



**ORTAÖĞRETİM KURUMLARI ORTAK SINAV ÇİZELGELEME
PROBLEMİNE SEZGİSEL ALGORİTMALARLA ÇÖZÜM YAKLAŞIMI**

Seval VATANSEVER

**YÜKSEK LİSANS TEZİ
ELEKTRONİK BİLGİSAYAR EĞİTİMİ ANA BİLİM DALI**

**GAZİ ÜNİVERSİTESİ
BİLİŞİM ENSTİTÜSÜ**

HAZİRAN 2020

Seval VATANSEVER tarafından hazırlanan “ORTAÖĞRETİM KURUMLARI ORTAK SINAV ÇİZELGELEME PROBLEMİNE SEZGİSEL ALGORİTMALARLA ÇÖZÜM YAKLAŞIMI” adlı tez çalışması aşağıdaki jüri tarafından OY BİRLİĞİ ile Gazi Üniversitesi Elektronik Bilgisayar Eğitimi Ana Bilim Dalında YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Danışman: Doç. Dr. Nursal ARICI
Bilgisayar Mühendisliği, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

.....

Başkan: Prof. Dr. Cevriye GENCER
Endüstri Mühendisliği, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

.....

Üye: Doç. Dr. Mustafa SERT
Bilgisayar Mühendisliği, Başkent Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

.....

Tez Savunma Tarihi: 24 / 06 / 2020

Jüri tarafından kabul edilen bu tezin Yüksek Lisans Tezi olması için gerekli şartları yerine getirdiğini onaylıyorum.

.....
Doç. Dr. Aslıhan TÜFEKÇİ
Bilişim Enstitüsü Müdürü

ETİK BEYAN

Gazi Üniversitesi Bilişim Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

İmza
Seval VATANSEVER
24 / 06 / 2020

ORTAÖĞRETİM KURUMLARI ORTAK SINAV ÇİZELGELEME PROBLEMİNE SEZGİSEL ALGORİTMALARLA ÇÖZÜM YAKLAŞIMI

(Yüksek Lisans Tezi)

Seval VATANSEVER

GAZİ ÜNİVERSİTESİ
BİLİŞİM ENSTİTÜSÜ

Haziran 2020

ÖZET

Ortaöğretim kurumlarında gerçekleştirilen ortak sınavlar için kapsamlı bir sınav çizelgesi hazırlanması gerekmektedir. Sınavların zaman ve yer olarak planlanması, gözetmen görevlendirmelerinin yapılması ve sınav oturum planlarının hazırlanması bu kapsam içerisindedir. Bu çizelgelerin insan çabası ile hazırlanması oldukça zahmetli ve zaman alıcıdır. Bu tez çalışmasında amaç; ortak sınav çizelgeleme probleminin tüm aşamalarına çözüm olacak sınav çizelgeleme yazılımını (GENÇİZ) geliştirmektir. Özellikle uygun sınav oturum planlarının hazırlanması ortaöğretimde ölçme ve değerlendirme sürecinin doğru ve güvenli işleyişi açısından oldukça önemlidir. Çalışmada hızlı ve kolay bir şekilde uygun oturum planları hazırlamak için genetik algoritma ve yapay bağışıklık sistemi kullanılarak iki farklı sezgisel çözüm yöntemi uygulanmıştır. Bu yöntemler, yapılan deneylerle karşılaştırmalı olarak incelenmiş, bulgular grafik ve tablolarla verilmiştir. Genetik algoritma ile daha kısa sürelerde daha iyi oturum planları elde edilmiştir. Bu sebeple GENÇİZ yazılımında oturum planlarının hazırlanması aşamasında genetik algoritma uygulanan yöntem kullanılmıştır. Yazılım ile farklı sınav sayıları, farklı sınav tarihleri, farklı sınıf yapıları ve farklı öğrenci sayılarıyla pek çok sınav senaryosu denenmiş ve başarılı sonuçlar elde edilmiştir.

Bilim Kodu : 92432

Anahtar Kelimeler : Ortak sınav, sınav çizelgeleme, oturum planı, sezgisel algoritmalar, genetik algoritma, yapay bağışıklık sistemi, optimizasyon.

Sayfa Adedi : 90

Danışman : Doç. Dr. Nursal ARICI

THE SOLUTION APPROACH TO THE COMMON EXAM TIMETABLING
PROBLEM OF SECONDARY EDUCATION INSTITUTIONS WITH HEURISTIC
ALGORITHMS
(M. Sc. Thesis)

Seval VATANSEVER

GAZİ UNIVERSITY
INFORMATICS INSTITUTE

June 2020

ABSTRACT

It is necessary to prepare exam timetabling for common exams performed in secondary education institutions. It includes the planning of exam place and time schedules, supervisors' assignment, and the preparation of exam session plans. It is difficult to prepare this timetabling by human afford. The purpose of this thesis study is to develop exam timetabling software (GENÇİZ) for common exams timetabling problems. It is crucial and necessary to prepare appropriate common exam session plans in secondary education institutions in terms of measurement and the evaluation of process reliability. It is applied two heuristic methods that are genetic algorithm and artificial immune system for the preparation of exam session plans quickly and easy. This method is examined comparatively with the experiments and findings are represented in tables and graphs. It is obtained better results with genetic algorithm. Therefore, genetic algorithm is utilized in developing software named as GENÇİZ. It is tried with different scenarios that include different exams number, exams date, class structures and students' number and it is obtained successful results.

Science Code : 92432

Key Words : Common exam, exam timetabling, session plan, heuristic algorithms, genetic algorithm, artificial immune system, optimization.

Page Number : 90

Supervisor : Assoc.Prof. Dr. Nursal ARICI

TEŞEKKÜR

Çalışmalarım boyunca etkili, yapıcı ve özenli rehberliği ile beni yönlendiren ve her aşamasında iyi ki benimle dediğim çok kıymetli danışmanım Doç. Dr. Nursal ARICI'ya, özellikle GENÇİZ yazılımının uygulanmasında idareci olarak gösterdiği katkılar ve aynı zamanda beni hep güçlü kılan manevi desteğinden dolayı sevgili eşim Veli VATANSEVER'e, tez çalışmalarım boyunca yüreklendirici konuşmaları ile sıcak ve lezzetli konukseverliği için ablam Nihal ALTINIŞIK'a, her gün en az iki saatlik tez çalışma süreme gösterdikleri anlayış için hayat enerjim, biricik oğullarım Onur VATANSEVER ve Mert VATANSEVER'e en içten teşekkürlerimi sunarım.



İÇİNDEKİLER

	Sayfa
ÖZET..	iv
ABSTRACT.....	v
TEŞEKKÜR.....	vi
İÇİNDEKİLER	vii
ÇİZELGELERİN LİSTESİ.....	ix
ŞEKİLLERİN LİSTESİ.....	x
RESİMLERİN LİSTESİ	xi
KISALTMALAR.....	xii
1. GİRİŞ	1
2. SINAV ÇİZELGELEME PROBLEMİ.....	5
2.1. Tanımı	5
2.2. Literatür İncelemesi	6
2.2.1. Genetik algoritma ile yapılan sınav çizelgeleme çalışmaları	8
2.2.2. Yapay bağışıklık sistemi ile yapılan sınav çizelgeleme çalışmaları	14
3. METODOLOJİ	15
3.1. Yönteme İlişkin Kuramsal Temeller.....	15
3.1.1. Sezgisel algoritmalar	15
3.1.2. Genetik algoritmalar	16
3.1.3. Yapay bağışıklık sistemleri	27
3.2. Ortak Sınav Çizelgeleme Probleminin Tanımı	37
3.3. Yöntem ve Geliştirilen Yazılım-GENÇİZ	39
3.3.1. Ortak sınav çizelgeleme problemi için geliştirilen yazılım-GENÇİZ	39
3.3.2. Genetik algoritmanın oturum planı hazırlama problemine uygulanması ..	53

Sayfa

3.3.3. Yapay bağışıklık sisteminin oturum planı hazırlama problemine uygulanması.....	61
4. YÖNTEME İLİŞKİN DENEY TASARIMI VE BULGULAR.....	69
4.1. Genetik Algoritma ile Oturum Planı Hazırlama Uygulama Sonuçları	69
4.2. Yapay Bağışıklık Sistemi ile Oturum Planı Hazırlama Uygulama Sonuçları.....	72
4.3. Genetik Algoritma ve Yapay Bağışıklık Sistemi Uygulanan Yöntemlerin Karşılaştırılması	77
5. SONUÇ VE ÖNERİLER	81
KAYNAKLAR	85
ÖZGEÇMİŞ	90

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 2.1. Sınav çizelgele problemi üzerine yapılan çalışmalar.....	7
Çizelge 3.1. Rulet tekeri seçim yöntemi örneği	22
Çizelge 3.2. GA’da uygulanan rulet tekeri seçim örneği.....	57
Çizelge 4.1. GA deney sonuçları (Elitizm oranı: %20)	69
Çizelge 4.2. GA deney sonuçları (Elitizm oranı: %10)	70
Çizelge 4.3. GA’nın farklı popülasyon büyüklüğü değerlerine göre çalışma sonuçları.....	71
Çizelge 4.4. GA’nın farklı iterasyon adedi değerlerine göre çalışma sonuçları	72
Çizelge 4.5. YBS deney sonuçları (Atık yüzdesi: %10).....	73
Çizelge 4.6. YBS deney sonuçları (Atık yüzdesi: %20).....	74
Çizelge 4.7. YBS’nin farklı atık yüzde değerlerine göre çalışma sonuçları	75
Çizelge 4.8. YBS’nin farklı popülasyon büyüklüğü değerlerine göre çalışma sonuçları.....	75
Çizelge 4.9. YBS’nin farklı mutasyon yöntemi seçimlerine göre çalışma sonuçları	76
Çizelge 4.10. YBS’nin farklı iterasyon adedi değerlerine göre çalışma sonuçları	77
Çizelge 4.11. Algoritma karşılaştırma deney sonuçları (Oturum şekli 5*3)	77
Çizelge 4.12. Algoritma karşılaştırma deney sonuçları (Oturum şekli 4*4)	79
Çizelge 4.13. Algoritma karşılaştırma deney sonuçları (Oturum şekli 6*3)	80

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 3.1. Genetik algoritma uygulamaları	18
Şekil 3.2. Tek noktalı çaprazlama örneği.....	23
Şekil 3.3. İki noktalı çaprazlama örneği	24
Şekil 3.4. Üniform çaprazlama örneği	24
Şekil 3.5. Rassal mutasyon örneği	26
Şekil 3.6. Bit tersleme mutasyon örneği	26
Şekil 3.7. Takas mutasyon örneği	26
Şekil 3.8. YBS'nin genel yapısı.....	30
Şekil 3.9. YBS algoritmaları.....	31
Şekil 3.10. Temsili bir ortak sınav saati örneği	37
Şekil 3.11. Temsili bir oturum planı örneği	38
Şekil 3.12. GENÇİZ yazılımına ait use case diyagramı	40
Şekil 3.13. Dizi tanımlaması.....	54
Şekil 3.14. Çaprazlamaya tabi olacak ebeveyn bireyler	58
Şekil 3.15. Çaprazlamada kullanılan maske ve elde edilen çocuk bireyler	58
Şekil 3.16. Çaprazlama sonrası tamir fonksiyonu örneği	59
Şekil 3.17. Antikorlara ait klon sayısı hesaplama örneği.....	64
Şekil 3.18. Rastgele mutasyon örneği.....	65
Şekil 3.19. İyileştirilmiş mutasyon örneği	66
Şekil 4.1. 20 deneme için algoritma tamamlanma süresi karşılaştırma grafiği (5*3).....	78
Şekil 4.2. 20 deneme için ceza puanı değişimi karşılaştırma grafiği (5*3)	78
Şekil 4.3. 20 deneme için algoritma tamamlanma süresi karşılaştırma grafiği (4*4).....	79
Şekil 4.4. 20 deneme için algoritma tamamlanma süresi karşılaştırma grafiği (6*3).....	80

RESİMLERİN LİSTESİ

Resim	Sayfa
Resim 3.1. GENÇİZ yazılımı ana ekranı	40
Resim 3.2. Oturum planı raporu	42
Resim 3.3. Oturum öğrenci listesi raporu	43
Resim 3.4. Öğrenci kayıt işlemleri ekranı	44
Resim 3.5. Sınava girecek sınıfların seçimi ekranı.....	45
Resim 3.6. Sınav salonu kayıt ekranı.....	46
Resim 3.7. Sınav salonu seç ekranı.....	47
Resim 3.8. Sınav dağıtımı ekranı.....	48
Resim 3.9. Ders kayıt işlemleri ekranı.....	49
Resim 3.10. Gözetmen kayıt işlemleri ekranı.....	49
Resim 3.11. Okul genel bilgileri kayıt ekranı	50
Resim 3.12. Veri tabanında yer alan tablolar.....	51
Resim 3.13. GA uygulanan oturum planı hazırlama programı kullanıcı arayüzü	61
Resim 3.14. YBS uygulanan oturum planı hazırlama programı kullanıcı arayüzü	68

KISALTMALAR

Bu çalışmada kullanılmış bazı kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Kısaltmalar	Açıklama
BAMA	Bağışık ağ modeli algoritması
GA	Genetik algoritma
KSA	Klonal seçim algoritması
KSP	Klonal seçim prensibi
NP	Non polynomial
NSA	Negatif seçim algoritması
ms	Milisaniye
SÇP	Sınav çizelgeleme problemi
YBS	Yapay bağışıklık sistemi

1. GİRİŞ

Eğitim sürecinin başarılı bir şekilde yürütülmesinde önemli öğelerden biri de ölçme ve değerlendirme sürecidir. Ölçme ve değerlendirme sürecinin sonucunda eğitim süreci ile ilgili önemli kararlar alınır. Öğretim sürecinin yeniden düzenlenmesi ve öğrencilerin bir üst öğrenime yerleştirilmesi bu kararlardandır (Tekindal, 2011: 2, 7).

Bu kararların doğruluğu ile etkinliği ölçme ve değerlendirme sürecinin doğru, kusursuz ve güvenli işleyişine bağlıdır. Bu süreçte en sık kullanılan araçlardan biri sınavlardır. Millî Eğitim Bakanlığı Ortaöğretim Kurumları Yönetmeliği'nde (Ortaöğretim Kurumları Yönetmeliği, 2013: madde 45(b)) belirtildiği üzere ortaöğretim kurumlarında aynı sınıf seviyesinde okutulan aynı derslerin sınavları ortak olarak yapılmaktadır. Ortaöğretim kurumları 9-12. sınıf düzeylerini kapsamaktadır. Her sınıf düzeyinde ortak sınava tabi tutulması gereken bir çok ders yer almaktadır. Ortak sınavlar için her sınav döneminde genelde iki haftalık bir süre sınav haftası olarak belirlenmekte ve sınavlar günde en fazla iki sınav olacak şekilde farklı sınıf düzeyleri için karma olarak gerçekleştirilmektedir.

Yapılan bu sınavların sonuçları öğrencilerin ortaöğretim mezuniyet ortalamalarını etkilemektedir. Ortaöğretim kurumlarından mezun olan öğrenciler bir üst öğrenime devam etmek istediklerinde ülke çapında gerçekleştirilen merkezi bir sınava girmektedirler. Sınav sonuçlarıyla birlikte mezuniyet ortalamaları üniversite kurumlarına başvuru puanlarının hesaplanmasında kullanılmaktadır. Öğrencilerin üniversite tercihlerinde çok küçük puan farkları bile büyük rol oynamaktadır. O halde ölçme ve değerlendirme sürecinde istenmeyen bir yardımlaşmanın gerçekleşmesi tüm öğrencileri olumsuz yönde etkilemektedir. Sürecin güvenilirliği bu yüzden oldukça önemlidir.

Ölçme ve değerlendirme süreci sadece notların belirlendiği bir süreç değildir. Sınavlardan sonra öğretme ve öğrenme sürecinin yeterliliğini değerlendirmek için sınav analizleri yapılmaktadır. Analizler ile belirlenen hedeflere ne kadar ulaşılabilirdi ortaya konmaktadır. Yapılan bu değerlendirmeler sonucunda elde edilen bulgulara dayanarak öğretme ve öğrenme süreci yeniden düzenlenebilmektedir. Ölçme ve değerlendirme süreci ne kadar doğru bir şekilde gerçekleşirse eğitim-öğretim sürecinin etkisi ve verimliliği de o kadar artacaktır.

Ortak sınavlar için sınavların tarih ve zaman olarak planlanması, sınav yerlerinin belirlenmesi, gözetmen öğretmen görevlendirmelerinin yapılması ve sınav oturum planlarının hazırlanması aşamalarını içeren kapsamlı bir sınav çizelgesi yapılması gerekmektedir. Bu çizelgelerin hazırlanması oldukça zahmetli ve hata yapmaya açıktır. Her sınav için sınav yerlerinin sayısı ve yerleşim yapısı, sınava girecek sınıf ve öğrenci sayıları farklılık göstermektedir. Değişen bu parametreler ile özellikle uygun oturum planlarının insan çabası ile hazırlanması oldukça zordur ve çalışma zamanı olarak maliyeti yüksektir.

Sınav yapılacak sınıfların ve gözetmen öğretmen sayısının kısıtlı olması sebebiyle ortak sınavlarda kalabalık öğrenci grupları aynı anda sınava girmektedir. Öğrenciler arasında istenmeyen yardımlaşmanın önüne geçmek için bazı önlemler alınması gerekmektedir. Aynı sınıf seviyesinde yer alan öğrencilerin çaprazda, yan yana ya da önlü arkalı gelmeyecek şekilde oturum yerleşiminin yapılması alınabilecek bu önlemlerden bazılarıdır. Bu suretle ölçme ve değerlendirme sürecinin doğru ve güvenli işleyişi sağlanabilmektedir.

Bu tez çalışmasında amaç; ortak sınav çizelgeleme probleminin tüm aşamalarına çözüm olacak sınav çizelgeleme yazılımını geliştirmektir. Bu kapsamda probleminin tüm aşamalarına destek olabilecek ortak sınav çizelgeleme yazılımı geliştirilmiştir. Bu yazılım GENÇİZ olarak adlandırılmıştır. Bu yazılımda uygulanan temel iş aşamaları şunlardır:

- Sınavların belirlenen tarih aralığında aynı günde aynı sınıf seviyesi için en fazla iki sınav olacak şekilde uygun zaman dilimine atanması,
- Uygun sınav yerlerinin belirlenmesi,
- Belirlenen sınav yerleri için sınav oturum planlarının hazırlanması,
- Sınavlara gözetmen atamalarının yapılması.

Uygun sınav oturum planlarının hazırlanması aşaması için iki farklı sezgisel yöntem uygulanmıştır. Bunlar; yapay bağışıklık sistemi (YBS) ve genetik algoritmadır (GA). GA literatürde sınav çizelgeleme problemlerinin (SÇP) çözümünde yaygın olarak kullanılan ve başarılı sonuçlar elde edilen sezgisel bir yaklaşım olduğu için tercih edilmiştir. YBS ise literatürde optimizasyon ve çizelgeleme problemlerinde kullanılmakla birlikte, nispeten daha yeni bir sezgisel yaklaşım olduğu için tercih edilmiştir. Her iki yöntemle de en uygun oturum planlarının hazırlanmasında başarılı sonuçlar elde edilmiştir. Farklı parametreler ile

algoritmaların çalışma performansını değerlendirmek için deneyler yapılmıştır. Elde edilen deney bulguları doğrultusunda daha iyi çözümlerin daha kısa sürede elde edilmesinden dolayı GA yöntemi GENÇİZ yazılımında oturum planlarının hazırlanması aşamasında tercih edilmiştir.

Tezin ikinci bölümünde SÇP kavramı ele alınmış ve literatürde SÇP üzerine yapılan çalışmalardan ulaşılabilenler özetlenmiştir. Üçüncü bölümde sezgisel algoritmalar, GA ve YBS hakkında kavramsal bilgiler sunulmuş ve ortak sınav çizelgeleme probleminin tanımı verilmiştir. Yine bu bölümde GENÇİZ yazılımının çalışma mantığı ve genel özellikleri ile sınav oturum planları için GA ve YBS kullanılarak uygulanan yöntemler ayrıntılı olarak açıklanmıştır. Dördüncü bölümde bu yöntemlerin uygulama sonuçları ile iki yöntemin performanslarının karşılaştırmalı incelemesine yer verilmiştir.



2. SINAV ÇİZELGELEME PROBLEMİ

Kesim 2.1’de genel olarak SÇP kavramı ele alınmış ve Kesim 2.2’de literatürde SÇP’ye çözüm yaklaşımı olarak sunulan çalışmalara yer verilmiştir.

2.1. Tanımı

SÇP; zaman, yer, öğrenci, gözetmen öğretmen öğeleri ile belirlenen esnek ve sert kısıtları göz önünde bulundurarak sınavları atama işlemidir. SÇP’nin çözümünde amaç kısıtları karşılayan sınav programını oluşturmaktır. Mutlaka yerine getirilmesi gereken kısıtlar sert kısıtlar; yerine getirilmesi istenen, fakat ihlali de kabul edilebilir kısıtlar esnek kısıtlar olarak tanımlanmaktadır. SÇP’de yer alan bu kısıtlar eğitim kurumlarının öncelik, ihtiyaç ve taleplerine göre kurumlar arasında farklılık göstermektedir. SÇP üniversite düzeyinde gerçekleştirilen tüm sınavlarda, lise düzeyinde gerçekleştirilen ortak sınavlarda ve merkezi olarak gerçekleştirilen tüm sınavlarda karşımıza çıkmaktadır. SÇP’de ele alınan problem basamakları dört aşamada değerlendirilebilir (Vatansever ve Arıcı, 2019).

1. Sınavların belirlenen tarih ve zaman aralıklarında uygun zaman dilimine atanması.
2. Sınavlar için uygun sınav yerlerinin belirlenmesi.
3. Sınav yerleri için uygun öğrenci oturum planlarının hazırlanması.
4. Sınav yerlerine gözetmen atamalarının yapılması.

Karar değişkenlerinin çok olması ve karmaşık karar yapıları sebebiyle SÇP’ler, NP (Non Polynomial) zor problem olarak adlandırılan optimizasyon problemleridir. Bu sebeple SÇP’lerinin çözümünde klasik algoritmalar yetersiz kalmaktadır. Genellikle bu problemlerin çözümünde sezgisel yöntemler uygulanmaktadır.

Optimizasyon problemlerin çözümünde kullanılan yapay zeka yöntemleri aynı zamanda meta sezgisel yöntemler olarak bilinmektedir. Bu yöntemler ile geliştirilen algoritmalar kesin çözümü garanti etmemekle birlikte kabul edilebilir sürelerde uygun bir çözüm sunarlar. Tüm optimizasyon problemlerinin çözümünde kullanılabilecek genel bir yöntemden bahsedilemez. Uygulama sonuçlarına göre bu yöntemlerin başarısı problemden probleme farklılık göstermektedir. Literatürde optimizasyon problemlerinin çözümü için geliştirilen ve uygulanan pek çok meta sezgisel yöntem vardır. Karınca kolonileri

algoritması, tabu araştırması algoritması, tavlama benzetimi algoritması, GA, diferansiyel gelişim algoritması ve YBS bunlar arasındadır (Coşkun, 2007).

2.2. Literatür İncelemesi

Literatürde SÇP üzerine yapılan çalışmalar incelendiğinde genellikle üniversite düzeyinde ele alındığı görülmüştür. Üniversite SÇP'lerinde sınavların öğrenci, sınav yeri ve zaman açısından çakışma olmadan planlanması oldukça önemli ve zorlu bir aşamadır. Ortaöğretim kurumları ortak sınav çizelgeleme probleminde ise en önemli ve planlanması zor olan aşama uygun sınav oturum planlarının hazırlanmasıdır.

Literatürde SÇP'ye çözüm yaklaşımı olarak sunulan pek çok çalışma bulunmaktadır. Kesim 2.2.1'de GA ile yapılan SÇP çalışmalarına, Kesim 2.2.2'de YBS ile yapılan SÇP çalışmalarına yer verilmiştir. Bunların haricinde SÇP üzerine yapılan çalışmalardan bazıları Çizelge 2.1'de özetlenmektedir.

Çizelge 2.1. Sınav çizelgele problemi üzerine yapılan çalışmalar

Çalışma Konusu	Çalışma Türü	Ele Alınan Kısıtlar	Kullanılan Yöntem	Yazar
Üniversitede sınavların uygun zaman dilimine atanması ve sınav yerlerinin belirlenmesi	Makale	- Sınavların zaman ve yer olarak çakışma olmadan planlanması. - Belirli şartlar dışında farklı sınavların aynı sınav yerinde yapılmaması. - Aynı sınav için sınav yerlerinin aynı binada olması. - Her bir öğrenci grubunun sınavlarının belirlenen zaman aralığında dağılımının eşit yapılması. - Aynı sınav için belirlenen sınav yerlerinin yakın mesafede olması. - Bir sınavın mümkün olduğunca tek bir sınav yerinde yapılması.	Graf tabanlı sezgisel yaklaşım	Kahar ve Kendall (2010)
Üniversitede sınavların uygun sınav yerlerine atanması ve gözetmen görevlendirilmesi	Makale	- Asgari sınav yeri kullanılması ve gözetmen ihtiyacının azaltılması. - Aynı gruba uygulanan ardışık sınavlarda sınavların aynı sınav yerinde ya da minimum mesafedeki sınav yerlerinde yapılması. - Öğretim görevlilerine girdikleri derslerin sınavlarında görevlendirme yapılmasının gözetilmesi.	Sezgisel bir yaklaşım	Ayob ve Malik (2011)
Ortaöğretim kurumlarında sınav oturumlarının belirlenmesi	Yüksek Lisans Tezi	Öğrencilerin kopya çekme ihtimalinin azaltılması	Bulanık mantık	Topal (2013)
Üniversitede sınavların uygun zaman dilimine atanması	Yüksek Lisans Tezi	- Sınavların çakışma olmadan uygun zaman dilimine atanması. - Öğrencilere sınavlar arasında uygun çalışma süresinin sağlanması.	Öğretme-öğrenme tabanlı optimizasyon algoritması	S.Yıldırım (2014)
Sınavların uygun zaman dilimine atanması, sınav yerlerinin belirlenmesi ve gözetmen görevlendirilmesi	Yüksek Lisans Tezi	- Öğretmenlerin mazeretleri ve öğrencilerin istekleri gözetilerek sınavların atanması. - Art arda gelen oturumlar için gözetmelere sınav görevi verilmemesi. - Gözetmenlere oturumların ortalaması kadar görev verilmesi. - Gözetmenler arasında görev sürelerinin dengeli olması ve kıdemi dikkate alınarak gözetmenlik görevleri verilmesi.	Karma tam sayılı programlama ile hedef programlama	Özdamar (2015)
Üniversite sınavlarında sınav oturum planlarının hazırlanması	Makale	- Kopya çekmeyi önlemek için aynı sınava girecek öğrencilerin yana yana oturmasının önlenmesi. - Sınav yerlerinin kapasitesinin aşılmaması. - Sınav oturumlarında maksimum doluluk sağlanarak az sayıda sınav yeri kullanımı.	Matematiksel programlama	Chaki ve Anirban (2016)
Üniversitede sınavların uygun zaman dilimine atanması	Makale	Aynı günde aynı öğrenci grubu için belirlenen sınav sayısının en aza indirgenmesi.	Matematiksel programlama ve tavlama benzetimi algoritması	Koca (2018)

2.2.1. Genetik algoritma ile yapılan sınav çizelgeleme çalışmaları

Kalaycı (2008) yüksek lisans tez çalışmasında; üniversite SÇP'sinin sınavların yapılacağı zaman dilimlerinin belirlenmesi aşamasında çalışan, öğrenci başarısını gözetken bir sınav çizelgeleme modeli geliştirmiştir. Geliştirilen modelde sınavların zorluk derecelerine bağlı olarak sınavlar arasında hazırlanmak ve dinlenmek için öğrencilere gerekli süreyi tanıyan sınav programının hazırlanması amaçlanmıştır. Modelde sınavların zorluk derecelerini de içeren cezalandırma temelli bir amaç fonksiyonu kullanılmıştır. Modelin geliştirilmesi ve test edilmesinde MATLAB programı kullanılmıştır. Çalışmada GA ile iki farklı yöntem uygulanmıştır. İlk yöntemde zaman dilimi çakışması yüksek ceza puanı ile cezalandırılarak önlenmeye çalışılmıştır. Tamir fonksiyonu modeli olarak adlandırılan ikinci yöntemde birebir çakışmaların düzeltilmeye çalışılması, düzeltilemediği takdirde kromozomun yok edilmesi sağlanmıştır. Tamir fonksiyonu modelinin diğer modele göre daha yavaş çalışmasına rağmen daha yüksek performans ortaya koyduğu belirtilmiştir.

Pillay ve Banzhaf (2010) çalışmalarında; SÇP'ye çözüm üretmek için bilgilendirilmiş GA olarak adlandırdıkları bir sistem geliştirmişlerdir. Çalışma GA'lar kullanılarak iki aşamalı olarak gerçekleştirilmiştir. Çalışmanın ilk aşamasında sert kısıtlamaları karşılayan çizelgeler üretilmiş; ikinci aşama da üretilen bu çizelgeler üzerinde esnek kısıtlamaları sağlayacak iyileştirmeler yapılmıştır. İlk popülasyon için çizelgelerin oluşturulmasında rastgele, en iyi slot ve sezgisel olarak adlandırdıkları üç farklı yöntem kullanmışlardır. Rastgele yönteminde hem atanacak sınav hem de zaman rastgele seçilmiştir. En iyi slot yönteminde atanacak sınav rastgele seçilmekle birlikte zaman için minimum maliyeti sağlayacak bir seçim yapılmıştır. Sezgisel olarak adlandırılan yöntemde ise sınavlar için zorluk derecesine göre, zamanda ise minimum maliyeti sağlayacak şekilde seçim yapılmıştır. Çalışmada GA parametrelerinden çaprazlama operatörü kullanımının geliştirilen sistemin performansını artırmadığı belirtilerek sadece mutasyon operatörü kullanılmıştır.

Arogundade, Akinwale ve Aweda (2010) çalışmalarında; Nijerya'nın Abeokuta şehrindeki Federal Ziraat Üniversitesi'nde SÇP için GA'lar ile çözüm geliştirmişlerdir. Çalışmada sert kısıtlar olarak; sınavların öğrenciler ve zaman açısından çakışma olmadan uygun zaman ve sınav yerlerine atanması, esnek kısıt olarak; hiçbir öğrencinin aynı günde üç sınava girmemesi gözetilmiştir. Aynı zamanda, öğrencilere sınavlar arasında dinlenme süreleri

oluşturmak üzere; sınav tarihleri süresince sınavları yaymaya çalışan bir amaç fonksiyonu kullanılmıştır.

