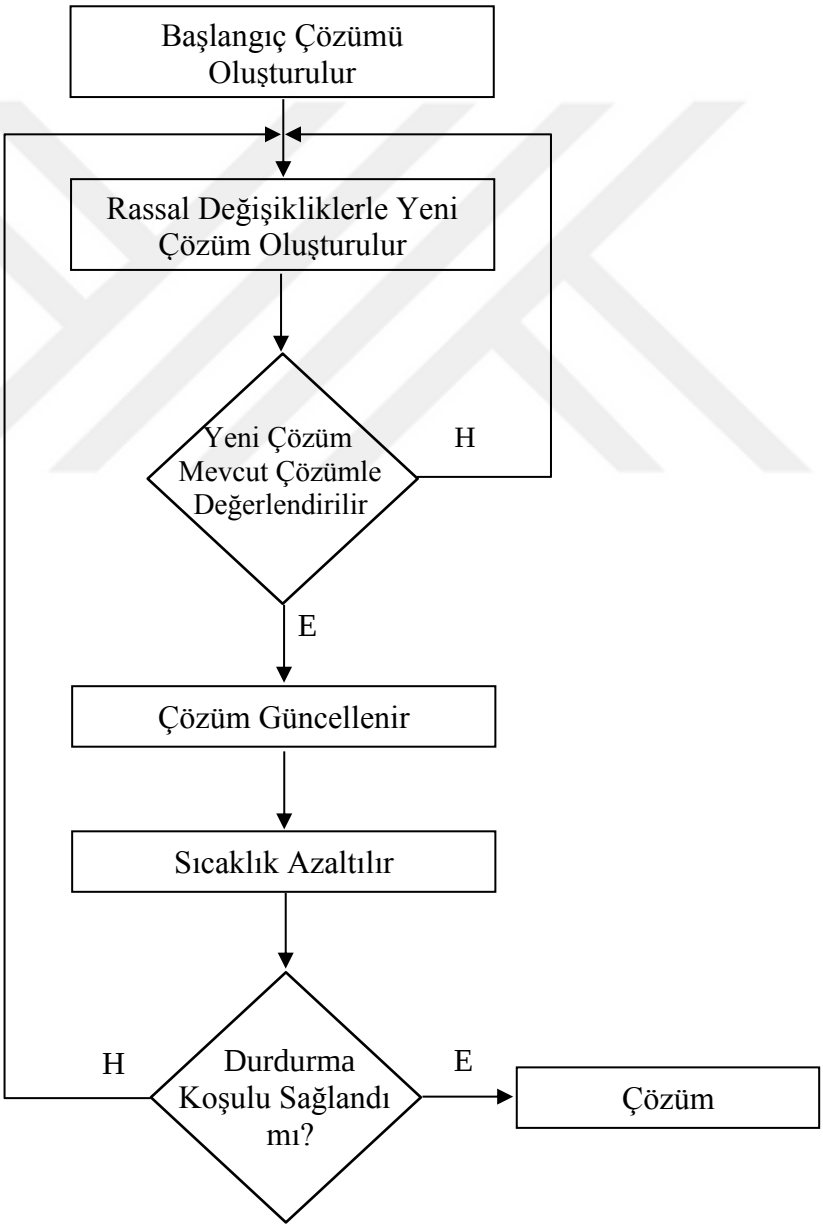
Tavlama Benzetimi yöntemi, ilk olarak 1983 yılında Kirkpatrick ve arkadaşları (Kirkpatrick v.d., 1983) tarafından tanıtılmıştır. Yöntem, katıların, özellikle metallerin, tavlama sürecindeki ısıl işlemlerine benzetilerek geliştirilmiştir. Bundan dolayı yöntemin adı literatürde de Tavlama Benzetimi (Simulated Annealing) olarak geçmektedir. Kirkpatrick ve arkadaşlarına göre, istatistiksel mekanik ve çok değişkenli veya kombinatoryal optimizasyon arasında derin ve kullanışlı bir bağlantı vardır (Kirkpatrick v.d., 1983: 671). Yöntemin geliştirilmesinde faydalanılan ve baz alınan algoritma 1953 yılında, Monte Carlo tekniği üzerine fiziksel tavlama işlemi için geliştirilmiş olan algoritmadır (Metropolis v.d., 1953).

Katıların tavlanması işleminde katılar, öncelikle yüksek ısılara getirilirler ve daha sonra da yavaş yavaş soğutularak belirli bir yapısal duruma gelmeleri sağlanır. Tavlama işlemi sırasında ısı kaybeden elementlerin gerçekleştirdiği hareketler, aslında canlılarda da var olan potansiyel enerjinin minimize edilmesinden ileri gelmektedir. Isıtılmış olan element yavaş yavaş soğutulduğu zaman, içinde bulunan potansiyel enerjiyi minimize etmek adına, dizilimleri bir süre sonra üstün bir yapısal bütünlük gösterecek hale getirir. Bu şekilde ısıl değişimlerle gerçekleştirilen tavlama işlemi gibi bilgisayar ortamında, oluşturulmuş olan bir fonksiyonun minimum ya da maksimum değere sahip olduğu noktalar da Tavlama Benzetimi yöntemi yardımıyla bulunabilmektedir.

Tavlama Benzetimi yöntemi, bir başlangıç noktasından komşuluk arama tekniğini kullanarak optimuma yaklaşmayı hedeflemektedir. Bu yönüyle de yöntemin bir avantajı; çözüm tekniği dolayısıyla da problemin tipine göre lokal minimum ya da maksimuma takılmamasıdır. Tavlama Benzetiminin temel özelliği, global optimumun bulunması amacıyla tepe tırmanma hareketlerine izin vererek yerel optimumdan kaçmayı sağlamasıdır (Henderson v.d., 2003: 287).

Tavlama Benzetimi ile problemlerin çözümünde, öncelikle bir başlangıç çözümü tespit edilir. Daha sonra, oluşturulmuş olan bu başlangıç çözümünde değişiklikler yapılarak yeni çözümler oluşturulur. Yeni çözümlerin değerleri fonksiyonlar yardımıyla hesaplanarak, mevcut çözümden daha iyi olup olmadıkları değerlendirilir.

Oluşturulmuş olan yeni çözüm, mevcut çözümden iyi değil ise tekrar yeni bir çözüm aranır. Eğer oluşturulmuş olan yeni çözüm, mevcut çözümden daha iyi bir değere sahip ise mevcut çözüm yeni çözümle değiştirilir ve sonraki aşamada sıcaklık azaltılarak durdurma koşulu sağlanana kadar bu şekilde işlemlere devam edilir. Diğer sezgisel yöntemlerde olduğu gibi tavlama benzetimi yönteminin de literatürde yer alan farklı uygulamaları mevcuttur. Aşağıda yer alan Şekil 2.12'de, Tavlama Benzetiminin genel çözüm yapısı ana hatlarıyla gösterilmektedir.



Şekil 2.12: Tavlama Benzetimi Çözüm Yapısı.

2.5.1. Tavlama Benzetiminde Kullanılan Yapılar ve Tanımlar

Bu kısımda, istatistiksel mekanikten uyarlanarak geliştirilmiş olan Tavlama Benzetimi yönteminin daha iyi bir şekilde anlaşılabilmesi için yöntemde kullanılan yapılar ve tanımlar ayrıntılı bir şekilde aktarılmaya çalışılacaktır.

Komşu Çözüm: Mevcut çözüm etrafında bulunan çözümlere komşu çözüm adı verilir. Komşu çözümlerin seçilmesi sırasında belirli bir kural ya da bir fonksiyon kullanılır. Problemin çözümünün başlangıcında seçilen çözümden itibaren sonraki çözümler, rastgele olarak seçilebileceği gibi belirli bir kural ya da bir fonksiyon yardımıyla da üretilebilmektedir.

Komşu Fonksiyonu: Bir çözümün değiştirilerek kullanılması ve komşu çözümlerin değerlendirilmesine olanak sağlayan fonksiyondur.

Başlangıç Sıcaklığı: Çözüm algoritmasının başlangıcında ayarlanan sıcaklık değeridir. İyi bir tavlama planında başlangıç sıcaklığı, sistemin düzensiz bir hale ulaşmasını sağlamaya yetecek kadar yüksektir (Durand, M. D., 1989: 10). Buradan da anlaşılmaktadır ki; başlangıç sıcaklığının yüksek olarak seçilmesi, optimum sıcaklık değerine ulaşma ihtimalini arttırmaktadır.

Soğutma Planı: Sıcaklık değerinin sistemli olarak düşürülmesi sırasında uygulanan plandır. Soğutma planları iki sınıfta gruplanır: tamamen algoritma başlamadan önce belirlenen statik planlar ve algoritmanın çalıştırılması sırasında elde edilen bilgilerden sıcaklığın azalma oranını ayarlayan uyarlamalı planlar (Henderson v.d., 2003: 304). Soğutma planına son sıcaklığa ulaşılana kadar; diğer bir ifade ile artık daha fazla iyileştirme mümkün olmayana kadar devam edilir.

Son Sıcaklık: Maliyette önemli bir iyileşme daha fazla beklenmiyorsa, ulaşılan sıcaklıktır (Duque–Anton v.d., 1993: 17). Burada bahsedilen son sıcaklık ifadesi; diğer sezgisel yöntemlere benzer şekilde belirli sayıda iterasyon sonrasında elde edilen değer olabileceği gibi, arka arkaya tekrarlanan iterasyonlar sonrasında daha fazla iyileşmenin beklenmediği değer de olabilir.

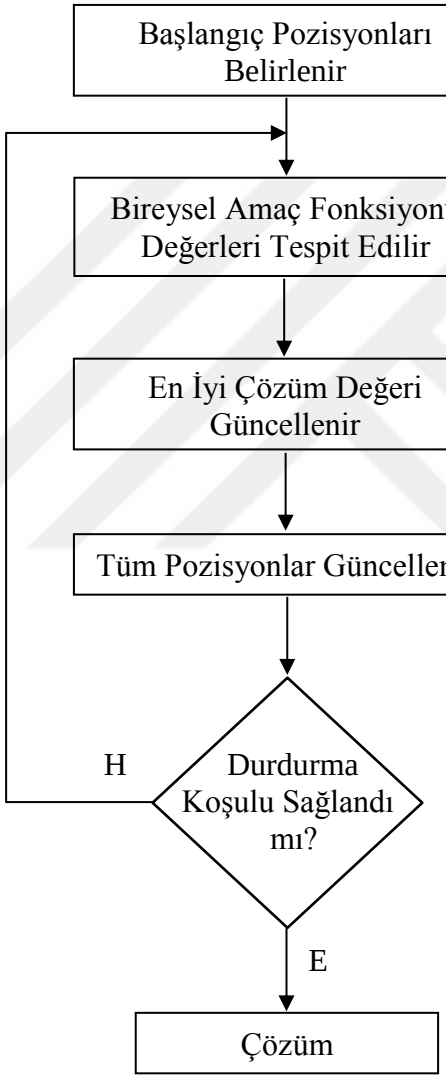
2.6. PARÇACIK SÜRÜ OPTİMİZASYONU

Parçacık Sürü Optimizasyonu yöntemi ilk olarak 1995 yılında Kennedy ve Eberhart tarafından, özellikle balık ve kuş gibi sürüler halinde gezen hayvanların yiyecek aramaları sırasındaki davranışlarından esinlenerek geliştirilmiştir (Eberhart ve Kennedy, 1995). Doğadan esinlenerek geliştirilmiş diğer sezgisel yöntemlerde olduğu gibi bu yöntemde de, kuşların doğal davranışları optimizasyon problemlerine uyarlanmıştır. Karınca Kolonisi Optimizasyonu ve Yapay Arı Kolonisinde olduğu gibi, Parçacık Sürü Optimizasyonunda da çözümde yer alan birimler, aramalar sırasında birlikte hareketler gerçekleştirerek çözüme gitmektedir. Bu şekilde, çözümlerin kalitesi yükselmekte ve optimum çözüme ulaşma daha kısa sürede gerçekleşmektedir.

Doğada sürüler halinde gezen hayvan grupları, yiyecek aramaları sırasında birbirleriyle iletişim halinde bulunurlar ve her zaman yiyecek kaynağına en yakın olan bireyin iletmiş olduğu bilgileri kullanarak o yiyecek kaynağına yönelirler. Örneğin, bir bölgenin üzerinde uçan bir kuş sürüsü olsun. Bu kuş sürüsü başlangıçta bölge üzerinde yayılmış bir halde bulunmaktadır. Her kuş kendi bulunduğu yerdeki yiyecek bilgisini diğerleri ile paylaşır ve diğer kuşlar da en kaliteli yiyecek kaynağını tespit etmiş olan kuşun bulunduğu bölgeye yönelirler. Kuşlar, bu yönelme hareketleri sırasında o bölgeye ulaşırlarken geçecek oldukları yerleri de tarayacaklardır. Böylece, daha iyi bir kaynak bulunduğu zaman sürünün yönelimleri de değişecektir. Kuş sürüleri bu şekilde en iyi yiyecek kaynağını keşfedebileceklerdir.

Parçacık Sürü Optimizasyonu yönteminde benzer şekilde, öncelikle tüm bireylerin başlangıç pozisyonları tespit edilir. Bu yöntemde adından da anlaşılacağı gibi, popülasyonda yer alan bireyler parçacık olarak adlandırılmaktadır. Başlangıç pozisyonları rastgele olarak tespit edilmiş olan parçacıkların komşulukları da taranarak her bir parçacık için en iyi yiyecek konumu belirlenir. Tüm parçacıkların amaç fonksiyonu değerleri karşılaştırılarak en iyi yiyecek konumu tespit edilir. Tespit edilen en iyi yiyecek konumu doğada olduğu gibi diğer parçacıklara da iletilerek tüm parçacıkların yeni yönelimleri düzenlenir ve tüm parçacıklar için yeni pozisyonlar tespit edilir. Bu şekilde, durdurma kriteri sağlanana kadar iterasyonlara

aynı şekilde devam edilir. Tüm bu iterasyonlar adım adım gözlendiği zaman; geniş bir alanda yayılmış olan parçacıkların yavaş yavaş küçülerek bir alana yığılım gerçekleştirdiği ve son olarak da en iyi yiyecek konumunda buluştukları görülür. Aşağıda yer alan Şekil 2.13'de, Parçacık Sürü Optimizasyonunun genel çözüm yapısı ana hatlarıyla gösterilmektedir.



Şekil 2.13: Parçacık Sürü Optimizasyonu Çözüm Yapısı.

Diğer sezgisel yöntemlerde olduğu gibi Parçacık Sürü Optimizasyonunun da literatürde yer alan geliştirilmiş çok çeşitli uyarlamaları bulunmaktadır. Burada aktarılan, en basit haliyle ve genel hatlarıyla gerçekleştirilen bir Parçacık Sürü Optimizasyonudur.

2.6.1. Parçacık Sürü Optimizasyonunda Kullanılan Yapılar ve Tanımlar

Popülasyon tabanlı bir sezgisel optimizasyon yöntemi olan Parçacık Sürü Optimizasyonunda kullanılan önemli yapılar ve tanımlar, bu kısımda ayrıntılı bir şekilde incelenecektir.

Parçacık: Çözüm için oluşturulmuş olan popülasyonda yer alan bireylerden her biri parçacık olarak adlandırılır.

Sürü Büyüklüğü: Çözüm algoritmasında yer alan parçacıkların tamamının oluşturduğu sürü için kullanılan bireyler topluluğu ifadesidir. Genetik Algoritmalarda kullanılan popülasyon ifadesiyle benzerdir. Literatürde yer alan uygulamalarda, sürü büyüklüğü genellikle 30 civarında alınmasına karşın, büyük boyutlu ve geniş kapsamlı problemlerde 100 üzerinde parçacık alındığı da olmaktadır. Benzer şekilde küçük boyutlu problemlerde de, 10'un altında olan sayılarda da parçacık alındığı görülmektedir. Problemlerin çeşidine ve çözüm algoritmasına göre parçacık sayılarının doğru bir miktarda ayarlanması önemli bir durumdur. Bu durum basit bir örnekle incelenecek olursa; küçük boyutlu basit bir problemde çok fazla sayıda parçacık tercih edilmiş olması, problemin çözümüne hızlı bir şekilde ulaşmaktan ziyade gereksiz işlem yoğunluğu oluşturacağından dolayı, optimum çözüme ulaşma süresini hayli uzatabilmektedir.

Deneyim: Deneyim kavramı, parçacıkların geçmiş tecrübelerinden elde ettikleri pozisyon bilgilerini ifade etmektedir. Bundan dolayı da literatürde, deneyim ifadesiyle birlikte pozisyon ifadesinin kullanımı da mevcuttur. Uygulamalarda kullanılan iki tip deneyim bilgisi bulunmaktadır; parçacığın deneyimi ve sürünün deneyimi. Parçacığın deneyimi, parçacıkların kendi bireysel tecrübelerini, pozisyon bilgilerini ifade ederken, sürünün deneyimi de sürüde yer alan tüm parçacıkların pozisyonlarını ifade etmektedir.

Parçacık Hızı: Parçacık hızı, parçacığın pozisyonundaki değişim oranını ifade etmektedir. Parçacık Sürü Optimizasyonunda yer alan her bir birey, hem kendi bireysel uçuş deneyimlerine, hem de diğerlerinin uçuş deneyimlerine göre dinamik olarak ayarlanan bir hız ile arama uzayında uçarlar (Khan ve Sahai, 2012: 26).

Literatürde, parçacık hızlarının yalnızca bireysel deneyimlerle belirlendiği çalışmalar da mevcuttur. Parçacıkların hızının ve yönünün kontrolü ise atalet ağırlığı tarafından gerçekleştirilmektedir. Parçacıkların pozisyon değiştirmesinde kullanılan ilk parametrenin hız olduğu düşünülecek olursa, hızın doğru ve verimli bir şekilde ayarlanması, çözüm algoritmasının ilerleyişi açısından önem teşkil etmektedir.

Atalet Ağırlığı: Atalet ağırlığı kavramı ilk olarak, 1998 yılında Shi ve Eberhart tarafından tanıtılmıştır (Shi ve Eberhart, 1998). Atalet ağırlığı, parçacıkların çözüm uzayındaki hareketlerinin hızının ve yönünün kontrol edilmesine olanak sağlamaktadır. Parçacık Sürü Optimizasyonunun en önemli özelliğinin parçacıkların iletişim halinde ve birlikte hareketleri olduğu düşünülürse, atalet ağırlığı da çözüm için önemli bir parametre haline gelmektedir. Atalet ağırlığı, arama ve işletme süreci arasında denge sağlama sürecinde kilit bir rol oynar (Bansal v.d., 2011: 633). O halde, yanlış ayarlanmış bir atalet ağırlığı parametresi, çözüme ulaşmayı zorlaştırmanın yanı sıra yanlış yönde ilerlemelerle çözüme hiç ulaşılamamasına da yol açabilecektir.

Destek ve Güven: Destek ve güven, bir kuralın tutulup tutulmayacağının belirlenmesi için değerlendirilir (Vyas ve Chauhan, 2013: 259). Destek ve güven sayesinde, düşük kalitede olan kurallar elenir ve belirli bir kalitenin üzerindeki kurallar da tespit edilerek saklanırlar. Başlangıçta, asgari destek ve güven kullanıcı tarafından tanımlanarak girdi olarak alınır ve daha sonra ise problemden elde edilen verilerle bu asgari destek ve güven güncellenir. En iyi parçacığın destek ve güveni, asgari destek ve asgari güveni temsil edebilir (Kuo v.d., 2011: 330).

Özgüven: Parçacıkların hızlanma hareketlerinin hesaplanması sırasında, her bir parçacığın kendi bireysel deneyimlerinin dahil edilmesi için kullanılan katsayıya, özgüven katsayısı adı verilir.

Sürüye Güven: Özgüven katsayısına benzer şekilde, parçacıkların hızlanma hareketlerinin hesaplanması sırasında, sürü deneyimlerinin dahil edilmesi için kullanılan katsayıya sürüye güven katsayısı adı verilir. Literatürde, sosyal bileşen olarak da ifade edilmektedir.

2.7. DİĞER SEZGİSEL YÖNTEMLER

Buraya kadar aktarılmış olan sezgisel yöntemler, literatürde de yaygın kullanım görmüş ve farklı uygulama alanları için de tercih edilmiş olan yöntemlerdir. Literatürde, buraya kadar aktarılmış olan altı yöntemden çok daha fazla sezgisel yöntem bulunmaktadır ve bu kısımda da, literatürde yer alan ve erişilebilmiş olan diğer sezgisel yöntemlerden kısaca bahsedilmeye çalışılacaktır. Aşağıda yer alan sezgisel yöntemlerin büyük bir çoğunluğu, uygulamada az kullanım görmüş ya da diğer temel sezgisel yöntemlerden esinlenilerek geliştirilmiştir.

Yapay Sinir Ağları (Artificial Neural Networks): Yapay Sinir Ağlarına (YSA) dair yapılan çalışmaların temeli, ilk olarak 1943 yılında McCulloch ve Pitts (McCulloch ve Pitts, 1943) tarafından gerçekleştirilen çalışmaya dayanmaktadır. Bu çalışmayı takip eden yirmi beş yıl boyunca, YSA adına sınırlı sayıda çalışma gerçekleştirilmiştir. Daha sonra, Minsky ve Papert 1969 (Minsky ve Papert, 1969) yılında yayınlamış oldukları kitapta, bilimsel kaygıları da gidererek Yapay Sinir Ağlarının kullanımının önünü açmışlardır. Günümüzde oldukça yaygın kullanımı bulunan YSA'nın gelişimi, 90'lı yıllardan itibaren teknolojinin gelişmesiyle birlikte üssel olarak artan bir hız kazanmıştır.

Yapay Bağışıklık Sistemi (Artificial Immune System): Yöntem, ilk olarak 1990 yılında Ishida tarafından, Bağışıklık Ağ Modeli olarak gerçekleştirilen çalışmada geçmektedir (Ishida, 1990). Bu çalışma temel alınarak ve günümüze kadar gerçekleştirilen uygulamalarla güncellenerek, bugünkü şekli olan Yapay Bağışıklık Sistemi (YBS) halini almıştır. Adından da anlaşılacağı gibi yöntem, yapay bir bağışıklık sistemi üzerine geliştirilmiştir.

Diferansiyel Gelişim Algoritması (Differential Evolution Algorithm): Global optimizasyon arama problemlerinin çözümü için, ilk olarak 1995 yılında Storn ve Price tarafından "Differential Evolution" olarak tanıtılmıştır (Storn ve Price, 1995). İlerleyen yıllarda geliştirilerek bugünkü Diferansiyel Gelişim Algoritması (DGA) halini almıştır. DGA, çok boyutlu vektörlerden faydalanan paralel doğrudan arama yöntemidir.

Harmoni Arama (Harmony Search): Yöntem, ilk olarak 2001 yılında Geem ve arkadaşları tarafından tanıtılmıştır (Geem v.d., 2001). Müzikten ve müzisyenlerden esinlenerek geliştirilmiş olan bir Harmoni Arama (HA) yöntemidir. Müzisyenlerin yaptığı gibi doğaçlamalar vasıtasıyla yerel ve global arama planları gerçekleştirilerek, optimum değerlere ulaşmaya çalışır.

Yerçekimsel Arama Algoritması (Gravitational Search Algorithm): Yerçekimsel Arama Algoritmasından (YAA) ilk olarak 2007 yılında Rashedi (Rashedi, 2007) tarafından Shahid Bahonar University of Kerman'da gerçekleştirmiş olan tez çalışmasında bahsedilmiştir. Daha sonra ilk olarak 2009 yılında, Rashedi ve arkadaşları (Rashedi v.d., 2009) tarafından geliştirilmiştir. Yöntemin geliştirilmesinde, 1993 yılında Holliday ve arkadaşları (Holliday v.d., 1993) tarafından yayınlanmış olan fiziğin temelleri kitabında yer alan yerçekimi kanunundan esinlenilmiştir.

Geliştiricileri ya da kaynaklarına dair net bilgiler bulunamamış olan fakat literatürde kullanımı bulunan diğer bazı sezgisel yöntemler de aşağıda listelenmiştir.

- Açgözlü Arama (Greedy Search)
- Değişken Komşuluk Arama (Variable Neighborhood Search)
- Tepe Tırmanma Algoritması (Hill-Climbing Algorithm)
- Demet Arama (Beam Search)
- Dağınık Arama (Scatter Search)
- A* Arama Algoritması (A* Search Algorithm)
- Yarasa Algoritması (Bat Algorithm)
- Kurbağa Sıçrama Algoritması (Frog Leaping Algorithm)
- Ateşböceği Algoritması (Firefly Algorithm)
- Guguk Kuşu Arama Algoritması (Cuckoo Search Algorithm)
- Çok Sürülü Optimizasyon (Multi-Swarm Optimization)
- Kendinden Hareketli Parçacıklar (Self-Propelled Particles)
- Akıllı Su Damlaları (Intelligent Water Drops)
- Stokastik Difüzyon Araması (Stochastic Diffusion Search)
- Fotosentez ve Enzim Algoritması (Photosynthetic and Enzyme Algorithm)

- Emperyalist Rekabetçi Algoritma (Imperialist Competitive Algorithm)
- Göç Eden Kuşlar Optimizasyonu (Migrating Birds Optimization)
- Tekrarlanan Yerel Arama (Iterated Local Search)
- Grup Arama Eniyileyici (Group Search Optimizer)
- Sosyal Bilişsel Optimizasyon (Social Cognitive Optimization)
- Kuantum Tavlama (Quantum Annealing)
- Ekstremum Optimizasyonu (Extremal Optimization)

3. ÜNİVERSİTE SINAV ÇİZELGELEME

Tez çalışmasının bu kısmında, üzerine çalışılmış olan üniversite sınav çizelgeleme problemine dair ayrıntılı olarak yapılacak incelemelere yer verilecektir. Bu amaçla, öncelikle üniversite sınav çizelgeleme probleminin tanımı yapılacak ve probleminde yer alan bileşenler, ayrıntılı olarak tanımlamalarıyla birlikte aktarılacaktır. Daha sonra da geniş kapsamlı bir literatür taraması gerçekleştirilecektir. Literatür taramasından elde edilen bulgular da olabildiğince karşılaştırmalı olarak incelenmeye çalışılacaktır.

Literatürde yer alan üniversite sınav çizelgeleme çalışmalarında, genellikle, gerçek hayat problemlerinin çözümleri üzerine odaklanıldığı görülmektedir. Gerçek hayat problemleri ile gerçekleştirilen uygulamaların yanında literatürde, gerçek hayat problemleri olan ve internet üzerinden açık erişime sunulmuş olan bir takım kıyaslama veri setleri de bulunmaktadır. Herkesin erişimine açık olan bu kıyaslama veri setleri ile önerilen algoritmalar test edilerek sonuçlar raporlanmakta ve algoritmaların çözüm kaliteleri ve sonuçları birbirleri ile karşılaştırılabilmektedir. Böylece, önerilen algoritmaların ve sezgisel yöntemlerin başarılı veya eksik yönleri daha iyi bir şekilde görülebilmektedir. Bu detaylı karşılaştırmalar sayesinde de geliştirilecek olan yeni algoritmalar, daha başarılı bir şekilde oluşturulma ve daha iyi sonuçlar elde etme şansına sahip olabilecektir.

Literatürde yayınlanmış olan geniş kapsamlı ilk kıyaslama veri seti 1996 yılında Carter ve arkadaşları (Carter v.d., 1996) tarafından gerçekleştirilen çalışmada kullanılan Toronto Üniversitesi Kıyaslama Veri Setidir[1]. Toronto Üniversitesi Kıyaslama Veri Seti, aynı zamanda literatürde en çok kullanılan kıyaslama veri setidir. Çalışmalarda sıklıkla kullanılan diğer kıyaslama veri setleri ise 1996 yılında Burke ve arkadaşları (Burke v.d., 1996) tarafından yayınlanan Nottingham

[1] ftp://ftp.mie.utoronto.ca/pub/carter/testprob/, Erişim Tarihi: 01.07.2015.

Üniversitesi Kıyaslama Veri Seti[2], Merlot ve arkadaşları (Merlot v.d., 2002) tarafından 2002 yılında gerçekleştirilen IV. PATAT (The Practice and Theory of Automated Timetabling IV) konferansında tanıtılan Melbourne Üniversitesi Kıyaslama Veri Seti[3] ve McCollum ve arkadaşları (McCollum v.d., 2007) tarafından 2007 yılında gerçekleştirilen İkinci Uluslararası Çizelgeleme Müsabakasında (International Timetabling Competition 2007) tanıtılan ITC2007 veri setidir[4]. Aktarılacak olan ayrıntılı literatür taraması kıyaslama veri setlerine göre gerçekleştirilecektir. Bundan dolayı da kıyaslama veri setlerinde yer alan verilere ait detaylı bilgiler literatür taramasında ilgili başlıkların altında aktarılacaktır.

3.1. ÜNİVERSİTE SINAV ÇİZELGELEME PROBLEMİ VE BİLEŞENLERİ

Tez çalışmasının bu kısmında gerek literatür taramasının gerekse de gerçekleştirilmiş olan tez çalışmasının daha iyi bir şekilde aktarılabilmesi için Üniversite Sınav Çizelgeleme Problemi (ÜSÇP) incelenecektir. Bu amaçla öncelikle, üniversite sınav çizelgeleme tanımlanacak ve sonrasında da ÜSÇP'nin bileşenleri ayrıntılı olarak incelenecektir.

3.1.1. Üniversite Sınav Çizelgeleme Problemi

ÜSÇP, üniversitelerin gerçekleştirecek olduğu sınavları, belirli bir zaman diliminde ya da olabildiğince az kuşak sayısında, üniversitenin mevcut imkanlarını etkin bir şekilde kullanılacak şekilde bir sınav takviminin oluşturulmasıdır. Problemin zor olan kısımlarından biri, öğrenciler aynı anda iki derslikte bulunamayacağından dolayı hiçbir öğrencinin herhangi iki sınavının aynı kuşakta yer almaması gerekliliğidir. Doğru oluşturulmamış bir algoritmada her sınav kuşağı ve her ders için öğrencilerin derslerinin tek tek karşılaştırılması hem çözümü zorlaştırmakta hem de problemlerin çözüm zamanlarını gereksiz yere uzatabilmektedir. Bu durumun yanında, üniversitelerin derslik ve gözetmen kısıtları da problemi daha da zorlaştırmaktadır. Tüm bu kısıtlamaların yanında üniversitelere veya fakültelere özgü kısıtlamalar da bulunmaktadır. Kısıtlamalar arttıkça problemin çözümü ve sınav takviminin oluşturulması daha da zorlaşmaktadır.

[2] http://www.cs.nott.ac.uk/~pszrq/data.htm, Erişim Tarihi: 01.07.2015.
[3] http://www.cs.nott.ac.uk/~pszrq/data.htm, Erişim Tarihi: 01.07.2015.
[4] http://www.cs.qub.ac.uk/itc2007/index_files/competitiontracks.htm, Erişim Tarihi: 01.07.2015.

