



# PANDEMİ KOŞULLARI ALTINDA DERS PROGRAMI ÇİZELGELEME PROBLEMİNİN GENETİK ALGORİTMA İLE ÇÖZÜMÜ: BİR UYGULAMA

TUĞÇE NUR YILMAZ<sup>1\*</sup> & BİRSEN İREM KUVVETLİ<sup>2</sup>

<sup>1</sup> Arş. Gör., Erciyes Üniversitesi, İktisadi ve İdari Bilimler Fakültesi, İşletme Bölümü, tugcenur@erciyes.edu.tr, <https://orcid.org/0000-0002-9323-0379>. <sup>2</sup> Dr. Öğr. Üyesi, Osmaniye Korkut Ata Üniversitesi, İktisadi ve İdari Bilimler Fakültesi, İşletme Bölümü, birsenselamoglu@osmaniye.edu.tr, <https://orcid.org/0000-0002-7730-098X>

Bu çalışma "Ders Programı Çizelgeleme Probleminin Genetik Algoritma ile Çözümü: Bir Uygulama" isimli yüksek lisans tezinden üretilmiştir.

## ÖZ

Birçok yöntemden yararlanılarak çözülen ders programı çizelgeleme probleminin tek bir çözüm yöntemi yoktur. Bu çalışmada, ders programı çizelgeleme problemine Genetik Algoritma ile bir çözüm aranmıştır. Osmaniye Korkut Ata Üniversitesi, İşletme Bölümü için matematiksel model oluşturulmuş ve Genetik Algoritma ile çözülmüştür. Bu çalışmanın amacı, herhangi bir salgın sonucunda öğrenciler ve öğretim üyeleri arasında gerçekleşebilecek bulaşın önüne geçilmesini sağlamak adına temasın azaltılması, bina içi dolaşımın azaltılması gibi durumları göz önünde bulundurarak ders programı çizelgesini oluşturmaktır. Bu örnek problem için tek noktalı çaprazlama kullanıldığında, popülasyon büyüklüğü 50 iken 0.8 çaprazlama oranı ve 0.05 mutasyon oranı ile optimal sonuç 23 saniyede bulunmuştur. Aynı problem için sıralı çaprazlama kullanıldığında ise, yine popülasyon büyüklüğü 50 iken 0.8 çaprazlama oranı ve 0.05 mutasyon oranı ile optimal sonuç 60 saniyede bulunmuştur. Problem, iki farklı çaprazlama yöntemi ile 100 birimlik popülasyon için de çözülmüş ve sonuçlar tartışılmıştır. Ayrıca senaryo analizinde çevrimiçi ders yüzdelerindeki değişime göre sonuçlar değerlendirilmiştir.

**Anahtar Kelimeler:** Sezgisel Algoritmalar, Genetik Algoritma, Ders Programı Çizelgeleme Problemi, Pandemi.

### Editör / Editor:

Ahmet HASKÖSE,  
Erciyes Üniversitesi, Türkiye

### \*Sorumlu Yazar/ Corresponding Author:

Tuğçe Nur YILMAZ,  
tugcenur@erciyes.edu.tr

### JEL:

C44, C49, C61

**Geliş:** 18 Mayıs 2024

**Received:** May 18, 2024

**Kabul:** 25 Eylül 2024

**Accepted:** September 25, 2024

**Yayın:** 30 Aralık 2024

**Published:** December 30, 2024

### Atıf / Cited as (APA):

Yılmaz, T. N. & Kuvvetli, B. İ. (2024),  
Pandemi Koşulları Altında Ders Programı  
Çizelgeleme Probleminin Genetik Algoritma  
ile Çözümü: Bir Uygulama, Erciyes Üniversitesi  
İktisadi ve İdari Bilimler Fakültesi Dergisi, 69,  
79-94,

doi: 10.18070/erciyesiibd.1486042

## SOLVING THE COURSE SCHEDULING PROBLEM UNDER THE PANDEMIC CONDITIONS WITH GENETIC ALGORITHM: AN APPLICATION

### ABSTRACT

There is no single solution to the course scheduling problem, which is solved using various methods. In this study, a solution to the course scheduling problem was sought with a Genetic Algorithm. A mathematical model of the problem for Osmaniye Korkut Ata University, Department of Business Administration and was solved with the Genetic Algorithm. The purpose of this study is creating a course schedule by taking into account situations such as reducing contact and reducing indoor circulation to prevent contamination between students and faculty members as a result of any epidemic. For this problem, when single-point crossover was used, the optimal result was found in 23 seconds with a crossover rate of 0.8 and a mutation rate of 0.05 when the population size was 50. When sequential crossover was used for the same problem, the optimal result was found in 60 seconds with a crossover rate of 0.8 and a mutation rate of 0.05 when the population size was 50. The problem was also solved for a population of 100 units with two different crossover methods and the results were discussed. In addition, the results were evaluated according to the change in online course percentages in the scenario analysis.

**Keywords:** Heuristics, Genetic Algorithm, Course Scheduling, Pandemic

## GİRİŞ

Günlük yaşantılarımızın birçok opsiyon ile bölümlenmesi, zamanı etkin ve verimli kullanmamızı çoğu kez engellemektedir. Bu durum söz konusu kurum ve kuruluşlar olduğunda ise daha karmaşık bir hal almaktadır. Girdi ile çıktı arasındaki süreyi minimize etmek durumunda olan kurum ve kuruluşlar; yıllık, aylık, haftalık ve günlük çizelgelerden yararlanmaktadır. Bu noktada kullanılan çizelgeler 'zaman çizelgeleme' olarak karşımıza çıkmaktadır.

Zaman çizelgeleme en genel tanımla kısıtları himaye ederek, kıt ve kısıtlı kaynakların en uygun ve olurlu göreve en uygun ve olurlu zaman dilimine atanması ile ilgili problemlerle ilgilenmektedir. 1950'li yıllardan itibaren araştırmacıların perspektifine takılan zaman çizelgeleme problemleri her ülke, şehir ve kuruma göre değişkenlik göstermesiyle birlikte belirli yahut net bir analitik ya da sezgisel çözüm kalıbına oturtulamamıştır. Bunun nedeni tek bir çözümün bütün koşulların aynı olması temelinde dayanmasına rağmen söz konusu netliğin dünya çapında yahut ülke çapında dahi sağlanamamasıdır.

Bütün kurum ve kuruluşlarda etkin zaman yönetimi için kullanılan çizelgeleme günlük, haftalık ya da aylık olarak planlanabilir. Örneğin özel güvenlik olarak varlığını yürüten firmaların vardiya planlaması, okul servislerinin hangi zaman diliminde hangi okulda bulunması hatta taksilerin yolcularını hangi güzergahtan alıp hangi güzergâhlara bırakabileceği dahi zaman çizelgelemesi problemidir. Bir başka deyişle kısıtları himaye etmek koşuluyla, kısıtlı kaynakların en uygun görev ve zaman dilimine tayin edilmesidir. Bunların yanı sıra çizelgeleme; bir ya da daha fazla problemin optimizasyonunu hedefler (Pinedo, 2008; Wren, 1995). Çoğu zaman büyük örneklerle çalışılan zaman çizelgelemenin zorlaşması nedeniyle kısıtları karşılayan optimal ya da optimale yakın çözümler üretmek için verimli arama yöntemleri kullanılmaktadır (Pongcharoen, Promtet, Yenradee, & Hicks, 2008).

Son çeyrek asırdaki zaman çizelgelemesi problemleri çözüm ve önerilerine bakıldığında en çok rağbet gören noktanın eğitimsel zaman çizelgeleme problemleri olduğu söylenebilir. Eğitimsel zaman çizelgelemesi problemi (EZÇP) zaman çizelgeleme problemleri alanında sürekli üzerinde çalışılan bir konu ve problemidir (Kamışlı Öztürk, Kasımbeyli, Özdemir, Acar, & Özçetin, 2015). Eğitim kurumları ve özellikle üniversitelerde bu durum probleme dönüşmektedir. Çok sayıda bölüm, ders, öğretim üyesi ve sınıf içeren bu problemi elde hazırlanan bir çizelgeyle çözmek zaman ve emek zayıfatı olacaktır. Problemin girdi boyutu büyüdükçe çözüm uzayı da buna paralel olarak büyümektedir. Tüm bileşenleri (derslik, öğretim üyesi...) aynı anda ele almak bir başka deyişle problemin asıl amacı haline getirmek çözümü olanaksız hale getireceğinden birkaçı ya da bir tanesi üzerinden maksimum fayda sağlanmaya çalışılır. Yapılan çalışmalarda artık çözümün bulunması yeterli olmamakla birlikte sonuçta ulaşmanın aldığı zaman yani çözüm süresi büyük önem arz etmektedir. Bu çalışmanın amacı, herhangi bir salgın sonucunda öğrenciler ve öğretim üyeleri arasında gerçekleşebilecek bulaşın önüne geçilmesini sağlamak adına temasın azaltılması, bina içi dolaşımın azaltılması gibi durumları göz önünde bulundurarak ders programı çizelgesini oluşturmaktır.