Baysal (2011) yüksek lisans tez çalışmasında; sınavları uygun zaman dilimine atayan, sınav yerlerini belirleyen, sınavlarda görev alacak gözetmenleri atayan ve öğrencilerin sınav oturumlarını belirleyen bir uygulama geliştirmiştir. Uygulama iki aşamalıdır. İlk aşamada; sınav verileri ilgili birimlerden web ortamında alınmakta, ikinci aşamada ise geliştirilen masaüstü uygulaması ile sınav programı hazırlanmaktadır. Uygulamada sınavlar çakışma olmayacak şekilde rastgele yerleştirilmektedir. Çalışmada hangi öğrencilerin hangi sınıfta sınava gireceğinin belirlenmesinde; program sınavları sırayla tarayarak sınavlara girecek öğrencileri numaralarına göre listelemekte ve sınıfların kapasitesine göre bu listesinin bir üstünden bir altından öğrenci almak suretiyle sınıflara öğrenci yerleşimini gerçekleştirmektedir. Yani sınav yapılacak sınıflara öğrencilerin atanmasında herhangi bir özel koşul yoktur. Çalışma GA ile ASP.NET yazılımında geliştirilmiştir. GA'da cezalandırma yöntemi kullanılmış ve sistemi minimize etmek amaçlanmıştır. Çalışmada ebeveynlerden yeni çocuklar üretilmesinde sadece mutasyon işlemi uygulanmış, GA'nın önemli bir ögesi olan çaprazlama operatörü kullanılmamıştır.

Altıntaş (2011) yüksek lisans tez çalışmasında; SÇP sorununun çözümü için GA, karınca kolonisi algoritması ve tabu arama algoritması olmak üzere üç farklı sezgisel algoritma ile çalışan bir yazılım geliştirmiştir. SÇP'de sınavlar arasındaki sürenin uzatılması da gözetilerek sınavların uygun zaman dilimi ve sınav yerlerine atanması ile gözetmen ataması aşamaları ele alınmıştır. Kısıtların belirlenmesinde eğitim kalitesi, öğrenci ve öğretim elemanlarının performansı, öğretim elemanı istekleri ve idari ihtiyaçların temel alındığı belirtilmiştir. GA' da 20 nesil boyunca en iyi bireyin uygunluk değeri değişmediğinde, belirli miktarda artan mutasyon oranı kullanılmıştır. Yazılım ASP.Net C# programlama dili ile geliştirilmiştir. Yapılan deneylerde algoritmaların farklı parametreler ile çalıştırıldığı, fakat en iyi çözüme tam olarak hiçbiri ile ulaşamadığı belirtilmiştir. Bununla birlikte elde edilen çözümlerin tamir fonksiyonundan geçirilmesi suretiyle en iyi çözümlerin elde edilebildiği belirtilmiştir.

S. Yıldırım (2013) yüksek lisans tez çalışmasında; Işık Üniversitesi SÇP'sine çözüm için GA ile çözüm yaklaşımı geliştirmiştir. Çalışmada öğrencilere mümkün olduğunca aynı günde üç veya daha fazla sınav yapılmaması amaçlanmıştır. Çalışmanın anahtar

özelliklerinden birinin popülasyondaki ilk nüfusun belirlenmesinin üniversitenin diğer SÇP kısıtlamalarını karşılayan doğrusal bir tamsayı modelinin tam çözümüne dayanması olduğu belirtilmiştir. Geliştirilen yaklaşım Java programlama dilinde kodlanmış ve Işık Üniversitesinde kullanılan sınav çizelgeleme programına entegre edilmiştir.

Jha (2014) çalışmasında; İbra Bilgi ve İletişim Teknolojileri Mühendislik Bölümü SÇP'sinin çözümü için GA ile çalışan bir uygulama programı hazırlamıştır. Öğrencilere aynı günde birden fazla sınav atanmaması, sınavlar arasında uygun dinleme süresinin olması, sınav yerlerinde gerçekleşen ardışık zamanlı sınavlar olmaması gibi durumlar kısıt olarak ele alınmış ve ihlallerinde ceza puanı verilmiştir. Uygulama programı C dilinde yazılmıştır.

Bulut ve Subaşı (2015) çalışmalarında; merkezi sınavlardaki kopya çekme ihtimalini en aza indirmeyi amaçlayarak; birbiri ile tanışıklığı olan kişilerin aynı yerde sınava girmelerini önleyen bir model önermişlerdir. Çalışmalarını GA ile geliştirmişlerdir. Birbiriyle tanışık olan kişilerin aynı sınav salonunda yan yana veya arka arkaya oturma durumlarında ceza puanı verilmiştir. Tanışıklık durumunun belirlenmesinde sınıf arkadaşlığı, akrabalık ve aynı sokakta ya da semtte yaşama durumları değerlendirilmiştir.

Erden, Kökçam ve Demir (2016) çalışmalarında; üniversite SÇP'nin sınavların uygun zaman dilimlerine atanması, uygun sınav yerlerinin belirlenmesi ve sınavlara gözetmen ataması aşamalarının çözümü için GA ile çalışan bir yazılım geliştirmişlerdir. Çalışmada gözetmenlerin isteklerini mümkün olduğunca karşılayan uygun sınav çizelgeleri hazırlamak amaçlanmıştır. Kısıtların ihlalinde ceza puanı verilmiştir. Çalışma Visual Studio 2010 geliştirme ortamında C++ dili ile yazılmıştır.

Ağalday (2018) yüksek lisans tez çalışmasında; merkezi sınavlarda en iyi oturma planı optimizasyonunu gerçekleştirmeyi amaçlamıştır. Bu sınavlarda kopya çekme ihtimalini azaltmak için sınava giren bireylerden birbirini tanıma ihtimali olan kişilerin yan yana veya arka arkaya oturmalarını engellemek üzere öğrenci tanışıklığı durumunu değerlendiren bir yazılım geliştirmiştir. Öğrencilerin tanışıklığının değerlendirilmesinde; Mardin Artuklu Üniversitesi Uzaktan Eğitim Uygulama ve Araştırma Merkezi'nden alınan öğrencilere ait doğum yeri, okuduğu program ve ikamet bilgilerinin bulunduğu gerçek veriler kullanılarak öğrenciler arasındaki yakınlık ilişkisine bakılmaktadır. Yazılım GA ile Python

programlama dili kullanılarak geliştirilmiştir. GA’da cezalandırma yöntemi kullanılmış ve sistemi minimize etmek amaçlanmıştır. Çalışmada “İyileştirilmiş Tek Boyutlu Yöntem” ve “Katsayı Belirlemeye Dayalı Çok Boyutlu Yöntem” olarak adlandırılan iki GA modeli geliştirilmiştir. Tek boyutlu modelde öğrencinin okuduğu bölüm, doğum yeri ve ikamet adresi gibi tüm benzerlikler aynı kabul edilmiş ve yakınlık katsayısı 1 olarak alınmıştır. Çok boyutlu yöntemde yakınlık katsayıları aynı programda okuyan öğrenciler için 4, aynı bölgede ikamet edenler için 2 ve aynı doğum yerine sahip olanlar için 1 kabul edilmiştir. Çalışmada geliştirilen iki modelin farklı parametrelerin farklı değerleri ile karşılaştırma verilerine de yer verilmiştir.

Aksu (2018) tez çalışmasında; üniversite SÇP’sinde sınav yerlerine gözetmen ataması aşamasına çözüm için GA kullanmıştır. Görevlendirmelerde öğretim üyeleri arasında adil bir dağılım yapılması amaçlanmıştır. Sert kısıtlar olarak; sınav yerlerine ihtiyacı karşılayacak sayıda gözetmen görevlendirmesinin yapılması, görevlendirmelerde çakışma olmaması, aynı kişiye yapılan görevlendirmeler arasında en az 10 dakika olması ve öğretim üyelerine okulda olmadıkları günlerde gözetmenlik görevi verilmemesi belirlenmiştir. Esnek kısıtlar olarak; her bir öğretim üyesine akademik unvanına göre eşit sayıda görev verilmesi ve her bir öğretim üyesinin akademik unvanına göre toplamda eşit sürelerde görevlendirilmesi gözetilmiştir. Problemin çözümü için iki farklı GA uygulanmıştır. İkinci GA yönteminde algoritma çalışma zamanında en iyi uygunluk değeri belirli seviyeleri geçtiğinde popülasyon büyüklüğünde azaltma yapılarak, çözüm süresi kısaltılmaya çalışılmıştır. Gözetmen atama problemi için aynı zamanda matematiksel model oluşturulmuştur. Bu model ve iki GA modeli için karşılaştırma deneyleri yapılmıştır. Matematiksel model ile geliştirilen yazılımda karşılaşılan bellek hatası sorunu yüzünden hiçbir denemede optimum sonuca ulaşılamamıştır. Karşılaştırmalarda çözüm süresinde en iyi sonucu ikinci GA yöntemi, ortalama sonuçlarda en iyi sonucu birinci GA yöntemi vermiştir. Sonuç olarak bu tarz problemlerin çözümünde sezgisel algoritmaların kullanımının daha verimli olduğu belirtilmiştir.

Dener ve Calp (2018) çalışmalarında; merkezi sınavlar için GA ile çözüm yaklaşımı geliştirmişlerdir. Çalışmanın ilk aşamasında dersler için sınav oturumları belirlenmiştir. İkinci aşamasında ise belirlenen oturumlarda sınava girecek öğrencilerin sınav salonlarına yerleştirilmesi yapılmıştır. Çalışmada sınav oturumlarında yer alan ortak öğrenci sayısının

arttırılması, oturumlarda en az sayıda bina kullanımı ve minimum sınıf-sıra kullanılarak gözetmen görevlendirmesinin azaltılması amaçlanmıştır.

Seyfi (2018) yüksek lisans tez çalışmasında; sınavları uygun zaman dilimlerine atayan SCP'nin çözümü için GA, parçacık sürü optimizasyonu algoritması ve yapay arı kolonisi algoritması kullanarak üç farklı yöntem geliştirmiştir. Çalışmada belirtilen yöntemler ile SCP'nin çözülmesi, bu yöntemlerin karşılaştırılması, sınavlar arasında boş zamanların arttırılarak öğrencilere ders çalışabilecekleri zaman sağlanması ve tamir fonksiyonu kullanmayan çözümler kullanılması amaçlanmıştır. Çalışmada geliştirilen tüm algoritmalarının probleminin çözümündeki performansları incelenmiştir. En iyi sonuçların genelde GA yöntemi ile elde edildiği belirtilmiştir.

Suvay (2019) tez çalışmasında üniversite SCP'sine çözüm için GA'yı kullanarak bir web uygulaması geliştirmiştir. Uygulama C# programlama dilinde yazılmıştır. Uygulamaya ait veriler Sqlite veri tabanında tutulmuştur. GA iki aşamalı olarak ele alınmıştır. İlk aşamada; belirlenen tarih aralığına göre sınavların uygun zamanlara atanması, ikinci aşamada sınav salonu atamaları gerçekleştirilmiştir. Belirlenen sert ve esnek kısıtların ihlalinde ceza puanı verilmiştir. Uygulama Sivas Cumhuriyet Üniversitesi, İktisadi ve İdari Bilimler Fakültesi'nden alınan gerçek veriler ile sınav çizelgeleri hazırlamak için kullanılmıştır. Sonuçta GA'nın kısa zamanda SCP'de belirlenen kısıtları sağlayan çözümler üretmede başarılı olduğu belirtilmiştir.

Goli, Tirkolae, Mahdavi ve Zamani (2019) çalışmalarında; üniversite SCP'sine çözüm için GA ve ateşböceği algoritmasını kullanmışlardır. GA, geliştirilmiş GA, ateşböceği algoritması ve geliştirilmiş ateşböceği algoritması olarak adlandıkları dört farklı çözüm yaklaşımı geliştirmişlerdir. Sınavların zaman ve yer açısından çakışma olmadan planlanması, öğrencilerin arka arkaya sınavlarının olmaması, sınav yerlerinin sınava girecek öğrenci sayısına uygun olarak seçilmesi ele alınan kısıtlar arasındadır. Bunun yanında sınav yeri özel olan sınavlar ile aynı günde veya aynı anda yapılması gereken sınavların olması durumları da değerlendirilmiştir. Sonuçta en iyi uygunluk değeri sonuçlarının geliştirilmiş ateşböceği algoritması ile alındığı, fakat çalışma süresi olarak bu yaklaşımın geliştirilmiş GA'dan daha kötü performans ortaya koyduğu belirtilmiştir. Çalışmada GA'nın optimum sonuca yakınsamasını arttırmak için her nesilde en iyi

kromozomlar saklanmıştır. Bu uygulamayla birlikte bir noktada yerel optimuma takılma ihtimalini azaltmak için mutasyon oranının artırılması gerektiği belirtilmiştir.

Erdős ve Kővári (2019) çalışmalarında; akademik final sınavı programlamasına GA ile çözüm yaklaşımı geliştirmişlerdir. Bu programlarda sınavlar için uygun zaman dilimlerinin belirlenmesi ile birlikte her sınav için başkan, gözetmen ve sekreter atamalarının yapılması gerekmektedir. Algoritmanın uygulanmasında çaprazlama operatörü kullanılmamıştır. Mutasyon operatörü ise üç şekilde ele alınmıştır. İlki; algoritmanın yerel optimuma yakınsamasını engellemek üzere çözümler üzerinde uygulanan rassal mutasyondur. Diğerleri ise sırasıyla sert ve esnek kısıtları gerçekleştirmek üzere uygulanan mutasyonlardır. Mutasyonlar farklı kombinasyonlarda ve farklı olasılıklar ile uygulanmıştır. Tüm mutasyonlar kullandığında algoritmanın en iyi sonucu ürettiği belirtilmiştir. Algoritmalarını Budapeşte Teknoloji ve Ekonomi Üniversitesi Otomasyon ve Uygulamalı Enformatik Bölümü'nden alınan gerçek bir veri setinde test etmişlerdir. Veri seti 100 öğrenci 48 öğretmen ve 7 kurstan oluşmaktadır ve olası program sayısı yaklaşık 10462'dir. Sonuçta, uygulanan GA ile insan çabasıyla hazırlanmış programlardan daha iyi çözümler elde ettiklerini, bununla birlikte algoritmanın en büyük eksikliğinin görevlendirmelerde iş yükünde eşitliği sağlayamaması olduğu belirtilmiştir.

O. A. H. Hassan, Qtaish, Abuhamdeh ve M. A. H. Hassan (2019) çalışmalarında; üniversite SÇP'sine, grafik renklendirme ve GA'ları birleştiren melez bir çözüm yaklaşımı önermişlerdir. Sınav çizelgelerinin hazırlanmasında öğrenci memnuniyeti ön plana alınmıştır. Çalışmada öğrencilere sınavlar arasında çalışma zamanı oluşturmak üzere ardışık sınavlar arasındaki süreyi uzatmak ve öğrencilerin aynı günde iki sınava girme durumunu en aza indirmek amaçlanmıştır. Çalışmanın amaçlarından biri de çaprazlama ve mutasyon operatörlerinin etkisi ve gücünü değerlendirmektir. Değerlendirmelerin sonucunda; çaprazlama operatörüne kıyasla mutasyon operatörünün daha etkili olduğu ve hesaplama maliyeti açısından daha az maliyetli olduğu belirtilmiştir. Önerilen yaklaşım ile sınav dönemi boyunca öğrencinin rahatını artıran daha optimize edilmiş programlar oluşturmak konusunda ümit verici sonuçlar elde ettiklerini belirtmişlerdir.

2.2.2. Yapay bağışıklık sistemi ile yapılan sınav çizelgeleme çalışmaları

Malim, Khader ve Mustafa (2006) çalışmalarında; klonal seçim algoritması (KSA), negatif seçim algoritması (NSA) ve bağışık ağ modeli algoritması (BAMA) olmak üzere üç temel YBS algoritmasının eğitim çizelgeleri hazırlamak için uyarlanabileceğini göstermek istemişlerdir. Algoritmalarını ders programı çizelgeleme ve SÇP'nin sınavları uygun zaman dilimine atama aşaması için geliştirmişlerdir. SÇP ile ilgili olarak geliştirdikleri algoritmalarını 12 Carter veri setinde, ders programı çizelgeleme ile ilgili geliştirdikleri algoritmalarını 3 Schfafer veri setinde test etmişlerdir. Deneylerde algoritmaları beşer kez çalıştırıp karşılaştırmışlardır. Sonuç olarak; geliştirilen SÇP algoritmalarında KSA ve NSA'nın kaliteli çizelgeler üretmede BAMA'dan daha etkili olduğunu, ders programı çizelgeleme algoritmalarında NSA ve BAMA'nın KSA'dan daha etkili olduğunu belirtmişlerdir. İşlemci süresine bağlı olarak ise BAMA'nın sınav veri kümelerinde KSA ve NSA'dan daha hızlı çalıştığını, KSA'nın ders veri kümelerinde BAMA ve NSA'dan daha hızlı çalıştığını belirtmişlerdir.

Gonsalves ve Oishi (2015) çalışmalarında; Tokyo Sophia Üniversitesi Bilgi ve İletişim Bilimleri Bölümü için gerçek bir SÇP üzerine çözüm yaklaşımı geliştirmişlerdir. SÇP son sınıf sunumlarının sınav oturumlarının planlanmasını kapsamaktadır. Çalışmada SÇP'yi çözmenin yanı sıra YBS'nin optimizasyon problemlerinin çözümünde diğer algoritmalar kadar yetkin olduğunu kanıtlamak amaçlanmıştır. Çalışmada YBS algoritmalarından KSA kullanılmıştır. Sert kısıtlar olarak; sınav oturumları için uygun zaman ve yerlerin belirlenmesi ile gözetmenlerin atanması, esnek kısıtlar olarak; gözetmenlerin oturumlara atanmaları konusunda ki tercihleri belirlenmiştir. Algoritma iki aşamalı olarak ele alınmıştır. Birinci aşamada; sert kısıtlamaları ihlal etmeyen uygulanabilir çözümler oluşturulurken, ikinci aşamada; esnek kısıtlamaları karşılamak için birinci aşamadaki çözümler geliştirilmiştir. Aynı zamanda uygunluk değeri olarak kullanılan amaç fonksiyonu oluşturulurken kısıtların ihlalinde ceza puanı verilmiştir.

3. METODOLOJİ

Kesim 3.1’de sezgisel algoritmalar, GA ve YBS ile ilgili kavramsal bilgiler sunulmuştur. Kesim 3.2’de ortak sınav çizelgeleme probleminin tanımına yer verilmiştir. Kesim 3.3’te GENÇİZ yazılımı ve uygun sınav oturum planlarının hazırlanmasında uygulanan yöntemler ayrıntılı olarak açıklanmıştır.

3.1. Yönteme İlişkin Kuramsal Temeller

Kesim 3.1.1’de sezgisel algoritmalarından genel olarak bahsedilmiştir. Kesim 3.1.2 ve Kesim 3.1.3’de sırasıyla GA ve YBS ile ilgili temel kavramlar, çalışma prensipleri ve algoritma parametreleri ayrıntılı olarak açıklanmıştır.

3.1.1. Sezgisel algoritmalar

Bir problemin çözümünde beklentileri karşılayan birden fazla çözüm yöntemi olabilir. Burada önemli olan mühendislik çalışması bu çözümler içerisinde en iyisinin en kısa zamanda bulunmasıdır. Bu mühendislik çalışması optimizasyon problemi olarak adlandırılmaktadır. Bu problemler için yapılan bilimsel çalışmalar oldukça eskilere dayanmaktadır. Tüm mühendislik alanları ile ilgili olduğundan optimizasyon önemli bir bilim alanıdır. Son 30-40 yıldır optimizasyon problemleri üzerinde çalışan bilim adamları yeni yöntemler geliştirmek için doğal yöntemleri incelemeye yönelmişlerdir. Doğada var olan sistemleri ve yöntemleri temel alan optimizasyon algoritmaları sezgisel algoritmalar olarak adlandırılmaktadır. Popüler olarak kullanılan sezgisel algoritmalar aşağıdaki gibidir (Karaboğa, 2018: v).

1. Genetik algoritma
2. Diferansiyel gelişim algoritması
3. Karınca kolonisi algoritması
4. Parçacık sürü optimizasyonu algoritması
5. Tabu araştırma algoritması
6. Yapay arı kolonisi algoritması
7. Yapay bağışıklık sistemi algoritması
8. Yapay ısıtma işlem algoritması

3.1.2. Genetik algoritmalar

1950'lerde ve 1960'larda birkaç bilgisayar bilimcisi mühendislik problemlerinde optimizasyon aracı olarak kullanılabileceği düşüncesiyle evrimsel sistemleri incelediler. Bu incelemelerin sonucunda; doğal genetik çeşitlilikten ve doğal seçimden ilham alan operatörler kullanılarak belirli sorunlar için aday çözümler içeren popülasyonlar geliştirme fikri oluştu. 1960'larda Rechenberg, uçak folyoları gibi cihazlar için gerçek değerli parametreleri optimize etmek için kullandığı bir yöntem olan “Evrimsel Stratejilerini” (Evolutions Strategie) tanıttı. Evrimsel stratejileri evrimsel hesaplama alanının temelini oluşturmaktadır (Mitchell, 1996: 2, 3).

Çunkaş (2006: 6) GA'yı, “biyolojik süreci modelleyerek fonksiyonları optimize eden evrim algoritmaları” olarak tanımlanmaktadır.

GA, biyolojik evrimi taklit eden, doğal bir seçim sürecine dayanan, hem kısıtlı hem de sınırsız optimizasyon problemlerini çözmek için kullanılan bir yöntemdir (Ahmad, Sufahani, Ali and Razali, 2017).

GA'lar 1960'larda John Holland tarafından icat edildi ve 1960'larda ve 1970'lerde Michigan Üniversitesi'ndeki öğrencileri ve arkadaşları tarafından geliştirildi. Holland'ın asıl amacı, özel problemlerin çözümünde kullanılmak üzere algoritma tasarlamak değil; doğada ortaya çıkan adaptasyon mekanizmalarını incelemek ve bu mekanizmaları bilgisayar sistemlerine aktarmaktı. Holland'ın 1975'teki “Doğal ve Yapay Sistemlerde Adaptasyon” (Adaption in Natural and Artificial Systems) adlı kitabı GA'yı biyolojik evrimin bir soyutlaması olarak teorik bir çerçevede sundu. Holland'ın GA'sı bir kromozom popülasyonu üzerinde çaprazlama ve mutasyon gibi genetik süreçlerin işletilerek; bir çeşit doğal seleksiyon ile yeni bireylerin oluşturabildiği bir metot olarak tanımlanabilir (Mitchell, 1996: 3).

İlk başlarda GA pratik yararı olmayan bir araştırma konusu olarak düşünülmüştür. Holland'ın öğrencisi David E. Goldberg 1985 yılında gaz boru hatlarının GA ile optimizasyonu üzerine doktora tezini vermiş; 1989 yılında ise “Genetic Algorithm in Search, Optimization and Machine Learning” (Arama, Optimizasyon ve Makine Öğrenmede Genetik Algoritma) kitabını yayınlamıştır. Goldberg'in gaz boru hatlarının

denetimi üzerine yaptığı doktora tezinin 1985'te "National Science Foundation Genç Araştırmacı" ödülünü kazanması ve yayınladığı kitapta yer alan 83 farklı GA uygulaması GA'ların pratik kullanımının da olduğunu kanıtlamıştır (Goldberg, 1989 / Altıntaş, 2011: 8).

Genetik bilimindeki temel mekanizmaları model olarak oluşturulan GA ile özellikle optimizasyon problemlerinin çözümünde oldukça başarılı sonuçlara ulaşılmıştır (Özşen, 2009).

Birçok optimizasyon problemi karışık (sürekli ve ayrık) değişkenler içerdiğinden, arama alanlarında süreksizlik olduğundan standart doğrusal olmayan programlama teknikleri ile çözümleri etkin olmayacak ve yüksek hesaplama maliyeti gerektirecektir. Bu nedenle GA bu tarz problemlerin çözümü için iyi bir seçenektir. Bu algoritmaların en büyük yararı, karmaşık alanlarda sağlam bir arama sağlamaları ve çoğu diğer optimizasyon çözümleriyle karşılaştırıldığında, hesaplama söz konusu olduğunda genellikle daha az maliyetli olmasıdır. GA'lar çeşitli alanların büyük ölçekli optimizasyon problemlerinde geniş bir uygulama alanına sahiptirler (Bingül, Sekmen, Palaniappan ve Sabatto, 2000).

Çizelgeleme problemlerinin kısıt ve kaynaklara göre çözüm yöntemleri değişebilmekte ve çözüm uzayı matematiksel olarak kesin bir şekilde ifade edilememektedir. Bu gibi problemlerin çözümünde GA başarılı bir şekilde uygulanmaktadır (Çolak, 2015: 8).

SCP'de özellikle küçük boyutlu popülasyonlar ile çalışıldığında GA'nın uygulanması kolaydır ve etkili sonuçlar verir (Gashgari ve diğerleri, 2018).

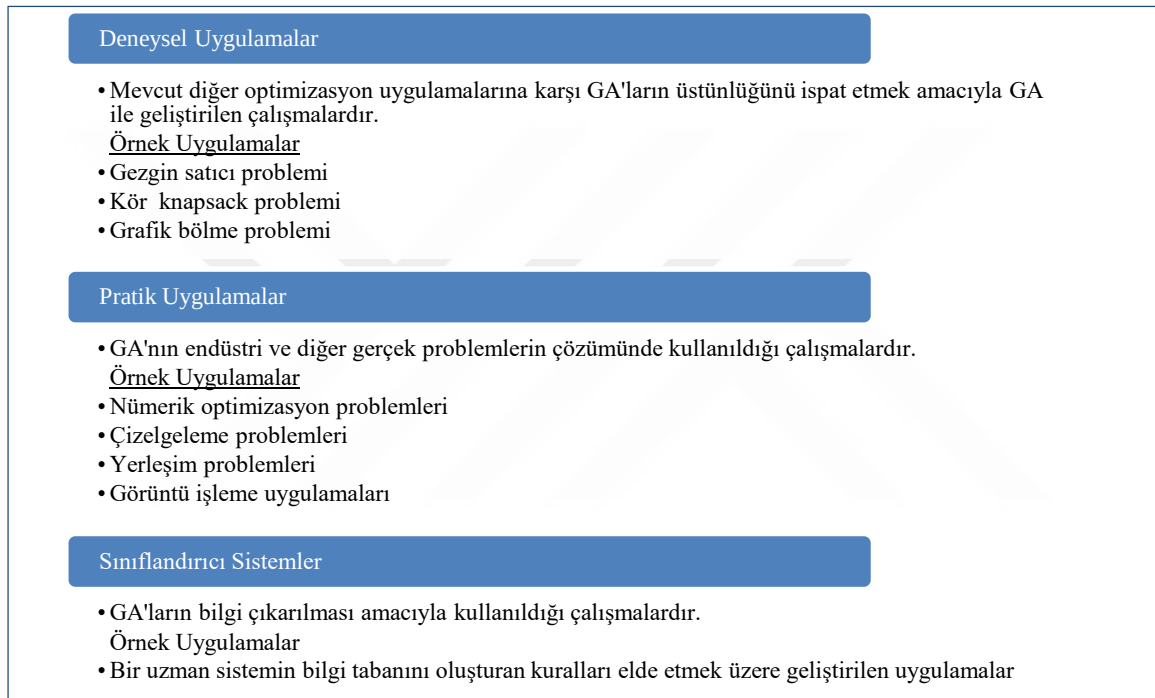
GA'lar değişen şartlara uyum sağlayabilen, ön bilgisi olmadan bilgiyi öğrenebilen ve toplayabilen algoritmalardır. GA'ların avantajları aşağıdaki gibi sıralanabilir (Çunkaş, 2006: 7).

- Türevsel bilgiler gibi yüksek düzey matematik bilgisine gerek duymazlar.
- Amaç fonksiyonunu geniş bir çözüm alanında araştırırlar.
- Çok sayıda parametre ile çalışabilirler.
- Paralel PC üzerinde çalıştırılabilirler.

- Karmaşık amaç fonksiyonu parametrelerini, yerel minimum veya maksimumlara takılmadan optimize edebilirler.

Genetik algoritma uygulamaları

GA'ların günümüzdeki uygulamaları üç ana gruba ayrılabilir. Bu uygulamalar Şekil 3.1'de verilmiştir.



Şekil 3.1. Genetik algoritma uygulamaları (Karaboğa, 2018: 86, 87)

Standart bir GA aşağıdaki kısımlardan oluşmaktadır (Nabiyev, 2016: 617, 618).

1. Olası çözümleri içeren popülasyon oluşturulur. Popülasyon büyüklüğünün belirlenmesinde problemin karmaşıklığı ve arama derinliği önemlidir. Olası her bir çözüm kromozom olarak adlandırılmaktadır. Her probleme özgü olarak kromozomlar farklı biçimde kodlanmaktadır.
2. Popülasyondaki her kromozomun uygunluk değeri hesaplanır. Bu amaçla kullanılan fonksiyona uygunluk fonksiyonu denir. GA'ların başarısı bu fonksiyonun başarısı ve hassasiyetine bağlıdır.
3. Seçilen kromozomlara çaprazlama ve mutasyon işlemleri uygulanır. Kromozomların seçimi uygunluk değerlerine bağlı olarak GA seçim

yöntemlerinden probleme göre uygun olanı seçilerek yapılmaktadır. Çaprazlama ve mutasyon ile popülasyonda çeşitlilik sağlanmaktadır. Çaprazlama ve mutasyon işlemleri için GA'da kullanılan farklı yöntemler vardır. Bazı problemlerin çözümünde çaprazlama ve mutasyon işlemine tabi olmuş kromozomlara, başlangıçta belirlenen kromozom kodlama yapısını koruyabilmek için tamir fonksiyonu da uygulanabilmektedir.

4. Probleme özgü olarak GA'da yeni kromozomlara yer açmak için eski kromozomlar popülasyondan çıkarılır. Bu işlem öncesinde elitizm olarak adlandırılan yöntemle uygunluk değeri daha iyi olan kromozomlardan bazıları korumaya alınabilir.
5. Yeni popülasyondaki kromozomlar için uygunluk değerleri hesaplanır.
6. İşlemler belirlenen sayılarda ya da en iyi çözüm bulunana kadar tekrarlanır. Her tekrarlama iterasyon olarak adlandırılmaktadır.
7. Sonuçta yeni popülasyonların oluşturulması sırasında en iyi bireylerin bulunduğu çözüm elde edilmiş olur.

Temel genetik kavramlar

GA'lar, mutasyon oranı, çaprazlama oranı ve popülasyon büyüklüğü gibi parametreler için iyi değerler seçildiğinde çok verimli çalışırlar. Bu parametreler en uygun şekilde ayarlandığında; GA daha az sayıda nesilde daha iyi uygunluk değerlerine sahip çözümler üretecektir (Bingül ve diğerleri, 2000).

Gen: Kromozom yapısında genetik bilgi taşıyan en ufak birimdir. Çözümü oluşturan bit yapıları olan genler ikilik sayı tabanında 0 ve 1 değeri alabileceği gibi programcının kromozomu kodlanmasına bağlı olarak gerçek değerlerde alabilmektedir.

Kromozom: GA'da kromozom problem için bir aday çözümü ifade eder ve bir dizi genin bir araya gelmesiyle oluşur. Kromozomlar problemin çözümünü ifade ettiği için doğru kodlanması oldukça önemlidir.