Literatürde birbirinden farklı ÜSÇP ve çözümleri yer almaktadır. Yalnızca sınav takvimi oluşturmaya yönelik olarak literatürde yer alan çalışmalar, öğrencilere ait kısıtlamaların dışında genellikle iki unsur üzerinde yoğunlaşmaktadır; sınav kuşağı sayısı ve derslik. Bu kısıtlamaların bulunduğu problemlerde farklı sınav süreleri de bulunduğu taktirde ÜSÇP'nin çözümü daha da zor bir hale gelebilmektedir. Bazı problemlerde 3 saat gibi çok uzun süreli sınavlar da yarım saat gibi çok kısa süreli sınavlar da bulunabilmektedir. Böyle bir problemde, sınav süresi çok kısa olan bir sınavla sınav süresi çok uzun olan bir sınavın aynı kuşağa koyulması, hem derslikleri gereksiz yere işgal ederek sınavların daha geç bitmesine, hem de sınavdan geç çıkan öğrencilerin dikkatlerinin dağılmasına neden olacaktır. Aynı kuşağa üç, dört farklı sınav süresine sahip sınavların yerleştirilmesi, öğrenciler için çok daha kötü bir durum oluşturacaktır. Burada aktarılan kısıtlamalar, ÜSÇP'nin en temel kısıtlamalarıdır. Bu kısıtlamaların yanında sınav takviminin oluşturulmasına yönelik çok çeşitli kısıtlamalar da bulunabilmektedir. Bu kısıtlamalara da kısaca bir sonraki başlıkta değinilecektir.

Burada aktarılan temel kısıtlamalar dikkate alındığı taktirde ÜSÇP, aşağıdaki şekilde üç grup altında toplanabilir.

- Kuşak sayısını en az sayıda tutmaya çalışan problemler ve çözümler
- Belirli kuşak sayılarında gerçekleştirilen problemler ve çözümler
- Derslik kısıtlaması bulunan problemler ve çözümler

Burada aktarılan tüm ÜSÇP ve çözümleri bu kısıtlamaların yanında, öğrencilere sınavları arasında olabildiğince fazla çalışma zamanı tanıyabilecek şekilde bir sınav takvimi oluşturmayı hedeflemektedir.

3.1.2. Üniversite Sınav Çizelgeleme Probleminin Bileşenleri

Çalışmanın bu kısmında, ÜSÇP'nde yer alan bileşenler tanımlanarak detaylı bir şekilde incelenecektir. İncelenecek olan bu bileşenler literatürde yer alan çalışmalarda da bulunan ve genel olarak kullanılan tanımlamalar ve kısıtlamalardır. Her bir üniversitenin veya fakültenin kendine özgü tanımlamaları ya da uygulamaları

bulunabilmektedir. Burada aktarılacak olan ifadeler, en genel kullanımı bulunan bileşenler olacaktır.

Üniversite: Türk Dil Kurumu tarafından yapılan tanımlamaya göre üniversite, "Bilimsel özerkliğe ve kamu tüzel kişiliğine sahip, yüksek düzeyde eğitim, öğretim, bilimsel araştırma ve yayın yapan fakülte, enstitü, yüksekokul vb. kuruluş ve birimlerden oluşan öğretim kurumu, darülfünun" olarak ifade edilmektedir. Tanımlamada göstermektedir ki; üniversiteler birbirinden farklı çok çeşitli birimlerin bulunduğu bir öğretim kurumlarıdır. Fakülteler, enstitüler, yüksekokullar gibi farklı birimlerin ve bu birimlerin altında bulunan çok çeşitli birim ve bölümlerden oluşan bir yapıya sahip olan üniversiteler; bu farklılık ve çeşitliliklerle birlikte geniş bir alana yayılma ve farklı kampüslerde öğretim yapma gerekliliğinde olmaktadırlar. Özellikle, farklı kampüslerde derslerini gerçekleştiren üniversiteler ve birimler için sınav çizelgelemesinin gerçekleştirilmesi sırasında bu durumların dikkate alınması en önemli kısıt haline gelmektedir.

Öğrenci: Bir öğretim kurumunda öğrenim amacıyla derslere katılan kişi olarak tanımlaması yapılabilecek olan öğrenci, sınav takviminin oluşturulması sırasında dikkate alınacak en temel bileşendir. Sınav takvimlerinin oluşturulması sırasında dikkate alınan kısıtların büyük bir çoğunluğu öğrenciler üzerinedir.

Ders: Öğrencilere belirli bir bilgiyi aktarma ya da bir takım beceriler kazandırmak için ayrılan süredir. Günümüzde üniversitelerde işlenen dersler yalnızca sınıflarda gerçekleştirilmemektedir. Hızla gelişen teknolojiyle birlikte üniversiteler bazı dersleri uzaktan öğretimde olduğu gibi internet üzerinden gerçekleştirebilmektedir. Bu durum da derslik, kapasite vs gibi imkanlardan bağımsız bir şekilde planlama yapılmasına imkan sağlamaktadır.

Sınav: Öğrencilerin bilgi ve beceri düzeylerinin tespit edilmesi için yapılan kontroldür. Üniversitelerde, özellikle lisans düzeyinde işlenen derslere ait sınavlar dersliklerde ve belirli zaman aralıklarında gerçekleştirilmektedir. Bazı derslerde devam zorunluluğunun bulunmaması dolayısıyla öğrenciler birbirinden bağımsız ve aynı ders saatine denk gelen çeşitli dersleri alabilmektedirler. Bu gibi durumlarda

derslere ait sınavların birbirinden bağımsız bir şekilde gerçekleştirilmesini imkansız hale getirmektedir. Üniversitelerde sınavlar için belirli bir zaman dilimi, genellikle bir ya da iki haftalık bir süre ayrılarak, sınavlar bu zaman diliminde gerçekleştirilir. Bu durum da tüm öğrencileri dikkate alan ve tüm öğrencilerin tüm sınavlarına girebilmesini sağlayacak bir sınav takviminin oluşturulması gerekliliği anlamına gelmektedir.

Sınav Salonu: Sınavların gerçekleştirildiği derslik veya salondur. Sınav salonu kısıtı da sınav çizelgeleme probleminin önemli olan kısıtlarından biridir. İki farklı derse ait sınavın aynı sınav kuşağında gerçekleştirilebilmesi mümkün olabilir, fakat o iki sınava girecek tüm öğrenciler için yeterli sınav salonu bulunmuyorsa eğer, böyle bir planlamanın yapılması imkansız hale gelebilmektedir. Herhangi bir ÜSÇP'nin çözümü sırasında, her adımda bu kontrolün de gerçekleştirilmesi ve sınav takviminin bu plana göre oluşturulması gerekmektedir. Bu kısıt da, çözüm zamanını uzatabilen ve problemi zorlaştıran faktörlerden biridir.

Sınav Süresi: Öğrencilerin sınavı tamamlaması için verilen maksimum süredir. Sınav süreleri, derslere ve fakültelere göre değişiklikler gösterebilmektedir. Bazı fakülteler ve bölümler, standart olarak tüm sınavlar için belirli bir zaman aralığını şart koşabilmekte ve sınav süreleri bu zaman aralığından fazla ya da az olamamaktadır. Bazılarında ise sınav süreleri tamamen serbest bırakılabilmekte ve sınav takvimi, önceden belirlenmiş bu değişken zamanlara göre oluşturulabilmektedir. Fakat daha önceden de değinildiği gibi, sınav sürelerinin dikkate alındığı problemlerde, birbirinden farklı sınav süresine sahip sınavların aynı sınav kuşağına koyulmamasına özen gösterilmesi gerekmektedir.

Gözetmen: Sınavların usulüne uygun bir şekilde gerçekleştirilmesiyle görevli kişilerdir. Üniversitelerde, sınavların yürütülmesiyle görevli olan kişiler genellikle öğretim elemanları ve öğretim üyeleri olmaktadır. Yeterli sayıda öğretim elemanına sahip olan bazı fakültelerde, öğretim üyeleri olmadan yalnızca öğretim elemanları da sınavları yürütmekle görevli olabilmektedir. Sınav takviminin oluşturulması sırasında etkin bir bileşen olmayan gözetmenler, sınav görevli atamasının gerçekleştirildiği çizelgeleme problemlerinde etkin bir bileşen haline gelmektedir.

Literatürde, gözetmen tercihlerine göre gözetmen ataması gerçekleştiren çalışmalar olduğu gibi, hiçbir tercihi dikkate almadan gözetmen atama işlemini gerçekleştiren çalışmalar da mevcuttur. Bu gözetmen atama işleminde gerçekleştirilen ana prosedür de sınav takviminin oluşturulması sırasında gerçekleştirilen prosedür gibidir. Yalnızca, algoritmada bir takım küçük değişikler ve kısıtlamalarda farklılıklar bulunmaktadır.

Sınav Takvimi: Sınavların gerçekleştirildiği zaman dilimini ve bu zaman diliminde sınavların yerleşimini belirten takvim. Sınav takvimleri, derslerin yoğunluğuna göre genellikle bir ya da iki haftalık süreyi kapsar.

Sınav Kuşağı: Sınav takvimi içerisinde, sınavların gerçekleştirildiği belirli zaman aralıklarından her biri. Uygulamalarda sınav kuşakları iki farklı şekilde kullanılmaktadır. En sık kullanılan sınav kuşağı sistemi, sabit zamanlarda başlayan sınav kuşağı sistemidir. Bu tip uygulamalarda, bir günde gerçekleştirilecek sınav kuşağı sayısı ve sınav kuşaklarının başlama zamanları sabittir ve sınavlar bu kuşaklara atanarak uygulamalar gerçekleştirilir. İkinci tip sınav kuşağı uygulaması ise tüm sınavların farklı sınav sürelerine sahip olduğu uygulamalarda kullanılan sınav kuşağı sistemidir. Bu tip uygulamalarda tüm sınavların gerçekleştirileceği sınav süreleri bilinmektedir, fakat sınav kuşakları ve bir günde gerçekleştirilecek sınav kuşağı sayısı bilinmemektedir. Optimum sınav takviminin oluşturulması sırasında sınav süreleri de dikkate alınarak sınav kuşakları belirlenir ve sınav kuşaklarının yerleşimleri de bir günlük çalışma zamanı içerisine en uygun şekilde atanarak gerçekleştirilir.

Sınav Çizelgeleme: Belirli kısıtlar göz önünde bulundurularak elle ya da bilgisayar yardımıyla sınav takviminin oluşturulması işlemi. Sınav çizelgeleme problemleri literatürde iki farklı şekilde işlenmektedir. En fazla çalışılan sınav çizelgeleme problemi, yalnızca sınavların yerleşimi ve sınav takviminin oluşturulması işlemini kapsayan sınav çizelgeleme problemleridir. Diğer bir sınav çizelgeleme problemi ise sınav takviminde sınavların yürütülmesinde görevli olan gözetmenlerin atamasının gerçekleştirilmesi işlemini kapsayan sınav çizelgeleme problemleridir. Bu grupta

bulunan sınav çizelgeleme problemleri zaman değişkenliğinin de bulunmamasından dolayı, bazı kaynaklara göre atama problemleri olarak da nitelendirilebilmektedir.

Katı Kısıtlar (Hard Constraints): Sınav çizelgeleme işlemi sırasında kesinlikle dikkate alınması ve sağlanması gereken kısıtlar. Örneğin bir öğrencinin aynı anda iki farklı sınıfta bulunamaması, bir öğretim üyesinin aynı anda iki farklı salonda görevli olamaması, bir dersin farklı şubelere ait sınavlarının farklı zaman dilimlerinde gerçekleştirilmemesi gibi fiziken de gerçekleşmesinin mümkün olmadığı ya da zor olduğu kısıtlardır. Sınav çizelgeleme problemlerinde dikkate alınan temel katı kısıtlar genellikle aynıdır. Katı kısıtlar, üniversitelerin veya fakültelerin sınav kurallarına göre değişiklik gösterebilmektedir. Katı kısıtlar, öğrencilerin tüm sınavlarına girmesinin mümkün kılınması ve sınavların sorunsuz bir şekilde gerçekleştirilmesini sağlayan temel kurallar bütünüdür.

Hafif Kısıtlar (Soft Constraints): Sınav çizelgeleme işlemi sırasında kesinlikle dikkate alınması gerekmeyen, fakat olabildiğince yüksek seviyede sağlanması istenen kısıtlardır. Literatürde yer alan ve uygulamalarda kullanılan hafif kısıtlar, katı kısıtlar kadar sınırlı değildir. Her bir üniversitenin dikkate aldığı hafif kısıtlar farklılık göstermektedir ve tüm bu hafif kısıtların dikkate alınması mümkün değildir.

Literatürde en sık kullanılan katı kısıtlar ve hafif kısıtlar bir sonraki başlıkta aktarılmaya çalışılacaktır.

Ceza Fonksiyonu: Sınav çizelgeleme çözümünde dikkate alınan hafif kısıtların ne derece sağlandığının ölçülebilmesi için oluşturulmuş ve olabildiğince minimize edilmeye çalışılan matematiksel fonksiyon. Çalışmalarda belirli ceza fonksiyonlarının kullanılmasının bir diğer nedeni de, önerilen algoritmaların sonuçlarının ve etkinliklerinin karşılaştırılabilirliğinin sağlanmasıdır.

3.1.2.1. Üniversite Sınav Çizelgeleme Problemlerinde Yer Alan Kısıtlar

ÜSÇP'nin çözümünde dikkate alınan kısıtlar iki ana başlıkta toplanmaktadır; katı kısıtlar ve hafif kısıtlar. ÜSÇP'nde yer alan bu katı kısıtlar ve hafif kısıtlar farklı problemlerde çok çeşitli olabilmektedir. Literatürde yer alan veya üniversitelerde kullanılan farklı tüm kısıtlar incelenecek olursa yüzlerce kısıt ortaya çıkacaktır ve

tüm bu kısıtların dikkate alınması imkansızdır. Literatürde yer alan veya üniversitelerde kullanılan katı kısıtların ve hafif kısıtların en fazla kullanılanları aşağıda maddeler halinde aktarılmaktadır. Çalışmalarda ve uygulamalarda farklı yaklaşımlar yer aldığından dolayı burada aktarılan kısıtlar için, katı kısıtlar ve hafif kısıtlar olarak bir ayrım yapılmamıştır.

- Bir öğrencinin iki farklı sınavı aynı sınav kuşağına yerleştirilemez
- Her bir sınav tek bir sınav kuşağında gerçekleştirilmelidir
- Her bir sınav yeterli sayıda sınav salonuna atanmalıdır
- Bir sınav salonuna iki farklı sınav atanamaz
- Öğrencilerin birbirini takip eden iki sınavı arasında olabildiğince fazla çalışma zamanı bırakılmalıdır
- Öğrencilerin bir günde gireceği sınav sayısı olabildiğince azaltılmalıdır
- Arka arkaya iki gün sınava giren öğrenci sayısı olabildiğince azaltılmalıdır
- Her bir sınav kuşağı için sınav salonu sayıları ve kapasiteleri aşılmamalıdır
- Özel sınav salonlarında gerçekleştirilmesi gereken sınavlar, ilgili sınav salonlarına atanmalıdır
- Belirli bir sınav kuşağında gerçekleştirilmesi gereken sınavlar ilgili sınav kuşaklarına atanmalıdır
- Bazı sınavlarda öncelik ve sonralık ilişkisi bulunabilmektedir ve bu sınavların sınav kuşaklarına atanması sırasında bu ilişkiler dikkate alınmalıdır
- Farklı kampüslerde ya da binalarda gerçekleştirilen sınavlar için uzaklık faktörleri dikkate alınmalıdır
- Süreleri farklı olan sınavlar için aynı ya da yakın sınav sürelerine sahip sınavlar aynı sınav kuşağına atanmalıdır
- Zor olan sınavlar aynı güne ya da birbirini takip eden günlere koyulmamalıdır

Gerçekleştirilen çalışmalarda bu kısıtların tamamının katı kısıt olarak dikkate alınması imkansızdır. Bundan dolayı çözümler sırasında bir ceza fonksiyonu oluşturularak, hafif kısıtlar bu ceza fonksiyonuna bağlı olarak olabildiğince yüksek seviyede sağlanmaya çalışılır. Çözüm yöntemlerinin karşılaştırılabilirliğinin de sağlanması adına, literatürde yer alan belirli ceza fonksiyonları bulunmaktadır.

3.2. LİTERATÜR TARAMASI

Üniversite sınav çizelgeleme problemi, özellikle bilgi teknolojilerinin gelişimiyle birlikte sık incelenen alanlardan biri olmuştur. Problem, hem değişken sayısının hem de kısıt sayısının fazlalığı dolayısıyla zor problemler kategorisinde yer almaktadır. Problemlerin çözümlerinin fazla sayıda yerel minimuma sahip olması dolayısıyla da, deterministik yöntemlerle çözümleri imkansız hale gelmektedir. Bundan dolayı da literatürde yer alan çözümlerde en sık tercih edilen yöntemler, sezgisel yöntemler olmaktadır. Literatürde, çok fazla sayıda üniversite çizelgeleme çalışması bulunmaktadır. Bundan dolayı, burada gerçekleştirilecek olan literatür taramasında, yalnızca üniversite sınav çizelgeleme problemi üzerine yapılmış olan çalışmalara odaklanılacaktır.

3.2.1. Toronto Üniversitesi Kıyaslama Veri Seti Kullanılan Çalışmalar

Toronto Üniversitesi Kıyaslama Veri Seti (TÜKVS) ilk olarak 1996 yılında Carter ve arkadaşları (Carter v.d., 1996) tarafından yayınlanmış olan çalışmada kullanılan ve tanıtılan bir veri setidir. Veri setinde farklı üniversitelere ve fakültelere ait 13 adet veri seti yer almaktadır. Veri setinde yer alan sınav verileri birbirinden farklı sayıda ders, farklı sayıda öğrenci ve birbirinden farklı yoğunluk değerlerine sahip olduklarından dolayı, algoritmaların kıyaslanması için kullanılabilecek en etkin kıyaslama veri setidir. TÜKVS'ne ait detaylı bilgiler, Tablo 3.1'de ayrıntılı olarak verilmektedir.

Tablo 3.1: Toronto Üniversitesi Kıyaslama Veri Seti (Carter v.d., 1996: 376).

Kod	Fakülte / Üniversite	Sınav Sayısı	Öğrenci Sayısı	Öğrenci Sınavları Sayısı	Çakışma Matrisinin Yoğunluğu
CAR–F–92	Carleton University, Ottawa	543	18419	54062	0,14
CAR–S–91	Carleton University, Ottawa	682	16925	59022	0,13
EAR–F–83	Earl Haig Collegiate Institute, Toronto	181	941	6029	0,27
HEC–S–92	Ecole des Hautes Etudes Commerciales, Montreal	81	2823	10634	0,2
KFU–S–93	King Fahd University of Petroleum and Minerals, Dharan	486	5349	25118	0,06
LSE–F–91	London School of Economics	381	2726	10919	0,06
PUR–S–93	Purdue University, Indiana	2419	30032	120690	0,03
RYE–F–92	Ryerson University, Toronto	487	11483	45052	0,07
STA–F–83	St. Andrew's Junior High School, Toronto	139	611	5539	0,14
TRE–S–92	Trent University, Peterborough, Ontario	261	4360	14901	0,18
UTA–S–92	Faculty of Arts and Sciences, University of Toronto	622	21267	58981	0,13
UTE–S–92	Faculty of Engineering, University of Toronto	184	2750	11796	0,08
YOR–F–83	York Mills Collegiate Institute, Toronto	190	1125	8108	0,29

Carter ve arkadaşları yapmış oldukları çalışmada, beş farklı algoritma stratejisini öğrenci başına ceza puanlarını dikkate alarak ve almayarak uygulamışlardır. Bu uygulama neticesinde hem gerçekleştirilmiş olan beş farklı yöntemin karşılaştırma sonuçları elde edilebilmiş, hem de ceza puanı kullanımının algoritmaya olan etkisi gözlemlenebilmiştir. Carter ve arkadaşlarının TÜKVS'ne ait elde etmiş oldukları uygulama sonuçları Tablo 3.2'de ve Tablo 3.3'de gösterilmektedir.

Tablo 3.2: Carter v.d. (1996) TÜKVS Kuşak Sayısı Minimizasyonu Uygulama Sonuçları.

		Sınav Çizelgeleme (Ceza ile)					**Sınav Çizelgeleme (Cezasız)**				
		LD	**SD**	**LWD**	**LE**	**RO**	**LD**	**SD**	**LWD**	**LE**	**RO**
CAR F–92	KS	31	**28**	30	31	32	31	**28**	30	31	32
	Bt	137	175	211	35	199	104	510	432	6	205
	Sn	618,3	510,8	911,6	121,2	1143,4	392,5	227,2	1775,6	19,1	996,6
CAR S–91	KS	32	**28**	30	32	35	32	**28**	30	32	35
	Bt	60	23	255	54	114	1	311	210	107	59
	Sn	239,8	75,1	1046,4	210,4	712,7	8,9	1080,6	807,9	412,1	265,2
EAR F–83	KS	**23**	24	**23**	**23**	24	**22**	**22**	**22**	**22**	**22**
	Bt	18	29	38	39	27	100	1	35	56	114
	Sn	80,2	173,0	156,6	151,8	256,5	391,6	8,7	111,5	184,5	630,4
HEC S–92	KS	**17**	**17**	18	**17**	**17**	18	**17**	**17**	**17**	**17**
	Bt	44	28	28	249	90	5	0	102	155	39
	Sn	31,8	17,1	22,8	219,5	83,1	3,2	0,5	66,8	105,0	30,8
KFU S–93	KS	20	**19**	**19**	**19**	**19**	**19**	**19**	**19**	20	**19**
	Bt	4	1	247	180	137	27	101	124	6	247
	Sn	120,1	97,2	2396,5	1852,7	2217,0	522,1	1159,6	1156,9	73,6	3361,6
LSE F–91	KS	**17**	**17**	**17**	**17**	18	**17**	**17**	**17**	**17**	**17**
	Bt	7	14	124	126	45	95	10	25	164	79
	Sn	78,7	164,6	763,9	786,2	674,5	551,9	78,0	158,8	943,3	848,1
PUR S–93	KS	43	**38**	40	40	42	36	**35**	38	38	-
	Bt	35	141	24	26	145	111	1	19	20	-
	Sn	31922,2	73338,3	8782,3	9304,9	72028,5	53899,0	1608,8	10885,7	8703,6	
RYE F–93	KS	23	**21**	22	22	22	**21**	**21**	22	22	22
	Bt	6	4	10	12	53	245	6	309	18	7
	Sn	507,2	380,0	419,4	502,5	3467,4	4809,0	343,8	6619,0	516,7	662,8
STA F–83	KS	**13**	**13**	**13**	**13**	**13**	**13**	**13**	**13**	**13**	**13**
	Bt	1	0	0	0	5	0	0	0	0	2
	Sn	6,0	6,0	5,7	6,1	98,9	2,7	2,7	2,7	2,8	90,3
TRE S–92	KS	22	**21**	**21**	23	22	22	**20**	**20**	22	22
	Bt	45	9	116	9	65	51	2	63	2	132
	Sn	398,8	108,4	1067,9	107,4	836,3	468,3	32,8	402,6	29,8	1342,5
UTA S–93	KS	**33**	35	35	34	35	33	**32**	33	33	34
	Bt	51	29	109	97	200	5	2	92	11	89
	Sn	4294,1	2557,2	9223,0	7340,4	17323,1	549,8	272,3	6878,4	755,6	6539,2
UTE S–92	KS	**10**	**10**	**10**	**10**	**10**	**10**	**10**	**10**	**10**	**10**
	Bt	3	0	5	7	8	0	0	2	0	4
	Sn	6,3	4,7	8,5	9,1	9,9	1,6	1,6	2,7	1,6	4,3
YOR F–83	KS	21	**19**	20	20	20	20	20	20	**19**	20
	Bt	81	60	18	166	72	255	1	378	310	208
	Sn	649,4	190,4	67,1	516,4	351,5	673,1	6,6	966,5	705,8	740,2

*** LD**: En Büyük Derece; **SD**: Doygunluk Derecesi; **LWD**: En Büyük Ağırlıklandırılmış Derece; **LE**: En Büyük Kayıt; **RO**: Rastgele Sıralama

Tabloda kısaltma olarak gösterilen KS ifadesi Kuşak Sayısını, Bt ifadesi *Backtracks*'i, Sn ifadesi Saniye cinsinden süreyi ve CP ifadesi Ceza Puanını ifade etmektedir. Ayrıca tablolarda her bir problem için elde edilen en iyi sonuçlar koyu renkle belirtilmiştir.

Tablo 3.3: Carter v.d. (1996) TÜKVS Ceza Puanı Minimizasyonu Uygulama Sonuçları.

		Sınav Çizelgeleme (Ceza ile)					**Sınav Çizelgeleme (Cezasız)**				
		LD	**SD**	**LWD**	**LE**	**RO**	**LD**	**SD**	**LWD**	**LE**	**RO**
CAR F–92	CP	7,6	6,6	6,6	**6,2**	8,2	0	0	0	0	0
	Bt	134	**0**	170	12	199	46	**0**	37	1	205
	Sn	637,9	**15,0**	886,0	47,0	1143,4	201,5	**5,4**	1625,2	7,7	996,6
CAR S–91	CP	7,9	**7,1**	7,4	7,6	11,9	0	0	0	0	0
	Bt	3	**0**	14	14	114	**0**	**0**	**0**	**0**	59
	Sn	29,2	**20,7**	67,5	58,3	712,7	**6,6**	**6,6**	**6,6**	**6,6**	265,2
EAR F–83	CP	**36,4**	46,5	37,3	42,3	42,1	0	0	0	0	0
	Bt	**3**	29	18	57	27	**0**	**0**	5	4	77
	Sn	24,7	**6,0**	76,1	242,4	256,5	24,7	**6,0**	21,2	18,6	492,5
HEC S–92	CP	**10,8**	12,7	15,8	15,9	20,0	0	0	0	0	0
	Bt	**8**	9	28	9	56	5	**0**	28	9	28
	Sn	7,4	**6,5**	22,8	8,7	54,9	3,2	**0,5**	17,2	5,6	23,2
KFU S–93	CP	**14,0**	15,9	22,1	20,8	21,0	0	0	0	0	0
	Bt	**4**	31	8	15	35	**2**	12	9	6	43
	Sn	**120,2**	559,4	147,3	208,9	977,6	**53,0**	171,4	95,8	73,6	909,7
LSE F–91	CP	12,0	12,9	13,1	**10,5**	13,3	0	0	0	0	0
	Bt	**2**	**2**	29	**2**	45	0	**0**	**0**	2	2
	Sn	50,3	50,5	233,5	**48,0**	674,5	14,9	**15,0**	**15,0**	22,8	311,7
PUR S–93	CP	4,4	4,1	5,0	**3,9**	-	0	0	0	0	0
	Bt	35	28	**17**	34	-	**0**	**0**	**0**	**0**	-
	Sn	31922,1	22921,9	**10601,1**	21729,4		1175,1	1178,2	**1172,1**	1175,3	
RYE F–93	CP	**7,3**	7,4	10,0	7,7	11,0	0	0	0	0	0
	Bt	6	**4**	8	8	50	97	**0**	309	18	7
	Sn	507,2	403,0	**360,0**	391,9	3019,2	3386,0	**68,1**	6619,0	516,7	662,8
STA F–83	CP	162,9	165,7	**161,5**	**161,5**	184,1	0	0	0	0	0
	Bt	1	**0**	**0**	**0**	5	**0**	**0**	**0**	**0**	2
	Sn	**6,0**	**6,0**	**5,7**	6,1	98,9	**2,7**	**2,7**	**2,7**	2,8	90,3
TRE S–92	CP	11,0	10,4	9,9	**9,6**	13,0	0	0	0	0	0
	Bt	13	**6**	7	9	27	51	**0**	2	2	132
	Sn	150,6	89,3	**74,1**	107,4	468,4	468,3	**13,5**	29,7	29,8	1342,5
UTA S–93	CP	4,5	**3,5**	5,3	4,3	6,4	0	0	0	0	0
	Bt	15	**5**	64	28	83	**0**	**0**	1	2	103
	Sn	1287,8	**664,3**	5929,7	2705,1	7135,6	**113,0**	113,2	197,8	231,0	8092,5
UTE S–92	CP	38,3	31,5	26,7	**25,8**	31,6	0	0	0	0	0
	Bt	3	**0**	5	7	8	**0**	**0**	2	**0**	4
	Sn	6,3	**4,7**	8,5	9,1	9,9	**1,6**	**1,6**	2,8	**1,6**	43
YOR F–83	CP	49,9	44,8	**41,7**	45,1	46,5	0	0	0	0	0
	Bt	262	142	78	58	**37**	255	**1**	378	46	208
	Sn	215,7	428,0	271,4	**174,5**	231,8	673,1	**66**	966,5	123,8	740,2

*** LD**: En Büyük Derece; **SD**: Doygunluk Derecesi; **LWD**: En Büyük Ağırlıklandırılmış Derece; **LE**: En Büyük Kayıt; **RO**: Rastgele Sıralama

Bu bölümde buraya kadar olan kısımda, TÜKVS'ne ait detaylar ve bu veri seti ile Carter ve arkadaşlarının elde etmiş oldukları ilk sonuçlar detaylı olarak incelenmiştir. Literatürde kıyaslama veri seti olarak kullanılan TÜKVS ile gerçekleştirilmiş olan çalışmalar da kronolojik olarak bu bölümde incelenecektir. Erişilebilmiş olan tüm çalışmalar aktarıldıktan sonra bölüm sonunda çalışmalarda elde edilmiş olan sonuçlar da karşılaştırmalı tablo olarak verilmiştir.

Burke ve arkadaşları 2002 (Burke v.d., 2002) yılında gerçekleştirdikleri çalışmalarında yeni bir yerel arama tekniği önerisinde bulunmuşlardır. Çalışmada, tavlama benzetimi gibi yerel arama tekniklerinin ortak dezavantajı olan parametrelerin belirlenmesinin zor olmasından ve bu nedenle de algoritmaların probleme has çözümler üretmesinden bahsedilmiştir. Çalışmada ayrıca, tepe tırmanma algoritması gibi tekniklerde ise parametre bulunmamasına karşın, elde edilen sonuçların kötü olması gibi bir sorun bulunduğuna da değinilmiştir. Önerilmekte olan yeni yerel arama tekniğinin, belirli arama süresine sahip tek parametreli bir çözüm tekniği olduğu aktarılmıştır. Bu arama zamanı parametresinin yüksek tutulduğu durumlarda çok iyi sonuçların elde edildiği ve kullanıcıların, bu arama zamanı parametresi değeri ile elde edilen sonuçlar arasındaki denge noktasını tespit etmesinin gerekliliği de çalışmada vurgulanmıştır.

