

**ÇİZELGELEME PROBLEMİNİN
SEZGİSEL OPTİMİZASYON YAKLAŞIMIYLA
ÇÖZÜMÜ**

**2011
YÜKSEK LİSANS TEZİ
ELEKTRONİK VE BİLGİSAYAR EĞİTİMİ**

Abdullah ELEN

**ÇİZELGELEME PROBLEMİNİN SEZGİSEL OPTİMİZASYON
YAKLAŞIMIYLA ÇÖZÜMÜ**

Abdullah ELEN

**Karabük Üniversitesi
Fen Bilimleri Enstitüsü
Elektronik ve Bilgisayar Eğitimi Anabilim Dalında
Yüksek Lisans Tezi
Olarak Hazırlanmıştır**

**KARABÜK
Ocak 2011**

Abdullah ELEN tarafından hazırlanan “ÇİZELGELEME PROBLEMİNİN SEZGİSEL OPTİMİZASYON YAKLAŞIMIYLA ÇÖZÜMÜ” başlıklı bu tezin Yüksek Lisans Tezi olarak uygun olduğunu onaylarım.

Yrd.Doç.Dr. İbrahim ÇAYIROĞLU

Tez Danışmanı, Mekatronik Mühendisliği Anabilim Dalı



Bu çalışma, jürimiz tarafından oy birliği ile Elektronik ve Bilgisayar Eğitimi Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiştir. 26/01/2011

Unvanı, Adı SOYADI (Kurumu)

İmzası

Başkan : Yrd.Doç.Dr. İbrahim ÇAYIROĞLU (KBÜ)



Üye : Yrd.Doç.Dr. Salih GÖRGÜNOĞLU (KBÜ)



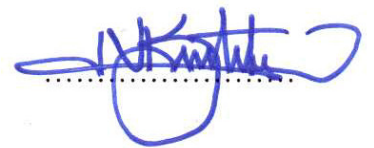
Üye : Yrd.Doç.Dr. Baha ŞEN (KBÜ)



..... /..... /2011

KBÜ Fen Bilimleri Enstitüsü Yönetim Kurulu, bu tez ile Yüksek Lisans derecesini onaylamıştır.

Doç.Dr. Nizamettin KAHRAMAN
Fen Bilimleri Enstitüsü Müdürü



“Bu tezdeki tüm bilgilerin akademik kurallara ve etik ilkelere uygun olarak elde edildiğini ve sunulduğunu; ayrıca bu kuralların ve ilkelerin gerektirdiği şekilde, bu çalışmadan kaynaklanmayan bütün atıfları yaptığımı beyan ederim.”

Abdullah ELEN

ÖZET

Yüksek Lisans Tezi

ÇİZELGELEME PROBLEMİNİN SEZGİSEL OPTİMİZASYON YAKLAŞIMIYLA ÇÖZÜMÜ

Abdullah ELEN

Karabük Üniversitesi

Fen Bilimleri Enstitüsü

Elektronik ve Bilgisayar Eğitimi Anabilim Dalı

Tez Danışmanı:

Yrd. Doç. Dr. İbrahim ÇAYIROĞLU

Ocak 2011, 72 sayfa

Bu çalışmada, üniversitelerde kullanılan öğrenci işleri otomasyonu içerisindeki Ders Çizelgeleme probleminin çözümü, Sezgisel Optimizasyon yaklaşımlarından olan Genetik Algoritma yöntemi kullanılarak gerçekleştirilmiştir. Çizelgeleme problemi, yapılacak işlerin belirlenen kısıtlar dahilinde zaman aralıklarına optimum düzeyde yerleştirilmesi işlemidir. Üniversitelerde bu problemin çözümü, çözüm uzayının büyük olması ve kısıtlamaların çok sayıda bulunması nedeniyle analitik yöntemlerle imkânsız hale gelmektedir. Bu nedenle bu türden problemler için çok iyi sonuçlar veren Genetik Algoritma yöntemi yeni bir yaklaşımla, en iyi sonucu veren parametreler araştırılarak kullanılmıştır.

Üniversitedeki gerçek veriler üzerinden (504 öğretim elemanı, 4163 ders, 203 derslik ve 10525 öğrenci) uygulamalar yapıp, geliştirilen algoritmanın performansı ölçülmüştür.

Geliştirilen algoritmanın üniversite ortamında uygulanabilmesi, veri girişlerinin yapılabilmesi ve sonuçlarının raporlanabilmesi için ihtiyaç duyulan gerekli diğer tüm alt yapı yazılımları hazırlanarak tamamı bir paket program haline getirilmiştir.

Bu amaçla geliştirilen yazılım üç ayrı modülden oluşmaktadır. Bunlar Sistem Yönetim Modülü, Ders Modülü ve Sınav Modülüdür.

Bütün bu modüllerin ara yüzleri, veritabanı ve tabloları, bu tablolar arasındaki bağlantılar ve sorgular programlanarak kullanıma hazır hale getirilmiştir. Programlama alt yapısında veri girişleri ve raporlama kısımları internet ortamında ASP.NET, C# Programlama Dili ve SQL Server veritabanı kullanılarak gerçekleştirilmiştir. Geliştirilen algoritma ise C# Programlama Dili ve Microsoft SQL Server veritabanı kullanılarak Windows Application uygulaması şeklinde gerçekleştirilmiştir.

Anahtar Sözcükler : Sezgisel metot, yapay zeka, genetik algoritma, zaman çizelgeleme.

Bilim Kodu : 702.1.178

ABSTRACT

M.Sc. Thesis

SOLVING OF SCHEDULING PROBLEM WITH HEURISTIC OPTIMIZATION APPROACH

Abdullah ELEN

Karabük University

Graduate School of Natural and Applied Sciences

Department of Electronic and Computer Education

Thesis Advisor:

Assist. Prof. Dr. İbrahim ÇAYIROĞLU

January 2011, 72 pages

In this study, solution of the problems in Course Scheduling that is used in the automation of student affairs at universities was carried out using the genetic algorithm which is one of the Heuristic Approaches. Scheduling problem is the process of placement of the work to be done to the time span according to the constraints set at the optimum level. Solution to this problem in universities is impossible due to large solution space and very large number of restrictions. Therefore, the genetic algorithm method that gives very good results for this kind of problems was used with a new approach investigating the parameters that gives the best results.

The performance of the algorithm, which was developed and practiced, was measured on the actual data used in the University (504 Academic staffs, 4163 lessons, 203 classrooms and 10525 students)

In order for the algorithm to be applied to the university environment, for the data entry and for the results to be reported, the needed background software has been supplied and made into a package program.

Therefore, developed software consists of three separate modules. These are the System Management Module, Course and Examination Modules.

All of these modules' interfaces, database and tables, the links between tables and queries have been programmed ready to use. In the programming background, Asp.net, C # programming language and Microsoft SQL Server database were used for the data entry and reporting on the internet. Parts of the placement algorithm in the lecture and examination were carried out in application of the Windows application.

Key Words : Heuristic approach, artificial intelligence, genetic algorithm, time scheduling.

Science Code : 702.1.178

TEŐEKKÜR

Bu tez alıőmasının planlanmasında, araőtırılmasında, yürütülmesinde ve oluşumunda ilgi ve desteęini esirgemeyen, bilgi ve tecrübelerinden yararlandıęım, yönlendirme ve bilgilendirmeleriyle alıőmamı bilimsel temeller ışığında şekillendiren sayın hocam Yrd. Do. Dr. İbrahim AYIROęLU'na sonsuz teőekkürlerimi sunarım.

Her daim, maddi ve manevi, hiçbir desteęini esirgemeyen, canımdan ok sevdięim aileme, araőtırma ve alıőmalarım sırasında destek olan herkese gönülden teőekkür ederim.

İÇİNDEKİLER

	Sayfa
ÖZET	iv
ABSTRACT.....	vi
TEŞEKKÜR.....	viii
İÇİNDEKİLER	ix
ŞEKİLLER DİZİNİ.....	xii
ÇİZELGELER DİZİNİ	xiv
EKLER DİZİNİ.....	xv
SİMGELER VE KISALTMALAR DİZİNİ	xvi
BÖLÜM 1	1
GİRİŞ	1
BÖLÜM 2	5
LİTERATÜR ARAŞTIRMASI	5
BÖLÜM 3	11
GENETİK ALGORİTMALAR	11
3.1. GENETİK ALGORİTMANIN ÖZELLİKLERİ.....	11
3.2. GENETİK ALGORİTMANIN ÇALIŞMA PRENSİBİ.....	12
3.2.1. Genetik Algoritmanın İşlem Adımları.....	15
3.3. GENETİK ALGORİTMANIN TEMEL KAVRAMLARI	15
3.3.1. Gen ve Kodlama	15
3.3.2. Kromozom	16
3.3.3. Popülasyon.....	16
3.4. GENETİK OPERATÖRLER	17
3.4.1. Seçim Operatörü	17

3.4.1.1. Rulet Tekeri	18
3.4.1.2. Sıralı (Rank).....	18
3.4.1.3. Turnuva.....	19
3.4.1.4. Sabit Durum.....	19
3.4.2. Çaprazlama Operatörü	20
3.4.2.1. Tek Noktalı Çaprazlama	20
3.4.2.2. İki Noktalı Çaprazlama.....	20
3.4.2.3. Düzgün (Uniform) Çaprazlama	21
3.4.2.4. Yeniden Dizilim Yapısına Sahip Çaprazlama Operatörleri.....	21
3.4.3. Mutasyon Operatörü	26
3.4.4. Durdurma Kriteri	28
BÖLÜM 4	29
PROBLEMİN ÇÖZÜMÜNE UYGUN GELİŞTİRİLEN MODEL	29
4.1. ÖĞRENCİ İŞLERİ OTOMASYONUNUN MODÜLLERİ	31
4.1.1 Yönetim Modülü.....	31
4.1.2 Ders Modülü	32
4.1.3 Sınav Modülü	33
4.2. TASARLANAN GENETİK MODEL	33
4.2.1. Kromozom ve Gen Yapısı	34
4.2.2. Kullanılan Seçim Yöntemleri	37
4.2.3. Kullanılan Çaprazlama Yöntemleri	40
4.2.3.1. Tek Noktalı Çaprazlama	42
4.2.3.2. İki (Çift) Noktalı Çaprazlama	42
4.2.3.3. Periyodik Çaprazlama.....	42
4.2.4. Kullanılan Mutasyon Yöntemleri	42
4.2.4.1. Karşılıklı Değişim Yöntemi.....	42
4.2.4.2. Özel Yöntem.....	42
4.3. PROBLEMİN ÇÖZÜMÜNDE KULLANILAN KISITLAR	44
4.3.1. Eğitim/Öğretim Verimliliği	44

4.3.2. Öğretim Elemanı Memnuniyeti	45
4.3.3. Ders Yüğü Ağırlığı	45
4.3.4. Bölünen Derslerin Buluşma Zamanları	46
4.3.5. Ders Saatleri Arasındaki Boş Zamanlar	47
4.4. GELİŞTİRİLEN YAPININ MATEMATİKSEL MODELİ.....	47
4.5. GELİŞTİRİLEN UYGULAMA ARAYÜZÜ	50
BÖLÜM 5	57
DENEYSEL ÇALIŞMALAR	57
BÖLÜM 6	65
SONUÇLAR	65
KAYNAKLAR	67
ÖZGEÇMİŞ	72

ŞEKİLLER DİZİNİ

	<u>Sayfa</u>
Şekil 3.1. GA'nın genel akış diyagramı.	14
Şekil 3.2. Rulet tekeri kromozom dağılım grafiği.....	18
Şekil 3.3. Sıralamadan önce (uygunluk çizelgesi).	19
Şekil 3.4. Turnuva seçim yöntemi.	20
Şekil 4.1. Geliştirilen öğrenci işleri otomasyonunun arayüz görünümü.....	29
Şekil 4.2. Geliştirilen öğrenci işleri otomasyonunun genel yapısı.....	30
Şekil 4.3. Geliştirilen modelin genel yapısı.	34
Şekil 4.4. Kromozom ve gen yapısı.	36
Şekil 4.5. Çakışmaların tespit edilmesi.	37
Şekil 4.6. Kullanılan rulet tekeri seçim yönteminin prensibi.....	38
Şekil 4.7. Kullanılan sıralı seçim yönteminin prensibi.	39
Şekil 4.8. Kullanılan turnuva seçim yönteminin prensibi.....	39
Şekil 4.9. Kullanılan tek noktalı çaprazlama yönteminin prensibi.	40
Şekil 4.10. Kullanılan iki noktalı çaprazlama yönteminin prensibi.	41
Şekil 4.11. Kullanılan periyodik çaprazlama yönteminin prensibi.....	42
Şekil 4.12. Kullanılan karşılıklı değişim mutasyonunun prensibi.	42
Şekil 4.13. Kullanılan özel yöntem mutasyonunun prensibi.....	43
Şekil 4.14. Pedagojik gün ve saat verimliliği.....	44
Şekil 4.15. Öğretim elemanı memnuniyet çizelgesi.....	45
Şekil 4.16. Dersler arasındaki boş saatleri ayarlayan kısıtlar.	47
Şekil 4.17. Geliştirilen uygulama arayüzü ana ekranı.....	51
Şekil 4.18. GA parametrelerinin seçimi.	52
Şekil 4.19. Kısıt tanımlama ekranı.	53
Şekil 4.20. Sonuç grafikleri.....	54
Şekil 4.21. Öğretim elemanı haftalık ders çizelgesi.....	55
Şekil 4.22. Öğrenci haftalık ders çizelgesi.....	55
Şekil 4.23. Derslik haftalık ders çizelgesi.....	56

Şekil 5.1. Başarı oranları grafiđi.	59
Şekil 5.2. Uygunluk deđerleri.	60
Şekil 5.3. Pedagojik gn ve saat verimliliđi.....	60
Şekil 5.4. Başarı oranları grafiđi.	61
Şekil 5.5. Uygunluk deđerleri.	62
Şekil 5.6. Başarı oranları grafiđi.	63
Şekil 5.7. Uygunluk deđerleri.	64

ÇİZELGELER DİZİNİ

	<u>Sayfa</u>
Çizelge 4.1. Kısıt ağırlıkları tablosu.....	47
Çizelge 4.2. Pedagojik verimlilik (günler).....	49
Çizelge 4.3. Pedagojik verimlilik (saatler).....	49
Çizelge 5.1. Deney verilerinin büyüklüğü.....	57

EKLER DİZİNİ

- Ek I. Öğrenci işleri otomasyonunun veritabanı tablo yapısı ve ilişkilerinin gösterilmesi (ER-Diyagramı).

SİMGELER VE KISALTMALAR DİZİNİ

KISALTMALAR

GA	: Genetik Algoritma(lar)
UD	: Uygunluk Deęeri
NP	: Non-deterministic Polynomial
UBYS	: Üniversite Bilgi Yönetim Sistemi

BÖLÜM 1

GİRİŞ

Çizelgeleme problemi, yapılacak işlerin belirlenen kısıtlar dahilinde zaman aralıklarına optimum düzeyde yerleştirilme işlemidir. Üniversitelerde haftalık ders programlarının hazırlanması işlemi çizelgeleme problemlerinden biridir. Bu tip bir problem polinom zamanda çözülemeyen NP (Non-deterministic Polynomial) tipi bir problemdir. Bazı problemlerin çözümü için kullanılan en etkili algoritmanın çalışma süresi, girilen verilerin büyüklüğüne bir polinom cinsinden bağlı olduğu bilinmektedir. Bu tip problemler P (Polinomal) problemlerdir. Eğer hesaplama için gerekli süreyi ya da tekrar miktarını verecek böyle bir polinom bulunamıyor ise, bu tip problemler NP (Polinomal olmayan) problemler olarak adlandırılır.

Üniversitede ders yerleştirme işlemlerinin olasılık hesaplarının yapıldığında çok büyük ihtimaller ortaya çıkmaktadır. Yaklaşık bir hesapla, ortalama 3 saatlik 240 adet dersin bulunduğu bir fakültede tüm derslerin haftalık çizelgeye yerleştirilmesi işlemi 400^{240} gibi çok büyük ihtimal hesabını ortaya çıkarmaktadır. Bunu bir örnekle açıklamak gerekirse;

Karabük Üniversitesinde; herhangi bir akademik program için, haftanın 5 günü ve günlük 10 saat (periyot) olmak üzere ders işlenebilmektedir. Örneğin Teknik Eğitim Fakültesinde 240 adet ders açılmış olsun ve bunların her birinin 3 saatlik ders olduğunu kabul edelim. Ayrıca her bir ders için, dersin işlenebileceği 10 farklı derslik tanımlanmış olsun.

Açılan bu 240 dersten herhangi birinin çizelgeye yerleşme olasılığını hesaplayalım;

G: Çizelgedeki gün sayısı

S: Çizelgedeki bir güne ait toplam saat sayısı

D: Bir dersin işleneceği saat sayısı

$$F1 = (G) \times (S - D + 1)$$

$$F1 = 5 \times (10 - 3 + 1)$$

F1 = 40 farklı şekilde yerleşebilmektedir.

Şimdi 10 farklı dersliđi de hesaba katalım;

$$F2 = F1 \times 10$$

F2 = 400 farklı şekilde yerleşebilmektedir.

Tüm açılan dersler için toplam olasılık hesabı ise;

$$F3 = F2^{240}$$

F3 = 400²⁴⁰ olarak hesaplanabilir.

Yukarıda verilen örnekte hiçbir kısıt hesaba katılmamıştır. Genetik Algoritma ile çözümlenmesi gereken problemin büyüklüğüne (çözüm uzayı) dair yaklaşık bir hesaplama yapılmıştır.

Ayrıca bu işleme ek olarak gerekli kısıtlayıcı bilgilerde eklendiğinde problem daha da karmaşık hale gelmektedir. Bu yüzden bu tip problemlerin geleneksel yöntemlerle ve matematiksel algoritmalarla çözümlenmesi neredeyse imkânsızdır ve hesaplama süresi ya da döngü sayısı net olarak bilinemez.

Zaman çizelgeleme (Time-Scheduling) problemleri 1960 yılından bu yana birçok araştırmacı tarafından çalışılmaktadır. Bunun sebebi eğitimle ilgili zaman çizelgeleme problemlerinin çok geniş bir alan olmasıdır. Çözüm yaklaşımları da problemin büyüklüğüne göre değişmektedir. Ancak her zaman çizelgeleme probleminin özünde, çözümlenmesi gereken bir kombinasyon problemi bulunmaktadır. Bu yüzden ayrık optimizasyon problemleri için bilinen teknikler, zaman çizelgeleme problemlerinde de kullanılabilir [1].

Ayrık optimizasyon problemleri için “genel olarak, tanımlanması kolay çözümleri zor problemler” denilebilir. Bu problemler genellikle belirli bir matematiksel fonksiyon olarak ifade etmek güçtür.

Bu nedenle, bu türdeki problemleri klasik optimizasyon yöntemleri ile çözmek zordur. Çözüm için yalın çözüm üreten sezgisel algoritmalara ihtiyaç vardır. Genetik bilimi ve tabii seçme üzerine dayalı olan genetik algoritmalar bu türden algoritmalar olup, bunlar çeşitli ayırık optimizasyon problemleri için kullanılmış ve iyi sonuçlar elde edilmiştir [2].

Genetik algoritmalar, problemlere tek bir çözüm üretmek yerine farklı çözümlerden oluşan bir çözüm kümesi üretir. Böylelikle, arama uzayında aynı anda birçok nokta değerlendirilmekte ve sonuçta bütünsel çözüme ulaşma olasılığı yükselmektedir. Çözüm kümesindeki çözümler birbirinden tamamen bağımsızdır. Genetik algoritmalar problemlerin çözümü için evrimsel süreci bilgisayar ortamında taklit ederler.

Bu çalışmada, Karabük Üniversitesi bünyesinde 2009–2010 yılına ait gerçek veriler kullanılarak (504 öğretim elemanı, 4163 ders, 203 derslik ve 10525 öğrenci) algoritma geliştirilmiş ve uygulama detayları gösterilmiştir.

Hazırlanan çalışmanın amacı, üniversitelerdeki ders çizelgilerinin hazırlanması sırasında yaşanan zorlukları gidermek, akademisyenler üzerindeki iş yükünü azaltmak, kaynakların etkin kullanımını sağlamak, öğrenci ve öğretim elemanı memnuniyetini göz önünde bulundurarak, pedagojik verimliliğin artırılmasını hedef almaktadır.

Tez çalışması 6 bölümden oluşmaktadır, Bunlar;

Birinci bölümde, çalışma konusu ve kullanılan yöntem hakkında genel bilgiler verilmiştir.

İkinci bölümde, Çizelgeleme problemleri ve genetik algoritmaların kullanıldığı çizelgeleme problemleri hakkında daha önceden yapılan çalışmalar anlatılarak geniş bir literatür araştırması sunulmuştur.

Üçüncü bölümde; Genetik algoritmaların hakkında genel bilgiler verilmiştir.

Dördüncü bölümde, geliştirilen Öğrenci İşleri Otomasyonunun tanıtımı, probleme uygun genetik algoritmanın tasarımı, geliştirilen uygulama arayüzü tanıtılmıştır.

Beşinci bölümde genetik algoritmanın verimli çalışması için gerekli olan optimum parametrelerin elde edilmesi için yapılan deneyler anlatılmıştır.

Altıncı ve son bölümde ise yapılan çalışmadan elde edilen sonuçların bir özeti verilmiştir.

BÖLÜM 2

LİTERATÜR ARAŞTIRMASI

Çizelgeleme problemleri araştırmacıların en çok ilgilendiği konulardan biridir. Bu konu üzerine 1960'lı yıllardan bu yana çalışmalar yapılmaktadır. Problemin karmaşıklığı ve uygulamaların farklılığı nedeniyle kullanılan yöntemler sürekli olarak geliştirilmektedir. Bu konuda yapılan çalışmaları üç grup altında toplamak mümkündür. Bunlar yapay zeka, insan makine etkileşimi ve yöneylem araştırması. Yapay zeka olarak kısıt programlama, uzman sistemler, yapay sinir ağları gibi teknolojileri sayabiliriz. İnsan makine etkileşimi olarak yinelemeli işlem uygulamaları ve makine çözümünün ardından insan kabulü şeklinde yapılan uygulamaları sayabiliriz. Yöneylem araştırmaları ise üç gruba ayrılabilir. Bunlar; ardışık yöntemler, kümeleme yöntemleri ve meta sezgisel yöntemlerdir. Yapılan bu çalışma bu gruplandırma içerisinde Yöneylem araştırması içerisinde geçen meta sezgisel yöntemlerden Genetik Algoritma başlığı altında bulunmaktadır.

Genetik algoritma ile ilgili olarak ilk çalışmalar 1970 yıllarda başlamıştır. Bu konuda ilk çalışmaları yapan John Holland doğal evrimden ve canlılardaki bu süreçten yararlanarak, makine öğrenmesi üzerine çalışmalarını yoğunlaştırmıştır. Holland'ın asıl amacı, belirli problemleri çözmek için algoritma tasarlamak değil; uyum olgusu üzerinde çalışmak ve doğal uyumun bilgisayar sistemlerinde kullanılabilmesini sağlamak olmuştur.

Genetik algoritmaların çizelgeleme problemine ilk uygulama çalışması, Davis tarafından 1985 yılında yapılmıştır [3]. 1987'de Liepins et al., belirli teslim tarihleri ve işlem süreleri olan işlerin çizelgelenmesi problemini araştırmışlardır [4]. Bu problem en basit çizelgeleme problemi olarak adlandırılmaktadır. 1993'de Gupta et al., akış zamanını minimize etme amacını taşıyan tek makine modeli üzerindeki çalışmalarını yayınlamışlardır [5]. Lee et al. 1995'de gecikme ve sarkma cezalarını da modele katan çalışmalarını sunmuşlardır [6, 7].

Bunun dışında; iş atölyesi çizelgeleme problemi için Biegel and Davern'nin 1990'da, akış atölyesi problemi için Badami and Parks'ın 1991'de, süreç planlama probleminin çözümü için Vancza and Markus'un 1991'de yayınlanmış çalışmaları bulunmaktadır [8]. Genel olarak genetik algoritmalar, çizelgeleme problemlerine optikale yakın çözüm bulmuşlardır. Çözüm süreleri diğer çözüm yöntemlerine göre oldukça hızlı olmuştur.

Son yapılan çalışmalardan AL-Milli yaptığı çalışmada, ders çizelgeleme probleminin çözümünde Genetik Algoritma ve Great Deluge algoritmasına dayalı yeni bir melez algoritma kullanmıştır [9].

Wijaya and Manurung yaptıkları çalışmada, Genetik Algoritma ile Endonezya Üniversitesi Bilgisayar Bilimleri Fakültesi için gerçek kısıtlarla modellenmiş bir çizelgeleme problemini başarılı bir şekilde tüm kısıtları karşılayan bir çözüm üretmişlerdir [10].