Kodlama: Bir GA uygulaması geliştirirken ilk adım problem için arama uzayını en iyi temsil edecek kodlama yapısının seçilmesidir. En yaygın olarak ikili kodlama kullanılmakla birlikte gezgin satıcı, çizelgeleme, karesel atama gibi eniyileme problemlerinde ikili kodlama arama uzayını tam olarak temsil edememektedir. Bu tarz

problemlerde alfasayısal veya gerçek sayısal gösterimlerle kodlama yapılmaktadır. Dolayısıyla kodlamanın ne şekilde yapılacağı çözümü istenen problemin özelliğine göre değişmektedir (Elmas, 2018: 443). Kullanılan kodlama yapıları aşağıdaki gibi sınıflandırılabilir (Nabiyev, 2016: 619, 620).

1. İkilik kodlama: Her kromozom, 0 ve 1'lerden oluşan bir bit dizisidir.
2. Permutasyon kodlama: Her kromozom bir numaralar dizisidir. Permutasyon kodlamanın kullanıldığı uygulamaya gezgin satıcı problemi örnek olarak verilebilir.
3. Değer kodlama: Her kromozom birtakım değerler dizisidir. Bu değerler problemin özelliğine göre gerçek sayı, karakter veya kompleks nesneler olabilir. Değer kodlamanın kullanıldığı bir uygulama olarak yapay sinir ağının ağırlık katsayılarının bulunması örnek olarak verilebilir.
4. Ağaç kodlama: Her kromozom nesneler ve nesneler arası işlemleri içeren bir ağaç yapısından oluşur.

Popülasyon: Kromozomların yani aday çözümlerin bir araya gelmesiyle popülasyon oluşur. Genellikle kullanıcı tarafından belirlenen bir parametre olan popülasyon büyüklüğü, GA'ların performansını etkileyen önemli faktörlerden biridir. Çözülmek istenen problemin özelliğine göre popülasyon büyüklüğü programcı tarafından belirlenmelidir. Popülasyon büyüklüğünün artması işlem süresinin gereksiz yere artmasına; azalması ise sistemin erken yakınsamasına, yani optimum çözüme ulaşmadan istenmeyen bir çözümde takılıp kalmasına sebep olabilmektedir (Sastry, Goldberg ve Kendall, 2005: 97, 98).

Uygunluk değeri: Kromozomların problemin çözümünde gösterdikleri başarıyı ifade eden; başka bir ifade ile mevcut kromozomun çözüme ne kadar uygun olduğunu gösteren değerdir. Yeni neslin oluşturulmasında hangi kromozomların seçileceğine; çaprazlama ve mutasyon işlemlerinden sonra yeni nesle hangi kromozomların aktarılacağına veya hangi kromozomların yok edileceğine uygunluk değeri ile karar verilir. Uygunluk değeri uygunluk fonksiyonu veya amaç fonksiyonu ile bulunur.

Amaç fonksiyonu ve uygunluk fonksiyonu: GA'nın başarısında aday çözümlerin değerlendirilmesinde kullanılan teknik oldukça etkilidir. GA ile bir problemin çözümünde mutlaka bir uygunluk fonksiyonuna veya amaç fonksiyonuna ihtiyaç vardır.

Kromozomların problemin çözümü için ne kadar iyi olduğunu bulan fonksiyona uygunluk fonksiyonu denir (Coşkun, 2006: 35). Bir diğer tanımla uygunluk fonksiyonu GA'da problemin çözümünde kromozomun yani aday çözümün yeterli olup olmadığını değerlendiren amaç fonksiyonudur. Birçok problemde amaç fonksiyonu ile sistem minimize ya da maksimize edilir. Sistem problemin uygunluk değerine göre minimize edilecekse uygunluk değeri düşük çözümler; maksimize edilecekse uygunluk değeri yüksek çözümler aranır. Çünkü amaç fonksiyonu bir maksimizasyon (maksimum kar veya fayda) ise aday çözümlerde başarı sağlanan kısıtlar için ödül puanı; amaç fonksiyonu bir minimizasyon (minimum zarar) ise çözümlerde başarı sağlanamayan kısıtlar için ceza puanı verilmektedir.

GA operatörleri

GA'da yer alan seçim, çaprazlama, mutasyon, değiştirme ve elitizm operatörleri aşağıda ayrıntılı olarak açıklanmıştır.

Seçim operatörü: Mevcut popülasyondan yeni nesiller oluşturma aşamasında hangi kromozomların bir sonraki nesle direkt aktarılacağı (elitizm) ya da hangi kromozomların ebeveyn bireyler olarak seçilerek, çaprazlama ve mutasyon işlemine tabi olacağının belirlenmesi işlemi seçim olarak adlandırılabilir. Bu seçim işlemi probleme özgü olarak belirlenen seçim yöntemleri ile gerçekleştirilir. Kullanılan seçim yöntemleri aşağıdaki şekilde sınıflandırılabilir.

Rulet tekeri seçimi (orantısal seçim): Kromozomların bir sonraki popülasyonda yer alma olasılıkları uygunluk değerlerine bağlıdır. Bu seçim yönteminde her bir kromozomun uygunluk değeri tüm kromozomların toplam uygunluk değerine bölünerek elde edilen oransal değer kullanılır (Satman, 2016: 109). Böylece her kromozom için 0-1 aralığında bir seçilme ihtimali elde edilmiş olur. Fakat uygunluk değeri daha iyi olan kromozomların seçilme garantisi yoktur. Bununla birlikte uygunluk değeri daha iyi olan kromozomların uygunluk değeri kötü olan kromozomlara göre seçilme olasılığı daha yüksektir. Seçilme ihtimalleri bulunduktan sonra birikimli seçilme değerleri hesaplanır. Seçim işleminde 0 ile 1 arasında üretilen rastgele sayı hangi birikimli seçilme değerleri arasında ise o kromozom seçilir. Örneğin Çizelge 3.1'de yer alan değerlere göre rulet tekeri seçim yönteminde belirlenen rastgele sayı 0,5 kabul edilirse; bu değer 0,461 ile 0,769 arasında yer aldığı için

4 numaralı kromozom ebeveyn birey olarak seçilecektir. Bu örnekte uygunluk değeri yüksek olan çözümler daha iyi kabul edilmektedir.

Çizelge 3.1. Rulet tekeri seçim yöntemi örneği

Kromozom	1	2	3	4	5
Uygunluk Değeri	20	12	4	24	18
Seçilme Olasılığı Değeri	$=20/(20+12+4+24+18)$ ~0,256	~0,154	~0,051	~0,308	~0,231
Birikimli Seçilme Değeri	~0,256	~,41	~0,461	~0,769	1

Turnuva seçimi: Bu seçim yönteminde mevcut popülasyonda yer alan kromozomlardan genelde iki olmakla beraber belirlen değer kadar (turnuva genişliği) kromozom rastgele seçildikten sonra uygunluk değeri yüksek olan kromozom ebeveyn birey olarak seçilmektedir.

Rank seçimi (sıralama seçimi): Rulet tekeri seçimi ile yapılan seçimlerde bazı kromozomların seçilme olasılıkları 1' e yakınsayabilir ve bu durumda popülasyonun diğer üyelerinin bir sonraki nesilde yer edinme şansları ortadan kalkabilir. Bu durumu önlemek için Rulet tekeri seçim yönteminin gelişmiş bir versiyonu olarak rank seçimi geliştirilmiştir. Bu seçim yönteminde kromozomların uygunluk değerine göre derecelendirmesi yapılmakta ve kromozomlara en kötü uygunluk değerine sahip olandan başlayarak sabit aralıklarda puan verilmektedir. Seçim işleminde her bir kromozomun puanı tüm kromozomların toplam puanına bölünerek elde edilen oransal değer kullanılır (Satman, 2016: 112).

Budama seçimi: Bu seçim yöntemi mevcut popülasyonun belirli bir oranının atılması işlemine dayanır. Popülasyonda yer alan kromozomların belirli bir oranı bir sonraki nesle direkt olarak aktarılacak yeni nesil oluşturulur. (Satman, 2016:111).

Çaprazlama operatörü: Mevcut popülasyondan yeni nesiller üretmek amacıyla çocuk bireyler elde etmek için seçilen kromozomlara çaprazlama işlemi uygulanır. Bu işlem uygulanırken ebeveyn bireyler seçildikten sonra 0-1 aralığında rastgele bir sayı üretilir. Üretilen sayı belirlenen çaprazlama oranından küçükse seçilmiş kromozomlar çaprazlama

işlemine tabi tutulur. Üretilen sayı çaprazlama oranından büyükse seçilen iki ebeveyn birey yeni nesle çocuk bireyler olarak direkt aktarılmaktadır. Çaprazlama oranı probleme özgü olarak programcı tarafından (0-1 aralığında reel bir sayı değeri) belirlenmektedir.

Çaprazlama oranı çaprazlama işleminin kullanım sıklığını kontrol eder. Genellikle 0,50 - 0,95 aralığında çaprazlama oranı kullanılmaktadır. Yüksek çaprazlama oranı, popülasyon değişikliğinin hızlı bir şekilde gerçekleşmesine sebep olurken; düşük çaprazlama oranı ise en iyi çözüm için aramanın yavaş gerçekleşmesine sebep olmaktadır (Elmas, 2018: 446).

Çaprazlamanın amacı, popülasyonda yer alan iyi çözümlerden daha iyi çözümler elde etmektir. Bu amaçla seçilen iki ebeveyn bireyin gen yapıları belirli tekniklerle birbirleriyle değiştirilerek çocuk bireyler elde edilmektedir. Literatürde kullanılan pek çok çaprazlama tekniği vardır. Aşağıda sık kullanılan çaprazlama tekniklerinden bazıları açıklanmıştır.

Tek noktalı çaprazlama: Popülasyonda yer alan kromozom yapılarının uzunluğuna göre rastgele bir değişim noktası belirlenir. Seçilen iki ebeveynden bu değişim noktasına göre genler alınarak çocuk bireyler oluşturulur. Örneğin kromozom uzunluğunun 6 olduğunu, değişim noktası olarak belirlenen rastgele sayının 2 olduğunu varsayarsak ilk çocuk birey ilk iki genini birinci ebeveynden sonraki genlerini ikinci ebeveynden alacaktır. İkinci çocuk birey ise ilk iki genini ikinci ebeveynden sonraki genlerini birinci ebeveynden alacaktır. Bu örnek Şekil 3.2’de gösterilmektedir.

1. Ebeveyn	1	0	1	1	1	0
2. Ebeveyn	0	1	0	1	1	0
1. Çocuk	1	0	0	1	1	0
2. Çocuk	0	1	1	1	1	0

Şekil 3.2. Tek noktalı çaprazlama örneği

İki noktalı çaprazlama: Tek nokta çaprazlamadan farklı olarak çaprazlama için rastgele 2 adet değişim noktası belirlenmektedir. Örneğin kromozom uzunluğunun 6 olduğunu,

değişim noktası olarak belirlenen noktaların sırasıyla 2 ve 4 olduğunu varsayarsak ilk çocuk birey ilk iki genini birinci ebeveynden sonraki iki genini ikinci ebeveynden ve son iki genini birinci ebeveynden alacaktır. İkinci çocuk birey ise ilk iki genini ikinci ebeveynden sonraki iki genini birinci ebeveynden ve son iki genini ikinci ebeveynden alacaktır. Bu örnek Şekil 3.3’de gösterilmektedir.

1. Ebeveyn	1	0	1	1	1	0
2. Ebeveyn	0	1	0	1	1	0
1. Çocuk	1	0	0	1	1	0
2. Çocuk	0	1	1	1	1	0

Şekil 3.3. İki noktalı çaprazlama örneği

Üniform çaprazlama: Bu çaprazlama yönteminde rastgele oluşturulan bir çaprazlama maskesi doğrultusunda iki ebeveynden genler alınarak çocuk bireyler oluşturulmaktadır (Coşkun, 2006: 42). Şekil 3.4’ de üniform çaprazlama örneği gösterilmiştir. İlk çocuk için çaprazlama maskesinde yer alan 0 değeri o genin birinci ebeveynden alınacağını, 1 değeri ise o genin ikinci ebeveynden alınacağını belirtmektedir. İkinci çocuk için çaprazlama maskesinde yer alan 0 değeri o genin ikinci ebeveynden alınacağını, 1 değeri ise o genin birinci ebeveynden alınacağını belirtmektedir.

1. Ebeveyn	A	B	C	D	E	F
2. Ebeveyn	G	H	I	J	K	L
Çaprazlama maskesi						
0 0 1 0 1 1						
1. Çocuk	A	B	I	D	K	L
2. Çocuk	G	H	C	J	E	F

Şekil 3.4. Üniform çaprazlama örneği

Mutasyon operatörü: Çaprazlama operatörün kullanılması sonucu eldeki mevcut popülasyonda yer alan kromozomlardan daha iyi uygunluk değerine sahip kromozomlar elde edilebilir. Fakat bir noktadan sonra kromozomlar kendini tekrar etmeye başlar. İki ebeveynde aynı gen değerlerine sahipse aday çözüm olarak farklı gen yapılarına sahip çocuklar elde etmek mümkün olmamaktadır. Popülasyon nüfusuna çeşitlilik katmak ve arama uzayını yeni çözümler ile genişletmek için mutasyon operatörü kullanılır. Evrimsel stratejilerin aksine, mutasyon genellikle GA'larda düşük olasılıkla gerçekleştirilen ikincil operatördür (Sastry ve diğerleri, 2005: 104, 105).

Çaprazlama operatörü ile seçilmiş kromozomlara benzer özellikte yeni aday çözümler üretilirken; mutasyon operatörü ile popülasyonda daha önce yer almamış yeni aday çözümler elde edilmektedir. Ancak bu değişimler sonucunda daha iyi aday çözümler elde edilebileceği gibi mevcut çözümler daha da kötüye gidebilir (Satman, 2016: 61). Bu yüzden mutasyon operatörü problemin özelliğine göre dikkatli bir şekilde kullanılmalıdır.

Mutasyon operatörünün kullanılmadığı bir GA'da kromozomların çaprazlama operatörü ile değişmediği varsayılırsa en iyi çözüm ancak, çözüm elamanlarının başlangıç popülasyonunda mevcut olması durumunda bulunabilir (Karaboğa, 2018: 79).

Popülasyon içerisinde yer alan aday çözümlerden en yüksek uygunluğa sahip olan yerel optimumdur. Bu çözüm çoğu zaman en iyi çözüm olmayabilir. Yeni neslin oluşturulmasında popülasyondaki en uygun bireyler seçildiğinden, algoritma popülasyonda yer alan en iyi çözüme yakınsama eğiliminde olacaktır. Mutasyonun amacı, popülasyon içerisinde çeşitliliği artırarak, en iyi çözüm için geniş bir alanda arama yapılmasını sağlamaktır. Böylece algoritmanın yerel optimuma yakınsaması önlenir (East, 2019: 7).

Mutasyon operatörü uygulanırken öncelikle probleme özgü olarak programcı tarafından 0-1 aralığında reel bir sayı değeri mutasyon oranı olarak belirlenir. Bu değer algoritmanın ilerleyişinde algoritmanın çalışma durumuna göre değiştirilebilir. Bir kromozomdaki mutasyon işlemi kromozomda yer alan genlerin tamamında bir bütün olarak ele alınabileceği gibi kromozomda yer alan genler için tek tek değerlendirme yapılarak da gerçekleştirilebilir. İlgili kromozom ya da ilgili gende mutasyon işleminin uygulanıp uygulanmayacağını belirlemek için 0-1 aralığında rastgele reel bir sayı üretilir, üretilen sayı mutasyon oranından düşükse ilgili kromozom ya da gene mutasyon işlemi uygulanır.

Mutasyon operatörü, popülasyondaki çeşitliliği sağlar ve genellikle düşük mutasyon oranı değeri ile uygulanır (Seyfi, 2018: 22). Mutasyon oranı ile mutasyon işleminin hangi sıklıkta olacağı belirlenir. Mutasyon oranının düşük olması arama uzayının tamamının araştırılmasını engellediğinden algoritma istenmeyen yerel çözümlerde takılabilir ve optimum sonuca ulaşamayabilir. Mutasyon oranının yüksek tutulması ise aday çözümlerin sürekli değişmesine sebep olarak çözümün rastgelelikte aranmasına sebep olabilir (Nabiyev, 2016: 624). Kullanılan bazı mutasyon teknikleri aşağıdaki gibidir.

Rassal mutasyon: Tekdüze mutasyon olarak ta adlandırılmaktadır. Rassal mutasyonda kromozomda rastgele seçilmiş bir gen değeri yine tanım aralığı içinde rastgele olarak belirlenen bir değer ile değiştirilir (Satman, 2016: 136). Şekil 3.5’de rassal mutasyon örneği gösterilmektedir.

(A E C D H F) → (A E B D H F)

Şekil 3.5. Rassal mutasyon örneği

Bit tersleme mutasyonu: Bu mutasyon yönteminde kromozom üzerinde bir veya daha fazla rastgele seçilen gen değeri ters çevrilir. Yani gen değeri 0 ise 1; 1 ise 0 yapılır. Bu mutasyon yöntemi kromozomların ikili olarak kodlandığı GA problemlerinde kullanılır (Seyfi, 2018: 22-23). Şekil 3.6’da bit tersleme mutasyon örneği gösterilmektedir.

(1 0 1 0 0 1) → (1 0 0 0 0 1)

Şekil 3.6. Bit tersleme mutasyon örneği

Takas mutasyonu: Bu mutasyon yönteminde kromozom üzerinde rastgele olarak iki nokta seçilir ve bu iki noktadaki gen değerleri karşılıklı olarak değiştirilir. Bu mutasyon yönteminin kullanımı permutasyon kodlama ile gösterim yapılan problemlerde yaygındır (Seyfi, 2018: 23). Şekil 3.7’de takas mutasyon örneği gösterilmektedir.

(3 1 2 4 5 6) → (3 6 2 4 5 1)

Şekil 3.7. Takas mutasyon örneği

Değiştirme operatörü (yeni popülasyonun oluşturulması): Çaprazlama ve mutasyon sonucu elde edilen çocuk kromozomların yani yeni aday çözümlerin yeni nesil olarak

oluşturulacak popülasyona aktarılmasıdır. Sık kullanılan değiştirme tekniklerinden bazıları aşağıda belirtilmiştir (Sastry ve diğerleri, 2005: 105).

1. Tümünü sil: Bu teknik mevcut popülasyonun tüm üyelerini siler ve onları yeni oluşturulmuş olan aynı sayıda kromozomla değiştirir.
2. Kararlı hal: Bu teknik ile n tane eski üye popülasyondan silinirken, onların yerine n tane yeni üye popülasyona eklenmektedir. Burada önemli olan n değerinin ne olacağı ve hangi üyelerin mevcut nüfustan silineceğine karar vermektir. n değerinin ne olacağı ve hangi n tane üyenin silineceği bu tekniğin parametreleridir. Silinecek üyeler en kötü uygunluk değerine sahip üyeler olabileceği gibi rastgele seçilmiş üyelerde olabilir.
3. Kararlı hal-yinelenmeyenler: Bu tekniği kararlı hal tekniğinden farkı algoritmanın popülasyona yinelenen kromozom eklenmediğini kontrol etmesidir. Bu kontrol ek hesaplama yüküne sebep olabilir ancak arama alanının çeşitliliğini artırır.

Elitizm operatörü: Yeni neslin oluşturulmasında mevcut popülasyonda yer alan en iyi çözümün bir sonraki nesle aktarılmasının bir garantisi yoktur. Elitizm operatörü ile mevcut nesildeki en iyi çözümlerinden bir kısmı yeni nesle aktarılabilir. Hangi sayıda bireyin yeni nesle doğrudan aktarılacağını elitizm oranı belirlemektedir. Bu oran problemin özelliğine göre programcı tarafından belirlenir. Elitizm oranının yüksek olması popülasyondaki çeşitliliği azaltıp, sistemin istenmeyen bir çözüme yakınsamasına neden olabilir. Elitizm oranının düşüklüğü ya da sıfır olması ise yeni nesilde bir önceki nesilden daha kötü uygunluk değerine sahip kromozomların yer almasına sebep olabilir.

3.1.3. Yapay bağışıklık sistemleri

İnsanoğlu doğayı model olarak problemlerin çözümünde kendine özgü metotlar geliştirmektedir. Problemlerin çözümünde kullanılmak üzere biyolojik sistemler temel alınarak oluşturulmuş metotlara Yapay Zekâ denir. YBS ise bağışıklık sisteminin matematiksel olarak modellenerek problemlerin çözümünde kullanılmasına dayanan bir yapay zekâ yöntemi olarak tanımlanabilir (Özşen, 2009).

Yapay zekâ ile ilgili çalışmaların başlangıcı 1940'lı yıllara dayanmakla birlikte ilk çalışmalar, insan beyninin çalışmasını model alan yapay sinir ağları ile 1950'li yıllarda

başlamış ve 1990'lı yıllarda hız kazanmıştır. Çoğu kişi tarafından yapay zekâ ile yapay sinir ağları eş değer tutulsa da yapay sinir ağları tek yapay zekâ yöntemi değildir. 1960'lı yıllardan sonra yapılan araştırmalarla bağışıklık sisteminin daha anlaşılır olmasıyla birlikte modellenecek bir sistem olarak bağışıklık sistemi irdelenmeye başlanmıştır. Bağışıklık sisteminde yer alan özelliklerin bir yapay zekâ sistemindeki problem çözme için gerekli olan pek çok özelliği karşıladığının görülmesiyle birlikte YBS çalışmaları başlamıştır. 1970'li yıllarda temeli atılan YBS çalışmaları özellikle 1990'lı yıllarda yoğunlaşmıştır. 1990-2002 yılları arasında bağışıklık sisteminin işleyişine katkıda bulunan pek çok mekanizma modellenmiştir. Bu modeller virüs tespiti, sınıflama problemleri, robotik uygulamalar ve optimizasyon problemleri gibi pek çok alanda kullanılmıştır. Bu uygulamalar ve kuramsal modelleme çalışmaları 2002 yılından sonra hız kazanmıştır (Özşen, 2009).

Doğal bağışıklık bilimin gelişmesi 1976 yılında Edward Jenner'in aşığı bulmasıyla başlamıştır. 1974 yılında Niels K.Jerne'nin "Bağışıklık Ağı Teorisi" adlı çalışması ve 1983 yılında Susumu Tonegawa'nın reseptör çeşitliliği ve yapay bağışıklık üzerine yaptığı çalışma Nobel Ödülü'ne layık görülmüştür. 1995 yılından sonra gelişmeye başlayan yapay zekâ çalışmalarında çoğunlukla insan biyolojik sisteminin çalışması model alınmıştır. 1990'lı yılların sonlarından itibaren YBS'lere yapay zekâ araştırmacılarının ilgisi artmıştır (Şahan, 2004: 29, 30).

Literatürde YBS alanında çalışan ve geliştirdikleri algoritmaları YBS algoritması olarak tanımlayan bir çok kişi olmasına rağmen, YBS'nin ne olduğu, ne zaman ortaya çıktığı ve en önemlisi, bir YBS'nin nasıl karakterize edileceği ve tasarlanacağı konusunda bir fikir birliği yoktur (De Castro ve Timmis, 2002).

Forrest tarafından 1996'da bilgisayar virüslerini tespit etmek için geliştirilen çalışma YBS'nin ilk uygulaması olarak kabul edilmektedir (Alaca, 2018: 42).

Tamamen YBS ile ilgili çalışmaların sunulduğu en önemli konferans olan "Uluslararası Yapay Bağışıklık Sistemleri Konferansı" (International Conference on Artificial Immune Systems- ICARIS), 2002'den beri her yıl düzenlenmektedir. Ayrıca, YBS'ler üzerinde yapılan çalışmaların sunulduğu diğer uluslararası konferanslar şunlardır: "Genetik ve Evrimsel Hesaplama Konferansı" (Genetic and Evolutionary Computation Conference -

GECCO), “IEEE Evrimsel Hesaplama Kongresi” (IEEE Congress on Evolutionary Computation), “IEEE Dünya Hesaplamalı Zeka Kongresi” (IEEE World Congress on Computational Intelligence - WCCI) ve “Uluslararası Yapay Zeka ve Esnek Hesaplama Konferansı” (International Conference on Artificial Intelligence and Soft Computing - ICAISC) (Dasgupta, Yu ve Nino, 2011).

Doğal bağışıklık sistemini oluşturan prensipler ve mekanizmalardan esinlenerek oluşturulan YBS algoritmalarının kullanım alanlarını aşağıdaki şekilde sınıflandırabilir (Atay, 2012: 1).

1. Çizelgeleme
2. Sınıflandırma
3. Optimizasyon
4. Veri Madenciliği
5. Bilgisayar ve Ağ Güvenliği

Doğal bağışıklık sistemi, insan vücudunda yer alan hücrelerde meydana gelen işlev bozuklukları ve vücuda dışarıdan girerek hastalığa sebep olan yabancı mikroorganizmalara karşı hücre, molekül ve organlardan oluşan savunma mekanizmasıdır (De Castro ve Von Zuben, 1999).

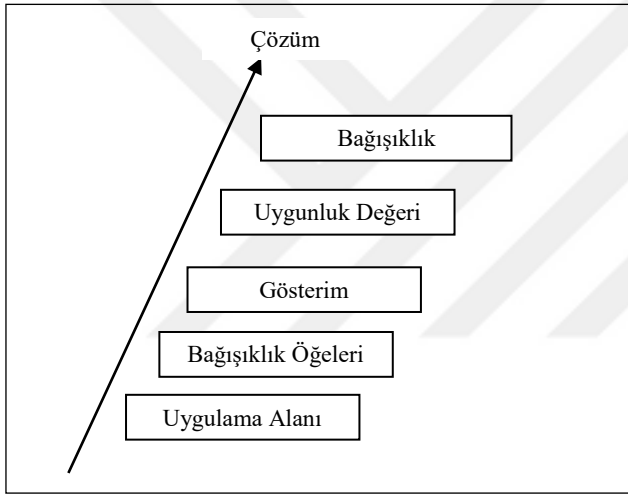
Doğal bağışıklık sistemi bakteri, virüs ve diğer hastalıklara neden olan organizmalara karşı temel savunma sistemimizdir. Savunma fonksiyonlarını etkin bir şekilde sağlamak için, bağışıklık sistemi öz molekülleri ve hücreleri, yabancı olanlardan (antijenler) ayırmak için örüntü tanıma görevlerini yerine getirmelidir. Bağışıklık sisteminin tanıyabileceği yabancı moleküllerin sayısı bilinmemektedir, ancak 10^{16} 'dan büyük olduğu tahmin edilmektedir. Bu muazzam tanıma kapasitesinin yanı sıra, omurgalı bağışıklık sistemini düşük organizmaların savunma sistemlerinden ayıran diğer bir özellik de bağışıklık sisteminin hafızayı öğrenmesi ve sergilemesidir. Böylece aynı antijenin ikinci saldırısına verilen tepki, ilk tepkiden daha hızlı ve kuvvetli bir şekilde gerçekleşir (Forrest ve Perelson, 1991).

YBS'ler; problemlerin çözümü için bağışıklık sisteminin teorik yapısını, bağışıklık fonksiyonları, prensipleri ve mekanizmalarını temel alarak geliştirilen bilgisayar

sistemleridir. Bir YBS sisteminin tasarımı için çerçeveyi oluşturan temel unsurlar kavramsal olarak aşağıdaki gibidir (De Castro ve Timmis, 2003).

1. Sistemi oluşturan bileşenleri temsil etmek için biçimsel bir yöntem belirlenir.
2. Bileşenlerin her birinin kalitesini bir anlamda çözüm için uygunluğunu değerlendiren fonksiyonlar belirlenir.
3. Sistemin genel davranışlarını yöneten ve kontrol eden bir dizi algoritma oluşturulur.

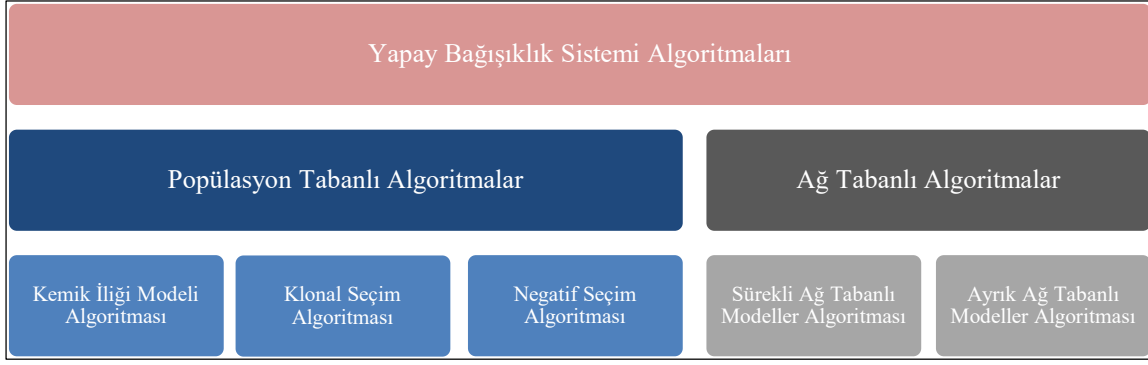
Şekil 3.8’de YBS ile bir problemi çözmek için gerekli adımlar gösterilmektedir.



Şekil 3.8. YBS'nin genel yapısı (Dasgupta, 2006)

YBS’de sistemi oluşturan parçaların gösterildiği ortama Şekil Uzayı (Shape Space); sistemi oluşturan parçaların etkileşimlerini değerlendirmek amacıyla kullanılan ölçü birimine ise Uyum Değeri (Affinity Measure) denilmektedir (Baykasoğlu, Saltabaş, Taşan ve Subulan, 2012). Şekil uzayı yaklaşımı ilk olarak 1979 yılında Perelson ve Oster tarafından geliştirilmiştir. Şekil uzayı yaklaşımı, uyum değerinin kullanılması aşamasında antikor ve antijen arasındaki etkileşimler için en çok tercih edilen modellemelerdendir (Özşen, 2008: 26).

YBS’de doğal bağışıklık sistemini oluşturan prensipler ve mekanizmalardan esinlenerek oluşturulan pek çok algoritma vardır. Bu algoritmalar Şekil 3.9’da gösterildiği gibi sınıflandırılabilir.



Şekil 3.9. YBS algoritmaları (Şahan, 2004: 40)

Literatürde YBS ile ilgili algoritmalar ve hesaplama araçlarının geliştirilmesine ilham vermek için Humoral İmmün Yanıt (Humoral Immune Response), Tehlike Teorisi (Danger Theory), Dendritik Hücre Fonksiyonları (Dendritic Cell Functions) ve Örüntü Tanıma Alıcısı Modeli (Pattern Recognition Receptor Model) olmak üzere birkaç alan bildirilmiştir. Ancak, bu yeni alanlar henüz araştırma ve geliştirme aşamasındadır (Dasgupta ve diğerleri, 2011).

Literatürde YBS ile ilgili veri madenciliği, ağ iletişimi ve bilgisayar güvenliği, optimizasyon, otomasyon ve tasarım, anomali tespiti, biyoenformatik, metin işleme, örüntü tanıma, kümeleme ve sınıflandırma gibi pek çok alanda çalışmalar yapılmıştır (Dasgupta ve diğerleri, 2011).