2004 yılında, Asmuni ve arkadaşları (Asmuni v.d., 2004) bulanık sıralama ile lineer sıralama sezgisellerinin karşılaştırıldığı bir çalışma gerçekleştirmişlerdir. Elde ettikleri sonuçları değerlendiren yazarlar, bulanık sıralama ile daha iyi sonuçların elde edildiğinden bahsetmişlerdir.

Burke ve Newall (Burke ve Newall, 2004) aynı yıl gerçekleştirmiş oldukları çalışmalarında, sezgisel sıralama tekniklerinin grafik renklendirme algoritmalarına çok benzediğini ve sınav çizelgeleme problemlerine başarı ile uygulandığını belirtmişler. Çalışmalarında, probleme özgü adaptif metodun kullanıldığı uygun olmayan sezgiseller yerine uygun sezgiselin, test problemlerinin sınandığı esnada seçildiğini belirtmişlerdir. Çalışmalarında ayrıca, rasgele bir sezgiselle başlatılmış bir problem çözümünün adaptif bir şekilde uygun sezgiseli seçerek çözüme devam

ettiğini de dile getirmişler. Yazarlar çalışmada elde ettikleri sonuçları, daha önce gerçekleştirilmiş olan üç çalışmanın sonuçları ile karşılaştırmışlardır.

Aynı yıl, Burke ve arkadaşları (Burke v.d., 2004a) tabu aramanın da yer aldığı bir hiper sezgisel teknik önerisinde bulunmuşlardır. Çalışmada, çözüm uzayının ve arama uzayının araştırılmasında karşılaşılan sorunlara da değinilmiştir. Uygulama sonuçlarını değerlendiren yazarlar, düşük seviyeli sezgisellerin daha fazla sayıda kullanılması durumunda hiper sezgisellerin sonuçlarının daha verimli olduğundan bahsetmişlerdir. Aynı zamanda, diğer sezgisel tekniklerde de olduğu gibi arama uzayı boyutunun arttırılması durumunda, hesaplama zamanının da arttığı vurgulanmıştır.

Burke ve arkadaşları (Burke v.d., 2004b) 2004 yılında gerçekleştirdikleri diğer bir çalışmada zamanı önceden belirlenmiş tavlama benzetimini ve great deluga algoritmasını kullanmışlardır. Çalışmada kullanılmış olan katı ve hafif kısıtlardan net bir şekilde bahsedilmemiştir. Önerilmekte olan tekniklerle gerçekleştirilen problem çözüm zamanları karşılaştırıldığında great deluga algoritmasının çok daha kısa sürede sonuçlandığı vurgulanmıştır.

2006 yılında ise Abdullah ve Burke (Abdullah ve Burke, 2006) iki aşamalı bir yaklaşım geliştirmişlerdir. Geliştirmiş oldukları çözüm yönteminin ilk aşamasında çoklu başlangıçla grafik algoritması kullanan yazarlar, ikinci aşamada da great deluga ve benzetilmiş tavlama tekniklerini uygulayarak yerel arama gerçekleştirmişlerdir. Yazarlar çalışmanın sonunda, great deluga algoritmasının benzetilmiş tavlamadan daha iyi sonuçlar ürettiğini belirtmişlerdir.

2007 yılına gelindiğinde ise Abdullah ve arkadaşları (Abdullah v.d., 2007) Ahuja–Orlin'in geniş komşuluk arama yaklaşımını kullanarak çizelgeleme probleminin çözümünü gerçekleştirmişlerdir. Çalışmada gerçekleştirilen çizelgeleme çözümünü kapasite kısıtlı ve kapasite kısıtsız olarak iki şekilde değerlendirmişlerdir.

Asmuni ve arkadaşları (Asmuni v.d., 2007) beş farklı sıralama tekniğini de içeren bulanık çoklu sezgisel sıralama teknikleri önermişlerdir. Yazarlar, gerçekleştirmiş

oldukları uygulamaların sonuçlarını son bölümünde karşılaştırmalı sonuçlar olarak değerlendirmişlerdir.

Aynı yıl Burke ve arkadaşları (Burke v.d., 2007), grafik tabanlı hiper sezgisel ve çok aşamalı grafik tabanlı hiper sezgisel tekniklerini kullanarak uygulama gerçekleştirmişlerdir. Çalışmalarında, önerdikleri yöntemin üstünlüklerinden bahseden yazarlar, elde ettikleri sonuçları literatürde yer alan sonuçlarla karşılaştırmışlardır. Literatürde yer alan sonuçların bir tanesinden daha iyi sonuç elde etmişlerdir.

2008 yılında Pais ve Amaral (Pais ve Amaral, 2008) tabu arama tekniği kullanan bir çözüm önerisinde bulunmuşlardır. Çalışmada, tabu süresinin algoritma performansını etkkileyen önemli bir faktör olduğundan bahsedilmiştir. Ayrıca çalışmanın amacının, tabu arama tekniğindeki hafıza yönteminin ayarlanması olduğu belirtilmiştir. Yazarlar, geliştirmiş oldukları tabu arama tekniğinde bulanık kural tabanlı sistemden faydalanmışlardır.

Grafik tabanlı sezgisel sıralama tekniğinin kullanıldığı bir çalışma da 2009 yılında Asmuni ve arkadaşları (Asmuni v.d., 2009) tarafından gerçekleştirilmiştir. Yazarlar çalışmalarında, literatürde yer alan çalışmaların çoğunda çözümlerin iyileştirilmeye çalışıldığına ve buna karşın en iyi yaklaşımın elde edilmesi için harcanan çabanın çok daha az olduğuna da dikkat çekmişlerdir.

Aynı yıl Burke ve arkadaşları (Burke v.d., 2009) da açgözlü rastgele uyarlamalı arama prosedürü uygulaması gerçekleştirmişlerdir. Başlangıç çizelgelemenin oluşturulması için ise doygunluk derecesi ve en büyük ağırlıklandırılmış derece teknikleri kullanılmıştır. Yazarlar, çalışmalarında çözüm kalitesini arttırmak için kısıtlı aday listesi kullandıklarını belirtmişlerdir.

Abdullah ve arkadaşları (Abdullah v.d., 2010) 2010 yılında tabu tabanlı bir memetik yaklaşım önerisinde bulunmuşlardır. Çalışmada tabu listeleri, komşuluk yapılarının değerlendirilmesi için kullanılmış ve problemin çözümü genetik operatörler kullanılarak gerçekleştirilmiştir. Çözümlerde katı kısıt olarak yalnızca aynı öğrencinin aynı anda birden fazla sınavda bulunmaması değerlendirilmiştir. Ayrıca

hafif kısıtların iyileştirilmesi üzerinde de durulmuştur. Önerilen yöntemin test edilmesi sırasında 13 adet test probleminden 11'i çalıştırılmış ve dört tanesinde en iyi sonuçların elde edildiğine değinilmiştir.

Aynı yıl Abounacer ve arkadaşları (Abounacer v.d., 2010) da entegre bir karınca kolonisi algoritması geliştirmişlerdir. Çalışmada geliştirilen algoritmada, hafıza yapılarından da faydalanılmıştır. Önerilen algoritma ile en iyi sonuçlar elde edilmiş olmasa da literatürde yer alan en iyi sonuçlara yakın olarak bulunmuştur.

Burke ve arkadaşları (Burke v.d., 2010) değişken komşuluk arama ve genetik algoritma yöntemlerini birleştiren bir yaklaşım geliştirmişlerdir. Uygulamada elde edilen sonuçların, en iyi sonuçlara yakın sonuçlar olduğu görülmüştür. Çalışmada, katı kısıtları tamamen sağlayan ve hafif kısıtları iyileştiren bir algoritma ile çözüme gidilmiştir.

Chaudhari ve De (Chaudhari ve De, 2010) bulanık genetik algoritma yaklaşımı kullanarak geliştirdikleri algoritmayı gerçek hayat problemlerine de uygulayarak elde ettikleri sonuçları farklı genetik algoritma yaklaşımları ile karşılaştırmışlardır. Diğer çalışmalarda olduğu gibi, hem katı kısıtlar hem de hafif kısıtları içeren bir çözüm geliştirmişlerdir.

Fatima ve Bader–El–Den (Fatima ve Bader–El–Den, 2010) genetik programlama ve genetik algoritma yöntemlerini birleştiren bir çalışma gerçekleştirmişlerdir. Önerilen yöntemi veri setinde yer alan 10 adet probleme uygulamış ve elde ettikleri sonuçları diğer çalışmalardan elde edilen sonuçlarla karşılaştırmışlardır. Literatürde yer alan en iyi sonuçlara nazaran biraz daha kötü sonuçlar elde etmişlerdir.

Aynı yıl Hadjidj ve Drias (Hadjidj ve Drias, 2010) açgözlü rastgele uyarlamalı arama prosedürü ve evrimsel dağınık arama yaklaşımlarını da içeren bir hibrit evrimsel algoritma kullanarak çözümleri elde etmişlerdir. Uygulamalar sonucunda elde ettikleri değerler literatür sonuçlarıyla karşılaştırıldığında, genelde daha iyi sonuçlar olduğu görülmektedir.

Malik ve arkadaşları (Malik v.d., 2010) tabakalı örnekleme tekniğini iki aşamalı entegre çoklu komşuluk tabu arama yönteminde kullanmışlardır. Veri setine uyguladıkları yöntemin sonuçlarını, literatürde yer alan diğer çalışmaların sonuçlarıyla karşılaştırmışlar ve en iyi sonuçlardan daha iyi sonuçlar elde etmemelerine karşın yakın sonuçları sağlamışlardır.

Mumford (Mumford, 2010) çok amaçlı memetik algoritma ve açgözlü algoritmayı arka arkaya kullanarak sınav çizelgeleme problemlerinin çözümünü sağlamıştır. Çok amaçlı memetik algoritmayla katı kısıtları da sağlayan bir çizelgeleme gerçekleştirildikten sonra, hafif kısıtların iyileştirilmesi için açgözlü algoritmayı kullanmıştır. Çalışmada açgözlü algoritma ile hafif kısıtların iyileştirilmesi gerçekleştirilirken, katı kısıtların sağlanma durumunun bozulmamasına da özen gösterilmiştir. Çalışmada önerilen yaklaşım özellikle, ağır kısıtlara sahip sınav çizelgeleme problemlerinin çözümü için geliştirilmiştir. Veri setine uygulanan yöntemin sonuçları, en iyi sonuçlarla yarışır niteliktedir.

Pillay ve Banzhaf (Pillay ve Banzhaf, 2010) iki aşamalı bir genetik algoritma yaklaşımı geliştirmişlerdir. Önerilen yöntemin ilk aşamasında katı kısıtların kesinlikle sağlanması garantilendikten sonra, hafif kısıtların da sağlanmasına odaklanılarak ceza puanlarının azaltılması gerçekleştirilmiştir. Önerilen yöntem, veri setine uygulanmış ve elde edilen sonuçlar diğer makalelerin sonuçlarıyla karşılaştırılmıştır.

2011 yılında ise Alzaqebah ve Abdullah (Alzaqebah ve Abdullah, 2011a) yapay arı kolonisi algoritması kullanarak sınav çizelgeleme problemlerinin çözümünü gerçekleştirmişlerdir. Katı kısıtlar için literatürde sıklıkla kullanılan kısıtları tercih eden yazarlar ayrıca, özel sınav salonlarında gerçekleştirilmesi gereken sınavlar için bu özel salonlara atanmayı garantileyen bir kısıt da eklemişlerdir. Hafif kısıtlar için de öğrenci sayısı fazla olan derslerin sınavlarını sınav kuşağının sonlarına atamamak, bazı sınav kuşaklarına sınav atanmasını engellemek ve bazı sınav salonlarına sınav atanmasını engellemek şeklinde literatürde sıklıkla kullanılan kısıtlardan farklı kısıtları da dikkate almışlardır. Uygulamalarda literatürde yer alan en iyi sonuçlardan daha iyi sonuç elde edemeyen yazarlar, en iyi sonuca yakın sonuçlar elde etmişlerdir.

Aynı yıl Alzaqebah ve Abdullah (Alzaqebah ve Abdullah, 2011b) temel arı kolonisi algoritmasının yanında kendilerinin önerdikleri bir arı kolonisi algoritmasını da tanıtmışlardır. Diğer çalışmalarındaki kısıtları bu çalışmada da kullanan yazarlar, her iki algoritmadan da elde ettikleri sonuçları literatürde yer alan sonuçlarla karşılaştırmışlardır. Yazarlar, veri setinde yer alan problemlerin birçoğunda kendileri tarafından önerilen arı kolonisi algoritmasının, temel arı kolonisi algoritmasına göre daha iyi sonuçlar ürettiğini tespit etmişlerdir. Fakat buna karşın literatürde yer alan en iyi sonuçlardan daha iyi sonuçlar elde edememişlerdir.

Malik ve arkadaşları (Malik v.d., 2011) da aynı yıl sınav çizelgeleme problemlerinin çözümü için, iki aşamalı entegre çoklu komşuluk tabu arama tekniği önerisinde bulunmuşlardır. Literatürde yer alan ve kapasitesiz sınav çizelgeleme problemlerinin çözümünde dikkate alınan genel kısıtları dikkate alan yazarlar, başlangıç için tabakalı tesadüfi örnekleme tekniğinden faydalanmışlardır. Literatürde yer alan en iyi sonuçlardan daha iyi sonuç elde edemeyen yazarlar, problemlerin çoğunda en iyi sonuçlara % 10'un üzerinde bir oranda yaklaşmışlardır.

2011 yılında Turabieh ve Abdullah (Turabieh ve Abdullah, 2011) etkin sezgisel operatörleri great deluga algoritmasının içerisinde kullanarak bir melez yaklaşım önerisinde bulunmuşlardır. Önerilen yöntemi literatürde yer alan 13 probleme de uygulayan yazarlar, o ana kadar bilinen en iyi sonuçlardan iki tanesinden daha iyi sonuç elde etmişlerdir. Diğer problemlerde de mevcut sonuçlara yakın sonuçlar elde etmişlerdir. Önerilen yaklaşımla elde edilen sonuçların kalitesine vurgu yapan yazarlar, her bir üniversitenin kendine özgü kısıtlarının da kolayca algoritma içerisine eklenebileceğinden ve uyarlanabileceğinden de bahsetmişlerdir.

Burke ve arkadaşları (Burke v.d., 2012) 2012 yılına gelindiğinde sezgisel tekniklerin lineer kombinasyonlarını kullanarak yeni bir yöntem önerisinde bulunmuşlardır. Önerilen yöntemde, sezgisel teknikler ağırlıklandırılarak sıralama sürecine dahil edilmiştir. Veri setinde yer alan 13 adet problemden 12 adedine uygulanan yöntemle, o ana kadar elde edilmiş olan en iyi sonuçların beş adedinde daha iyi sonuçlar bulunmuştur. Elde edilen diğer sonuçlar da literatürde yer alan sonuçlara çok yakın ve bu sonuçlarla yarışır niteliktedir.

Aynı yıl Demeester ve arkadaşları (Demeester v.d., 2012) çizelgeleme problemlerinin çözümü için turnuva tabanlı bir hiper sezgisel önerisinde bulunmuşlardır. Veri setinde yer alan problemlerin tamamına uyguladıkları çözüm yöntemiyle birçok problemde en iyi sonuçları elde eden yazarlar, diğer problemlerde de en iyi sonuca yakın sonuçlar elde etmişlerdir. Sonuçlar bir bütün olarak değerlendirilirse, bu kadar fazla sayıda problemde iyileştirmeyi aynı anda sağlayan bir yöntem olması dolayısıyla önerilen yöntem literatürde yer alan önemli bir yaklaşımdır.

Li ve Qu 2012 yılında (Li ve Qu, 2012) evrimsel kalıntı ve stokastik yeniden oluşturma adında yeni bir akıllı sistem sınıfı önerisinde bulunmuşlardır. Yazarlar tarafından önerilen yöntem, değişen koşulları öğrenen ve uyum sağlayabilen bir sistemdir. Klasik kalıntı ve yeniden oluşturma sürecini geliştirerek stokastik yeniden oluşturma adımını ekleyen yazarlar, 13 adet problemden 11 adedini çözmüşlerdir. Çözülmüş olan problemlerde mevcut en iyi sonuçlar geliştirilememiştir.

Pais ve Amaral (Pais ve Amaral, 2012) çizelgeleme problemlerinin çözümü için tabu arama tekniği önerisinde bulunmuşlardır. Önerilen yöntemde, bulanık çıkarım sistemi kullanılarak tabu süresinin belirlenmesi üzerinde durulmuştur. Yazarlar, tabu arama tekniğinde doğru seçilmiş bir tabu süresinin algoritma performansını etkilediğine vurgu yapmaktadırlar. Veri setinde yer alan 13 adet problemden 11 adedini çözmüş olan yazarlar, mevcut en iyi sonuçlardan daha iyi sonuçlar elde edememişler, fakat birçoğunda mevcut en iyi çözüme % 10'un altında bir oranda yaklaşmışlardır.

Aynı yıl Pillay (Pillay, 2012), kapasite kısıtsız sınav çizelgeleme problemlerinin çözümü için evrimsel hiper sezgisel önerisinde bulunmuştur. Yazar, sezgisel kombinasyonların üç farklı temsillerini uygulamıştır. Önerilmiş olan hiper sezgisel tekniklerden elde edilen sonuçlar, literatürde yer alan mevcut diğer hiper sezgisel tekniklerin sonuçları ile ayrıntılı olarak karşılaştırılmıştır. Çalışmada önerilmekte olan hiper sezgisel yöntemlerle, literatürde yer alan diğer hiper sezgisel tekniklerden elde edilen sonuçların birçoğu iyileştirilmiştir.

Sabar ve arkadaşları (Sabar v.d., 2012a) çizelgeleme problemlerinin çözümü için bir bal arısı çiftleşme optimizasyonu algoritması önerisinde bulunmuşlardır. Önerilen yöntem, gerçek bal arılarının çiftleşme sürecinden uyarlanarak geliştirilmiştir. Veri setinde yer alan 13 adet problemin tamamına uygulanan yöntemle birlikte, o ana kadar elde edilmiş en iyi sonuçların çoğunda iyileştirme gerçekleştirilmiştir. Önerilmekte olan yöntem, mevcut çizelgeleme teknikleriyle yarışır nitelikte sonuçlar üretmiştir.

Sabar ve arkadaşları (Sabar v.d., 2012b) aynı yıl, bir hiper sezgisel teknik önerisinde de bulunmuşlardır. Önerilen yöntem, sezgisellerin dört aşamalı bir hiyerarşik melezlemesinden faydalanmaktadır. Çalışmada, bir sınav için uygun olan sınav kuşağının olasılıksal seçimini gerçekleştirmek için algoritma sürecine, rulet tekerleği seçim tekniği dahil edilmiştir. Uygulanmış olan 13 adet problemden yalnızca iki adedinde, bilinen en iyi sonucun iyileştirildiği tablolaştırılarak gösterilmiştir.

2013 yılında kendiliğinden uyarlamalı hibrit bir arılar algoritması Abdullah ve Alzaqebah (Abdullah ve Alzaqebah, 2013) tarafından geliştirilmiştir. Önerilen algoritmada yazarlar, seçim için ceza puanlarını kullanmak yerine yıkıcı seçim, turnuva seçimi ve sıralayarak seçim stratejilerini kullanmayı tercih etmişlerdir. Bu üç seçim stratejilerini tercih etmelerinin nedeni olarak da popülasyon çeşitliliğini arttırmak istediklerini belirtmişlerdir. Buna ek olarak yerel aramada yoğunlaşma yeteneğinin de geliştirilmesi için komşuluk aramayı yönlendirme yeteneğine sahip, kendiliğinden uyarlamalı bir strateji kullanmışlardır.

Aynı yıl Abed ve Tang (Abed ve Tang, 2013), yerel arama için geliştirilmiş olan *Record–to–Record Travel (RRT)* Algoritması ile genetik algoritma tekniklerini birleştiren bir yöntem önerisinde bulunmuşlardır. Önerilmiş olan algoritma ile hedeflenen, genetik algoritma global aramaları ve RRT ile yerel aramaları gerçekleştirirken bir denge yakalamaktır. RRT kullanılmadan yalnızca genetik algoritma kullanılarak elde edilen sonuçlarla, iki yöntemin birlikte çalıştırılması neticesinde elde edilen sonuçlar karşılaştırıldığında, iki yöntemin birlikte çalıştırılması durumunda daha iyi sonuçlar elde edildiği gözlemlenmiştir. Çalışmalarında hibrit bir yöntem önerisinde bulunan yazarlar, metasezgisellerin

özellikle diğer tekniklerle hibrit bir şekilde kullanıldığı durumlarda daha iyi sonuçlar elde ettiğini de vurgulamaktadırlar.

Sınav çizelgelemenin deterministik optimizasyonu için alan dönüşüm yaklaşımı 2013 yılında Abdul Rahim ve arkadaşları (Abdul Rahim v.d., 2013a) tarafından önerilmiştir. Önerilmekte olan yöntem TÜKVS, NÜKVS ve ITC2007 veri setine uygulanmıştır. Elde edilen sonuçlar literatürde yer alan diğer çalışmalarla da karşılaştırılarak yöntemin performansı değerlendirilmiştir.

Rajah ve Pillay (Rajah ve Pillay, 2013) biyolojiden esinlenerek, sınav çizelgeleme problemlerinin çözümü için gelişimsel yaklaşım önerisinde bulunmuşlardır. Yazarlar tarafından önerilmekte olan gelişim yaklaşımına, hücre tüketim operatörü de dahil edilmiş ve TÜKVS'ne uygulanmıştır. Uygulamayla elde edilen sonuçlar, hem klasik gelişim yaklaşımıyla hem de biyolojiden esinlenilerek geliştirilmiş diğer yaklaşımlarla karşılaştırılmıştır. Önerilmekte olan yaklaşımla elde edilen sonuçlar, klasik gelişim yaklaşımıyla elde edilen sonuçlara göre daha iyi olmakla beraber, biyolojiden esinlenilerek geliştirilmiş diğer yaklaşımlarla elde edilen sonuçlarla da yarışır niteliktedir.

Sabar ve arkadaşları (Sabar v.d., 2013) tarafından kombinatoryal optimizasyon problemlerinin çözümü için dil bilgisel evrim hiper–sezgiseli de 2013 yılında geliştirilmiştir. Çalışmada önerilmekte olan dil bilgisel evrim yaklaşımı, farklı sezgisel bileşenlerden oluşan çevrimiçi bir çözücüyü güçlendirici olarak kullanılmaktadır. Yöntem, sınav çizelgeleme problemleri için TÜKVS ve ITC2007 veri setine ve kapasite kısıtlı araç rotalama problemlerine uygulanarak test edilmiştir.

2014 yılına gelindiğinde ise memetik algoritmaların ana bileşenlerinin kombinasyonlarının etkinliğini araştıran bir çalışma, Al–Betar ve arkadaşları (Al–Betar v.d., 2014) tarafından gerçekleştirilmiştir. Memetik algoritma bileşenlerinin kombinasyonları 17 farklı senaryo ile uygulanmıştır. Elde edilen tüm sonuçlar kullanılan kombinasyonlarla birlikte ayrıntılı olarak verilmiştir. TÜKVS'nde yer alan 13 adet problemden 12'sinin uygulandığı çalışmada, iki adet problem için bilinen en iyi sonuçlardan daha iyi sonuçlar elde edilmiştir.

Burke ve arkadaşları (Burke v.d., 2014) sınav çizelgelemenin geliştirilmesi için sezgisel tekniklerin uyarlamalı seçimini içeren bir yaklaşım önerisinde bulunmuşlardır. Önerilen yaklaşımın ilk aşamasında sezgisel diziler rastgele olarak seçilirler. İkinci aşamada da sezgiseller, en iyi sezgisel dizisini bulmak için boş pozisyonlara atanır. Son olarak, oluşturulmuş olan diziler problemlere uygulanırlar. Önerilmiş olan uyarlamalı iyileştirme hiper–sezgiseli TÜKVS için o ana kadar elde edilmiş en iyi sonuçlardan daha iyi sonuçlar elde edememiş olsa da, problemlerin çoğunda mevcut hiper–sezgisellerden daha iyi sonuçlar elde etmiştir.

Fong ve arkadaşları (Fong v.d., 2014) üniversite çizelgeleme problemlerinin çözümü için 2014 yılında sürü tabanlı hibrit bir emperyalist optimizasyon algoritması önerisinde bulunmuşlardır. Önerilmiş olan hibrit yaklaşım, *great deluga* ve yapay arı kolonisi algoritmalarının birleştirilmesiyle oluşturulmuştur. Ayrıca çalışmada, great deluga algoritmasının içine, Nelder–Mead simpleks arama tekniği de dahil edilmiştir. TÜKVS'nde yer alan 13 adet problemden 11'ine uygulanan yaklaşımla, 3 problemde en iyi sonuçlar geliştirilmiştir.

Kolonias ve arkadaşları da (Kolonias v.d., 2014) genetik algoritma ve açgözlü dik iniş bileşenlerini içeren bir hibrit evrimsel algoritma yaklaşımı geliştirmişlerdir. Önerilen algoritma yaklaşımı Grafik İşlemci Birimleri (GPUs) kullanılarak gerçekleştirilmiştir. GPU'nun hesaplama kapasitesi sayesinde çözümlerde, çok büyük boyutlarda popülasyonların kullanımı sağlanabilmektedir. GPU uygulaması sayesinde de bu boyuttaki problemler, yirmi altı kata kadar daha hızlı sürede çözümler üretmişlerdir. TÜKVS'inde yer alan 13 adet problemin tamamına uygulanan yaklaşımla birlikte, problemlerden dört adedinin en iyi sonuçları geliştirilmiştir.

Sınav çizelgeleme problemlerinin çözümü için geliştirilmiş çift evrimsel havuz memetik algoritması da Lei ve arkadaşları (Lei v.d., 2014) tarafından önerilmiştir. Önerilmiş algoritmada kullanılan çift evrimsel havuz yapısı, algoritmanın performansını geliştirmek için tercih edilmiştir. Kullanılan evrimsel havuz yapılarının her biri kendi çaprazlama, mutasyon ve seçim mekanizmalarına sahiptir. Ana evrimsel havuz yapısı, popülasyonun büyük bir kısmını oluşturmaktadır ve bu

yapıyla sağlanmak istenen amaç yeniden sıralama stratejisine dayalı bir yerel arama operatörüyle daha elverişli sınav çizelgesi oluşturmaktır. Daha sonra, ana evrimsel havuz yapısı ikincil evrimsel havuz yapısının içine aktarılarak burada çalıştırılmaktadır. Yazarlar, önermiş oldukları algoritmayı TÜKVS'nde yer alan 12 adet probleme uygulayarak test etmişlerdir.

Aynı yıl, uyarlamalı ayrıştırma ve sıralama stratejilerini içeren bir çalışma, Abdul Rahman ve arkadaşları (Abdul Rahman v.d., 2014a) tarafından yayınlanmıştır. Farklı sıralama ve karıştırma stratejileri içeren yaklaşım TÜKVS'nde bulunan 13 probleme de uygulanarak test edilmiştir. Elde edilen sonuçlar, o zamana kadar elde edilmiş en iyi sonuçlardan daha iyi olmasa da bu sonuçlarla yarışır niteliktedir.

Abdul Rahman ve arkadaşlarının (Abdul Rahman v.d., 2014b) 2014 yılında sınav çizelgesi oluşturmak için gerçekleştirdiği bir diğer çalışma da sezgisellerin uyarlamalı doğrusal kombinasyonları üzerinedir. Belirli bir problem veri setinin genelleştirilmesi adına her bir parametre için bir normalleşme stratejisine başvurulmaktadır. En büyük derece ve doygunluk derecesi çalışmada kullanılan iki tekniktir. Önerilmiş olan algoritma TÜKVS ve ITC2007 veri setine uygulanmış ve sonuçlar ayrıntılı olarak incelenmiştir.

2015 yılına gelindiğinde, hibrit bir arı kolonisi optimizasyonu Alzaqebah ve Abdullah (Alzaqebah ve Abdullah, 2015) tarafından önerilmiştir. Yazarlar çalışmalarında, komşuluk aramalarını geliştirmek için, bir komşuluk yapısının seçimi için kendiliğinden uyarlamalı bir mekanizma tanıtmışlardır. Ayrıca, önerilmekte olan algoritma içerisinde gerçekleştirilen yerel aramalar tepe tırmanma algoritması ve tavlama benzetimi algoritmasıyla desteklenmişlerdir. Önerilmekte olan algoritma, TÜKVS ve ITC2007 veri setine uygulanmış ve son zamanlarda yapılmış olan çalışmaların sonuçlarıyla da karşılaştırılarak ayrıntılı bir biçimde aktarılmıştır.

Aynı yıl, evrimsel kalıntı ve stokastik yeniden yapılandırma kullanılarak gerçekleştirilen bir çalışma da Li ve arkadaşları (Li v.d., 2015) tarafından gerçekleştirilmiştir. Önerilen algoritmada bulunan stokastik yeniden yapılandırma, arama süreci boyunca değişen çevre koşulları için öğrenen ve uyum sağlayabilen

süreçtir. Evrimsel kalıntı aşaması ise seçim ve mutasyon operatörleri kullanılarak geliştirilmiştir.

Qu ve arkadaşları (Qu v.d., 2015) tarafından 2015 yılında hibrit bir hiper–sezgisel yaklaşım önerisinde bulunulmuştur. Önerilmiş olan algoritmayla amaçlanan, optimizasyon problemlerinin kabul edilebilir bir kalitede çözümlerini üretmektir. Ayrıca, algoritmada üst seviyede bulunan arama işlemleri basit bir tahmin dağılım algoritmasına dayanmaktadır.

2015 yılında gerçekleştirilen bir diğer çalışma Lei ve arkadaşları (Lei v.d., 2015) tarafından hiper–sezgisel tabanlı memetik algoritma üzerine yapılmıştır. Çalışmada, beş farklı grafik sezgisellerinin kombinasyonları kullanılmaktadır. Ayrıca, uygun çözümlerin sayısının arttırılması için de mutasyon ve çaprazlama operatörleri algoritmaya dahil edilmiştir.

Mandal ve Kahar (Mandal ve Kahar, 2015), sınav çizelgeleme problemlerinin çözümü için great deluga algoritması kullanarak kısmi sınav atamaları gerçekleştirmişlerdir. Önerilmiş olan algoritma, iyi kalitede bir sınav çizelgesi oluşturmak amacıyla, iyileştirme mekanizması olarak tasarlanmıştır. Çözüm sürecinde, belirli sınavlar parçalı olarak seçilir ve sıralanır. Bu sıralama işleminden sonra ise great deluga algoritması kullanılarak çizelge iyileştirilir. Süreç bu şekilde tüm sınavların ataması gerçekleştirilene kadar da devam ettirilir.