Jat and Yang yaptıkları çalışmada, Genetik Algoritma ile rehberli bir araştırma stratejisi ve bir yerel arama tekniği kullanarak üniversite ders çizelgeleme probleminin çözümünde umut verici deneysel sonuçlar elde etmişlerdir [11].

Atanak ve Hocaoğlu yaptıkları çalışmada, genetik algoritmalar kullanarak sınıfların boyutunu, ders sayılarını, derse kaydolması gereken öğrenci sayısını ve dersi verecek olan öğretim elemanlarını girdi olarak kullanan ve verilen kısıtlar altında dönemlik ders programı hazırlayan bir otomasyon geliştirmiştir [12].

Bratkovic et al. yaptıkları çalışmada, çok kısıtlı bir laboratuvar örneği ele alınarak genetik algoritma ile zaman çizelgeleme probleminin çözümünde kullanmışlardır [13].

Aldasht et al. yaptıkları çalışmada, üniversite ders çizelgeleme probleminin çözümünde genetik algoritmalara benzer evrimsel algoritmalar kullanmışlardır. El ile hazırlanan çizelgelere göre daha uygun sonuçlar vermiştir [14].

Aydın yaptığı çalışmada, üniversite ders çizelgeleme probleminin çözümünde genetik algoritmaları kullanmıştır. Bahçeşehir Üniversitesi Fen Edebiyat Fakültesine ait gerçek verileri kullanarak hiç bir kısıtı göz ardı etmeden optimum sonuçlar elde etmiştir [15].

Kalender tarafından yapılan çalışmada üst-sezgisel algoritmalar ile Yeditepe Üniversitesi Bilgisayar mühendisliğinde kullanılmak üzere çözümlerin üretilebildiği, değiştirebildiği ve bilgilerin saklanabildiği otomatik ders çizelgeleme programı hazırlayan bir uygulama geliştirilmiştir [16].

Juang et al. planlanmış çalışan eğitim programları için Genetik Algoritma ile adaptif çizelgeleme sistemini bilgisayar destekli olarak makine endüstrisinde gerekli çalışanların eğitimi ve atamaları için geliştirmişlerdir [17].

Baç yaptığı çalışmada, akademik ders programlarının yapılması problemi için etkin çözümler sağlayan yeni bir matematiksel model geliştirmiştir. Ancak işlem gücü yeterince yüksek bir bilgisayar veya iş istasyonları yardımı ile geliştirdiği modelin, tüm fakülte için çalıştırılarak atama yapılmasının mümkün olabileceğini savunmuştur [18].

Temiz ve Erol tarafından yapılan çalışmada, m-makinelik akış tipi çizelgeleme probleminde işlem zamanları ve teslim tarihleri gibi zaman parametrelerinin belirsiz olduğu durum ele alınarak üretim tamamlanma zamanı, maksimum gecikme ve toplam akış zamanı amaçlarını eş zamanlı eniyileyen genetik algoritma temelli çok amaçlı bir yaklaşım geliştirilmiştir [19]. Algoritmanın orta ve büyük boyutlardaki problemler için makul zamanda etkin çözümleri ürettiği gösterilmiştir.

Gülcü yaptığı çalışmada, kombinatoriyel optimizasyon problemlerini çözmek amacıyla geliştirilmiş ve uzun yıllardır birçok alanda başarıyla kullanılmış algoritmalarından genetik algoritma ve tabu arama yöntemleri kullanarak, eğitim kurumları için haftalık ders programı hazırlamış ve genetik algoritmaların daha iyi sonuç verdiğini tespit etmiştir [20].

Datta et al. yaptıkları çalışmada, Kanpur Hint Teknoloji Enstitüsü sınıf zaman çizelgeleme probleminin çözümünde çok amaçlı evrimsel algoritma yöntemini kullanmışlardır. Pek çok üniversitenin aksine IIT Kanpur sınıf çizelgelerinin hazırlanması, sınıflarının birçoğunun bölünmüş ya da gruplandırılmış olmasından dolayı çok zahmetli ve karmaşıktır. Elde edilen sonuçlar manüel olarak hazırlanan çözümlerden daha iyi olduğu kabul edilmiştir [21].

Çayıroğlu ve Dizdar yaptıkları çalışmada uzman sistem temelli bir ders yerleştirme uygulaması geliştirmişlerdir. Hazırlanan program, hiçbir çakışma olmayacak şekilde, tüm haftalık ders programını hazırlayarak, gerekli dokümanları vermektedir [22].

Yiğit tarafından yapılan çalışmada, meslek liseleri için haftalık ders çizelgelerinin genetik algoritmalar yardımıyla oluşturulması sağlanmıştır [23]. Bu çalışmada genetik algortmada kullanılan mutasyon operatöründen sonra kromozomların yapısında oluşan bozuklukların giderilmesi için “tamir operatörü” kullanılmıştır.

Dammak et al. yaptıkları çalışmada belirli kapasitelere sahip dersliklere uygun ve çakışmayan bir sınav atama probleminin çözümünde basit bir sezgisel prosedür geliştirerek başarılı ve temel sonuçlar elde etmiştir [24].

Kazarlis et al. yaptıkları çalışmada, Üniversite zaman çizelgeleme probleminin çözümünde Mikro-GA’lar ve sezgisel yerel arama operatörleri içeren Gelişmiş Genetik Algoritma yöntemi kullanmışlardır [25].

Biroğul tarafından yapılan çalışmada, atölye çizelgeleme probleminin genetik algoritma ile çözümü gerçekleştirilmiştir. Girilen atölye bilgilerine göre gant şeması çizimi yapılmıştır [26].

Head and Shaban yaptıkları çalışmada eş zamanlı ders/öğrenci çizelgelemesi için geliştirdikleri sezgisel fonksiyon tabanlı sistemi, Birleşik Arap Emirlikleri Üniversitesi tarafından denenerek, üniversitenin fiziki ve insan kaynakları optimal kullanıma yakın ve öğrencilerin zamanı en iyi bir şekilde yapılmış olarak öğrenciler ve idare tarafından kabul görmüştür [27].

Güneş ve arkadaşları tarafından yapılan çalışmada, genetik algoritma kullanılarak geliştirilmiş bir yazılım ile çok zaman alan askeri nöbet çizelgesi hazırlama işlemlerinin; harcanan süre sağlanan doğruluk yönü ile eniyilemesi amaçlanmıştır [28].

Daban ve Özdemir tarafından yapılan çalışmada, Genetik Algoritma kullanılarak, öğretim elemanı ve öğrenci verimliliklerine göre ders ağırlıkları alınarak daha etkin bir eğitim öğretim sağlamak için, ders programı hazırlayan ve optimize eden bir program geliştirilmiştir [29]. Yapılan simülasyon sonuçlarında pedagojik esaslara göre, ders ağırlıkları dikkate alınarak hazırlanan ders programlarında, ele alınan kriterlerde başlangıca göre %35–40 oranında iyileşme sağlanmıştır.

Huerta-Amante and Terashima-Marin yaptıkları çalışmada, Kongre çizelgeleme probleminin çözümünde genetik algoritma ve adaptif ceza ağırlıklarını kullanmışlardır [30]. Ceza ağırlıklarının sabitlenmesi yerine kısıtlama türüne göre tekrar ceza ağırlıklarını değiştirerek başarılı sonuçlar elde etmişlerdir.

Kanoh and Sakamoto interaktif zaman çizelgeleme sisteminde bilgi tabanlı genetik algoritma kullanarak yeni bir çözüm geliştirmişlerdir [31]. Problemin çözümünde hem bilgi tabanı hem de kısıtları kullanmaktadır. Öğretim elemanlarının kişisel isteklerini değerlendirerek ve önceki zaman çizelgeleri avantajlarını koruyarak daha verimli çizelgeler elde etmişlerdir.

Özcan ve Alkan tarafından yapılan çalışmada, çok kısıtlı üniversite ders programı hazırlamaya yönelik kararlı hal genetik algoritması (SSGA) geliştirilmiştir [32]. Yeni melezleme mutasyon uzmanları ile paralel genetik algoritmalarından esinlenilerek çok nüfuslu SSGA denenmiştir.

Yu and Sung yaptıkları çalışmada, üniversite haftalık ders çizelgeleme probleminin çözümünde genetik algoritma kullanmışlardır [33]. Lee and Hermosilla yaptıkları çalışmada, genetik algoritma ile kısıtlı sistemi çizelgelemeyi incelemişlerdir [34]. Nedjah and Mourelle yaptıkları çalışmada, ders/zaman çizelgeleme probleminin çözümünde Genetik Algoritma yöntemini kullanmışlardır [35].

Bambrick yaptığı çalışmada, ders zaman çizelgeleme probleminin çözümünde basitleştirilmiş bir genetik algoritma yöntemi kullanmıştır [36].

BÖLÜM 3

GENETİK ALGORİTMALAR

Genetik algoritmalar (GA), doğada gözlemlenen evrimsel sürece benzer bir şekilde çalışan arama ve eniyileme yöntemidir. Karmaşık çok boyutlu arama uzayında en iyinin hayatta kalması ilkesine göre bütünsel en iyi çözümü arar [37].

Genetik algoritmalar problemlerin çözümü için evrimsel süreci bilgisayar ortamında taklit ederler. Diğer eniyileme yöntemlerinde olduğu gibi çözüm için tek bir yapının geliştirilmesi yerine, böyle yapılardan meydana gelen bir küme oluştururlar. Problem için olası pek çok çözümü temsil eden bu küme genetik algoritma terminolojisinde popülasyon (nüfus) adını alır. popülasyonlar vektör, kromozom veya birey adı verilen sayı dizilerinden oluşur. Birey içindeki her bir elemana gen adı verilir. Popülasyondaki bireyler evrimsel süreç içinde genetik algoritma işlemcileri tarafından belirlenirler [37].

Genetik algoritma mühendislik, bilim ve ekonomi gibi çok değişik alanlardaki problemlerde başarıyla uygulanmıştır. Bu çalışmalara; gezgin satıcı, yerleşim, atölye çizelgeleme, ders/sınav çizelgelemesi problemlerinin çözümünü örnek gösterebiliriz. Son yıllarda genetik algoritma, optimizasyon problemlerinin çözümü için büyük önem kazanmıştır. Genetik algoritmanın çalışmaları arasında haberleşme şebekeleri tasarımı, elektronik devre tasarımı, gaz boruları şebeke optimizasyonu, görüntü ve ses tanıma, veri tabanı sorgulama optimizasyonu, uçak tasarımı bulunmaktadır [38].

3.1. GENETİK ALGORİTMANIN ÖZELLİKLERİ

Genetik algoritmayı diğer arama yöntemlerinden (tabu arama, benzetilmiş tavlama gibi.) ayıran en belirgin özellikleri çözüm arama şeklinin farklı oluşudur.

Aşağıda bu farklılıklar açıklanmaktadır [39].

1. GA, parametrelerin kendisiyle değil, doğrudan parametre kodlarıyla uğraşır.
2. GA tek bir noktadan değil, popülasyonun büyüklüğü kadar noktadan arama yapar.
3. GA, ne yaptığını değil, nasıl yaptığını bilir. Yani GA önceden elde edilmiş bilgiyi değil, sadece amaç fonksiyonundan elde edilen bilgiyi kullanır.
4. GA'nın uygulamasında kullanılan operatörler rastlantısal yöntemlere dayanır; belirli ve kesin yöntemler kullanmazlar.

3.2. GENETİK ALGORİTMANIN ÇALIŞMA PRENSİBİ

Genetik algortmada kullanılan evrimle ilgili kavramlar, biyolojideki evrim teorisine benzer anlamda kullanılmaktadır. Doğal yasadaki popülasyonlar bireylerin bir arada bulunmasıyla oluşmaktadır. GA algoritması için oluşturulan popülasyon da çok sayıda bireyin bir araya gelmesiyle, başka bir deyişle çok sayıda olası çözüm adaylarının bir araya gelmesiyle oluşmaktadır. Aday çözümler, probleme uygun şekilde kodlanmış diziler halinde tutulurlar. Bu diziyi oluşturan her bir elemana birey denir. Her bir birey arama uzayında belirli bir bölgeyi temsil eder.

Genetik algortmada ilk başlangıç bireyleri genellikle rasgele olarak üretilirler fakat bu bir zorunluluk değildir. Özellikle çok kısıtlı optimizasyon problemlerinde, başlangıç bireylerini oluşturmak için, tanımlanan kısıtlamaların bir kısmına dikkat edilerek daha iyi adaylar oluşturulabilir. Bireylerin, uygunluk fonksiyonu işlemine tabi tutulması sonucunda, çözümün optimal çözüme ne kadar yaklaştığını değerlendiren uygunluk değeri belirlenir. Başlangıç popülasyonu oluşturulmuş genetik algoritma üç evrim operatörüyle çalışır. Bunlar; seçim, çaprazlama ve mutasyon operatörleridir. Genel olarak bu operatörlerin her biri, yeni nesilde oluşacak olan popülasyonun her bireyine uygulanır.

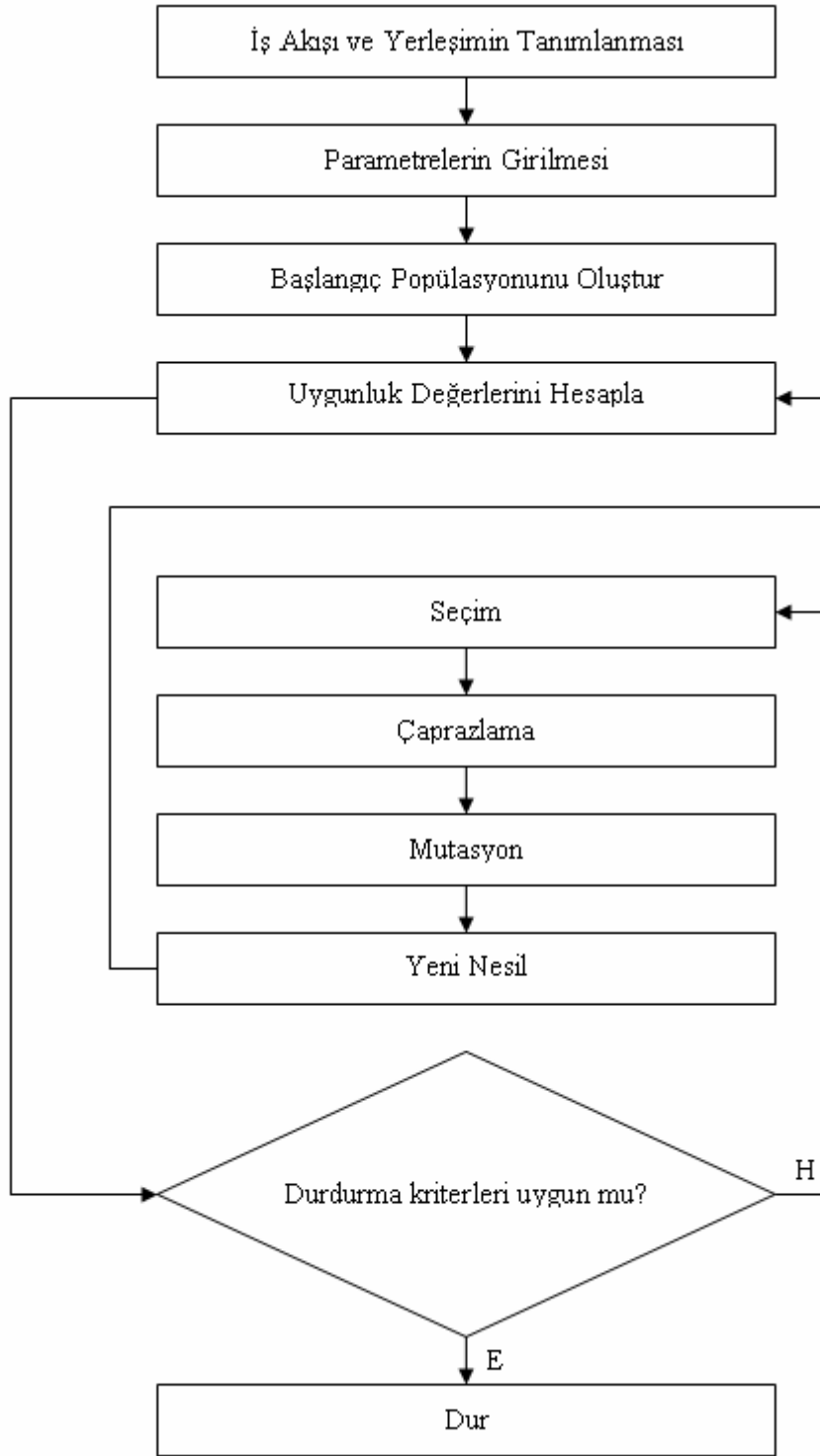
Seçim işlemi, popülasyondaki bireyleri uygunluk değerlerine bağlı olarak, yeni bireyleri oluşturmak için, ebeveyn birey seçmesi işlemidir. Çaprazlama operatörü, seçim işleminden sonra uygulanır ve ebeveyn bireylere ait kromozomların belirli

kısımlarının karşılıklı yer deęiřtirmesini ve böylece yeni özellikte bireylerin oluşmasını ifade eder. Mutasyon işlemi ise yeni oluşan bireyin kromozomlarından herhangi birinin içindeki bir geni mutasyon olasılığına baęlı olarak deęiřtirme işlemidir.

GA parametresinde tanımlanan popülasyondaki toplam birey sayısı kadar birey oluştuęunda, yeni bir nesil meydana gelmiş olur. Bu yeni nesil önceki neslin yerini alır. Her nesilde bireylerin temsil ettięi çözüm deęerlendirilerek, buldukları popülasyonun en iyi bireyi seçilir. Bulunan en iyi birey, önceki nesillerde bulunan en iyi bireylerden daha iyiyse onun yerine geçer.

GA'nın isleyişini etkileyen bir dięer faktör de parametrelerdir. Genel olarak bu parametreler popülasyon büyüklüęü, mutasyon tipi ve olasılığı, çaprazlama tipi ve olasılığı, nesil sayısı ve seçim operatörünün tipidir. Literatürde, bu parametrelerin uygun deęerlerini bulmak için yapılmış teorik veya uygulamaya yönelik pek çok çalışma vardır [40, 41]. Fakat bu çalışmalarda bulunan deęerler sadece belirli problem tiplerine özgüdür.

Genetik algoritma işlemini sonlandırmak için çeşitli yöntemler bulunmaktadır. Bu yöntemler; algoritmanın çalışması esnasında istenen çözüm bulunduęunda, GA'nın başlangıcında tanımlanan toplam iterasyon sayısına ulaşıldığında veya uygunluk deęeri sürekli olarak sabit kaldığında, bulunan en iyi bireyin temsil ettięi çözüm, problem için bulunmuş en uygun çözüm olarak sunulur.



Şekil 3.1. GA'nın genel akış diyagramı.

3.2.1. Genetik Algoritmanın İşlem Adımları

1. Başlangıç: n kromozomdan oluşan rasgele popülasyon oluşturulur (problemin olası çözümleri)
2. Uygunluk: Popülasyondaki her x kromozomu için f(x) uygunluk değeri değerlendirilir.
3. Yeni Toplum: Aşağıdaki adımlar izlenerek yeni bireyler üretilir;
 - a. Seçim: Popülasyon uygunluklarına göre iki ata seçilir (daha uygun olanın seçilmeansı daha fazladır).
 - b. Çaprazlama: Çaprazlama olasılığı ile ataları yeni yavru oluşturmak için birbirleriyle eşleştirilir. Eğer çaprazlama yapılmazsa, yavru ataların tıpatıp aynısı olacaktır.
 - c. Mutasyon: Mutasyon olasılığı ile yeni yavru üzerinde mutasyon işlemi yapılacaktır.
 - d. Kabul: Yeni yavru, yeni topluma eklenir.
4. Değiştir: Yeni popülasyon algoritmanın tekrar işlenmesinde kullanılır.
5. Deney: Eğer bitiş durumu sağlandıysa, durup toplumdaki en iyi çözüm döndürülür.
6. Döngü: Adım 2'ye gidilir [43].

Yukarıda da görüldüğü gibi, genetik algoritmanın akışı oldukça kolaydır. Birçok parametre ve ayar farklı problemler için farklı şekillerde gerçekleştirme için vardır.

3.3. GENETİK ALGORİTMANIN TEMEL KAVRAMLARI

Genetik algoritmada; kısıtlara uyum sağlayan çözüme ulaşmak için algoritma yapısının oluşturulması ve parametrelerin belirlenmesi gerekmektedir. Aşağıda bu kavramlara ve algoritma için gerekli olan parametrelere yer verilmiştir.

3.3.1. Gen ve Kodlama

Yapısında probleme ait en küçük bilgiyi taşıyan birime gen denir. GA'nın kullandığı programlama yapısında bu gen yapıları programcının tanımlamasına bağlıdır.

Bir genin yapısında sadece ikili tabandaki (binary) sayıları içerebileceği gibi, gray, tamsayı, gerçel sayı veya ağaç biçimini ve farklı sembolik ifadeleri de içerebilir [41].

Kodlama biçimi, GA'nın performansını oldukça önemli oranda etkiler; fakat kodlama biçimi programa bağlı olduğundan bütün problemler için geçerli en uygun kodlama biçimini söylemek imkânsızdır. Michalewicz belli bir problem tipi için yapmış olduğu çalışmada gerçel sayı gösteriminin daha çabuk sonuca ulaştığını göstermiştir [41].

3.3.2. Kromozom

Bir veya birden fazla gen yapısının bir araya gelerek problemin çözümüne ait bilgilerin bir kısmını oluşturan dizilere kromozom denir [44]. Kromozom, GA yaklaşımında üzerinde durulan en önemli birim olduğu için bilgisayar ortamında iyi ifade edilmesi gerekir.

3.3.3. Popülasyon

Olası çözüm bilgilerini içeren bireylerin bir araya gelmesiyle oluşan topluluğa popülasyon denir. Popülasyondaki birey sayısı problemin özelliğine göre, genetik algoritmayı tasarlayan tarafından belirlenir.

Popülasyon büyüklüğü problemin çözüm süresini etkilemektedir. Popülasyondaki birey sayısının gereğinden fazla olması çözüm süresini uzatırken, birey sayısının az olması popülasyonun istenen çözüm değerine ulaşamamasına sebep olabilir. Problemin özelliğine göre seçilecek olan popülasyondaki birey sayısı genetik algoritmayı hazırlayan kişi tarafından iyi belirlenmelidir. Grefensette, GA için en uygun popülasyon büyüklüğünün 10 ile 160 birey arasında olmasının uygun olacağını öne sürmüştür [45].

3.4. GENETİK OPERATÖRLER

GA'nın temel yapısını oluşturan ve algoritmanın işleyişi sırasında mevcut popülasyon üzerinde uygulanan işlemlere, genetik operatörler denir. Bu operatörler, seçme (selection) ya da tekrar üreme (reproduction) operatörü, çaprazlama (crossover) operatörü ve mutasyon (mutation) operatörüdür.

Bunlara ilaveten kısıtlı eniyileme problemlerinde mutlak suretle kullanılması gereken ve probleme özgü olarak geliştirilen diğer bir operatör de tamir (reparation) operatörüdür [46]. Genetik operatörler, daha iyi özelliklere sahip nesiller üreterek çözüm uzayını genişletir. Kullanılan üç standart operatör vardır [26]:

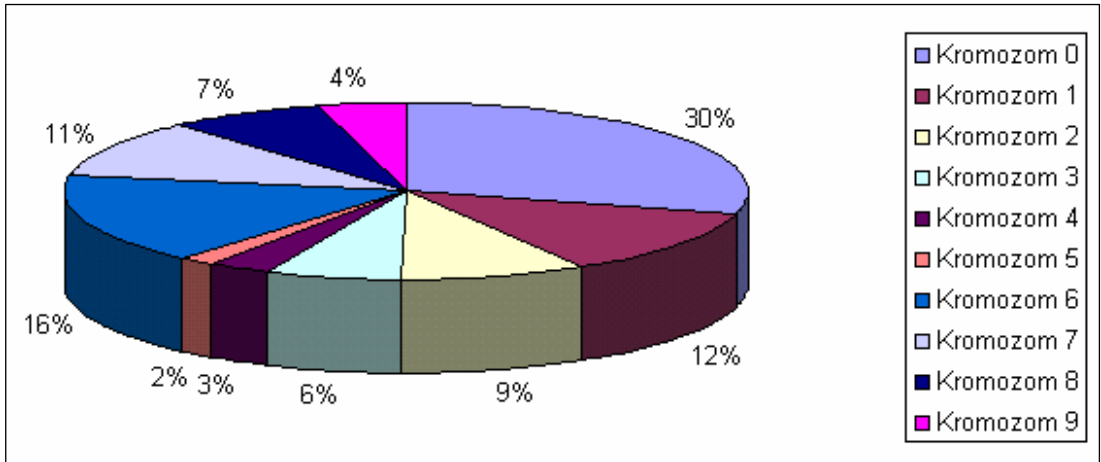
1. Seçim (Selection), var olan bireyi genetik yapısında herhangi bir değişiklik yapmadan yeni nesile kopyalar.
2. Çaprazlama (Crossover), iki bireyin yapılarının rastlantısal olarak birleştirilerek yeni bireyler oluşturulmasıdır. İşlem, ikili dizilerin parçalarının değiş tokuşu ile gerçekleştirilir.
3. Mutasyon (Mutation), var olan bir bireyin genlerinin bir ya da birkaçının yerlerinin değiştirilmesiyle oluşturulur.