Klonal seçim algoritması

KSA klonal seçim prensibi'ne (KSP) dayanmaktadır. Bağışıklık sisteminin görevi vücuda dışarıdan giren, antijen olarak adlandırılan zararlı ve yabancı maddelere karşı insan vücudunu korumaktır. Bağışıklık sisteminde bir antijen tespit edildiğinde lenfosit hücrelerinden B hücreleri uyarılır ve klonlanarak çoğalıp plazma hücreleri olarak adlandırılan ve antikor salgılayan hücrelere dönüşürler. Çoğalma işlemi antijeni tanıma düzeylerine göre en iyi tanıyan hücreler, daha az tanıyan hücrelere göre daha fazla olmak üzere klonlama işlemi gerçekleşir. Klonlanan hücreler antijene olan uyumlarının artması için somatik hipermutasyon işlemine tabi olur. Uyarılmamış veya antijene uyumu az olan B hücreleri ölürler, Bu hücrelerin yerine yeni hücreler üretilir. Sonuç olarak bağışıklık

sistemi antijeni en iyi tanıyan ve yok eden hücreleri üretmiş olur. Bu işlemlerin tamamı KSP olarak adlandırılır (Babayiğit ve Güney, 2006).

KSP veya klonal seçim teorisi, bir antijenik uyarana karşı bağışıklık sistemi tarafından üretilen bağışıklık tepkisinin temel özelliklerini tarif etmek için kullanılan algoritmadır. Yalnızca antijenleri tanıyan hücrelerin çoğaldığı yani antijeni daha iyi tanıyan hücrelerin daha az tanıyan hücrelere göre seçildiği fikrini ortaya koymaktadır. Seçilen hücreler, seçici antijenlere olan uyumlarını artıran somatik hipermutasyon işlemine tabi tutulur. B hücrelerinin repertuarı, temel olarak somatik hipermutasyon ve reseptör düzenleme olarak adlandırılan iki mekanizma ile çeşitlendirilir. Somatik hipermutasyon ve reseptör düzenlemeye ek olarak, popülasyonun çeşitliliğini korumak için popülasyona yeni gelen hücrelerin bir kısmı eklenir. Algoritma örüntü tanıma ve çok modlu optimizasyon gibi karmaşık makine öğrenme görevlerini çözme yeteneğine sahiptir (De Castro ve Von Zuben, 1999).

KSA'da bağışıklık tepkisinin hızlı bir şekilde olgunlaşması için hızlı bir mutasyon birikimi gereklidir, ancak mutasyon sonucu oluşan değişikliklerin çoğu uygunluk değeri daha düşük antikörlere yol açabilir. Gerçekleşen son mutasyon sonucunda antijene benzerliği artmış olan bir hücre, bir sonraki bağışıklık yanıtları sırasında aynı oranda mutasyona uğramaya devam ederse bazen antijene olan benzerliğin azalması ile karşı karşıya kalınabilir. Bu durumu önlemek için seçim mekanizması, mutasyon işleminin düzenlenmesinde uygunluk değerine bağlı olarak çalışabilir. Daha kötü uygunluk değerine sahip hücreler daha yüksek oranda mutasyona uğrayabilir ve uygunluk değerleri iyileşmezse yok olabilirler. Bununla birlikte, daha iyi uygunluk değerine sahip antikor hücrelerinde mutasyon genellikle aşamalı olarak etkisiz hale gelebilir (De Castro ve Von Zuben, 2002).

De Castro ve Von Zuben (1999) tarafından geliştirilen KSA'ya ait algoritma adımları aşağıdaki gibidir.

1. Aday çözümleri içeren bir (P) kümesi oluştur. (P) kümesi hafızada saklanan iyi çözümlerin oluşturduğu (M) kümesini ve geriye kalan popülasyon (Pr) kümesini kapsamaktadır. ($P=M+Pr$)
2. Uygunluk değerine göre en iyi çözümlerin n adetini alarak (P_n) kümesini oluştur.

3. (P_n) kümesini oluşturan en iyi çözümleri klonlayarak (C) kümesini oluştur. Klon sayısı uygunluk değerine göre bir fonksiyon ile belirlenir.
4. (C) kümesinde yer alan bireylere uygunluk değerine göre somatik hipermutasyon uygulayarak (C^*) kümesini oluştur.
5. Geliştirilmiş bireyleri (C^*) kümesinden seçerek hafıza setini yani (M) kümesini oluştur. (P) kümesinin bazı üyeleri (C^*) kümesinin diğer geliştirilmiş üyeleri ile yer değiştirebilir.
6. Çeşitliliği korumak için popülasyonun d adet en düşük uygunluk değerine sahip bireyini silip yerlerine yeni çözümler oluştur. Adım 1' e dön.

De Castro ve Von Zuben (2002) tarafından CLONALG adlı KSA algoritması, ilk olarak makine öğrenmesi ve örüntü tanıma görevlerini yerine getirmek için önerilmiş ve daha sonra optimizasyon problemlerini çözmek için uyarlanmıştır. Optimizasyon problemlerinde kullanılan CLONALG isimli KSA algoritmasının adımları aşağıdaki gibidir.

1. Maksimize veya minimize eden bir amaç fonksiyonu vardır. Bir antikorun uygunluk değeri aynı zamanda amaç fonksiyonun değeridir. Aynı bir bellek seti ($Ab(m)$) olmaksızın her bir antikor Ab_i (Ab) popülasyonun bir üyesi olacaktır.
2. (Ab) popülasyonundaki tüm antikorların uygunluk değeri hesaplanır.
3. (Ab) popülasyonundaki en iyi uygunluk değerine sahip n tane antikor seçilerek yeni bir seçim kümesi oluşturulur ($Ab-n$).
4. Seçilen n adet antikor, uygunluk değerleri ile orantılı olarak klonlanır (çoğaltılır) ve (C) kümesi oluşturulur. Daha iyi uygunluk değerine sahip antikorlar daha kötü uygunluk değerine sahip antikora göre daha fazla sayıda klonlanır.
5. (C) kümesinde yer alan antikorlar uygunluk değeriyle ters orantılı olacak şekilde somatik hipermutasyon işlemine tabi olur ve (C^*) kümesi elde edilir. Uygunluk değeri ne kadar iyi olursa, mutasyon oranı o kadar küçük olur.
6. (C^*) kümesinde yer alan antikorların uygunluk değeri hesaplanır.
7. (Ab) kümesine eklenmek üzere (C^*) kümesinden en iyi uygunluk değerine sahip n adet antikor seçilir. Seçilen antikorlar (Ab) kümesindeki mevcut antikordardan daha iyi uygunluk değerine sahipse ilgili antikorlar ile yer değiştirilir.
8. (Ab) kümesinde yer alan en kötü uygunluk değerine sahip d tane antikor yeni oluşturulan antikorlar ile yer değiştirilir. Adım 1' e dönlür.

KSA'da çözülmesi istenen problem antijen olarak kabul edilirken; problem için geliştirilen muhtemel çözümler antikorları oluşturmaktadır. Tüm çözümlerin bir araya toplanması ile hafıza hücreleri meydana gelmektedir. Hafıza hücreleri, popülasyon olarak ta ifade edilebilir. Problemin çözümünde optimum sonuca ulaşmak için hafıza hücrelerini oluşturan antikorlar klonlama ve somatik hipermutasyon işlemine tabi tutulur. Klonlama ve somatik hipermutasyon ile mevcut antikordardan daha iyi çözümler elde etmek amaçlanmaktadır. Klonlama işleminde antijene uyumu yüksek olan yani en iyi çözüme daha yakın olan antikorlar kopyalanır. Kopyalan bu antikorlar somatik hipermutasyon işlemine tabi tutularak farklı ve daha iyi çözümler elde edilmeye çalışılır. Antijene uyumu yüksek olan antikorlar yeni nesil hafıza hücrelerini oluşturmak üzere ayrılırken antijene uyumu kötü olan antikorlar yok edilmektedir. Tüm bu işlemlerin sonunda yeni nesil hafıza hücreleri meydana gelmektedir. Uygunluk değeri GA'daki gibi bir amaç fonksiyonu veya uygunluk fonksiyonu kullanılarak hesaplanabilir. YBS'de sistemi oluşturan elemanların gösteriminde şekil uzayı yaklaşımı kullanıldığında uygunluk değerinin hesaplanmasında Oklid, Manhattan ve Hamming uzaklık ölçüleri kullanılabilir.

Şekil uzayı modeli, antijenler (Ag) ve antikorlar (Ab) arasındaki etkileşimleri nicel olarak tanımlamayı amaçlamaktadır (Ag-Ab). Ag-Ab temsiliinde (ikili veya gerçek değerli), bu moleküller arasındaki etkileşimin derecesini hesaplamak için kullanılacak bir uzaklık ölçüsü vardır (De Castro ve Von Zuben, 2002).

Bir antijen ve bir antikor arasındaki benzerlik, iki dize (veya vektörler) arasındaki uzaklık ile ilgilidir. Antijen ve antikoru gerçek değerli koordinatlar simgeliyorsa Öklid veya Manhattan uzaklık ölçüleri kullanılabilir. Öklid uzaklık ölçüsünde antikoru koordinatları $\langle ab_1, ab_2, \dots, ab_L \rangle$ şeklinde; antijenin koordinatları $\langle ag_1, ag_2, \dots, ag_L \rangle$ şeklinde belirtilirse, aralarındaki mesafe Eş. 3.1' deki gibi bulunur. Eş. 3.2 ise Manhattan uzaklık ölçüsü hesaplamasını göstermektedir (De Castro ve Von Zuben, 1999).

$$D = \sqrt{\sum_{i=1}^L (ab_i - ag_i)^2} \quad (3.1)$$

$$D = \sqrt{\sum_{i=1}^L |ab_i - ag_i|} \quad (3.2)$$

Literatürde yapılan çalışmalara bakıldığında uzaklık ölçüsü olarak genellikle Öklid uzaklık ölçüsü kullanılmıştır. Öklid uzaklık ölçüsüne başka bir alternatif, antijenlerin ve antikörlerin sembol dizileri (k büyüklüğünde bir alfabe üzerinde) olarak temsil edildiği Hamming şekil alanıdır. Eş. 3.3 Hamming uzaklık ölçüsünü göstermektedir (De Castro ve Von Zuben, 1999).

$$D = \sum_{i=1}^L \delta, \quad \delta = \begin{cases} 1, & \text{Eğer } ab_i \neq ag_i \\ 0, & \text{Diğer hallerde} \end{cases} \quad (3.3)$$

Klonal seçim algoritması temel kavramları

KSA'da yer alan antijen, antikör, gen, popülasyon büyüklüğü, affinite değeri, klonlama sayısı, somatik hipermutasyon, reseptör denetimi, elitizm ve kuşak sayısı kavramları aşağıda ayrıntılı olarak açıklanmıştır.

Antijen: Çözülmesi istenen problem olarak tanımlanabilir.

Antikör: YBS'de aday çözümleri temsil eder. GA'da yer alan kromozomlar gibi genlerden oluşur.

Gen: Bir antikorda bilgi taşıyan en küçük yapıdır.

Popülasyon büyüklüğü: Aday çözüm olarak gösterilen antikör sayısını ifade etmektedir.

Affinite değeri (Benzerlik değeri): Antikörlerin antijene olan benzerliklerinin ifade eder. Başka bir ifade ile antikörün problemin çözümü için ne kadar uygun olduğunu ifade etmektedir. Uygunluk değeri olarak ta ifade edilebilir.

Klonlama sayısı: Yeni popülasyonu oluşturmak için mevcut popülasyondaki antikörler benzerlik değerleri ile orantılı olarak belirli sayılarda kopyalanır. Her bir antikörün klonlama sayısı özel bir eşitlikle belirlenmektedir.

KSA'da antikörlerin klonlama sayısını belirlemek için kullanılan eşitlikler aşağıdaki gibidir (De Castro ve Von Zuben, 2002).

Seçim havuzuna alınan antikorlar uygunluk değerine göre farklı sayılarda klonlanacaksa Eş. 3.4 kullanılır. Antikorlar uygunluk değerine göre artan bir sırada sıralanır. Böylece daha iyi uygunluk değerine sahip antikorlar daha kötü uygunluk değerine sahip antikora göre daha fazla sayıda klonlanacaktır. $n=N$ olarak alınırsa tüm antikorlar klonlama işlemine tabi olacaktır. Popülasyonda ki tüm antikorlar eşit sayıda klonlanacaksa Eş. 3.5 kullanılır.

C : Klonlanan toplam antikor sayısı,

β : Klonlanma katsayısı,

n : Seçim havuzuna alınan antikor sayısı,

N : Toplam antikor sayısı,

x : Antikoru sırası ve numarası olmak üzere;

$$C = \sum_{x=1}^n \text{round} \left(\frac{\beta * N}{x} \right) \quad (3.4)$$

$$C = \sum_{x=1}^N \text{round}(\beta * N) \quad (3.5)$$

Somatik hipermutasyon: Antijene benzerliği yüksek olan antikorlar klonlandıktan sonra benzerliği daha da artırmak ve daha iyi bir çözüme ulaşmak için GA'daki gibi benzer şekilde mutasyon işlemine tabi olurlar. YBS'de GA'da yer alan çaprazlama operatörü olmadığı için somatik hipermutasyon antikor farklılığını yani popülasyonda çeşitliliği sağlamak için kullanılan en önemli operatördür.

Reseptör denetimi: Kötü çözümlerin popülasyondan silinerek yerlerine yeni aday çözümlerin getirilmesi işlemidir.

Elitizm: En iyi çözümlerin hafıza hücresi olarak saklanması yani yeni popülasyona direkt aktarılmasıdır. GA'da yer alan elitizm operatörüne benzerdir.

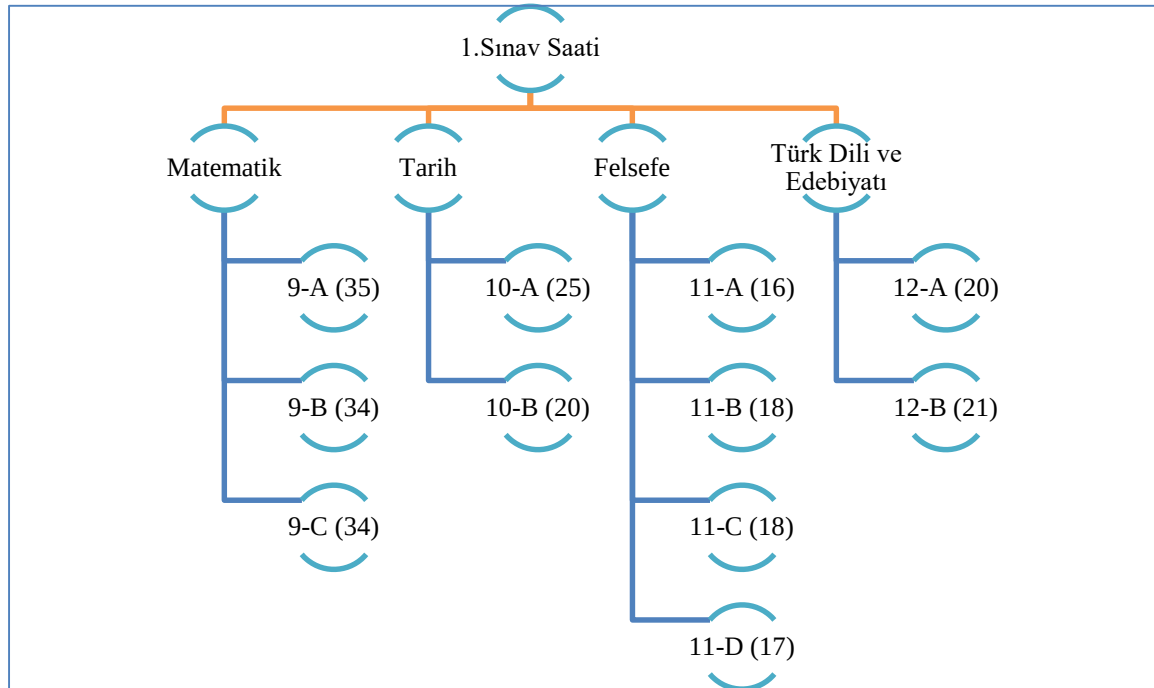
Kuşak sayısı: GA'daki iterasyon sayısına benzerdir. Geliştirilen YBS algoritmasının kaç kez çalışacağını ifade eder.

3.2. Ortak Sınav Çizelgeleme Probleminin Tanımı

Ortak sınav çizelgeleme probleminde ele alınan problem basamakları dört aşamada değerlendirilebilir:

1. Sınavları belirli veya sınırlı bir zaman dilimine atama,
2. Sınavlar için uygun sınav yerlerini belirleme,
3. Sınav yerleri için öğrenci oturma planlarının hazırlanması,
4. Sınavlar için gözetmen atamasının yapılması.

Ortak sınavlarda sınavlar farklı sınıf düzeyleri içi karma olacak şekilde yapılmaktadır. Aynı sınıf seviyesinde yer alan öğrenciler aynı sınava tabi olmakta ve bu sınavlar ortak sınav olarak yapılmaktadır. Temsili bir ortak sınav saati Şekil 3.10’da verilmiştir. Bu örnekte aynı sınav saatinde 4 farklı sınav yapılmaktadır. 9. sınıf seviyesindeki öğrenciler “Matematik”, 10. sınıf seviyesindeki öğrenciler “Tarih”, 11. sınıf seviyesindeki öğrenciler “Felsefe” ve 12. sınıf seviyesindeki öğrenciler “Türk Dili ve Edebiyatı” dersi sınavına girmektedirler.



Şekil 3.10. Temsili bir ortak sınav saati örneği

Sınav yapılacak sınıfların yapısı kurumdan kuruma ve kurum içerisinde farklılık gösterebilmektedir. Sınıflarda tek oturumlu sıralar ve çift oturumlu sıralar yer alabilmekte ve ihtiyaca göre farklı yerleşim düzeninde olabilmektedir. Ortak sınavlarda kalabalık öğrenci grupları aynı anda sınava girdiğinden sınıf kapasiteleri genellikle maksimum düzeyde kullanılmaktadır. Özellikle çift oturumlu sıralarda öğrenciler çok yakın bir oturum şekliyle sınava girmektedirler. Eğer sınıfın boyutu küçükse farklı sıralarda bile olsa öğrenciler aynı hizada yer alan sıralarda yana yana yakın bir oturum yerleşiminde sınava girebilmektedir. Sınav oturum planları hazırlanırken öğrenciler arasında istenmeyen yardımlaşmanın önüne geçmek için aynı sınıf seviyesinde yer alan öğrencilerin çaprazda, yan yana ya da önlü arkalı gelmeyecek şekilde oturum yerleşiminin yapılması gözetilmelidir. Temsili bir oturum planı örneği Şekil 3.11’de verilmiştir. Aynı sınıf seviyesinde yer alan öğrencilerin aynı sırada yan yana, seçime göre yatayda aynı hizada yer alan farklı sıralarda yan yana, dikeyde aynı hizada yer alan farklı sıralarda çaprazda çakışarak ve arka arkaya oturumları istenmeyen yerleşim şekli olarak kabul edilmektedir. Oturum planları mümkün olduğunca istenmeyen yerleşimlerin önüne geçecek şekilde hazırlanmalıdır.

<u>9-9</u>	<u>10-11</u>	<u>12-9</u>	<u>11-12</u>
<u>11-12</u>	<u>9-10</u>	<u>11-10</u>	<u>9-10</u>
<u>10-9</u>	<u>12-11</u>	<u>9-12</u>	<u>11-12</u>
<u>12-9</u>	<u>9-10</u>	<u>11-12</u>	<u>9-10</u>

Şekil 3.11. Temsili bir oturum planı örneği

Bu açıklamalar çerçevesinde ortak sınav çizelgeleme problemine ait kısıtlar aşağıdaki gibi özetlenebilir:

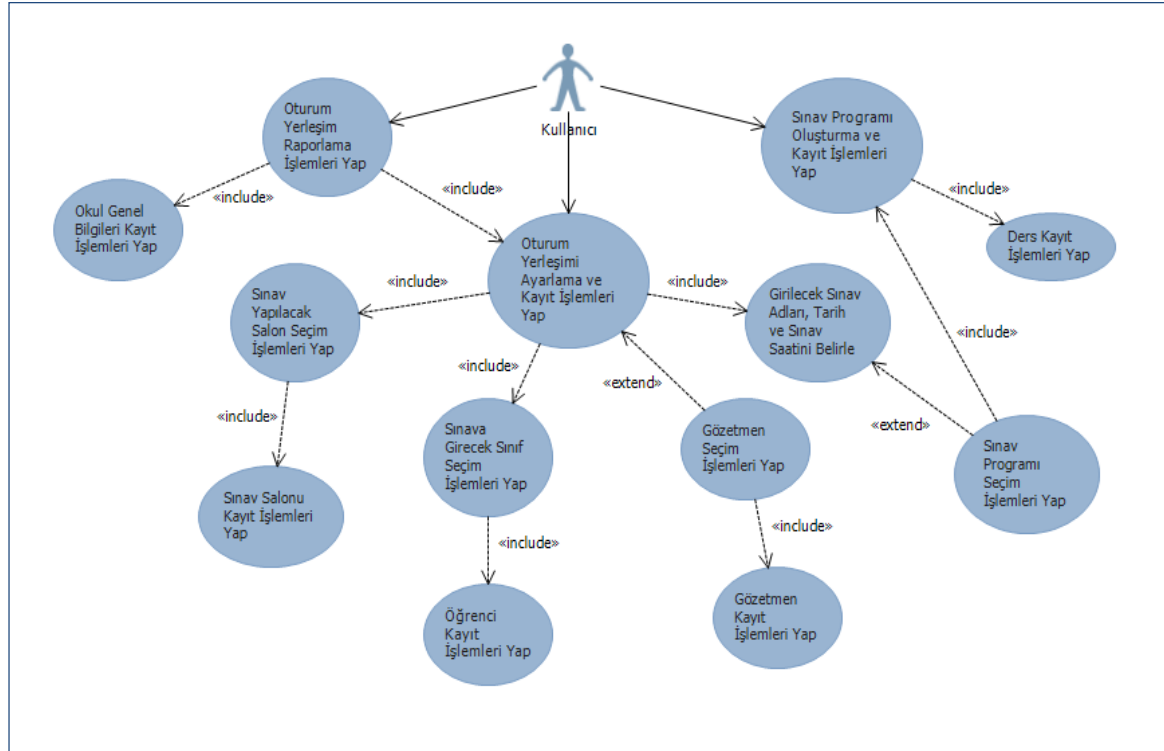
1. Sınavların belirlenen tarih aralığı ve sınav saatine göre her sınıf düzeyi için günde en fazla iki sınav olacak şekilde uygun zaman dilimine atanması,
2. Sınava girecek sınıf düzeyleri ve öğrenci sayılarına göre kapasite olarak uygun sayıda sınav yerinin seçilmesi,
3. Sınav yerlerinde her bir sınıf düzeyi için sınava girecek öğrenci sayılarının sınav yerlerinin kapasitesine göre mümkün olduğunca eşit dağılım oluşturacak şekilde belirlenmesi,
4. Aynı sınav yerinde sınava girecek aynı sınıf seviyesinde yer alan öğrencilerin çaprazda, yan yana ya da önlü arkalı gelmeyecek şekilde yerleşimlerinin yapılması gözetilerek uygun oturum planlarının hazırlanması,
5. Sınavlar için gözetmen atamasının yapılması.

3.3. Yöntem ve Geliştirilen Yazılım-GENÇİZ

Kesim 3.3.1’de GENÇİZ yazılımında kullanılan veri tabanı yapısı ve yazılımının içeriği ile kullanımı ayrıntılı olarak açıklanmıştır. Kesim 3.3.2 ve Kesim 3.3.3’de sırasıyla uygun sınav oturum planlarının hazırlanmasında GA ve YBS uygulanan çözüm yöntemlerinin detaylı açıklamalarına yer verilmiştir.

3.3.1.Ortak sınav çizelgeleme problemi için geliştirilen yazılım-GENÇİZ

GENÇİZ yazılımı, GA uygulanan oturum planı hazırlama yöntemi ile birlikte çalışacak şekilde geliştirilmiştir. Yazılımın geliştirilmesinde Visual Studio ortamında C#.NET yazılımı kullanılmıştır. Kullanılan veri tabanı Microsoft Access ile oluşturulmuştur. GENÇİZ yazılımına ait use case (kullanım senaryosu) diyagramı Şekil 3.12’de verilmiştir.



Şekil 3.12. GENÇİZ yazılımına ait use case diyagramı

GENÇİZ yazılımının ana ekranı Resim 3.1’ de verilmiştir.

GENÇİZ

Öğrenci ve Sınıf İşlemleri Sınav Salonu İşlemleri Sınav Programı İşlemleri Gözetmen Kayıt İşlemleri Okul Genel Bilgileri Kayıt İşlemleri

☒ Sınav Bilgilerini Sınav Programından Al

Sınav Girecek Öğrenci Sayıları: 9.Sınıf: 100 10.Sınıf: 61 11.Sınıf: 94 12.Sınıf: 91

☐ Aynı sıralarda da yan yana kısıt ekle

Toplam öğrenci sayısı: 346

Sınav yer oturum sayısı: 348

Olası Dağılım Göster Oturumu Ayarla

Dağılımı Kaydet

Sınavlara Göre Dağılım Ceza Puanları

303: ceza Puanı: 2
304: ceza Puanı: 2
305: ceza Puanı: 2
306: ceza Puanı: 2
307: ceza Puanı: 2
308: ceza Puanı: 2
309: ceza Puanı: 2
401: ceza Puanı: 2
402: ceza Puanı: 2
404: ceza Puanı: 2
Salon1: ceza Puanı: 8

Seçili Sınav Salonları

303: Tekli Sıralar (28)
304: Tekli Sıralar (28)
305: Tekli Sıralar (28)
306: Tekli Sıralar (28)
307: Tekli Sıralar (28)
308: Tekli Sıralar (28)
309: Çiftli Sıralar (28)
401: Çiftli Sıralar (28)
402: Tekli Sıralar (28)
403: Tekli Sıralar (28)
Salon1: Tekli Sıralar (36)
404: Çiftli Sıralar (32)

Sınav Girecek Sınıflar

9B Mevcut: 32
9C Mevcut: 34
10A Mevcut: 31
10B Mevcut: 30
11A Mevcut: 33
11B Mevcut: 30
11C Mevcut: 31
12A Mevcut: 29
12B Mevcut: 31
12C Mevcut: 31
9A Mevcut: 34

Gözetmenler ☐ Tümünü Seç

☐ Sevil Vatansever
☐ Veli Vatansever
☐ Ömer Faruk Bakla
☐ Nihal Altınışık
☐ Ahmet İlcan
☐ Dilşad Nur Altınışık
☐ Ayşe Kaya
☐ Gülseren Al
☐ Meral Yılmaz
☐ Selver Pek
☐ Salih Canel
☐ Onur Guçlu
☐ Mert Şen

Toplam Oturum Yeri: 348 Toplam Öğrenci Sayısı: 346 Seçilebilir Gözetmen Sayısı: 13 Seçilen Gözetmen Sayısı: 0

Gözetmen Seçimlerini Uygula

Kayıtlı Oturumlar: Yeni Oturum Sınav Yeri: 101

Sınav Tarihi: 27 Mayıs 2019, Pazartesi

Sınav Saati: 1

Seçili Oturum Programını Sil

Tüm Oturum Kayıtlarını Sil

	1	2	3	4	5	6
1	12 (Yabancı Dil)	9 (Türk Dili ve Edebiyatı)	11 (Tarih)	12 (Yabancı Dil)	11 (Tarih)	9 (Türk Dili ve Edebiyatı)
2	11 (Tarih)	10 (Türk Dili ve Edebiyatı)	10 (Türk Dili ve Edebiyatı)	9 (Türk Dili ve Edebiyatı)	12 (Yabancı Dil)	9 (Türk Dili ve Edebiyatı)
3	12 (Yabancı Dil)	9 (Türk Dili ve Edebiyatı)	11 (Tarih)	12 (Yabancı Dil)	11 (Tarih)	10 (Türk Dili ve Edebiyatı)
4	10 (Türk Dili ve Edebiyatı)	11 (Tarih)	10 (Türk Dili ve Edebiyatı)	9 (Türk Dili ve Edebiyatı)	9 (Türk Dili ve Edebiyatı)	12 (Yabancı Dil)
5	9 (Türk Dili ve Edebiyatı)	12 (Yabancı Dil)	11 (Tarih)	12 (Yabancı Dil)		

Oturum Planının Raporunu Al

Oturum Öğrenci Listesi Raporu Al

Resim 3.1. GENÇİZ yazılımı ana ekranı

Yazılım ana ekranından sınav oturumu hazırlayabilmek için yapılması gereken tüm işlemlere hızlı ve kolay bir şekilde erişim söz konusudur. Daha önceden kaydedilmiş sınav programlarından yapılan sınav programı ve bu doğrultuda sınav tarihi ve saati seçimleri ile oturum planı yapılacak sınav seçimleri gerçekleştirilmektedir. “Sınav Bilgilerini Sınav Programından Al” kontrol kutusunun (checkbox) seçimi kaldırıldığında; oluşturulan sınav programlarından bağımsız olarak sınav bilgileri anlık olarak da girilebilmekte böylece hızlı bir şekilde sınav oturum planları hazırlanabilmektedir.

Yapılan sınav seçimi ya da girilen sınav bilgilerinden sonra “Sınav Seçimlerine Göre Sınava Girecek Sınıfları Otomatik Seç” düğmesi (button) ile yapılacak sınavlara uygun sınıf seçimleri programa otomatik olarak yaptırılabilir. “Sınava Girecek Sınıflara Göre Sınav Salonlarını Otomatik Seç” düğmesi ile de yapılan sınıf seçimlerine göre sınav salonu seçimleri asgari oturum yerini sağlayacak şekilde otomatik olarak seçtirilebilmektedir.

Seçili sınav salonları ve sınava girecek sınıflar ana ekranda listelenmekte; istenirse özel seçimlerin yapılabilmesi için “Sınav Yeri Seç” düğmesi ile “Sınav Salonu Seç” ekranına, “Sınava Girecek Sınıfları Seç” düğmesi ile de “Sınava Girecek Sınıfların Seçimi” ekranlarına hızlı bir şekilde ulaşılabilir. Yapılan seçimlere göre sınavlara girecek öğrenci sayıları ana ekranda gösterilmektedir. “Oturumu Ayarla” düğmesi ile yapılan seçimlere uygun olarak oturum planları hazırlanmakta ve “Yeni Oturum” başlığı altında sınav salonları bazında yerleşimler görüntülenebilmektedir. Sınavlarda görev alması istenen gözetmen öğretmen seçimleri de ana ekrandan yapılabilir. Oturum istenirse kullanıcı tarafından belirlenen bir isim ile gözetmen seçimli ya da gözetmen seçimsiz olarak kaydedilebilmektedir.

Daha önceden oluşturulmuş ve kaydedilmiş oturumların sınav salonu bazında yerleşimleri yine ana ekranda görüntülenebilmektedir. Kayıtlı oturumların silme işlemi de ana ekrandan gerçekleştirilebilmektedir. Ana ekranda yer alan “Oturum Planının Raporunu Al” düğmesi ile kayıtlı oturumlardan yapılan seçime göre öğrencilerin sınav salonları bazında sıralara yerleşimini gösteren oturum planı raporu alınabilmektedir. Oturum planı örnek raporu Resim 3.2’de verilmiştir.