Aynı yıl Wong ve Assal (Wong ve Assal, 2015), çoklu yerel arama operatörlerini kullanarak çoklu komşuluk yapılarını da içeren bir çözüm yaklaşımı geliştirmişlerdir. Önerilen yaklaşımın ayrıntılı olarak aktarıldığı çalışmada, TÜKVS'nde yer alan test problemlerinin tamamının değil yalnızca dört tanesinin çözümü gerçekleştirilmiştir. TÜKVS için en iyi sonuçlardan daha iyi sonuçların elde edilmediği çalışmada, PUR–S–93 probleminin çözümü de gerçekleştirilmiştir. Elde edilen çözümlere ulaşılan çalışma zamanları da makalede verilmiş olup, PUR–S–93 probleminin çözümünün elde edilmesi için geçen zamanın 9 saat olması ve bilinen en iyi sonuca çok yakın bir sonuca ulaşılmış olması da önemli noktalardan biridir. Makalede çözülmüş olan diğer problemler de NÜKVS ve Melbourne Üniversitesi Kıyaslama Veri Setidir.

2016 yılına gelindiğinde Leite ve arkadaşları (Leite v.d., 2016), öğrencilerin sınavlarındaki çakışmaları minimize edebilmek adına, kurbağa sıçrama algoritması tabanlı hibrit bir sezgisel teknik önerisinde bulunmuşlardır. Önerilmekte olan sezgisel teknikte, üç ana operatör kullanılarak her bir iterasyonda kurbağalar geliştirilmeye çalışılmıştır. Çaprazlama ve mutasyon operatörlerinin yanı sıra benzetilmiş tavlama tabanlı bir yerel arama operatörü de algoritmaya dahil edilmiştir. Önerilmiş olan sınav çizelgeleme yaklaşımı TÜKVS'nde yer alan 13 adet problemin tamamına uygulanmıştır. Çalışmada elde edilmiş olan sonuçların, bilinen en iyi sonuçlara çok yakın olmasının yanı sıra, PUR–S–93 problemi için bilinen en iyi sonuç geliştirilerek 4,47 değerine de ulaşılmıştır.

Tablo 3.4: Literatürde Yer Alan TÜKVS Sonuçları–I.

Problem Kodu	Burke v.d. 2002	Asmuni v.d. 2004	Burke ve Newall 2004	Burke v.d. 2004a	Burke v.d. 2004b	Abdullah ve Burke 2006	Abdullah v.d. 2007	Asmuni v.d. 2007	Burke v.d. 2007	Pais ve Amaral 2008	Asmuni v.d. 2009
CAR–F–92	4,2	4,45	4,32	4,84	4,2	4,1	4,4	4,32	4,79	4,57	4,54
CAR–S–91	4,8	5,24	4,97	5,41	4,8	4,8	5,2	5,19	5,41	5,46	5,29
EAR–F–83	35,4	36,99	36,16	38,19	35,0	36,0	34,9	36,16	38,19	33,50	37,02
HEC–S–92	10,8	11,71	11,61	12,72	10,6	10,8	10,3	11,60	12,55	10,52	11,78
KFU–S–93	13,7	15,34	15,02	15,76	13,7	15,2	13,5	15,03	15,76	14,05	15,80
LSE–F–91	10,4	11,43	10,96	13,15	10,4	11,9	10,2	11,35	13,11	-	12,09
PUR–S–93	4,8	-	-	-	4,8	-	-	-	-	-	-
RYE–F–92	8,9	10,30	-	-	8,9	-	8,7	9,75	-	-	10,38
STA–F–83	159,1	159,15	161,91	141,08	159,0	159,0	159,2	158,64	141,08	157,29	160,42
TRE–S–92	8,3	8,64	8,38	8,85	8,3	8,5	8,4	8,47	8,85	8,71	8,67
UTA–S–92	3,4	3,55	3,36	3,54	3,4	3,6	3,6	3,52	3,54	3,71	3,57
UTE–S–92	25,7	27,64	27,41	32,01	25,7	26,0	26,0	27,55	31,65	25,18	28,07
YOR–F–83	36,7	40,68	40,77	40,13	36,7	36,2	36,2	39,25	40,13	39,08	39,80
TOPLAM	326,2	335,12	324,87	315,68	325,5	316,1	320,6	330,83	315,06	302,07	337,43

Tablo 3.5: Literatürde Yer Alan TÜKVS Sonuçları–II.

Problem Kodu	Burke v.d. 2009	Abdullah v.d. 2010	Abounacer v.d. 2010	Burke v.d. 2010	Chaudhari ve De 2010	Fatima ve Bader-El-Den 2010	Hadjidj ve Drias 2010	Malik v.d. 2010	Mumford 2010	Pillay ve Banzhaf 2010
CAR–F–92	4,45	3,82	5,9	3,9	4,27	5,10	-	5,24	4,5	4,22
CAR–S–91	5,37	**4,35**	6,9	4,6	4,95	6,71	-	5,97	5,3	4,92
EAR–F–83	36,52	33,76	42,4	32,8	36,78	41,2	34,00	40,12	34,2	35,87
HEC–S–92	11,78	10,29	11,0	**10,0**	11,85	12,68	10,34	11,48	10,3	11,5
KFU–S–93	15,45	**12,86**	17,3	13,0	14,50	15,41	-	15,38	13,9	14,37
LSE–F–91	12,12	10,23	14,9	10,0	11,14	14,04	-	11,07	11,3	10,89
PUR–S–93	-	-	-	-	4,72	-	-	-	-	4,65
RYE–F–92	-	-	-	-	9,65	-	-	15,24	9,1	9,3
STA–F–83	158,94	155,98	155,7	-	158,30	156,61	**136,43**	157,61	157,0	157,81
TRE–S–92	8,99	8,21	9,9	7,9	8,37	9,32	-	9,41	8,5	8,38
UTA–S–92	3,50	3,22	4,5	3,2	3,35	3,82	3,50	4,08	3,7	3,35
UTE–S–92	26,62	25,41	32,0	24,8	28,86	-	**18,42**	26,37	25,2	27,24
YOR–F–83	42,19	36,35	44,1	34,9	40,72	45,60	-	40,71	37,2	39,33
TOPLAM	325,93	304,48	344,6	145,1	337,46	310,49	202,69	342,68	320,2	331,83

Tablo 3.6: Literatürde Yer Alan TÜKVS Sonuçları–III.

Problem Kodu	Alzaqebah ve Abdullah 2011a	Alzaqebah ve Abdullah 2011b	Malik v.d. 2011	Turabieh ve Abdullah 2011	Burke v.d. 2012	Demeester v.d. 2012	Li ve Ou 2012	Pais ve Amaral 2012	Pillay 2012	Sabar v.d. 2012a
CAR–F–92	4,92	4,76	4,83	4,10	4,22	3,78	4,7	4,57	4,18	3,9
CAR–S–91	5,86	5,67	5,22	4,80	5,03	4,52	5,3	5,46	4,93	4,79
EAR–F–83	38,34	37,67	39,34	34,92	36,06	32,49	**29,7**	33,50	36,64	34,69
HEC–S–92	11,51	11,33	11,18	10,73	11,71	10,03	10,1	10,52	11,21	10,66
KFU–S–93	16,04	15,23	15,24	13,00	16,02	12,90	13,7	14,05	14,12	13,0
LSE–F–91	12,42	12,66	12,64	10,01	11,15	10,04	10,2	**9,11**	10,76	10,0
PUR–S–93	7,6	-	-	4,73	-	5,67	-	-	-	4,76
RYE–F–92	10,73	-	9,74	9,65	9,42	**8,05**	-	-	9,23	10,97
STA–F–83	158,01	157,59	157,12	158,26	158,86	157,03	157,3	157,29	157,39	157,04
TRE–S–92	9,58	9,0	9,07	7,88	8,37	**7,69**	8,4	8,71	8,37	7,87
UTA–S–92	3,99	3,88	4,48	3,20	3,37	3,13	3,3	3,71	3,32	**3,1**
UTE–S–92	27,80	27,58	24,98	26,11	27,99	24,77	25,3	25,18	26,95	25,94
YOR–F–83	41,44	39,28	39,31	36,22	39,53	**34,64**	37,8	39,08	39,63	36,15
TOPLAM	348,24	324,65	333,15	323,61	331,73	314,74	305,8	311,18	326,73	322,87

Tablo 3.7: Literatürde Yer Alan TÜKVS Sonuçları–IV.

Problem Kodu	Sabar v.d. 2012b	Abdullah ve Alzaqqebah 2013	Abed ve Tang 2013	Abdul Rahim v.d. 2013	Rajah ve Pillay 2013	Sabar v.d. 2013	Al-Betar v.d. 2014	Burke v.d. 2014	Fong v.d. 2014	Kolonias v.d. 2014	Lei v.d. 2014
CAR–F–92	4,7	3,94	4,54	4,49	4,04	4,0	4,29	4,31	**3,77**	4,47	5,2
CAR–S–91	5,14	4,76	5,38	5,19	4,85	4,62	4,99	5,19	4,71	5,24	6,4
EAR–F–83	37,86	33,61	36,27	37,57	34,85	34,71	34,42	35,79	33,15	34,41	39,8
HEC–S–92	11,90	10,56	10,73	11,47	10,78	10,68	10,4	11,19	10,38	10,39	11,8
KFU–S–93	15,30	13,44	15,17	14,36	13,84	13,0	13,5	14,51	13,69	13,77	16,2
LSE–F–91	12,33	10,87	11,87	11,90	10,48	10,11	10,48	10,92	10,25	11,06	14,5
PUR–S–93	5,37	-	-	4,88	-	4,8	-	-	-	5,25	-
RYE–F–92	10,71	8,81	8,6	9,80	9,06	10,79	8,79	-	-	8,61	12,3
STA–F–83	160,12	157,09	158,16	158,25	157,13	158,02	157,04	157,18	157,03	157,05	157,2
TRE–S–92	8,32	7,94	8,86	8,74	8,27	7,90	8,16	8,49	7,84	8,51	9,5
UTA–S–92	3,88	3,27	3,59	3,59	3,29	3,12	3,43	3,44	**3,10**	3,63	4,3
UTE–S–92	32,67	25,36	25,34	27,37	26,03	26,0	25,09	26,70	25,32	24,87	28,6
YOR–F–83	40,53	35,74	40,26	41,10	37,38	36,2	35,86	39,47	36,06	37,15	40,5
TOPLAM	348,83	315,39	328,77	338,71	320,0	323,95	316,45	317,19	305,3	324,41	346,3

Tablo 3.8: Literatürde Yer Alan TÜKVS Sonuçları–V.

Problem Kodu	Abdul Rahman v.d. 2014a	Abdul Rahman v.d. 2014b	Alzaqebah ve Abdullah 2015	Li v.d. 2015	Qu v.d. 2015	Lei v.d. 2015	Mandal ve Kahar 2015	Wong ve Assal 2015	Leite v.d. 2016
CAR–F–92	4,74	4,41	3,88	4,91	4,09	4,53	3,82	-	3,86
CAR–S–91	5,17	5,12	4,38	5,81	4,95	5,03	4,58	-	4,59
EAR–F–83	40,91	36,91	33,34	35,08	34,97	37,42	33,23	-	32,72
HEC–S–92	12,26	11,31	10,39	10,44	11,11	11,53	10,32	-	10,08
KFU–S–93	15,85	14,75	13,23	15,08	14,09	14,93	13,34	13,0	12,87
LSE–F–91	12,58	11,41	10,52	11,28	10,71	11,23	10,24	-	9,85
PUR–S–93	5,87	5,87	-	-	4,73	-	-	4,7	**4,47**
RYE–F–92	10,11	9,61	8,92	-	9,2	-	9,79	-	8,00
STA–F–83	158,12	157,52	157,06	157,06	157,64	157,91	157,12	-	157,03
TRE–S–92	9,30	8,76	7,89	8,39	8,27	8,52	7,84	8,1	7,78
UTA–S–92	3,65	3,54	3,13	-	3,33	3,75	3,13	-	3,15
UTE–S–92	27,71	26,25	25,12	25,17	26,18	26,3	25,28	-	24,76
YOR–F–83	43,98	39,67	35,49	36,97	37,88	41,04	35,46	36,2	34,85
TOPLAM	350,25	335,13	313,35	310,19	327,15	322,19	314,15		314,01

3.2.2. Nottingham Üniversitesi Kıyaslama Veri Seti Kullanılan Çalışmalar

ÜSÇP'nde kullanılan diğer bir test kıyaslama veri seti de Nottingham Üniversitesi Kıyaslama Veri Setidir (NÜKVS). NÜKVS ilk olarak 1996 yılında Burke ve arkadaşları (Burke v.d., 1996) tarafından yapılmış bir çalışma ile tanıtılmıştır. Tanıtılmış olan veri seti Nottingham Üniversitesinin 1994/1995 yılı verilerinden elde edilmiştir. NÜKVS'ne ait detaylı bilgiler aşağıda yer alan tabloda verilmektedir.

Tablo 3.9: Nottingham Üniversitesi Kıyaslama Veri Seti (Burke v.d., 1996).

Sınav Sayısı	800
Öğrenci Sayısı	7896
Öğrenci Sınavları Sayısı	34265
Çakışma Matrisinin Yoğunluğu	0,03
Sınav Kuşağı Sayısı	23
Kapasite	1550

NÜKVS'nin TÜKVS'nden önemli farkı, kapasite kısıtlı bir problem olmasıdır. TÜKVS'nde çizelgeleme işlemi gerçekleştirilirken, kapasite kısıtları bulunmadığından dolayı, sınavlar kuşaklara atanırken herhangi bir sınırlama olmaksızın yerleştirilebilmekteydi. Bir diğer ifadeyle, bir kuşağa toplam sınav sayısının % 20'si yerleştirilebilirken, diğer bir kuşağa da toplam sınav sayısının yalnızca % 2'si yerleştirilebilmektedir. Fakat NÜKVS'nde kapasite kısıtı bulunduğundan dolayı, bir sınav kuşağına sınırlı sayıda sınav yerleştirilebilmektedir. Bu durum dolayısıyla da bir kuşağa yerleştirilebilecek çok çeşitli sınav kombinasyonları oluşturulabilecektir. Bu kombinasyonların incelenebilmesi ve optimum sınav çizelgelemesinin gerçekleştirilebilmesi için, çizelgeleme işlemleri sırasında sınavların yer değişimleri önemli bir çözüm adımı olabilmektedir.

Bu tez çalışmasında odaklanılan çözüm TÜKVS olduğundan dolayı ağırlıklı olarak TÜKVS'ne ait literatür taramasına yer verilmiştir. Çizelgeleme problemlerinin çözümü için geliştirilmiş olan farklı tekniklerin de incelenmesi adına çalışmanın bu kısmında da NÜKVS'ne ait erişilebilmiş olan literatür taramasının bir kısmına yer verilmeye çalışılmıştır.

NÜKVS ilk olarak 1996 yılında Burke ve arkadaşları (Burke v.d., 1996) tarafından geliştirilmiş olan bir memetik algoritma yaklaşımıyla birlikte tanıtılmıştır. Çalışmada, sınav çizelgeleme problemlerinin çözümünde dikkate alınan ve özellikle kapasite kısıtlı problemlerde daha yoğun olarak kullanılan aşağıda yer alan 3 farklı kısıta da değinilmiştir.

- A sınavı, B sınavından önce çizelgelenmelidir.
- A sınavı ile B sınavı aynı anda çizelgelenmelidir.
- Bazı sınavlar belirli bir odada gerçekleştirilmelidir.

Kullanılan tekniklerin ayrıntılı olarak aktarıldığı çalışmada, tepe tırmanma operatörü de kullanılarak yerel optimum değerlere takılmanın engellenmesi gerçekleştirilmeye çalışılmıştır. NÜKVS, çalışmada 23, 26, 30, 31 ve 32 adet sınav kuşağı sayılarıyla çözülmüş ve sonuçlar listelenmiştir. Elde edilen sonuçlar aşağıdaki şekildedir.

Tablo 3.10: Burke v.d. (1996) NÜKVS Uygulama Sonuçları.

Sınav Kuşağı Sayısı	Değerlendirmeler	Çizelgelenmeyen	İhlaller
23	9950	0	348
26	15500	0	49
30	12600	0	0
31	5450	0	0
32	1850	0	0

Gerçekleştirilmiş olan çalışmada üç ana kısıt üzerinde durulmuştur; hiçbir öğrencinin bir sınav kuşağında iki adet sınavı olamaz, maksimum 1550 olan kapasite kısıtı aşılamaz ve bir öğrencinin aynı günde iki sınava girmesi gerekirse bu iki sınavı arasında tam bir kuşak olmalıdır. Bu üç kısıttan ilk iki kısıt zaten ihlal edilmesi mümkün olmayan kısıtlardır. İhlal edilebilir olan tek kısıt üçüncü kısıttır ve tabloda belirtilen ihlal değerleri de yalnızca bu üçüncü kısıtın ihlalinden elde değerleri belirtmektedir.

Bu çalışmadan sonra NÜKVS ile gerçekleştirilen çalışmalar, 23 adet sınav kuşağı ile gerçekleştirilerek sonuçların karşılaştırılabilirliği sağlanmıştır.

Burke ve Newall 2004 yılında (Burke ve Newall, 2004) gerçekleştirdikleri çalışmalarında, hem TÜKVS'nin hem de NÜKVS'nin uygulamasını gerçekleştirmişlerdir. Çalışmada sezgisel sıralama tekniklerinin grafik renklendirme algoritmalarına çok benzediğini ve sınav çizelgeleme problemlerine başarı ile uygulandığını belirtmişler. Çalışmalarında, probleme özgü adaptif metodun kullanıldığı uygun olmayan sezgiseller yerine uygun sezgiselin test problemlerinin sınandığı esnada seçildiğini belirtmişlerdir. Çalışmalarında ayrıca, rasgele bir sezgiselle başlatılmış bir problem çözümünün adaptif bir şekilde uygun sezgiseli seçerek çözüme devam ettiğini de dile getirmişler. Çalışmadan ayrıca Başlık 3.2.1'de de bahsedilmiş ve TÜKVS için elde edilen sonuçlar verilmiştir.

2009 yılında Abdul Rahim ve arkadaşları (Abdul Rahim v.d., 2009), granül modelleme ile sınav yerleşim önerisinde bulundukları çalışmalarında, yalnızca NÜKVS kullanarak bir uygulama gerçekleştirmişlerdir. Gerçekleştirilen uygulamada hem katı kısıtlar hem de hafif kısıtlar üzerinde durulmuştur. Önerilen yöntemin ayrıntılı olarak aktarıldığı çalışmada, NÜKVS uygulaması kapasite kısıtlı ve kapasite kısıtsız olarak farklı şekillerde değerlendirilmiştir.

Abdul Rahim ve arkadaşları (Abdul Rahim v.d., 2013a) 2013 yılına gelindiğinde de sınav çizelgeleme problemlerinin deterministik optimizasyonu için alan dönüşüm yaklaşımı önerisinde bulunmuşlardır. Önerilen yöntem TÜKVS, NÜKVS ve ITC2007 veri setinin üçüne de uygulanmıştır. Gerek önerilen yöntemin gerekse de uygulama sonuçlarının ayrıntılı olarak verildiği çalışmada, mevcut sonuçlara yakın ve rekabet edebilen çıktılar elde edilmiştir.

Yazarlar aynı yıl gerçekleştirdikleri bir diğer çalışmalarında (Abdul Rahim v.d., 2013b) sınav çizelgeleme problemlerinin çözümü için tepe tırmanma algoritması ve genetik algoritma yaklaşımlarını uygulamışlardır. Hem TÜKVS hem de NÜKVS için ayrı ayrı gerçekleştirilen uygulamalardan elde edilen sonuçlar karşılaştırmalı olarak ayrıntılı bir şekilde verilmiştir. Tüm sonuçlar incelendiği zaman, tepe tırmanma algoritmasından elde edilen sonuçların daha iyi olduğu gözlemlenmiştir. Genetik algoritma yaklaşımıyla uygulamada elde edilen sonuçlarda ise başlangıç değerleri ile sonuç değerleri arasındaki farkların dar olduğu dikkat çekmektedir.

2015 yılına gelindiğinde ise Wong ve Assal (Wong ve Assal, 2015), çoklu yerel arama operatörlerini kullanarak, çoklu komşuluk yapılarını da içeren bir çözüm yaklaşımı geliştirmişlerdir. TÜKVS ve Melbourne Üniversitesi Kıyaslama Veri Setinin de çözüldüğü makalede, NÜKVS'nin çözümü 2 saat sonunda elde edilmiş ve en iyi ceza puanı olarak 6,4 değerine ulaşılmıştır.

3.2.3. Melbourne Üniversitesi Kıyaslama Veri Seti Kullanılan Çalışmalar

NÜKVS'nden sonra literatüre kazandırılan bir diğer kıyaslama veri seti de Melbourne Üniversitesi Kıyaslama Veri Setidir (MÜKVS). MÜKVS ilk olarak 2002 yılında Merlot ve arkadaşları (Merlot v.d., 2002) tarafından tanıtılmıştır. Veri setinde, detayları aşağıda yer alan tabloda da verilmekte olan iki farklı problem yer almaktadır.

Tablo 3.11: Melbourne Üniversitesi Kıyaslama Veri Seti.

Problem	Sınav Sayısı	Öğrenci Sayısı	Öğrenci Sınavları Sayısı	Sınav Kuşağı Sayısı
MEL–F–01	521	20656	62248	28
MEL–S–01	562	19816	60637	31

Tabloda yer alan bilgilerden de anlaşılabileceği gibi, MÜKVS de belirli sayıda sınav kuşağında gerçekleştirilmesi gereken kapasite kısıtlı bir sınav çizelgeleme problem veri setidir. Literatürde, diğer kıyaslama veri setlerine nazaran MÜKVS kullanılarak gerçekleştirilmiş olan çalışma fazla değildir. Literatürde, MÜKVS kullanılarak gerçekleştirilmiş olan, erişilebilmiş iki çalışmaya aşağıda yer verilmiştir.

2004 yılında Côté ve arkadaşları (Côté v.d., 2004), hibrit çok amaçlı evrimsel algoritma önerisinde bulunmuşlardır. Önerilen hibrit yaklaşımda, geleneksel genetik operatörler yerine yerel arama operatörleri kullanılmıştır. Çalışmada kullanılan iki adet arama operatöründen biri, başlangıçta ve mutasyon operatörleri tarafından üretilen mümkün olmayan çizelgelerin tamiri için, diğeri de basitleştirilmiş değişken komşuluk metasezgiseli uygulayarak, yakınsama maliyetinin iyileştirilmesi için tercih edilmiştir.

2005 yılında ise Wong ve arkadaşları (Wong v.d., 2005), çoklu komşuluk ve çoklu operatör stratejileri içeren bir algoritma önerisinde bulunmuşlardır. Algoritmaya dahil edilen bu çoklu yapılar, arama işlemleri sırasında çeşitlendirme ve yoğunlaştırma arasında verimli bir etkileşimin sağlanması için gerçekleştirilmiştir. Önerilen yaklaşım, MÜKVS'nin yanı sıra TÜKVS ve NÜKVS'ne de uygulanmıştır. Gerçekleştirilen uygulamalar neticesinde elde edilen sonuçlar, literatürde yer alan farklı sonuçlarla da karşılaştırılarak ayrıntılı olarak raporlanmıştır. Merlot ve arkadaşları (Merlot v.d., 2002) tarafından elde edilmiş olan sonuçlar, çalışmada gerçekleştirilen uygulamalarla % 10 oranında iyileştirilmiştir.

3.2.4. ITC2007 Veri Seti Kullanılan Çalışmalar

ÜSÇP'nde kullanılan test kıyaslama veri setlerinden bir diğeri de Uluslararası Çizelgeleme Müsabakası 2007'de (ITC2007) tanıtılan sınav çizelgeleme veri setidir. ITC2007'de üç farklı veri seti tanıtılmıştır ve bu veri setlerinden ilki sınav çizelgeleme üzerinedir. ITC2007'de tanıtılmış olan sınav çizelgeleme veri setine ait detaylar, McCollum ve arkadaşları tarafından 2007 yılında yayınlanmış olan teknik raporda verilmiştir (McCollum v.d., 2007). 12 adet çizelgeleme problemi içeren veri setine ait detaylar ayrıca aşağıda yer alan tabloda da ayrıntılı olarak verilmektedir (Abdul Rahman v.d., 2011).

Tablo 3.12: İkinci Uluslararası Çizelgeleme Müsabakası Veri Seti (ITC2007).

Sınavlar	Çakışma Yoğunluğu	Sınav Sayısı	Öğrenci Sayısı	Sınav Kuşağı	Sınav Salonları	Periyot HC	Salon HC
Sınav–1	5,05	607	7891	54	7	12	0
Sınav–2	1,17	870	12743	40	49	12	2
Sınav–3	2,62	934	16439	36	48	170	15
Sınav–4	15	273	5045	21	1	40	0
Sınav–5	0,87	1018	9253	42	3	27	0
Sınav–6	6,16	242	7909	16	8	23	0
Sınav–7	1,93	1096	14676	80	15	28	0
Sınav–8	4,55	598	7718	80	8	20	1
Sınav–9	7,84	169	655	25	3	10	0
Sınav–10	4,97	214	1577	32	48	58	0
Sınav–11	2,62	934	16439	26	40	170	15
Sınav–12	18,45	78	1653	12	50	9	7

ITC2007 de NÜKVS gibi kapasite kısıtlı bir problem veri setidir. ITC2007 de diğer kapasite kısıtlı problemlere benzer, katı ve hafif kısıtları içermektedir. Bu problemde ayrıca özellikle belirtilmiş olan katı kısıt, kuşak uzunluklarının ihlal edilemez olduğudur. Tez çalışmasında odaklanılan veri seti TÜKVS olduğundan dolayı, NÜKVS'nde olduğu gibi, ITC2007 veri seti için de erişilebilmiş olan literatür taramasının bir kısmına aşağıda yer verilmeye çalışılmıştır.

Abdul Rahman ve arkadaşları 2011 yılında (Abdul Rahman v.d., 2011) gerçekleştirmiş oldukları çalışmalarında, sınav takviminin oluşturulmasında sezgisellerin sıralamalarının lineer kombinasyonlarını araştırmışlardır. Çalışmada yazarlar, her bir parametre için belirli problem verilerinin genelleştirilmesi adına bir normalleşme stratejisine başvurmuşlardır. Her bir sınav takviminin ataması sırasında, grafik renklendirme tekniklerinden iki farklı tekniğin lineer kombinasyonları elde edilmeye çalışılmıştır. ITC2007 veri setine uygulanan çalışmada, her bir problem

için en iyi sonucun elde edildiği kombinasyonlar ve mevcut literatür sonuçlarıyla karşılaştırmalı tablolar ayrıntılı olarak verilmiştir.

Aynı yıl Turabieh ve Abdullah (Turabieh ve Abdullah, 2011) da etkin sezgisel operatörleri great deluga algoritmasının içerisinde kullanarak, bir melez yaklaşım önerisinde bulunmuşlardır. Önerilen yaklaşımı hem TÜKVS'ne hem de ITC2007'ye uygulayan yazarlar, önerilen yaklaşımla elde edilen sonuçların kalitesine vurgu yapmışlardır. Ayrıca, üniversitelerin kendine özgü kısıtlarının da kolayca algoritma içerisine dahil edilebileceğinden de bahsedilmiştir. ITC2007 uygulamaları neticesinde elde edilen sonuçlar, bilinen en iyi sonuçlara oldukça yakın değerler içermektedir.

2013 yılında Bykov ve Petrovic (Bykov ve Petrovic, 2013), sınav çizelgeleme problemlerinin çözümü için yeni bir tepe tırmanma algoritması yaklaşımı önerisinde bulunmuşlardır. Önerilen yaklaşımda, çözüm zamanlarını uzatmasına karşın daha kötü çözümlere giden hareketlere izin verilerek, en iyi çözümün elde edilmesi hedeflenmiştir. 12 adet probleme de uygulanan yaklaşımla, problemlerin bilinen en iyi sonuçları iyileştirilmiştir.

Sabar ve arkadaşları (Sabar v.d., 2013) da 2013 yılında kombinatoryal optimizasyon problemlerinin çözümü için, dil bilgisel evrim hiper–sezgiseli geliştirmişlerdir. Çalışmada önerilmekte olan dil bilgisel evrim yaklaşımı, farklı sezgisel bileşenlerden oluşan çevrimiçi bir çözücüyü, güçlendirici olarak kullanılmaktadır. Yöntem, sınav çizelgeleme problemleri için TÜKVS ve ITC2007 veri setine ve kapasite kısıtlı araç rotalama problemlerine uygulanarak test edilmiştir.

2014 yılında Battistutta ve arkadaşları (Battistutta v.d., 2014), tek aşamalı tavlama benzetimi yaklaşımı önerisinde bulunmuşlardır. ITC2007 problemlerini uygulayan yazarlar, çalışma boyunca özellikle parametrelerin etkilerini de anlamak adına, istatistiksel temele dayanan deneysel analizler de gerçekleştirmişlerdir. 12 adet problemin tamamında da en iyi ya da en iyiye çok yakın sonuçlar elde edilememiş olmasına karşın, yöntemin sonuçları rekabet edebilir niteliktedir.

Aynı yıl Hamilton–Bryce ve arkadaşları (Hamilton – Bryce v.d., 2014) da, uyarlama aşamasında elde edilen ağırlıklandırmalı listelerdeki bilgilerle güçlendirilmiş bir stokastik yöntem önerisinde bulunmuşlardır. Bu öneride bulunmalarının amacının da, akıllıca yönlendirilen bir sınav seçim tekniği tanıtmak olduğunu belirtmişlerdir. Algoritmaya, ağırlıklandırılmış listenin en etkili bölümlerinin tanıtılması ve genel sonuçların iyileştirilmesi amacıyla da takviyeli öğrenme tekniği geliştirilmiştir. Yalnızca ITC2007'ye uygulanmış olan çözüm önerisi neticesinde elde edilen sonuçlar, mevcut sonuçlarla rekabet edebilir niteliktedir.

2015 yılına gelindiğinde, Alzaqebah ve Abdullah (Alzaqebah ve Abdullah, 2015) tarafından, sınav çizelgeleme problemlerinin çözümü için hibrit arı kolonisi optimizasyonu önerilmiştir. Yazarlar çalışmalarında, komşuluk aramalarını geliştirmek amacıyla bir komşuluk yapısının seçimi için kendiliğinden uyarlamalı bir mekanizma tanıtmışlardır. Ayrıca, önerilmekte olan algoritma içerisinde gerçekleştirilen yerel aramalar, tepe tırmanma algoritması ve tavlama benzetimi algoritmasıyla desteklenmişlerdir. Önerilmekte olan algoritma, TÜKVS ve ITC2007 veri setine uygulanmış ve son zamanlarda yapılmış olan çalışmaların sonuçlarıyla da karşılaştırılarak ayrıntılı bir biçimde aktarılmıştır.