3.4.1. Seçim Operatörü

Seçim yöntemleri, genel olarak “en iyi olan yaşar” prensibine dayalıdır. Amaç; yeni nesilde daha yüksek uygunluk değerine sahip kromozomların sayısını arttırmaktır. Uygunluk değeri yüksek olan kromozomların sonraki nesile aktarılma olasılıkları daha yüksek olsa da; seçme işlemi, daha düşük değeriğe sahip kromozomların da seçilmesine olanak sağlayacak şekilde dengeli olmalıdır. Aksi takdirde popülasyon tamamen iyi kromozomlardan oluşabilir. Bu durumda sonraki nesillerdeki kromozomların uygunluk değerlerinin iyileşmesini önleyecek farklılıklar kaybolabilir [38].

3.4.1.1. Rulet Tekerı

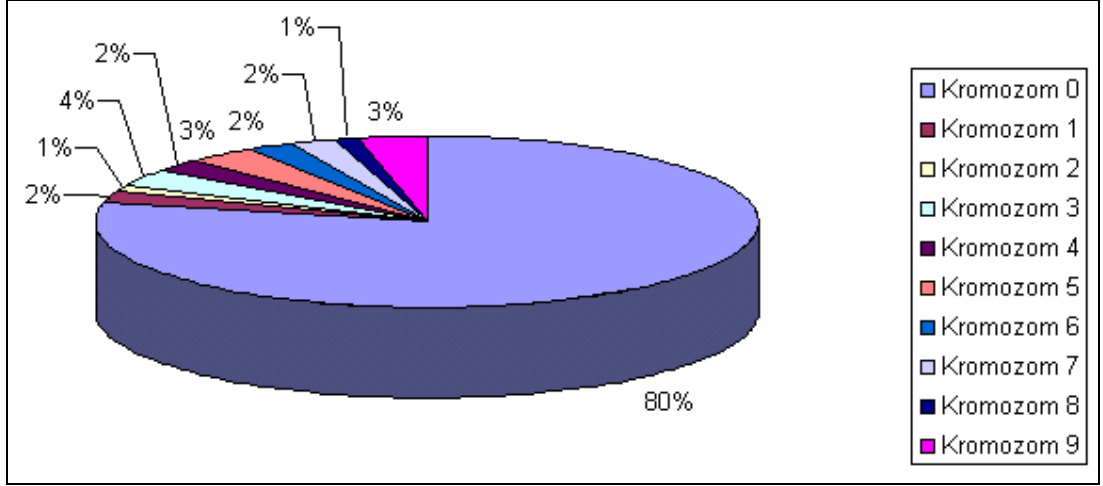
Rulet Tekerı seçim operatöründe, bütün kromozomlar uygunluk değerlerine göre bir rulet etrafında dizilirler. Rulet üzerinde uygunluk değerlerine göre sıralanan kromozomlar rasgele olarak seçilirler. Bu şekilde her birey seçilmek için kendi uygunluk değerine göre bu rulet tekerinden bir pay almaktadır. Daha büyük alana sahip bireyin seçilme şansı daha fazla olacaktır. Bu metot yardımıyla kromozomlar istatistiksel yöntemler kullanılarak uygunluk fonksiyonu değerlerinin toplam uygunluk fonksiyonuna oranları ölçüsünde seçilirler. Ancak bu seçim yönteminde uyum değeri büyük olan bireylerin seçilme olasılığı yüksek olduğu için, hep aynı kromozomların seçilmesine neden olmaktadır. Bu da popülasyon içindeki çeşitliliği etkileyerek sorun yaratır.



Şekil 3.2. Rulet tekeri kromozom dağılım grafiği.

3.4.1.2. Sıralı (Rank)

Popülasyondaki kromozomlar uygunluk değerlerine göre büyükten küçüğe sıralanır. Uygun bir fonksiyonla en yüksek uyumlular havuza daha fazla kopya bırakır, en kötü uyumlular ise kopya elde edememiş olur. Bu yöntemde arama hızı yüksektir. Kromozomun seçilme şansı, sıralanan liste içindeki yerine bağlıdır. Oluşturulan yeni jenerasyonun kromozom sayısı ile önceki jenerasyonun kromozom sayısı aynı alınır.



Şekil 3.3. Sıralamadan önce (uygunluk çizelgesi).

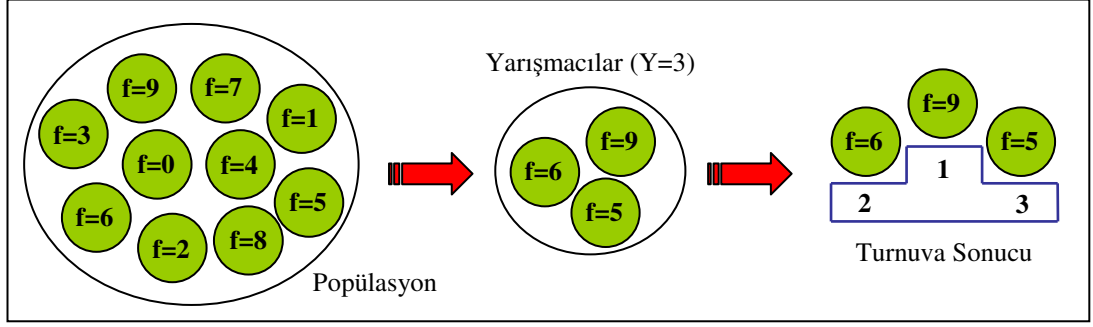
3.4.1.3. Turnuva

Bu seçim yönteminde, bireyler rasgele olarak gruplanır ve gruptaki bireyler aralarında seçim işlemi yapılmak üzere rekabete sokulur. Grup içinde en yüksek uygunluk değerine sahip olan birey, yeni nesli oluşturmak için ebeveyn bireylerden biri olarak seçilir. Bu işlem, toplam birey sayısına ulaşıncaya kadar devam eder.

Bu yöntemde grup büyüklüğü önemlidir ve seçim yönteminin performansını önemli ölçüde etkiler. Bazı uygulamalarda grup büyüklüğü iki olarak seçilirken, bazılarında çok daha büyük gruplar oluşturulur. Turnuva seçim yöntemi, küçük popülasyonlu uygulamalarda, uygunluk değeri orantılı seçim yöntemlerinden daha iyi sonuç verir. Şekil 3.4'te turnuva seçim yöntemi gösterilmiştir. Popülasyondaki bireylerin uygunluk değerleri f ile bireyler arasından rasgele seçilen grup büyüklüğü Y ile gösterilmiştir [45].

3.4.1.4. Sabit Durum

Sabit Durum seçim operatöründe, ebeveynlerin seçimi için kromozomların büyük parçaları bir sonraki jenerasyona taşınmalıdır. Her nesilde yeni bir birey oluşturmak için büyük uygunlukta iyi olan birkaç kromozom seçilir. Daha sonra az uygunlukta bazı kromozomlar atılır ve yeni birey onların yerine getirilir. Popülasyonun geri kalanı değiştirilmeden yeni nesile aktarılır.



Şekil 3.4. Turnuva seçim yöntemi [34].

3.4.2. Çaprazlama Operatörü

Bu operatörde ebeveyn olarak seçilen iki bireyin temsil ettiği çözümlerin belirli parçaları karşılıklı olarak yer değiştirilerek yeni özellikte bireyler oluşturulur.

Genetik algorithmda çaprazlama noktası rasgele seçilir. GA'da geleneksel olarak çaprazlama nokta sayısı birdir. Fakat daha fazla çaprazlama noktası kullanımını teklif eden deneysel çalışmalar vardır [41].

3.4.2.1. Tek Noktalı Çaprazlama

Tek noktalı çaprazlama operatörü, rasgele bir nokta (kesme) seçilir. Her iki kromozomda ilk genden başlanarak seçilen noktaya kadar olan genler alınır ve kromozomlar arasında yer değiştirilir.

$$\begin{array}{lcl}
 \text{P1} & \underline{10010}|110110 & \Rightarrow & \text{C1} & \underline{00111}|110110 \\
 \text{P2} & \underline{00111}|011011 & \Rightarrow & \text{C2} & \underline{10010}|011011
 \end{array}$$

3.4.2.2. İki Noktalı Çaprazlama

Bazı durumlarda tek noktalı çaprazlama yöntemi yetersiz kalabilir ya da büyük parçalı blokların bozulması performansı düşürebilir. Bu sebeple iki noktalı çaprazlama yöntemi tercih edilebilir.

İki noktalı çaprazlama operatörü, rasgele iki nokta seçilir ve bu iki nokta arasında kalan bloklar kromozomlar arasında yer değiştirilir. Bu yöntem popülasyondaki kromozomların performansını arttırabilir. Aşağıda bu çaprazlama yöntemi için bir örnek verilmiştir.

$$\begin{array}{lcl}
 \text{P1} & 100|\underline{101101}|10 & \Rightarrow & \text{C1} & 100|\underline{110110}|10 \\
 \text{P1} & 001|\underline{110110}|11 & \Rightarrow & \text{C1} & 001|\underline{101101}|11
 \end{array}$$

3.4.2.3. Düzgün (Uniform) Çaprazlama

Bir diğer çaprazlama operatörü de, Syswerda tarafından tanımlanan düzgün (uniform) çaprazlamadır. Bunun için kromozomlardaki gen uzunluğunda ve rasgele olarak üretilmiş bir bit dizisi (kalıp) gerekir. Bu kalıp üzerinde “0” bit değerine sahip alanlar çaprazlama için gerekli olan geni birinci ebeveynden, “1” bit değerine sahip alanlar ise ikinci ebeveynden alınarak çaprazlama yapılır. İkinci yavru kromozom için bu durumun tersi geçerlidir [47].

$$\begin{array}{lcl}
 \text{P1} & \underline{10010100} & \Rightarrow & \text{C1} & \underline{11011101} \\
 \text{P2} & \underline{01001101} & \Rightarrow & \text{C2} & \underline{00000100} \\
 & \text{Kalıp} & & & \text{01001011}
 \end{array}$$

3.4.2.4. Yeniden Dizilim Yapısına Sahip Çaprazlama Operatörleri

Bazı optimizasyon problemlerin çözümünde yukarıda bahsedilen çaprazlama operatörleri faydalı olmayabilir. Bu tip problemlerde parametre değerlerinin yanı sıra kullanılan niteliklerin konumu ve düzeni önemlidir. Çizelgeleme Problemi ve Gezgin Satıcı Problemi bu tipteki problemlerdir. Standart çaprazlama operatörleri ile başarı sağlanmamasına rağmen yeniden dizilim sağlayan (re-ordering) operatörlerin bazı formlarında kombinasyonel optimizasyon problemlerine çözüm sağlanabilmektedir.

Yeniden dizilim sağlayan operatörler gen dizilerinin farklı şekilde yerleşimlerine ve değerlendirilmesine imkân sağlar [2].

Kısmen Uyumlu Çaprazlama (PMX)

Goldberg tarafından geliştirilen bu yöntem gezgin satıcı ve araç rotalama problemlerinde yaygın olarak kullanılmaktadır. Bu tip problemlerde dizileri oluşturan genlerin aynı dizi içinde tekrar etmemesi gerekir. PMX tekrarı engelleyen bir yapıya sahiptir. Ayrıca bu yöntemde seçilen bir alt dizinin korunması sağlanarak çocuk diziye aynen aktarılmaktadır. Bu yöntem aşağıda bir örnek üzerinde açıklanmaktadır [48]:

Çaprazlama için seçilen kromozomlar, A ve B olarak adlandırılmıştır ve her biri sekiz adet gen içermektedir.

$$A = 2 \ 8 \ \mathbf{6} \ \mathbf{4} \ \mathbf{5} \ 7 \ 1 \ 3$$

$$B = 8 \ 7 \ \mathbf{2} \ \mathbf{1} \ \mathbf{3} \ 4 \ 6 \ 5$$

A ve B kromozomlarından ilk olarak ortak bir aralık rastsal bir şekilde seçilir. Daha sonra, seçilmiş iki aralıktaki elementlerin değişim planları belirlenir. Bu örnekte, seçilmiş aralıklar arasındaki plan 6'ya 2; 4'e 1 ve 5'e 3'tür. Daha sonra, A ve B kromozomlarındaki seçilen genler karşılıklı olarak değiştirilir. Bu işlem sonrasında kromozomlardaki genler tekrarlı olarak yer aldığından her iki yapının da uygun olmadığı görülmektedir. Buradan elde edilen kromozomlar geçici sonuçlar olarak değerlendirilir.

$$A' = \underline{2} \ 8 \ \mathbf{2} \ \mathbf{1} \ \mathbf{3} \ 7 \ \underline{1} \ \underline{3}$$

$$B' = 8 \ 7 \ \mathbf{6} \ \mathbf{4} \ \mathbf{5} \ \underline{4} \ \underline{6} \ \underline{5}$$

Bundan dolayı, yeni yapıları uygun olmayan A' ve B' kromozomlarında, değişim planının uygulanması gerekir. Bu örnekte, A' kromozomunun 1, 7 ve 8. genlerinde yer alan; 2, 1 ve 3 değerleri sırasıyla 6, 4 ve 5 değerleriyle değiştirilir. B' kromozomunda ise 6, 7 ve 8. genlerindeki 4, 6 ve 5 değerleri sırasıyla 1, 2 ve 3 değerleriyle değiştirilir. Yavru kromozomlar aşağıdaki gibi oluşmaktadır.

$$A_Y = 6 \ 8 \ 2 \ 1 \ 3 \ 7 \ 4 \ 5$$

$$B_Y = 8 \ 7 \ 6 \ 4 \ 5 \ 1 \ 2 \ 3$$

Periyodik Çaprazlama

Bu çaprazlama tipinde işlemler belli bir periyoda göre yapılır. Aşağıdaki örnekte bu durum açıkça gösterilmiştir. Bunun için bir ebeveynde belirli genler sabit konumlarını korurken diğer genler ikinci ebeveyndeki aynı türdeki genlerle yer değiştirir.

$$\begin{array}{l}
 P1 \quad 1 \ \underline{2} \ 3 \ \underline{4} \ 5 \ 6 \ \underline{7} \ 8 \quad \Rightarrow \quad C1 \quad 1 \ \underline{8} \ 3 \ \underline{2} \ 7 \ 6 \ \underline{5} \ 4 \\
 P2 \quad 3 \ \underline{8} \ 6 \ \underline{2} \ 7 \ 1 \ \underline{5} \ 4 \quad \Rightarrow \quad C2 \quad 3 \ \underline{2} \ 6 \ \underline{4} \ 5 \ 1 \ \underline{7} \ 8
 \end{array}$$

Örneğe göre P1’de ve P2’de işaretli olan gen değerleri sabit noktaları oluşturur. P1’de 1, 3, 6 konumlarındaki yine, 1, 3 ve 6 değerleri ile P2’de aynı konumda yer alan 3, 6 ve 1 değerleri sabit genlerdir. O halde bunlar dışında kalan genler sırasıyla birbirleriyle yer değiştirir [2].

Sıralı Çaprazlama (OX)

Bu yöntem, Davis, Goldberg ve Lingle tarafından geliştirilmiştir. Bu yöntemde, gen havuzundan rastsal olarak iki kromozom seçilir. Bu kromozomlar üzerinde yine rastsal olarak iki ayrı kesim noktası belirlenir. Bu kesim noktaları arasındaki kromozom sayısının her iki kromozomda da aynı olmasına dikkat edilir. Kesim noktaları arasındaki kromozomlar karşılıklı olarak yer değiştirilir. Kesim bölgesi dışında yer alan genler içerisinde tekrarlı genler oluşursa bunlar yerine sıra ile soldan sağa doğru kromozomda bulunmayan genler yazılır. OX yönteminin işleyişi aşağıda A ve B şeklinde kodlanmış olan iki kromozom üzerinde gösterilmektedir [48]:

$$A = 9 \ 8 \ 4 \ 5 \ 6 \ 7 \ 1 \ 2 \ 3 \ 0$$

$$B = 8 \ 7 \ 1 \ 2 \ 3 \ 0 \ 9 \ 5 \ 4 \ 6$$

Sıralı çaprazlama yöntemine göre, 5, 6 ve 7 genleri yerine; 2, 3 ve 0 genleri atanır ve A kromozomunda daha önce 2, 3 ve 0 bulunan yerlere X yazılır, buna göre aşağıdaki ifade elde edilir:

$$\begin{aligned} A &= 9 \ 8 \ 4 \ 2 \ 3 \ 0 \ 1 \ \mathbf{X} \ \mathbf{X} \ \mathbf{X} \\ B &= 8 \ \mathbf{X} \ 1 \ 5 \ 6 \ 7 \ 9 \ \mathbf{X} \ 4 \ \mathbf{X} \end{aligned}$$

X yerine dizide olmayan işler eklendiğinde yeni kromozomlar aşağıdaki sonuçlar elde edilir:

$$\begin{aligned} AI &= 9 \ 8 \ 4 \ 2 \ 3 \ 0 \ 1 \ 5 \ 6 \ 7 \\ BI &= 8 \ 2 \ 1 \ 5 \ 6 \ 7 \ 9 \ 3 \ 4 \ 0 \end{aligned}$$

Sıraya Dayalı Çaprazlama (OBX)

Bu yöntem ilk olarak Davis tarafından önerilmiştir [49]. OBX yöntemi kullanıldığında bir ebeveynden seçilen konumlardaki gen dizisi diğer ebeveyndeki aynı değerli genlerin üzerine konulur. Bunun için rasgele olarak belirlenen bir kalıp gereklidir. Aşağıdaki örnekte yer alan kalıp üzerindeki değeri 1 olan genler önem taşımaktadır. Bu genlerin konumlarına bağlı olarak P2'den sırasıyla 7, 2 ve 3 değerleri çaprazlanacak genler olarak seçilmiştir. P2'de belirlenen bu 7, 2 ve 3 değerleri, P1 üzerinde bulunan sıraya dizilmiş şekli olan 2, 3 ve 7'nin bulunduğu konumlara aynı sırada yerleştirilir. Böylece C1 oluşturulmuş olur. Aynı kalıba göre P1 üzerinde 1, 5 ve 7 değerleri seçilmiştir. P2'de bu değerlerin bulunduğu konumlara 1, 5 ve 7 genleri transfer edilir. Bu şekilde de C2 oluşur. Bu tür çaprazlama kromozomu oluşturan karakterlerin sayı ve sıralarının önem taşıdığı durumlarda kullanılır [50].

$$\begin{array}{rcccccccc} \mathbf{P1} & \mathbf{1} & \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{5} & \mathbf{6} & \mathbf{7} & \mathbf{8} & \Rightarrow & \mathbf{C1} & \mathbf{1} & \mathbf{\underline{7}} & \mathbf{\underline{2}} & \mathbf{4} & \mathbf{5} & \mathbf{6} & \mathbf{\underline{3}} & \mathbf{8} \\ \mathbf{P2} & \mathbf{7} & \mathbf{4} & \mathbf{6} & \mathbf{1} & \mathbf{2} & \mathbf{8} & \mathbf{3} & \mathbf{5} & \Rightarrow & \mathbf{C2} & \mathbf{\underline{1}} & \mathbf{4} & \mathbf{6} & \mathbf{\underline{5}} & \mathbf{2} & \mathbf{8} & \mathbf{3} & \mathbf{\underline{7}} \\ & & & & & & & & & & \mathbf{Kalıp} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{0} \end{array}$$

Doğrusal Sıralı Çaprazlama (LOX)

Falkenauer ve Bouffouix tarafından geliştirilmiştir. Dairesel (CX) çaprazlamanın bir çeşididir. İşlem adımları aşağıda verilmektedir [49]:

1. Mevcut popülasyon içerisinden rastsal olarak iki ebeveyn seç,
2. Seçilen bu iki dizi (kromozom) üzerinde rastsal olarak iki alt dizi seç (P_1, P_2),
3. P_1 dizisinden seçilen alt diziyi kromozomdan kopar ve boş kalan yerleri belirle, benzer şekilde P_2 dizisinde de aynı işlemleri gerçekleştir,
4. Birinci alt diziyi P_1 'e ve ikinci alt diziyi P_2 'ye yerleştir.

Konuma Dayalı Çaprazlama (PBX)

Bu yöntemde binary formdaki bir kalıp sabit konumları belirlemede kullanılır. P_1 'deki bu konumlarda var olan genler C_1 içine kopyalanırken, bu genler dışında kalan genler P_2 'de aynı kalıbın seçtiği genlerle doldurulur. Aynı işlem C_2 içinde uygulanır [47]. Aşağıda bu çaprazlama açıkça gösterilmiştir.

$$\begin{array}{rcl} \mathbf{P1} & \mathbf{1\ 0\ 1\ 1\ 0\ 0\ 1} & \Rightarrow \mathbf{C1\ 1\ 0\ 1\ 1\ 1\ 0\ 1} \\ \mathbf{P2} & \underline{\mathbf{0\ 0\ 1\ 0\ 1\ 0\ 0}} & \Rightarrow \mathbf{C2\ 0\ 0\ 1\ 0\ 0\ 0\ 0} \\ & \mathbf{Kalıp\ 1\ 1\ 0\ 1\ 0\ 0\ 1} & \end{array}$$

Dairesel Çaprazlama (CX)

Davis, Goldberg ve Lingle tarafından geliştirilmiş bir yöntemdir. Bu yöntemde ilk kromozomdan en baştaki gen seçilir ve bu gen yeni diziyeye yerleştirilir. Bu gene karşılık gelen ikinci kromozomdaki gen belirlenir bu değer de yeni kromozom üzerine yerleştirilerek dairesel bir şekilde bütün genler belirlenir [39]. İşlem adımları aşağıdaki örnekte görülmektedir [48]:

İki ayrı ebeveyn (kromozom) C ve D olarak kodlanmış olsun,

$$\begin{aligned} \mathbf{C} &= 9 \ 8 \ 2 \ 1 \ 7 \ 4 \ 5 \ 0 \ 6 \ 3 \\ \mathbf{D} &= 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8 \ 9 \ 0 \end{aligned}$$

Çaprazlama haritasında C kromozomundan (ilk bireyden) en sondaki değer olan 9 seçilir. C bireyinden seçilen 9'un karşılığı, D bireyinde 1 olmaktadır; C'de 1 geni yerine yazılır, 1'in karşılığı D'de 4 olmaktadır, 4 geni yerine yazılır ve aşağıdaki ifade elde edilir:

$$\mathbf{CI} = 9 \ _ \ _ \ 1 \ _ \ 4 \ _ \ _ \ 6 \ _$$

Kalan boşluklara diğer genler yazılarak yeni kromozomlar elde edilir:

$$\begin{aligned} \mathbf{CI} &= 9 \ 2 \ 3 \ 1 \ 5 \ 4 \ 7 \ 8 \ 6 \ 0 \\ \mathbf{DI} &= 1 \ 8 \ 2 \ 4 \ 7 \ 6 \ 5 \ 0 \ 9 \ 3 \end{aligned}$$

3.4.3. Mutasyon Operatörü

Genetik algoritmada çözüm, belli bir nesil sayısından sonra popülasyon içerisindeki bireyler gitgide birbirlerine benzemektedir. Bu durum çözüm uzayının daralmasına neden olmaktadır. Bireylere ne kadar çaprazlama operatörü uygulansa da belli bir nesil sayısından sonra birey çeşitliliği sağlanmamaktadır. Bu durumda bireyi oluşturan genlerden rasgele bir tanesi seçilir. Rasgele seçilen genin değeri değiştirilir. Böylelikle popülasyon içindeki bireylerin çeşitliliğinin devamı sağlanmış olunur [38].

Genetik Algoritmalarda, popülasyonda çeşitliliğin sağlanması için mutasyondan faydalanılır. Mutasyon problemin herhangi bir yerel optimum noktaya yakınsamasını engelleyebilmektedir. Mutasyon oranları genellikle düşük tutulmaktadır. Bunun nedeni çaprazlama sonucu elde edilen uyum değeri yüksek dizileri kaybetmemektir. Mutasyonda genlerden biri rastlantısal olarak değiştirilir. Genellikle kullanılan mutasyon oranı mevcut genlerin sayısının % 1 kadardır [51].