2018-2019 Yılı Mesleki ve Teknik Anadolu Lisesi 2. Dönem Sınav Oturum Yerleşimi						
Tarih: 27.05.2019 Sınav Saati: 1.Oturum						
	9.Sınıfların Sınavı	10.Sınıfların Sınavı	11.Sınıfların Sınavı	12.Sınıfların Sınavı	Gözetmenler	
Ders Adı:	Türk Dili ve Edebiyatı	Türk Dili ve Edebiyatı	Aseptik Uygulamaları	T.C. İnkılap Tarihi	Toplam	Seval Vatansever
Sınava Girecek Öğrenci Sayısı:	9	5	9	5	28	Ahmet İlcan



Sınıf Kodu:11B	1 Öğrenci Ad Soyad1 11B 184 Tarih	Çiftli Sıra	2 Öğrenci Ad Soyad2 9B 5 Türk Dili ve Edebiyatı	11 Öğrenci Ad Soyad11 9C 255 Türk Dili ve Edebiyatı	Çiftli Sıra	12 Öğrenci Ad Soyad12 11A 7 Tarih	21 Öğrenci Ad Soyad21 9B 212 Türk Dili ve Edebiyatı	Çiftli Sıra	22 Öğrenci Ad Soyad22 11A 40 Tarih
	3 Öğrenci Ad Soyad3 10A 471 Türk Dili ve Edebiyatı	Çiftli Sıra	4 Öğrenci Ad Soyad4 12A 386 Yabancı Dil	13 Öğrenci Ad Soyad13 10B 29 Türk Dili ve Edebiyatı	Çiftli Sıra	14 Öğrenci Ad Soyad14 12B 318 Yabancı Dil	23 Öğrenci Ad Soyad23 9C 166 Türk Dili ve Edebiyatı	Çiftli Sıra	24 Öğrenci Ad Soyad24 12A 387 Yabancı Dil
	5 Öğrenci Ad Soyad5 11C 75 Tarih	Çiftli Sıra	6 Öğrenci Ad Soyad6 Türk Dili ve Edebiyatı	15 Öğrenci Ad Soyad15 11C 407 Tarih	Çiftli Sıra	16 Öğrenci Ad Soyad16 9C 224 Türk Dili ve Edebiyatı	25 Öğrenci Ad Soyad25 11C 91 Tarih	Çiftli Sıra	26 Öğrenci Ad Soyad26 10A 472 Türk Dili ve Edebiyatı
	7 Öğrenci Ad Soyad7 12B 416 Yabancı Dil	Çiftli Sıra	8 Öğrenci Ad Soyad8 10A 481 Türk Dili ve Edebiyatı	17 Öğrenci Ad Soyad17 10B 611 Türk Dili ve Edebiyatı	Çiftli Sıra	18 Öğrenci Ad Soyad18 12B 322 Yabancı Dil	27 Öğrenci Ad Soyad27 11C 290 Tarih	Çiftli Sıra	28 Öğrenci Ad Soyad28 9A 213 Türk Dili ve Edebiyatı
	9 Öğrenci Ad Soyad9 9C 44 Türk Dili ve Edebiyatı	Çiftli Sıra	10 Öğrenci Ad Soyad10 11B 836 Tarih	19 Öğrenci Ad Soyad19 11C 433 Tarih	Çiftli Sıra	20 Öğrenci Ad Soyad20 9A 14 Türk Dili ve Edebiyatı		Çiftli Sıra	

Resim 3.2. Oturum planı raporu

Yine ana ekranda yer alan “Oturum Öğrenci Listesi Raporunu Al” düğmesi ile kayıtlı oturumlardan yapılan seçime göre sınıf bazında öğrencilerin sınav salonlarına yerleşimini gösteren öğrenci listesi raporu alınabilmektedir. Raporlar excel, pdf ve word formatında dosya olarak kaydedilebilmekte ya da çıktısı alınabilmektedir. Oturum öğrenci listesi örnek raporu Resim 3.3’ de verilmiştir.

**2018-2019 Yılı Mesleki ve Teknik Anadolu Lisesi 2. Dönem
Sınav Yerleşimi Sınıf Listesi**

Tarih: 27.05.2019
Sınav Saati: 1.Oturum

Sınıf	Şube	Sıra No	Numara	Ad	Soyad	Sınıf Kodu	Yer No
9	A	1	2	Öğrenci Ad1	Öğrenci Soyad1	9B	25
		2	4	Öğrenci Ad2	Öğrenci Soyad2	11C	11
		3	14	Öğrenci Ad3	Öğrenci Soyad3	11B	20
		4	15	Öğrenci Ad4	Öğrenci Soyad4	10A	8
		5	24	Öğrenci Ad5	Öğrenci Soyad5	10A	16
		6	111	Öğrenci Ad6	Öğrenci Soyad6	9A	9
		7	156	Öğrenci Ad7	Öğrenci Soyad7	Salon1	27
		8	159	Öğrenci Ad8	Öğrenci Soyad8	Salon1	25
		9	170	Öğrenci Ad9	Öğrenci Soyad9	9A	21
		10	181	Öğrenci Ad10	Öğrenci Soyad10	12B	18
		11	185	Öğrenci Ad11	Öğrenci Soyad11	11C	20
		12	190	Öğrenci Ad12	Öğrenci Soyad12	12B	1
		13	201	Öğrenci Ad13	Öğrenci Soyad13	11C	5
		14	202	Öğrenci Ad14	Öğrenci Soyad14	11A	8
		15	207	Öğrenci Ad15	Öğrenci Soyad15	11C	24
		16	208	Öğrenci Ad16	Öğrenci Soyad16	11A	14
		17	213	Öğrenci Ad17	Öğrenci Soyad17	11B	28
		18	214	Öğrenci Ad18	Öğrenci Soyad18	12B	13
		19	220	Öğrenci Ad19	Öğrenci Soyad19	9C	2
		20	221	Öğrenci Ad20	Öğrenci Soyad20	10A	6
		21	226	Öğrenci Ad21	Öğrenci Soyad21	12A	26
		22	227	Öğrenci Ad22	Öğrenci Soyad22	10A	24
		23	232	Öğrenci Ad23	Öğrenci Soyad23	11A	12
		24	233	Öğrenci Ad24	Öğrenci Soyad24	9A	15
		25	238	Öğrenci Ad25	Öğrenci Soyad25	9A	23
		26	239	Öğrenci Ad26	Öğrenci Soyad26	11A	26
		27	244	Öğrenci Ad27	Öğrenci Soyad27	9B	21
		28	245	Öğrenci Ad28	Öğrenci Soyad28	12A	11
		29	251	Öğrenci Ad29	Öğrenci Soyad29	12A	8
		30	253	Öğrenci Ad30	Öğrenci Soyad30	9C	26
		31	262	Öğrenci Ad31	Öğrenci Soyad31	11A	19
		32	263	Öğrenci Ad32	Öğrenci Soyad32	12B	28
		33	275	Öğrenci Ad33	Öğrenci Soyad33	9B	15
		34	294	Öğrenci Ad34	Öğrenci Soyad34	11B	6

Okul Müdürü

Resim 3.3. Oturum öğrenci listesi raporu

Ana ekran üst kısmında yer alan menü öğeleri ve bu menüler ile gerçekleştirilen işlemler aşağıdaki gibidir.

Öğrenci ve sınıf işlemleri

Bu menü altında “Öğrenci Kayıt İşlemleri” ve “Sınava Girecek Sınıfları Belirle” alt menüleri yer almaktadır.

Öğrenci kayıt işlemleri

Resim 3.4’de gösterilen “Öğrenci Kayıt İşlemleri” ekranından excel dosya formatında; belirlenen sütun ve sayfa yapısına uygun dosyalardan öğrenci verileri alınabilmekte ve veri tabanına aktarılabilir. Aynı zamanda “Öğrenci Kayıtlarını Düzenle” kontrol kutusu işaretlenerek veri tabanında yer alan öğrenci verileri üzerinde “Kaydet/Güncelle”, “Kaydı Sil” ve “Tüm Kayıtları Sil” işlemleri gerçekleştirilebilmektedir. Aynı zamanda “Yeni Kayıt” düğmesi ile öğrenci verileri tek tek de eklenebilmektedir.

Öğrenci Kayıt İşlemleri

Veri Tabanında Kayıtlı Öğrenciler

sınıf	sube	ogr_no	ogr_adi	oç
9	A	2		
9	A	4		
9	A	14		
9	A	15		
9	A	24		
9	A	111		
9	A	156		
9	A	159		
9	A	170		
9	A	181		
9	A	185		

sınıf	sube	ogr_no	ogr_adi	ogr_soyadi
9	B	34		
9	B	160		
9	B	167		
9	B	173		
9	B	178		
9	B	186		
9	B	192		
9	B	195		
9	B	203		
9	B	209		
9	B	212		

☐ Öğrenci Kayıtlarını Düzenle

Excel'den Öğrencileri Al Veri Tabanına Aktar

Dosya Alımında Dikkat Edilmesi Gereken Hususlar:

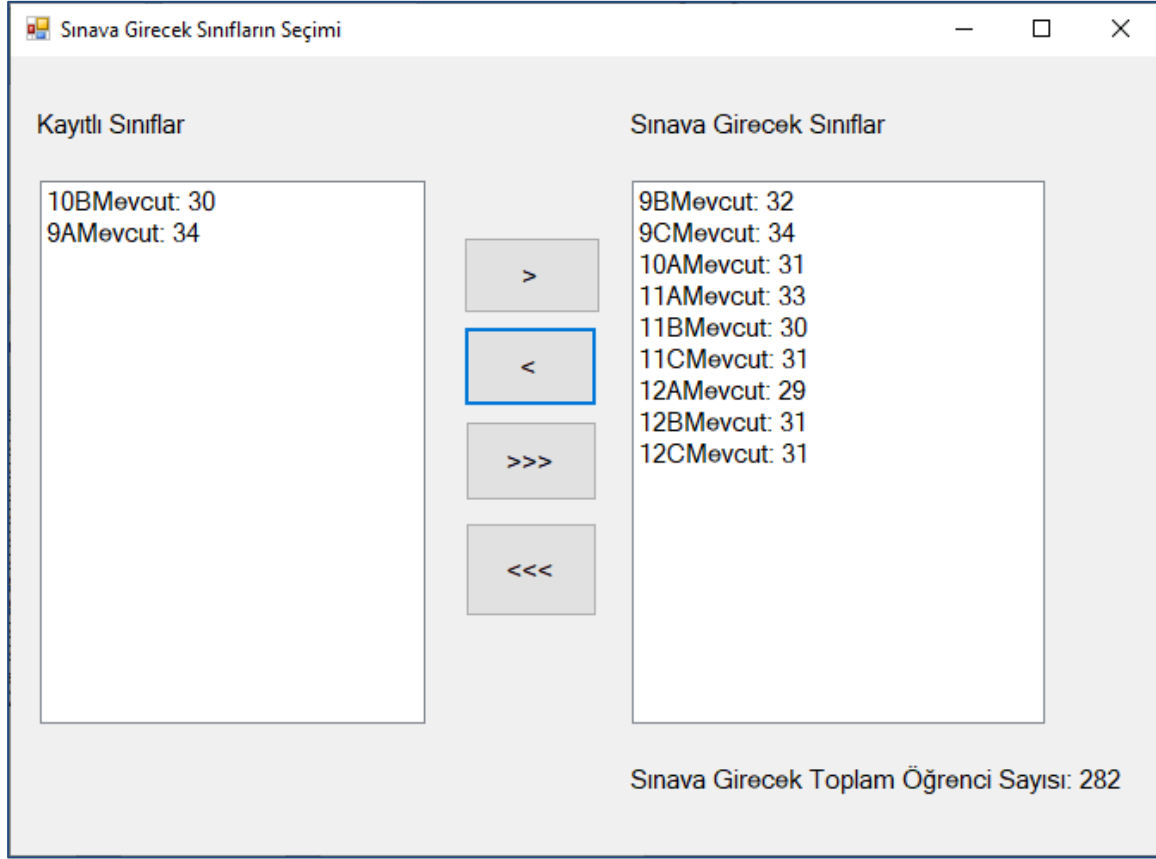
1. Excel dosyasında veriler Sayfa 1 isimli çalışma sayfasında yer almalıdır.
2. Sayfadaki veri yerleşimi yandaki veri izgarasında görünen sütun yapılan ile birebir aynı yapıda olmalıdır.

Resim 3.4. Öğrenci kayıt işlemleri ekranı

Sınava girecek sınıfları belirle

Resim 3.5’de gösterilen “Sınava Girecek Sınıfların Seçimi” ekranından hazırlanan sınav oturumunda sınava girecek olan sınıfların seçimi yapılmaktadır. Sınıflar “Öğrenci Kayıt

İşlemi” ekranı üzerinden daha önce girilmiş öğrenci bilgileri verilerine göre oluşturulmaktadır.



Resim 3.5. Sınava girecek sınıfların seçimi ekranı

Sınav salonu işlemleri

Bu menü altında, sınav salonu işlemlerini yapabilmek için “Sınav Salonu Kayıt” ve “Sınav Salonu Seç” alt menüleri yer almaktadır.

Sınav salonu kayıt

Resim 3.6’da gösterilen “Sınav Salonu Kayıt” ekranından “Sıra Şekli” açılır listesinden (comboBox) ve “Yerleşim Parametreleri” sayısal açılır listelerinden (numericupdown) yapılan seçimlere göre yeni sınav salonları oluşturulabilmektedir. “Çift Oturumlu Sıralar” ve “Tek Oturumlu Sıralar” olmak üzere iki çeşit sıra şekli seçeneği vardır. Sıraların satır ve sütunda yerleşim parametreleri girildiğinde sınıf oturum düzeni temsili olarak ekranda gösterilmektedir. İstenirse son satır sıra yerleşiminde sıra eksiltmesi yapılarak daha özel

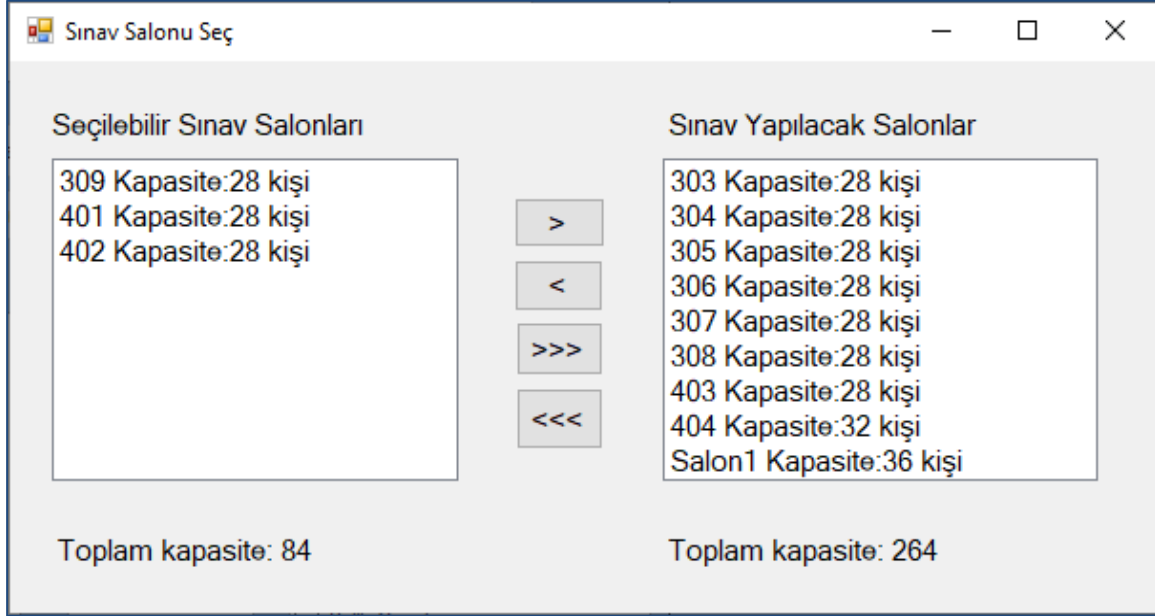
sınıf oturum yerleşimleri de oluşturulabilmektedir. “Ekle” düğmesi ile oluşturulan salonlar veri tabanına kaydedilmekte; “Güncelle” düğmesi ile tablo üzerinde yapılan değişiklikler veri tabanına aktarılmakta; “Vazgeç” düğmesi ile tablo üzerinde yapılan değişiklikler iptal edilmekte ve “Sil” düğmesi ile seçilen sınav salonu kayıtları silinebilmektedir.

Sınıf Kodu	Yatayda Sıra Sayısı	Dikeyde Sıra Sayısı	Sıra Şekli	Toplam Oturum Yeri
303	4	7	tek	28
304	4	7	tek	28
305	4	7	tek	28
306	5	6	tek	28
307	4	7	tek	28
308	4	7	tek	28
309	5	3	çift	28
401	5	3	çift	28
402	4	7	tek	28
403	4	7	tek	28
Salon 1	6	6	tek	36

Resim 3.6. Sınav salonu kayıt ekranı

Sınav salonu seç

Resim 3.7’de gösterilen “Sınav Salonu Seç” ekranından hazırlanan sınav oturumunda sınav salonu olarak kullanılacak yerlerin seçimi yapılmaktadır.



Resim 3.7. Sınav salonu seç ekranı

Sınav programı işlemleri

Bu menü altında sınav programı işlemlerini gerçekleştirmek için “Sınav Programı Hazırla” ve “Ders Kayıt İşlemleri” alt menüleri yer almaktadır.

Sınav programı hazırla

Resim 3.8’de gösterilen “Sınav Dağıtımı” ekranından seviye, ders ve tarih seçimlerine göre sınav programı hazırlanabilmekte ve hazırlanan program istenirse veri tabanına kaydedilebilmektedir. Var olan sınav programları görüntülenebilmekte ve seçime göre programlar silinebilmektedir.

Sınav Dağıtımı

Seviye Seç: 12

Ders Seç: ☐ Tümünü Seç

☐ Biyoloji
☐ Coğrafya
☐ Din Kültürü ve Ahlak Bilgisi
☒ Felsefe
☐ Fizik
☒ Kimya
☒ Matematik
☐ Mesleki Gelişim
☐ Peygamberimizin Hayatı
☐ Sağlık Bilgisi ve Trafik Kültürü
☐ Tarih
☐ Türk Dili ve Edebiyatı

Kayıtlı Sınav Programları

Sınav Yapılacak Dersler ve Sınıf Sevdeleri

10.Sınıf Fizik Dersi Sınavı
10.Sınıf Kimya Dersi Sınavı
10.Sınıf Matematik Dersi Sınavı
10.Sınıf Mesleki Gelişim Dersi Sınavı
11.Sınıf Peygamberimizin Hayatı Dersi Sınavı
11.Sınıf Sağlık Bilgisi ve Trafik Kültürü Dersi Sınavı
11.Sınıf Tarih Dersi Sınavı
12.Sınıf Felsefe Dersi Sınavı
12.Sınıf Kimya Dersi Sınavı
12.Sınıf Matematik Dersi Sınavı
9.Sınıf Biyoloji Dersi Sınavı
9.Sınıf Coğrafya Dersi Sınavı
9.Sınıf Din Kültürü ve Ahlak Bilgisi Dersi Sınavı
9.Sınıf Felsefe Dersi Sınavı

Sınavların Başlangıç Tarihi: 27 Mayıs 2019 Pazartesi
Sınavların Bitiş Tarihi: 30 Mayıs 2019 Perşembe

Sınav Günleri: ☒ Tümünü Seç ☐ Haftasonunu seçimlerde göster

☒ 27 Mayıs 2019, Pazartesi
☒ 28 Mayıs 2019, Salı
☒ 29 Mayıs 2019, Çarşamba
☒ 30 Mayıs 2019, Perşembe

	27 Mayıs 2019, Pazartesi	28 Mayıs 2019, Salı	29 Mayıs 2019, Çarşamba	30 Mayıs 2019, Perşembe
► (9) 1. Sınav	9.Sınıf Biyoloji	9.Sınıf Felsefe	9.Sınıf Din Kültürü ve Ahlak Bilgisi	9.Sınıf Coğrafya
(10) 1. Sınav	10.Sınıf Kimya	10.Sınıf Matematik	10.Sınıf Mesleki Gelişim	10.Sınıf Fizik
(11) 1. Sınav	11.Sınıf Sağlık Bilgisi ve Trafik Kültürü	11.Sınıf Peygamberimizin Hayatı	11.Sınıf Tarih	
(12) 1. Sınav	12.Sınıf Felsefe	12.Sınıf Kimya	12.Sınıf Matematik	
(9) 2. Sınav	9.Sınıf Biyoloji			
(10) 2. Sınav				
(11) 2. Sınav				
* (12) 2. Sınav				

Resim 3.8. Sınav dağıtımı ekranı

Ders kayıt işlemleri

Resim 3.9’da gösterilen “Ders Kayıt İşlemleri” ekranından veri tabanına ders kaydı yapılabilmekte, var olan ders kayıtları üzerinde silme ve güncelleme işlemleri gerçekleştirilebilmektedir.

DersAdi
Türk Dili ve Edebiyatı
Matematik
Fizik
Kimya
Biyoloji
Din Kültürü ve Ahlak Bilgisi
Mesleki Gelişim
Tarih
Coğrafya
Felsefe
Yabancı Dil
Sağlık Bilgisi ve Trafik Kültürü
Peygamberimizin Hayatı

Ders Adı: Türk Dili ve Edebiyatı

Yeni Ders Kaydı

Vazgeç

Kaydet/Güncelle

Ders Kaydını Sil

Tüm Kayıtları Sil

Resim 3.9. Ders kayıt işlemleri ekranı

Gözetmen Kayıt İşlemleri

Resim 3.10’da gösterilen “Gözetmen Kayıt İşlemleri” ekranından sınav oturumlarında gözetmen olarak görevlendirilecek görevlilerin kayıt, güncelleme ve silme işlemleri gerçekleştirilmektedir.

AdSoyad
Seval Vatansever
Veli Vatansever
Nihal Altınışık
Ahmet İlcan
Dilşad Nur Altınışık

Ad Soyad Seval Vatansever

Yeni Kayıt

Kaydet/Güncelle

Vazgeç

Sil

Resim 3.10. Gözetmen kayıt işlemleri ekranı

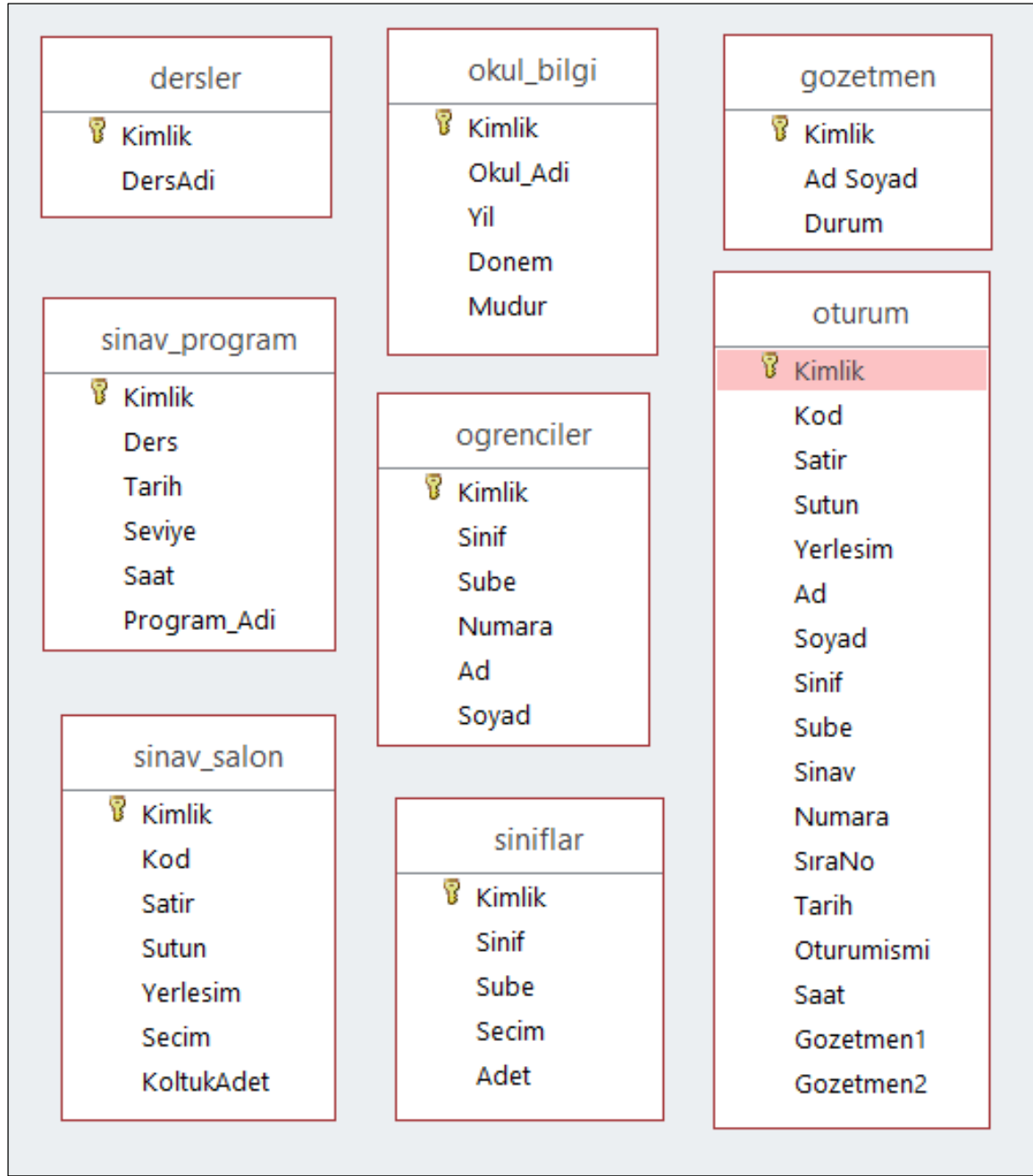
Okul Genel Bilgileri Kayıt İşlemleri

Resim 3.11’de gösterilen “Okul Genel Bilgileri Kayıt” ekranından sınav oturum planı raporlarında yer alacak olan okul genel bilgilerinin kayıt ve güncelleme işlemleri yapılmaktadır.

Resim 3.11. Okul genel bilgileri kayıt ekranı

Ortak sınav çizelgeleme problemi için geliştirilen yazılımda kullanılan veri tabanı yapısı

Yazılımda kullanılan veri tabanını oluşturan tablolar ve yapıları Resim 3.12’de verilmiştir. Veri tabanında yer alan tabloların yapıları ve kullanımları aşağıda açıklanmıştır.



Resim 3.12. Veri tabanında yer alan tablolar

“dersler” tablosunda ortak sınav olarak yapılacak tüm derslere ait ders adı bilgileri tutulmaktadır. Sınav programı oluşturulurken sınav yapılacak ders bilgilerinin seçimi için gerekli ders adı verileri program ara yüzüne (“Sınav Dağıtımı” ekranı) bu tablodan aktarılmaktadır.

“sinav_program” tablosunda oluşturulan sınav programlarına ait veriler tutulmaktadır. Program ana ekranında; oturum planının hazırlanması aşamasında sınav programı seçimi için

gerekli veriler bu tablodan aktarılmaktadır. “Sınav Bilgilerini Sınav Programından Al” seçeneği seçilmişse oluşturulan oturum planlarına ait verilerin “oturum” tablosuna kaydedilmesi aşamasında sınav programına ait veriler bu tablodan alınmaktadır.

“sinav_salon” tablosunda ortak sınavların gerçekleştirileceği sınav salonlarına ait veriler tutulmaktadır. Program ana ekranında; oturum planının hazırlanması aşamasında sınav yerlerinin seçimi için gerekli veriler bu tablodan aktarılmaktadır. Oluşturulan oturum planlarına ait verilerin “oturum” tablosuna kaydedilmesi aşamasında sınav salonlarına ait veriler bu tablodan alınmaktadır.

“gozetmen” tablosunda sınavlarda gözetmen olarak atanacak öğretmenlere ait veriler tutulmaktadır. Program ana ekranında; oturum planının hazırlanması aşamasında gözetmenlerin seçimi için gerekli veriler bu tablodan aktarılmaktadır. Oluşturulan oturum planlarına ait verilerin “oturum” tablosuna kaydedilmesi aşamasında görevli sınav gözetmenlerine ait veriler bu tablodan alınmaktadır.

“okul_bilgi” tablosunda okula ait genel bilgiler tutulmaktadır. Bu bilgiler oturum planlarının raporlama aşamasında kullanılmaktadır. Raporlarda eğitim-öğretim yılı, öğretim dönemi, okul adı ve müdür adı verileri bu tablodan alınmaktadır.

“ogrenciler” tablosunda öğrencilere ait veriler tutulmaktadır. Oluşturulan oturum planlarına ait verilerin “oturum” tablosuna kaydedilmesi aşamasında öğrencilere ait veriler bu tablodan alınmaktadır.

“sınıflar” tablosunda mevcut sınıflara ait veriler tutulmaktadır. Bu tablo “ogrenciler” tablosunda yer alan bilgiler doğrultusunda program tarafından oluşturulmaktadır. Program ana ekranında; oturum planının hazırlanması aşamasında sınava girecek sınıfların seçimi için gerekli veriler bu tablodan aktarılmaktadır. Oluşturulan oturum planlarına ait verilerin “oturum” tablosuna kaydedilmesi aşamasında hangi öğrencilere ait verilerin alınacağı bu tabloya göre belirlenmektedir.

“oturum” tablosu “sinav_program”, “sinav_salon”, “gozetmen”, “ogrenciler” ve “sınıflar” tablosundan gerekli veriler alınarak program tarafından oluşturulmaktadır. Bu tabloda program ana ekranından yapılan seçimlerle oluşturulan ve kaydedilen oturum planlarına ait

veriler tutulmaktadır. Oturum planlarının raporlama aşamasında bu tablodaki veriler kullanılmaktadır.

3.3.2. Genetik algoritmanın oturum planı hazırlama problemine uygulanması

GA'nın sözde kodları (pseudo code) aşağıda verilmiştir.

1. *Kromozomların gen değerlerini, 9-12 arasında rastgele üreterek pb adet SP ilk popülasyonunu oluştur.*
2. *SP popülasyonunda yer alan her bir kromozom için amaç fonksiyonunu (aynı zamanda uygunluk değeri) hesapla.*
3. *SP popülasyonundan e (elitizm oranına göre belirle) adet en iyi uygunluk değerine sahip bireyi yeni popülasyonu oluşturacak olan SPY kümesine aktar.*
4. *SP popülasyonundan seçim yöntemine göre n (pb-e) adet kromozomu ebeveynler olmak üzere seç.*
5. *Seçilen n adet kromozomdan her ikili ebeveyni çaprazlama oranına göre çaprazlama işlemine tabi tutarak çocuk kromozomları oluştur.*
6. *Oluşturulan n adet çocuk kromozoma çaprazlama sonrası tamir işlemini uygula.*
7. *Tamir işleminden geçen n adet çocuk kromozoma mutasyon oranına göre mutasyon uygula.*
8. *Elde edilen n adet çocuk kromozomun tamamını daha önce e adet elit kromozomun eklendiği SPY kümesine ekle.*
9. *SP popülasyonu SPY kümesi ile değiştir.*
10. *İterasyon sayısı tamamlandıysa ya da belirlenen en iyi uygunluk değerine ulaşıldıysa algoritmayı sonlandır. Aksi takdirde 2. adıma geç.*

GA'da yer alan kavramlar ile uygulama adımlarının ayrıntılı açıklaması aşağıdaki gibidir.