3.2.5. Diğer Gerçek Hayat Problemleri Kullanılan Çalışmalar

Literatür taramasının buraya kadar aktarılmış olan bölümlerinde, literatürde en fazla bilinen ve sıklıkla kullanılan kıyaslama veri setleri tanıtılmaya ve bu veri setlerinin kullanıldığı çalışmalara dair ayrıntılar belirtilmeye çalışılmıştır. Çalışmanın bu bölümünde de kıyaslama veri setlerini kullanmayıp, daha önce gerçekleştirilmemiş gerçek hayat problemleri kullanılan çalışmalar aktarılmaya çalışılacaktır. Literatürde sınav çizelgelemesine dair çok çeşitli çalışmalar bulunduğundan dolayı burada aktarılacak olan erişilebilmiş literatür taraması, genetik algoritma kullanılan çalışmalarla sınırlandırılacaktır.

Wong ve arkadaşları 2002 yılında (Wong v.d., 2002), genetik algoritma kullanarak final sınavlarının çizelgelenmesi hakkında bir çalışma gerçekleştirmişler. Önerilmiş olan algoritma, Ecole de Technologie Superieure'den elde edilmiş olan sınav veri setine uygulanmıştır. Elde edilen sonuçlara göre hiç sınav çakışması olmadığı, hiç bir

öğrencinin arka arkaya 3 sınava girmediği, her bir periyotta yapılması gereken sınav sayısı koşulunun gerçekleştirildiği, öğrencilerin % 0,8'inin arka arkaya 2 sınavının olduğu, % 11'inin ise 1 veya 2 boşluklu bir çift sınavının bulunduğu, diğer bütün öğrencilerin 2 boşluklu sınavlarının olduğu belirtilmiş. Çalışmanın sonuç kısmında, sınav çizelgeleme için kullanılan genetik algoritmanın, algoritma geliştirme ve uygulama açısından oldukça kolay olduğu ve başka bir sezgisele ihtiyaç duyulmadan gerçekleştirildiği belirtilmiş.

Aynı yıl, genetik algoritma kullanılarak geliştirilen bir diğer çalışma da Yasin ve arkadaşları (Yasin v.d., 2002) tarafından gerçekleştirilmiştir. Geliştirilmiş olan algoritma, Universiti Utara Malaysia'da uzaktan eğitim biriminin final sınavlarının çizelgelemesi için uygulanmıştır. Önerilmiş olan sistemin, farklı uygun çözümler ürettiğini ve bunu kısa bir zamanda gerçekleştirdiğini dile getiren yazarlar, son olarak ise operatörlerin uygun çözüme olan etkilerini incelemişler.

2005 yılında Hung ve arkadaşları (Hung v.d., 2005), genetik algoritmanın ve yerel arama operatörünün bir arada kullanıldığı bir çözüm yaklaşımı önerisinde bulunmuşlardır. Önerilmiş olan yaklaşım, HoChiMinh City University of Technology'den elde edilmiş olan 324 ve 1820 adet sınav grupları içeren gerçek hayat veri setlerine uygulanmıştır. Yazarlar, genetik algoritma çözümünden elde edilen sonuçların, tavlama benzetimiyle elde edilen sonuçlardan daha iyi olduğunu ve ayrıca başlangıç ve parametre değişikliklerinden de nispeten daha az etkilendiğini belirtmişlerdir. Genetik algoritmanın, çeşitli zorluktaki gerçek hayat çizelgeleme problemlerinin üstesinden gelmede güçlü bir algoritma olduğuna inandıklarını da çalışmada belirtmişlerdir.

Cupic ve arkadaşları (Cupic v.d., 2009) da 2009 yılında genetik algoritma kullanılmış olan bir çözüm önerisinde bulunmuşlardır. Veri setine ait detaylı bilgilendirmenin bulunmadığı çalışmada, 77 derse ait sınavların 10 gün içerisinde gerçekleştirilecek olan 30 adet kuşağa yerleştirileceğinden bahsedilmiştir. Çalışmada, herhangi bir karşılaştırma yapılmadan yalnızca gerçekleştirilmiş olan uygulamaya ait sonuçlara yer verilmiştir.

2010 yılına gelindiğinde, Alsharafat ve Alsharafat (Alsharafat ve Alsharafat, 2010) üç farklı genetik algoritma yaklaşımı geliştirmiştir. Geliştirmiş oldukları algoritmaları Al AI-Bayt Üniversitesinden edindikleri veri seti ile çalıştırmışlardır. Önerilen algoritma, literatürde yer alan bir test veri seti ile çalıştırılmadığından dolayı sonuçların karşılaştırmaları mümkün olmamıştır.

Arogundade ve arkadaşları (Arogundade v.d., 2010) Nijerya Ziraat Üniversitesi veri seti olan, gerçek hayat problemine uyguladıkları bir genetik algoritma yaklaşımı geliştirmişlerdir. Çalışmada, hiyerarşik yapı için ağırlıklandırmalar kullanılmıştır. Bir öğrencinin aynı anda iki sınava girmemesi katı kısıtının yanında hafif kısıt olarak, öğrencilerin sınavları arasında çalışma zamanları oluşturabilmek için boşluklar bırakılmasına da odaklanılmıştır.

To ve Wim (To ve Win, 2010), belirli kısıtlar altında gerçekleştirilen sınav çizelgeleme problemlerinin çözümü için hiyerarşik kümeleme yaklaşımı geliştirmişlerdir. Önerilen yöntem, rastgele oluşturulmuş üç farklı veri seti ile test edilmiştir. Yöntemde, iki farklı amaç fonksiyonu oluşturularak her ikisinin de iyileştirilmesi hedeflenmiştir. Yüzdelik olarak elde edilen bu sonuçlarda, katı kısıtların ve hafif kısıtların tamamen sağlanması değil yüzdeliklerin sıfıra yaklaştırılması amaçlanmıştır.

2011 yılında Komar ve arkadaşları (Komar v.d., 2011), genetik algoritmanın da içinde bulunduğu dört farklı sezgisel tekniği kullanarak, evrimsel hesaplamalı algoritmaların dağıtılmasını içeren bir yaklaşım önerisi gerçekleştirmişlerdir. Önerilmekte olan yaklaşımın uygulaması, Hırvatistan'da bulunan Zagreb Üniversitesi'nin Faculty of Electrical Engineering and Computing'den elde edilmiş veri seti ile gerçekleştirilmiştir. Gerçekleştirilmiş olan uygulama sonuçları yalnızca zaman olarak değerlendirilmiş ve iyi kalitede sonuçların elde edildiği yöntemin daha da geliştirilebileceğinden bahsedilmiştir.

2012 yılında Kalaycı ve Güngör (Kalaycı ve Güngör, 2012), öğrenci başarısına odaklanan, genetik algoritma tabanlı bir sınav çizelgeleme önerisinde bulunmuşlardır. Öğrenci başarısına odaklanan bu yaklaşım, öğrencilerin sınavları arasında yeterli miktarda çalışma ve dinlenme zamanları bırakmayı hedeflemektedir. Bunun için de çalışmada kullanılan amaç fonksiyonu, sınavların zorluklarını da göz önünde bulundurarak öğrencilerin sınavları arasında süreyi maksimize etmeye odaklanmıştır. Çözüm için geliştirilmiş olan iki farklı genetik algoritma modelinden biri, uygun olmayan çözümleri popülasyondan elemeye odaklanırken, diğeri de popülasyona yeni katılacak olan bireylerin katı kısıtları sağlayıp sağlamadıklarını kontrol etmektedir. Yazarlar, önermiş oldukları algoritmayı Pamukkale Üniversitesi Mühendislik Fakültesinden elde edilmiş verilere uygulamışlardır.

Innet 2013 yılında (Innet, 2013), sınav takvimi düzenlemesi için çaprazlama, mutasyon ve seçim operatörlerini içeren bir genetik algoritma yaklaşımında bulunmuştur. Önerilmiş olan yöntem, hem derslere ait sınavların çizelgelenmesini hem de öğrencilerin sınav için gruplanmalarını içermektedir. Önerilmiş olan algoritmanın uygulaması, University of the Thai Chamber of Commerce (UTCC) Mühendislik Fakültesi'nde 2012–2013 yılı lisans düzeyi final sınavlarının düzenlenmesi için gerçekleştirilmiştir. Çalışmada ayrıca, uygulama sırasında bazı sonuçların elde edilmesi için simülasyon uygulaması gerçekleştirildiğinden de bahsedilmiştir.

2014 yılına gelindiğinde Hosny ve Al–Olayan (Hosny ve Al–Olayan, 2014), hem sınav salonlarının hem de gözetmen atamalarının gerçekleştirilmesi için mutasyon tabanlı bir genetik algoritma önerisinde bulunmuşlardır. Önerilmiş olan algoritmanın, King Saud Üniversitesi Bilgisayar ve Enformasyon Bilimleri Fakültesi Bilgisayar Bilimleri Bölümünün, özellikle yüksek lisans programlarında bulunan kadın bölümlerine uygulandığından bahsedilmiştir. Çalışmada yazarlar, cinsiyet farklılığı gibi kültürel kısıtlamalar dolayısıyla gelen çeşitli kriterlerin de dikkate alındığından bahsetmiştir. Çalışma, çaprazlama operatörü dışında klasik genetik algoritmayla aynı kuralları içermektedir. Mutasyon operatörü ise çalışmada ana genetik operatör olarak tercih edilmektedir. İki aşamada gerçekleştirilen çalışmanın ilk aşaması, sınav

kapasiteleri ve özelliklerini de dikkate alarak sınıf atamalarını, ikinci aşaması ise gözetmen atamalarının gerçekleştirilmesini içermektedir. Problem tiplerinin de farklılıkları dolayısıyla, her iki aşama da sağlanması gereken farklı katı kısıtları içermektedir. Çalışmada önerilen algoritmanın yoğunlaştığı ana nokta, kültürel farklılıklardan kaynaklanan katı kısıtların sağlanmasını içerdiğinden dolayı algoritmanın etkinliğinin ölçülmesi de bu yönde gerçekleştirilmiştir.

Aynı yıl Akinwale ve arkadaşları (Akinwale v.d., 2014), sınav çizelgeleme problemlerinin çözümü için, genetik algoritma ve tavlama benzetimi yöntemlerini birlikte içeren hibrit bir yaklaşım önerisinde bulunmuşlardır. Önerilmiş olan algoritma, 20100 adet öğrencinin aldığı 652 adet dersin sınavlarını içeren Nigeria Ladoke Akintola University of Technology 2010/2011 birinci dönem sınavlarının, 52 adet sınav yerine ve Cumartesi–Pazar hariç olmak üzere 17 güne yerleştirilmesi için uygulanmıştır. Hiçbir sınavın ya da gözetmenin iki defa çizelgelenmeden başarılı bir çizelgeleme işleminin gerçekleştirildiğini belirten yazarlar, çalışmanın sonuçlarının önemli zaman kazancı gerçekleştirdiğinden de bahsetmişlerdir.

2015 yılında Xuecheng (Xuecheng, 2015), Hebei University of Architecture'da iki dönemde gerçekleştirilen bütünleme sınavlarının düzenlenmesi için hibrit genetik algoritma önerisinde bulunmuştur. Çalışmada gerçekleştirilmiş olan öneride, derslik sayısını en aza indirmek amacıyla açgözlü algoritma ve orijinal bir mutasyon operatörü birlikte kullanılmıştır. Tanıtılmış olan yeni operatörün, çaprazlama ve yerel arama operatörlerinin bir kombinasyonu olduğunu belirten yazar, çok güçlü bir genetik algoritmaya ulaşmayı sağladığına da değinmiştir.

Abdelhalim ve El Khayat (Abdelhalim ve El Khayat, 2016), üniversite çizelgeleme problemlerinin çözümü için geliştirdikleri kullanım tabanlı bir genetik algoritma yaklaşımını 2016 yılında tanıtmışlardır. Çalışmada belirtilen ve gerçekleştirilmeye çalışılan amaç, kaynak kullanımını maksimize etmeyi dikkate almaktadır. Tanıtılmış olan genetik algoritma yaklaşımı, uygun ve iyi kalitede bir başlangıç popülasyonu oluşturmak için bazı sezgisel teknikler kullanmaktadır. Yerel aramalar için hedeflenen bir mutasyon operatörü kullanılmaktadır. Gözetmenlerin sınavları arasındaki boşlukları azaltarak, bir gündeki sınav sayılarını maksimum düzeyde

tutmayı hedefleyen bir amaç fonksiyonu kullanılmış olan çalışmada, profesörlerin tercihlerini dikkate almak ve çakışmaları da önlemek adına basit bir ağırlıklı toplam formülü de kullanılmaktadır. Önerilmiş olan algoritma, yazarların da görevli olduğu Alexandria University Faculty of Commerce veri setine uygulanmıştır. Sonuçları değerlendiren yazarlar, büyük üniversitelerin çizelgelemelerinin oluşturulması için çalışmada önerilmiş olan algoritmanın etkili bir araç olduğunu da belirtmişlerdir.

Tez çalışmasının buraya kadar olan bölümlerinde, mevcut bilgiler ve çalışmalar aktarılmaya ve değerlendirilmeye çalışılmıştır. Bu amaçla öncelikle ilk bölümde; uzman sistemler, kullanım alanları, avantajları ve dezavantajları ayrıntılı bir biçimde incelenmiştir. İkinci bölümde, literatürde en yaygın olarak kullanım görmüş olan altı adet sezgisel yöntem detaylı olarak aktarılmış ve bölümün son kısmında, diğer sezgisel tekniklere de kısaca değinilmeye çalışılmıştır. Çalışmanın üçüncü bölümü de üniversite sınav çizelgeleme problemine ve literatür taramasına ayrılmıştır. Bu bölümde öncelikle, üniversite sınav çizelgeleme problemi tanıtılmış ve daha sonra bileşenleri ayrıntılı olarak incelenmiştir. Bu bölümün ikinci kısmı olan literatür taraması kısmında ise problem çeşitlerine göre ayrıntılı bir literatür taramasına yer verilmiştir. Özellikle uygulaması gerçekleştirilen, TÜKVS kullanılmış olan çalışmalar ve sonuçlarına dair detaylı bir raporlama da bu bölümde aktarılmaya çalışılmıştır.

Buraya kadar aktarılmış olan üç bölümden de edinilmiş olan bilgiler ışığında, öncelikle dördüncü bölümde, tez çalışmasında önerilmekte olan üniversite sınav çizelgeleme yöntemi ayrıntılı bir biçimde, iki başlık altında aktarılacaktır. Beşinci bölümde de, önerilmekte olan yöntem kullanılarak gerçekleştirilmiş olan uygulamalar ve sonuçları yine iki başlık altında ayrıntılı olarak aktarılacaktır.

4. ÖNERİLEN ÜNİVERSİTE SINAV ÇİZELGELEME YÖNTEMİ

Tez çalışmasının buraya kadar olan kısımlarında üç ana başlık halinde uzman sistemler, sezgisel yöntemler ve üniversite sınav çizelgelemesine ait detaylı incelemelere yer verilmiştir. Bu bölümde ise, üniversite sınav çizelgeleme problemlerinin çözümü için önerilen çözüm yöntemi ayrıntılı olarak aktarılacaktır.

Önerilen üniversite sınav çizelgeleme problemi çözüm yöntemi iki ana kısımdan oluşmaktadır. İlk kısımda uzman sistem kullanılarak, derslerin kuşak gruplandırmaları en az kuşak sayısı içerecek şekilde gruplanacaktır. Sonrasında gerçekleştirilecek sezgisel çözüm yöntemiyle de, kuşak gruplandırmalarının öğrenciler için optimum sınav düzeni oluşturacak şekilde düzenlenmesi gerçekleştirilecektir.

Önerilmekte olan üniversite sınav çizelgeleme problemi çözüm yönteminin tamamı java programlama dili kullanılarak gerçekleştirildiğinden dolayı, yapılacak olan anlatımlar java'ya uygun şekilde gerçekleştirilecektir.

4.1. ÖNERİLEN UZMAN SİSTEM YAKLAŞIMI

Literatürde yer alan çalışmalar incelendiği zaman, üniversite sınav çizelgeleme probleminin çözümü için yapılan çalışmaların tamamının sezgisel yöntemlerle gerçekleştirilmiş çözüm önerileri olduğu görülmektedir. Yapay zeka tekniklerinden biri olan uzman sistem yaklaşımı kullanılarak gerçekleştirilmiş olan bir çalışmanın ise erişilebilmiş olan çalışmalarda çok az gerçekleştirildiği gözlemlenmiştir. Bu durumdan dolayı bu tez çalışmasında ağırlıklı olarak uzman sistem çözümünde yoğunlaşılmıştır.

Bir uzmanın tercihleri ve kararları dikkate alınarak gerçekleştirilmiş bir uzman sistem yardımıyla etkili çözüm önerilerinin bulunabildiği, literatürde yer alan farklı uygulama çalışmaların sonuçlarıyla görülmektedir. Yapılan literatür taramaları ve uzman sistem incelemeleri sonrasında, bu çalışmada gerçekleştirilecek olan

çizelgeleme problemlerinin çözüm yaklaşımında da, uzman sistem kullanılması durumunda etkili ve verimli sonuçlar elde edilebileceği düşünülmüştür. Önerilmekte olan çözüm yaklaşımının uygulaması gerçekleştirildiğinde de, sonuçların bu savı destekler nitelikte olduğu görülmüştür. Elde edilen uygulama sonuçlarının ayrıntılı ve karşılaştırmalı olarak incelenmesi de, uygulama bölümü ile tartışma ve sonuç bölümünde verilmektedir.

Literatürde yer alan çalışmalarda, öğrencilerin almış oldukları ders listelerine göre karşılaştırmalar yapılarak, farklı derslerin yan yana gelip gelemeyeceklerinin kontrolü gerçekleştirilmektedir. Bu kontrolün yapılmasının amacı, bir öğrencinin aynı sınav kuşağında almakta olduğu farklı derslerin sınavlarına aynı anda giremeyecek olması durumundan dolayıdır. Yani, bu kontroller sayesinde bir öğrencinin aynı sınav kuşağında, aynı anda birden fazla sınavının olmasını engellemektedir. Gerçekleştirilmiş olan bir çözüm algoritmasında her bir iterasyonda bu kontrollerin yapılması, çözüm sürelerini hayli uzatmaktadır. Örneğin, 20.000 adet öğrencinin ve 1.000 adet de dersin bulunduğu bir sistemde, her seferinde on adet ders için bile kontrol yapılsa, bu işlem bir milyondan fazla karşılaştırma demektir. Bu karşılaştırmaların da her iterasyonda gerçekleştirilmesi, süre uzatmaktan ziyade bir durum olmamaktadır. Özellikle sezgisel yöntemler gibi, atanmış olan derslerin yer değiştirmelerinin ve tekrar atamalarının gerçekleştirildiği bir sistemde bu sayı hiçbir zaman azalmayacaktır.

Karşılaştırmaların ve atamaların gerçekleştirilmesinden sonra da, yapılan atamanın uygunluğunun kontrolü için bir de ceza fonksiyonu hesaplaması yapılmaktadır. Bu ceza fonksiyonu hesaplaması için de, her bir iterasyonda tek tek öğrencilerin almış oldukları derslerin bulunduğu kuşaklara ve bu kuşakların birbirleri ile olan mesafelerine bakılmaktadır. Yukarıda verilen öğrenci sayısına sahip bir sınav çizelgeleme probleminde bu fonksiyon hesaplaması da, yine bir milyonu aşan toplam döngü sayılarını bulabilmektedir. Bu kadar yüksek döngülerin ve hesaplamaların bulunduğu bir sistemde, yine çözüm zamanları yüksek rakamlara ulaşabilmektedir.

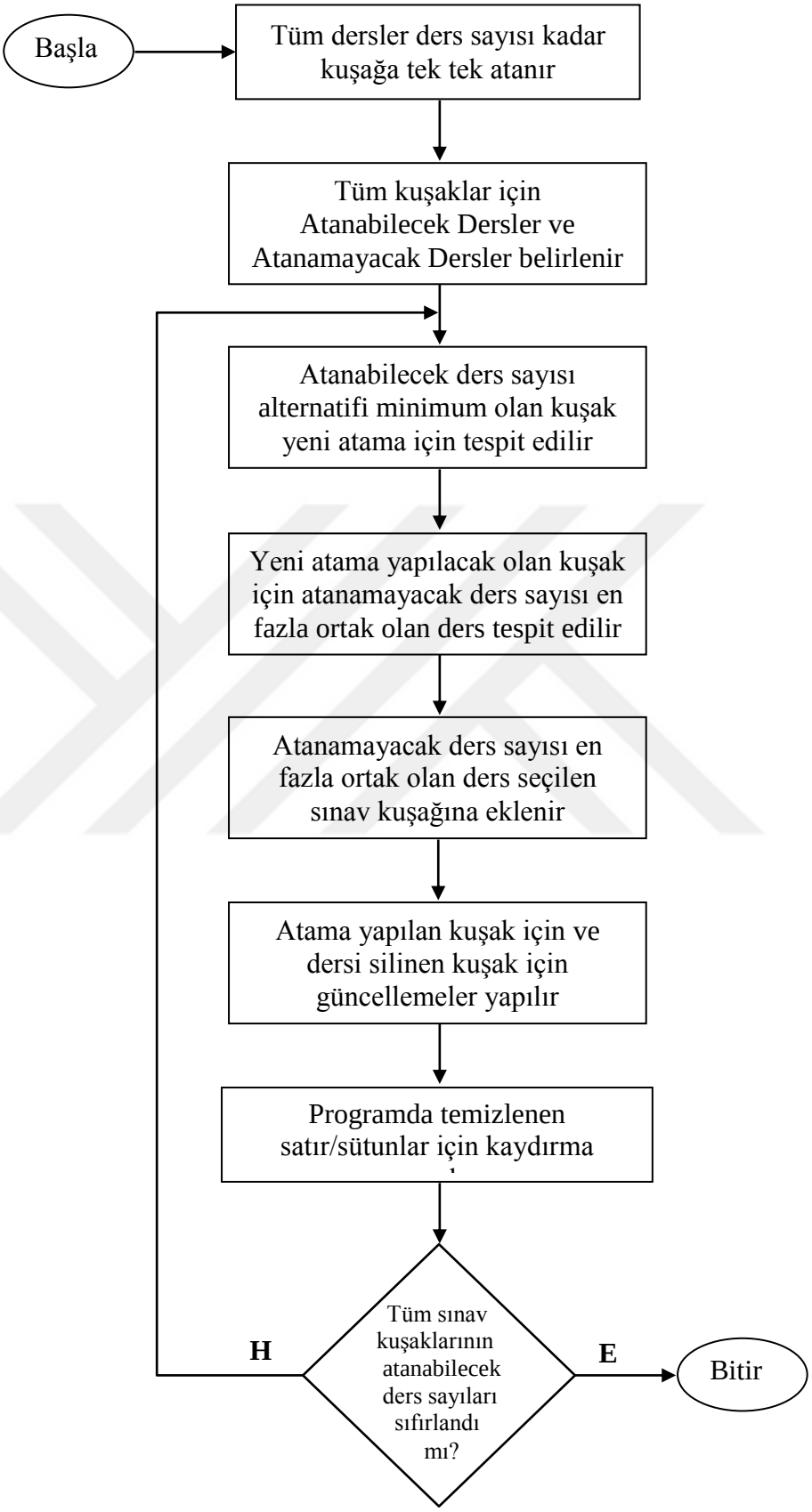
Öyle ki; literatürde yer alan üniversite sınav çizelgeleme problemi çözüm yöntemi önerileri ve uygulamaları incelendiği zaman, özellikle TÜKVS kullanarak

gerçekleştirilen uygulamalarda, yüksek öğrenci sayılarına ve yüksek ders sayılarına sahip problemlerin (Örneğin Pur93 problemi) çözümlerinin, çalışmaların büyük bir çoğunluğunda gerçekleştirilmediği ve karşılaştırmalara dahil edilmediği dikkat çekmektedir.

Bu tez çalışmasında gerçekleştirilecek olan çözüm yönteminin en önemli önerilerinden biri, öğrencilerin ders alma listelerine dayalı bu karşılaştırmaları ortadan kaldırarak, çözüm zamanlarını üstel bir şekilde azaltmaya çalışmaktır. Bunun için de bir defaya mahsus olmak üzere; öğrencilerin ders listelerinin derslere göre karşılaştırmaları yapılarak, her ders için aynı kuşağa koyulamayacak dersler listesi oluşturulacaktır. Daha sonra gerçekleştirilecek tüm işlemler bu liste üzerinden yapılacaktır. Bu sayede, yüksek sayıda öğrenci listeleri için yapılan karşılaştırmalar, yalnızca sınırlı sayıda dersler için yapılmış olacaktır.

Uzman sistemle sağlanacak olan ikinci önemli avantaj ise, kuşak gruplamalarının gerçekleştirilmesi sırasında kural tabanlı atamalar yapıldığından ve herhangi bir ceza fonksiyonu hesaplaması yapılmadığından dolayı, tüm öğrencilere ait sınav programlarının tek tek incelenmesi gerekmemesidir. Tekrar eden iterasyonlarla birlikte uzun bir süreç haline gelen bu ceza fonksiyonu hesaplama süreci olmadığından dolayı da, çözüm sürecinin kuşak gruplandırma adımı oldukça kısa sürede tamamlanacaktır. Bu durum, özellikle yüksek öğrenci sayılarına ve ders sayılarına sahip çizelgeleme problemlerinin çözümü için avantajlı bir olanak sağlamaktadır.

Önerilen uzman sistem yaklaşımının bu süre avantajlarının yanında, sağlamış olduğu diğer avantajlara, ilerleyen kısımlarda çözümde yeri geldikçe değinilecektir. Uzman sistem yaklaşımının ayrıntılı olarak incelemesine geçmeden önce, aşağıda yer alan Şekil 4.1'de, önerilmekte olan uzman sistem yaklaşımı çözüm önerisine ait akış şeması, en genel haliyle verilmektedir.



Şekil 4.1: Uzman Sistem Çözümü Akış Şeması.

4.1.1. Önerilen Uzman Sistemde Kullanılan Yapılar

ÜSÇP'nin çözümü için önerilen sistemin daha iyi bir şekilde incelenebilmesi için, bu kısımda öncelikle, kullanılacak olan değişkenler ve listeler, açıklamaları ile birlikte aktarılmaya çalışılacaktır.

Derslerin Listesi: ÜSÇP'nde yer alan tüm derslerin ve o dersleri alan öğrenci sayılarının bulunduğu listedir. Ders sayısı kadar satır ve iki sütundan oluşmaktadır.

Öğrencilerin Aldığı Dersler Listeleri: Sistemde bulunan her bir öğrencinin almış olduğu derslerin kodlarının bulunduğu listedir. Öğrenci sayısı kadar satırdan ve öğrencilerin almış olduğu ders adetleri kadar sütundan oluşmaktadır.

Derslerin Öğrenci Listeleri: ÜSÇP'nde yer alan her bir ders için, o dersi alan öğrencilerin numaralarının sıralı olarak saklanmış olduğu listedir. Ders sayısı kadar satırdan ve dersleri alan öğrenci sayıları kadar sütundan oluşmaktadır.

Öğrencileri Çakışan Dersler Listeleri: Her bir dersin kendi listesinde bulunan öğrencilerin aldığı diğer dersler kontrol edilerek, bu derslerin eklendiği listedir. Ders sayısı kadar satır ve ders sayısının bir eksiği kadar sütundan oluşmaktadır. Böylece ilerleyen süreçlerde her seferinde öğrencilere ait listelerin kontrol edilmesi durumu ortadan kaldırılacak ve sadece bu liste baz alınarak işlemler gerçekleştirilecektir. Bundan dolayı, bu listenin doğru bir şekilde oluşturulması, uzman sistemin en önemli adımlarından birisidir.

Kuşağa Koyulamayacak Dersler Listeleri: Her bir sınav kuşağı için, o kuşağa atanan derslerin öğrencileri çakışan dersler listelerinden dersler çekilerek oluşturulmuş olan listedir. Ders sayısı kadar satır ve sütundan oluşmaktadır. Özellikle kuşak atamaları değerlendirilirken, öğrencileri çakışan dersler listesinden ziyade bu listelere bakılarak değerlendirmeler gerçekleştirilecektir.

Kuşağa Koyulabilecek Dersler Listeleri: Kuşağa koyulamayacak dersler listelerinde olmayan diğer derslerden oluşturulmuş olan listedir. Ders sayısı kadar satır ve sütundan oluşmaktadır. Bir kuşağa atama gerçekleştirileceği sırada, bu

listelerde bulunan dersler arasından seçim yapılacaktır. Yalnızca listede bulunan dersler kullanılarak atamalar yapılacağından dolayı süre avantajı oluşacaktır.

Sınav Programı: Başlangıçta, ders sayısı kadar satır ve sütundan oluşan ve zamanla kuşaklara ders atamaları gerçekleştirildikçe daralacak olan sınav programında yer alan derslerin kaydedildiği listedir.

4.1.2. Önerilen Uzman Sistem

Üniversite sınav çizelgeleme problemleri genellikle, yalnızca ders listesinden ve öğrencilerin aldığı dersler listelerinden oluşmaktadır. Bundan dolayı da önerilen uzman sistemin ilk adımı, sistemde de kullanılacak olan derslerin öğrenci listelerinin oluşturulmasıdır.

```
for(derslerin sonuna kadar)
   for(öğrencilerin sonuna kadar)
      for(öğrencinin ders listesi kadar)
         if(öğrencinin almış olduğu ders == listesi oluşturulan ders)
            öğrencinin numarasını dersin öğrenci listesine ekle
         endif
      endfor
   endfor
endfor
```

Şekil 4.2: Derslerin Öğrenci Listelerinin Oluşturulması.

Şekil 4.2'de belirtilen kod yardımıyla tüm derslere ait öğrenci listeleri oluşturulur. Ders kodlarının sıralı sayılar şeklinde olduğu düşünülürse, 6 dersten ve 10 öğrenciden oluşan bir sistem için, tüm derslerin öğrenci listelerinin oluşturulmasından sonra elde edilecek dizi, aşağıda gösterilen örnekteki gibi olabilecektir.

1	2
2	4
3	2
4	4
5	3
6	4

1	5		
3	7	9	10
2	3		
3	5	8	10
1	4	9	
2	4	6	8

Şekil 4.3: Derslerin Listesi ve Derslerin Öğrenci Listeleri Örneği.

Burada verilen örnekte, ilk dersi yalnızca 1 ve 5 numaralı öğrenciler almaktadır, ikinci dersi de 3, 7, 9 ve 10 numaralı öğrenciler almaktadır. Tüm dersler için liste bu mantıkla yorumlanacaktır.