## I. KAVRAMSAL ÇERÇEVE

Eğitim kurumları için zaman çizelgeleme; ders programı çizelgelemesi problemi (DPÇP) ve sınav programı çizelgelemesi problemi olarak kullanılır. Hangi derslikte hangi öğretim üyesinin hangi dersi hangi sınıfa vereceği konusunda kullanılan DPÇP bir eğitimsel zaman çizelgelemesi problemidir.

1960'lardan bu zamana araştırmacıların perspektifine takılan ders programı çizelgeleme problemleri (DPÇP) için belirli yahut net çözüm yöntemi halen bulunamamıştır. Bunun en önemli nedeni ise bölgeden bölgeye değişen kimi zaman aynı bölge içinde dahi değişkenlik gösteren ihtiyaç ve taleptir (Altunay & Eren, 2017).

Carter ve Laporte tarafından 1998 yılında ortaya atılan DPÇP'nin, çok boyutlu bir atama problemi olduğu ve ders, öğretim üyesi ve sınıf kıt kaynakların, olası zaman dilimlerine tayin edilmesi olarak tanımlanmıştır. Bir üniversite ya da bölüme bağlı DPÇP'nin verimli ve aktif bir biçimde çözümlenebilmesi için öncelikle; ders, sınıf ve öğretim üyeleri gibi kıt kaynakların fitratının kusursuz ve sahici şekilde tanımlanması gerekmektedir (Özyandı, 2010). Gerekli veriler Tablo 1'de gerekli veriler açıklanmaktadır.

Veriler doğru ve net bir şekilde belirlendikten sonra oluşturulacak matematiksel model kurulmalıdır. Bu noktada öncelikle kıt yapıları

**TABLO 1 | Başlangıç Girdileri**

Ders Bilgileri	Derslik Bilgileri	Öğretim Üyesi Bilgileri	Diğer Bilgiler
Ders saati	Kapasite	Sorumlu olduğu dersler	Günlük mümkün ders saati
Zorunlu-seçmeli ders bilgisi	Kullanılabilir zaman dilimi	Çalışma durumu (tam-yarı zamanlı)	Öğle arası ve ders araları harici uygun zaman dilimleri
Dersin öğretim üyesi	Laboratuvar mı, derslik mi?	Ders saati için uygun zaman aralıkları	

**Kaynak:** (Eren, Taş, & Bedir, 2018)

ve amaç fonksiyonu belirlenmelidir. Kıst; soruna özgü belirleyici sınırlamalardır (Topcu & Kabak, 2021). Kıst yapıları esnek kıstlar ve zorunlu kıstlardan oluşur.

*Zorunlu (Kati) kıstlar;* olası bir çizelgenin elde edilmesi için sağlanması şart olan kıstlardır (Özyandı, 2010). Bir bakıma derslerin, öğrencilerin, öğretim üyelerinin ve dersliklerin çakışmasını engellemekle yükümlü olan kıstlardır. Herhangi bir zaman diliminde bir dersliğe birden fazla ders, öğretim üyesi ve öğrenci grubu atanamaz, bir öğrenci aynı anda birden fazla derslikte bulunamaz, belirli bir ders aynı zamanda farklı dersliklerde aynı öğretim üyesince verilemez gibi kıstlar bu kategoride değerlendirilir. Literatürde yer edinmiş zorunlu kıstlar şu şekildedir (Özyandı, 2010; Altunay & Eren, 2017):

ZK1. Herhangi bir öğrenci ya da öğretim üyesi aynı zamanda birden çok derse ya da sınıfa atanamaz.

ZK2. Bir öğrencinin girmekle mükellef olduğu herhangi bir derse birden çok öğretim üyesi atanamaz.

ZK3. Ders programındaki tüm derslerin atamaları eksiksiz tamamlanmalıdır.

ZK4. Derslerin atandığı sınıfın kapasitesi, dersten mükellef olan öğrenci sayısına özdeş veya çok olmalıdır.

ZK5. Herhangi bir sınıfa, aynı zamanda birden fazla ders atanamaz.

ZK6. Her öğretim üyesi önceden belirlenen adet kadar derse atanabilir.

ZK7. Bir öğretim üyesinin öncesinde kesinleşmiş bir zaman aralığı veya iş gününe atanması gerekebilir

ZK8. Kimi dersler arasındaki mümkün öncelik bağına önem verilmelidir.

ZK9. İki ve ikiden fazla saat süren derslere ait oturumlar devam niteliğinde zaman dilimlerinde yer almalıdır.

ZK10. Herhangi bir dersin aynı gündeki ders saati üçten az olmalıdır.

ZK11. Derslerin mevcut özelliklerini ve taleplerini karşılayacak biçimde ders-sınıf atamaları yapılmalıdır.

ZK12. Öğrenciler ve öğretim üyelerinin beklentilerine göre belirli günlere veya belirli zaman dilimlerine ders ataması yapılmaması istenebilir.

*Esnek (Yumuşak) kıstlar;* karşılanmama olasılığı olan ancak karşılanırsa ders programının randımanını arttıracak kıstlardır. Öğretim üyelerinin gün tercih skalası, ders vermek istedikleri saatler, öğle aralarının uzunluğu, zor olarak betimlenen derslerin sabah saatlerine atanması gibi kıstlar esnek kıstlar altında işlenir.

Literatürde kabul görmüş ancak değişkenlik gösteren esnek kıstlar ise şu şekilde sıralanabilir (Altunay & Eren, 2017; Özyandı, 2010):

EK1. Öğretim üyelerinin mümkün mertebe kendi tercihleri olan zaman dilimleri ve sınıflara atanması,

EK2. Öğretim üyelerinin ders yüklerinin olanca dengelenmesi,

EK3. Bir öğrenci grubu için bir günde sadece bir ders atanmasının engellenmesi,

EK4. Derslerin ve ders saatlerinin bir haftalık zaman dilimine, üniversitenin kendi bünyesinde ayarladığı saatlere dengeli bir şekilde dağıtılması,

EK5. Üç veya daha fazla ders saati içerdiği için bölünmüş bir dersin, başka günlerdeki diğer oturumlarının da aynı sınıfta verilmesi,

EK6. Öğle paydosu nedeniyle boş bırakılan zamanın bütün öğrenciler için aynı zamana ve daha spesifik olarak 12.00-14.00 saatleri arasında denk getirilmesi,

EK7. Öğle paydosu saatleri belirlenirken o andaki öğle arasında olan

öğrenci sayısının göz önüne alınması,

EK8. Her bir öğrenci için arka arkaya olan saatlerdeki derslerin atandığı sınıfların aynı binada olması,

EK9. Herhangi bir gün için ders saati miktarının doğru oranda sınırlandırılması,

EK10. Sınıf atamasının mümkün mertebe önceden kesinleşen sınıflara yapılması,

EK11. Göreceli olarak anlaşılması güç derslerin sabah saatlerine atanması,

EK12. Gün içindeki ilk ve son ders arasındaki zaman zarfının olabildiğince az olması,

EK13. Ders programının, haftanın en az bir gününü boş bırakacak şekilde planlanması,

EK14. Konsantrasyonun azaldığı yahut verimin düştüğü zayıf saatlere olabildiğince az ders ataması yapılması, şeklinde ifade edilebilmektedir.

Kısıtların türü çoğu zaman araştırmacıya göre değişkenlik gösterebilmektedir. Örneğin sınıf kapasitesi Burke ve Petrovic (Burke & Petrovic, 2002; 140 (2)) tarafından zorunlu kısıt olarak kabul edilirken Alvarez-Valdes vd. (Alvarez-Valdes, Crespo, & Tamarit, 2002: 137 (3)) tarafından esnek kısıt olarak kabul edilmiştir.