Kromozom yapısı: Kromozom gösteriminde dizi yapısı kullanılmıştır. Dizinin yapısını, sınav yerleşimin yapılacağı sınıfın sıra oturum düzeni belirlemektedir. Sınıfta öğrencilerin oturabileceği her bir oturum yeri kromozomdaki bir genle yani dizinin bir elemanı ile

temsil edilmektedir. Bu genlerin alabileceği değerler sınıf seviyelerini temsil eden 9-12 arasında bir rakamdır. Eğer ilgili oturum yerinde hiçbir öğrenci oturmayacaksa kromozomun ilgili gen değeri yani dizinin ilgili elemanın değeri “boş” olarak belirlenmektedir. Algoritmada kullanılan dizi Şekil 3.13’deki gibi tanımlanmıştır.

```
string [,] sp =new string[satir,sutun]
```

Şekil 3.13. Dizi tanımlaması

Örnek olarak sınıfta ki sıra yapısı 5×3 şeklinde (dikey hizada 5 ve yatay hizada 3 sıra) olduğunda sınıfta toplam 15 sıra yer alır. Her sırada 2 kişi oturduğu için $3 \times 5 \times 2 = 30$ adet sınav oturum yeri söz konusudur. Dolayısıyla dizi sütun boyutu 5, satır boyutu 6 (3×2) olmak üzere dizinin elaman sayısı ve kromozomdaki toplam gen sayısı 30 olmaktadır.

İlk popülasyonun oluşturulması: İlk popülasyonda yer alan kromozomlar belirlenen popülasyon büyüklüğü kadar rastgele oluşturulmaktadır. Kromozomlar, bilgisayar tarafından 9-12 arasında rastgele üretilen rakamın her bir gene atanması ile oluşturulmaktadır. Bu atamada 9-12 arasında kaç adet rakam üretileceği; sınıfta sınava girecek öğrencilerin her bir sınıf düzeyine göre toplam sayılarına göre belirlenmektedir. Her bir sınıfta düzeylere göre kaç tane öğrencinin sınava gireceği sınıflar arasında eşit bir dağılım sağlayacak şekilde önceden belirlenmektedir.

Örneğin sınava girecek öğrenci sayıları 9. ve 10. sınıf düzeyleri için 70, 11. ve 12. Sınıf düzeyleri için 80 ve sınav yeri olarak 10 adet sınıf belirlendiği düşünüldüğünde; her bir sınıfta sınava girecek öğrenci sayıları 9. ve 10. sınıf düzeyleri için 7’şer öğrenci 11. ve 12. sınıf düzeyleri için 8’er öğrenci olacaktır.

Amaç fonksiyonu ve uygunluk fonksiyonun belirlenmesi: Amaç fonksiyonu aynı zamanda uygunluk fonksiyonudur. Amaç fonksiyonu cezalandırma yöntemiyle oluşturulmuştur. En iyi kromozom en az cezayı alan kromozom olacaktır. Dolayısıyla sistem minimize edilecektir. Cezalandırma yapısı aşağıdaki gibidir (Vatansever ve Arıcı, 2019).

1. Bir kromozomda oturum planına göre aynı seviyede yan yana gelen öğrenciler için 10 ceza puanı verilmektedir. Bu kısıt iki farklı şekilde değerlendirilmektedir.

Program arayüzünde yer alan “Ayrı sıralarda da yan yana kısıtı ekle” seçeneği aktif edildiğinde farklı sıralarda da olsa aynı hizada yer alan sıralarda yan yana gelen öğrenciler için 10 ceza puanı verilmektedir. Kısıt aktif değilse ayrı sıralarda, fakat aynı hizada yan yana gelen öğrenciler için ceza puanı verilmemektedir.

2. Bir kromozomda oturum planına göre aynı seviyede çaprazda çakışan öğrenciler için 5 ceza puanı verilmektedir.
3. Bir kromozomda oturum planına göre aynı seviyede arka arkaya gelen öğrenciler için 2 ceza puanı verilmektedir.

y : x kromozomunda yan yana gelen öğrenci sayısı,

a : x kromozomunda arka arkaya gelen öğrenci sayısı,

c : x kromozomunda çaprazda çakışan öğrenci sayısı olmak üzere;

x kromozomunda yan yana gelen öğrencilerin toplam ceza puanı (cp) Eş. 3.6 ile hesaplanmaktadır.

$$\sum cp = 10 * y \quad (3.6)$$

x kromozomunda arka arkaya gelen öğrencilerin toplam ceza puanı (ca) Eş. 3.7 ile hesaplanmaktadır.

$$\sum ca = 2 * a \quad (3.7)$$

x kromozomunda çaprazda çakışan öğrencilerin toplam ceza puanı (cc) Eş. 3.8 ile hesaplanmaktadır.

$$\sum cc = 5 * c \quad (3.8)$$

Amaç fonksiyonu (Uygunluk fonksiyonu) ise Eş. 3.9 ile hesaplanmaktadır.

$$F(x) = \sum cp + \sum ca + \sum cc \quad (3.9)$$

Elit bireylerin seçilmesi: Popülasyonda en az ceza puanına sahip bireylerden elitizm oranına göre belirlenen sayıda birey bir sonraki popülasyona direkt olarak aktarılır. Elitizm

oranı %20 olarak belirlendiğinde popülasyonun %80'i ebeveyn birey olarak seçilmektedir. Ebeveyn olarak seçilecek birey sayısı belirlendikten sonra kalan birey sayısı elit birey adetini belirtmektedir. Seçilen her iki birey ebeveyn kabul edilerek; çaprazlama ve mutasyona tabi tutulup iki tane çocuk elde edilmektedir. Bu sebeple ebeveyn olarak seçilen kromozom adeti çift sayı olmalıdır. Örneğin elitizm oranının %20 olarak belirlendiğinde; popülasyon büyüklüğü 15 olarak kabul edilirse çaprazlama ve mutasyon için seçilmesi gereken ebeveyn sayısı 12 ($15 \cdot 0.8$) olacaktır. Bu durumda elit birey sayısı da 3 ($15-12$) olacaktır.

Çaprazlama ve mutasyona tabi tutulacak bireylerin seçilmesi: Seçim yöntemi olarak rulet tekeri ve turnuva seçim yöntemleri kullanıcının seçimine göre kullanılabilir. Ebeveyn ikililerinin seçiminde farklı iki birey seçilmesi sağlanmıştır. Rulet tekerinde her bir kromozom uygunluk değerine göre 0-1 arasında birikimli seçilme olasılığı olarak adlandırılan bir değer rulet çarkında kendine yer edinmektedir. 0-1 arasında rastgele olarak elde edilen rakama göre rulet çarkında uygun aralıkta yer alan birey (kromozom) seçilmektedir.

Rulet tekeri seçim yöntemi için seçilme olasılığının hesaplanmasında Eş. 3.10 kullanılmaktadır (Aladağ, 2010).

pb : Popülasyon büyüklüğü,

$F(x)$: İlgili kromozoma ait uygunluk değeri ve

$Enb(F)$: Popülasyondaki en büyük amaç fonksiyonu değeri (en büyük ceza değeri) olmak üzere;

$$F(x) = 0 \text{ ise } s(x) = 1$$

$$F(x) > 0 \text{ için } s(x) = (Enb(F) + 1) - F(x) \quad (3.10)$$

$$\text{Seçilme Olasılığı: } s(x) / \sum_{n=1}^{pb} (s(n))$$

Çizelge 3.2'de örnek olarak 15 elemanlık bir popülasyon gösterilmektedir. İlk değer uygunluk değeri; ikinci değer rulet tekeri için seçilme olasılığı ve üçüncü değer birikimli seçilme olasılığı değeridir. Rulet tekeri seçiminde bilgisayar tarafından rastgele üretilen

değerin 0,3 olduğunu kabul edersek; birikimli seçilme olasılığı değerine göre bu seçimde popülasyondaki 5. Kromozom seçilecektir.

Çizelge 3.2. GA’da uygulanan rulet tekeri seçim örneği

Birey	Uygunluk Değeri	Seçilme Olasılığı Değeri	Birikimli Seçilme Olasılığı Değeri
1. Kromozom	83	0,0681897572173282	0,0681897572173282
2. Kromozom	71	0,0797147866061724	0,147904543823501
3. Kromozom	86	0,0658110447562586	0,213715588579759
4. Kromozom	90	0,0628861094337582	0,276601698013517
5. Kromozom	70	0,0808535692719749	0,357455267285492
6. Kromozom	104	0,0544206716253677	0,41187593891086
7. Kromozom	100	0,0565974984903824	0,468473437401242
8. Kromozom	118	0,0479639817715105	0,516437419172753
9. Kromozom	64	0,0884335913912225	0,604871010563975
10. Kromozom	57	0,0992938570006709	0,704164867564646
11. Kromozom	106	0,0533938665003608	0,757558734065007
12. Kromozom	96	0,0589557275941483	0,816514461659156
13. Kromozom	128	0,0442167956956113	0,860731257354767
14. Kromozom	101	0,0560371272182004	0,916768384572967
15. Kromozom	68	0,0832316154270329	1

Turnuvada ise rastgele seçilen iki bireyden ceza puanı en az olan birey ebeveyn bireylerden biri olarak seçilmektedir.

Çaprazlama: Çaprazlama yöntemi olarak üniform çaprazlama kullanılmıştır. Çaprazlama oranına göre seçilen birey ikilileri çaprazlama işlemine tabi tutulmaktadır. Rastgele 0-1 arasında üretilen değer çaprazlama oranından düşükse çaprazlama işlemi gerçekleşmektedir. Şekil 3.14’de üniform çaprazlama örneği için çaprazlamaya tabi olacak ebeveyn bireyler gösterilmektedir.

	1	2	3	4	5	6
5	11	9	11	12	12	9
4	12	11	10	9	11	10
3	10	9	10	11	9	9
2	12	11	11	10	10	12
1	9	10	12	11	11	12

(a) Ebeveyn Birey 1

	1	2	3	4	5	6
5	12	10	11	9	12	9
4	9	11	10	11	11	10
3	10	12	12	11	11	12
2	12	9	9	9	10	12
1	9	10	10	11	11	12

(b) Ebeveyn Birey 2

Şekil 3.14. Çaprazlamaya tabi olacak ebeveyn bireyler

Çaprazlamada kullanılan maske ve elde edilen çocuk bireyler Şekil 3.15’de gösterilmektedir. 1. Çocuk için çaprazlama maskesinde 0 görülen yerde birinci ebeveynden; 1 görülen yerde 2. ebeveynden gen değeri alınmaktadır. 2. Çocuk için çaprazlama maskesinde 0 görülen yerde ikinci ebeveynden; 1 görülen yerde 1. ebeveynden gen değeri alınmaktadır.

	1	2	3	4	5	6
5	0	0	1	1	1	0
4	1	0	1	1	1	0
3	1	1	0	1	0	0
2	0	1	1	1	0	1
1	0	0	1	1	1	0

(a) Çaprazlamada kullanılan maske

	1	2	3	4	5	6
5	11	9	11	9	12	9
4	9	11	10	11	11	10
3	10	12	10	11	9	9
2	12	9	9	9	10	12
1	9	10	10	11	11	12

(b) Çocuk birey 1

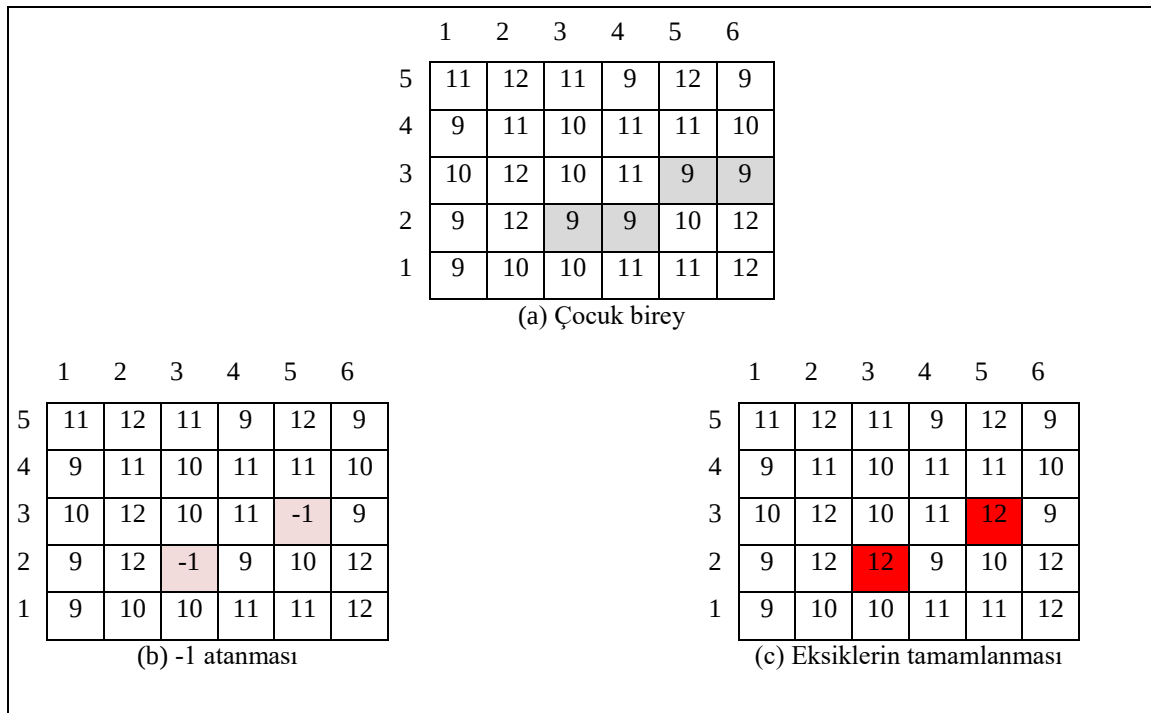
	1	2	3	4	5	6
5	12	10	11	12	12	9
4	12	11	10	9	11	10
3	10	9	12	11	11	12
2	12	11	11	10	10	12
1	9	10	12	11	11	12

(c) Çocuk birey 2

Şekil 3.15. Çaprazlamada kullanılan maske ve elde edilen çocuk bireyler

Çaprazlama sonrası tamir fonksiyonu: Çaprazlama işleminden sonra sınıf seviyelerinin sayısal olarak dağılımı değişebilmektedir. Her bir sınıfta her sınıf seviyesinden kaç öğrencinin yer alacağı bellidir. Bu sayılara göre kromozomda fazla olan sınıf seviyelerinin yerine -1 değeri atanmaktadır. Eksik olan sınıf seviyeleri ise -1 olarak belirlenen genlere atanmaktadır. Fazla olan seviyelere -1 atanmasında öncelik oturum planına göre sırasıyla yan yana, çapraz ve arka arkaya aynı değeri içeren genlere verilmiştir. -1 değerlerinin yerine eksik olan sınıf seviyelerinin atanmasında sıralı yöntem kullanılmıştır.

Örneğin çaprazlama sonrası elde edilen çocuk bireyde sınıf seviyelerine göre yerleşim 9.sınıf seviyesi için 9 kişi, 10.sınıf seviyesi için 7 kişi, 11.sınıf seviyesi için 8 kişi ve 12.sınıf seviyesi için 6 kişi olacak şekilde gerçekleştiğini düşünürsek çaprazlama sonrası tamir fonksiyonu Şekil 3.16'daki gibi gerçekleşecektir. Başlangıçta sınıf içerisinde belirlenen seviye dağılımı 9.sınıf seviyesi için 7 kişi, 10.sınıf seviyesi için 7 kişi, 11.sınıf seviyesi için 8 kişi ve 12.sınıf seviyesi için 8 kişidir. Buna göre Şekil 3.14'de gösterilen çocuk bireyde 2 adet 9. sınıf seviyesi fazla, 2 adet 12. sınıf seviyesi ise eksiktir. Bu durumda 2 adet 9 değeri içeren gene yukarıda belirtilen önceliklere göre -1 değeri atanacaktır. Daha sonrasında -1 değeri atanmış genlerin değeri eksik olan sınıf seviyesi değeri 12 ile değiştirilecektir.



Şekil 3.16. Çaprazlama sonrası tamir fonksiyonu örneği

Mutasyon: Mutasyonda belirlenen mutasyon oranına göre çaprazlama işlemi sonucu elde edilen çocuk bireyler mutasyona tabi olmaktadır. Rastgele 0-1 arasında üretilen değer mutasyon oranından düşükse mutasyon işlemi gerçekleşmektedir. Mutasyon işleminde rassal mutasyon yöntemi kullanılmıştır. Seçilen birey rastgele yeniden oluşturulmuştur. Algoritmanın istenmeyen bir çözümde yakınsamasını önlemek ve popülasyonda çeşitliliği sağlamak için çalışma anında mutasyon oranı değeri değiştirilmiştir. Her 13 iterasyonda bir ceza puanı 0'ın üstünde ise sadece bir iterasyon için mutasyon oranı değeri 1 olarak belirlenmiştir.

İterasyon Adedi: Algoritmanın tekrarlama yani çalışma sayısını göstermektedir. Bu parametre ile algoritmanın öğrenme değeri belirlenebilmektedir. İterasyon sayısını artırmak istenilen çözüme ulaşılmasını sağlamakla beraber sistem bir çözümde takılabilir. İtersayon sayısının gereksiz yere artırılması işlem süresini gereksiz yere artırır. Algoritma en iyi çözüme ulaştığında iterasyon sayısını tamamlamadan kırılmaktadır.

GA'nın oturum planı hazırlama problemine uygulanmasına ait programın kullanıcı arayüzü Resim 3.13'de verilmiştir.

Genetik Algoritma ile Oturum Planı Hazırlama Programı

Sınava Girecek Öğrenci Sayıları:

9.Sınıf: 10.Sınıf:

11.Sınıf: 12.Sınıf:

Sınıf Sayısı:

Sınıf Sıra Düzeni: ☒ 5x3 ☐ 4x4 ☐ 6x3

☐ Özel Seçim X

☐ Aynı sıralarda da yan yana kısıtı ekle

Popülasyon Büyüklüğü:

Çapazlama Oranı:

Mutasyon Oranı:

İterasyon Sayısı:

Elitizm Yüzdesi:

Seçim Yöntemi: ☐ Rulet Teker ☒ Turnuva

	1	2	3	4	5	6
▶ 1	9	10		12	11	11
2	11	12		10	9	9
3	9	10		11	12	11
4	11	12		10	9	10
* 5	9	10		12	11	11

Deneme 1: En iyi ceza Puanı: 0 İterasyon: 792 Sure: 209
Deneme 2: En iyi ceza Puanı: 0 İterasyon: 248 Sure: 63
Deneme 3: En iyi ceza Puanı: 0 İterasyon: 440 Sure: 97
Deneme 4: En iyi ceza Puanı: 0 İterasyon: 726 Sure: 151
Deneme 5: En iyi ceza Puanı: 0 İterasyon: 1365 Sure: 261
Deneme 6: En iyi ceza Puanı: 0 İterasyon: 137 Sure: 39
Deneme 7: En iyi ceza Puanı: 0 İterasyon: 1673 Sure: 335
Deneme 8: En iyi ceza Puanı: 0 İterasyon: 148 Sure: 43
Deneme 9: En iyi ceza Puanı: 0 İterasyon: 800 Sure: 153
Deneme 10: En iyi ceza Puanı: 0 İterasyon: 410 Sure: 90

Resim 3.13. GA uygulanan oturum planı hazırlama programı kullanıcı arayüzü

3.3.3. Yapay bağışıklık sisteminin oturum planı hazırlama problemine uygulanması

YBS ile ortak sınavlar için en iyi oturum planının hazırlanmasında optimizasyon problemlerinin çözümü için geliştirilen CLONAG isimli KSA temel alınmıştır. Problemin çözümü için uygulanan probleme özgü uyarlanmış KSA sözde kodları (pseudo code) aşağıdaki gibidir.

1. Antikorların gen değerlerini 9-12 arasında rastgele üreterek N adet Ab popülasyonunu (hafıza hücreleri-girdi kümesi) oluştur.
2. En iyi oturma planının optimizasyonu için cezalandırma temelli, sistemi minimize edecek bir amaç fonksiyonu (F) bulunmaktadır. Amaç fonksiyonu aynı zamanda uygunluk değeridir. Her bir x (Ab_i) antikoru için $F(x)$ uygunluk değerini hesapla.
3. En yüksek uygunluğa sahip n adet antikoru Ab popülasyonundan seç. Seçilen bu antikorlar ile $Ab-n$ kümesini oluştur.
4. Seçilen n tane antikorun her birini uygunluk değerleri ile orantılı olarak klonla (kopyala). Daha iyi uygunluk değerine sahip olan antikorlar daha fazla sayıda klonlanır. Bu klonlar ile C kümesini oluştur.
5. C kümesinde yer alan antikolara, uygunluk değeri ile ters orantılı olarak mutasyon işlemi uygula. Daha iyi uygunluk değerine sahip olan antikorlar daha az mutasyona uğrar. Mutasyona uğramış antikorlar ile C^* kümesini oluştur.
6. C^* kümesinde yer alan antikorların uygunluk değerlerini hesapla.
7. Hafıza hücrelerini oluşturan Ab popülasyonu ile klonlanmış ve mutasyona uğramış antikorların oluşturduğu C^* kümesini birleştir. Seçim yöntemine göre N adet antikoru seçerek Ab popülasyonu yeniden oluştur.
8. Antikorların gen değerlerini 9-12 arasında rastgele üreterek d adet yeni antikor oluştur. Bu antikorlar ile $Ab-d$ kümesini oluştur.
9. Ab popülasyonda yer alan antikolardan en düşük uygunluk değerine sahip d tane antikoru, $Ab-d$ kümesiyle yer değiştir.
10. Kuşak sayısı tamamlandıysa ya da Ag (antijen) için en iyi uygunluk değerine ulaşıldıysa algoritmayı sonlandır. Aksi takdirde 2. adıma geç.

YBS’de yer alan kavramlar ile uygulama adımlarının ayrıntılı açıklaması aşağıdaki gibidir.

Antikor gösterimi: Antikor gösterimi GA’da kullanılan kromozom gösterimi ile aynı şekildedir.

Hafıza hücreleri girdi kümesinin oluşturulması (Popülasyon) ($Ab-N$): Popülasyonu oluşturan yani çözüme aday olarak gösterilen antikorlar topluluğudur. İlk popülasyon GA’daki gibi rastgele oluşturulmaktadır.

Seçim havuzunun boyutu (Ab-n): Popülasyondan klonlanmak yani kopyalanmak için seçilen antikor topluluğudur. Uygunluk değeri daha iyi olan yani antijene karşı en iyi uyarılan antikorlar seçim havuzuna alınmaktadır. Örneğin popülasyon büyüklüğü 15 ve seçim havuzu boyutu (yüzdesi) %90 olduğunda seçilen antikor sayısı Eş. 3.11 ile hesaplanır.

$$n = \text{round}(N * 0.9) \rightarrow n = \text{round}(15 * 0.9) \rightarrow n = 14 \quad (3.11)$$

Sonuçta 14 adet en iyi uygunluk değerine sahip antikor klonlanmak üzere seçim havuzuna alınmaktadır.

Ag (Antijen) hedefi: En iyi oturma planının hazırlanmasında mümkün olduğunca sağlanması gereken kısıtlar şunlardır:

1. Aynı sırada ve seçime göre aynı hizada yer alan sıralarda aynı sınıf seviyesinde öğrenciler yan yana oturmamalıdır.
2. Aynı hizada yer alan yani arka arkaya gelen sıralarda çaprazda aynı sınıf seviyesinde öğrenciler yer almamalıdır.
3. Aynı hizada yer alan sıralarda yani arka arkaya gelen sıralarda aynı sınıf seviyesinde olan öğrenciler önlü arkalı olacak şekilde oturmamalıdır.

Uygunluk değerinin belirlenmesi: GA uygulanan yöntem ile aynı uygunluk fonksiyonu (aynı zamanda amaç fonksiyonu) kullanılmaktadır (Bkz. Eş. 3.9).

Klon havuzu boyutu(C): Seçim havuzunda yer alan antikorlar uygunluk değeri ile orantılı olarak klonlanmaktadır. Klonlanan antikorlar klon havuzunu yani C kümesini oluşturmaktadır. Daha iyi uygunluk değerine sahip olanlar daha kötü uygunluk değerine sahip olan antikora göre daha fazla sayıda klonlanmaktadır. Klon havuzunun boyutu seçim havuzunun boyutuna da bağlıdır. Toplam klon sayısının belirlenmesinde Eş. 3.12 ve seçim havuzuna alınan her bir antikoru kaç adet klonlanacağını belirlenmesinde Eş. 3.13 kullanılmaktadır (De Castro ve Von Zuben, 2002).

x_i : x antikoruğun uygunluk değerine göre popülasyondaki sırası,

β : Klonlanma katsayısı ve

n: Seçim havuzunda yer alan antikoru sayısı olmak üzere;

Klon havuzu boyutu(C):

$$C = \sum_{x=1}^n \text{round}\left(\frac{\beta \cdot n}{x_i}\right) \quad (3.12)$$

x antikoruına ait klon sayısı($ka(x)$):

$$ka(x) = \text{round}\left(\frac{\beta \cdot n}{x_i}\right) \quad (3.13)$$

Seçim havuzuna alınan antikoruların klon sayılarının belirlenmesine ilişkin örnek hesaplama Şekil 3.17’de verilmiştir. Örnekte seçim havuzunda uygunluk değerine göre en iyiden en kötüye doğru sıralı bir şekilde 14 adet antikoru yer almaktadır. Bu örnekte seçim havuzunda yer alan her bir antikoru klon sayısı toplandığında toplam klon sayısı (C) olarak 41 değeri elde edilecektir.

$ka(1) = \text{round}\left(\frac{0.9 \cdot 14}{1}\right) \rightarrow 13$ adet klon	$ka(2) = \text{round}\left(\frac{0.9 \cdot 14}{2}\right) \rightarrow 6$ adet klon
$ka(3) = \text{round}\left(\frac{0.9 \cdot 14}{3}\right) \rightarrow 4$ adet klon	$ka(4) = \text{round}\left(\frac{0.9 \cdot 14}{4}\right) \rightarrow 3$ adet klon
$ka(5) = \text{round}\left(\frac{0.9 \cdot 14}{5}\right) \rightarrow 3$ adet klon	$ka(6) = \text{round}\left(\frac{0.9 \cdot 14}{6}\right) \rightarrow 2$ adet klon
$ka(7) = \text{round}\left(\frac{0.9 \cdot 14}{7}\right) \rightarrow 2$ adet klon	$ka(8) = \text{round}\left(\frac{0.9 \cdot 14}{8}\right) \rightarrow 2$ adet klon
$ka(9) = \text{round}\left(\frac{0.9 \cdot 14}{9}\right) \rightarrow 1$ adet klon	$ka(10) = \text{round}\left(\frac{0.9 \cdot 14}{10}\right) \rightarrow 1$ adet klon
$ka(11) = \text{round}\left(\frac{0.9 \cdot 14}{11}\right) \rightarrow 1$ adet klon	$ka(12) = \text{round}\left(\frac{0.9 \cdot 14}{12}\right) \rightarrow 1$ adet klon
$ka(13) = \text{round}\left(\frac{0.9 \cdot 14}{13}\right) \rightarrow 1$ adet klon	$ka(14) = \text{round}\left(\frac{0.9 \cdot 14}{14}\right) \rightarrow 1$ adet klon

Şekil 3.17. Antikoruara ait klon sayısı hesaplama örneği

Somatik Hipermutasyon: Klon havuzunda (C) yer alan antikoru uygunluk değeri ile ters orantılı olarak mutasyon işlemine tabi olur. Uygunluk değeri daha iyi olan antikoruların mutasyona uğrama oranı uygunluk değeri daha kötü olan antikoruara göre daha düşüktür. Klon havuzunda yer alan her bir antikoru ait mutasyon oranının hesaplamasında Eş. 3.14 kullanılmaktadır (Vatansever ve Arıcı, 2019).

x_i : x antikorunun uygunluk değerine göre klon havuzundaki sırası,

C: Klon havuzu boyutu

F(x): x antikorunun uygunluk değeri ve

F(n): Klon havuzunda yer alan antikorlardan en kötü uygunluk değerine sahip antikorun uygunluk değeri olmak üzere;

x antikoruna ait mutasyon oranı(α):

$$\alpha = F(x) * x_i / (F(n) * C) \quad (3.14)$$

Klon havuzunda yer alan antikorlar seçime göre iki farklı şekilde mutasyona tabi tutulmaktadır.

Rastgele Mutasyon: Mutasyon için her bir antikorun her bir geni tek tek değerlendirilmektedir. Bilgisayar tarafından 0-1 aralığında rastgele bir reel sayı üretilmekte; üretilen sayı ilgili antikor için hesaplanan mutasyon oranından küçükse ilgili gen mutasyona uğramaktadır. Mutasyon işleminde; ilgili gen değeri ile antikorun rastgele seçilen bir gen değeri değiştirilmektedir. Şekil 3.18’de verilen örnekte antikor için mutasyon oranı olarak $\alpha=0,7$ olarak belirlendiği, 2.satır 2.sütunda yer alan gen değerine göre işlem yapıldığı ve bilgisayar tarafından rastgele üretilen değerin 0,54 olduğunu farz edilmektedir. Üretilen değer mutasyon oranından küçük olduğu için ilgili gen mutasyon işlemine tabi tutulacaktır. Rastgele belirlenen satır değeri 5 ve sütun değeri 3 olarak düşünülmüştür. Bu durumda 5.satır 3.sütunda yer alan gen değeri ile 2. Satır ve 2.sütunda yer alan gen değeri yer değiştirecektir.

	1	2	3	4	5	6
5	9	11	12	11	10	11
4	12	10	10	10	9	12
3	10	12	10	12	11	11
2	10	11	12	12	9	9
1	9	12	11	9	9	11

(a) Mutasyon Öncesi

	1	2	3	4	5	6
5	9	11	11	11	10	11
4	12	10	10	10	9	12
3	10	12	10	12	11	11
2	10	12	12	12	9	9
1	9	12	11	9	9	11

(b) Mutasyon Sonrası

Şekil 3.18. Rastgele mutasyon örneği

İyileştirilmiş Mutasyon: Mutasyon için her bir antikorun her bir geni tek tek değerlendirilmektedir. Bilgisayar tarafından 0-1 aralığında rastgele bir reel sayı üretilmekte; üretilen sayı ilgili antikor için hesaplanan mutasyon oranından küçükse ilgili gen mutasyona uğramaktadır. Mutasyon işleminde ilgili gen değeri ile antikorun şarta göre seçilmiş bir gen değeri değiştirilmektedir. Mutasyon işleminde sırasıyla aşağıdaki seçim şartları uygulanmaktadır:

- İlk olarak oturma planı açısından yan yana gelen yerleşimlerde aynı seviye değerini içeren genler varsa bu genlerden dizi elamanı olarak önde olanı seçilir.
- İlk seçim şartına göre işlem yapılamamışsa ikinci olarak oturma planı açısından çaprazda aynı seviye değerini içeren genlerden dizi elamanı olarak önde olanı seçilir.
- Birinci ve ikinci seçim şartlarına göre işlem yapılamamışsa oturma planı açısından arka arkaya aynı seviye değerini içeren genlerden dizi elamanı olarak önde olanı seçilir.
- I,II ve III. seçim şartlarına göre işlem yapılamamışsa değişim için rastgele bir gen seçilmektedir.