Tüm derslerin öğrenci listelerinin oluşturulmasından sonra ise, öğrencileri çakışan dersler tespit edilerek, listeleri oluşturulur. Öğrencileri çakışan dersler listeleri, aynı kuşakta gerçekleştirilemeyecek sınavların listelerini vermektedir. Bu listeler sayesinde her bir kuşağa koyulabilecek derslere bakılırken, bu listelerde olan dersler elenecek ve kontrol edilecek olan ders sayıları azaltılacaktır. Aşağıda yer alan Şekil 4.4'te, öğrencileri çakışan dersler listelerinin oluşturulması verilmektedir.

```
for(derslerin sonuna kadar)
  for(dersin listesinde bulunan öğrencilerin sonuna kadar)
    for(listeden alınan öğrencinin ders listesinin sonuna kadar)
        if(öğrencinin aldığı ders dersin Çakışan Dersler Listesinde yoksa)
          dersi Öğrencileri Çakışan Dersler listelerine ekle
        endif
    endfor
  endfor
endfor
```

Şekil 4.4: Öğrencileri Çakışan Dersler Listelerinin Oluşturulması.

İlk verilen örnekte bulunan dersler için öğrencisi çakışan dersler listesinin oluşturulması gerçekleştirildikten sonra, aşağıda yer alan örnekteki gibi listeler oluşturulacaktır.

1	4	5		
2	4	5		
2	3	4	6	
1	2	3	4	6
1	2	5	6	
3	4	5	6	

Şekil 4.5: Öğrencisi Çakışan Dersler Listeleri Örneği.

Örnekten de incelenebileceği gibi, ilk ders ile aynı öğrencilere sahip olduklarından dolayı, bu ders ile aynı kuşağa 4. ve 5. ders koyulamayacaktır. İkinci ders için de durum aynıdır. Üçüncü ders ise aynı mantıkla 2., 4. ve 6. derslerle aynı kuşağa koyulamayacaktır. 4., 5. ve 6. dersler de, tabloda kendilerine ait satırlarda listelenen derslerle aynı kuşaklara atanamayacaklardır. Derslerin kendi kendine ataması yapılamayacağından dolayı, bu adımda derslerin kendileri de çakışan dersler listelerine eklenirler.

Bu adımdan sonra yapılacak işlemler için sınav programı gerekmektedir. Bundan dolayı da bir başlangıç sınav programı oluşturulmalıdır. Başlangıç sınav programı; hiçbir atama kontrolü gerektirmeyen ve ilerleyen adımların daha rahat bir şekilde gerçekleştirilmesine olanak sağlayan bir sistem olması için, mümkün olan en geniş şekilde yani, tüm dersler tek bir kuşakta olacak şekilde atanırlar. Sınav programı oluşturulduktan sonra da her bir kuşak için, o kuşağa koyulamayacak dersler listeleri oluşturulur. Algoritmanın başlangıcında, her bir kuşakta dersler sırasıyla ve tek başına yer aldığından dolayı; öğrencisi çakışan dersler listesi ile kuşağa koyulamayacak dersler listesi aynı olacaktır. Kuşağa koyulamayacak dersler listesinde bulunmayan diğer dersler de, kuşağa koyulabilecek dersler listesi olarak oluşturulur. Bu işlemin gerçekleştirilmesi de aşağıda yer alan Şekil 4.6'da verilmektedir.

```
for(derslerin sonuna kadar)
  for(derslerin sonuna kadar)
    for(Kuşağa Koyulamayacak Dersler Listelerinin sonuna kadar)
      if(seçilen ders Kuşağa Koyulamayacak Dersler Listelerinde yoksa)
          dersi Kuşağa Koyulabilecek Dersler Listelerine ekle
      endif
    endfor
  endfor
endfor
```

Şekil 4.6: Kuşağa Koyulabilecek Dersler Listeleri Oluşturma.

İncelemekte olduğumuz örneğimiz için, kuşağa koyulabilecek dersler listeleri Şekil 4.6'da verilen kod yardımıyla oluşturulursa, aşağıda verilen Şekil 4.7'de bulunan listeler elde edilecektir.

2	3	6
1	3	6
1	5	
5		
3	4	
1	2	

Şekil 4.7 : Kuşağa Koyulabilecek Dersler Listeleri Örneği.

Her bir ders kuşaklara tek tek atandığından dolayı, 6 adet dersten her bir ders için öğrencisi çakışan dersler listesinde bulunmayan dersler, o kuşak için kuşağa koyulabilecek dersler listesine eklenir. Yani ilk kuşağa, 2., 3. ve 6. dersten biri eklenebilecektir. 4. kuşağa ise yalnızca 5. ders eklenebilecektir. Önerilmekte olan algoritma ile, kuşaklara yapılacak olan ders atamalarını maksimize edebilecek ve bu sayede toplam ders ataması yapılan kuşak sayısını minimize edebilecek bir uzman sistem oluşturabilmek için, her bir iterasyonda; Kuşağa Koyulabilecek Ders Listeleri'nde bulunan, ders sayıları minimum olan kuşağa atama yapılarak ilerlenir. Burada verilen örnekte bulunan 5. ders, 3. veya 4. kuşağa atanabilecektir. Bu dersin atamasının 4. değil de 3. kuşağa yapılması durumunda, 4. kuşakta bulunan 4. dersin yanına başka hiçbir ders atanamayacaktır. Oysa 5. dersin 4. kuşağa atanması durumunda, 4. kuşağa iki ders atanmış olacak ve 3. kuşağa da 1. ders atanabilecektir. İlk durumda; 1., 3., 4. ve 5. dersler üç kuşağa atanacakken, bu şekilde

gerçekleştirilmiş bir sınav ataması ile yalnızca iki kuşağa atanarak, daha minimum kuşak ataması ile sonlandırılabilecektir.

Burada bahsedildiği şekilde 5. dersin 4. kuşağa atanması durumunda, 5. kuşak boş kalacağından dolayı, 6. kuşak bir satır kaydırılarak 5. kuşağa atanır. Her bir ders ataması iterasyonundan sonra bu satır kaydırma işlemi gerçekleştirilir.

Başlangıçta akış şeması verilen, önerilmekte olan uzman sistem yaklaşımının taslak kodu aşağıda gösterilmektedir.

```
do{
  Kuşağa Koyulabilecek Ders Listeleri sayısı minimum olan kuşağı seç
  seçilen kuşağın Kuşağa Koyulamayacak Dersler Listesi ile Kuşağa
      Koyulabilecek Dersler Listesinde bulunan derslerin Öğrencisi Çakışan
      Dersler Listesinde en fazla sayıda ortak dersi olan dersi seç
  seçilen dersi Sınav Programında bulunduğu yerden sil
  seçilen dersi Sınav Programında seçilen kuşağa ekle
  yalnızca dersin silindiği ve eklendiği kuşağa ait listelerini güncelle
  Sınav Programında boş kalan satır için tüm listelerin satırlarını kaydır
}while(Kuşağa Koyulabilecek Dersler Listelerinde ders varsa)
```

Şekil 4.8: Uzman Sistem Kod Taslağı.

Şekil 4.8'de aktarılan kod taslağında, ilgili güncelleme ve atama adımları önceki kısımlarda verildiğinden dolayı burada tekrarda verilmemiştir. Önerilmekte olan uzman sistemin çalıştırılmasından sonra, hiçbir kuşağa atanabilecek ekstra bir ders kalmamış olacaktır. Bu adımdan sonra katı kısıtları da sağlayacak şekilde, dersler arasında değişimler gerçekleştirilerek ya da literatürde yer alan sezgisel yöntem önerilerinde bulunan farklı teknikler sistem içerisine entegre edilerek, ceza fonksiyonu iyileştirmesi yapılabilecektir.

Bu tez çalışmasında önerilmekte olan uzman sistem yaklaşımında hiçbir şekilde ceza fonksiyonu hesaplaması dahil edilmemiş, yalnızca kural tabanlı bir sistem geliştirilmiştir. Ceza fonksiyonunun iyileştirilmesi süreci, önerilen yöntemin ikinci kısmını oluşturan genetik algoritma yöntemi ile gerçekleştirilecektir.

Önerilmekte olan uzman sistemin çalışma prensibinin daha iyi bir şekilde aktarılmasını sağlamak amacıyla verilmiş olan küçük örneğe ait sınav çizelgelemesinin adım adım çözümü, önerilen uzman sistem yaklaşımı ile gerçekleştirilmeye çalışılacaktır.

1. İterasyon: İlk iterasyonda daha önce de bahsedildiği gibi, 5. Ders 4. Kuşağa atanacaktır. Elde edilen Sınav Programı, Kuşağa Koyulamayacak Dersler Listeleri ve Kuşağa Koyulabilecek Dersler Listeleri sırasıyla aşağıda yer alan şekillerde verilmektedir.

1	
2	
3	
4	5
6	

1	4	5			
2	4	5			
2	3	4	6		
1	2	3	4	5	6
3	4	5	6		

2	3	6
1	3	6
1		
1	2	

Şekil 4.9: Örneğe Ait 1. İterasyon Sonuçları.

2. İterasyon: İlk iterasyonda yapılan atama ile, 4. kuşağa atanabilecek ders kalmamıştır. 2. iterasyonda yapılacak atama için, atanabilecek tek dersi kalan 3. kuşağa bakılacaktır. Bu kuşağa atanabilecek tek ders olan 1. ders bu kuşağa atanacak ve listeden ilk satır silinecektir. 2. İterasyonda sonra elde edilecek Sınav Programı, Kuşağa Koyulamayacak Dersler Listeleri ve Kuşağa Koyulabilecek Dersler Listeleri sırasıyla aşağıda yer alan şekillerde verilmektedir.

2	
1	3
4	5
6	

2	4	5			
1	2	3	4	5	6
1	2	3	4	5	6
3	4	5	6		

6
2

Şekil 4.10: Örneğe Ait 2. İterasyon Sonuçları.

3. İterasyon: İlk iki iterasyondan sonra geriye kalan mümkün tek atama gerçekleştirilerek, 2. ve 6. ders birlikte ilk kuşağa atanacaktır. Son durumda elde edilen Sınav Programı, Kuşağa Koyulamayacak Dersler Listeleri ve Kuşağa Koyulabilecek Dersler Listeleri de sırasıyla aşağıda yer alan şekillerde verilmektedir.

2	6
3	1
4	5

1	2	3	4	5	6
1	2	3	4	5	6
1	2	3	4	5	6

Şekil 4.11: Örneğe Ait 3. İterasyon Sonuçları.

Bu adımdan sonra, Kuşağa Koyulabilecek Dersler Listelerinde başka ders kalmadığından dolayı iterasyonlar sonlandırılır. Verilen örneğin sınav çizelgelemesi üç farklı kuşakta gerçekleştirilebilecektir.

4.2. ÖNERİLEN GENETİK ALGORİTMA YAKLAŞIMI

Tez çalışmasının bu kısmında, ilk bölümde oluşturulmuş olan kuşaklara ve ders dizilimlerine bağlı kalarak, ceza puanlarının düşürülmesi işlemi gerçekleştirilecektir. Çalışmada önerilmekte olan algoritmayla, bu ceza puanlarının düşürülmesi işlemi, derslerin bulunduğu kuşaklara müdahale edilmeden, yalnızca kuşakların dizilimleri arasında değişimler yapılarak gerçekleştirilecektir. Bu durum dolayısıyla da önerilen algoritma, en iyi çözümü elde etmeyi garantilemeden, mümkün olan en kısa sürede, en düşük ceza puanıyla, olası en iyi çizelgelemeyi oluşturmayı hedeflemektedir.

Literatürde yer alan çalışmalarda, ceza puanının hesaplanması için kullanılan matematiksel formül aşağıdaki şekildedir (Burke v.d., 2002; Abdullah ve Burke, 2006; Asmuni v.d., 2007).

$$F_C = \frac{\sum_{i=1}^{N-1} \sum_{j=i+1}^{N} c_{ij} \cdot prox(i,j)}{M} \tag{4.1}$$

$$prox(i,j) = \begin{cases} 2^{5-|t_i - t_j|} & 1 \leq |t_i - t_j| \leq 5 \\ 0 & diğer\ durumlar \end{cases} \tag{4.2}$$

Burada yer alan formülde kullanılan ifadelerin açıklamaları şu şekildedir:

M : Toplam öğrenci sayısı

N : Toplam sınav sayısı

P : Toplam sınav kuşağı sayısı

c_{ij} : Hem i. sınava ve hem de j. sınava girmesi gereken öğrenci sayısı

$C = [c_{ij}]_{N\times N}$: Çakışmalar matrisi

t_k : k. sınavın bulunduğu sınav kuşağı $(1 \leq t_k \leq P)$

$T = [t_k]_N$: Sınavların bulunduğu kuşakları gösteren vektör

Bu şekilde komplike bir formülün sezgisel bir yöntem içerisinde kullanılması sırasında, her seferinde sınavlar arası karşılaştırmalar yapmak ve gerekli atamalar sonrasında sınavlar arası çakışmalar kullanılarak ceza fonksiyonlarını hesaplamak hayli zaman alacak bir işlemdir. Örneğin bir sınavın belirli bir kuşağa atanması sonrasında ceza fonksiyonları karşılaştırılır ve ceza fonksiyonunda bir iyileşme gerçekleştirilemiyorsa aynı işlemler başka bir sınav seçilerek tekrarlanır. Bu şekilde uzun fonksiyon hesaplamalarıyla tekrarlanan işlemler, sınav atama sürecini de hayli uzatmaktadır. Bu tez çalışmasında önerilmekte olan algoritmanın çalışması sürecinde, sınavların yerlerinin değiştirilmesi işlemi gerçekleştirilmeyeceğinden dolayı, denklem (4.1) ve denklem (4.2)'de gösterilen formüller kullanılmayacaktır. Bu formüller yerine, sınav kuşakları arasında oluşturulacak olan alternatif bir formül kullanılacaktır.

Oluşturulacak ve kullanılacak olan yeni formül için P, yani sınav kuşağı sayısı boyutunda olan ve kuşaklar arası çakışan öğrenci sayılarını gösteren yeni bir matris oluşturulacaktır. Bu matrisin oluşturulması sırasında C matrisi ve T vektöründen faydalanılacaktır. Oluşturacağımız yeni matrisi $P_C = [P_{kl}]_{P\times P}$ olarak ifade edecek olursak, bu P_C matrisi elemanları aşağıdaki şekilde hesaplanacaktır.

$$P_{kl} = \sum_{i}^{i\in k}\sum_{j}^{j\in l} c_{ij} \qquad i \neq j \;\; ve \;\; k \neq l \tag{4.3}$$

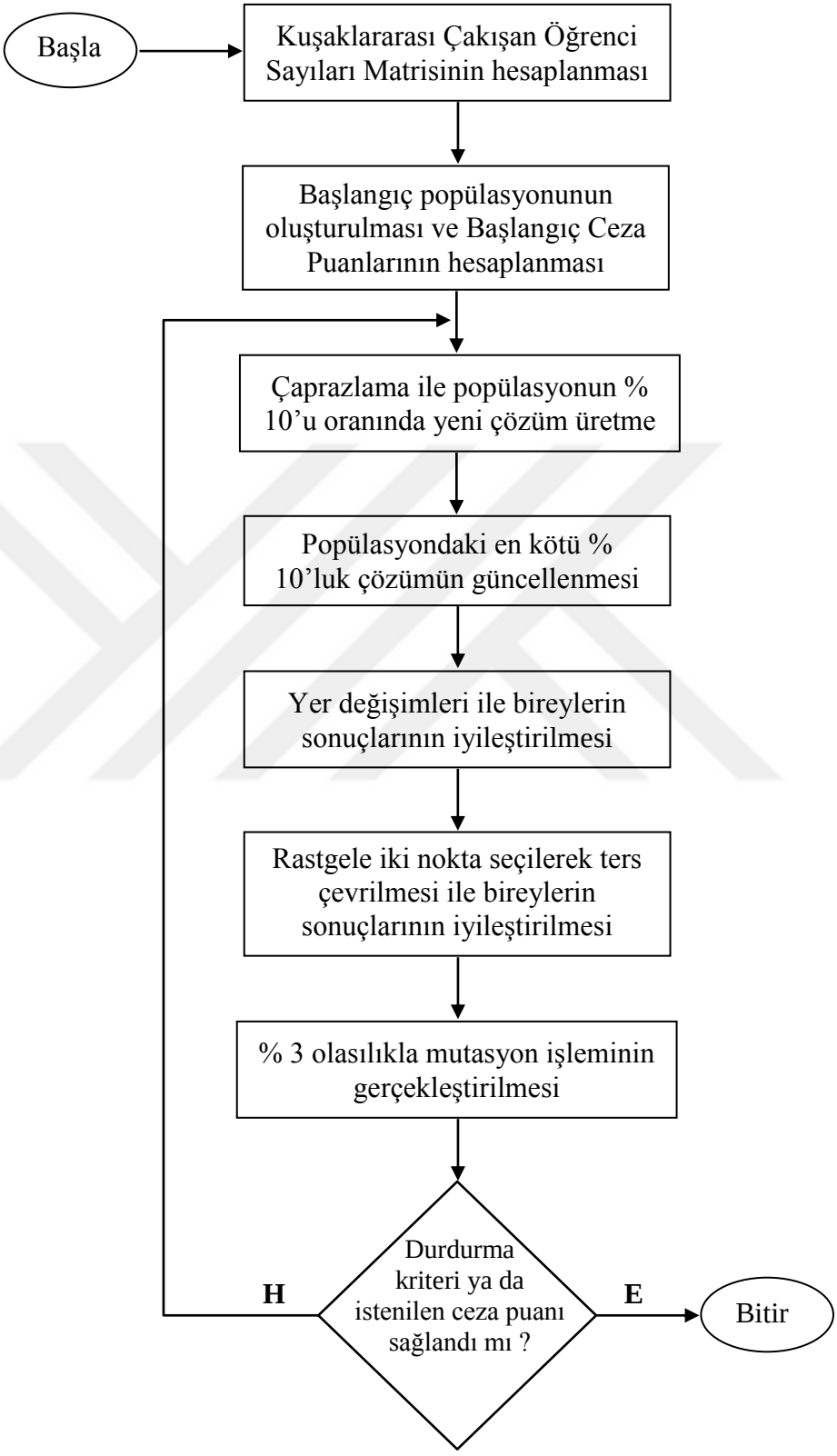
Bir sınav aynı anda iki kuşağa birden atanamayacağından ve aynı kuşağın kendi içerisindeki çakışmaları dikkate alınmayacağından dolayı, formülde $i \neq j$ ve $k \neq l$ olacaktır. Denklem (4.3)'te verilen formül, Denklem (4.2) ile belirtilen ve sınav kuşaklarının birbirine yakınlığına göre değişen ceza katsayısını içermemektedir. Bu

formül yalnızca, sınav kuşakları arasındaki çakışan öğrenci sayılarını içermektedir ve oluşturulmuş olan P_C matrisi bu haliyle, araç rotalama problemlerinde bulunan mesafe matrisi ile aynı yapıdadır. Bu formül yardımıyla, çözümler sırasında sınavlar tek tek karşılaştırılmayarak, yalnızca sınav kuşakları arası karşılaştırmalar yapılacaktır. Denklem (4.2)'de bulunan ceza katsayısı da, ceza puanının hesaplanmasında dikkate alınacaktır. Denklem (4.2)'de yer alan ceza katsayısı, bu tez çalışmasında yer alan algoritmada hesaplamaya dahil edilirken, Denklem (4.4)'te gösterildiği formatta kullanılacaktır.

Oluşturulmuş olan P_C matrisi kullanılarak ceza puanının hesaplanması için de aşağıda yer alan formülden faydalanılacaktır.

$$F_C = \frac{\sum_{i=1}^{5} \sum_{j=1}^{P-i} P_{j(j+i)} \cdot 2^{5-i}}{M} \tag{4.4}$$

Buraya kadar olan kısımda, sınav çizelgeleme problemlerinin çözümü için önerilmiş olan genetik algoritma yaklaşımının temel farklılığı aktarılmıştır. Genetik algoritma yaklaşımının diğer farklılıkları ve detaylı aktarımına geçilmeden önce, aşağıda yer alan Şekil 4.12'de, önerilmekte olan genetik algoritma yaklaşımı çözüm önerisine ait akış şeması en genel haliyle verilmektedir.



Şekil 4.12: Genetik Algoritma Çözümü Akış Şeması.

4.2.1. Önerilen Genetik Algoritmada Kullanılan Yapılar

ÜSÇP'nin çözümü için önerilen sistemin genetik algoritma bölümünde öncelikle, uzman sistem yaklaşımından elde edilen çıktılar değerlendirilerek, Kuşaklararası Çakışan Öğrenci Sayıları Matrisi oluşturulacaktır. Bu matrisle gerekli işlemlerin yapılması ve optimum kuşak diziliminin incelenmesi için bir de Kuşak Vektörü kullanılacaktır. Genetik algoritma bölümünde yalnızca ceza puanı minimizasyonu gerçekleştirileceğinden dolayı, sistemde bulunan diğer yapıların bu bölümde kullanımı gerekmemektedir. Bu bölümde kullanılacak olan iki yapı aşağıda ayrıntılı olarak açıklanmaktadır.

Kuşak Vektörü: Her bir problem için gereken toplam kuşak sayısı, P boyutunda olan ve ilk kuşak dizilimi baz alınarak, birden P'ye kadar numaralandırılmış olan bir vektördür. Çözümün ilerleyen aşamalarında bu vektör dizilimi değiştirilerek, optimum ceza paunı değeri elde edilmeye çalışılacaktır.

Kuşaklararası Çakışan Öğrenci Sayıları Matrisi: ÜSÇP'nin orijinal çözümünde bulunan sınavlar arası çakışan öğrenci sayıları matrisi (C Matrisi) kullanılarak, başlangıç Kuşak Vektörüne göre oluşturulmuş olan matristir. P_C olarak ifade edilmiş olan bu matrisin boyutu da, her bir problem için gereken toplam kuşak sayısı, P boyutunda olacaktır. Bu matrisin yapısı değiştirilmeden, Kuşak Vektörü dizilimine göre yalnızca ceza puanı hesaplaması için kullanılacaktır. Oluşturulmuş olan Kuşaklararası Çakışan Öğrenci Sayıları Matrisinin yapısı ve kullanımı, araç rotalama problemlerinde veya gezgin satıcı problemlerinde bulunan mesafe matrisiyle aynı yapıdadır.

Ceza Puanı: Önerilmekte olan sınav çizelgelemesinin kalitesinin ölçülmesi ve değerlendirilmesi için kullanılan ve bir fonksiyon sonucu olarak elde edilen puandır. Literatürde yer alan çalışmalarda Denklem (4.1) kullanılarak hesaplanmakta olan ceza puanı, bu tez çalışmasında Denklem (4.4) kullanılarak hesaplanacaktır. Denklem (4.1) ve Denklem (4.4) ile aynı sonuçlar elde edilmekle birlikte, yalnızca ifadeleri ve kullanımları farklılık göstermektedir.

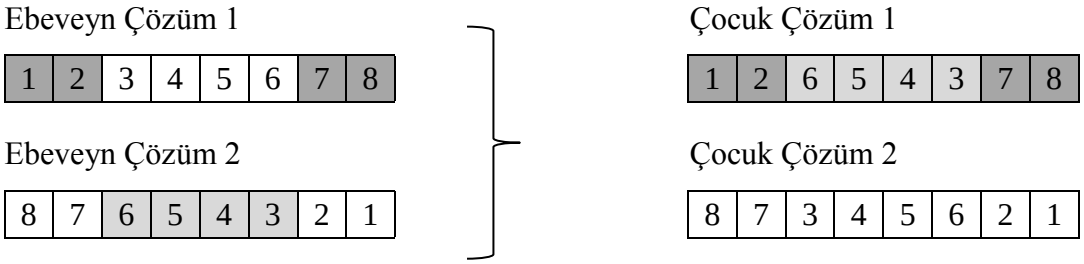
4.2.2. Önerilen Genetik Algoritma

Tez çalışmasında önerilmekte olan genetik algoritma yaklaşımının ilk aşaması olan kuşaklararası çakışan öğrenci sayıları matrisinin oluşturulması işlemi ve ceza puanı hesaplamasında kullanılacak olan formül, Başlık 4.2'nin giriş bölümünde aktarılmıştı. Bu kısımda, Şekil 4.12'de gösterilmekte olan genetik algoritma akış şeması, ayrıntılı olarak aktarılmaya çalışılacaktır.

Başlangıç popülasyonunun oluşturulması ve ceza puanlarının hesaplanmasından sonra, rastgele seçimlerle toplam popülasyonun % 10'u oranında ikili çaprazlamalar gerçekleştirilmiştir. Bu ikili çaprazlama işlemleri sırasında rastgele seçim işlemleri gerçekleştirilirken, rulet tekerleği seçim tekniği kullanılmıştır. Rulet tekerleği seçim tekniği sayesinde, ceza puanı iyi olan çözümlerin çaprazlama işlemi için seçilme ihtimalleri artmaktadır. Rulet tekerleği seçim tekniği ile her bir çözümün seçilme olasılığı (p_i) için aşağıda yer alan formül kullanılmıştır.

$$p_i = \frac{F_{C_i}}{\sum F_{C_j}} \tag{4.5}$$

Rulet tekerleği seçim tekniği kullanılarak iki adet çözüm seçildikten sonra, çift nokta çaprazlama tekniği ile yeni çözümler üretilir. Üretilen çözümlerden düşük ceza puanına sahip olan çözüm saklanır. Bu işlem, popülasyonun % 10'u kadar yeni çözüm üretilene kadar devam ettirilir. Sonrasında, üretilmiş olan yeni çözümler, popülasyonda yer alan çözümlerin en kötü % 10'u ile değiştirilir. 8 adet kuşak içeren bir probleme ait çift nokta çaprazlama ile yeni çözümler üretme işlemi, örnek olarak aşağıda yer alan şekilde gösterilmektedir.



Şekil 4.13: Çift Nokta Çaprazlama Tekniği ile Yeni Çözümler Üretme Örneği.

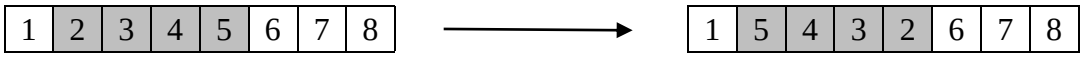
Burada kullanılan çözümlerde, klasik genetik algoritmalar çözümlerinde kullanılan 0–1 dizilimlerinden farklı olarak, kuşak sayısı adedince gen içeren ve her bir kuşağın kendi sayısal değeriyle ifade edildiği dizilimler kullanılmaktadır. Elde edilen çocuk çözümlerden, daha iyi ceza puanına sahip olanı yeni nesle aktarılır ve tüm çaprazlama işlemleri tamamlandıktan sonra diğer adıma geçilir.

Çözümün bir sonraki adımında, elde edilen çözümlerin iyileşme hızlarının arttırılması amacıyla, kuşak dizilimlerinde ikili olarak yer değişimleri gerçekleştirilerek ceza puanları düşürülmeye ve popülasyon iyileştirilmeye çalışılır. Ceza puanı düşürmeyen kuşak yer değiştirmelerine izin verilmez. Aşağıda yer alan şekilde, 8 kuşak içeren bir probleme ait ikili yer değiştirme işlemi örnek olarak verilmektedir.



Şekil 4.14: İkili Yer Değiştirme İşlemi ile Çözümlerin İyileştirilmesi Örneği.

Kuşakların ikili olarak yer değiştirilmesi işleminden sonra da benzer şekilde, iyileşmeyi hızlandırmak amacıyla dizilimde yer alan rastgele iki nokta seçilerek, bu iki nokta arasındaki dizilim tamamen ters çevrilir. Bu adımda da, ceza puanı düşürmeyen ters çevirme işlemlerine izin verilmez. 8 adet kuşak içeren bir probleme ait ters çevirme işlemi, örnek olarak aşağıda yer alan şekilde gösterilmektedir.



Şekil 4.15: Ters Çevirme İşlemi ile Çözümlerin İyileştirilmesi Örneği.

Son olarak da, kuşak dizilimleri % 3 olasılıkla mutasyon işlemine tabi tutulur. 0–1 dizilimleri yerine sıralı sayılar kullanıldığından ve hiçbir sayı tekrar etmediğinden dolayı mutasyon işlemi, yer değiştirme işlemi ile benzer bir işlem haline gelmektedir. Bu iki uygulama arasındaki farklılık; yer değiştirme işlemi, iyileştirme olması halinde gerçekleştirilirken, mutasyon işleminin zorunlu bir şekilde gerçekleştirilmesidir. Mutasyon işleminin algoritmaya dahil edilmesinin temel nedeni; yerel minimumlara ve aynı çözümlere takılmayı engellemektir. Popülasyon tabanlı bir çözüm gerçekleştirildiğinden dolayı, mutasyon işlemi çözümleri tamamen kötüleştirmemektedir. Yalnızca belirli sayıda çözümler kötüleştirilmektedir.

Burada aktarılan genetik algoritma işlemlerine, durdurma kriteri sağlana kadar ya da istenilen ceza puanına ulaşılana kadar devam edilir. Literatürde yer alan uygulamalarda durdurma kriteri olarak, genellikle, ceza puanının belirli sayıda iterasyonda iyileştirilememesi kullanılmaktadır. Bu tez çalışmasında gerçekleştirilen uygulamalarda da durdurma kriteri olarak ceza puanı iyileştirmesinin devam etmemesi kullanılmıştır.

Önerilmekte olan genetik algoritma yaklaşımına ait, Şekil 4.12'de verilmiş olan akış şemasının taslak kodu aşağıda gösterilmektedir.

```
başlangıç popülasyonu oluşturulur ve ceza puanları hesaplanır
do{
for(popülasyonun % 10'u kadar)
  rastgele iki adet çözüm seçilir
  seçilen çözümler çaprazlanır ve yeni çözümler üretilir
  üretilen çözümler karşılaştırılarak iyi olanı saklanır
endfor
for(popülasyonun % 10'u kadar)
  popülasyondaki en kötü % 10'luk çözüm çaprazlamalarla elde edilen çözümlerle değiştirilir
endfor
for(popülasyondaki tüm çözümlerin sonuna kadar)
  rastgele iki kuşak seçilerek yerleri değiştirilir
  yeni kuşak diziliminin ceza puanı hesaplanır
  ceza puanı mevcut çözümden daha kötü ise işlem ters çevrilir
endfor
for(popülasyondaki tüm çözümlerin sonuna kadar)
  rastgele iki kuşak seçilerek aradaki dizilim ters çevrilir
  yeni kuşak diziliminin ceza puanı hesaplanır
  ceza puanı mevcut çözümden daha kötü ise işlem ters çevrilir
endfor
for(popülasyondaki tüm çözümlerin sonuna kadar)
  rastgele bir kuşağa % 3 olasılıkla mutasyon işlemi gerçekleştirilir
endfor
}while(ceza puanı iyileştirmesi yapılabildiği sürece)
en iyi çözümü veren kuşak dizilimi ve ceza puanı yazdırılır
```

Şekil 4.16: Genetik Algoritma Yaklaşımı Kod Taslağı.