3.4.3.1. Değer Değiştirme Yöntemi

Basit genetik algorithmanda dizideki herhangi bir konumun değerinin 1 iken 0 yada tersini yaparak gerçekleştirilir.

$$P1 = 1,0,0,1,1,\underline{1},1,0$$

$$P1' = 1,0,0,1,1,\underline{0},1,0$$

3.4.3.2. Kaydırma Yöntemi

Dizi içerisinde rastsal olarak belirlenen bir blok genin yine aynı dizi içerisinde rasgele bir konuma yerleştirilmesidir.

$$P1 = AKDEM\underline{Z}IF$$

$$P1' = A\underline{E}MZIKDF$$

3.4.3.3. Yerleştirme Yöntemi

Dizi içinde rasgele bir genin seçilerek, yine aynı dizi içinde rasgele başka bir konuma yerleştirilmesidir.

$$P1 = A\underline{K}DEMZIF$$

$$P1' = ADEMZ\underline{K}IF$$

3.4.3.4. Karşılıklı Değişim Yöntemi

Dizi içerisinde Rastsal olarak iki gen seçilmekte ve bu iki genler birbiriyle yer değiştirmektedir.

$$P1 = A\underline{K}DEM\underline{Z}IF$$

$$P1' = A\underline{Z}DEM\underline{K}IF$$

3.4.4. Durdurma Kriteri

Üreme, Çaprazlama ve Mutasyon işlemlerinden sonra yeni bir nesil oluşmaktadır. Tüm bu işlemler sonsuz döngü içerisinde yapılır. Eğer bir durdurma kriteri belirlenmez ise bu süreç sonsuza kadar devam eder. Durdurma kriteri olarak şu yöntemler kullanılabilir.

3.4.4.1. Hesaplama Zamanı Kriteri

Bu yöntemde önceden bir hesaplama zamanı veya döngü sayısı belirlenmekte, bu zaman veya döngü sayısına ulaşıldığında durdurulmaktadır. Bu yöntemde belirlenen döngü sayısı gerektiğinden fazla ya da eksik olabilir.

3.4.4.2. Optimizasyon Hedefi Kriteri

Önceden ulaşılması istenen amaç fonksiyonu değeri bilinmektedir. Uyum değeri bu değere ulaştığında algoritma durdurulmaktadır.

3.4.4.3. Minimum iyileşme kriteri

Genetik algoritma problemlerinde bulunan en iyi çözümler önce hızlı daha sonra yavaş yavaş artış göstermektedir. Bulunan değerlerdeki iyileşme hızının giderek azalması ve sifıra yaklaşması, artık daha fazla iyileşme beklenmemesi gerektiğini gösterebilir. Çözüme harcanacak zaman ile çözümden beklenecek kalite arasında bir denge kurularak durdurma gerçekleştirilir.

BÖLÜM 4

PROBLEMİN ÇÖZÜMÜNE UYGUN GELİŞTİRİLEN MODEL

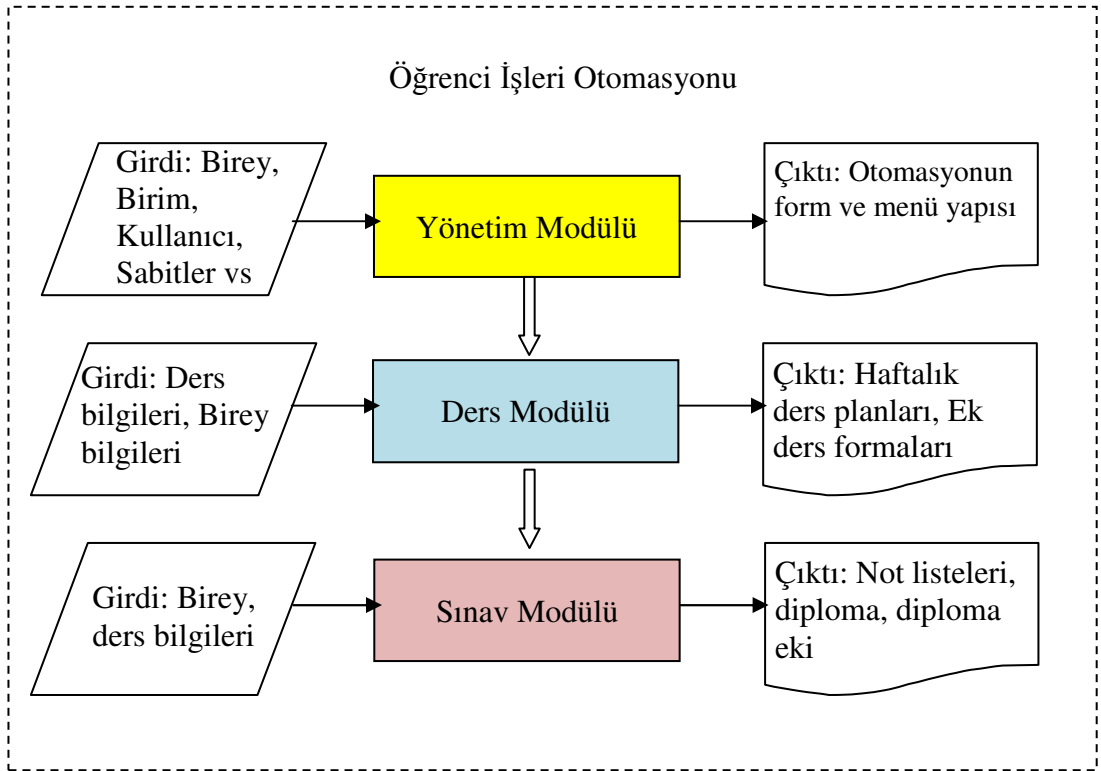
Öğrenci işleri otomasyonu, bir üniversitede kullanılan en önemli otomasyon sistemlerinden biridir. Her akademik dönemde binlerce öğrenci ve yüzlerce öğretim elemanı tarafından gerçekleştirilen ders alma, ders değerlendirme, not girişleri, mezuniyet işlemleri vb. süreçler bu otomasyon üzerinden yapılmaktadır.

Bu çalışma kapsamında gerçekleştirilen çizelgeleme problemi öğrenci işleri otomasyonu içerisinde kullanılmaktadır. Geliştirilen algoritmanın gerçek verileri kullanabilmesi ve pratik hayata katkısının sağlanabilmesi için algoritmanın ihtiyaç duyduğu verilerin girilmiş olması ve algoritmanın ürettiği sonuçların raporlanması gerekmektedir. Bu amaçla geliştirilen modelin entegre edileceği otomasyonun da programlanmış olması gerekmektedir. Aksi takdirde geliştirilen modelin pratik sonuçlarından istifade edilmesi mümkün olmamaktadır. Bu amaçla algoritmanın kullanılacağı Öğrenci İşleri Otomasyonu programlanıp yazılıma eklenmiş ve algoritma ile bütünlüğü sağlanmıştır (Şekil 4.1).

The screenshot shows the KBU Technology system interface. The top navigation bar includes the university logo and name, and language options for English and Turkish. The user is logged in as 'Oğr. Gör. Abdullah ELEN'. The main content area is titled 'BİREY İŞLEMLERİ' (Individual Operations) and contains a form for entering personal information. The form fields are: 'TC Kimlik No:', 'Ad:', and 'Soyadı:'. A 'Bireyi Kaydet' button is located at the bottom right of the form. The sidebar menu on the left lists various system management options under 'SİSTEM YONETİMİ' and 'Kullanıcı İşlemleri'.

Şekil 4.1. Geliştirilen öğrenci işleri otomasyonunun arayüz görünümü.

Öğrenci işleri otomasyonu üç modülden oluşmaktadır (Şekil 4.2). Bunlar Yönetim Modülü, Ders Modülü ve Sınav Modülüdür. Bu modüllerin veritabanı yapısı oluşturulup tablolar arasındaki ilişkileri (ER-Diyagramı) kurulmuştur (Ek I). Microsoft ASP.NET platformunda C# programlama dili kullanılarak hazırlanmış ve SQL Server veritabanı üzerine veriler kaydedilmiştir. Algoritmanın oluşturulduğu kodlar ise Windows Application uygulaması olarak hazırlanmıştır. Yazılım nesne tabanlı olarak, dinamik bir menü sistemi kullanılarak geliştirilmiştir. Verileri sorgulamada kullanılan kodlar, kullanım esnasında fazla sayıda kullanılıyor ise veritabanı içerisinde Stored Procedure olarak, az sayıda kullanılıyor ise nesne olarak kodlanmıştır. Kod miktarını en aza indirmek için tüm alt işlemler nesne yapısında dinamik parametreler kullanılarak programlanmıştır.



Şekil 4.2. Geliştirilen öğrenci işleri otomasyonunun genel yapısı.

4.1. ÖĞRENCİ İŞLERİ OTOMASYONUNUN MODÜLLERİ

4.1.1 Yönetim Modülü

Yönetim modülü, programın kayıt sistemine ait temel verilerin kaydedildiği, kullanıcı arayüz ve yönetim kısımlarının bulunduğu bölümdür. Bu modül kendi içerisinde Birimler, Bireyler, Kullanıcılar, Menüler, Sabitler, Derslikler ve Akademik Takvimler gibi alt bölümlerden oluşmaktadır. Her bir bölümün hangi bilgileri içerdiği ve içerisinde bulunan tablolar hakkında aşağıda bilgiler sunulmuştur.

1. Birimler Bölümü: Bu bölüm içerisinde sadece Birimler Tablosu bulunmaktadır. Bu tablo içerisine üniversitedeki Rektörlük, Fakülte, Bölüm, Anabilim Dalı, Daire Başkanlıkları vs gibi kurumsal alt yapıyı oluşturan görev tanımları yapılmış tüm birimlerin bilgileri kaydedilmektedir.
2. Bireyler Bölümü: Programda her kişi bir birey olarak tanımlanmıştır. Her bireyin kimlik ve iletişim bilgileri ayrı tablolarda tutulmuştur. Birey ile Birim arasında bağlantı tablosu kurularak Bireyin hangi birimde hangi görevde olduğu tespit edilebilmektedir.
3. Kullanıcı Bölümü: Her bireyin sisteme giriş yapabilmesi için kullanıcı tanımları yapılmış olmalıdır. Kullanıcılar ise kendi içinde gruplara ayrılmış ve buna göre yetkilendirmeleri yapılmıştır.
4. Menüler Bölümü: Sistemin arayüz görünümünü oluşturan menü yapısı hiyerarşik olarak 3 aşamalı olarak oluşturulmuştur. Bunlar modüller, Ana menüler ve Alt menülerdir. Bu tablolar kullanıcılar bölümü ile ilişkilendirilmiştir.
5. Diğer Bölümler: Yukarıda verilen bölümlerin dışında programın değişik yerlerinde ihtiyaç oldukça kullanılan Derslikler, Akademik Takvim, Sabitler gibi temel tablolarda bulunmaktadır.

4.1.2 Ders Modülü

Bu modül kendi içerisinde bir Anabilim Dalı açıldığında o Anabilim Dalında müfredat oluşturma, ders açma, Öğretim Elemanı atama gibi temel işlemlerin yürütüldüğü modüldür. Modülün alt bölümleri maddeler halinde aşağıda açıklanmıştır.

1. Müfredat oluşturma bölümü: Tüm üniversitede açılacak olan dersler bir kez kaydedilebilmesi için öncelik Ders Havuzunda oluşturulmaktadır. Daha sonra bu havuzdan çekilen derslerle Anabilim dallarının müfredat programları oluşturulmaktadır.
2. Ders açma bölümü: Her dönem açılan dersler müfredat tablosundan süzülerek çıkarılan veriler üzerinden açılmaktadır. Bu aşamada açılan dersler şubeler oluşturularak (öğrenciler bölünerek) açılabilmektedir.
3. Öğretim elemanı atama bölümü: Açılan şubesiz ya da şubeli dersler kendi içinde Teorik, pratik olarak da bölünüp her birine ayrı ayrı öğretim elemanı atanabilmelidir. Bu işlemler bu bölüm içerisinde yapılmaktadır.
4. Çizelgeleme algoritması: Öğretim elemanları da atandıktan sonra artık derslerin hangi gün ve hangi saatlerde nerede işleneceği bilgisinin hiç bir ders çakışmadan yapılabilmesi gerekmektedir. Bu konu bu tez kapsamında ele alınan ana problem olarak çözülmüştür. Geliştirilen genetik algoritmaya bağlı olarak tüm dersler otomatik olarak yerleştirilmektedir. Detayları bir sonraki bölümde verilmiştir.
5. Haftalık ders programı ve ek ders: Dersler çizelgeleme algoritması ile otomatik olarak yerleştirildikten sonra bu verilere bağlı olarak panolara asılacak tablolar, derslik kapılarına asılacak tablolar, öğretim elemanı görevlendirmeleri gibi yüzlerce çıktı otomatik olarak oluşturulmaktadır. Bu çıktılardan farklı olarak ayrıca her öğretim elemanın ek ders form ve yan formları da otomatik olarak oluşturulmaktadır.

4.1.3 Sınav Modülü

Açılan dersler ve bunlara atanacak öğretim elemanları belirlendikten sonra artık öğrencilerin bu dersleri alıp ona göre sınav tablolarının oluşturulması gerekmektedir. Sınavları yapıldıktan sonra öğrenciye ait çıktıların alınabilmesi gerekmektedir. Bu modülde bu işlemlere ait uygulamalar bulunmaktadır. Alt bölümleri aşağıda maddeler halinde sunulmuştur.

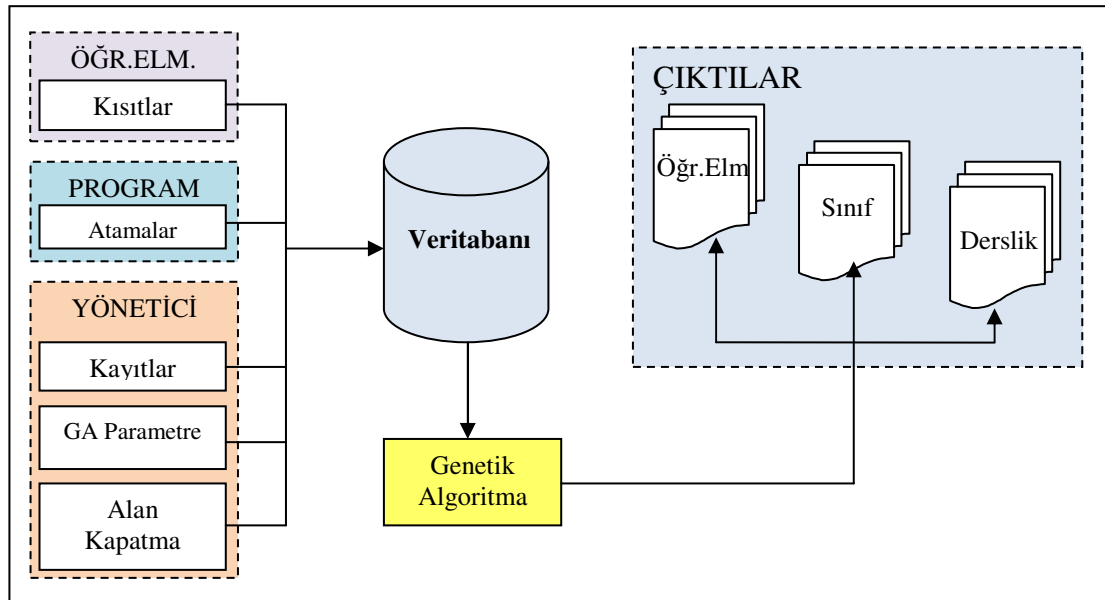
1. Ders alma bölümü: Bu bölümde bir önceki modülde açılan dersleri öğrenciler almaktadır. Her öğrencinin kendi durumuna bağlı olarak alacağı dersler otomatik olarak karşısına çıkarılmaktadır. Bu bölüm altında aynı zamanda öğrencilere danışman atamaları da yapılmaktadır.
2. Sınav açma bölümü: Bu bölümde her dersin sınavları otomatik olarak oluşturulmaktadır. Sınavların yapılacağı gün ve saatler girilen belli parametrelere bağlı olarak (örn: hangi dersler ortak sınav olarak yapılacak) otomatik olarak belirlenmektedir. Aynı zamanda otomatik gözetmen ataması da yapılmaktadır.
3. Sınav not girişi bölümü: Sınavlar yapıldıktan sonra notların girileceği tablolara bu bölümde doldurulmaktadır. Vize, final ve bütünleme not girişleri tamamlanınca sistem otomatik olarak bu notları değerlendirip harf notuna dönüştürmekte ve kalıcı değerlendirme tablolarında saklamaktadır. Daha sonra bu tablolara mezuniyet durumuna gelen öğrenciler için Diploma, Diploma eki, transkript gibi dokümanları otomatik olarak oluşturmak için kullanılmaktadır.

4.2. TASARLANAN GENETİK MODEL

Problemin çözümünde kullanılacak olan Genetik Model, algoritmanın kullanılacağı Üniversitedeki ihtiyaç ve kısıtlamaları karşılayacak şekilde olmalıdır. Her Üniversitenin kendine ait özel ihtiyaçları, öncelikleri ve bazı ön koşulları bulunabilir. Bu sebeple çözülecek olan problemin modellemesi de bu ihtiyaçları karşılayacak düzeyde olmalıdır.

Geliştirilen model tasarımı Karabük Üniversitesi'ndeki ihtiyaçlar göz önüne alınarak yapılmıştır. Üniversitedeki gerçek veriler üzerinde çalışacak GA modeli tasarlanmıştır.

Algoritmaya veri girişleri Birimler (Akademik Programlar), Öğretim Elemanları ve Yönetici tarafından yapılmaktadır. Birimler, hangi derse hangi Öğretim Elemanın atanacağını, dersin işlenebileceği dersliklerin atamasını yapmaktadır. Öğretim elemanları ise haftanın hangi günleri ve saatlerinde derse girmek istediğine dair kısıtları girmektedir. Yönetici ise, genetik algoritmanın parametrelerini, zorunlu ders konulmayacak alanları ve akademik birimlere ait ders ve öğretim elemanlarının bilgilerini girmektedir.



Şekil 4.3. Geliştirilen modelin genel yapısı.

4.2.1. Kromozom ve Gen Yapısı

Genetik algoritmada kullanılacak olan kromozomların uzunluğu, aynı anda yerleştirilecek olan derslerin sayısına bağlıdır. Bir seferde yerleştirilecek olan dersler için tüm üniversitedeki dersleri ya da sadece fakülte içindeki dersleri ele alabiliriz. Bir üniversitedeki tüm dersler aynı anda çözülmeye çalışıldığında kromozomun uzunluğu binlerle ifade edilebilecek gen sayısına ulaşmaktadır.

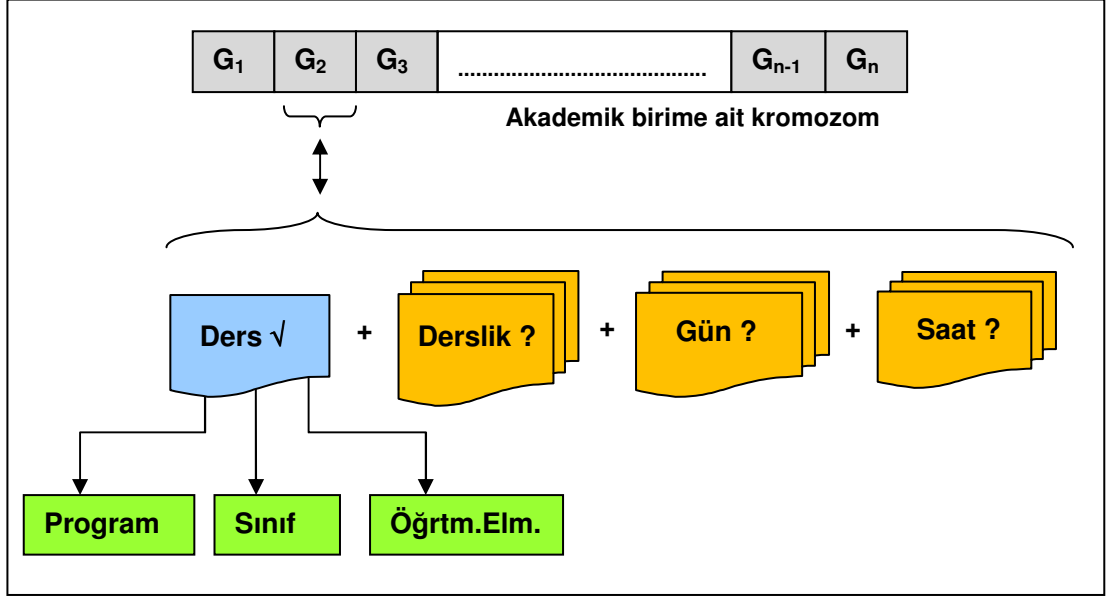
Bu ise çözüm uzayını, işlem zamanı artırmakta ve hafıza kaynaklarını yetersiz hale getirmektedir. Bu sebeple aynı anda yerleştirilecek olan dersler için Fakülte içindeki dersler ele alınmıştır.

Bir Fakülte'deki dersler yerleştirildikten sonra diğer fakülte'deki derslerin yerleştirilmesi gerekir. Her iki fakülte'de derse giren Öğretim Elemanı için birinci fakülte yerleri kesinleşen derslerin gün ve saat yerleri ikinci Fakülte'de kapatılmış alan olarak kullanılmaktadır. Böylece her iki fakülte arasındaki oluşabilecek çakışmalar engellenmiş olmaktadır.

İlk yerleştirilen fakültenin uygunluk değeri başlangıç kısıtları daha az olduğundan sonrakilere nazaran daha yüksek olmaktadır. Bu ise fakülteler arasında adaletsizlik oluşturabilmektedir. Bunu engellemek için Fakültelerin yerleştirme sırası rasgele yapılabilir ya da her dönem için belli bir sıraya konulabilir.

Fakülte'deki her bir ders, Kromozom içerisinde bir geni temsil etmektedir (Şekil 4.4). Bir gen ise 4 adet bilgiyi barındırmaktadır. Bu bilgiler Ders bilgisi, Derslik bilgisi, Gün bilgisi ve Saat bilgisidir. Ders bilgisi mevcut veritabanı içerisinde bilinen ve seçilen bir bilgidir. Bu bilgi seçildiğinde derse ait Program, Sınıf (öğrenciler) ve Öğretim Elemanı veritabanından direk olarak okunabilmektedir. Fakat ilgili dersin hangi dersliğe, hangi gün ve saate konulacağı ise bilinmemektedir. Bu üç alanın Genetik Algoritma tarafından çözülmesi istenmektedir.

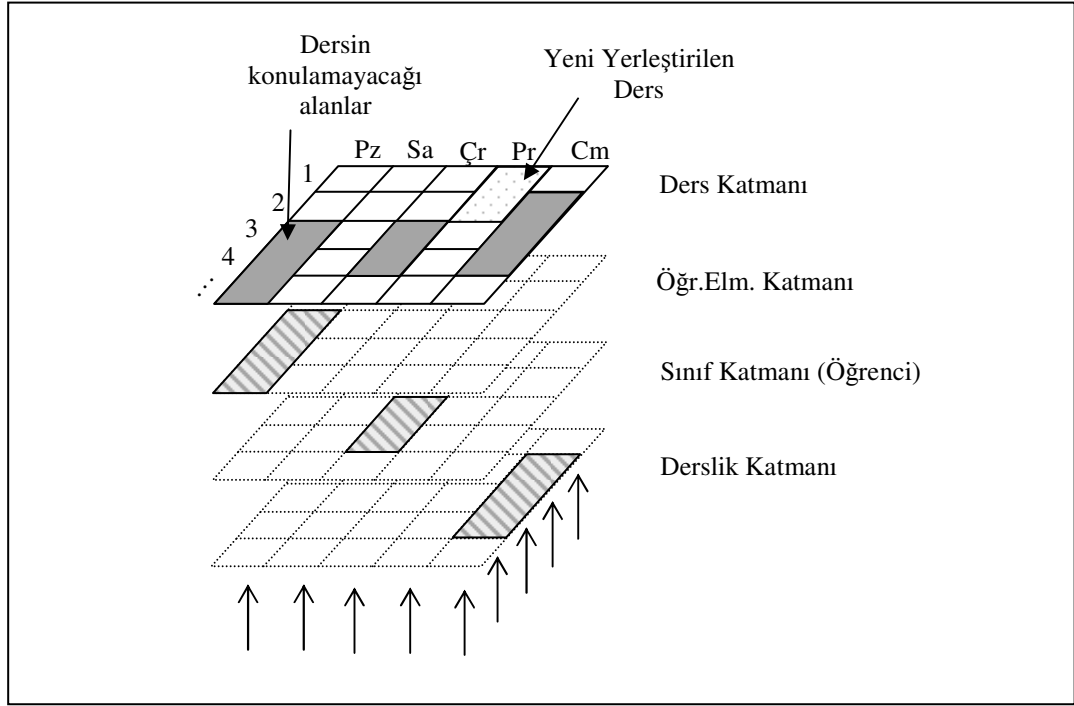
Oluşturulan her bir kromozom içerisindeki genlerin yeri (derslerin yeri) sabit konum olarak belirlenmiştir. Örneğin 5 numaralı ID değerine sahip ders, kromozom içerisinde 23 numaralı konumda bulunuyorsa, tüm genetik işlemler boyunca (çaprazlama, mutasyon vb.) aynı konumda bulunacaktır. Genetik operatörler, genin içerisindeki Derslik, Gün ve Saat alanlarında uygulanmaktadır.