Şekil 3.19’da verilen örnekte 2.satır 2.sütunda yer alan genin iyileştirilmiş mutasyon işlemine uğradığı farz edilmektedir. Birinci kontrolde 2.satır 5 ve 6 nolu sütunlarda (aynı sırada) aynı sınıf düzeyinde iki öğrenci yan yana geldiği için değişim için seçilen genin satır numarası 2, sütun numarası 5 olacaktır. Eğer birinci şart sağlanmasaydı, seçim için sırasıyla diğer şartlara bakılacaktı.

	1	2	3	4	5	6
5	9	11	12	11	10	11
4	12	10	10	10	9	12
3	10	12	10	12	11	11
2	10	11	12	12	9	9
1	9	12	11	9	9	11

(a) Mutasyon Öncesi

	1	2	3	4	5	6
5	9	11	12	11	10	11
4	12	10	10	10	9	12
3	10	12	10	12	11	11
2	10	9	12	12	11	9
1	9	12	11	9	9	11

(b) Mutasyon Sonrası

Şekil 3.19. İyileştirilmiş mutasyon örneği

Tüm Çözümlerin Toplanması: Klonlama ve hipermutasyon işlemleri sonucu elde edilen antikorlar (C^* kümesi) ile hafıza hücrelerini ($Ab-N$) oluşturan antikorlar birleştirilir.

Yeni Popülasyonun Oluşturulması: Birleştirilen tüm çözümlerden ($Ab-N+C^*$) tercihe göre iki şekilde yeni hafıza hücreleri yani popülasyon oluşturulmaktadır.

1. *En İyi:* Tüm çözümleri içeren antikor topluluğu uygunluk değerine, yani ceza puanına göre artan şekilde sıralanmaktadır. Sıralama sonucuna göre ilk N adet antikor yeni popülasyonu oluşturmak üzere seçilmektedir.
2. *Farklı En İyi:* Tüm çözümleri içeren antikor topluluğu uygunluk değerine, yani ceza puanına göre artan şekilde sıralanmaktadır. Sıralama sonucuna göre ilk N adet antikor aynı uygunluk değerine sahip ikinci bir antikor seçilmemek şartı gözetilerek yeni popülasyonu oluşturmak üzere seçilmektedir.

Popülasyondaki En Kötü Bireylerin Atılması (En Kötü Atıkların Atılması): Yeni nesil popülasyonda yer alan en kötü uygunluk değerine sahip antikorlardan d tanesi popülasyondan atılmakta, yerine rastgele yeni antikorlar oluşturulmaktadır. Popülasyondan atılacak antikor sayısını atık yüzde değeri belirlemektedir. Atık yüzde değerinin %20 ve popülasyon büyüklüğünün 15 olarak belirlendiğini düşünürsek, atılacak antikor sayısı (d) Eş. 3.15 ile hesaplanır.

$$d = \text{round}(15 * 0.2) \rightarrow 3 \quad (3.15)$$

Bu durumda popülasyondaki en kötü 3 adet antikor silinerek, yerine rastgele antikorlar oluşturulacaktır. Bu işlem ile popülasyonda çeşitlilik sağlanarak, istenmeyen bir çözümde algoritmanın takılıp kalması engellenmektedir.

Kuşak sayısı: GA'da yer alan iterasyon sayısı ile aynıdır. Bir kuşak için algoritma bir kez çalışmaktadır. GA'da olduğu gibi bu parametre ile algoritmanın öğrenme değeri belirlenebilmektedir. Kuşak sayısını artırmak istenilen çözüme ulaşılmasını sağlamakla beraber sistem bir çözümde takılabilir. Kuşak sayısının gereksiz yere artırılması işlem süresini gereksiz yere uzatır. Algoritma en iyi çözüme ulaştığında kuşak sayısını tamamlamadan kırılmaktadır.

YBS'nin oturum planı hazırlama problemine uygulanmasına ait programın kullanıcı arayüzü Resim 3.14'de verilmiştir.

Yapay Bağışıklık Sistemi ile Oturum Planı Hazırlama Programı

Sınava Girecek Öğrenci Sayıları:

9.Sınıf: 10.Sınıf:

11.Sınıf: 12.Sınıf:

Sınıf Sayısı:

Sınıf Sıra Düzeni: ☒ 5x3 ☐ 4x4
☐ 6x3

☐ Özel Seçim X

☐ Aynı sıralarda da yan yana kısıtı ekle

Populasyon Büyüklüğü:

Klonlama Değeri:

Seçim Havuzu Yüzdesi:

Atık Antikor Yüzdesi:

İterasyon Sayısı:

Seçim Yöntemi:

☐ En İyi'ler ☒ Farklı En İyi'ler

Hipermutasyon Şekli:

☐ Rasgele ☒ İyileştirilmiş

ÇALIŞTIR

	1	2		3	4		5	6
▶ 1	12	9		11	12		10	11
2	10	11		9	10		9	12
3	12	9		11	12		10	11
4	11	10		10	9		12	9
* 5	12	9		11	12		11	10

Deneme 1: Toplam İterasyon: 1080 Son Ceza: 0 Sure: 1021
Deneme 2: Toplam İterasyon: 612 Son Ceza: 0 Sure: 587
Deneme 3: Toplam İterasyon: 862 Son Ceza: 0 Sure: 772
Deneme 4: Toplam İterasyon: 2000 Son Ceza: 2 Sure: 1792
Deneme 5: Toplam İterasyon: 1320 Son Ceza: 0 Sure: 1191
Deneme 6: Toplam İterasyon: 2000 Son Ceza: 2 Sure: 1761
Deneme 7: Toplam İterasyon: 2000 Son Ceza: 2 Sure: 1870
Deneme 8: Toplam İterasyon: 2000 Son Ceza: 4 Sure: 2054
Deneme 9: Toplam İterasyon: 2000 Son Ceza: 2 Sure: 1817
Deneme 10: Toplam İterasyon: 225 Son Ceza: 0 Sure: 235

Resim 3.14. YBS uygulanan oturum planı hazırlama programı kullanıcı arayüzü

4. YÖNTEME İLİŞKİN DENEY TASARIMI VE BULGULAR

GA ve YBS uygulanan yöntemler için farklı parametrelerle çalışma deneyleri yapılmıştır. Bu deneylerden elde edilen bulgular çizelgeler ile özetlenmiştir. Kesim 4.1’de GA ve Kesim 4.2’de YBS için yapılan deneyler ile elde edilen bulgular açıklanmıştır. Kesim 4.3’te iki yöntemin çalışması karşılaştırmalı olarak incelenmiş ve sonuçlar çizelgeler ve grafiklerle sunulmuştur.

4.1. Genetik Algoritma ile Oturum Planı Hazırlama Uygulama Sonuçları

Aşağıdaki seçimler doğrultusunda algoritma Çizelge 4.1’de yer alan farklı GA parametreleri için 10’ar kez çalıştırılarak 16 deney gerçekleştirilmiştir. Her bir deney için elde edilen en iyi ceza puanlarının ve algoritma tamamlanma sürelerinin ortalaması Çizelge 4.1’de verilmiştir.

Sınıf düzeylerine göre öğrenci sayıları: 9.Sınıf:7, 10.Sınıf: 7, 11.Sınıf: 8, 12.Sınıf: 8

Sıra yapısı: 5*3 (30 oturum yeri)

İterasyon Adedi: 2000

Elitizm oranı: %20 şeklinde kabul edilmektedir.

Çizelge 4.1. GA deney sonuçları (Elitizm oranı: %20)

Parametreler	Deney1	Deney2	Deney3	Deney4	Deney5	Deney6	Deney7	Deney8	Deney9	Deney10	Deney11	Deney12	Deney13	Deney14	Deney15	Deney16
Popülasyon Büyüklüğü	10	10	10	10	10	10	10	10	15	15	15	15	15	15	15	15
Çaprazlama Oranı	0,7	0,7	0,7	0,7	0,9	0,9	0,9	0,9	0,7	0,7	0,7	0,7	0,9	0,9	0,9	0,9
Mutasyon Oranı	0,03	0,03	0,1	0,1	0,03	0,03	0,1	0,1	0,03	0,03	0,1	0,1	0,03	0,03	0,1	0,1
Seçim Yöntemi	Rulet Tekerli	Turnuva	Rulet Tekerli	Turnuva	Rulet Tekerli	Turnuva	Rulet Tekerli	Turnuva	Rulet Tekerli	Turnuva	Rulet Tekerli	Turnuva	Rulet Tekerli	Turnuva	Rulet Tekerli	Turnuva
Ortalama Ceza Puanı	6,9	4,6	4,7	3,4	5,2	2,2	4,8	2	5,2	3,2	5,6	3,8	3,2	1,6	4,8	1,8
Ortalama Süre(ms)	131,3	125,4	126,3	135,6	136,9	135,9	152,2	148,3	185	190,8	194,8	198,9	204,1	197,8	214,2	228,9

Aşağıdaki seçimler doğrultusunda algoritma Çizelge 4.2’de yer alan farklı GA parametreleri için 10’ar kez çalıştırılarak 16 deney gerçekleştirilmiştir. Her bir deney için

elde edilen en iyi ceza puanlarının ve algoritma tamamlanma sürelerinin ortalaması Çizelge 4.2’de verilmiştir.

Sınıf düzeylerine göre öğrenci sayıları: 9.Sınıf:7, 10.Sınıf: 7, 11.Sınıf: 8, 12.Sınıf: 8

Sıra yapısı: 5*3 (30 oturum yeri)

İterasyon Adedi: 2000

Elitizm oranı: %10 şeklinde kabul edilmektedir.

Çizelge 4.2. GA deney sonuçları (Elitizm oranı: %10)

Parametreler	Deney1	Deney2	Deney3	Deney4	Deney5	Deney6	Deney7	Deney8	Deney9	Deney10	Deney11	Deney12	Deney13	Deney14	Deney15	Deney16
Popülasyon Büyüklüğü	10	10	10	10	10	10	10	10	15	15	15	15	15	15	15	15
Çaprazlama Oranı	0,7	0,7	0,7	0,7	0,9	0,9	0,9	0,9	0,7	0,7	0,7	0,7	0,9	0,9	0,9	0,9
Mutasyon Oranı	0,03	0,03	0,1	0,1	0,03	0,03	0,1	0,1	0,03	0,03	0,1	0,1	0,03	0,03	0,1	0,1
Seçim Yöntemi	Rulet Teker	Turnuva	Rulet Teker	Turnuva	Rulet Teker	Turnuva	Rulet Teker	Turnuva	Rulet Teker	Turnuva	Rulet Teker	Turnuva	Rulet Teker	Turnuva	Rulet Teker	Turnuva
Ortalama Ceza Puanı	5,6	5,4	6,2	3,4	4,4	2,8	5	2,8	1,4	0,6	2,4	0,4	1,4	0,2	1,6	0,4
Ortalama Süre(ms)	130,3	132,8	145,3	139,7	127,8	151,4	152,7	164	156,2	128,5	218,7	141	191	122,6	216,6	123,7

Deneylerden elde edilen bulgular doğrultusunda aşağıdaki değerlendirmeler yapılabilir.

Elitizm oranının %10 olarak alındığı deneylerde elitizm oranının %20 olarak alındığı deneylere göre çoğunlukla daha iyi ortalama ceza puanları elde edilmiştir. Bu sonuçlar doğrultusunda ideal elitizm oranı %10 olarak kabul edilmektedir.

Sadece seçim yöntemi parametresinin farklı olduğu deneylere bakıldığında, “Turnuva” seçim yönteminin kullanıldığı deneylerde “Rulet Teker” seçim yönteminin kullanıldığı deneylere göre daha iyi ortalama ceza puanlarının elde edildiği görülmektedir. Bu yüzden ideal seçim yöntemi olarak “Turnuva” seçim yöntemi belirlenmiştir.

Sadece çaprazlama oranı parametresinin farklı olduğu deneylere bakıldığında, çoğunlukla çaprazlama oranı olarak 0,9 değerinin kullanıldığı deneylerde çaprazlama oranı olarak 0,7

değerinin kullanıldığı deneylere göre daha iyi ortalama ceza puanlarının elde edildiği görülmektedir. Bu yüzden ideal çaprazlama oranı olarak 0,9 değeri belirlenmiştir.

Sadece mutasyon oranı parametresinin farklı olduğu deneylere bakıldığında, çoğunlukla mutasyon oranı olarak 0,03 değerinin kullanıldığı deneylerde mutasyon oranı olarak 0,1 değerinin kullanıldığı deneylere göre daha iyi ortalama ceza puanlarının elde edildiği görülmektedir. Bu yüzden ideal mutasyon oranı olarak 0,03 değeri belirlenmiştir.

Sadece popülasyon büyüklüğü parametresinin farklı olduğu deneylere bakıldığında, çoğunlukla popülasyon büyüklüğünün 15 olarak alındığı deneylerde popülasyon büyüklüğünün 10 alarak alındığı deneylere göre daha iyi ortalama ceza puanları elde edilirken; popülasyon büyüklüğünün artmasıyla birlikte işlem yapılacak kromozom sayısının artmasıyla paralel olarak algoritma tamamlanma süresi ortalamalarında ise artış görülmektedir.

Popülasyon büyüklüğü değiştirilerek Çizelge 4.2’de yer alan Deney14’ e ait GA parametreleri ile algoritma 10’ar kez çalıştırıldığında, Çizelge 4.3’deki bulgular elde edilmiştir. Bu bulgular doğrultusunda en ideal popülasyon büyüklüğü 15 olarak belirlenmiştir.

Çizelge 4.3. GA’nın farklı popülasyon büyüklüğü değerlerine göre çalışma sonuçları

Popülasyon Büyüklüğü	10	15	20
Ortalama Ceza Puanı	3,4	0,2	0,8
Algoritma Tamamlanma Süresi Ortalaması (ms)	149,8	150,8	219,8

Sadece iterasyon adedi parametresi değiştirilerek Çizelge 4.2’de yer alan Deney14’ e ait GA parametreleri ile algoritma 10’ar kez çalıştırıldığında, Çizelge 4.4’deki bulgular elde edilmiştir. Bu bulgulardan görüldüğü üzere; iterasyon adedinin artmasıyla birlikte daha iyi ortalama ceza puanı değerleri elde edilmekte, fakat algoritma çalışma sayısının artması ile birlikte algoritma tamamlanma süresi ortalamalarında ise artış görülmektedir.

Çizelge 4.4. GA'nın farklı iterasyon adedi değerlerine göre çalışma sonuçları

İterasyon Adedi	1000	1500	2000	2500
Ortalama Ceza Puanı	1,4	1	0,2	0,2
Algoritma Tamamlanma Süresi Ortalaması (ms)	97,6	172,9	231,9	271,3

4.2. Yapay Bağışıklık Sistemi ile Oturum Planı Hazırlama Uygulama Sonuçları

Aşağıdaki seçimler doğrultusunda Çizelge 4.5'de yer alan farklı YBS parametreleri için algoritma 10'ar kez çalıştırılarak 16 deney gerçekleştirilmiştir. Deneylerde yeni popülasyonun oluşturulmasında mutasyon yöntemi olarak “İyileştirilmiş Mutasyon” yöntemi kullanılmıştır. Her bir deney için elde edilen en iyi ceza puanlarının ve algoritma tamamlanma sürelerinin ortalaması Çizelge 4.5'de verilmiştir.

Sınıf düzeylerine göre öğrenci sayıları: 9.Sınıf:7, 10.Sınıf: 7, 11.Sınıf: 8, 12.Sınıf: 8

Sıra yapısı: 5*3 (30 oturum yeri)

İterasyon Adedi: 2000

Atık Yüzdesi: %10 şeklinde kabul edilmektedir.

Çizelge 4.5. YBS deney sonuçları (Atık yüzdesi: %10)

Parametreler	Deney1	Deney2	Deney3	Deney4	Deney5	Deney6	Deney7	Deney8
Popülasyon Büyüklüğü	10	10	15	15	10	10	15	15
Klonlama Değeri	0,7	0,7	0,7	0,7	0,9	0,9	0,9	0,9
Seçim Havuzu Yüzdesi	70	70	70	70	70	70	70	70
Seçim Yöntemi	En İyiler	Farklı En İyiler	En İyiler	Farklı En İyiler	En İyiler	Farklı En İyiler	En İyiler	Farklı En İyiler
Ortalama Ceza Puanı	7,3	6,8	4,4	3,8	7,2	5,4	3,2	2,8
Ortalama Süre (ms)	1370	1326	2802	2069	1669	1314	3286	2351
Parametreler	Deney9	Deney10	Deney11	Deney12	Deney13	Deney14	Deney15	Deney16
Popülasyon Büyüklüğü	10	10	15	15	10	10	15	15
Klonlama Değeri	0,9	0,9	0,9	0,9	0,7	0,7	0,7	0,7
Seçim Havuzu Yüzdesi	90	90	90	90	90	90	90	90
Seçim Yöntemi	En İyiler	Farklı En İyiler	En İyiler	Farklı En İyiler	En İyiler	Farklı En İyiler	En İyiler	Farklı En İyiler
Ortalama Ceza Puanı	4,2	3	1,8	1,8	5,5	3,5	3	3
Ortalama Süre (ms)	2190	1532	6151	2745	2302	1598	5146	3323

Aşağıdaki seçimler doğrultusunda Çizelge 4.6’ da yer alan farklı YBS parametreleri için algoritma 10’ar kez çalıştırılarak 16 deney gerçekleştirilmiştir. Deneylerde yeni popülasyonun oluşturulmasında mutasyon yöntemi olarak “İyileştirilmiş Mutasyon” yöntemi kullanılmıştır. Her bir deney için elde edilen en iyi ceza puanlarının ve algoritma tamamlanma sürelerinin ortalaması Çizelge 4.6’da verilmiştir.

Sınıf düzeylerine göre öğrenci sayıları: 9.Sınıf:7, 10.Sınıf: 7, 11.Sınıf: 8, 12.Sınıf: 8

Sıra yapısı: 5*3 (30 oturum yeri)

İterasyon Adedi: 2000

Atık yüzdesi: %20 şeklinde kabul edilmektedir.

Çizelge 4.6. YBS deney sonuçları (Atık yüzdesi: %20) (Vatansever ve Arıcı, 2019)

Parametreler	Deney1	Deney2	Deney3	Deney4	Deney5	Deney6	Deney7	Deney8
Popülasyon Büyüklüğü	10	10	15	15	10	10	15	15
Klonlama Değeri	0,7	0,7	0,7	0,7	0,9	0,9	0,9	0,9
Seçim Havuzu Yüzdesi	70	70	70	70	70	70	70	70
Seçim Yöntemi	En İyiler	Farklı En İyiler	En İyiler	Farklı En İyiler	En İyiler	Farklı En İyiler	En İyiler	Farklı En İyiler
Ortalama Ceza Puanı	4,6	4,2	2,2	3	9	5,2	3,2	1,8
Ortalama Süre (ms)	4231,1	1908,3	4314,6	3121	2935	2129	5845	3384,2
Parametreler	Deney9	Deney10	Deney11	Deney12	Deney13	Deney14	Deney15	Deney16
Popülasyon Büyüklüğü	10	10	15	15	10	10	15	15
Klonlama Değeri	0,9	0,9	0,9	0,9	0,7	0,7	0,7	0,7
Seçim Havuzu Yüzdesi	90	90	90	90	90	90	90	90
Seçim Yöntemi	En İyiler	Farklı En İyiler	En İyiler	Farklı En İyiler	En İyiler	Farklı En İyiler	En İyiler	Farklı En İyiler
Ortalama Ceza Puanı	2,2	2	2,6	0,8	2,2	2,4	2,8	1,6
Ortalama Süre (ms)	727	1015,4	1651,1	1623,6	878,6	729	1583	1700,4

Deneylerden elde edilen bulgular doğrultusunda aşağıdaki değerlendirmeler yapılabilir.

Atık yüzde değeri %10 ve %20 için aynı parametrelerle farklı 16 deney gerçekleştirilmiştir. Çoğunlukla atık yüzdesi olarak %20 değeri belirlenen deneylerde ortalama olarak daha kısa sürelerde daha iyi ceza puanı ortalamaları elde edilmiştir. Bu durum Çizelge 4.7’de görülmektedir. En ideal atık yüzde değeri %20 olarak belirlenmiştir.

Çizelge 4.7. YBS'nin farklı atık yüzde değerlerine göre çalışma sonuçları

Atık Yüzdesi:%20	Deney1	Deney2	Deney3	Deney4	Deney5	Deney6	Deney7	Deney8	Deney9	Deney10	Deney11	Deney12	Deney13	Deney14	Deney15	Deney16
Ortalama Ceza Puanı	4,6	4,2	2,2	3	9	5,2	3,2	1,8	2,2	2	2,6	0,8	2,2	2,4	2,8	1,6
Ortalama Süre (ms)	4231	1908	4315	3121	2935	2129	5845	3384	727	1015	1651	1624	878,6	729	1583	1700
Atık Yüzdesi:%10	Deney1	Deney2	Deney3	Deney4	Deney5	Deney6	Deney7	Deney8	Deney9	Deney10	Deney11	Deney12	Deney13	Deney14	Deney15	Deney16
Ortalama Ceza Puanı	7,3	6,8	4,4	3,8	7,2	5,4	3,2	2,8	4,2	3	1,8	1,8	5,5	3,5	3	3
Ortalama Süre (ms)	1370	1326	2802	2069	1669	1314	3286	2351	2190	1532	6151	2745	2302	1598	5146	3323

Sadece popülasyon büyüklüğü parametresinin farklı olduğu deneylere bakıldığında, çoğunlukla popülasyon büyüklüğünün 15 olarak alındığı deneylerde popülasyon büyüklüğünün 10 alarak alındığı deneylere göre çok daha iyi ortalama ceza puanları elde edilirken; popülasyon büyüklüğünün artmasıyla birlikte işlem yapılacak antikör sayısının artmasıyla paralel olarak daha uzun algoritma tamamlanma süre ortalamaları elde edilmiştir. Popülasyon büyüklüğü değiştirilerek Çizelge 4.6'da yer alan Deney12'e ait YBS parametreleri ile algoritma 10'ar kez çalıştırıldığında, Çizelge 4.8'deki bulgular elde edilmiştir. Bu bulgular doğrultusunda en ideal popülasyon büyüklüğü 15 olarak belirlenmiştir.

Çizelge 4.8. YBS'nin farklı popülasyon büyüklüğü değerlerine göre çalışma sonuçları

Popülasyon Büyüklüğü	10	15	20
Ortalama Ceza Puanı	1,8	0,8	1,2
Algoritma Tamamlanma Süresi Ortalaması (ms)	510,4	713,7	1402,2

Sadece seçim yöntemi parametresinin farklı olduğu deneylere bakıldığında, çoğunlukla “Farklı En İyiler” seçim yönteminin belirlendiği deneylerde seçim yöntemi olarak “En İyiler” yönteminin belirlendiği deneylere göre daha iyi ceza puanı ortalamaları elde edilmiştir. Bu nedenle “Farklı En İyiler” yöntemi ideal seçim yöntemi olarak belirlenmiştir.

Sadece seçim havuzu yüzdesi parametresinin farklı olduğu deneylere bakıldığında, çoğunlukla seçim havuzu yüzde değerinin 90 olduğu deneylerde seçim havuzu yüzde

değerinin 70 olarak alındığı deneylere göre daha iyi ceza puanı ortalamaları elde edilmiştir. En ideal seçim havuzu yüzde değeri 90 olarak belirlenmiştir.

Sadece klonlama değeri parametresinin farklı olduğu deneylere bakıldığında, çoğunlukla klonlama değerinin 0,9 olduğu deneylerde klonlama değerinin 0,7 olarak alındığı deneylere göre daha iyi ceza puanı ortalamaları elde edilmiştir. En ideal klonlama değeri 0,9 olarak belirlenmiştir.

Sadece Mutasyon yöntemi değiştirilerek (Bkz. Çizelge 4.6) Deney12'e ait YBS parametreleri ile algoritma 10'ar kez çalıştırıldığında, Çizelge 4.9'daki bulgular elde edilmiştir.

Çizelge 4.9. YBS'nin farklı mutasyon yöntemi seçimlerine göre çalışma sonuçları

Mutasyon Yöntemi	Rastgele Mutasyon	İyileştirilmiş Mutasyon
Ortalama Ceza Puanı	5,9	1,6
Algoritma Ortalama Tamamlanma Süresi (ms)	462	2594,5

Çizelge 4.9'da görüldüğü üzere “İyileştirilmiş Mutasyon” ile çok daha iyi ceza puanı ortalaması elde edilmekle birlikte bu mutasyon yönteminde yer alan karar ifadelerinin işlem süresini uzatması ile birlikte algoritma ortalama tamamlanma süreside uzamaktadır. Fakat algoritmanın en iyi sonucu bulması halinde iterasyon sayısı tamamlanmadan sonlandırılması sebebiyle işlem süresinin uzaması durumu en iyi çözümün bulunduğu ihtimallerde önemini kaybetmektedir.

Sadece iterasyon adedi parametresi değiştirilerek (Bkz. Çizelge 4.6) Deney12'e ait YBS parametreleri ile algoritma 10'ar kez çalıştırıldığında, Çizelge 4.10'daki bulgular elde edilmiştir. Bu bulgulardan görüldüğü üzere iterasyon adedinin artmasıyla birlikte daha iyi ortalama ceza puanı elde edilmekte, fakat algoritma çalışma sayısının artması ile birlikte algoritma ortalama tamamlanma süresi de artmaktadır.

Çizelge 4.10. YBS'nin farklı iterasyon adedi değerlerine göre çalışma sonuçları

İterasyon Adedi	1000	1500	2000	2500
Ortalama Ceza Puanı	2,6	1,8	0,8	0,8
Algoritma Ortalama Tamamlanma Süresi (ms)	1006,3	1282,9	1382,7	1688

4.3. Genetik Algoritma ve Yapay Bağışıklık Sistemi Uygulanan Yöntemlerin Karşılaştırılması

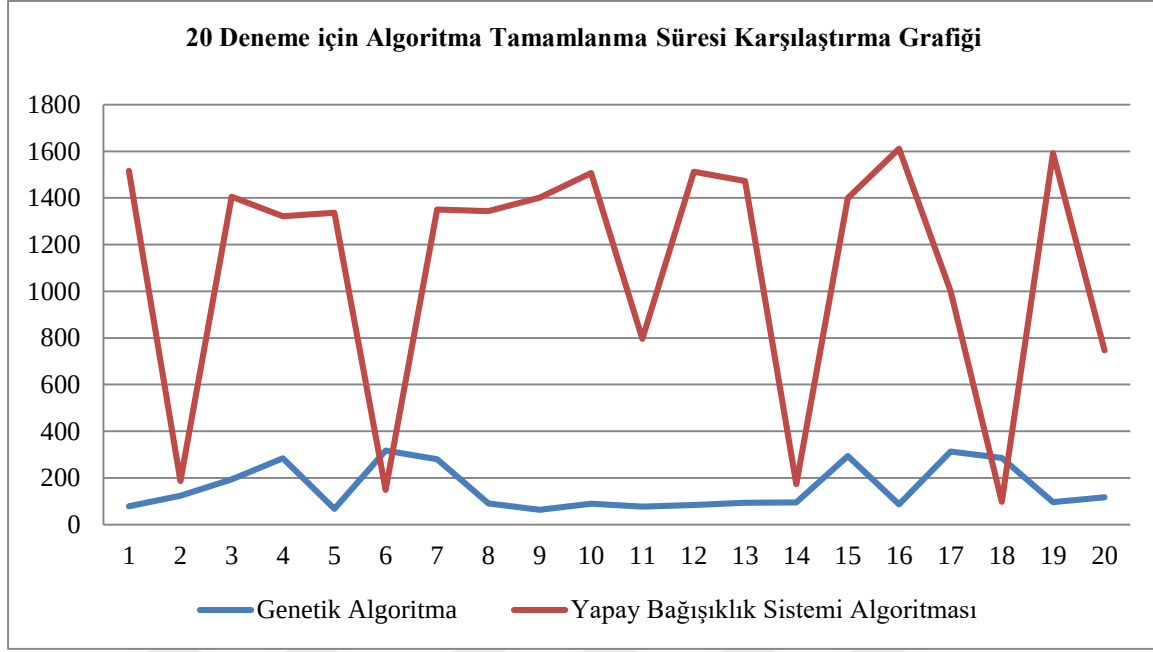
GA ve YBS uygulan iki yöntemi karşılaştırmak için sınıflardaki sıraların yerleşim şeklinin farklı seçimleri ile 3 farklı deney gerçekleştirilmiştir. Deneylerde algoritmalar 20'şer kez çalıştırılmış ve sonuçların ortalaması alınmıştır. Popülasyon büyüklüğü 15, toplam sınıf adedi 10 ve iterasyon adedi 2000 olarak belirlenmiştir. Her iki algortmada da algoritmalara ait diğer parametrelerin değerleri algoritmalar için yapılan deneyler sonucunda en iyi ceza puanı ortalamalarının elde edildiği deneylerde ki değerlere göre belirlenmiştir. GA için (Bkz. Çizelge 4.2) Deney14' e ait değerler; YBS için (Bkz. Çizelge 4.6) Deney 12' e ait değerler kullanılmıştır.

Deney 1: Sınava girecek öğrenci sayıları 9.Sınıf:70, 10.Sınıf: 70, 11.Sınıf: 80, 12.Sınıf: 80 ve sıra yapısı 5*3 (30 oturum yeri) şeklinde kabul edilmektedir. Elde edilen sonuçlar Çizelge 4.11'de verilmiştir.

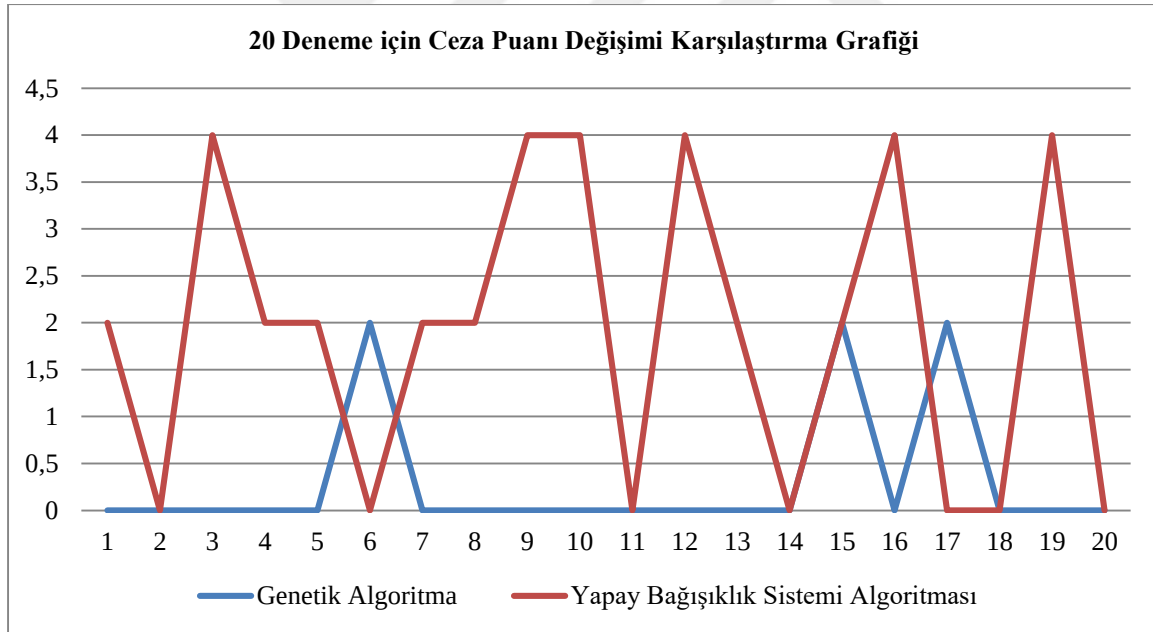
Çizelge 4.11. Algoritma karşılaştırma deney sonuçları (Oturum şekli 5*3)

Parametreler ve Sonuçlar	Genetik Algoritma	Yapay Bağışıklık Sistemi Algoritması
Popülasyon Büyüklüğü	15	15
İterasyon Adedi	2000	2000
Minumum Ceza Puanı	0	0
Maksimum Ceza Puanı	2	4
Ortalama Ceza Puanı	0,3	1,9
Algoritma Ortalama Tamamlanma Süresi (ms)	156,6	1095,95

20 deneme için algoritmaların karşılaştırmalı tamamlanma süreleri Şekil 4.1'de, 20 deneme için algoritmaların karşılaştırmalı ceza puanları sonuçları da Şekil 4.2'de verilmiştir.