Amaç fonksiyonu ise; kurulan matematiksel modeldeki değişkenlerin optimal değerlere ulaşabilmesini sağlar (Altunay & Eren, 2017). Amaç fonksiyonu, problemin çözümüne istinaden maksimum veya minimum olarak hesaplanır. DPCP'nin çözümüne ilişkin kurulan modellerin geneli esnek kısıtları sağlamak üzerine kurulur. Böylece hem bir çizelge elde edilmiş hem de verim ve etkinlik edinilmiş olur. Öğrenci istek ve taleplerinin eniyilmesi, tercih dahilinde olmayan saatlere mümkün mertebe ders ataması yapılmaması gibi ifadeler amaç fonksiyonu olarak değerlendirilebilir. Genel olarak literatürde kabul gören amaç fonksiyonu ifadeleri (Özyandı, 2010; Altunay & Eren, 2017):

AF1. Öğrencilerin beklenti ve isteklerinin mümkün olan en üst seviyede karşılanması,

AF2. Öğretim üyelerinin tercihlerinin karşılanma düzeyinin eniyilmesi,

AF3. Öncelik verilmeyen bir zamana yahut güne atanan derslerin sayısının en aza indirgenmesi,

AF4. Sınıf içerisinde ayaktaki öğrenci miktarının en aza indirgenmesi,

AF5. Kullanılan sınıf sayısının en küçüklenmesi,

AF6. Mümkün ders saatleri ve sınıf çakışmalarının en düşük seviyelere çekilmesi,

AF7. Bir günde, bir öğrencinin mesuliyetinde olan iki ders arasında kalan boş vaktin en küçüklenmesi şeklinde örneklenebilir.

## II. GENETİK ALGORİTMANIN TARİHSEL İLĞİ VE EVRİMİ

'Genetik algoritma' kelimelerinin ilk kullanımı ve genetik algoritmanın (GA) ilk yayınlanmış uygulaması Bagley'in (1967) öncü tartışmasında geçmektedir. Oyun oynama programlarına ilginin arttığı dönemde Bagley 3×3 satranç tahtası üzerinde oynanan Hexapawn oyununun rakibinin kalibresini ayarlayarak görevin doğrusal olmadığını kontrol edebilmiştir. Bagley değerlendirme fonksiyonlarında parametre kümelerini aramak için GA'lar inşa ederek bunları korelasyon algoritmalarıyla karşılaştırmıştır.

Bagley ile aynı zamanda çalışan Rosenberg (1967), doktora tezinde GA'ları araştırarak biyolojik ve simüle yönlerinin genetiğe katkılarını vurgulamaktadır.

1970 yılında gelindiğinde Cavicchio 'Adaptive Search Using Simulated Evolution' (Cavicchio, 1970) adlı çalışmasında yapay aramanın iki problemine GA uygulamıştır: bir alt yordam seçim problemi ve bir kalıp yeniden ateşleme yöntemi.

Yine 1970 yılında Weinber 'Computer Simulation of a Living Cell' (Weinberg & Berkus, 1970) adlı tez çalışmasında farklı simüle edilmiş hücrelerinin çalışmasını kontrol eden 15 hız sabiti kümesi seçmek için çok katmanlı GA kullanımını önermiştir. Ancak GA'yı matematiksel optimizasyonun saf bir problemine uygulayan ilk tez Hollstein'in 'Artificial Genetic Adaptation in Computer Control Systems' çalışması kabul edilmektedir (Hollstien, 1971).

1972 yılında Frantz ve konumsal etki; Bosworth, Foo ve Zeigler gerçek genler üzerine çalışmalarıyla silsileyi devam ettirmektedir

(Frantz (1972); Bosworth, Foo, ve Zeigler (1972); Goldberg, 1989).

## III. LİTERATÜR TARAMASI

Tung ve arkadaşları (Tung, Jafreezal, Hoang Nguyen, & Ngoc Bui, 2021), yaptıkları çalışmada çok amaçlı bir optimizasyon yaklaşımını tartışmaktadır. Model karışık tam sayılı programlama ve ikili değişkenlerin kombinasyonunu kullanmaktadır. 3000 öğrenciyi içeren gerçek hayat verileriyle ulaşılan sonuçlar hem çözüm kalitesi hem de hesaplama maliyetleri açısından etkinliğini kanıtlamıştır.

Alnowaini ve Aljomai (Alnowaini & Aljomai, 2021), bir GA kullanarak bir fakülte zaman çizelgesi oluşturmak için otomatik bir sistem önermektedir. Mühendislik ve bilişim fakültesi ders programlarını dinamik kromozom büyüklüğünde ve her bölümün ders sayılarına göre esnek olacak şekilde planlamak için genetik algoritma kullanılmıştır. Önerilen sistem, manuel programlama veya mevcut sistemlerle karşılaştırıldığında, değerlendirme aşamasında yaklaşık %93 oranında optimuma yakın sonuçlar elde etmektedir.

Rezaeipanah ve ark. (Rezaeipanah, Matorri, & Ahmadi, 2021), yaptıkları araştırma ile geliştirilmiş paralel GA ve yerel aramaya dayanan hibrit bir yöntem ile çalışmaktadır. Yapılan çalışmada yerel arama GA'yı güçlendirmek için kullanılmaktadır. BENPAechter ve ITC-2007 olarak adlandırılan literatürdeki son teknoloji teknikler ile uygulama sonuçları test edilmiş ve deneysel sonuçlar önerilen çözüm yöntemini ve üstünlüğünü doğrulamıştır.

Kakkar vd. (Kakkar, ve diğerleri, 2021, Ağustos), kısıtlı zaman çizelgeleme problemlerini çözmek için evrimsel algoritmaları temel alan bir yöntem önermişlerdir ki bu yöntemler üniversiteler için sadece sınıf değil laboratuvarları da içeren bir teoride çizelge oluşturmaktadır. GA ve memetik algoritma zaman çizelgeleme problemleri açısından karşılaştırılmış, memetik algoritma ve GA kullanılarak kapsamlı çözüm sunmak için mevcut seçeneklerden en iyi çözümü seçerek zaman çizelgeleme problemlerinde çıktı optimize edilmiştir.

Tian ve ark. (Tian, Si, Guo, Zhao, & Feng, 2021), yaptıkları sistemde, özel bir uygunluk fonksiyonu tasarlayarak bu fonksiyonun değerini belirli bir dersin çizelgeleme sonucunun kısıtları karşılanma derecesini ölçmek için kullanmaktadır. Sınıf-öğretmen-ders ve konularının eşleşmesini makul hale getirmek niyetiyle atlama ve filtreleme işlemlerine sahip bir rulet algoritması sisteme eklenmiştir.

Geleneksel GA ile Monte Carlo GA'sına dayalı bir yazılım geliştiren Shuai (Shuai, 2021), otomatik ders çizelgeleme sistemi önermektedir. Donanım, sistem yönetimi, ders çizelgeleme bilgi girişi, ders çizelgeleme yöntemi ve sorgusunu içermektedir. Yazılım kısmında ise süreci optimize etmek için Monte Carlo GA'sı kullanılmaktadır.

Chen ve diğerleri (Chen, Yue, Zhumadillayeva, & Liu, 2021), genel olarak ders çizelgeleme sistemlerinde tepe tırmanma algoritmaları, tabu arama algoritmaları, karınca kolonisi optimizasyonu ve simüle edilmiş tavlama algoritmalarının eksikliklerinden yola çıkarak mutasyon GA'sını araştırarak deneyelemektedir. Ders çizelgeleme sisteminin ayrıntılı işleyişinde, ders çizelgeleme esas olarak manuel çizelgeleme ve otomatik sınıf çizelgeleme kullanılarak tamamlanmıştır. Temelde herhangi bir zamanda en uygun sınıf ayarlanmıştır.

Duan ve Lu (Duan & Lu, 2021), araştırma süreci için bir üniversiteden ders planlama sistemi seçilmiş ve orijinal sistem A ve genetik-karınca hibrit algoritma geliştirilmiş sistem B olarak adlandırılmaktadır. B sistemi uygunluk, programlama birimi, çalışma süresi ve sistem uygunluğu açısından karşılaştırıldığında orijinal olan A sisteminden daha rasyonaliteye sahip olduğu görülmektedir.

Amrulloh ve Sela (Amrulloh & Sela, 2021), programlama modeli için GA ve tabu arama algoritması kullanarak ders programı oluşturacak bir sistem kurmaktadır. 561 saniyelik işlem süresine sahip program 265 veriyi öğretim üyeleri ve öğrencilerin ders programı çizelgesiyle çalışmadan sunmaktadır.