Şekil 4.4. Kromozom ve gen yapısı.

Gen kodlamasında gerçek sayılar kullanılmaktadır. Bu sayılar ilgili alanın veritabanındaki ID (Identification) bilgisinden gelmektedir. Örneğin Derslere ait Teori, Pratik saat ve Sınıf bilgileri GNL_DersOgretimElemaniAtama tablosundan, Öğretim Elemanı bilgileri GNL_Bireyler tablosundan, Dersliklere ait bilgiler GNL_Derslikler tablosundan gelmektedir. Gün (1: Pazartesi, 2: Salı, ..., 5: Cuma) ve saat değerleri (1 ile 10 arasında) rakamlarla ifade edilmektedir. Genin içerisindeki Saat alanına dersin ilk saati kaydedilmektedir. İkinci öğretimler için tüm uygulamalar aynı olmakta, sadece saat sabitleri çizelgelemede farklı alınmaktadır.

Kromozom içerisinde Gen olarak yer alan bir dersin çakışma durumunu kontrol ederken, derse ait Öğretim Elemanın, Sınıfın (Öğrencilerin) ve Dersliğin Haftalık tablolarına bakılır. Bu üç tablonun iz düşümü Dersin konulacağı Tabloya düşürülerek çakışma durumu kontrol edilir. Bunu Şekil 4.5’de gösterildiği gibi şeffaf 3 adet cam ve bir perde ile ifade etmek mümkündür. Camlardan oluşan her bir katman, Öğretim Elemanı, Öğrenci ve Derslik katmanı olarak adlandırılır. Her katman üzerinde bulunan ve daha önceden yerleştirilmiş derslerin izdüşümü, perde üzerine düşürülerek yeni konulacak dersin perde üzerindeki çakışma durumu görülebilir.



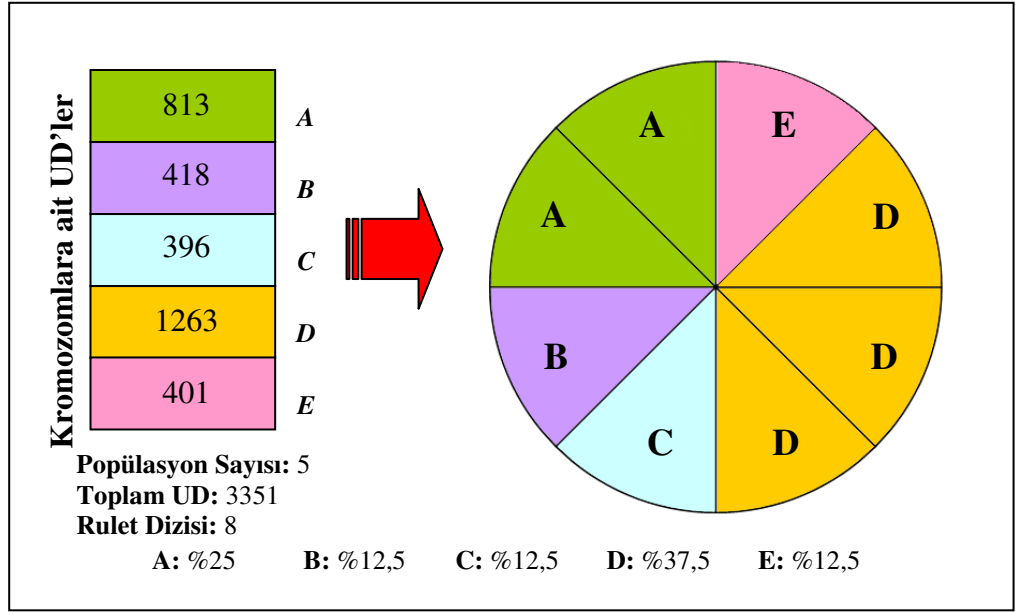
Şekil 4.5. Çakışmaların tespit edilmesi.

4.2.2. Kullanılan Seçim Yöntemleri

Genetik algoritma içerisinde kullanılan seçim yöntemlerinin detayları aşağıda maddeler halinde sunulmuştur. Bu seçim yöntemleri kullanıcı tarafından opsiyonel olarak seçilebilmektedir.

4.2.2.1. Rulet Tekeri Yöntemi

Bu seçim yönteminde, popülasyondaki her bir kromozoma ait uygunluk değerinin, popülasyondaki toplam uygunluk değerine oranı, 100 üzerinden hesaplanmış ve buna göre bir diziye kopyalanmıştır. Örneğin n. kromozomun uygunluk değerinin, popülasyondaki kromozomların toplam uygunluk değerine oranı %7 ise, bu kromozomdan diziye 7 kez kopyalama yapılmıştır. Daha sonra popülasyon sayısı kadar bir döngü kurularak, bu diziden rasgele kromozom seçilmiştir. Kromozomlardan uygunluk değeri büyük olan Rulet dizisinde daha çok bulunacağından seçilme şansında bir o kadar yüksek olmaktadır.



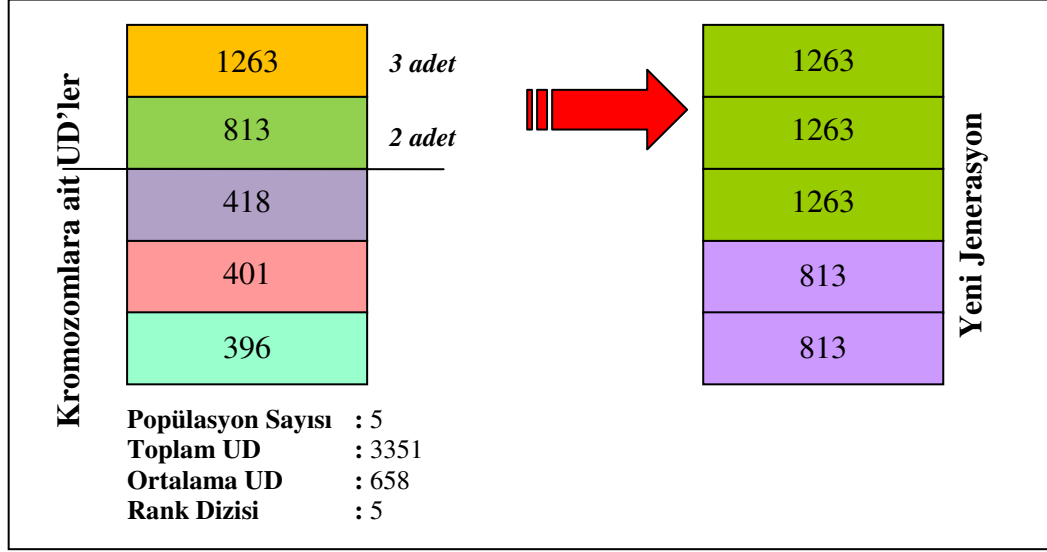
Şekil 4.6. Kullanılan rulet tekeri seçim yönteminin prensibi.

Şekil 4.6’da görüldüğü gibi, uygunluk değeri büyük olanın seçilme olasılığı da büyük olmaktadır. Burada D kromozomu %37,5 oranla en yüksek seçilme şansına sahiptir. Bunu %25’lik oranla A kromozomu takip etmektedir. B, C ve E kromozomları ise %12,5’lik oranla en düşük seçilme olasılığına sahip kromozomlardır.

4.2.2.2. Rank (Sıralı) Yöntemi

Sıralı seçim yönteminde, popülasyondaki kromozomlar uygunluk değerlerine göre büyükten küçüğe doğru sıralanmıştır. Popülasyondaki kromozomların uygunluk değerleri toplamı, popülasyon sayısına bölünerek ortalama bir uygunluk değeri bulunmuştur. Ortalamanın üzerinde uygunluk değerine sahip olan kromozomlar rasgele seçilerek yeni jenerasyona eklenmiştir. Yeni jenerasyondaki kromozom sayısı eski jenerasyondaki kromozom sayısına eşit oluncaya kadar işlem devam ettirilmiştir.

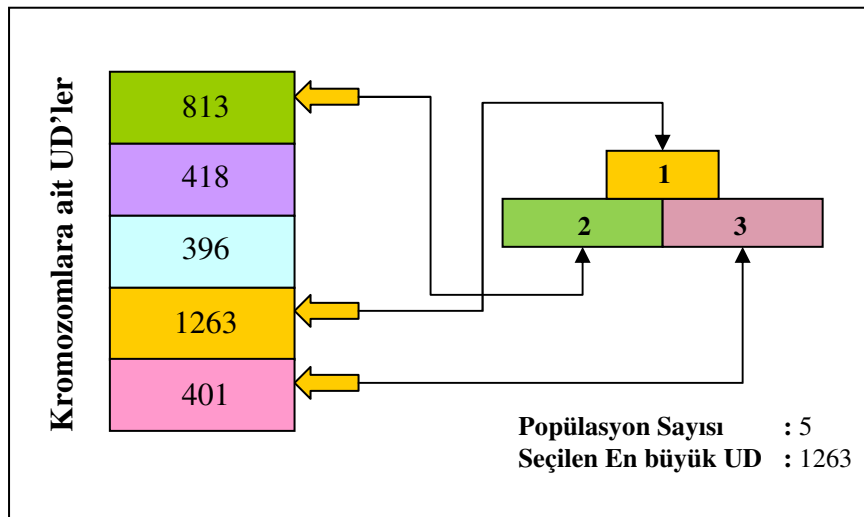
Şekil 4.7’de görüldüğü üzere sıralı seçim işleminden sonra yeni jenerasyona 2 farklı kromozom seçilmiştir. Uygunluk değerleri ortalamanın altında olan kromozomlar sıralamaya giremediğinden dolayı yeni jenerasyonda bulunamamıştır.



Şekil 4.7. Kullanılan sıralı seçim yönteminin prensibi.

4.2.2.3. Turnuva Yöntemi

Bu yöntemde öncelikle turnuvaya katılacak olan kromozom grubu seçilir. Bu gruptaki eleman sayısı 2-5 arasında ve rasgele seçilir. Oluşturulan bu grup içerisinde en iyi uygunluk değerine sahip olan kromozom turnuvayı kazanarak yeni Jenerasyonda yerini alır. Bu işleme popülasyon sayısına eşitleninceye kadar devam edilir.



Şekil 4.8. Kullanılan turnuva seçim yönteminin prensibi.

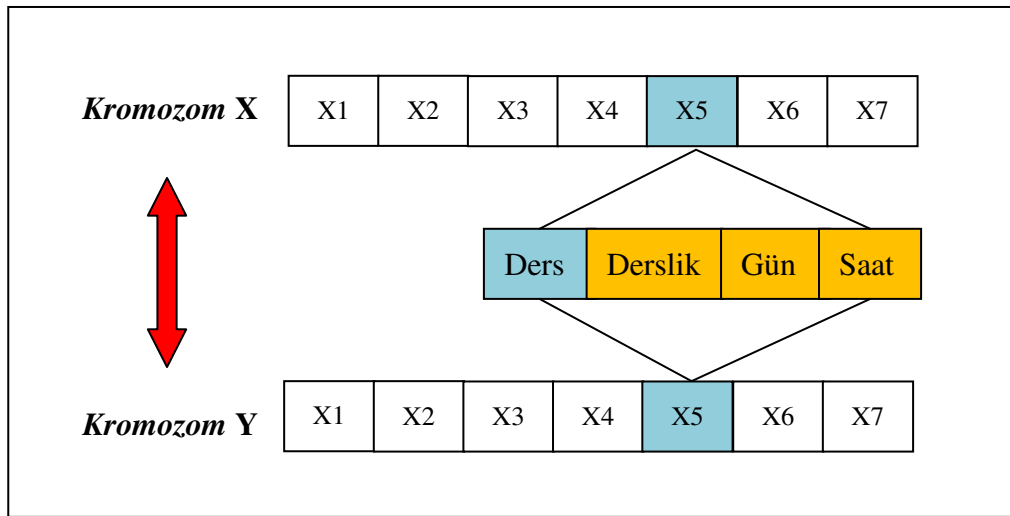
Şekil 4.8’de gösterildiği gibi, 2 ile 5 arasında rasgele seçilen sayı 3’tür. Buna göre popülasyondan rasgele 3 tane kromozom seçilmiş ve bunlardan UD’si en büyük olan kromozom yeni jenerasyona aktarılmıştır.

4.2.3. Kullanılan Çaprazlama Yöntemleri

4.2.3.1. Tek Noktalı Çaprazlama

Popülasyondan çaprazlamaya girecek olan kromozomlarına sayısı Çaprazlama oranı değişkeni ile belirlenir. Bu değer genelde %85 olarak alınmıştır. Çaprazlamaya girecek olan bireyler seçilirken her seferinde 2 ebeveyn seçileceğinden döngü tekrar sayısı bu oranın yarısı kadar alınmıştır. Rasgele seçilen her iki birey (kromozom) çaprazlama işlemine tabi tutulur. Tek noktalı çaprazlama işlemi yapılırken yine kromozom içerisindeki genler arasından rasgele bir konum seçilir. Bulunan bu gen diğer kromozomdaki eşiti ile yer değiştirilir.

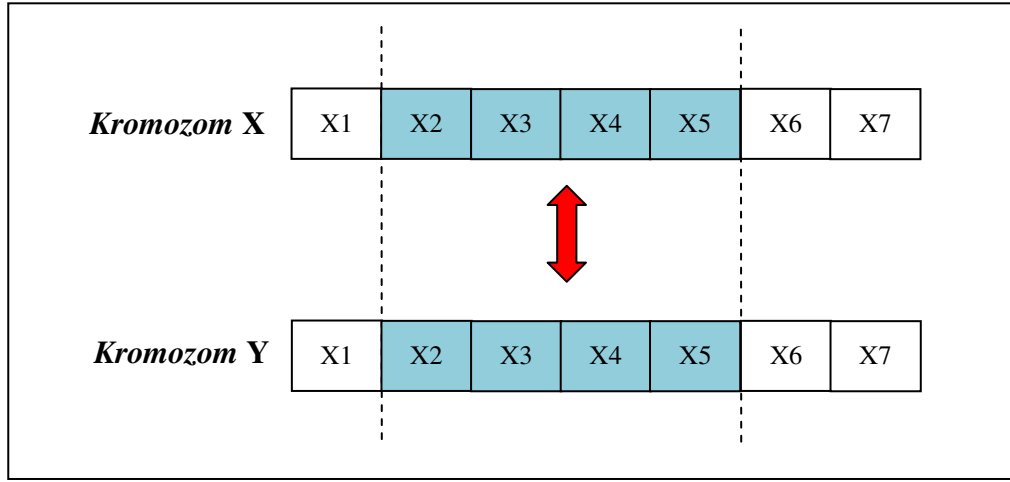
Kromozomlardaki genlerin (dersler) yerleri her kromozomda aynı yerde bulunmaktadır. Bu nedenle yer değiştiren genlerin içindeki dört adet bilgidен (Ders, Derslik, Gün, Saat) derse ait bilgiler aynı olacağından, diğer üç adet bilgi çaprazlanmış olmaktadır.



Şekil 4.9. Kullanılan tek noktalı çaprazlama yönteminin prensibi.

4.2.3.2. İki (Çift) Noktalı Çaprazlama

Aynı şekilde popülasyondan yine çaprazlama oranı kadar kromozom rasgele seçilir. Seçilen her kromozom arasında ise çift noktalı çaprazlama uygulanır. Kromozom üzerinde çaprazlama yapılacak noktalar ise rasgele belirlenir. Bu noktaların kromozomun ilk ve son genleri olmamasına dikkat edilir. Aksi takdirde bir kromozom olduğu gibi diğerinin yerine geçecektir.

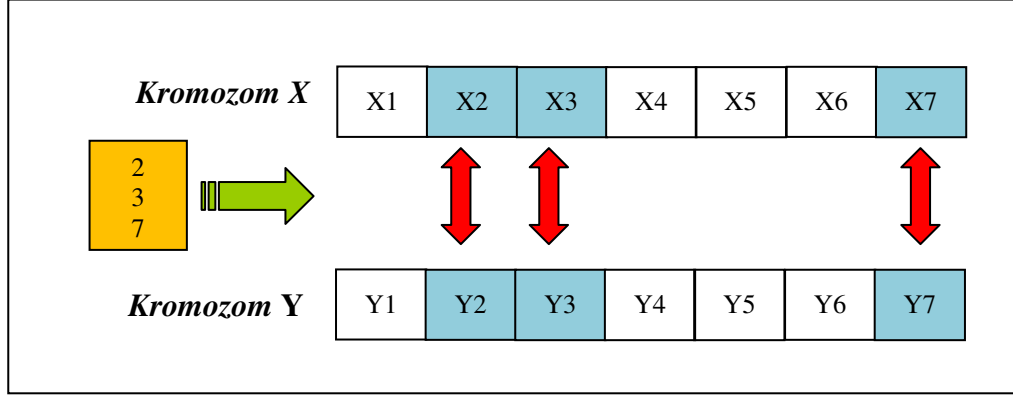


Şekil 4.10. Kullanılan iki noktalı çaprazlama yönteminin prensibi.

Şekil 4.10'da görüldüğü gibi, popülasyondan rasgele seçilen iki kromozoma çift noktalı çaprazlama uygulanırken çaprazlama noktaları rasgele 2 ve 5 olarak belirlenmiştir. X ve Y kromozomlarının bu noktalar arasındaki genleri olduğu gibi çaprazlanmış olmaktadır.

4.2.3.3. Periyodik Çaprazlama

Popülasyondan rasgele seçilen iki kromozom için Periyodik çaprazlama yapılırken her iki kromozomda karşılıklı olarak çaprazlanacak genler (dersler) rastgele belirlenen bir sayı kümesine göre yapılır. Belirlenen sayı kümesinin işaret ettiği genler çaprazlanarak işlem tamamlanır.



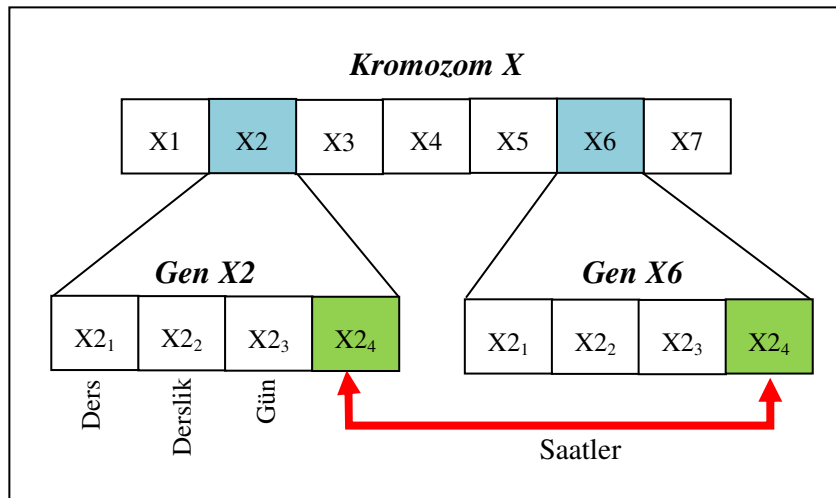
Şekil 4.11. Kullanılan periyodik çaprazlama yönteminin prensibi.

Şekil 4.11’de görüldüğü üzere, rasgele seçilen sayı kümesi 2., 3. ve 7. genlere işaret etmektedir. X ve Y kromozomları içinde bu sayı kümesinin işaret ettiği genler birbiriyle yer değiştirilerek çaprazlama işlemi gerçekleştirilmiş olur.

4.2.4. Kullanılan Mutasyon Yöntemleri

4.2.4.1. Karşılıklı Değişim Yöntemi

Popülasyondan rasgele seçilen bir kromozom içindeki genlerden iki tanesi yine rasgele seçilir. Seçilen bu genler mutasyona uğrayacak genlerdir. Mutasyon işlemi ise bu genlerin içerisindeki 4 adet bilgidен sadece saat alanı içerisinde uygulanmıştır.

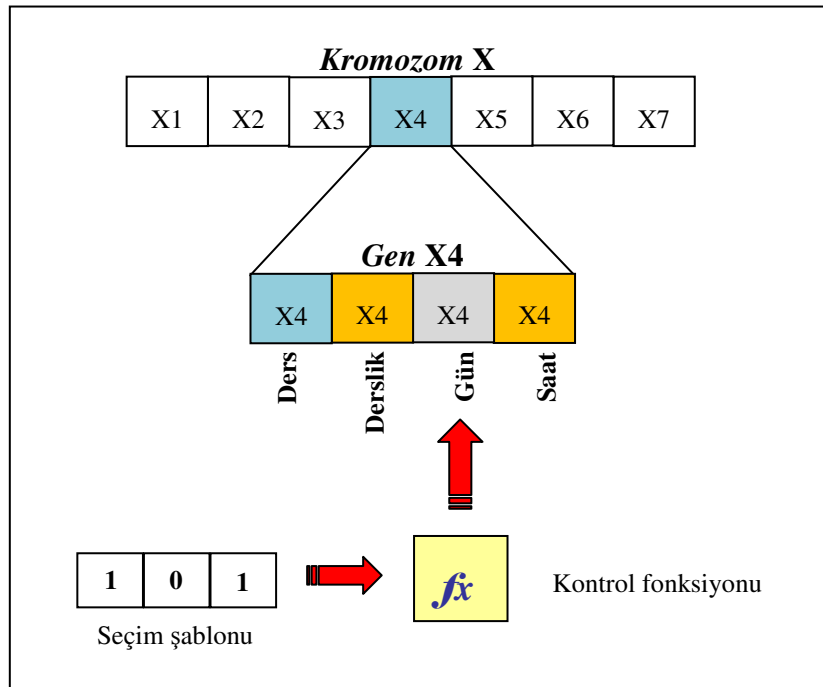


Şekil 4.12. Kullanılan karşılıklı değişim mutasyonunun prensibi.

Şekil 4.12’de görüldüğü gibi, X kromozomundan rasgele seçilen X2 ve X6 genleri (her bir gen Ders, Derslik, Gün, Saat bilgilerini tutmaktadır) arasında X2₄ ve X6₄ (Saat bilgileri) yer değiştirmiştir.

4.2.4.2. Özel Yöntem

Popülasyondan rasgele seçilen bir kromozom içinde bulunan genlerden rasgele bir tanesi seçilir. Seçilen bu gen üzerinde mutasyona uğrayabilecek Derslik, Gün ve Saat olmak üzere üç adet bilgi bulunmaktadır. Bu bilgilerden hangilerine mutasyon işlemi uygulanacağı ise yine rasgele oluşturulan bir şablona göre yapılmaktadır. Bu şablonda 1 değerleri mutasyon uygulanacağını ve 0 alan değerler ise mutasyon uygulanmayacağını göstermektedir. Ancak, mutasyon işlemi Derslik ve Saatlere uygulanırken belli bir ön fonksiyondan geçirilerek yapılmalıdır. Örneğin bir ders için rasgele bir Derslik seçilemez. O dersin verilebileceği derslikler bellidir. Mutasyon işlemi bu derslikler arasında yapılmalıdır. Dersin yerleştirileceği ders saatleri rasgele olsa bile bazı kısıtlamaların uygulanması gerekir. Örneğin günlük 10 periyotluk bir çizelgede, üç saatlik bir ders 9. saatten itibaren konulamaz. Bunun gibi öğle arasından önce ve sonraya dersler bölünemez ise bunun da kontrol edilmesi gerekir.



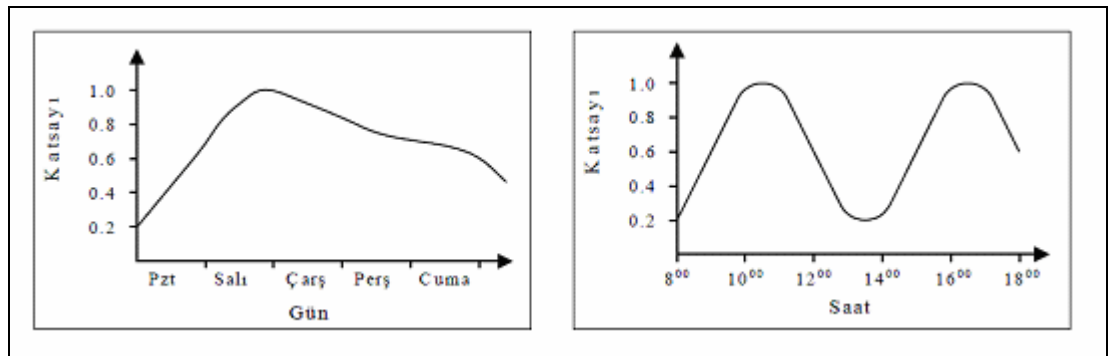
Şekil 4.13. Kullanılan özel yöntem mutasyonunun prensibi.