Daha önce de bahsedildiği gibi, Şekil 4.16'da kod taslağı aktarılan genetik algoritma yaklaşımı, yalnızca kuşak değişimlerini içermekte ve ders değişimlerini kullanmamaktadır. Bu durum dolayısıyla da yaklaşım, literatürde yer alan en iyi çözüme ulaşmayı ya da iyileştirmeyi garantilememektedir. Önerilen yaklaşım, özellikle büyük boyutlu yani çok sayıda sınav içeren problemlerin, hızlı bir şekilde en iyiye yakın çözümlerinin oluşturulmasını hedeflemektedir.

5. UYGULAMA

Bu tez çalışmasında önerilmekte olan uzman sistem ve genetik algoritma tekniğinin birlikte kullanımını içeren yaklaşım, TÜKVS'ne uygulanmıştır. Uygulamanın TÜKVS ile gerçekleştirilmesinin nedeni, literatürde de daha önce sıklıkla kullanılan bir veri seti olması, gerçek hayat problemlerini içermesi ve hem yoğunluk hem de problem boyutu açısından birbirinden farklı ve çok çeşitli problemler içeriyor olmasından dolayıdır. Başlık 3.2.1 altında ayrıntılı olarak aktarılmış olan TÜKVS, 81 adet sınav içeren problemden 2419 adet sınav içeren probleme kadar, toplam 13 adet sınav çizelgeleme problemi içermektedir. Bunlar da önerilmekte olan algoritmanın hem farklı açılardan değerlendirilmesini, hem de literatürde önerilmekte olan farklı algoritma sonuçları ile karşılaştırılmasını sağlamaktadır.

TÜKVS uygulaması, 2,6 GHz Intel(R) Core(TM) i7–4720HQ işlemci ve 16 GB RAM'e sahip Windows 8 işletim sistemi içeren bir bilgisayarda gerçekleştirilmiştir. Önerilmekte olan yaklaşım ile elde edilmiş olan sonuçlar, Başlık 3.2.1 altında aktarılmış olan literatür taraması sonuçlarında tespit edilen en iyi sonuçlarla da karşılaştırılmıştır. Literatürde bulunan ve TÜKVS ile gerçekleştirilen çalışmalar incelendiğinde, hem olabilecek minimum kuşak sayısının tespit edilmesi, hem de belirli kuşak sayılarında ceza fonksiyonu puanlarının minimize edilmesi olarak iki farklı şekilde ele alındığı tespit edilmiştir. Bundan dolayı, bu tez çalışmasında her iki uygulama şekli de ele alınarak sonuçlar raporlanmıştır.

5.1. SINAV KUŞAĞI SAYISININ AZALTILMASI

Sınav kuşağı sayısının azaltılması uygulaması, sınavların olabildiğince az sayıda kuşakta gerçekleştirilmesini içermektedir. Bu sınav kuşağı sayısı azaltılma işlemi gerçekleştirilirken dikkat edilmesi gereken en önemli durum, hiçbir öğrencinin aynı sınav kuşağında birden fazla sınavının bulunmamasıdır. Bu kısıt dışındaki diğer kısıtlar, problemin tipine ve uygulama gerçekleştirilen üniversitenin ya da fakültenin imkanlarına göre esnetilebilmekte ya da değiştirilebilmektedir. Örneğin, sınav kuşağı

sayısının azaltılması, sınavların gerçekleştirileceği salon sayısı ya da sınavları gerçekleştirecek olan görevli sayısına göre farklı şekillerde gerçekleştirilebilecektir. Bu gibi durumlar, algoritma içerisine yerleştirilecek küçük bir kod ile kolaylıkla dahil edilerek çözülebilecektir. TÜKVS'nde yer alan sınav çizelgeleme problemlerinin, sınav kuşağı sayısının azaltılması olarak ele alınması durumunda, herhangi bir görevli ya da sınav salonu kısıtı bulunmamaktadır. Bundan dolayı, gerçekleştirilen uygulamalar, yalnızca öğrencilerin aynı sınav kuşağında birden fazla sınavının bulunmaması olarak ele alınmıştır.

Uygulamalar neticesinde elde edilen sonuçlar, aşağıda yer alan tabloda gösterilmektedir. Her bir problemin çözüm zamanı da saniye cinsinden tabloda belirtilmektedir.

Tablo 5.1: TÜKVS için Sınav Kuşağı Sayısının Azaltılması Uygulama Sonuçları.

Problem Kodu	**Sınav Sayısı**	**Uygulama ile Elde Edilen Sonuçlar**	
		Sınav Kuşağı Sayısı	**Süre (Saniye)**
CAR–F–92	543	32	123
CAR–S–91	682	32	292
EAR–F–83	181	24	1
HEC–S–92	81	18	0,1
KFU–S–93	486	20	61
LSE–F–91	381	18	29
PUR–S–93	2419	38	17851
RYE–F–93	487	23	74
STA–F–83	139	**16**	0,5
TRE–S–92	261	22	6
UTA–S–93	622	35	193
UTE–S–92	184	10	1,4
YOR–F–83	190	**22**	1,3

TÜKVS'nde yer alan 13 adet problemden STA–F–83 ve YOR–F–83 problemleri hariç diğer problemler için, literatürde bulunmuş olan ve algoritmaların karşılaştırılmasına imkan sağlayan kuşak sayılarına eşit sayıda ya da bunlardan daha küçük sayıda sınav kuşağı elde edilmiştir. Problemlerin çözümü için gereken çalışma zamanları, literatürde yer alan sonuçlarla rekabet edebilir niteliktedir. Özellikle, PUR–S–93 problemi gibi çok büyük miktarda sınav sayılarına sahip problemlerin, genellikle literatürde yer alan uygulamalarda çözülmediği düşünülürse, önerilmekte olan algoritma çözümü önemli bir avantaj teşkil etmektedir. Önerilmekte olan algoritmada, her iterasyonda bir sınav kuşağına atanmış olan dersler, listelerden kaldırıldığından dolayı, algoritma her bir iterasyonda daha da hızlı bir şekilde çözüme gidebilmektedir. Bu durumla birlikte, çok yüksek sayıda sınav miktarına sahip problemlerin de reel bir zaman diliminde çözümleri elde edilebilmektedir.

5.2. CEZA FONKSİYONU MİNİMİZASYONU

Literatürde yer alan çok fazla sayıda sezgisel teknik önerisi bulunmaktadır. Bunlardan temel ve en bilinen olanları, tez çalışmasının ikinci bölümünde ayrıntılı olarak aktarılmıştır. Sınav çizelgeleme problemlerinin çözümü için önerilmiş olan sezgisel tekniklerin karşılaştırılabilmesi için, test problemlerinde sabit sınav kuşağı sayıları kullanılmaktadır. Bu sabit sınav kuşağı sayılarına göre, geliştirilen sezgisel teknikler ceza puanlarını minimize etmeye odaklanmaktadır. Böylece ceza puanları kullanılarak, önerilmekte olan sezgisel tekniklerin etkinlikleri karşılaştırılabilmektedir. Bu şekilde gerçekleştirilen çalışmaların asıl amacı her ne kadar en iyi yaklaşımın elde edilmesi olsa da, 2009 yılında Asmuni ve arkadaşlarının da (Asmuni v.d., 2009) belirttiği gibi, literatürde yer alan çalışmaların çoğunda en iyi yaklaşımın geliştirilmesi yerine en iyi sonuçlar iyileştirilmeye çalışılmış ve bunun için çok daha fazla çaba harcanmıştır.

Bu bölümde yer alan ikinci kısımda, bu başlık altında önerilmekte olan yaklaşımın, literatürde yer alan mevcut sezgisel tekniklerle de karşılaştırılabilirliğinin sağlanması adına, sabit sınav kuşağı sayılarına göre ceza puanı minimizasyonu işlemi gerçekleştirilecektir. Bunun için, literatürde yer alan uygulamalarda tercih edilen sabit sınav kuşağı sayıları baz alınmıştır.

Gerçekleştirilmiş olan sezgisel teknik uygulamasıyla elde edilen sınav çizelgelemelerine ait ceza puanları sonuçları aşağıda yer alan Tablo 5.2'de ayrıntılı olarak verilmektedir.

Tablo 5.2: TÜKVS için Ceza Puanı Minimizasyonu Uygulama Sonuçları.

Problem Kodu	Sınav Kuşağı Sayısı	Literatür En İyi Ceza Puanı	Uygulama Sonuçları		
			Kuşak Sayısı	En İyi Ceza Puanı	Fark (%)
CAR–F–92	32	3,77	32	4,15	10
CAR–S–91	35	4,35	35	5,56	28
EAR–F–83	24	29,7	24	33,89	14
HEC–S–92	18	10	18	10,18	2
KFU–S–93	20	12,86	20	13,94	8
LSE–F–91	18	9,11	18	10,30	13
PUR–S–93	42	4,47	42	5,21	17
RYE–F–93	23	8,05	23	10,04	25
STA–F–83	**13**	136,43	**16**	94,60	—
TRE–S–92	23	7,69	23	**5,26**	**– 32**
UTA–S–93	35	3,1	35	**2,02**	**– 35**
UTE–S–92	10	18,42	10	**16,53**	**– 10**
YOR–F–83	**21**	34,64	**22**	26,92	—
TOPLAM		278,82		234,45	
TOPLAM[5*]		111,52		112,93	1
ORTALAMA[*]		10,78		10,64	– 1

Daha önce de belirtildiği gibi, TÜKVS'nde yer alan 13 adet problemden STA–F–83 ve YOR–F–83 problemleri için, algoritmaların karşılaştırılmasına imkan sağlayan kuşak sayılarına eşit sayıda ya da bunlardan daha küçük sayıda sınav kuşağı elde edilemediği belirtilmişti. Bu iki problemde daha küçük kuşak sayıları elde

[*] STA–F–83 ve YOR–F–83 problemleri hariç toplam/ortalama değeri.

edilemediğinden dolayı, elde edilmiş olan kuşak sayıları üzerinden ceza puanı minimizasyonu işlemleri gerçekleştirilmiştir. Bu iki problem için elde edilmiş olan ceza puanları, bilinen en iyi ceza puanlarından daha küçüktür. Fakat bu farklılığın, kuşak sayısının fazla olmasından kaynaklandığı da aşikardır. Problemlerin bu çözümleri itibariyle bir karşılaştırmadan bahsedilmesi mümkün değildir.

Diğer problemlerde ise kuşak sayısı mevcut çözümlerle aynı sayıda elde edilmiş olan çözümler için, aynı kuşak sayılarıyla ceza puanı minimizasyon işlemleri gerçekleştirilmiştir. Kuşak sayısı mevcut çözümlerden daha küçük olarak elde edilmiş olan çözümlerde ise, kuşaklarda yer alan sınav sayıları ve öğrenci sayıları yakın olacak şekilde bir düzenlemeye gidilerek, ceza puanı minimizasyonu işlemlerine devam edilmiştir. Bu 11 adet problemden 8 adedinde, literatürde yer alan bilinen en iyi sonuçlara çok yakın sonuçlar elde edilmiştir. Literatürde yer alan diğer sonuçlar da göz önüne alındığında, elde edilmiş olan bu sonuçların rekabet edebilir ve kabul edilebilir sonuçlar olduğu görülmektedir.

Son olarak, TRE–S–92, UTA–S–93 ve UTE–S–92 problemlerinde ise sırasıyla % 32, % 35 ve % 10 oranlarında, bilinen en iyi sonuçlar üzerinde iyileştirmeler gerçekleştirilmiştir. Gerçekleştirilmiş olan bu iyileştirme oranları ve literatürde yer alan diğer sonuçlar göz önüne alındığında, % 30 un üzerinde iyileştirmenin, ciddi anlamda fark yaratan bir iyileştirme olduğu da görülmektedir.

Elde edilmiş olan tüm sonuçlar birlikte değerlendirildiğinde, önerilmekte olan iki aşamalı sınav çizelgeleme algoritmasının, sınav çizelgeleme problemlerinin çözümü için etkili ve uygulanabilir sonuçlar ürettiği söylenebilecektir. Önerilmekte olan yöntem, her iki aşamasında da farklı tekniklerle birlikte entegre edilerek geliştirilebilecek ve daha iyi sonuçlar üreten sistemler elde edilebilecektir. Özellikle, yöntemin ikinci aşaması olan ve genetik algoritma tekniği kullanılarak geliştirilen bölümde, literatürde yer alan ve etkili sonuçlar üretebilen diğer tüm sezgisel tekniklerle uygulamalar gerçekleştirilerek, yöntemlerin etkinlikleri de değerlendirilebilecektir.

TARTIŞMA VE SONUÇ

Optimizasyon problemleri ve karar verme, insanlığın tarih boyunca her alanda karşılaştığı problemlerden olmuştur. Bilişim sistemlerinin olmadığı ve daha yalın sistemlerin bulunduğu zamanlarda, karar verme daha basit sorunlar içermekteydi ve daha basit yollarla manuel olarak çözülebilmekteydi. Zamanla sistemler gelişmeye başlamış ve gerekliliklerle de birlikte, iç içe geçen karmaşık ve yoğun sistemler, hayatımızın her alanında etkin bir hale gelmeye başlamıştır. Özellikle bilişim sistemlerinin ve teknolojinin de gelişmesiyle birlikte, insan–makine etkileşimi de sistemler içerisinde önemli bir etken haline gelmiştir. İnsan–makine etkileşiminin en önemli kısımlarından biri de insan–bilgisayar etkileşimidir. Günümüzde birçok sistem, bilgi teknolojileri yardımıyla daha verimli hale getirilebilmektedir. Bunun yanında, bilgi teknolojileri kullanılarak, optimizasyon problemleri ve karar verme işlemlerinin neredeyse tamamı gerçekleştirilebilmektedir. Bilgi teknolojileri kullanılarak gerçekleştirilen çözümlerde, süreçler de hayli hızlı ve verimli bir şekilde tamamlanabilmektedir.

Sistemlerin en önemli sorunlarından birisi de çizelgeleme problemidir. Çizelgeleme problemleri çok çeşitli şekillerde olduğu gibi, çok fazla bileşen de içermektedir. Günümüz karmaşık sistemlerinin hemen hemen hepsinde çizelgeleme gerektiren eylemler bulunmakta ve bu eylemler sistemlerin önemli bir bileşeni olmaktadır. Sistemlerin de ihtiyacı olan ve literatürde de çalışmalar bulunan çizelgeleme problemlerinin en önemlilerinden bazıları, üretim hattı çizelgeleme, havayolu çizelgeleme, hastane çizelgeleme, vardiya çizelgeme, araç çizelgeleme ve sınav çizelgeleme problemleridir. Bu problemlerden bazıları tek boyutlu çizelgelemeleri içerirken, bazıları da çok çeşitli boyutları ve bileşenleri içeren karmaşık çizelgeleme problemleridir. Özellikle karmaşık ve çok fazla değişken içeren sistemlerin çizelgeleme işlemi, bilgi sistemleri kullanılarak dahi saatler sürebilmektedir. Bu karmaşıklık ve zorlukları dolayısıyla da çizelgeleme problemlerinin çözümü için literatürde çok çeşitli yöntemler geliştirilmiştir.

Bu tez çalışmasında da çizelgeleme problemlerinden biri olan üniversite sınav çizelgeleme problemi ele alınmış ve problemin çözümü üzerinde durulmuştur. Önerilmiş olan çözüm yöntemi yaklaşımı iki aşamada gerçekleştirilmiştir. İlk aşamada, uzman sistem yaklaşımı kullanılarak kuşaklara atanacak olan dersler tespit edilmiş ve sonrasında gerçekleştirilen ikinci aşamada da, genetik algoritma yaklaşımı kullanılarak sınav çizelgelemelerine ait ceza puanlarının minimizasyonu işlemi gerçekleştirilmeye çalışılmıştır. Bu amaçla; öncelikle, tez çalışmasının ilk bölümünde, uzman sistemler incelenmiştir. İkinci bölümde gerek uygun sezgisel tekniğin tespit edilebilmesi, gerekse de farklı sezgisel tekniklerin önemli noktalarının görülebilmesi adına, literatürde yer alan ve en sık tercih edilen sezgisel teknikler incelenmiştir. Üçüncü bölümde de tez çalışmasına konu olan üniversite sınav çizelgeleme problemi incelenerek, literatürde yer alan çalışmalar değerlendirilmiştir. Önerilmekte olan üniversite sınav çizelgeleme yaklaşımı da dördüncü bölümde, iki kısım altında ayrıntılı olarak aktarılmıştır. Beşinci ve son bölümde ise önerilmiş olan çözüm yaklaşımının, Toronto Üniversitesi Kıyaslama Veri Setinde yer alan 13 adet probleme uygulanmasıyla elde edilmiş olan sonuçlara ve değerlendirmelere yer verilmiştir.

Özellikle gelişen teknolojilerle birlikte, uzman sistemlerin önemi ve kullanımı hızla artmıştır. Bir uzmanın karar verme süreci göz önüne alındığında, uzun süren değerlendirme ve analiz aşamaları, süreci oldukça yavaşlatmakta ve sistemi aksatabilmektedir. Bunun yerine, bilgi teknolojileri kullanılarak geliştirilmiş bir uzman sistemin devreye sokulmasıyla, aksayan ve yavaşlayan sistemler hızlandırılabilmektedir. Örneğin bankaların kredi verme sistemi ele alınacak olursa; uzmanların, başvuru dosyasında yer alan evrakların tamamını ve başvuranın kişisel bilgilerini değerlendirerek bir kredi kararı vermesi, hayli uzun zaman alabilecektir. Zira verilerin ve kişisel bilgilerin, hem ayrı ayrı hem de birlikte analiz edilmesi gerekmektedir. Bu şekilde bir kişisel değerlendirme sisteminin yerine, uzmanlarla görüşülerek elde edilecek olan kural tabanı ve bilgi tabanı sayesinde, bilgi teknolojileri de kullanılarak, bir kredi kararı çok daha hızlı ve hatta anlık bir şekilde verilebilecektir.

Günümüzde özel uzmanlık gerektiren, neredeyse her alanda uygulanmakta olan uzman sistemlerin en önemli avantajlarından birisi, kişisel yargılardan arındırılmış kararlar veriyor olmasıdır. Uzman sistemlerin bu özelliği bir avantaj olduğu gibi aynı zamanda da bir dezavantajdır. Bu avantajın bir dezavantaja dönüşmesi, kural tabanında yer almayan ve kişisel tecrübelerle karar verilebilecek, sisteme tanıtılmamış durumlarda, herhangi bir sonuç elde edilememesi ya da yanlış sonuçlar elde edilebilmesidir. Avantaj ve dezavantajları birinci bölümde ayrıntılı olarak değerlendirilmiş olan uzman sistemler, özellikle uzun vadede ciddi kazançlar sağladığından dolayı tercih edilen bir sistem halini almıştır.

Sezgisel yöntemler de gelişen teknolojiyle birlikte kullanımı artan yöntemlerdendir. Konkav çözüm uzayına sahip ve bundan dolayı da çok fazla sayıda yerel minimuma sahip olan NP–zor ve NP–tam problemlerin çözümünde tercih edilen sezgisel yöntemler, genel olarak doğadan esinlenilerek geliştirilmiştir. İnsanların gözlemlerinden ve incelemelerinden elde etmiş oldukları tecrübeler göstermektedir ki; doğada yer alan düzende, canlılar ve sistemler her zaman mükemmele ve iyiye yönelme eğilimindedirler. Literatürde yer alan, doğadan esinlenilerek geliştirilmiş çok fazla sayıda sezgisel yöntem bulunmaktadır. Geliştirilmiş olan bu yöntemlerin çoğu, uygulamalarda çok fazla kullanım görebilmiş olan yöntemler değildir. Literatürde en sık tercih edilen ve uygulamalarda da sıklıkla karşılaşılan yöntemler, doğal düzende de birçoğumuzun bildiği sistemlerdir. Sezgisel yöntemler de, bu sistemlerin matematiksel olarak formülize edilerek uygulanabilir hale getirilmesi neticesinde elde edilmektedir. En çok tercih edilen sezgisel yöntemler olan genetik algoritmalar, karınca kolonisi optimizasyonu, yapay arı kolonisi, tabu arama algoritması, tavlama benzetimi ve parçacık sürü optimizasyonu, tez çalışmasının ikinci bölümünde detaylı bir şekilde aktarılmaya çalışılmıştır. Hem sezgisel yöntemlerin temelini oluşturması, hem de gerçekleştirilmiş olan çalışmaların en iyi sonuçlarının genellikle bu yöntemden elde edilmesi dolayısıyla, tez çalışmasında kullanılan sezgisel yöntem olarak genetik algoritmalar tercih edilmiştir. Tercih edilen genetik algoritmada kullanılan yapılar, operatörler ve uygulama adımları da dördüncü kısımda ayrıntılı olarak aktarılmıştır.

Sınav çizelgeleme problemleri iki ana kısımdan oluşmaktadır; sınavların çizelgelenmesi ve gözetmenlerin çizelgelenmesi. Her iki kısımda da çizelgelemenin gerçekleştirildiği üniversiteye göre farklı kısıtlar olsa da, problemler genel olarak benzer kısıtlar altında değerlendirilmektedirler. En temel olarak kullanılan katı kısıtlar şu şekildedir:

- Bir öğrenci aynı kuşakta birden fazla sınava giremez.
- Her bir sınav tek bir kuşakta gerçekleştirilmelidir.
- Bir gözetmen aynı anda iki salona atanamaz.
- Kapasite bulunan problemlerde kapasite kısıtları aşılamaz.
- Özel salon gerektiren sınavlar bu salonlarda gerçekleştirilmelidirler.

Burada aktarılmış olan katı kısıtlar, tüm problemlerde kullanılmakta olan temel kısıtlardır. Bu kısıtların yanında, her bir üniversitenin mutlaka olmasını istediği bir takım farklı katı kısıtlar da bulunabilmektedir. Bu katı kısıtların yanında, uygulamalarda kullanılan en temel hafif kısıtlar da şu şekildedir:

- Öğrencilerin birbirlerini takip eden sınavları arasında olabildiğince fazla çalışma zamanları bırakılmalı.
- Öğrencilerin bir günde gireceği sınav sayısı olabildiğince azaltılmalı.
- Birbirini takip eden sınavları olan öğrenci sayısı olabildiğince azaltılmalı.
- Zor olan sınavlar aynı güne ya da birbirini takip eden günlere yerleştirilmemeli.

Burada aktarılmış olan hafif kısıtlar da, tüm problemlerde kullanılmakta olan temel kısıtlardır. Bu kısıtların yanında, her bir üniversitenin mutlaka olmasını istediği bir takım farklı ve özel nitelikte hafif kısıtlar da bulunabilmektedir. Esnek bir şekilde tasarlanmış olan başarılı bir algoritmaya, farklı kısıtlar küçük değişikliklerle rahatlıkla eklenebilecek ve istenen sonuçlar elde edilebilecektir. Bu çalışmada gerçekleştirilmiş olan algoritma da, bu durum dolayısıyla esnek bir şekilde tasarlanmaya çalışılmış ve hiçbir değişiklik gerçekleştirilmeden TÜKVS'nde yer alan problemlerin tamamına uygulanmıştır. Farklı kısıtlara sahip veya özel nitelikli problemler için gerçekleştirilecek olan çözümlerde tasarlanmış olan algoritmalarda küçük değişiklikler gerçekleştirilerek, başarılı sonuçlar elde edilebilecektir.

Bu tez çalışmasında, tüm bu bilgilerden ve kazanımlardan yola çıkılarak, üniversite sınav çizelgeleme probleminin çözümü için yapay zeka tekniklerinden oluşan bir yaklaşım önerisinde bulunulmuştur. Önerilmiş olan yaklaşımın ilk aşamasında, bilgi ve kural tabanlı bir yapıyla desteklenen uzman sistem kullanılmıştır. İkinci aşamada ise, ilk aşamada elde edilen kuşak yerleşimlerine göre ceza puanları, genetik algoritmalar kullanılarak olabildiğince azaltılmaya çalışılmıştır. Önerilen yöntemin tüm adımları, dördüncü bölümde örneklerle birlikte detaylı bir şekilde aktarılmaya çalışılmıştır.

Önerilen üniversite sınav çizelgeleme yaklaşımının uygulamasının gerçekleştirildiği beşinci bölümde ise, etkili sonuçlar elde edilmiş ve tablolar halinde listelenmiştir. Literatürde de bu alanda iki farklı uygulama bulunduğundan dolayı, beşinci bölümde raporlanmış olan uygulamalar iki farklı şekilde gerçekleştirilmiştir; sınav kuşağı sayısının azaltılması ve ceza fonksiyonu minimizasyonu.

Sınav kuşağı sayısının azaltılması aşamasında, iki adet problem haricindeki problemlerde, en iyi sonuçlar ya da en iyiye yakın sonuçlar elde edilmiştir. Tez çalışmasının bu aşamasında elde edilen en önemli sonuçlardan biri ise, uygulamaların çok hızlı bir şekilde sonuçlanmış olmasıdır. TÜKVS problemlerinin neredeyse yarısının çözümü bir saniye civarındaki sürelerde elde edilebilmiştir. Diğer problemlerin çözümlerinin de, literatürde yer alan ve bilinen sürelerden daha kısa bir şekilde gerçekleştirildiği gözlemlenmiştir.

Ceza fonksiyonu minimizasyonu aşamasında da üç adet problem, bilinen en iyi sonuç değerlerinin altında ceza puanları elde edilerek çözülmüştür. Diğer problemlerden sekiz adedinde de, bilinen en iyi sonuçlara yakın ve rekabet edebilen sonuçlar elde edilmiştir. Özellikle, literatürde yer alan diğer tüm sonuçlarla birlikte değerlendirildiğinde, elde edilen bu sonuçların kabul edilebilir, etkili ve rekabet edebilir sonuçlar olduğu gözlemlenebilecektir. Veri setinde yer alan diğer iki adet problemde ise önerilen yaklaşımla, bilinen en iyi sonuçların elde edildiği kuşak sayılarına ulaşılamadığından dolayı, karşılaştırılabilir sonuçlar elde edilememiştir. Önerilmekte olan yaklaşım, özellikle yüksek sayıda sınav içeren problemlerin kısa sürede çözülmesinde etkin bir şekilde çalışabilmektedir.

Önerilmekte olan yaklaşım, farklı kısıt ve özellikteki problemlerde kullanılabileceği gibi, algoritmada gerçekleştirilecek küçük değişikliklerle gözetmen çizelgeleme olarak da uygulanabilecektir. Bunun yanında, algoritmanın ikinci bölümü olan genetik algoritma yaklaşımı yerine farklı yöntemler kullanılarak daha iyi sonuçlar da elde edilebilecektir. Burada gerçekleştirilmiş olan uygulamalarda etkili ve verimli sonuçlar elde edilebilmiş olan yaklaşım, farklı yöntemlerle de entegre bir şekilde çalıştırılarak, daha etkili bir hale getirilebilecek ve farklı problem çeşitlerine de uygulanabilecektir.

KAYNAKLAR

Abdelhalim E. A., El Khayat, G. A. — "A Utilization-based Genetic Algorithm for Solving the University Timetabling Problem (UGA)", **Alexandria Engineering Journal**, 2016. DOI: http://dx.doi.org/10.1016/j.aej.2016.02.017

Abdul Rahim, S. K. N., Bargiela, A., Qu, R. — "Granular Modelling Of Exam To Slot Allocation", **ECMS**, 2009, 861–866.

Abdul Rahim, S. K. N., Bargiela, A., Qu, R. — "Domain transformation approach to deterministic optimization of examination timetables", **Artificial Intelligence Research**, C.2, No: 1, 2013a, 122–138.

Abdul Rahim, S. K. N., Bargiela, A., Qu, R. — "Hill climbing versus genetic algorithm optimization in solving the examination timetabling problem", **2nd International Conference on Operations Research and Enterprise Systems**, Barcelona, Spain, 2013, 43–52.

Abdul Rahman, S., Bargiela, A., Burke, E. K., Özcan, E., McCollum, B., — "Linear Combination of Heuristic Orderings in Constructing Examination Timetable", **European Journal of Operational Research**, 2011, 1–17.

Abdul Rahman, S., Burke, E. K., Bargiela, A., McCollum, B., Özcan, E. — "A constructive approach to examination timetabling based on adaptive decomposition and ordering", **Annals of Operations Research**, C.218, No: 1, 2014a, 3–21.

Abdul Rahman, S., Bargiela, A., Burke, E. K., Özcan, E., McCollum, B., McMullan, P. — "Adaptive linear combination of heuristic orderings in constructing examination timetables", **European Journal of Operational Research**, C.232, No: 2, 2014b, 287–297.

Abdullah, S., Alzaqebah, M. — "A hybrid self-adaptive bees algorithm for examination timetabling problems", **Applied Soft Computing**, C.13, No: 8, 2013, 3608–3620.

Abdullah, S., Burke, E. K. — "A Multi-Start Very Large Neighbourhood Search Approach with Local Search Methods for Examination Timetabling", **ICAPS**, 2006, 334–337.

Abdullah, S., Ahmadi, S., Burke, E. K., Dror, M.	"Investigating Ahuja–Orlin's large neighbourhood search approach for examination timetabling", **OR Spectrum**, C.29, No: 2, 2007, 351–372.
Abdullah, S., Turabieh, H., McCollum, B.	"A Tabu-based Memetic Approach to the Examination Timetabling Problem", **Rough Set and Knowledge Technology**, Springer, Berlin, 2010, 574–581.
Abed, M. H., Tang, A. Y. C.	"Hybridizing Genetic Algorithm and Record-to-Record Travel Algorithm for Solving Uncapacitated Examination Timetabling Problem", **electronic Journal of Computer Science and Information Technology (eJCSIT)**, C.4, No: 1, 2013, 25–31.
Abounacer, R., Boukachour, J., Dkhissi, B., El Hilali Alaoui, A.	"A hybrid Ant Colony Algorithm for the exam timetabling problem", **Revue ARIMA**, C.12, 2010, 15–42.
Abraham, Ajith	"Rule–Based Expert Systems", **Handbook of Measuring System Design**, Ed. Peter H. Sydenham ve Richard Thorn, John Wiley & Sons Ltd., 2005, 909–919.
Akay, B.	**Nümerik Optimizasyon Problemlerinde Yapay Arı Kolonisi (Artificial Bee Colony) Algoritmasının Performmans Analizi**, Erciyes Üniversitesi Fen Bilimleri Enstitüsü, Kayseri, 2009.
Akinwale, O. C., Olatunde, O. S., Olusayo, O. E., Temitayo, F.	"Hybrid MetaHeuristic of Simulated Annealing and Genetik Algorithm for Solving Examination Timetabling Problem", **International Journal of computer Science and Engineering (IJCSE)**, C.3, No: 5, 2014, 7–22.
Al–Betar, M. A., Khader, A. T., Doush, I. A.	"Memetic techniques for examination timetabling", **Annals of Operations Research**, C.218, No: 1, 2014, 23–50.
Al–Yakoob, S. M., Sherali, H. D., Al–Jazzaf, M.	"A mixed-integer mathematical modeling approach to exam timetabling", **Computational Management Science**, C.7, No: 1, 2010, 19–46.
Alsharafat, W. S., Alsharafat M. S.	"Adaptive Steady State Genetic Algorithm for Scheduling University Exams", **International Conference on Networking and Information Technology, IEEE**, 2010, 70–74.