Zhang ve ark. (Zhang, Li, & Zhang, 2021), Çin'deki yeni üniversite giriş sınavı olan CEE sınavına yönelik yaptıkları çizelgeleme çalışmalarında GA kullanarak, sınıf doluluk oranlarını %10 oranında azaltabilmişlerdir.

Yang (Yang, 2021), geleneksel öğretim modunda var olan sorunları hedefleyerek çok yönlü mutasyon GA ve sinir ağları optimizasyonu kullanarak İngilizce öğretim kurslarını akıllıca optimize etmeyi hedeflemiştir.

Xu (Xu, 2021), çok amaçlı optimizasyon problemi olan İngilizce derslerinin çizelgeleme problemi için geliştirilmiş bir GA önermektedir.

Çolak ve Yiğit (Çolak & Yiğit, 2021), ders çizelgelemesinde ihlallerin cezalandırıldığı ve bahsi geçen cezaların en aza indirilmesine olanak sağlayan bir çalışma yapmışlardır.

Subagio ve ark. (Subagio, Putri, Sokibi, & Harahap, 2021), çakışan çizelgelemeler olmadan hızlı bir şekilde planlama süreci ve öğretim üyelerinin öğretmeye istekli oldukları günleri baz alan sonuçlara ulaşmıştır.

Fedkin vd. (Fedkin, Denissova, Krak, & Dyomina, 2021), eğitim programlarının otomatikleştirilmesi noktasında çizelgelerin derlenmesi, optimizasyonun geliştirilmesi ve uygulanmasını konu almıştır.

Pérez vd. (Pérez, ve diğerleri, 2021), çalıştıkları makalede eğer çaprazlama yüzdesi mutasyon yüzdesinden daha büyükse çözüme ulaşmak için birkaç nesil daha gerektiğine dikkat çekmektedir. Fenotip ve genotip, uygunluk fonksiyonu, kısıtlar, cezalar, çaprazlama yüzdesi ve metodu, mutasyon yüzdesi ve metodu, bireylerin ve jenerasyonların numarası gibi sorunlar hakkında analizler yapılmıştır.

Hu vd. (Hu, Wang, & Duan, 2021), genellikle ana kısıt olarak alınan öğretmen ve öğrenci tercihlerinin yanı sıra sınıf verimliliğini de konu almaktadır. Çalışmada GA ve parçacık sürü optimizasyonu, geliştirilmiş GA ile karşılaştırılmaktadır. Çok boyutlu analiz, geliştirilmiş GA'nın üstünlüğünü kanıtlamaktadır.

Sun vd. (Sun, Luo, & Liu, 2021), binaların enerji tüketimini, doluluk süresini ve kullanıcı sayısını göz önünde bulundurarak enerji tüketimini minimize etmeyi hedeflemektedir. Enerji tüketimini ve ders programı çizelgelemesiyle optimize etmek için GA kullanılarak Bina Kontrolleri Test Yatağı platformuyla ilintili bir ortak simülasyon yöntemi geliştirilmiştir.

Şimşek (Şimşek, 2021), çok amaçlı bir matematiksel model önermektedir. Model 7 bölümden oluşan fakülte için 2020-2021 güz yarıyılı gerçek verileriyle test edilmektedir. Modelde 12084 karar değişkeni ve 15567 kısıtla en uygun çözüm 28 dakikada elde edilmektedir.

Zaulir vd. (Zaulir, Abdülaziz, & Aizam, 2022), yaptıkları çalışmayla genel model alansal veri setleri ile test edilmiş ileri düzey interaktif boyutlu modelleme sistemi kullanılarak problemi çözmektedir. Modelin çıktısı dört ana temaya (öğretim elemanlarının tercihleri, sınıflar, zaman dilimleri ve kapasiteler) dayalı olarak analiz edilmektedir.

Wong vd. (Wong, Goh, & Likoh, 2022), Malezya Üniversitesi Sabah-Labuan International kampüsü için kayıt sonrası çizelgeleme problemini ele alan bir GA önermektedir. Örneklemeye ve Pertürbasyon ile tabu arama, GA'da uygun çözümler havuzunu başlatmak için kullanılmaktadır.

Nasien ve Andi (Nasien & Andi, 2022), yaptıkları çalışma ile ders ortamındaki sorunları ve süreci ele almaktadır. Çalışma optimal bir zamanlama çözümü üretmektedir.

Martin'in yazdığı Peluffo-Ordóñez'in yönettiği tez çalışmasında (Martin & Peluffo-Ordóñez, 2022) çeşitli konulardaki kursları, tek bir uzaktan öğretim kursuna dönüştürmeyi amaçlamaktadır. Sonuçlar zorunlu kısıtları çözemeyen normal GA'ya göre önerilen dinamik uzay algoritmasının daha verimli sonuçlar elde ettiğini göstermektedir.

Yang (Yang, 2022), yabancı dil olarak İngilizce eğitiminin Çin'deki artışını konu almaktadır. Çalışma GA'yı baz alarak üniversite ve kolejlerin çizelgeleme problemini etkileyen ana faktörleri, kısıtları ve çözüm hedeflerini daha pratik ve kapsamlı bir şekilde tartışmaktadır.

Sari vd. (Sari, Ramdhania, & Purnomo, 2022), vaka çalışması olarak Universitas Bhayangkara Jakarta Raya'nın kullanıldığı bir GA kullanılmaktadır. Bu çalışmada amaç fonksiyonu, öğretim elemanlarının ve öğrencilerin iş yüklerinin bir haftadaki günlük ortalama varyansını en aza indirmektir. GA yönteminin uygulanması, öğretim elemanlarının ve bir grup öğrencinin iş yükünün standart değer sapmasında bir hafta içinde 0.114 (%3,68) ve 3.11 (%55,7) oranında azalma ile sonuçlanmaktadır.

Li vd. (Li, Xie, & Zhang, 2022), geliştirilmiş bir genetik karınca kolonisi hibrit optimizasyon algoritmasına dayalı bir kolej zamanlama algoritması tasarımı önermektedir. Deneysel sonuçlar, hibrit optimizasyon algoritmasının, çizelgeleme başarısı ve çizelgeleme süresi açısından performans açısından diğer algoritmalarından daha iyi performans gösterdiğini göstermektedir.

Nugroho vd. (Nugroho, Permadi, & Yasifa, 2022), bir kromozom 154 gen olacak şekilde sınıf, ders, çalışma süresi, öğretim üyesi ve bölüm verileri kullanılarak 31. nesilde optimal ders programı elde etmektedir.

Chaouachi ve Harrabi (Chaouachi & Harrabi, 2022), optimizasyon sürecine hibrit bir GA'yı entegre eden bir etkileşimli karar destek sistemi tasarımının, optimal çözümler sağlamak için nasıl performans gösteren bir yaklaşım olduğunu göstermektedir.

Trenggonowati vd. (Trenggonowati, ve diğerleri, 2022), bibliyometrik analiz önermektedir. Arama ve veri analizi sonuçları, üniversite zaman çizelgeleme probleminin eğiliminin GA ve benzetilmiş tavlama gibi meta-sezgisel modellerle çözümlenmesinin daha muhtemel olduğunu göstermektedir.

Zheng vd. (Zheng, Peng, Guo, & Chen, 2022), geleneksel ders çizelgelemesinde düşük verimliliğin dezavantajlarının ve yüksek çatışma olasılığının üstesinden gelmek için, geliştirilmiş bir ikili guguk kuşu arama algoritmasına dayalı bir ders çizelgeleme yöntemi önermektedir. Deneysel sonuçlar, geliştirilmiş ikili guguk kuşu arama algoritmasının etkin zamanda yakınsayabildiğini göstermektedir.

Huang ve Wang (Huang & Wang, 2022), mevcut üniversite öğretim yönetim sistemi ve GA'yı derinlemesine incelemektedir. Bu çalışmada, kolejler ve üniversitelerdeki öğretim yönetiminin pratik ihtiyaçlarını analiz etmekte, geleneksel GA'yı optimize ederek geliştirmektedir.

Luo vd. (Luo, ve diğerleri, 2022), üniversite binalarında karbon nötrlüğünü sağlamak için, bina enerji verimliliği ve öğrencilerin zamanla değişen termal algısını dikkate alarak ders zaman çizelgesini optimize etmeyi amaçlamaktadır. DPÇP optimizasyon modeli MATLAB ortamında oluşturularak GA yardımıyla çözülmektedir.