4.3. PROBLEMİN ÇÖZÜMÜNDE KULLANILAN KISITLAR

Haftalık ders çizelgeleme kısıtları, üniversitelerin ihtiyaçlarına göre düzenlenmesi gerekir. Kısıtlar, eğitim/öğretim kalitesinin iyileştirilmesi, öğrencilerin ve öğretim elemanlarının performansını arttırabilmesi, öğretim elemanı isteklerinin karşılanması ve idari ihtiyaçlara cevap verebilmesi esas alınarak oluşturulmuştur [23].

4.3.1. Eğitim/Öğretim Verimliliği

Pedagojik esaslara göre, öğrencilerin anlama yeteneği haftanın günlerine göre değişmektedir. Buna göre, hafta içindeki günlerin eğitim-öğretim verimlilik dereceleri farklıdır. Haftanın ilk ve son günleri genel olarak verimin az olduğu günlerdir [52]. Günlerin uygunluk değerinin hesaplanması için, Şekil 4.14'de gösterilen günlere göre verim eğrisi dikkate alınarak, günlere ağırlık değerleri verilmiştir. Yine pedagojik araştırmalara göre, gün içindeki çalışma saatlerinin de verimlilik dereceleri farklıdır. Günün çalışma saatlerinin ilk ve son dilimlerinde verim düşüktür. Saatlerin uygunluk değerinin hesaplanması için, Şekil 4.14'de gösterilen günün çalışma saatlerine göre verim eğrisi dikkate alınarak, çalışma saatlerine uygun ağırlık değerleri verilmiştir. Grafikte, gün içinde çalışma saati sayısı 9 dilim olarak alınmıştır [29].



a) Pedagojik gün verimliliği.

b) Pedagojik saat verimliliği.

Şekil 4.14. Pedagojik gün ve saat verimliliği [52].

4.3.2. Öğretim Elemanı Memnuniyeti

Öğretim elemanlarının derslerini işlemek istedikleri gün ve saatleri belirleyebilmesi, akademik çalışma, danışmanlık, idari görev vb. durumlar için uygun zamanların oluşturulmasına yardımcı olmaktadır.

Öğretim elemanı memnuniyeti için Şekil 4.15’de görüldüğü gibi bir kısıt çizelgesi oluşturulmuştur. Akademisyen tarafından verilen ders saati sayısı kadar kısıt çizelgesi üzerinden seçim yapılması istenir. Belirlenen gün ve saatlere göre kısıt puanı hesaplanır ve kromozomun uygunluk değeri güncellenir.

ÖĞRETİM ELEMANI MEMNUNİYET ÇİZELGESİ						
Yrd.Doç.Dr. Baha ŞEN						
Haftalık 23 saatlik dersinizden 0 seçim hakkınız kaldı.						
Güvenli Çıkış						
GÜN / SAAT	PAZARTESİ	SALI	ÇARŞAMBA	PERŞEMBE	CUMA	
ÖRGÜN ÖĞRETİM MEMNUNİYET ÇİZELGESİ						
1	08:00 - 08:45	İstenen Alan	İstenen Alan			
2	08:50 - 09:35	İstenen Alan	İstenen Alan			
3	09:40 - 10:25	İstenen Alan	İstenen Alan			
4	10:30 - 11:15	İstenen Alan				İstenen Alan
5	11:20 - 12:05	İstenen Alan				İstenen Alan
6	13:00 - 13:45					
7	13:50 - 14:35					
8	14:40 - 15:25					
9	15:30 - 16:15					
10	16:20 - 17:05					
İKİNCİ ÖĞRETİM MEMNUNİYET ÇİZELGESİ						
11	17:10 - 17:55	İstenen Alan	İstenen Alan			
12	18:00 - 18:45	İstenen Alan	İstenen Alan			

Şekil 4.15. Öğretim elemanı memnuniyet çizelgesi.

Ayrıca, öğretim elemanları için üniversite idaresince izin verilen gün ve saatlerde Alan Kapatma kısıtı da kullanılmaktadır. Bu alanlara kesinlikle ders konulamamaktadır.

4.3.3. Ders Yüğü Ağırlığı

Ders yüğü ağırlığının çizelge üzerinde dağılımının nasıl yapılacağı üç farklı şekilde belirlenebilir:

1. Yoğun: Bazı akademik birimlerde, bölüm/program sayısının fiziki yapı şartlarına göre fazla olmasından ve/veya az sayıda derslik bulunmasından dolayı bu türde bir ders yerleştirme tercihi yapılabilir.
2. Dengeli: Öğrencilerin ve öğretim elemanlarının günlük ders saatlerinin sayısını mümkün olduğunca haftanın her gününe eşit sayıda dağıtmayı sağlamaktadır.
3. Serbest: Yukarıda bahsedilen iki durumun dışında hiçbir ders yükü ağırlığı kısıtlarının kontrolü yapılmadan da ders yerleştirme işlemi yapılabilmektedir. Kısaca bu türde bir seçim yapıldığında, ders yükü ağırlığı için bir kısıt puanı hesaplanmayacaktır.

4.3.4. Bölünen Derslerin Buluşma Zamanları

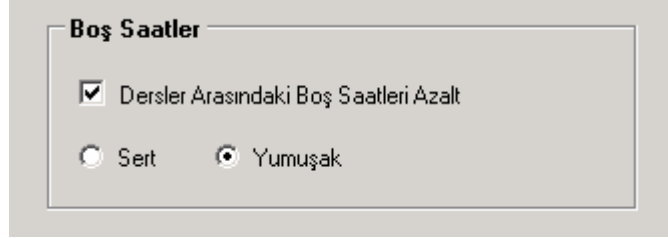
Bir dersin, çizelge üzerinde teori, uygulama ve laboratuvar olarak parçalanması ya da gruplara ayrılması istenebilir veya bu dersin her bir parçasına farklı öğretim elemanları atanabilir. Bu gibi durumlarda, bölünen dersler için buluşma zamanları farklı günlerde ya da aynı günde olması istenebilir.

Bölünen derslerin buluşma zamanları için çizelge üzerinde dağılımının nasıl yapılacağı üç farklı şekilde belirlenebilir:

1. Farklı Gün: Bölünen dersler için buluşma zamanları farklı günlere atanabilmesi için bu türde bir tercih yapılabilir.
2. Aynı Gün: Bölünen dersler için buluşma zamanlarını aynı günlere atanabilmesi için bu türde bir tercih yapılabilir.
3. Serbest: Yukarıda bahsedilen iki durumun dışında hiçbir (bölünen derslerin buluşma zamanları) kısıt kontrolü yapılmadan da ders yerleştirme işlemi yapılabilmektedir. Kısaca bu türde bir seçim yapıldığında, bölünen derslerin buluşma zamanları için bir kısıt puanı hesaplanmayacaktır. Dersler rasgele olarak yerleştirilecektir.

4.3.5. Ders Saatleri Arasındaki Boş Zamanlar

Ders saatleri arasındaki boşlukları sert ve yumuşak seçimler kullanılarak ayarlanabilir ya da bu kısıt devre dışı bırakılabilir (Şekil 4.16).



Şekil 4.16. Dersler arasındaki boş saatleri ayarlayan kısıtlar.

Genetik işlemler sırasında yukarıda anlatılan kısıtlara ait ağırlık değerleri Çizelge 4.1’de gösterilmektedir.

Çizelge 4.1. Kısıt ağırlıkları tablosu

No	Kısıt Adı	Kısıt Puanı
1	Çakışmalar	-1000
2	Öğretim Elemanı Memnuniyeti	500
3	Ders Yüğü Ağırlığı	-10
4	Pedagojik Verimlilik	Çizelge 4.2. ve Çizelge 4.3.
5	Dersler Arasındaki Boş Saatler	-100

4.4. GELİŞTİRİLEN YAPININ MATEMATİKSEL MODELİ

1. Çakışma kısıt puanı -1000 olarak belirlenmiştir. Aşağıda Öğretim Elemanı, Derslik ve Akademik Programlara ait çakışma kısıtları toplamını ifade eden fonksiyon tanımı yapılmıştır.

C_{AC} : Öğretim Elemanı Çakışma Kısıtı

C_{CC} : Derslik Çakışma Kısıtı

C_{PC} : Akademik Program Çakışma Kısıtı

m : Öğretim Elemanı Sayısı

n : Derslik Sayısı
r : Akademik Program Sayısı
d : Günler h : Saatler

$$f_{C_{AC}} = \sum_{x=1}^m C_{AC[x][d, h]} \quad \forall d \in (1,2,3,4,5) \quad \forall h \in (1,2,3,\dots,10) \quad (4.1a)$$

$$f_{C_{CC}} = \sum_{x=1}^n C_{CC[x][d, h]} \quad \forall d \in (1,2,3,4,5) \quad \forall h \in (1,2,3,\dots,10) \quad (4.1b)$$

$$f_{C_{PC}} = \sum_{x=1}^r C_{PC[x][d, h]} \quad \forall d \in (1,2,3,4,5) \quad \forall h \in (1,2,3,\dots,10) \quad (4.1c)$$

2. Öğretim Elemanı memnuniyeti (gün ve saat isteği) kısıt puanı 500 olarak belirlenmiştir. Aşağıda Öğretim Elemanı Memnuniyet Kısıtları toplamını ifade eden fonksiyon tanımı yapılmıştır.

C_A : Öğretim Elemanı Memnuniyet Kısıtı
n : Öğretim Elemanı Sayısı
d : Günler
h : Saatler

$$f_{C_A} = \sum_{x=1}^n C_{A[x][d, h]} \quad \forall d \in (1,2,3,4,5) \quad \forall h \in (1,2,3,\dots,10) \quad (4.2)$$

3. Ders Yüğü Ağırlığı kısıt puanı -10 olarak belirlenmiştir. Aşağıda Öğretim Elemanı, Derslik ve Akademik Programlara ait ders yüğü ağırlığı kısıtları toplamını ifade eden fonksiyon tanımı yapılmıştır.

C_{AL} : Öğretim Elemanı Ders Yüğü Ağırlığı Kısıtı
 C_{CL} : Derslik Ders Yüğü Ağırlığı Kısıtı
 C_{PL} : Akademik Program Ders Yüğü Ağırlığı Kısıtı
m : Öğretim Elemanı Sayısı
n : Derslik Sayısı
r : Akademik Program Sayısı

d : Günler

h : Saatler

$$f_{CAL} = \sum_{x=1}^m C_{AL[x][d, h]} \quad \forall d \in (1,2,3,4,5) \quad \forall h \in (1,2,3,\dots,10) \quad (4.3a)$$

$$f_{CCL} = \sum_{x=1}^n C_{CL[x][d, h]} \quad \forall d \in (1,2,3,4,5) \quad \forall h \in (1,2,3,\dots,10) \quad (4.3b)$$

$$f_{CPL} = \sum_{x=1}^r C_{PL[x][d, h]} \quad \forall d \in (1,2,3,4,5) \quad \forall h \in (1,2,3,\dots,10) \quad (4.3c)$$

4. Pedagojik Verimlilik kısıt puanları Çizelge 4.2 ve Çizelge 4.3’de verilmiştir. Aşağıda Öğretim Elemanı, Derslik ve Akademik Programlara ait çakışma kısıtları toplamını ifade eden fonksiyon tanımı yapılmıştır.

Çizelge 4.2. Pedagojik verimlilik (günler).

Günler	Pazartesi	Salı	Çarşamba	Perşembe	Cuma
Kısıt Puanı	4	9	8	7	6

Çizelge 4.3. Pedagojik verimlilik (saatler).

Saatler	1	2	3	4	5	6	7	8	9	10
Kısıt Puanı	3	4	10	8	6	2	4	5	8	10

C_{PD} : Pedagojik Verimlilik (Gün) Kısıtı

C_{PH} : Pedagojik Verimlilik (Saat) Kısıtı

m : Akademik Program Sayısı

d : Günler h : Saatler

$$f_{CPD} = \sum_{x=1}^m C_{PD[x][d, h]} \quad \forall d \in (1,2,3,4,5) \quad \forall h \in (1,2,3,\dots,10) \quad (4.4a)$$

$$f_{CPH} = \sum_{x=1}^m C_{PH[x][d, h]} \quad \forall d \in (1,2,3,4,5) \quad \forall h \in (1,2,3,\dots,10) \quad (4.4b)$$

Uyguluk deęerinin hesaplanmasında, her bir kromozom için tüm kısıt puanlarının toplam deęerleri alınarak bulunur.

$$f_{FV} = f_{CAC} + f_{CC} + f_{PC} + f_{CA} + f_{CAL} + f_{CL} + f_{PL} + f_{PD} + f_{PH} \quad (4.5)$$

Amaç Fonksiyonu;

Amaç fonksiyonu, popülasyondaki her bir kromozomun toplam uygunluk deęerleri içerisinde en büyük deęere sahip olan muhtemel çözümdür.

n : Popülasyondaki kromozom sayısı

$$f = \text{Max} \sum_{x=1}^n (f_{FV[x]}) \quad (4.6)$$

Bir kromozomdaki başarı oranları aşığıdaki gibi elde edilir.

C_C : Çakışmalar

C_P : Memnuniyetler

n : Kromozomdaki Gen Sayısı

m : Talep Edilen Toplam Memnuniyet Saati Sayısı

d : Günler

h : Saatler

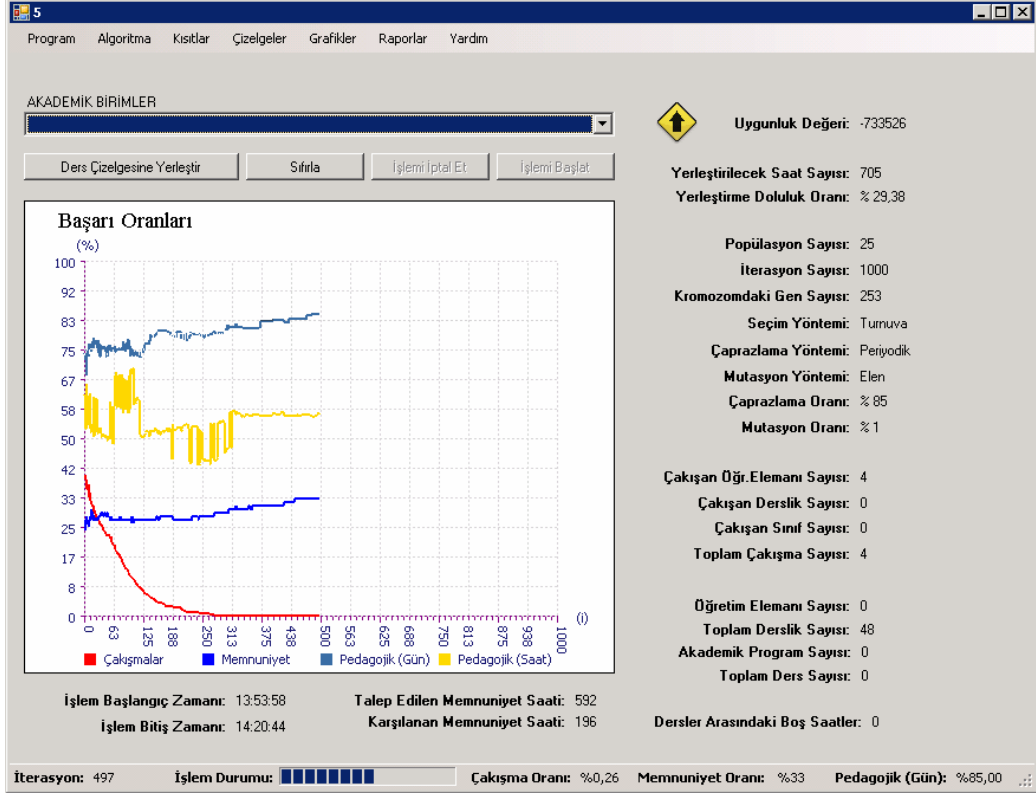
$$f_{C_C} = \frac{\sum_{x=1}^n Cc[x][d,h]}{n} \quad \forall d \in (1,2,3,4,5) \quad \forall h \in (1,2,3,\dots,10) \quad (4.7a)$$

$$f_{C_P} = \frac{\sum_{x=1}^n Cp[x][d,h]}{m} \quad \forall d \in (1,2,3,4,5) \quad \forall h \in (1,2,3,\dots,10) \quad (4.7b)$$

4.5. GELİŞTİRİLEN UYGULAMA ARAYÜZÜ

Microsoft Visual Studio .NET platformunda C# programlama dili kullanılarak geliştirilmiştir. Geliştirilen uygulama arayüzü ana ekranı şekil 4.17’de gösterilmiştir.

Ana ekran üzerinde, önceden belirlenmiş parametre değerleri, işlem sırasında gerçekleşen değişim değerleri ve önyükleme değerlerini gösteren etiketler bulunmaktadır. Sistemin çalışması sırasında, hesaplanan başarı oranlarının gerçek zamanlı olarak grafikleri de çizdirilmektedir.



Şekil 4.17. Geliştirilen uygulama arayüzü ana ekranı.

Sistemin çalıştırabilmesi için, öncelikle “Akademik Birimler” çoktan seçmeli kutusundan ilgili birim (Fakülte, Enstitü, Yüksekokul ve Meslek Yüksekokulu) seçilir. Daha sonra seçilen birime göre, Algoritma menüsünden GA'nın parametre değerleri ayarlanır. Bir sonraki işlem adımı ise kısıtlardır. Kısıtlar menüsü altından “Yapılandır” sekmesi tıklanır. Birimlerin istekleri göz önünde bulundurularak gerekli ayarlar yapılır. Böyle sistem çalışmaya hazır hale getirilmiş olacaktır. Son olarak İşlemi Başlat butonuna tıklanarak sistem çalışır hale getirilir.

GA'nın operatörleri ve parametre değerlerinin belirlenmesinde Şekil 4.18'deki ekran kullanılmaktadır. Seçim Yöntemi olarak en çok kullanılan Rulet, Turnuva ve Rank olmak üzere üç farklı tercih yapılabilmektedir. Çaprazlama Yönteminde Tek Noktalı,

Çift (iki) Noktalı ve Periyodik olmak üzere yine üç farklı tercih yapılabilir. Mutasyon Yönteminde ise Karşılıklı Değişim ve Özel Yöntem olarak iki farklı şekilde uygulanabilmektedir.

Algoritmalar

Genetik Algoritma

Seçim Yöntemi

Rulet

Turnuva

Rank

Çaprazlama Yöntemi

Tek Noktalı

Çift Noktalı

Periyodik

Mutasyon Yöntemi

Karşılıklı Değişim

Özel Yöntem

Populasyon Sayısı : 25

İterasyon Sayısı : 1000

Çaprazlama Oranı (%) : 85

Mutasyon Oranı : 2

Elitizm Sayısı : 1

Durdurma Kriteri : Başarı Oranı İterasyon Sayısı

Otomatik Ayarla

Vazgeç Kaydet

Şekil 4.18. GA parametrelerinin seçimi.

Parametre değerlerinde Mutasyon oranı sistem tarafından otomatik olarak kontrol edilebilmektedir. Bunun sebebi ise belli bir iterasyon süresi boyunca en iyi kromozom değerinin sabit kalmasından dolayıdır. Son olarak üç farklı durdurma kriteri bulunmaktadır. Bunlar İterasyon Sayısı, Başarı Oranı ve çalışma zamanı operatör isteği olarak adlandırılmıştır.

Şekil 4.19’da görülen kısıt tanımlama ekranında, öğretim elemanı, derslik ve akademik program için ayrı ayrı ders yükü ağırlığı tanımı yapılabilir. Daha önce de bahsedildiği gibi bunlar Yoğun, Dengeli ve Serbest program olmak üzere üç farklı şekilde uygulanabilmektedir.

Pedagojik Gün ve Saat verimliliğinin hesaba katılıp katılmayacağı tanımını yapmak için Pedagojik Verimlilik grubundan seçim yapılır.

Dersler arasındaki boş saatlerin azaltılmasında sert ve yumuşak olmak üzere iki farklı türde ceza uygulanabilmektedir. Sert ceza uygulandığında dersler arası boşluk en aza indirilmektedir.

Kısıtlar

Ders Yüğü Ağırlığı

Öğretim Elemanı

Yoğun Program

Dengeli Program

Serbest Program

Derslik

Yoğun Program

Dengeli Program

Serbest Program

Akademik Program

Yoğun Program

Dengeli Program

Serbest Program

Pedagojik Verimlilik

Pedagojik Verimlilik Hesaplınsın

Boş Saatler

Dersler Arasındaki Boş Saatleri Azalt

Sert Yumuşak

Öğle Arası Ders Bölünmesi

Öğle Paydosu Ders Bölünmesini Onaylıyorum

Tamam

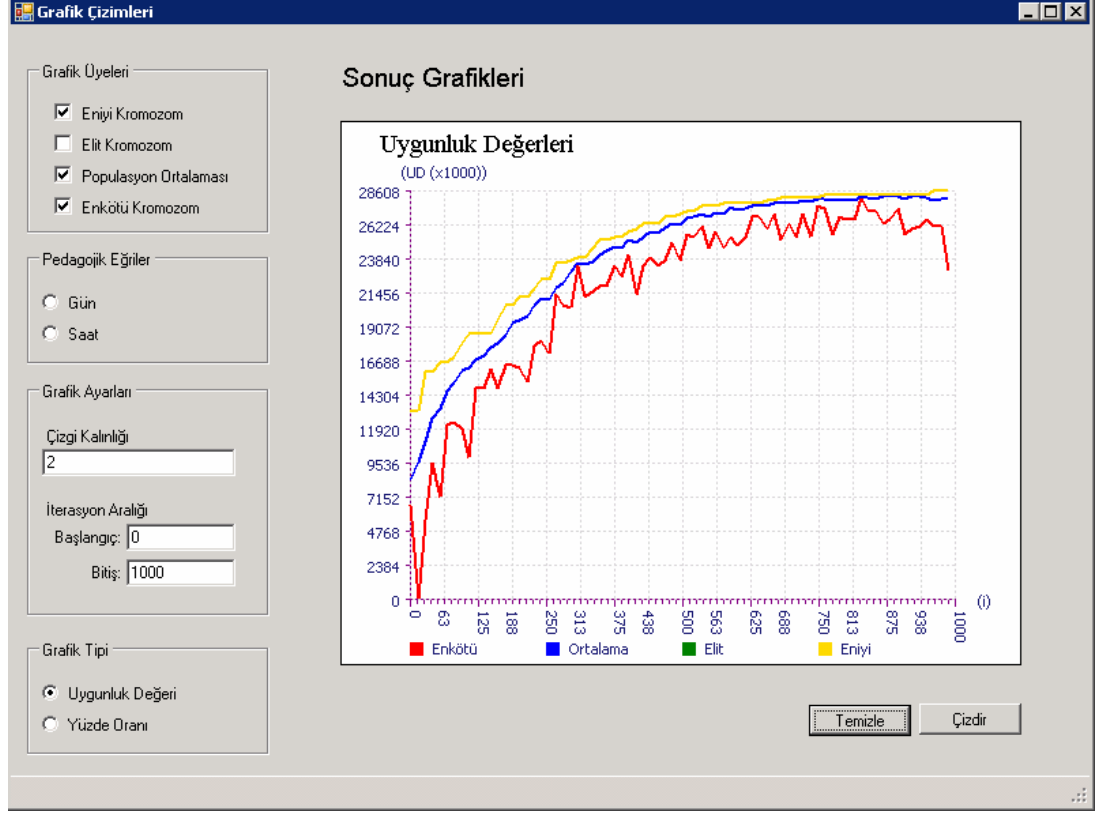
Şekil 4.19. Kısıt tanımlama ekranı.

Ders çizelgelerinin oluşturulması sırasında, derslerin öğle arası bölünmelerine izin vermek için Öğle Arası Ders Bölünmesi grubundan işaretleme yapılması gerekmektedir.

Kısıt tanımlarıyla ilgili işlemler tamamlandıktan sonra “Tamam” butonuna tıklanır.

Çizelgeleme işleminin tamamlanmasının ardından sonuç grafikleri Şekil 4.20’de görüldüğü gibi genetik değerler ve pedagojik gün ve saat değerleri ayrı ayrı çizdirilebilmektedir. Grafik ayarları grubundan çizgi kalınlığı ve görüntülenmesi

istenilen iterasyon aralığı belirlenebilmektedir. Grafik tipi grubundan ise her iterasyonda gösterilecek olan değerin tipi belirlenmektedir.



Şekil 4.20. Sonuç grafikleri.