Alzaqebah, M., Abdullah, S.	"Artificial bee colony search algorithm for examination timetabling problems", **International Journal of Physical Sciences**, C.6, No: 17, 2011a, 4264–4272.
Alzaqebah, M., Abdullah, S.	"The bees algorithm for examination timetabling problems", **International Journal of Soft Computing and Engineering (IJSCE),** C.1, No: 5, 2011b, 105–110.
Alzaqebah, M., Abdullah, S.	"Hybrid bee colony optimization for examination timetabling problems", **Computers & Operations Research**, C.54, 2015, 142–154.
Arogundade, O. T., Akinwale, A. T., Aweda, O. M.	"A Genetic Algorithm Approach for a Real-World University Examination Timetabling Problem", **International Journal of Computer Applications**, C.12, No: 5, 2010, 1–4.
Asmuni, H., Burke, E. K., Garibaldi, J. M.	"A comparison of fuzzy and non-fuzzy ordering heuristics for examination timetabling" **Proceeding of 5th International Conference on Recent Advances in Soft Computing**, 2004, 288–293.
Asmuni, H., Burke, E. K., Garibaldi, J. M., McCollum, B.	"Determining rules in fuzzy multiple heuristic orderings for constructing examination timetables", **Proceedings of the 3rd multidisciplinary international conference on scheduling: theory and application**, 2007, 59–66.
Asmuni, H., Burke, E. K., Garibaldi, J. M., McCollum, B., Parkes, A. J.	"An investigation of fuzzy multiple heuristic orderings in the construction of university examination timetables", **Computers & Operations Research**, C.36, No: 4, 2009, 981–1001.
Bansal, J. C., Singh, P. K., Saraswat, M., Verma, A., Jadon, S. S., Abraham, A.	"Inertia Weight Strategies in Particle Swarm Optimization", **Third World Congress on Nature and Biologically Inspired Computing, IEEE**, 2011, 633–640.
Battistutta, M., Schaerf, A., Urli, T.	"Feature-based tuning of single-stage simulated annealing for examination timetabling", **10th International Conference of the Practice and Theory of Automated Timetabling (PATAT 2014)**, York, United Kingdom, 26–29 August 2014, 1–16.
Burke, E., Newall, J., Weare, R. A.	"A memetic algorithm for university exam timetabling" *In Practice and Theory of Automated Timetabling*, **Lecture Notes in Computer Science**, C.1153, 1996, 241–250.

Burke, E. K., Bykov, Y., Newall, J. P., Petrovic, S.	"A new local search approach with execution time as an input parameter", **Computer science technical report no. NOTTCS-TR-2002-3**, 2002.
Burke, E. K., Newall, J. P.	"Solving examination timetabling problems through adaption of heuristic orderings", **Annals of operations Research,** C.129(1-4), 2004, 107–134.
Burke, E. K., Meisels, A., Petrovic, S., Qu, R.	"A graph-based hyper heuristic for timetabling problems", **Computer Science Technical Report No. NOTTCS-TR-2004-9**, University of Nottingham, 2004a.
Burke, E. K., Bykov, Y., Newall, J., Petrovic, S.	"A time-predefined local search approach to exam timetabling problems", **IIE Transactions**, C.36, No: 6, 2004b, 509–528.
Burke, E. K., McCollum, B., Meisels, A., Petrovic, S., Qu, R.	"A graph-based hyper-heuristic for educational timetabling problems", **European Journal of Operational Research**, C.176, No: 1, 2007, 177–192.
Burke, E. K., Qu, R., Soghier, A.	"Adaptive selection of heuristics within a grasp for exam timetabling problems", **Proceedings of the 4th Multidisciplinary International Scheduling: Theory and Applications (MISTA 2009)**, Dublin, Ireland, 10-12 August 2009, 93–104.
Burke, E. K., Eckersley, A. J., McCollum, B., Petrovic, S., Qu, R.	"Hybrid variable neighbourhood approaches to university exam timetabling", **European Journal of Operational Research**, C.206, No: 1, 2010, 46–53.
Burke, E. K., Pham, N., Qu, R., Yellen, J.	"Linear combinations of heuristics for examination timetabling", **Annals of Operations Research**, C.194, No: 1, 2012, 89–109.
Burke, E. K., Qu, R., Soghier, A.	"Adaptive selection of heuristics for improving exam timetables", **Annals of Operations Research**, C.218, No: 1, 2014, 129–145.
Bykov, Y., Petrovic, S.	"An initial study of a novel step counting hill climbing heuristic applied to timetabling problems", **Proceedings of the 6th Multidisciplinary International Scheduling Conference: Theory & Applications (MISTA)**, Gent, Belgium, 2013.

Carter, M. W., Laporte, G., Lee, S. Y.	"Examination timetabling: Algorithmic strategies and applications", **Journal of the Operational Research Society**, C.47, No: 3, 1996, 373–383.
Castillo, Enrique, Gutierrez, Jose Manuel, Hadi, Ali S.	**Expert Systems and Probabilistic Network Models**, Springer Science & Business Media, 2012.
Chaudhari, A., De, K.	"Fuzzy Genetic Heuristic for University Course Timetable Problem", **International journal of Advance Soft Computing Application**, C.2, No: 1, 2010, 100–123.
Colorni, A., Dorigo, M., Maniezzo, V.	"Distributed optimization by ant colonies", **In Proceedings of the first European conference on artificial life**, C.142, Aralık 1991, 134–142.
Côté, P., Wong, T., Sabourin, R.	"A hybrid multi-objective evolutionary algorithm for the uncapacitated exam proximity problem", **Practice and Theory of Automated Timetabling V.**, Springer, Berlin Heidelberg, 2004, 294–312.
Cupic, M., Golub, M., Jacobovic, D.	"Exam Timetabling Using Genetic Algorithm", **Proceedings of 31st International Conference on Information Technology Interfaces (ITI2009)**, Croatia, 22–25 June 2009, 357–362.
Cura, Tunçhan	**Yöneticiler için Bilişim Teknolojileri ve Enformasyon Sistemleri**, Sistem Yayıncılık, İstanbul, 2009.
Darwin, Charles	**The Origin of Species**, London: John Murray, Albemarle Street, 1859.
Dasic, Milos, Trajkovic, Slavisa, Tesanovic, Branko	"The Necessity of Using Expert Systems in Strategic Decision Making", **International Journal of Economics & Law**, C.1, No: 1, 2011, 27–35.
Demeester, P., Bilgin, B., Causmaecker, P. D., Berghe, G. V.	"A hyperheuristic approach to examination timetabling problems: benchmarks and a new problem from practice", **Journal of Scheduling**, C.15, No: 1, 2012, 83–103.
Duque–Anton, M., Kunz, D., Rüber, B.	"Channel Assignment for Cellular Radio Using Simulated Annealing", **IEEE Transactions On Vehicular Technology**, C.42, No: 1, 1993, 14–21.
Durand, M. D.	"Parallel Simulated Annealing: Accuracy vs. Speed in Placement", **IEEE Design & Test of Computers**, C.3, 1989, 8–34.

Eberhart, R., Kennedy, J.	"A New Optimizer Using Particle Swarm Theory", **Proceedings of the sixth international symposium on micro machine and human science**, C.1, 1995, 39–43.
Fatima, S., Bader–El–Den, M.	"Co-evolutionary Hyper-Heuristic Method for Auction Based Scheduling", **International Conference Evolutionary Computation, IEEE**, 2010, 1–8.
Findik, Oğuz, Kiran, Mustafa Servet, Babaoğlu, Ismail	"Investigation Effects of Selection Mechanisms for Gravitational Search Algorithm", **Journal of Computer and Communications**, C.2, No: 4, 2014, 117–126.
Fong, C. W., Asmuni, H., McCollum, B., McMullan, P., Omatu, S.	"A new hybrid imperialist swarm-based optimization algorithm for university timetabling problems", **Information Sciences**, C.283, 2014, 1–21.
Geem, Z. W., Kim, J. H., Loganathan, G. V.	"A New Heuristic Optimization Algorithm: Harmony Search", **Simulation**, C.76, No: 2, 2001, 60–68.
Glover, F.	"Tabu Search – Part I", **ORSA Journal on Computing**, C.1, No: 3, 1989, 190–206.
Glover, F., Laguna, M.	**Tabu Search**, Kluwer Academic Publishers, ABD, 1998.
Gonciarz, T.	"An expert system for supporting the design and selection of mechanical equipment for recreational crafts", **TransNav: International Journal on Marine Navigation and Safety of Sea Transportation**, C.8, No: 2, Haziran 2014, 275–280.
Goodarzi, Mahdi Hassani	"A web-based Implementation of a Student Education Advisor System based on Fuzzy Expert Systems", **4th International Conference Information & Communication Technology and System (ICTS)**, 2008, 300–305.
Hadjidj, D., Drias, H.	"A hybrid grasp and scatter search for the exam timetabling problem" **2010 International Conference on Machine and Web Intelligence (ICMWI), IEEE,** 2010, 297–302.
Halliday, D., Resnick, R., Walker, J.	**Fundamentals of Physics**, John Wiley and Sons, 1993.

Hamilton–Bryce, R., McMullan, P., McCollum, B. "Directed selection using reinforcement learning for the examination timetabling problem", **10th International Conference of the Practice and Theory of Automated Timetabling (PATAT 2014)**, York, United Kingdom, 26–29 August 2014, 218–232.

Henderson, D., Jacobson, S. H., Johnson, A. W. "The Theory and Practice of Simulated Annealing", **Handbook of Metaheuristics**, Ed. Fred Glover ve Gary A. Kochenberger, Kluwer Academic Publishers, 2003, 287–320.

Holland, John Henry **Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence**, Ann Arbor, University of Michigan Press, 1975.

Hosny, M., Al–Olayan, M. "A mutation-based genetic algorithm for room and proctor assignment in examination scheduling", **Science and Information Conference (SAI), IEEE**, London, United Kingdom, 27–29 August 2014, 260–268.

Hung, N. Q. V., Binh, T. Q., Anh, D. T. "A Memetic Algorithm for Timetabling", **Proceedings of 3nd Int, Conf. RIVF**, 2005, C.5, 289–294.

Innet, S. "A noval approach of genetic algorithm for solving examination timetabling problems: A case study of Thai Universities", **13th International Symposium on Communications and Information Technologies (ISCIT), IEEE**, 2013, 233–237.

Ishida, Y. "Fully Distributed Diagnosis by PDP Learning Algorithm: Towards Immune Network PDP Model", **International Joint Conference on Neural Networks, IEEE**, 1990, 777–782.

Joy, Chukwuchekwa Ulumma, Nkwachukwu, Chukwuchekwa "Prediction of the Suitability of Locations for wind Farms Using Flex Expert Systems", **Academic Research International**, C.1, No: 1, Temmuz 2011, 244–257.

Kalaycı, C. B., Güngör, A. "A Genetic Algorithm Based Examination Timetabling Model Focusing on Student Success for the Case of the College of Engineering at Pamukkale University, Turkey", **Gazi University Journal of Science**, C.25, No: 1, 2012, 137–153.

Kaleem, Mohammed K.	"Role of Expert Systems in Digital Mammogram Processing", **International Journal of Computer Science and Technology**, C.2, No: 2, Haziran 2011, 96–99.
Karaboğa, D.	"An Idea Based on Honey Bee Swarm for Numerical Optimization", **Technical Report-tr06, Erciyes University, Engineering Faculty, Computer Engineering Department**, C.200, 2005.
Karaboğa, D., Baştürk, B.	"On the Performance of Artificial Bee Colony (ABC) Algorithm", **Applied Soft Computing**, C.8, No: 1, 2008, 687–697.
Karaboğa, D., Gorkemli, B., Ozturk, C., Karaboğa, N.	"A Comprehensive Survey: Artificial Bee Colony (ABC) Algorithm and Applications", **Artificial Intelligence Review**, C.42, No: 1, 2014, 21–57.
Khan, K., Sahai, A.	"A Comparison of BA, GA, PSO, BP and LM for Training Feed forward Neural Networks in e-Learning Context", **International Journal of Intelligent Systems and Applications**, C.4, No:7, 2012, 23–29.
Kirkpatrick, S., Gelat, C. D., Vecchi, M. P.	"Optimization by Simulated Annealing", **Science**, C.220, No: 4598, 1983, 671–680.
Kolonias, V., Goulas, G., Gogos, C., Alefragis, P., Housos, E.	"Solving the Examination Timetabling Problem in GPUs", **Algorithms**, C.7, No: 3, 2014, 295–327.
Komar, M., Grbic, D., Cupic, M.	"Solving exam timetabling using distributed evolutionary computation", **Proceedings of the 33rd International Conference on Information Technology Interfaces (ITI2011)**, IEEE, Croatia, 27–30 June 2011, 301–306.
Korkmaz, A.	"Anlaşılabilir Arıcılık", **Samsun Gıda Tarım ve Hayvancılık İl Müdürlüğü**, 2013, (Çevrimiçi) http://samsun.tarim.gov.tr/Belgeler/Yayinlar/Kitaplarimiz/anlasilabilir_aricilik.pdf, Erişim Tarihi: 20.10.2015.
Kuo, R. J., Chao, C. M., Chiu, Y. T.	"Application of particle swarm optimization to association rule mining", **Applied Soft computing**, C.11, No: 1, 2011, 326–336.

Lalla–Ruiz, E., Melian–Batista, B., Moreno–Vega, J. M.	"Artificial intelligence hybrid heuristic based on tabu search for the dynamic berth allocation problem", **Engineering Applications of Artificial Intelligence**, C.25, No: 6, 2012, 1132–1141.
Lei, Y., Gong, M., Jiao, L., Li, W., Zuo, Y., Cai, Q.	"A Double Evolutionary Pool Memetic Algorithm for Examination Timetabling Problems", **Mathematical Problems in Engineering**, C.2014, 2014, 1–13.
Lei, Y., Gong, M., Jiao, L., Zuo, Y.	"A memetic algorithm based on hyper-heuristics for examination timetabling problems", **International Journal of Intelligent Computing and Cybernetics**, C.8, No: 2, 2015, 139–151.
Leite, N., Melicio, F., Rosa, A. C.	"A Hybrid Shuffled Frog-Leaping Algorithm for the University Examination Timetabling Problem", **Computational Intelligence**, Springer International Publishing, 2016, 173–188.
Li, J., Qu, R.	"Evolutionary Ruin And Stochastic Recreate: A Case Study On The Exam Timetabling Problem", **ECMS**, 2012, 347–353.
Li, J., Bai, R., Shen, Y., Qu, R.	"Search with evolutionary ruin and stochastic rebuild: A theoretic framework and a case study on exam timetabling", **European Journal of Operational Research**, C.242, No: 3, 2015, 798–806.
Liao, Shu-Hsien	"Expert system methodologies and applications–a decade review from 1995 to 2004", **Expert Systems with Applications**, C.28, No:1, 2004, 1–11.
Malik, A. M. A., Ayob, M., Hamdan, A. R.	"Stratified Random Sampling Technique for Integrated Two-stage Multi-neighbourhood Tabu Search for Examination Timetabling Problem", **International Conference on Intelligent Systems Design and Applications, IEEE**, 2010, 1326–1331.
Malik, A. M. A., Othman, A.K., Ayob, M., Hamdan, A. R.	"Hybrid integrated two-stage multi-neighbourhood tabu search-EMCQ technique for examination timetabling problem", **2011 3rd Conference on Data Mining and Optimization (DMO), IEEE**, 2011, 232–236.

Mandal, A. K., Kahar, M. N. M.	"Solving examination timetabling problem using partial exam assignment with great deluge algorithm", **2015 International Conference on Computer, Communications, and Control Technology (I4CT), IEEE**, 2015, 530–534.
Mansour, N., Haidar, G. S.	"Parallel Metaheuristic Algorithm for Exam Timetabling", **Sixth International Conference on Natural Computation**, 2010, 471–475.
Mathew, Tom V.	"Genetic Algorithm", **Indian Institute of Technology Bombay**, 1–15, (Çevrimiçi) http://www.civil.iitb.ac.in/tvm/2701_dga/2701-ga-notes/gadoc.pdf, Erişim Tarihi: 05.10.2015.
McCulloch, W. S., Pitts, W.	"A logical calculus of the ideas immanent in nervous activity", **The Bulletin of Mathematical Biophysics**, C.5, No: 4, 1943, 115–133.
McCollum, B., McMullan, P., Burke, E. K., Parkes, A. J., Qu, R.	The second international timetabling competition: Examination timetabling track. **Technical Report QUB/IEEE/Tech/ITC2007/-Exam/v4**, 0/17, Queen's University, Belfast, 2007, 1–21.
Mehra, Molly, Jayalal, M. L., Arul, A. John	"Design and Development of Genetic Algorithm for Test Interval Optimization of Safety Critical System for a Nuclear Power Plant", **Online Proceedings on Trends in Innovative Computing, Intelligent Systems Design and Applications Conference**, Hindistan, C.166, 2012, 166–170.
Merlot, L. T. G., Boland, N., Hughes, B. D., Stuckey, P. J.	"A hybrid algorithm for the examination timetabling problem", **The Practice and Theory of Automated Timetabling IV**, Springer Berlin Heidelberg, 2002, 207–231.
Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., Teller, E.	"Equation of State Calculations by Fast Computing Machines", **The Journal of Chemical Physics**, C.21, No:6, 1953, 1087–1092.
Minsky, M., Papert, S.	**Perceptrons: An Essay In Computational Geometry**, MIT Press, Cambridge, 1969.

Monedero, I., Martin, A., Elena, J. M., Guerrero, J. I., Biscarri, F., Leon, C. "A Practical Overview of Expert Systems in Telecommunication Networks, Medicine and Power Supplies", **Expert System Software: Engineering, Advantages and Applications**, Ed. Jason M. Segura ve Albert C. Reiter, Nova Science Publishers Inc., 2013, 177–210.

Moradi, M. H., Abedini, M. "A Combination of Genetic Algorithm and Particle Swarm Optimization for Optimal DG Location and Sizing in Distribution Systems", **Electrical Power and Energy Systems**, C.34, No: 1, 2012, 66–74.

Mumford, C. L. "A multiobjective framework for heavily constrained examination timetabling problems", **Annals of Operations Research**, C.180, No: 1, 2010, 3–31.

Özdemir, M. **Zaman Kısıtı Altında Takım Oryantiring Problemlerinin Yapay Arı Kolonisi Yaklaşımı ile Çözümü**, İstanbul Üniversitesi Sosyal Bilimler Enstitüsü, İstanbul, Yüksek Lisans Tezi, 2013.

Pais, T. C., Amaral, P. "Managing the Tabu List Length Using a Fuzzy Inference System: An Application to Exams Timetabling", **The 7th International Conference for the Practice and Theory of Automated Timetabling**, 2008, 1–6.

Pais, T. C., Amaral, P. "Managing the tabu list length using a fuzzy inference system: an application to examination timetabling", **Annals of Operations Research**, C.194, No: 1, 2012, 341–363.

Parpinelli, Rafael S., Lopes, Heitor S., Freitas, Alex A. "Data Mining With an Ant Colony Optimization Algorithm", **IEEE Transactions on Evolutionary Computing**, C.6, No: 4, 2002, 321–332.

Peng, L., Linling, K., Xiang, C., Jian, Y., Jianhua, L., Xiaojuan, W. "Asymmetric Path-Relinking Based Heuristics for Large-scale Job Scheduling Problem in TDRSS", **9th International Conference on Communications and Networking in China (CHINACOM), IEEE**, 2014, 115–121.

Pillay, N., Banzhaf, W. "An Informed Genetic Algorithm For the Examination Timetabling Problem", **Applied Soft Computing**, C.10, No: 2, 2010, 457–467.

Pillay, N. "Evolving Hyper-Heuristics for a Highly Constrained Examination Timetabling Problem", **Proceedings of the 8th International Conference on the Practice and Theory of Automated Timetabling**, 2010, 336–346.

Pillay, N. "Evolving hyper-heuristics for the uncapacitated examination timetabling problem", **Journal of the Operational Research Society**, C.63, No: 1, 2012, 47–58.

Qu, R., Pham, N., Bai, R., Kendall, G. "Hybridising heuristics within an estimation distribution algorithm for examination timetabling", **Applied Intelligence**, C.42, No: 4, 2015, 679–693.

Rajah, C., Pillay, N. "A Study of Cell Depletion in the Developmental Approach for the Uncapacitated Examination Timetabling Problem", **Proceedings of the 2013 ORSSA Annual Conference**, 2013, 102–111.

Ramezanian, R., Rahmani, D., Barzinpour, F. "An Aggregate Production Planning Model for Two Phase Production Systems: Solving with Genetic Algorithm and Tabu Search", **Expert systems with Applications**, C.39, No: 1, 2012, 1256–1263.

Rashedi, E. **Gravitational Search Algorithm**, Shahid Bahonar University of Kerman, Kerman, Iran, Yüksek Lisans Tezi, 2007.

Rashedi, E., Nezamabadi-pour, H., Saryazdi, S. "GSA: A Gravitational Search Algorithm", **Information Sciences**, C.179, No: 13, 2009, 2232–2248.

Ribeiro, C. C., Resende, M. G. C. "Path-relinking intensification methods for stochastic local search algorithms", **Journal of Heuristics**, C.18, No:2, 2012, 193–214.

Sabar, N. R., Ayob, M., Kendall, G., Qu, R. "A honey-bee mating optimization algorithm for educational timetabling problems", **European Journal of Operational Research**, C.216, No: 3, 2012a, 533–543.

Sabar, N. R., Ayob, M., Qu, R., Kendall, G. "A graph coloring constructive hyper-heuristic for examination timetabling problems", **Applied Intelligence**, C.37, No: 1, 2012b, 1–11.

Sabar, N. R., Ayob, M., Kendall, G., Qu, R.,	"Grammatical evolution hyper-heuristic for combinatorial optimization problems", **IEEE Transactions on Evolutionary Computation**, C.17, No: 6, 2013, 840–861.
Shi, Y., Eberhart, R.	"A Modified Particle Swarm Optimizer", **IEEE International Conference on Computational Intelligence**, 1998, 69–73.
Slowik, Adam	"Type–2 Fuzzy Logic Control of Trade-off Between Exploration and Exploitation Properties of Genetic Algorithms", **Swarm and Evolutionary Computation**, Springer Berlin Heidelberg, 2012, 368–376.
Storn, R., Price, K.	"Differential Evolution - A simple and efficient adaptive scheme for global optimization over continuous spaces", **International Computer Science Institute**, Berkeley, 1995.
Tino, A., Khan, G. N.	"High performance NoC synthesis using analytical modeling and simulation with optimal power and minimal IC area", **Journal of Systems Architecture**, C.59, No: 10, 2013, 1348–1363.
To, T. V., Win, S. S.	"Clustering Approach to Examination Scheduling", **3rd International Conference on Advanced Computer Theory and Engineering**, C.5, 2010, 228–232.
Tomar, P. P., Saxena, P. K.	"Architecture for medical diagnosis using rule-based technique", **The First International Conference on Interdisciplinary Research and Development**, Thailand, 31 May - 1 June 2011, C.25, 1–5.
Tuker, M., Ballı, S., Pembeci, İ.	"Çok Etmenli Sistemlerde Netlogo ile Karınca Kolonisi Optimizasyonu", **Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi**, C.19, No: 2, 2013, 88–96.
Turabieh, H., Abdullah, S.	"An integrated hybrid approach to the examination timetabling problem", **Omega**, C.39, No: 6, 2011, 598–607.
University of Nottingham, School of Business	Exam Timetabling Benchmark Data Sets, (Çevrimiçi), http://www.cs.nott.ac.uk/~pszrq/data.htm, Erişim Tarihi: 01.07.2015.
University of Toronto Benchmark Data	(Çevrimiçi) ftp://ftp.mie.utoronto.ca/pub/carter/testprob/, Erişim Tarihi: 01.07.2015.

Vyas, P., Chauhan, A. "Comparative Optimization of Efficient Association Rule Mining through PSO and GA", **International Conference on Machine Intelligence and Research and Advancement, IEEE**, 2013, 258–263.

Wong, T., Côté, P., Gely, P. "Final exam timetabling: a practical approach", **IEEE Canadian Conference on Electrical and Computer Engineering**, IEEE, 2002, C.2, 726–731.

Wong, T., Assal, S. "A Multi–Neighborhood Multi–Operator Algorithm for the Uncapacitated Exam Proximity Problem", **Systemics, Cybernetics and Informatics**, C.13, No: 5, 2015, 56–61.

Wong, T., Bigras, P., Kelper, B. "A Multi-Neighborhood Multi-Operator Algorithm for the Uncapacitated Exam Proximity Problem", **2005 IEEE International Conference on Systems, Man and Cybernetics**, Hawaii, USA, 10–12 October 2005, 3810–3816.

Xie, J., Mei, Y., Ernst, A. T., Li, X., Song, A. "A Restricted Neighbourhood Tabu Search for Storage Location Assignment Problem", **Congress on Evolutionary Computation, IEEE**, 2015, 2805–2812.

Xuecheng, S. "A Hybrid Genetic Algorithm for Make-Up Examination Arrangement", **Seventh International Conference on Measuring Technology and Mechatronics Automation (ICMTMA), IEEE**, 2015, 10–13.

Yasin, A., Puteh, N., Tahir, H. M. "Examination Timetabling with Genetic Algorithms", Jurnal Teknologi Maklumat Dan Sains Kuantitatif Bil., 2002, C.1, 11–24.

Zeng, W., Gao, H. "Image Reconstruction in Wavelet Domain Optimized by Tabu Search Algorithm", **Metallurgical and Mining Industry**, C.2015, No: 5, 2015, 254–260.

ÖZGEÇMİŞ

Kişisel Bilgiler

Adı Soyadı	: Yakup ÇELİKBİLEK
Doğum tarihi, Yeri	: 1987, Denizli
E-mail	: yakup.celikbilek@istanbul.edu.tr

Eğitim

Derece	Kurum/Anabilim Dalı/Programı	Yılı
Doktora	İ.Ü. Fen Bilimleri Enstitüsü / Endüstri Mühendisliği	
Doktora	İ.Ü. Sosyal Bilimler Enstitüsü / Sayısal Yöntemler	2016
Yüksek Lisans	İ.Ü. Fen Bilimleri Enstitüsü / Endüstri Mühendisliği	2015
Yüksek Lisans	İ.Ü. Sosyal Bilimler Enstitüsü / Sayısal Yöntemler	2013
Lisans	Yıldız Teknik Üniversitesi / Fen Edebiyat Fakültesi / Matematik	2011
Lisans	Anadolu Üniversitesi / İşletme Fakültesi / İşletme	2011
Lise	Denizli Anadolu Lisesi	2005

Tezleri

Sayısal Yöntemler Doktora, **"Üniversite Sınav Çizelgeleme Problemlerinin Yapay Zeka Teknikleri ile Çözümü"**, İstanbul Üniversitesi Sosyal Bilimler Enstitüsü Sayısal Yöntemler ABD, Temmuz 2016. Tez Danışmanı: Doç. Dr. Tunçhan CURA.

Endüstri Mühendisliği Yüksek Lisans, **"Karar Teorisinde Bulanık Kümeler ve Gri Sistem Teorisi"**, İstanbul Üniversitesi Fen Bilimleri Enstitüsü Endüstri Mühendisliği ABD, Ağustos 2015. Tez Danışmanı: Yrd. Doç. Dr. Fatih TÜYSÜZ.

Sayısal Yöntemler Yüksek Lisans, **"Çok Kişili Oyunlar ve Bir Uygulama"**, İstanbul Üniversitesi Sosyal Bilimler Enstitüsü Sayısal Yöntemler ABD, Temmuz 2013. Tez Danışmanı: Prof. Dr. Öner ESEN.

Makaleler

Çelikbilek, Y., Adigüzel Tüylü, A. N., Esnaf, Ş., "Industrial Coffee Machine Selection with the Fuzzy Analytic Hierarchy Process", **International Journal of Management and Applied Science**, C.2, No:2, 2016, 20–23.

Uluslararası Bildiriler

Tüysüz, F., Çelikbilek Y., "Analysis of the Factors for the Evaluation of Renewable Energy Resources Using Hesitant Fuzzy Linguistic T", **The 8th International Ege Energy Symposium and Exhibition (IEESE)**, 2016.

Özdemir, M., Çelikbilek, Y., Cura T., "A New Heuristic Optimization: Search and Rescue Algorithm", **The Economics, Finance, Healthcare Management, Accounting, IT, & Management Research Conference**, 2016.

Çelikbilek, Y., Adıgüzel Tüylü, A. N., "Evaluating the Technological Factors of Distance Learning", **The IAFOR International Conference on Education 2016 – Hawaii**, 2016.

Çelikbilek, Y., Adıgüzel Tüylü, A. N., Esnaf, Ş., "Industrial Coffee Machine Selection with the Fuzzy Analytic Hierarchy Process", **2015 The IIER 28th International Conference on Advances in Business Management and Information Technology**, 2015.

Çelikbilek , Y., Tüysüz, F., "A Fuzzy Multi-Criteria Decision Making Approach for Evaluating Renewable Energy Sources", **The 4th International Fuzzy Systems Symposium**, 2015.

Adıgüzel Tüylü, A. N., Çelikbilek, Y., "An Integrated AHP and VIKOR Approach for Excavator Selection in Open Pit Mining", **27. European Conference on Operations Research**, 2015, 12–12.

Çelikbilek, Y., Adıgüzel Tüylü, A. N., "Analysing the Components of E-learning with DEMATEL and ANP", **27. European Conference on Operations Research**, 2015, 79–79.

Çelikbilek, Y., "Devlet Üniversitelerinin Akademik Performanslarının Çok Boyutlu Ölçekleme Yöntemi ile Analizi", **15. Uluslararası Ekonometri, Yöneylem Araştırması ve İstatistik Sempozyumu**, 2014, 417–427.

Kitap Bölümleri

Çelikbilek, Y., "Üstünlük Tabanlı Kaba Küme Analizi", **Çok Kriterli Karar Verme Yöntemleri**, Ed. Bahadır F. Yıldırım ve Emrah Önder, Dora Yayıncılık, 2014, 313–338.

Editörlük

Alphanumeric Journal, Üretim Editörü, 2013 – 2015.