Abduljabbar ve Abdullah (Abduljabbar & Abdullah, 2022), zaman çizelgeleme problemini çözmek için genetik teorileri kullanarak evrimsel bir algoritma ilkesi uygulamaktadır. Her aşama için çok çözümlü bir zaman çizelgesi oluşturma yeteneği ile rastgele ve tam optimal zaman çizelgesi oluşturulmaktadır.

Behrenk ve arkadaşları (Behrenk, Güçlükol Ergin, & Toy, 2022, Ekim), Türkiye'de Yaşar Üniversitesi Endüstri Mühendisliği programı üzerinde gerçek hayat verileriyle DPÇP'ni ele almaktadır. Çalışma asgari durumda olan öğretim elemanı memnuniyetini yükseltmeyi amaçlamaktadır. Kullanılan gen yapısıyla matematiksel modelde birçok kısıtın uygulanabilirliği sağlanmaktadır.

Zhang (Zhang Q., 2022), GA'yı geliştirmek için evrimi kullanmaktadır. MATLAB yazılımında geleneksel ve geliştirilmiş GA'lar üzerinde simülasyon deneyleri yapılmaktadır. Sonuçlar, geliştirilmiş GA'nın aynı çaprazlama ve mutasyon olasılığı altında geleneksel GA'ya göre daha hızlı yakınsadığını ve daha iyi çözümler ürettiğini göstermektedir.

Junjun vd. (Junjun, Hexia, Oyam, & Yi, 2022), yaptıkları çalışmada DPÇP için paralel GA uygulamaktadır. Bu sistem arayüz aracılığıyla uygulama şeklinde kullanıma sunulmaktadır.

Ren ve Li (Ren & Li, 2022), akıllı ders planlama için BP (back propagation) sinir ağı önermektedir. Planlama sisteminin kendi kendine uyarlanmasını elde etme noktasında ise GA'dan yararlanılmaktadır. Ayrıca detaylı sistem tasarımı, veri tabanı tasarımı, planlama ve akıllı ders planlama sisteminin tüm fonksiyonlarının yazılım tarafından geliştirilmesi amaçlanmaktadır. Geliştirilen GA, standart GA'dan daha yüksek verim sağlamaktadır.

Wang ve Huang (Wang & Huang, 2023, Şubat), yaptıkları çalışma ile öğretim elemanlarının derslik ve ofisler arasındaki dolaşım süresini kısaltmayı amaçlamaktadır. Modele öğrencileri de dahil ederek herhangi bir salgın döneminde öğrenci ve öğretim elemanı etkileşimini düşüren model aynı zamanda belirli bir gün içindeki öğrenci grubu için aynı sınıfların kullanılmasını amaçlamaktadır. Çalışma Pekin'de bir yüksek meslek okulundaki çizelgelemeyi geliştirilmiş GA yöntemi ile gerçekleştirmektedir.

Ewi ve Radiles (Ewi & Radiles, 2023), sınırlı arama alanı nedeniyle erken yakınsamayı önlemek amacıyla nesil başına nüfus artışı için dinamik popülasyonun kullanılmasını önermektedir. Araştırma verileri, UIN SUSKA -Riau 2019-2020 yarıyılında Elektrik Mühendisliği Bölümü'ndeki planlama sürecine ve çeşitli akademisyenlerle yapılan görüşmelerin sonuçlarına dayanarak elde edilmiştir. Sonuçlar, 70 popülasyonunun çapraz geçiş olasılığının 0,5 ile başlatılmasının, DPÇP'ne bir çözüm bulmak için erken yakınsama sorununu çözebileceğini göstermektedir.

Rodprasert vd. (Rodprasert, Taetrugool, & Akkarajitsakul, 2023), gerçek hayat verileriyle yeni iki kısıta sahip bir matematiksel model

önermektedir. İlk kısıt çevrim içi ve çevrim dışı derslerin eşgüdümü ilerlemesidir. Bu kısıtın ortaya çıkış nedeni COVID-19 pandemisidir. İkinci kısıt ise derslerin birden fazla öğretim üyesiyle planlamasıdır. Çalışmada GA ve harmony arama algoritması sentezlenerek uygulanmaktadır. Deneysel bulgularda GA maliyet, planlanmamış ders saatleri ve pratiklik açısından harmony arama algoritması ve tamsayı programlamaya göre daha iyi performans göstermektedir.

Thang vd. (Thang, Giang, Son, & Anh, 2023), DPÇP'ni çok amaçlı bir optimizasyon problemi olarak modellemek için oyun teorisinin bir dalı olan NASH dengesini kullanmayı önermektedir. Bu sorunu çözmek için bir GA geliştirilmiştir. Algoritma 153 sınıf ve 25 eğitmenen oluşan gerçek dünya veri seti ile denenmiştir.

Mahlous ve Mahlous (Mahlous & Mahlous, 2023), öğrencileri tercihlerine göre sınıflarına atayan bir GA önermektedir. GA'nın performansı, farklı meta-sezgisel kavramların uygulanmasıyla ve genetik operatörlerin verilen probleme göre uyarlanmasıyla artırılmaktadır. Elde edilen sonuçlara göre algoritma, en karmaşık problemlerde bile öğrenci tercihlerinin %90'dan fazlasını karşılamının yanı sıra çözümlerin uygulanabilirliğini de garanti etmektedir.

DPÇP için kullanılan diğer yöntemlere ait özetlenmiş literatür tablosu Tablo 2'de görülmektedir.

**TABLO 2 | DPÇP için Kullanılan Diğer Yöntemlerin Literatür Tablosu**

Yazar	Yıl	Yöntem
Kaynar, Yurtsal	2019	GA
Muklason, Irianti, Marom	2019	Hibrit (Değişken Komşuluk Arama+ Tabu Arama)
Dele	2019	Karınca Kolonisi Optimizasyonu
Susan, Bhutani	2019	Memetik Algoritma+ Genetik Algoritma
Hossain, Akhand, Shuvo, Siddique, Adeli	2019	Parçacık Sürüsü Optimizasyonu
Kristiadi, Hartanto	2019	GA
Modibbo, Umar, Mijinyawa, Hafisu	2019	GA+ Microsoft Visual C++
Jiang, Liu	2019	Geliştirilmiş GA
Mauluddin, Ikbal, Nursikuwagus	2020	Karınca Kolonisi Optimizasyonu+ Genetik Algoritma
Ansari, Saubari	2020	GA
Aki	2020	GA
Sun, Li	2020	Geliştirilmiş GA
Sabtarini, Ciptayani, Purnama	2020	Geliştirilmiş GA
Gozali, Kurniawan, Weng, Fujimura	2020	Geliştirilmiş GA
Szea, Kuan, Chiewa, Tionga, Hengb	2020	GA
Thakare, Nikam, Patil	2020	GA
Mahmud	2021	Karınca Kolonisi Optimizasyonu+ Genetik Algoritma
Sakal, Fieldsend, Keedwell	2021	Karınca Kolonisi Optimizasyonu+ Yerel Arama
Tan, Goh, Sura, Kendall, Sabar	2021	Parçacık Sürüsü Optimizasyonu+ Tepe Tırmanışı Algoritması
Tung, Jafreezal, Hoang Nguyen, Ngoc Bui	2021	Hibrit (GA+ Tam Sayılı Programlama)
Alnowaini, Aljomai	2021	Otomatik Sistem GA
Rezaeipah, Matorri, Ahmadi	2021	Hibrit (Geliştirilmiş GA+ Yerel Arama)
Kakkar, Singla, Garg, Gupta, Srivasta, Kumar	2021	Hibrit (GA+ Memetik Algoritma)
Tian, Si, Guo, Zhao, Feng	2021	GA
Shuai	2021	Hibrit (Geleneksel GA+ Monte Carlo GA)
Chen, Yue, Liu, Zhumadillayeva, Liu	2021	Geliştirilmiş GA
Duan, Lu	2021	Hibrit (GA+ Karınca Kolonisi)
Amrulloh, Sela	2021	Hibrit (GA+ Tabu Arama)
Zhang, Li, Zhang	2021	GA
Yang	2021	Hibrit (GA+ Yapay Sinir Ağları)
Xu	2021	Geliştirilmiş GA
Çolak, Yiğit	2021	GA