Ders çizelgeleme işlemi tamamlandıktan sonra ana ekran üzerinden “Ders Çizelgesine Yerleştir” butonuna tıklayarak, Genetik Algoritma ile elde edilen en iyi sonuç veritabanına kaydedilir. Bu işlem sonucunda web tabanlı bir ortamda, Öğretim Elemanı, Derslik ve Akademik Program olmak üzere üç farklı türde çizelge oluşturulmaktadır.

Şekil 4.21’de bir öğretim elemanı için hazırlanmış ders çizelgesi görülmektedir. Her bir zaman diliminde, ders kodu, derslik kodu ve akademik program kodu bildirilmektedir.

Yine benzer şekilde akademik program ve derslikler içinde Şekil 4.22 ve Şekil 4.23’de görüldüğü gibidir.

Yrd.Doç.Dr. İbrahim ÇAYIROĞLU					
GÜN / SAAT	PAZARTESİ	SALI	ÇARŞAMBA	PERŞEMBE	CUMA
1	08:00 - 08:45				TLU401 (T) - MOB-2 [3] Yrd.Doç.Dr. İ. Çayiroğlu 0420
2	08:00 - 08:45				TLU401 (T) - MOB-2 [3] Yrd.Doç.Dr. İ. Çayiroğlu 0420
3	08:00 - 08:45	BLG211 (T) - ANFİ 2 [3] Yrd.Doç.Dr. İ. Çayiroğlu 0425			TLU401 (T) - MOB-2 [3] Yrd.Doç.Dr. İ. Çayiroğlu 0420
4	08:00 - 08:45	BLG211 (T) - ANFİ 2 [3] Yrd.Doç.Dr. İ. Çayiroğlu 0425			
5	08:00 - 08:45	BLG211 (T) - ANFİ 2 [3] Yrd.Doç.Dr. İ. Çayiroğlu 0425			
6	08:00 - 08:45		BLG403 (T) - ANFİ 10 [3] Yrd.Doç.Dr. İ. Çayiroğlu 0425		
7	08:00 - 08:45		BLG403 (T) - ANFİ 10 [3] Yrd.Doç.Dr. İ. Çayiroğlu 0425		BLG319 (T) - ANFİ 8 [3] Yrd.Doç.Dr. İ. Çayiroğlu 0425
8	08:00 - 08:45		BLG403 (T) - ANFİ 10 [3] Yrd.Doç.Dr. İ. Çayiroğlu 0425		BLG319 (T) - ANFİ 8 [3] Yrd.Doç.Dr. İ. Çayiroğlu 0425
9	08:00 - 08:45				BLG319 (T) - ANFİ 8 [3] Yrd.Doç.Dr. İ. Çayiroğlu 0425
10	08:00 - 08:45				

Şekil 4.21. Öğretim elemanı haftalık ders çizelgesi.

BİLGİSAYAR SİSTEMLERİ ÖĞRETMENLİĞİ - 4. Sınıf					
GÜN / SAAT	PAZARTESİ	SALI	ÇARŞAMBA	PERŞEMBE	CUMA
1	08:00 - 08:45				
2	08:00 - 08:45		BLG407 (T) - ANFİ 12 Doç.Dr. R. Bayır (2)	BLG405 (T) - ANFİ 11 Yrd.Doç.Dr. B. Şen (3)	EGT401 (T) - ANFİ 12 Prof.Dr. A. Özçifci (1)
3	08:00 - 08:45		BLG407 (T) - ANFİ 12 Doç.Dr. R. Bayır (2)	BLG405 (T) - ANFİ 11 Yrd.Doç.Dr. B. Şen (3)	
4	08:00 - 08:45	EGT403 (T) - OTO. 3 Yrd.Doç.Dr. A. Özyürek (2)		BLG405 (T) - ANFİ 11 Yrd.Doç.Dr. B. Şen (3)	
5	08:00 - 08:45	EGT403 (T) - OTO. 3 Yrd.Doç.Dr. A. Özyürek (2)			
6	08:00 - 08:45	BLG411 (T) - 101 Öğr.Gör. O. Erkamaz (3)	BLG403 (T) - ANFİ 10 Yrd.Doç.Dr. İ. Çayiroğlu (3)		
7	08:00 - 08:45	BLG411 (T) - 101 Öğr.Gör. O. Erkamaz (3)	BLG403 (T) - ANFİ 10 Yrd.Doç.Dr. İ. Çayiroğlu (3)	BLG401 (T) - ANFİ 9 Yrd.Doç.Dr. S. Görünözü (3)	
8	08:00 - 08:45	BLG411 (T) - 101 Öğr.Gör. O. Erkamaz (3)	EGT405 (T) - TES. 2 Prof.Dr. E. Candan (2)	BLG403 (T) - ANFİ 10 Yrd.Doç.Dr. İ. Çayiroğlu (3)	BLG401 (T) - ANFİ 9 Yrd.Doç.Dr. S. Görünözü (3)
9	08:00 - 08:45		EGT405 (T) - TES. 2 Prof.Dr. E. Candan (2)		BLG401 (T) - ANFİ 9 Yrd.Doç.Dr. S. Görünözü (3)
10	08:00 - 08:45				

Şekil 4.22. Öğrenci haftalık ders çizelgesi.

203						
GÜN / SAAT	PAZARTESİ	SALI	ÇARŞAMBA	PERŞEMBE	CUMA	
1	08:00 - 08:45					
2	08:00 - 08:45	MKM201 (T) [4] - 0210 Doç.Dr. O. Yiğit		BLM181 (T) [2] - 0201 Yrd.Doç.Dr. M. Özalo	MKM105 (T) [2] - 0201 Yrd.Doç.Dr. G. Sur	BLM181 (T) [2] - 0210 Yrd.Doç.Dr. M. Özalo
3	08:00 - 08:45	MKM201 (T) [4] - 0210 Doç.Dr. O. Yiğit	MAT181 (T) [3] - 0210 Yrd.Doç.Dr. M. Düz	BLM181 (T) [2] - 0201 Yrd.Doç.Dr. M. Özalo	MKM105 (T) [2] - 0201 Yrd.Doç.Dr. G. Sur	BLM181 (T) [2] - 0210 Yrd.Doç.Dr. M. Özalo
4	08:00 - 08:45	MKM201 (T) [4] - 0210 Doç.Dr. O. Yiğit	MAT181 (T) [3] - 0210 Yrd.Doç.Dr. M. Düz	TUR181 (T) [2] - 0201 Yrd.Doç.Dr. A. Öksüz	KIM181 (T) [2] - 0208 Yrd.Doç.Dr. M. Levent	
5	08:00 - 08:45	MKM201 (T) [4] - 0210 Doç.Dr. O. Yiğit	MAT181 (T) [3] - 0210 Yrd.Doç.Dr. M. Düz	TUR181 (T) [2] - 0201 Yrd.Doç.Dr. A. Öksüz	KIM181 (T) [2] - 0208 Yrd.Doç.Dr. M. Levent	
6	08:00 - 08:45					
7	08:00 - 08:45	GSM181 (T) [2] - 0201 Öğr.Gör. G. Hacısüleymanoğlu	FIZ181 (T) [3] - 0205 Prof.Dr. D. Paktaş	FIZ181 (T) [3] - 0208 Prof.Dr. D. Paktaş	FIZ181 (T) [3] - 0210 Prof.Dr. D. Paktaş	
8	08:00 - 08:45	GSM181 (T) [2] - 0201 Öğr.Gör. G. Hacısüleymanoğlu	FIZ181 (T) [3] - 0205 Prof.Dr. D. Paktaş	FIZ181 (T) [3] - 0208 Prof.Dr. D. Paktaş	FIZ181 (T) [3] - 0210 Prof.Dr. D. Paktaş	GSR181 (T) [2] - 0208 Yrd.Doç.Dr. A. Ertok atmaca
9	08:00 - 08:45		FIZ181 (T) [3] - 0205 Prof.Dr. D. Paktaş	FIZ181 (T) [3] - 0208 Prof.Dr. D. Paktaş	FIZ181 (T) [3] - 0210 Prof.Dr. D. Paktaş	GSR181 (T) [2] - 0208 Yrd.Doç.Dr. A. Ertok atmaca
10	08:00 - 08:45					

Şekil 4.23. Derslik haftalık ders çizelgesi.

BÖLÜM 5

DENEYSEL ÇALIŞMALAR

Hazırlanan çizelgeleme programı, Microsoft Windows Server 2003 kurulu 1024 MB hafızası olan Core2 Quad 2.40 GHz işlemcili bir bilgisayar üzerinde deneyleri yapılmıştır. Deneyler, en uygun genetik algoritma operatörlerinin ve parametre değerlerinin seçilmesi için yapılmıştır. Deneylerde kullanılan veriler gerçek veriler olup Üniversitenin tamamı için 504 öğretim elemanı, 4163 ders, 203 derslik ve 10525 öğrenciden oluşmaktadır. Böylelikle elde edilen sonuçlar operatörlerin ve parametrelerin değerlerinin gerçek performanslarını göstermiştir. Genetik algoritma kullanılarak çözümü bulunan ders programı çizelgelemesi Karabük Üniversitesi, Teknik Eğitim Fakültesinin 2009–2010 yılı güz dönemine aittir. Çizelgeleme haftanın 5 günü, günlük 10 ders ve süresi 40 dakika olan ders saatlerinden oluşmaktadır.

Fakültenin ders programı hazırlanırken, akademik programların ihtiyaçları göz önünde bulundurularak zorunlu ve esnek kısıtlar oluşturulmuştur. Deney verilerinin büyüklüğü ile ilgili veriler Çizelge 5.1’de gösterilmiştir.

Çizelge 5.1. Deney verilerinin büyüklüğü.

Öğretim Elemanı Sayısı	64
Derslik Sayısı	48
Sınıf Sayısı	32
Ders Sayısı	253
Ders Saati Sayısı	705
Akademik Program Sayısı	8

Deneyleerde turnuva seçim operatörü kullanılmıştır. Çaprazlama oranı ise %85 olarak sabit tutulmuştur. Problemin yapısına uygun bir mutasyon yöntemi geliştirilmiş ve mutasyon olasılığı %1 kabul edilerek deneyler yapılmıştır. Ayrıca her 20 iterasyon boyunca popülasyondaki en iyi bireyin uygunluk değeri değışmezse mutasyon oranı kademeli olarak %1 oranında arttırılmaktadır.

Deneyleerde iterasyon sayısı 1.000 olarak belirtilmiştir. Genetik algoritmada olası çözüm bulunurken hiç bir çakışma olmazsa, en iyi çözümün UD'i pozitif tam sayı değerlikli puan olmaktadır. Kısaca, algoritmanın çalışması esnasında 0 ve üzerinde bir UD'ye ulaşırsa, çakışma oranı %0 anlamına gelmektedir.

Arama işlemi, iterasyon sayısına, çakışma başarı oranına ya da kullanıcı isteğine bağlı olarak sonlandırılır.

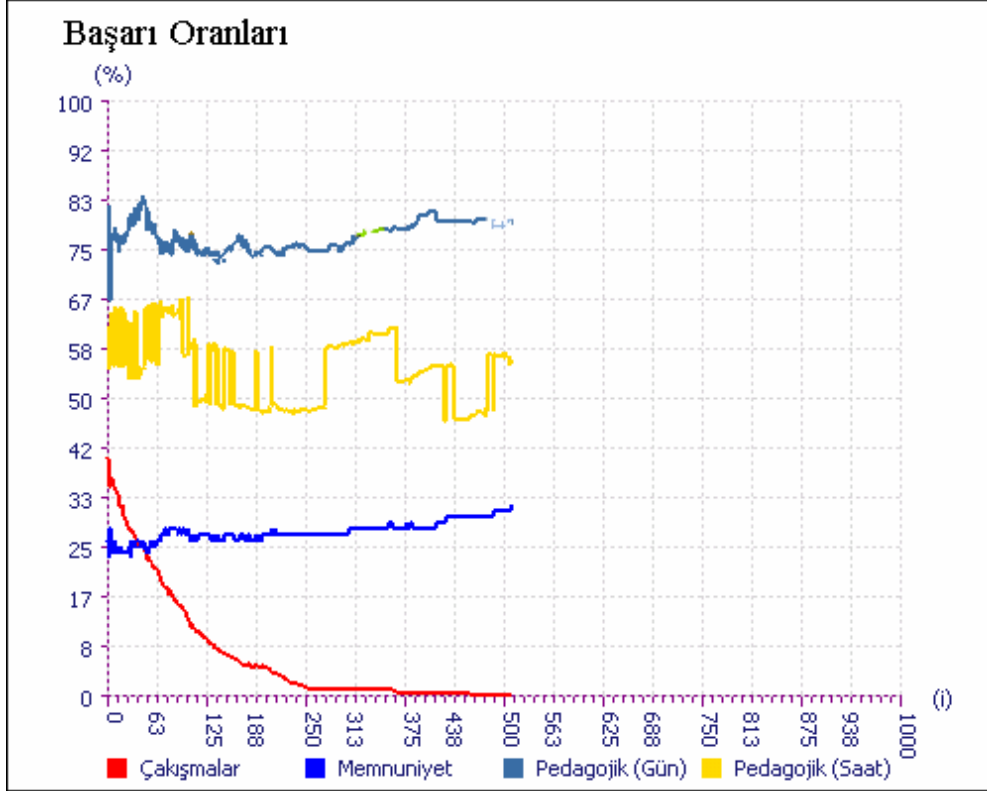
Arama uzayında, 1.000 iterasyon geçmesine rağmen halen istenilen sonuçlar bulunamaz ise algoritmanın çalışması durdurulur. 1.000 iterasyon sonrası çakışmaları bulunan ve bazı kısıtları yerine getiremeyen çözüm, tüm hata kodlarıyla veritabanına kayıt edilir. Böylelikle sonuçlarda esnek kısıtlara izin verilecek bir durum söz konusuysa, istenilen çözüm eğitim kurumu tarafından o dönemin ders programı çizelgelemesi olarak kullanılabilir.

Deney No: 1

Bu deneyde popülasyon sayısı 25, çaprazlama oranı %85 ve mutasyon olasılığı %1 olarak seçilmiştir. Deneyde periyodik çaprazlama operatörü kullanılmıştır. Mutasyon operatörü için geliştirilen modele uygun özel mutasyon yöntemi kullanılmıştır. Çizelgedeki ders yükü ağırlığı dengeli olarak seçilmiştir. Öğle arası ders bölünmesine izin verilmemiştir. Dersler arasındaki boş saatlerin azaltılmasında düşük ceza uygulanmış ve pedagojik gün ve saat verimliliği de kısıtlara eklenmiştir.

Yapılan bu deneyde hem öğretim elemanı memnuniyeti hem de öğrenciler için pedagojik gün ve saat verimliliği dikkate alınarak bir çözüm oluşturulmuştur. Şekil 5.1'de görüldüğü gibi çakışma oranı 445. iterasyonda %0'a indirilmiştir.

Elde edilen sonuçlara göre öğretim elemanı memnuniyeti oranı %32, pedagojik gün %80 ve pedagojik saat verimliliği ise %57 olarak hesaplanmıştır.



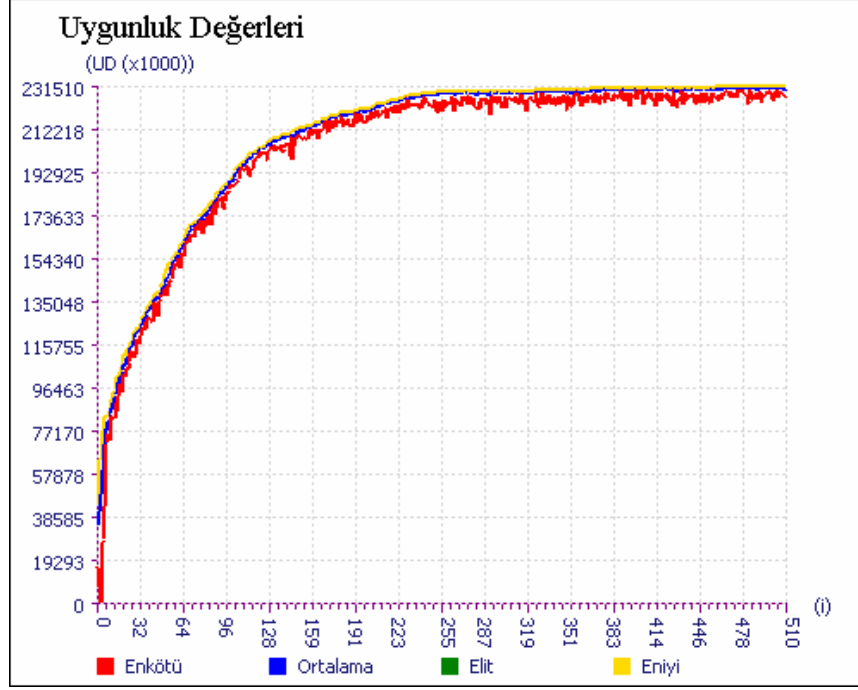
Şekil 5.1. Başarı oranları grafiği.

Şekil 5.2'de yapılan deney sonucunda elde edilen uygunluk değerleri (UD) gösterilmektedir. Yine şekil üzerinde belirtildiği gibi problemin çözümünde kullanılan en iyi UD sarı renkle, her bir iterasyondaki en iyi çözüm yeşil renkle, her iterasyondaki kromozomların ortalama değeri mavi renkle ve her iterasyondaki en düşük UD'ye sahip kromozom değerleri ise kırmızı renkle gösterilmektedir.

Deneylerdeki uygunluk değerleri grafiklerinde gösterilen UD değerleri, 0 ile en iyi çözümün UD değeri (UD_{MAX}) arasında tekrar ölçeklendirilerek negatif değerler kullanılmamıştır. Hesaplamalarda kullanılan çakışma cezaları negatif puan olarak belirlenmiştir.

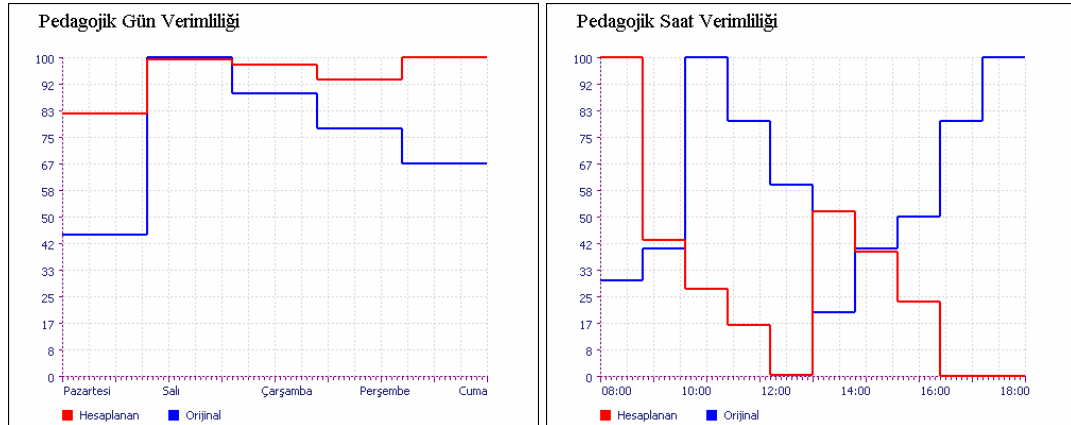
Şekil 5.3'te gösterilen pedagojik gün ve saat verimliliğinde, olması gereken değerler mavi renkle, genetik algoritma işlemleri sonucunda bulunan en iyi çözüm için

hesaplanan gün ve saat değerleri ise kırmızı renkle gösterilmiştir. Yapılan deney sonucunda gün verimliliği %80 oranında benzerlik gösterirken, saat verimliliği ise %57 olarak bulunmuştur.



Şekil 5.2. Uygunluk değerleri.

Bu deneyde, herhangi bir kısıt için öncelik tanınmamıştır. İterasyon devam ettirilerek öğretim elemanı memnuniyeti ve pedagojik gün-saat verimliliğinin daha iyi sonuçlar vermesi beklenebilir.



a) Pedagojik Gün Verimliliği.

b) Pedagojik Saat Verimliliği.

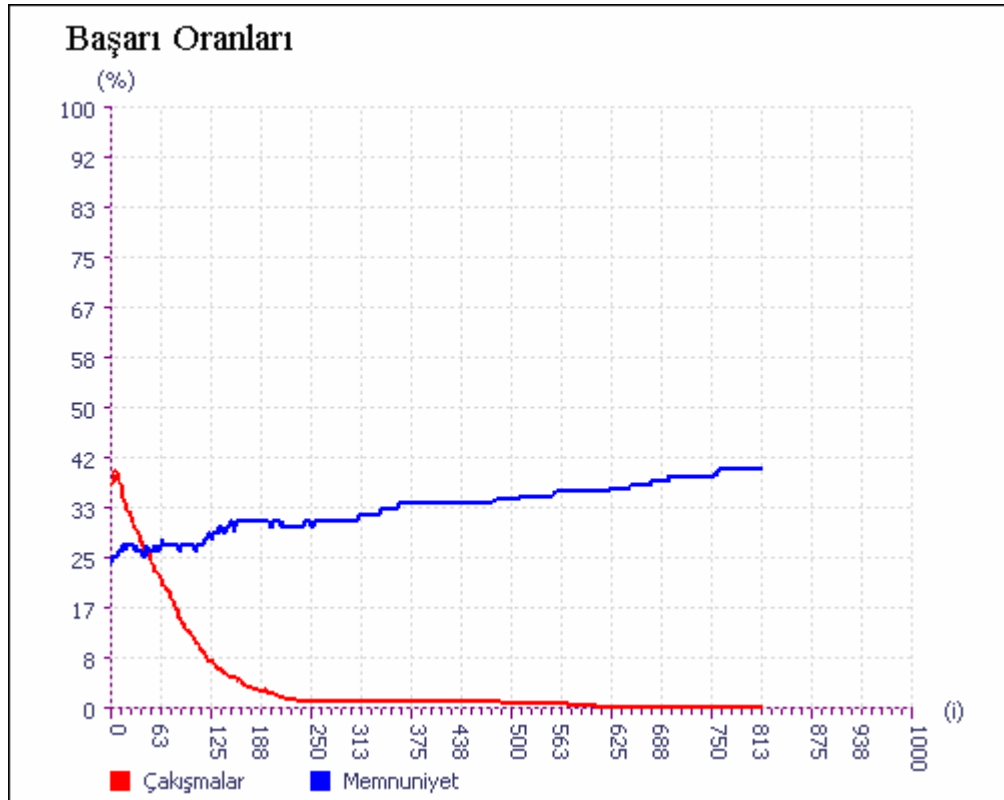
Şekil 5.3. Pedagojik gün ve saat verimliliği.

Deney No: 2

Bu deneyde popülasyon sayısı 20, çaprazlama oranı %85 ve mutasyon olasılığı %1 olarak seçilmiştir. Deneyde periyodik çaprazlama operatörü kullanılmıştır. Mutasyon operatörü için geliştirilen modele uygun özel mutasyon yöntemi kullanılmıştır.

Çizelgedeki ders yükü ağırlığı dengeli olarak seçilmiştir. Öğle arası ders bölünmesine izin verilmemiştir. Dersler arasındaki boş saatlerin azaltılmasında düşük ceza uygulanmış ve pedagojik gün ve saat verimliliği devre dışı bırakılmıştır.

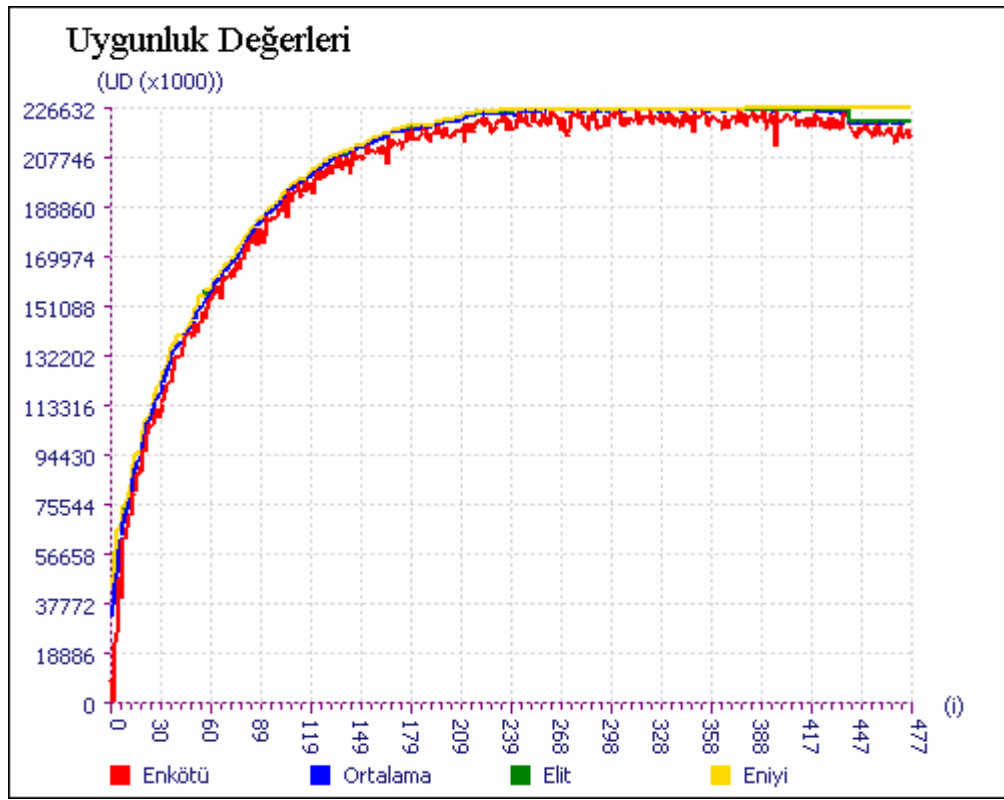
Yapılan bu deneyde öğretim elemanı memnuniyeti dikkate alınarak bir çözüm oluşturulmuştur. Şekil 5.4'de görüldüğü gibi çakışma oranı 607. iterasyonda %0'a indirilmiştir. Elde edilen sonuçlara göre öğretim elemanı memnuniyeti oranı %41 olarak hesaplanmıştır.