Şekil 4.1. 20 deneme için algoritma tamamlanma süresi karşılaştırma grafiği (5*3)



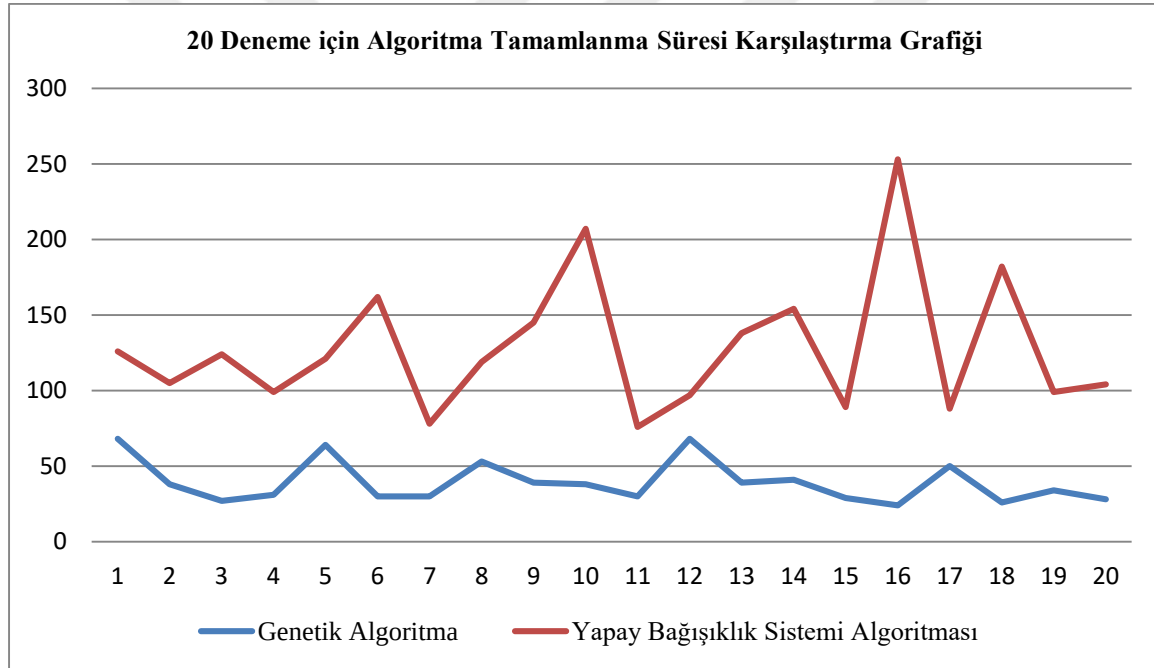
Şekil 4.2. 20 deneme için ceza puanı değişimi karşılaştırma grafiği (5*3)

Deney 2: Sınava girecek öğrenci sayıları 9.Sınıf:80, 10.Sınıf: 80, 11.Sınıf: 80, 12.Sınıf: 80 ve sıra yapısı 4*4 (32 oturum yeri) şeklinde kabul edilmektedir. Elde edilen sonuçlar Çizelge 4.12’de verilmiştir.

Çizelge 4.12. Algoritma karşılaştırma deney sonuçları (Oturma şekli 4*4)

Parametreler ve Sonuçlar	Genetik Algoritma	Yapay Bağışıklık Sistemi Algoritması
Popülasyon Büyüklüğü	15	15
İterasyon Adedi	2000	2000
Minumum Ceza Puanı	0	0
Maksimum Ceza Puanı	0	0
Ortalama Ceza Puanı	0	0
Algoritma Ortalama Tamamlanma Süresi (ms)	39,35	128,3

20 deneme için algoritmaların karşılaştırmalı tamamlanma süreleri Şekil 4.3’de verilmiştir. Deney2’de hem GA hem de YBS’ye ait tüm denemelerde ceza puanı olarak 0 değeri elde edildiği için algoritmaların ceza puanı değişim grafiğine yer verilmemiştir.



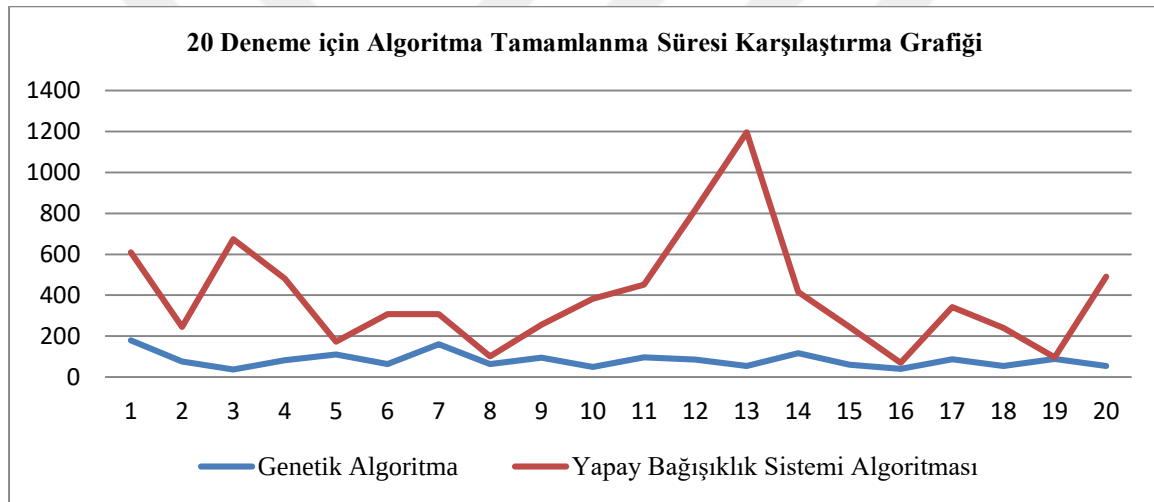
Şekil 4.3. 20 deneme için algoritma tamamlanma süresi karşılaştırma grafiği (4*4)

Deney 3: Sınava girecek öğrenci sayıları 9.Sınıf: 90, 10.Sınıf: 90, 11.Sınıf: 90, 12.Sınıf: 90 ve sıra yapısı 6*3 (36 oturma yeri) şeklinde kabul edilmektedir. Elde edilen sonuçlar Çizelge 4.13’de verilmiştir.

Çizelge 4.13. Algoritma karşılaştırma deney sonuçları (Oturum şekli 6*3)

Parametreler ve Sonuçlar	Genetik Algoritma	Yapay Bağışıklık Sistemi Algoritması
Popülasyon Büyüklüğü	15	15
İterasyon Adedi	2000	2000
Minumum Ceza Puanı	0	0
Maksimum Ceza Puanı	0	0
Ortalama Ceza Puanı	0	0
Algoritma Ortalama Tamamlanma Süresi(ms)	83	395,25

20 deneme için algoritmaların karşılaştırmalı tamamlanma süreleri Şekil 4.4’de verilmiştir. Deney3’te hem GA hem de YBS’ye ait tüm denemelerde ceza puanı olarak 0 değeri elde edildiği için algoritmaların ceza puanı değişim grafiğine yer verilmemiştir.



Şekil 4.4. 20 deneme için algoritma tamamlanma süresi karşılaştırma grafiği (6*3)

GA ve YBS için yapılan karşılaştırma deneyleri sonucunda aşağıdaki değerlendirmeler yapılabilmektedir.

1. Deneylerde GA’da YBS’ye göre daha kısa çalışma sürelerinde daha iyi ceza puanları elde edilmiştir.
2. Deneyler sonucu elde edilen grafikler incelendiğinde; GA’nın gerek algoritma tamamlanma süreleri, gerekse ceza puanları için YBS’ye göre daha kararlı bir durum sergilediği görülmektedir.

Bu değerlendirmeler ışığında, en iyi oturum planının hazırlanmasında başarılı olan ve GENÇİZ yazılımında kullanılacak yöntem GA olarak belirlenmiştir.

5. SONUÇ VE ÖNERİLER

Ortaöğretim kurumlarında ölçme ve değerlendirme süreci, öğrencilerin üst öğrenime devamı ve öğrenme sürecinin yeniden düzenlenmesi gibi pek çok önemli kararı etkilemektedir. Bu kararların doğruluğu ve etkinliği sürecin doğru ve güvenli işleyişine bağlıdır. Bu süreçte kullanılan en önemli araçlarından biri sınavlardır.

Ortaöğretim kurumlarında gerçekleştirilen ortak sınavlar için uygun sınav çizelgelerinin insan çabası ile hazırlanması oldukça zahmetli ve zaman alıcıdır. Bu tez çalışmasında; ortaöğretim kurumları ortak sınavları için en uygun sınav oturum planlarının hazırlanması başta olmak üzere tüm ortak sınav çizelgeleme aşamaları için hızlı ve kolay kullanım sağlayan GENÇİZ yazılımı geliştirilmiştir.

Ölçme ve değerlendirme sürecinin doğru ve güvenli işleyişi adına oturum planlarının öğrencilerin istenmeyen yardımlaşmasının önüne geçecek şekilde hazırlanması oldukça önemlidir. Oturum planlarında mümkün olduğunca aynı sınıf seviyesinde yer alan öğrencilerin çaprazda, yan yana ya da önlü arkalı gelmeyecek şekilde yerleşiminin yapılması gözetilmiştir. Tez çalışmasında oturum planlarının hazırlanması için çözüm yaklaşımı olarak GA ve YBS kullanılarak iki farklı sezgisel yöntem uygulanmış ve denenmiştir.

Her iki yöntem için kendi içlerinde ve karşılaştırmalı olarak uygulama deneyleri gerçekleştirilmiştir. Deney sonuçları grafikler ve çizelgeler ile özetlenmiştir.

Öncelikle farklı parametreler ile algoritmaların çalışma performansları değerlendirilmiştir. GA yöntemi için deneylerde elitizm oranı, popülasyon büyüklüğü, çaprazlama oranı, mutasyon oranı, seçim yöntemi ve iterasyon sayısı gibi farklı parametreler kullanılmıştır. YBS yöntemi için deneylerde atık yüzdesi, popülasyon büyüklüğü, klonlama değeri, seçim havuzu yüzdesi, mutasyon yöntemi, seçim yöntemi ve iterasyon sayısı gibi farklı parametreler kullanılmıştır.

Deneylerden elde edilen bulgulara göre GA parametreleri için en iyi değerler; popülasyon büyüklüğü için 15, çaprazlama oranı için 0,9 ve mutasyon oranı için ise 0,03 olarak belirlenmiştir. Seçim yöntemi olarak “Turnuva” yönteminde en iyi sonuçlar elde edilmiştir.

Çaprazlama sonrası kromozom yapıların bozulmasından dolayı tamir fonksiyonu kullanılması gerekmiştir. Yeni popülasyonun oluşturulmasında elitizm operatörü kullanılarak eski popülasyonda ki en iyi kromozomların yok olması önlenmiştir. Elitizm oranı için en iyi değer olarak %10 değeri belirlenmiştir. Algoritma uygulama aşamasında, popülasyon çeşitliliğinin azalmasıyla birlikte kimi zaman algoritmanın istenmeyen bir çözümde yakınsadığı görülmüştür. Bu sebeple her 13 iterasyonda bir, tek bir iterasyon için mutasyon oranı değeri olarak 1 değeri alınmıştır.

YBS yönteminde CLONAG isimli KSA temel alınmıştır. Deneylerden elde edilen bulgulara göre YBS parametreleri için en iyi değerler; popülasyon büyüklüğü için 15, seçim havuzu yüzdesi için %90, klonlama değeri için 0,9 ve atık yüzde değeri için %20 değeri olarak belirlenmiştir. Seçim havuzuna alınan antikörlerin hangi sayılarda klonlanacağını belirlemede antikörlerin uygunluk değerinin dikkate alındığı özel bir eşitlik (Bkz. Eş. 3.13) kullanılmıştır. Klonlanan antikörlerin mutasyon oranlarının belirlenmesinde ise yine antikörlerin uygunluk değerinin dikkate alındığı özel bir eşitlik (Bkz. Eş. 3.14) kullanılmıştır. En iyi mutasyon yöntemi olarak “İyileştirilmiş Mutasyon” ve yeni popülasyonun oluşturulmasında en iyi seçim yöntemi olarak “Farklı En İyiler” yöntemi belirlenmiştir.

Her iki algoritmada da aynı zamanda uygunluk fonksiyonu olan aynı amaç fonksiyonu (Bkz. Eş. 3.9) kullanılmıştır. Amaç fonksiyonu oturum planında istenmeyen yerleşimlere ceza puanı verilerek oluşturulmuş ve sistemi minimize etmek amaçlanmıştır.

Performans değerlendirmelerinde belirlenen en iyi parametre değerleri kullanılarak, algoritmalar için karşılaştırma deneyleri gerçekleştirilmiştir. Bu deneyler sonucunda GA yöntemi ile YBS yöntemine göre daha iyi çözümlerin daha kısa sürede bulunduğu gözlemlenmiştir. Bu sebeple GENÇİZ yazılımında oturum planların hazırlanması aşaması için GA yöntemi tercih edilmiştir.

GENÇİZ yazılımı ortak sınav çizelgeleme probleminin tüm aşamalarında kullanılabilecek şekilde hazırlanmıştır. Yazılımının çalışması farklı sınav sayıları, farklı sınav tarihleri, farklı sınıf yapıları ve farklı öğrenci sayıları ile oluşturulan pek çok sınav senaryosu ile denenmiş ve başarılı sonuçlar elde edilmiştir. Yazılım Çankırı’da bir ortaöğretim kurumunda 2018-2019 eğitim-öğretim yılının ikinci döneminin ikinci ortak sınavlarında

oturum planlarının belirlenmesi aşamasında kullanılmış ve herhangi bir sıkıntı olmadan başarılı bir şekilde sınavlar gerçekleştirilmiştir. Sınavların gerçekleştirilmesinden sorumlu olan idareciler hızlı ve kolay bir şekilde sınav çizelgeleri hazırladıklarını ve GENÇİZ yazılımın eski kullandıkları yazılıma göre daha iyi performans ortaya koyduğunu belirtmişlerdir.

Gelecekte, GENÇİZ yazılımına sınav oturum planları doğrultusunda öğrenci yerleşimlerine göre öğrenciye özel sınav kağıtları raporlayan bir menü ve çalışma ekranı ile öğrenciler için girecekleri tüm sınavları, tarihlerini ve sınav yerlerini gösteren raporların alınacağı bir menü ve çalışma ekranı eklenebilir.





KAYNAKLAR

- Ağalday M. F. (2018). *Genetik Algoritma ile Merkezi Sınavlarda Tek ve Çok Boyutlu Yakınlığa Göre En İyi Oturum Planının Oluşturulması*, Yüksek Lisans Tezi, Fatih Sultan Mehmet Vakıf Üniversitesi Mühendislik ve Fen Bilimleri Enstitüsü, İstanbul, 1-67.
- Ahmad I. R., Sufahani S., Ali M. and Razali S. N. (2017). A heuristics approach for classroom scheduling using genetic algorithm technique. *Journal of Science and Technology*, 9(3), 10-15.
- Aksu O.R. (2018). *Gözetmen Atama Problemine Yönelik Karşılaştırmalı Bir Matematiksel Optimizasyon ve Genetik Algoritma Yaklaşımı: Bir Üniversite Uygulaması*, Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, 1-29.
- Alaca Y. (2018). *Yapay Bağışıklık Sistemleri ile Bilgi Güvenliği ve Olay Yönetimi Geliştirilmesi*, Yüksek Lisans Tezi, Karabük Üniversitesi Fen Bilimleri Enstitüsü, Karabük, 42.
- Aladağ A. (2010). *Tekrar İşlemeli Esnek Atölye Tipi Çizelgeleme Problemi için Yapay Bağışıklık Sistemi ile Bir Çözüm Yaklaşımı*, Yüksek Lisans Tezi, Eskişehir Osmangazi Üniversitesi Fen Bilimleri Enstitüsü, Eskişehir, 52.
- Altıntaş C. (2011). *Sezgisel Algoritmalarla Sınav Çizelgeleme Problemi Çözümü*, Yüksek Lisans Tezi, Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü, Isparta, 1-70.
- Arogundade O. T., Akinwale A. T. and Aweda O. M. (2010, December). A genetic algorithm approach for a real-world university examination timetabling problem, *International Journal of Computer Applications*, 12(5), 1-4.
- Atay Y. (2012). *Yapay Bağışıklık Sistemleri ile Atölye Çizelgeleme Problemlerinin Optimizasyonu*, Yüksek Lisans Tezi, Selçuk Üniversitesi Fen Bilimleri Enstitüsü, Konya, 1.
- Ayob M. and Malik A. (2011, October). A new model for an examination-room assignment problem. *IJCSNS International Journal of Computer Science and Network Security*, 11(10), 187-190.
- Babayiğit B. ve Güney K., (2006, 6-10 Aralık). *Klonal seçme algoritması kullanılarak genlik ve faz kontrolü ile doğrusal anten dizi diyagramında sıfırların üretilmesi*. Elektrik-Elektronik-Bilgisayar Mühendisliği Sempozyumunda (ELECO'2006) sunuldu, Bursa.
- Baykasoğlu A., Saltabaş A., Taşan A. S. ve K. Subulan (2012, Ocak). Yapay bağışıklık sisteminin çoklu etmen benzetim ortamında realize edilmesi ve gezgin satıcı problemine uygulanması. *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, 27(4), 901-909.

- Baysal C. (2011). *Evrimsel Hesaplama Tekniği Kullanılarak Otomatik Sınav Programı Oluşturma*, Yüksek Lisans Tezi, Pamukkale Üniversitesi Fen Bilimleri Enstitüsü, Denizli, 1-54.
- Bingül Z., Sekmen A. S., Palaniappan S. and Sabatto S. (2000, 9 April). *Genetic algorithms applied to real time multiobjective optimization problems*. Paper presented at the IEEE SoutheastCon 2000, Nashville, Tennessee, United States of America.
- Bulut F. ve Subaşı S. (2015, Eylül). Merkezi sınavlar için genetik algoritmalar ile en iyi oturma planı. *Dokuz Eylül Üniversitesi Mühendislik Fakültesi Dergisi*, 17(3), 122-137.
- Chaki P. K. and Anirban S. (2016, 11-13 March). *Algorithm for efficient seating plan for centralized exam system*. Paper presented at the 2016 International Conference on Computational Techniques in Information and Communication Technologies (ICCTICT), New Delhi, India.
- Coşkun A. (2006). *Genetik Algoritma Kullanılarak Kimyasal Maddelerin Deriden Geçiş Katsayılarının ve Molekül Yapılarının Bulunması*, Doktora Tezi, Gazi Üniversitesi Fen Bilimleri Enstitüsü, Ankara, 35, 42.
- Coşkun A. (2007). Yapay zeka optimizasyon teknikleri: literatür değerlendirmesi. *Fırat Üniversitesi Doğu Anadolu Bölgesi Araştırmaları Dergisi*, 5(2), 142-146.
- Çolak R. (2015). *Sezgisel Algoritmalarla Ders Programı Çizelgeleme Problemi Çözümü*, Yüksek Lisans Tezi, Süleyman Demirel Üniversitesi, Fen Bilimleri Enstitüsü, Isparta, 8.
- Dasgupta D. (2006, November). Advances in artificial immune systems. *IEEE Computational Intelligence Magazine*, 1(4), 40-49.
- Dasgupta D., Yu S. and Nino F. (2011). Recent advances in artificial immune systems: models and applications. *Applied Soft Computing*, 11(2), 1574-1587.
- De Castro L. N. and Timmis J. (2002, 12-17 May). *An artificial immune network for multimodal function optimization*. Paper presented at the IEEE Congress on Evolutionary Computation (CEC'02), Honolulu, Hawaii, United States of America.
- De Castro L. N. and Timmis J. (2003, August). Artificial immune systems as a novel soft computing paradigm. *Soft Computing*, 7(8), 526-544.
- De Castro L. N. and Von Zuben F. J. (2002). Learning and optimization using the clonal selection principle. *IEEE Transactions on Evolutionary Computation*, 6(3), 239–251.
- Dener M. and Calp M. H. (2018). Solving the exam scheduling problems in central exams with genetic algorithms. *Mugla Journal of Science and Technology*, 4(1), 102-115.

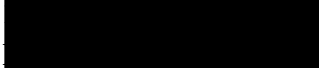
- East A. R. (2019). *Timetable Scheduling via Genetic Algorithm*, Final Year Thesis, National University of Ireland Computer Science and Information Technology, Galway, 7.
- Elmas Ç. (2018). *Yapay Zeka Uygulamaları* (Dördüncü Baskı). Ankara: Seçkin Yayınevi, 443, 446.
- Erden C., Kökçam A. and Demir H. (2016). A two phase modeling with genetic algorithm for invigilator timetabling problem. *Uluslararası Mühendislik ve Teknoloji Araştırmaları Dergisi*, 1(1), 13-22.
- Erdős S. and Kövári B. (2019, 23-24 May), *Genetic algorithm based solution for final exam scheduling*. Paper presented at the MultiScience - XXXIII. microCAD International Multidisciplinary Scientific Conference, Miskolc, Macaristan
- Forrest S. and Perelson A. S. (1991). *Genetic algorithms and the immune system*. Paper presented at the Parallel Problem Solving from Nature, Heidelberg, Berlin.
- Gashgari R., Alhashimi L., Obaid R., Palaniswamy T., Aljawi L. and Alamoudi A. (2018, 4-6 April). *A survey on exam scheduling techniques*. Paper presented at the 2018 1st International Conference on Computer Applications & Information Security (ICCAIS), Riyadh, Saudi Arabia.
- Goldberg D. E. (1989). *Genetic Algorithms in Search Optimization and Machine Learning*. Aktaran: Altıntaş C. (2011). *Sezgisel Algoritmalarla Sınav Çizelgeleme Problemi Çözümü*, Yüksek Lisans Tezi, Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü, Isparta, 8.
- Goli A., Tirkolaee E. B., Mahdavi I. and Zamani M. (2019, 23-26 July). *Solving a university exam scheduling problem using genetic and firefly algorithms*. Paper presented at Proceedings of the International Conference on Industrial Engineering and Operations Management, Pilsen, Czech Republic.
- Gonsalves T. and Oishi R. (2015, July). Artificial immune algorithm for exams timetable. *Journal of Information Sciences and Computing Technologies (JISCT)*, 4(2), 287-296.
- Hassan O. A. H., Qtaish O., Abuhamdeh M. and Hassan M. A. H. (2019), Hybrid Exam Scheduling Technique based on Graph Coloring and Genetic Algorithms Targeted Towards Student Comfort. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 10(3), 503-512
- İnternet: Çunkaş M. (2006). Genetik Algoritmalar ve Uygulamaları Ders Notları. URL: <https://docplayer.biz.tr/14536665-Yrd-doc-dr-mehmet-cunkas.html>, Son Erişim Tarihi: 07.10.2019.
- İnternet: De Castro L. N. and Von Zuben F. J. (1999). Artificial Immune Systems: Part I- Basic Theory And Applications (Technical Report TR-DCA 01/99). URL: ftp://ftp.dca.fee.unicamp.br/pub/docs/vonzuben/tr_dca/trdca0199.pdf, Son Erişim Tarihi: 07.10.2019.

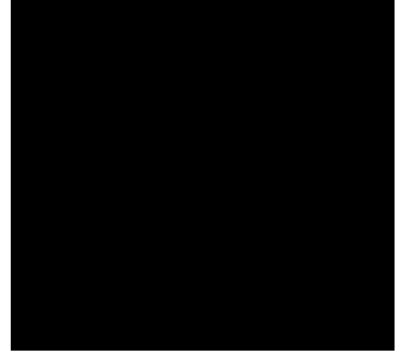
- Jha S. K. (2014). Exam timetabling problem using genetic algorithm. *International Journal of Research in Engineering and Technology*, 3(5), 649-654.
- Kahar M. N. M. and Kendall G. (2010, December). The examination timetabling problem at universiti malaysia pahang: comparison of a constructive heuristic with an existing software solution. *European Journal of Operational Research*, 207(2), 557-565.
- Kalaycı C. B. (2008). *Öğrenci Başarısına Odaklı Sınav Çizelgeleme Modeli Ve Yazılım Uygulaması*, Yüksek Lisans Tezi, Pamukkale Üniversitesi Fen Bilimleri Enstitüsü, Denizli, 1-117.
- Karaboğa D. (2018). *Yapay Zeka Optimizasyon Algoritmaları* (Altıncı Baskı). Ankara: Nobel Yayınevi, V, 79, 86-87.
- Koca A. (2018). *Sınav Çizelgeleme Problemi Çözümü İçin Farklı Yaklaşımlar*, Yüksek Lisans Tezi, Necmettin Erbakan Üniversitesi Fen Bilimleri Enstitüsü, Konya, 1-48.
- Malim M. R., Khader A. T. and Mustafa A. (2006, 30 August - 1 September). *Artificial immune algorithms for university timetabling*. Paper presented at the 6th International Conference on Practice and Theory of Automated Timetabling, Brno, Czech Republic.
- Milli Eğitim Bakanlığı Ortaöğretim Kurumları Yönetmeliği (2013). *T.C. Resmi Gazete* (28758, 7 Eylül 2013).
- Mitchell M. (1996). *An Introduction to Genetic Algorithms* (Fifth Edition). England: MIT Pres, 2-3.
- Nabiyev V.V. (2016). *Yapay Zeka* (Beşinci Baskı). Ankara: Seçkin Yayınevi, 617-620, 624.
- Özdamar S. (2015). *Sınav Çizelgeleme, Sınav Yeri ve Gözetmen Atama Problemlerinin Çözümü İçin Model Önerileri ve Uygulaması*, Yüksek Lisans Tezi, Kırıkkale Üniversitesi Fen Bilimleri Enstitüsü, Kırıkkale, 1-102.
- Özşen S. (2008). *Biyomedikal Sınıflama Problemleri için Problem-Tabanlı Bir Yapay Bağışıklık Sisteminin Geliştirilmesi ve Biyomedikal Sınıflama Problemlerine Uygulanması*, Doktora Tezi, Selçuk Üniversitesi Fen Bilimleri Enstitüsü, Konya, 26.
- Özşen S. (2009, Ağustos). Yapay zekâda bir adım daha yapay bağışıklık sistemleri. *Tübitak Bilim ve Teknik Dergisi*, 501, 78-83.
- Pillay N. and Banzhaf W. (2010). An informed genetic algorithm for the examination timetabling problem. *Applied Soft Computing*, 10(2), 457-467.
- Sastry K., Goldberg D. and Kendall G. (2005). Genetic algorithms. In E. K. Burke and G. Kendall (Eds), *Search methodologies: introductory tutorials in optimization and decision support techniques*. Berlin: Springer, pp. 97-125.

- Satman M. H. (2016). *Genetik Algoritmalar* (Birinci Baskı). İstanbul: Türkmen Kitabevi, 61, 109, 111-112, 136.
- Seyfi G. (2018). *Metasezgisel Algoritmalar Kullanılarak Sınav Çizelgeleme*, Yüksek Lisans Tezi, Selçuk Üniversitesi Fen Bilimleri Enstitüsü, Konya, 1-76.
- Suvay H. (2019). *Sınav Çizelgeleme Probleminin Çözümü için Genetik Algoritma Yaklaşımı*, Yüksek Lisans Tezi, Sivas Cumhuriyet Üniversitesi Sosyal Bilimler Enstitüsü, Sivas, 1-67.
- Şahan S. (2004). *Yapay Bağışıklıkta Yeni Algoritmalar*, Yüksek Lisans Tezi, Selçuk Üniversitesi Fen Bilimleri Enstitüsü, Konya, 29-30, 40.
- Tekindal S. (2011). *Eğitimde Ölçme ve Değerlendirme* (Üçüncü Baskı). Ankara: Pegem Akademi Yayınevi, 2-7.
- Topal M. (2013). *Bulanık Mantık Yöntemi ile Ortak Sınav Otomasyon Programı (Karma)*, Yüksek Lisans Tezi, Marmara Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, 1-78.
- Vatansever S. ve Arıcı N. (2019). Yapay bağışıklık sistemi ile ortaöğretim kurumları ortak sınavlarında en iyi sınav oturum planının oluşturulması. *Artificial Intelligence Studies*, 2(1), 1-14.
- Yıldırım S. (2013). *Işık Üniversitesi Final Sınav Programı için Genetik Algoritma Uygulaması*, Yüksek Lisans Tezi, Işık Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, 1-31.
- Yıldırım S. (2014). *Öğretme Ve Öğrenme Tabanlı Optimizasyon Algoritması ile Sınav Çizelgeleme Problemlerinin Çözümü*, Yüksek Lisans Tezi, Dokuz Eylül Üniversitesi Sosyal Bilimler Enstitüsü, İzmir, 1-70.

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı: : VATANSEVER, Seval
 Uyuğu : T.C.
 Doğum tarihi ve yeri : 
 Medeni hali : 
 e-mail : 



Eğitim

Derece	Eğitim Birimi	Mezuniyet Tarihi
Yüksek Lisans	Gazi Üniversitesi / Elektronik Bilgisayar Eğitimi	Devam ediyor
Lisans	Gazi Üniversitesi / Bilgisayar Eğitimi	2003
Lise	Sincan Süper Lisesi	1999

İş Deneyimi

Yıl	Yer	Görev
2008-Halen	Çankırı Mesleki ve Teknik Anadolu Lisesi	Öğretmen
2003-2008	Fatih Sultan Mehmet Mesleki ve Teknik Anadolu Lisesi	Öğretmen

Yabancı Dil

İngilizce

Yayınlar

Vatansever S. ve Arıcı N. (2019). Yapay bağışıklık sistemi ile ortaöğretim kurumları ortak sınavlarında en iyi sınav oturum planının oluşturulması. *Artificial Intelligence Studies*, 2(1), 1-14.

Hobiler

Yüzme, Kitap Okuma



GAZİLİ OLMAK AYRICALIKTIR..