**TABLO 2 (Devamı) | DPÇP için Kullanılan Diğer Yöntemlerin Literatür Tablosu**

Yazar	Yıl	Yöntem
Subagio Putri, Sokibi, Harahap	2021	GA
Fedkin, Denissova, Krak, Dyomina	2021	GA
Pérez, Rios, Bautista, Sanchez, Acevedo	2021	GA
Hu, Wang, Duan	2021	Hibrit (Geliştirilmiş GA+ Parçacık Sürüsü Optimizasyonu)
Sun, Luo, Liu	2021	GA
Şimşek	2021	GA
Al-Tarawneh0, Al- Kaabneh, Alhroob, Migdady, Alhadid	2022	Değişken komşuluk Arama
(Tablo 2 devamı)		
Ghaffar, Sattar, Munir, Qureshi	2022	Genetik Algoritma+ Yerel Arama+ Memetik Algoritma
Zaulir, Abdülaziz, Aizam	2022	GA
Wong, Goh, Likoh	2022	Hibrit (GA+ Tabu Arama)
Nasien, Andi	2022	GA
Martin, Peluffo-Ordóñez	2022	GA
Yang	2022	GA
Sari, Ramdhania, Purnomo	2022	GA
Li, Xie, Zhang	2022	Hibrit (Geliştirilmiş GA+ Karınca Kolonisi)
Nugroho, Permadi, Yasifa	2022	GA
Chaouachi, Harrabi	2022	Hibrit (GA+ Etkileşimli Karar Destek Sistemi)
Trenggonowati, Herlina, Febianti, Ilhami, Muharni, Kurniawan, Irman	2022	GA+ VOSviewer Yazılımı+ Publish and Perish Yazılımı
Zheng, Peng, Guo, Chen	2022	Hibrit (GA+ İkili Guguk Kuşu Algoritması)
Huang, Wang	2022	GA
Luo, Sun, Liu, Gao, Sun, Liu	2022	GA
Abduljabbar, Abdullah	2022	GA
Behrenk, Güçlükol Ergin, Toy	2022	GA
Cruz-Rosales, Cruz-Chávez, Alonso-Pecina, Peralta-Abarca, Ávila-Melgar, Martínez-Bahena, Enriquez-Urbano	2022	Hibrit (GA+ Tavlama Simülasyonu)
Zhang	2022	Geliştirilmiş GA
Junjun, Hexia, Oyam, Yi	2022	GA
Ren, Li	2022	Hibrit (GA+ BP Sinir Ağı)
Wang, Huang	2023	Geliştirilmiş GA
Ewi, Radiles	2023	GA
Rodprasert, Taetragool, Akkarajisakul	2023	Hibrit (GA+ Harmony Arama Algoritması)
Thang, Giang, Son, Anh	2023	GA+ NASH Dengesi
Mahlous, Mahlous	2023	GA

Kaynak: (Yılmaz, 2023)

## IV. YÖNTEM

GA, DPÇP'nde en çok tercih edilen sezgisellerden biridir. 1975'te John Holland'ın makine öğrenmesi ve canlılardaki genetik evrim sürecini makineye taşımasıyla ortaya çıkmıştır. Böylece tek bir makineyi geliştirmek yerine miktardaki makinenin "çiftleşme ve değişmesi" sonucu yeni üstün bireyler elde edeceğini savunmuştur.

Çözüm alanı oldukça geniş olan GA, olası bütün çözümler arasından en uygun çözümü bulmaya çalışır. Nesilden nesile geçtikçe kötü çözümler yok olmaya iyi çözümlere gelişmeye ve daha iyi çözümlere gebe olmaya eğilimlidir. Çözüm uzayının yalnızca bir kısmını tarayan GA, bu sayede etkin aramayla çok kısa bir sürede çözüme ulaşabilir. Genellikle ise problemi etkileyen faktörler fazla olduğu durumlarda tercih edilen bir algoritmadır.

GA, doğal seçim mekanizmasına ve doğal genetiğe dayanan arama algoritmalarıdır. GA evrim teorisinden esinlendiği üzere her şeyden

önce böyle bir evrimi yapabilecek ‘yaratıklara’ ihtiyaç duyulmaktadır. Bahsi geçen yaratıklar doğurabilmeli ve ölebilmelidir. Doğum sonrası kendilerinde olan bazı davranışları kalıntı olarak yeni doğan jenerasyona aktarmaktadırlar (Şen, 2004).

Dizi yapıları arasında en uygununun hayatta kalmasını, rastgele olarak yapılandırılmış bir bilgi alışverişi ile birleştirerek, insan aramasının yenilikçi yeteneklerinden bir kısmına sahip bir arama algoritması oluşturmaktadır. Her nesilde eskinin en uygun parçaları kullanılarak yeni yapay bir dizi kümesi oluşturulur ve yeni seri daha iyi bir sonuç için denir. Rastgele olsa dahi GA basit bir rastgelelik içermez. Özellikle verimli kullanılabilmek adına bahsi geçen kromozomlar (diziler) yeni beklenen geliştirilmiş performans için geçmiş bilgi düzeyine göre yeniden işlenmektedir. John Holland, meslektaşları ve öğrencileri tarafından geliştirilen GA'nın amacı araştırmacılar tarafından şu şekilde özetlenmiştir: (1) doğal sistemlerin uyarlanabilir süreçlerini soyutlamak ve titizlikle açıklamak, (2) doğal sistemlerin önemli mekanizmalarını koruyan yapay sistemler yazılımı tasarlamak. Bu yaklaşım hem doğal hem de yapay sistem biliminde önemli keşiflere yol açmıştır (Goldberg, 1989).

GA genel anlamda, dizilerden oluşan bir popülasyona çoğalma, çaprazlama ve mutasyon operatörlerinin uygulanmasını içerir. Bu operatörlerin uygulanmasından sonra yeni bir popülasyon (yavru popülasyon) oluşur. Yeni popülasyon eski (ebeveyn) popülasyon ile yer değiştirir. Her dizinin uyum değeri mevcuttur. Diziler uyum değerine göre seçilirler. Ortalama uyum değerinin üzerinde uyuma sahip dizilerin gelecek kuşaklarda temsil edilme olasılığı daha yüksektir. Evrim süreci, popülasyonun ortalama uyumunu giderek artırır ve ilerleyen kuşaklarda daha iyi uyum değeri elde edilmesi şansını artırır (Taşkın & Gökay Emel, 2009).

GA'ların esaslı doğal seçme ve genetik kurallarına dayanmaktadır. Bu kuralları ortama en fazla uyum sağlayan canlılar hayatına devam ederken uyum sağlamayanların elenmesi olarak algılanmaktadır. Bu sebeple denilebilir ki GA'lar bu iki kuralı bir arada kullanarak en iyiyi aramayı hedef edinen bir eniyileme yöntemidir (Şen, 2004).

GA ile bir sorunu çözmeye çalışırken birbirini takip eden beş aşamanın ön hazırlık olarak tamamlanması elzemdir.

1. Sorunun ne olduğuna ait kritik ve ayrıntılı veriler sözel ve mümkünse sayısal olarak elde edilmelidir.
2. Değişkenlerin kromozom (dizi) yapılarına karar verilerek veri hazırlanmalıdır.
3. Karar değişkenlerinden kromozoma (diziye) geçirmek için tüm alt yapı oluşturulmalıdır.
4. Hedef fonksiyonun karar değişkenlerine bağlı olarak analitik ifadesi tespit edilmelidir.
5. Hedef fonksiyonundan her bir karar takımına (kromozom) karşılık gelecek derecelerin hesaplanmasına ilişkin dönüşümü belirlenmelidir (Şen, 2004).

## A. MATEMATİKSEL MODEL VE GENETİK ALGORİTMA OPERATÖRLERİ

Çalışmaya özgü kısıtlar ve amaç fonksiyonunu sağlamak üzere oluşturulan matematiksel model ve açıklamaları aşağıda sunulmaktadır (Yılmaz, 2023).