Şekil 5.4. Başarı oranları grafiği.

Şekil 5.5'te yapılan deney sonucunda elde edilen uygunluk değerleri (UD) gösterilmektedir. Yine şekil üzerinde belirtildiği gibi problemin çözümünde kullanılan en iyi UD sarı renkle, her bir iterasyondaki en iyi çözüm yeşil renkle, her iterasyondaki kromozomların ortalama değeri mavi renkle ve her iterasyondaki en düşük UD'ye sahip kromozom değerleri ise kırmızı renkle gösterilmektedir.

Bu deneyde, herhangi bir kısıt için öncelik tanınmamıştır. İterasyon devam ettirilerek öğretim elemanı memnuniyetinin daha iyi sonuçlar vermesi beklenebilir.



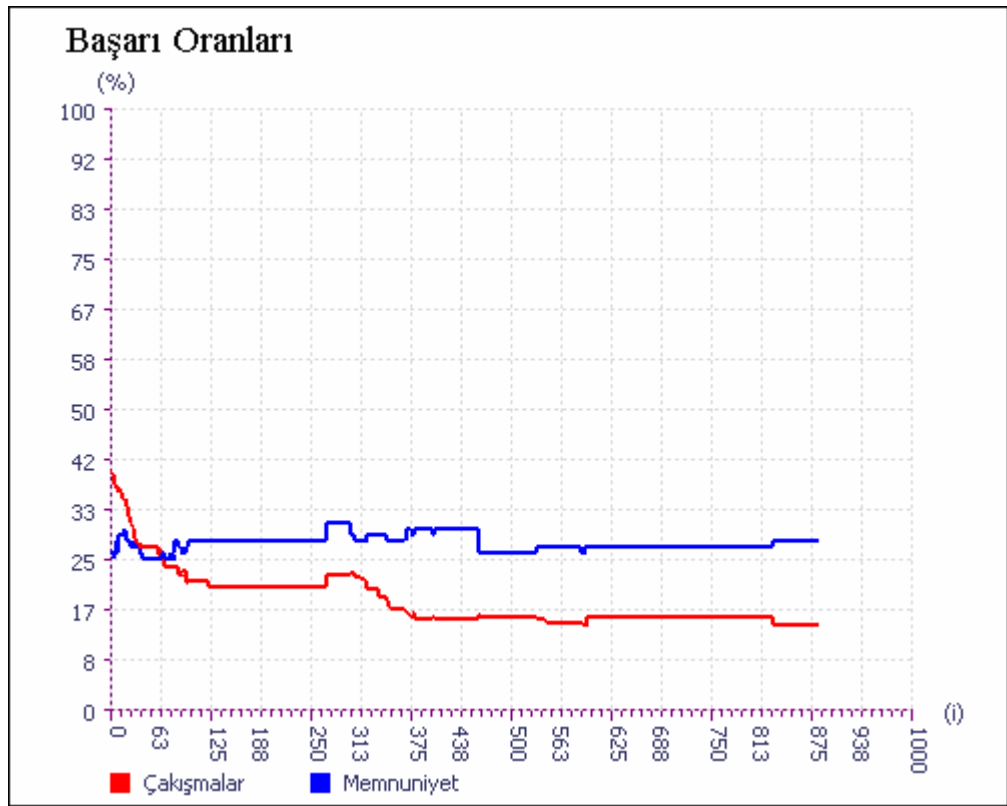
Şekil 5.5. Uygunluk değerleri.

Deney No: 3

Bu deneyde popülasyon sayısı 10, çaprazlama oranı %85 ve mutasyon olasılığı %1 olarak seçilmiştir. Deneyde periyodik çaprazlama operatörü kullanılmıştır. Mutasyon operatörü için geliştirilen modele uygun özel mutasyon yöntemi kullanılmıştır.

Çizelgedeki ders yükü ağırlığı dengeli olarak seçilmiştir. Öğle arası ders bölünmesine izin verilmemiştir. Dersler arasındaki boş saatlerin azaltılmasında düşük ceza uygulanmış ve pedagojik gün ve saat verimliliği devre dışı bırakılmıştır.

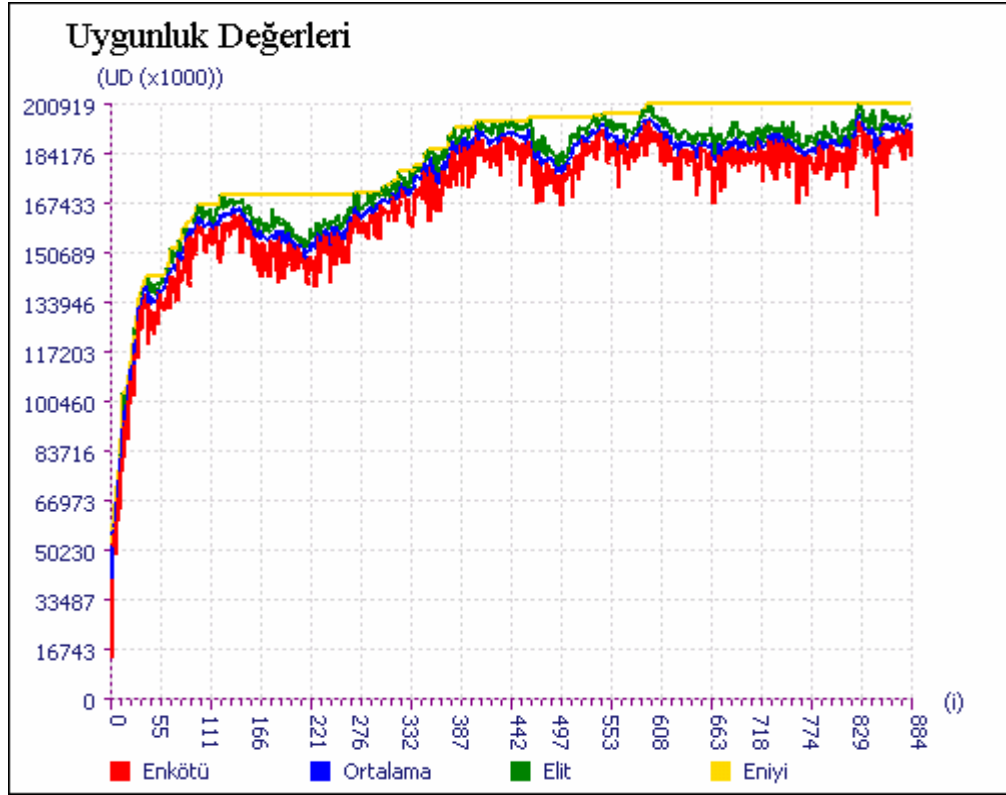
Yapılan bu deneyde öğretim elemanı memnuniyeti dikkate alınarak bir çözüm oluşturulmuştur. Şekil 5.6'de görüldüğü gibi çakışma oranı 880. iterasyonda bile halen %14 seviyesindedir. Elde edilen sonuçlara göre öğretim elemanı memnuniyeti oranı %28 olarak hesaplanmıştır.



Şekil 5.6. Başarı oranları grafiği.

Şekil 5.7'de yapılan deney sonucunda elde edilen uygunluk değerleri (UD) gösterilmektedir.

Kromozom uzunluğu 253 olan bu deneyde, popülasyon sayısının yetersiz olduğu tespit edilmiş ve popülasyon sayısında en ideal alt sınırın 25 olduğu gözlenmiştir.



Şekil 5.7. Uygunluk değerleri.

BÖLÜM 6

SONUÇLAR

Üniversite çizelgeleme probleminin çözümü için geliştirilen genetik algoritma ile çok kısa sürede fakülte, yüksekokul, enstitü ve meslek yüksekokullarına ait ders çizelgeleri %0 çakışma oranıyla oluşturulmuştur. Ayrıca öğretim elemanı memnuniyeti ve pedagojik gün-saat verimlilikleri de göz önünde bulundurularak optimum çözümler elde edilmiştir.

Yapılan deneylerde, öğrenci sayısı en fazla olan Karabük Üniversitesi Teknik Eğitim Fakültesi örnek olarak kullanılmıştır. 2009–2010 güz döneminde açılan 253 ders (toplam 705 saat), 48 farklı derslikte ve 64 farklı öğretim elemanı tarafından verilmiştir. Çizelgelerin ortalama oluşturulma süresi 15 dakika olarak hesaplanmıştır. Başka bir örnek olarak Karabük Üniversitesi Fen-Edebiyat Fakültesi, 2009–2010 güz döneminde açılan 56 ders (toplam 171 saat), 5 farklı derslikte ve 33 farklı öğretim elemanı tarafından verilmiştir. Çizelgelerin ortalama oluşturulma süresi ise 1–2 dakika arasında değişmektedir.

Optimum genetik algoritma parametreleri ve operatörleri incelendiğinde;

1. Problemin çözümünde en uygun seçim yöntemi Turnuva,
2. Periyodik çaprazlama yöntemi ve %85 çaprazlama oranı,
3. Mutasyon yönteminde ise geleneksel yapılara ek olarak bazı özel eklentiler yapılarak karşılıklı değişim metoduna benzer bir yöntem geliştirilmiş ve mutasyon oranı %1 olarak belirlenmiştir.
4. Son olarak popülasyon sayısı deneylerde kullanılan Teknik Eğitim Fakültesi için 25-35 arası en uygun değer olduğu saptanmıştır.

Literatür incelendiğinde, çizelgeleme problemlerin için yapılan çalışmaların büyük bir kısmında bölüm bazlı çizelgeleme yapıldığı görülmektedir. Çözüm uzayının büyük olması durumunda maliyet ve zaman problemleri ortaya çıkmaktadır. Aynı zamanda kısıtlama sayısının fazla olması performansı etkileyen diğer önemli faktörlerden birisidir.

Genetik Algoritma, yapılan arařtırmalar sonucunda çizelgeleme problemlerinin çözümünde en çok kullanılan yöntemlerden biri olduđu gözle çarpılmaktadır. Bunun sebebi GA'nın çözüm uzayında aynı anda birçok farklı noktadan arama yapabilmesi ve sonuca kısa bir sürede ulaşmasıdır.

Yapılan çalışmada, Karabük Üniversitesi'ne ait tüm akademik birimlerin çizelgeleri hazırlanmıştır. Performans arttırmaya yönelik her bir akademik birim (Fakülte, Yüksekokul, Enstitü) ayrı ayrı olarak çizelgeleri hazırlanmıştır. Her bir birim için ortalama çizelge hazırlama süresi 10-15 dk. arasında değişmektedir. Çizelgeleme işlemlerinin daha kısa sürede yapılabilmesi için bellek ve işlemci kapasitesi daha yüksek bilgisayarlar kullanılabilir.

KAYNAKLAR

1. Daskalaki, S., Birbas, T. and Housos E., “An integer programming formulation for a case study in university timetabling”, *European Journal of Operational Research*, 153: 117-135 (2004).
2. Bağış, A., “Genetik algoritma kullanılarak ders programının optimum şekilde düzenlenmesi”, Yüksek Lisans Tezi, *Erciyes Üniversitesi Fen Bilimler Enstitüsü*, Kayseri, 1-78 (1996).
3. Davis, L., “Job shop scheduling with genetic algorithms”, *Proceedings of the International Conference on Genetic Algorithms and Their Applications*, Carnegie Mellon University, Pittsburgh, PA, 136-140 (1985).
4. Liepins, G. E., Hilliards, M. R., Palmer, M. and Morrow, M., “GreedyGenetics: Genetic algorithms and their applications”, *Proceedings of the Second International Conference on Genetic Algorithms*, Cambridge, MA, 1-10 (1987).
5. Gupta, M. C., Gupta, Y. P and Kumar, A., “Minimizing flow time variance in a single machine system using genetic algorithms”, *European Journal of Operations Research*, 70: 289-303 (1993).
6. Lee, C. Y. and Choi, J. Y., “A genetic algorithm for job sequencing problems with distinct due dates and general early-tardypenalties”, *Computers and Operations Research*, 22: 857-869 (1995).
7. Wadhwa S. and Chopra A., “A genetic algorithm application: dynamic re-configuration in agile manufacturing systems”, *Studies in Informatics and Control*, 9: 1-9 (2000).
8. Yeniay, Ö., “An overview of genetic algorithms”, *Anadolu Üniversitesi Bilim ve Teknoloji Dergisi*, 2(1): 37-49 (2001).
9. Al-Milli, N. R., “Hybrid genetic algorithms with great deluge for course timetabling”, *IJCSNS International Journal of Computer Science and Network Security*, 10(4): 283-288 (2010).
10. Wijaya, T. and Manurung, R., “Solving university timetabling as a constraint satisfaction problem with genetic algorithm”, *In Proceedings of the International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, Depok, 1-10 (2009).

11. Jat, S. N. and Yang, S., “A guided search genetic algorithm for the university course timetabling problem”, *Multidisciplinary International Conference on Scheduling : Theory and Applications (MISTA)*, Ireland, 180-191 (2009).
12. Atanak, M. M. ve Hocaoglu, F. O., “Genetik algoritmalarla ders programi hazirlama otomasyonu tasarimi”, *3. Otomasyon Sempozyumu ve Sergisi*, Denizli, 1-78 (2005).
13. Bratković, Z., Herman, T., Omrčen, V., Čupić, M. and Jakobović, D., “University course timetabling with genetic algorithm: A laboratory exercises case study”, *Lecture Notes in Computer Science*, 5482: 240-251 (2009).
14. Aldasht, M., Alsaheb, M., Adi, S. and Qopita, M. A., “University course scheduling using evolutionary algorithms”, *4th International Multi-Conference on Computing in the Global Information Technology*, La Bocca, France, 47-51 (2009).
15. Aydın, M. A., “Solving university course timetabling problem using genetic algorithm”, Master Thesis, *Bahçeşehir University Institute of Sciences Industrial Engineering*, İstanbul, 1-78 (2008).
16. Kalender, M., “Ders çizelgeleme programi”, *TMMOB Elektrik Mühendisleri Odası İstanbul Şubesi 2006-2007 Öğretim Yılı Proje Yarışması*, 1-10 (2007).
17. Juang, Y., Lin, S. and Kao, H., “An adaptive scheduling system with genetic algorithms for arranging employee training programs”, *Expert Systems with Applications* 33: 642–651 (2007).
18. Baç, U., “Akademik ders programlarının yapılması probleminin matematiksel modeller ve algoritmalarla çözümü ve uygulaması”, Yüksek Lisans Tezi, *Hacettepe Üniversitesi Sosyal Bilimler Enstitüsü*, Ankara, 1-78 (2007).
19. Temiz, İ. ve Erol, S., “Bulanık akış tipi çizelgeleme problemi için çok amaçlı genetik algoritma”, *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, 22(4): 855-862 (2007).
20. Gülcü A., “Yapay zeka tekniklerinden genetik algoritma ve tabu arama yöntemlerinin eğitim kurumlarının haftalık ders programlarının hazırlanmasında kullanımı”, Yüksek Lisans Tezi, *Marmara Üniversitesi Fen Bilimleri Enstitüsü*, İstanbul, 1-78 (2006).
21. Datta, D., Deb, K. and Fonseca, C. M., “Solving class timetabling problem of IIT Kanpur using multi-objective evolutionary algorithm”, *KanGAL, Report No. 2006006*, India, 1-10 (2006).
22. Çayıroğlu İ., Dizdar, E. N., “Uzman sistem destekli online ders yerleştirme programi”, *Teknoloji Dergisi*, 9(4): 283-293 (2006).

23. Yiğit, T., “Meslek liseleri haftalık ders çizelgelerinin genetik algoritmalar yardımıyla oluşturulması”, *Gazi Üniversitesi Endüstriyel Sanatlar Eğitim Fakültesi Dergisi*, 19: 25-39 (2006).
24. Dammak, A., Elloumi, A. and Kamoun H., “Classroom assignment for exam timetabling”, *Advances in Engineering Software*, 37: 659-666 (2006).
25. Kazarlis, S., Petridis, V. and Fragkou, P., “Solving university timetabling problems using advanced genetic algorithms”, *5th International Conference on Technology and Automation*, Thessaloniki, Greece, 131-136 (2005).
26. Biroğul, S. ve Güvenç, U., “Genetik algoritma ile çözümü gerçekleştirilen atölye çizelgeleme probleminde ürün sayısının etkisi”, *Akademik Bilişim 2007*, Kütahya, 1-10 (2007).
27. Head, C. and Shaban, S., “A heuristic approach to simultaneous course/student timetabling”, *Computers & Operations Research*, 34: 919–933 (2007).
28. Güneş, A., Kahvecioğlu, A. ve Tuncel, H., “Askeri nöbet çizelgelerinin genetik algoritma kullanılarak eniyilemesi”, *Elektrik-Elektronik-Bilgisayar Mühendisliği 11. Ulusal Kongresi ve Fuarı*, İstanbul, 470-474 (2005).
29. Daban, F. ve Özdemir, E., “Çok parametrelili genetik algoritma kullanarak ders programlarının hazırlanması”, *2. Elektrik Elektronik Bilgisayar Mühendislikleri Eğitimi Sempozyumu*, 1-10 (2004).
30. Huerta-Amante, D. A. and Terashima-Marin, H., “Solving congress timetabling with genetic algorithms and adaptive penalty weights”, *The 5th International Conference for the Practice and Theory of Automated Timetabling*, Pittsburgh, PA U.S.A., 505–508 (2004).
31. Kanoh, H. and Sakamoto, Y., “Interactive timetabling system using knowledge-based genetic algorithms”, *IEEE International Conference on Systems, Man and Cybernetics*, 6: 5852-5857 (2004).
32. Özcan, E. ve Alkan, A., “Çok nüfuslu kararlı hal genetik algoritması kullanarak otomatik çizelgeleme”, *TBD 19. Bilişim Kurultayı*, 149-155 (2002).
33. Yu, E. and Sung, K., “A genetic algorithm for a university weekly courses timetabling problem”, *Intl. Trans. in Op. Res.*, 9: 703-717 (2002).
34. Lee, H. S. C. and Hermosilla, A. Y., “Timetabling constrained system via genetic algorithm”, *2nd Philippine Computing Science Congress*, Iligan City, Philippines, 1-10 (2001).
35. Nedjah, N. and Mourelle, L. M., “Timetabling with genetic algorithms”, *Proceedings of the 3rd WSEAS International Conference on Evolutionary Computation*, Switzerland, 3521-3525 (2002).

36. Bambrick, A. L., "Lecture timetabling using genetic algorithms", Master Thesis, *Department of Electrical and Computer Engineering The University of Queensland*, Queensland, U.S.A., 1-78 (1997).
37. Çivril, H., "Hemşire Çizelgeleme problemlerinin genetik algoritma ile çözümü", Yüksek Lisans Tezi, *Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü*, Isparta, 1-78 (2009).
38. Taç, K. C., "Genetik algoritma kullanılarak haftalık ders programı zaman çizelgeleme yazılımının geliştirilmesi", Yüksek Lisans Tezi, *Marmara Üniversitesi Fen Bilimleri Enstitüsü*, İstanbul, 1-78 (2006).
39. Goldberg, D. E., "Genetic algorithms in search optimization and machine learning", *Addion Wesley Publishing Company*, U.S.A., 1-10 (1989).
40. Colormi, A., Dorigo, M. and Maniezzo, V., "Genetic algorithms: a new approach to the time-table problem", *Published in Lecture Notes Computer Science, Springer Verlag*, 82: 235-239 (1992).
41. Michalewicz, Z., "Genetic Algorithms + Data Structures = Evolution Programs", *Springer Verlag*, USA, 1-10 (1996).
42. Gülsün, B., Tuzkaya, G. ve Duman, C., "Genetik algoritmalar ile tesis yerleşimi tasarımı ve bir uygulama", *Doğuş Üniversitesi Dergisi*, 10(1): 73-87 (2009).
43. İnternet: Obitko, M., "Genetic algorithms", Hochschule für Technik und Wirtschaft Dresden (FD) 1998, www.obitko.com/tutorials/genetic-algorithms (2010).
44. Biroğul, S., "Genetik algoritma yaklaşımıyla atölye çizelgeleme", Yüksek Lisans Tezi, *Gazi Üniversitesi Fen Bilimleri Enstitüsü*, Ankara, 28-47 (2005).
45. Topuz, V., "Bulanık genetik proses kontrolü", Doktora Tezi, *Marmara Üniversitesi Fen Bilimleri Enstitüsü*, İstanbul, 13-38 (2002).
46. Akbulut, D., "Evrimsel tasarım yöntemi ve yaratıcılığın süreç içerisindeki yeri", *Gazi Üniversitesi Güzel Sanatlar Fakültesi Sanat ve Tasarım Dergisi*, 2: 21-33 (2008).
47. Syswerda, G., "Uniform crossover in genetic algorithms", *In Proceedings of the Third International Conference on Genetic Algorithms*, U.S.A., 2-9 (1989).
48. Engin, O. ve Fığlalı, A., "Akış tipi çizelgeleme problemlerinin genetik algoritma yardımı ile çözümünde uygun çaprazlama operatörünün belirlenmesi", *Doğuş Üniversitesi Dergisi*, 6: 27-35 (2002).
49. Cheng, R., Gen, M. and Tsujimura, Y., "A tutorial survey of job shop scheduling problems using genetic algorithms: Part 2. Hybrid genetic search strategies", *Computers and Industrial Engineering*, 37: 51-55 (1999).

50. Engin, O., “Akış tipi çizelgeleme problemlerinin genetik algoritma ile çözüm performansının artırılmasında parametre optimizasyonu”, Doktora Tezi, ***İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü***, İstanbul, (2001).
51. Back, T., Fogel, D. B. and Michalewicz, T., “Evolutionary Computation 1: Basic algorithms and operators”, ***Institute of Physics Publishing*** , Bristol and Philadelphia, 1-10 (2000).
52. Binbaşıođlu, C., “Öğretim metodu ve uygulamaları”, ***Binbaşıođlu Yayınevi***, Ankara, 36-39 (1973).

ÖZGEÇMİŞ

Abdullah ELEN 1981 yılında Bursa’da doğdu; ilk ve orta öğrenimini aynı şehirde tamamladı. Tophane Anadolu Meslek Lisesi Bilgisayar Donanım Bölümü’nden 2001 yılında mezun oldu ve aynı yıl Uludağ Üniversitesi Teknik Bilimler Meslek Yüksekokulu Bilgisayar Donanımı Önlisans Programı’nda öğrenime başlayıp 2003 yılında iyi derece ile mezun oldu. 2003–2005 yılları arasında çeşitli firmalarda tekniker ve programcı olarak görev yaptı. 2005–2008 yılları arasında Zonguldak Karaelmas Üniversitesi Teknik Eğitim Fakültesi Bilgisayar Öğretmenliği programını tamamlayarak 2009 yılında Karabük Üniversitesi Fen Bilimleri Enstitüsü Elektronik ve Bilgisayar Eğitimi Anabilim Dalı’nda Yüksek lisansa başladı ve aynı yıl Karabük Üniversitesi’nde Öğretim Görevlisi olarak göreve başladı ve halen aynı yerde çalışmaya devam etmektedir.

ADRES BİLGİLERİ

Adres: 100. Yıl Mahallesi, Lale Kent Sitesi
A Blok Kat:2 Daire: 7,
78050, Merkez / KARABÜK
Tel: (0 507) 923 91 16
E-posta: aelen@karabuk.edu.tr

Ek I. Öğrenci işleri otomasyonunun veritabanı tablo yapısı ve ilişkilerinin gösterilmesi (ER-Diyagramı).