### Notasyonlar

İndisler	Kümeler	Açıklama
	$O$	Çevrim içi dersler
$ij$	$Y$	Yüz yüze dersler
	$I=O \cup Y$	Tüm dersler
$k$	$K$	Öğretim üyeleri
$l$	$L$	Derslikler
$r$	$R$	Günler
$h$	$H$	Zamanlar
$g$	$G$	Koridorlar

### Parametreler

$p_{ik}$	$\begin{cases} 1, & \text{eğer } i. \text{ ders } k. \text{ öğretim üyesi tarafından veriliyorsa} \\ 0, & \text{aksi halde} \end{cases}$
$d_i$	$i. \text{ dersi alan öğrenci sayısı}$
$q_l$	$l. \text{ dersliğin kapasitesi}$
$a_l$	$\begin{cases} 1, & \text{eğer } l. \text{ derslik özel amaçlı (amf1, lab vb) kullanılabiliriyorsa} \\ 0, & \text{aksi halde} \end{cases}$
$o_i$	$\begin{cases} 1, & \text{eğer } i. \text{ ders özel amaçlı (amf1, lab vb) derslik gerektiriyorsa} \\ 0, & \text{aksi halde} \end{cases}$
$n_{ij}$	$\begin{cases} 1, & \text{eğer } i. \text{ ve } j. \text{ dersler aynı sınıfa aitse} \\ 0, & \text{aksi halde} \end{cases}$
$m_{lg}$	$\begin{cases} 1, & \text{eğer } l. \text{ derslik ve } g. \text{ koridorda yer alıyorsa} \\ 0, & \text{aksi halde} \end{cases}$
$v$	Bir koridorda atanabilecek maksimum derslik sayısı

### Karar Değişkenleri

$X_{irhl}$	$\begin{cases} 1, & \text{eğer } i. \text{ ders } r. \text{ gün } h. \text{ zaman diliminde } l. \text{ dersliğe atandıysa} \\ 0, & \text{aksi halde} \end{cases}$
$S_{jyr}$	$\begin{cases} 1, & \text{eğer } r. \text{ günde oluşan } i. \text{ ile } j. \text{ çevrim içi ders atama sapması varsa} \\ 0, & \text{aksi halde} \end{cases}$
$b_{rl}$	$r. \text{ günde } l. \text{ dersliğe yapılan fazla yüz yüze ders atama sayısı}$
$C_{rhg}$	$r. \text{ günde } h. \text{ Zaman diliminde } g. \text{ Koridora fazladan yapılan derslik atama sayısı}$
$e_{rk}$	$r. \text{ günde } k. \text{ Öğretim üyesine fazladan yapılan yüz yüze ders atama sayısı}$

### Amaç Fonksiyonu

$$\min z = \sum_{i=1}^O \sum_{j=1}^O \sum_{r=1}^R S_{jyr} + \sum_{i=1}^I \sum_{r=1}^R \sum_{h=1}^H \sum_{l=1}^L X_{irhl} * (1 - o_i) * a_l + \sum_{r=1}^R \sum_{l=1}^L b_{rl} + \sum_{r=1}^R \sum_{h=1}^H \sum_{g=1}^G C_{rhg} + \sum_{r=1}^R \sum_{k=1}^K e_{rk} \quad (1)$$

### Kısıtlar

$$\sum_{r=1}^R \sum_{h=1}^H \sum_{l=1}^L X_{irhl} = 1 \quad \forall i \quad (2)$$

$$\sum_{l=1}^L X_{irhl} \leq 1 \quad \forall r, h, l \quad (3)$$

$$\sum_{l=1}^L X_{irhl} \leq 1 \quad \forall i, r, h \quad (4)$$

$$\sum_{l=1}^L \sum_{i=1}^I X_{irhl} * p_{ik} \leq 1 \quad \forall k, r, h \quad (5)$$

$$\sum_{l=1}^L X_{irhl} + \sum_{l=1}^L X_{jrhl} \leq 1 + M * (1 - n_{ij}) \quad \forall i, j, r, h \quad i \neq j \quad (6)$$

$$X_{irhl} * d_i \leq q_l \quad \forall i, r, h, l \quad (7)$$

$$X_{irhl} * o_i \leq a_l \quad \forall i, r, h, l \quad (8)$$

$$\sum_{h=1}^H \sum_{l=1}^L X_{irhl} \leq 1 + b_{rl} \quad \forall r, l \quad (9)$$

$$\sum_{h=1}^H \sum_{l=1}^L \sum_{i=1}^I X_{irhl} * p_{ik} \leq 1 + e_{rk} \quad \forall r, k \quad (10)$$

$$S_{jyr} \geq n_{ij} * \left( \sum_{h=1}^H \sum_{l=1}^L X_{irhl} - \sum_{h=1}^H \sum_{l=1}^L X_{jrhl} \right) \quad \forall r \quad \forall i, j \in O \quad i \neq j \quad (11a)$$

$$S_{jyr} \geq n_{ij} * \left( \sum_{h=1}^H \sum_{l=1}^L X_{jrhl} - \sum_{h=1}^H \sum_{l=1}^L X_{irhl} \right) \quad \forall r \quad \forall i, j \in O \quad i \neq j \quad (11b)$$

$$S_{jyr} \leq n_{ij} * \left( \sum_{h=1}^H \sum_{l=1}^L X_{jrhl} + \sum_{h=1}^H \sum_{l=1}^L X_{irhl} \right) \quad \forall r \quad \forall i, j \in O \quad i \neq j \quad (11c)$$

$$S_{jyr} \leq 2 - n_{ij} * \left( \sum_{h=1}^H \sum_{l=1}^L X_{jrhl} + \sum_{h=1}^H \sum_{l=1}^L X_{irhl} \right) \quad \forall r \quad \forall i, j \in O \quad i \neq j \quad (11d)$$

$$\sum_{l=1}^L \sum_{i=1}^I X_{irhl} * m_{lg} \leq v + c_{rhg} \quad \forall r, h, g \quad (12)$$

$$X_{irhl} \in (0, 1), S_{jyr} \geq 0 \text{ ve } \text{tamsayı}, e_a \geq 0 \text{ ve } \text{tamsayı}, b_r \geq 0 \text{ ve } \text{tamsayı}, c_{rhg} \geq 0 \text{ ve } \text{tamsayı} \quad \forall i, r, h, l \quad (13)$$

(1) numaralı denklem amaç fonksiyonunu göstermektedir. Bu denkleme göre, ders çakışmasını minimumda tutmak adına çakışmalar büyük bir ceza puanıyla ( $\alpha$ ) çarpılmış ve esnek kısıt ihlal sayıları toplanmıştır. (2) numaralı kısıt her dersin mutlaka bir kez atanma zorunluluğunu sağlamaktadır. (3) numaralı kısıt belirli bir gün, zaman dilimi ve derslik için en çok bir dersin atanmasını sağlamaktadır. (4) numaralı kısıt belirli bir gün, zaman dilimi ve dersin en çok tek bir dersliğe atanmasını sağlamaktadır. (5) numaralı kısıt öğretim üyelerinin belirli bir gün ve zaman diliminde en çok tek bir derslikte bulunmasını sağlamaktadır. Böylece belirli bir gün ve zaman dilimine tek bir öğretim üyesi atanacaktır. (6) numaralı kısıt belirli bir gün ve saatte aynı öğrenci grubuna ait iki dersin atanmamasını sağlamaktadır. Burada kullanılan  $M$ ,  $M > 0$  olup, çok büyük bir sayıyı ifade etmektedir. (7) numaralı kısıt derslik kapasitesinin aşılmamasını sağlamaktadır. (8) numaralı kısıt özel derslik ihtiyacına sahip derslerin bu dersliklere atanmasını sağlamaktadır. (9) numaralı kısıt her bir dersliğe, eğitim verilen her gün için tek bir yüz yüze ders atanmasını sağlamaktadır. (10) numaralı kısıt herhangi bir öğretim üyesinin bir günde birden fazla örgün ders vermemesini sağlamaktadır. (11) numaralı kısıt aynı sınıfa ait çevrim içi derslerin aynı günde yapılmaya çalışmasını zorlamaktadır. (12) numaralı kısıt ise aynı koridorda yer alan dersliklere belirli bir gün ve zaman diliminde atanan ders sayısının belirli bir sınırın altında olmasına zorlamaktadır. (13) değişkenler için tanım aralıklarını göstermektedir.

Önerilen matematiksel model değerlendirildiğinde problem örnekleri karmaşıklıkça optimal çözüme ulaşma süresinin artabileceği görülmektedir. Önceki çalışmalarda da DPÇP'nin NP-zor problem sınıfında olduğu vurgulanmıştır (Yang ve ark, 2017). Bu nedenle, klasik matematiksel modellerden elde edilecek kesin çözümlere alternatif olarak sezgisel çözümler sıkça kullanılmaktadır. Bu çalışmada da optimale yakın çözümler üretmek için meta-sezgisel yaklaşımlardan olan genetik algoritma ile probleme çözüm aranmıştır.

Genetik algoritma ile matematiksel modelin çözümünün bulunması için izlenen adımlar Algoritma 1'de sunulmuştur.

### Algoritma 1 | Ders Programı Çizelgesi için Genetik Algoritma

```

1: procedure GA = (pik, di, qi, ai, oi, ny, mlg, v, a, pop, cro, mo, itermax)
2: Başlangıç çözümünü üret (S0pop)
3: for e = 1 to pop do
4:   fe0 = fitness hesapla(Se0, pik, di, qi, ai, oi, ny, mlg, v, a)
5: end for
6: for iter = 1 to itermax do
7:   Popülasyondaki fitness değerlerine göre (fiiter-1) rulet tekeri olasılıklarını hesapla
8:   Çaprazlama listesini oluştur (cr)
9:   for i = 1 to pop do
10:     Siiter = Siiter-1
11:   end for
12:   for k = 1 to YukarıYuvarla(pop * cro / 2) do
13:     Rulet tekeri ile elemanı seç ve cr listesine ekle
14:   end for
15:   for b = 1 to cr step 2 do
16:     Çaprazlama operasyonlarını gerçekleştir ve newpop listesi olarak dön
17:   end for
18:   for i = 1 to count(newpop) do
19:     Spop+1iter = newpopi
20:   end for
21:   mr = YukarıYuvarla(pop * mo)
22:   Rastgele mr tane kromozom seç ve mutlist listesine ekle
23:   for b = 1 to mutlist do
24:     Mutasyon operasyonlarını gerçekleştir ve newmrpop listesine ekle
25:   end for
26:   Spop+count(newpop)+1iter = newmrpopi

```

### Algoritma 1 (Devamı) | Ders Programı Çizelgesi için Genetik Algoritma

```

27:   end for
28:   end for
29:   for e = 1 to pop + count(newpop) + count(newmrpop) do
30:     feiter = fitness hesapla(Seiter, pik, di, qi, ai, oi, ny, mlg, v, a)
31:   end for
32:   fiiter değerlerine göre Siiter kromozomlarını sırala
33:   for i = 1 to pop do
34:     Siiter = Siiter
35:     fiiter = fiiter
36:   end for
37:   if f0iter < zg then
38:     Sg ← S0iter
39:     zg ← f0iter
40:   end if
41: end for
42: return Sg
43: end procedure

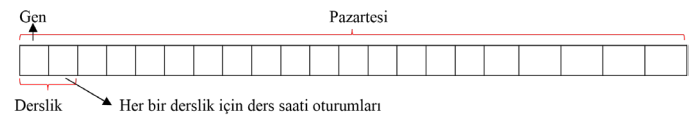
```

Çalışmada kullanılan kromozom modeli derslik sayısı, gün sayısı ve ders oturum saatlerine göre tasarlanmıştır. Bu modele göre; her bir kromozom derslik, gün ve ders oturumunun çarpımından oluşmaktadır. Bu varyasyon ise her bir derslik için her bir ders saati oturumu bir geni temsil etmektedir. Bu sayede kromozomdaki her bir gen diziyi oluşturmaktadır. Örnek vaka olan Osmaniye Korkut Ata Üniversitesi İşletme Bölümü için 1 adet amfi, 1 adet laboratuvar ve 8 adet derslik; öğleden önce ve sonra birer adet olan ders oturumu ve hafta içi olmak üzere beş gün kromozom yapısını oluşturmaktadır. Kromozomun matematiksel gösterimi aşağıda gösterilmektedir.

10(derslik) × 2 (ders oturma saatleri) × 5 (eğitim verilen gün sayısı)

Böylece kromozom yapısı 1 dizi ve 100 elemandan, bir başka deyişle 1 × 100'lük dizilerden oluşmaktadır. Tek bir gün için oluşturulan dizi Şekil 1'de gösterilmektedir (Yılmaz, 2023)

### ŞEKİL 1 | Örnek Kromozom Yapısı



Kaynak: (Yılmaz, 2023)

Bu çalışmada, ebeveyn seçimi için GA'da en çok kullanılan yöntemlerden biri olan rulet tekerleği kullanılmıştır. Kromozom seçimi için sıfır ila toplam uygunluk değeri arasında rastgele bir sayı üretilerek, üretilen sayının çember yani tekerlekteki isabet ettiği parçaya göre kromozom seçimi yapılmıştır. Kromozomların belirlenmesi ile uygunluk değeri daha başarılı olan bireylerin havuzda toplama şansları daha yüksek olmaktadır. Havuzdaki en başarılı bireyler ise ebeveynleri oluşturmaktadır. Rulet tekerleği uygulaması problem yapımıza uygun şekilde adapte edilmiştir.

Rulet tekerleği ile oluşturulan nesil, ebeveynleri içermektedir. Ebeveynler çocukları üretmek üzere çaprazlama işlemine tabi tutulmuştur. Çalışmada "tek nokta çaprazlama" ve "sıralı çaprazlama" operatörleri ayrı ayrı uygulanmış ve sonuca etkileri gözlemlenmiştir. Ayrıca çalışmada mutasyon türlerinden "rastgele seçimli mutasyon" tercih edilmiştir. Seçilen çaprazlama türleri Şekil 2 ve Şekil 3 ile mutasyon ise Şekil 4 ile örneklendirilmiştir.

**ŞEKİL 2 | Tek Nokta Çaprazla**

1.Ebeveyn	1	1	0	1	0	0
2.Ebeveyn	0	1	0	0	1	1
1. Yavru	1	1	0	1	1	1
2. Yavru	0	1	0	0	0	0

**Kaynak:** (Yılmaz, 2023)

**ŞEKİL 3 | Sıralamalı Çaprazlama**

1.Ebeveyn	1	2	3	4	5	6	7	8
2.Ebeveyn	5	8	2	7	1	4	6	3
1. Yavru	7	1	3	4	5	6	8	2
2. Yavru	4	5	2	7	1	6	8	3

**Kaynak:** (Yılmaz, 2023)

**ŞEKİL 4 | Rastgele Seçimli Mutasyon**

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

**Kaynak:** (Yılmaz, 2023)

**V. UYGULAMA**

Bu çalışmada Osmaniye Korkut Ata Üniversitesi İktisadi ve İdari Bilimler Fakültesi İşletme Bölümü için ders programı çizelgeleme problemi ele alınmaktadır. Bölüm müfredatı 8 yarı dönemden oluşmaktadır. Bu çerçeveye göre öğrenciler her dönem en az 6 en fazla 8 ders alabilmektedir. Dersler zorunlu ve seçimsiz olarak kategorize edilmektedir. Zaman çizelgesi beş gün ve günlük 8 saat olarak planlanmaktadır. Ayrıca bölümde 29 adet dersten; 1 adet ders laboratuvar, 1 adet ders amfide, diğer 27 ders ise dersliklerde ve 22 adet öğretim üyesi tarafından okutulmaktadır. Bu derslerden belirlenen sayıda ders çevrim içi şekilde yürütülmekte ve hiçbir ders için mevcut öğrenci sayısı sınıf kapasitesini aşmamaktadır.

**A. KISIT YAPILARI**

Önceki bölümlerde aktarıldığı üzere kısıt yapıları zorunlu, esnek kısıtlar olmak üzere ayrılmaktadır. Bu çalışmada zorunlu kısıtların yanı sıra, COVID-19 pandemisi ve daha sonra doğabilecek herhangi bir salgında kullanılmak üzere esnek kısıtlar geliştirilmiştir. Pandemi ve kapanma süreçlerinde üniversiteler uzaktan eğitim sistemine geçiş yapmış akabinde salgının azalmasıyla bazı üniversiteler hibrit eğitime başlamıştır. Bazı derslerin uzaktan eğitim bazı derslerin ise örgün eğitimde devam etmesi durumunda; gün içerisinde hem çevrim içi hem de yüz yüze derslerin bulunması öğrenciler açısından verimsiz bir süreç ortaya çıkarmıştır. Ayrıca dersliklerin kullanımında ise arka arkaya iki farklı grup öğrencinin aynı dersliği kullanması bulaşı arttırma ihtimalini barındırmaktadır. Yine bulaşı arttırabilecek bir başka durum olan belirli bir saat oturumunda koridordaki tüm sınıfların aktif kullanımı bu çalışmada göz önünde bulundurulmuş ve esnek kısıtlar bu minvalde oluşturulmuştur (Yılmaz, 2023).

**Zorunlu Kısıtlar;**

1. Belirli gün ve zaman dilimi için bir dersliğe birden fazla ders atanamaz.
2. Belirli bir gün ve zaman dilimindeki ders birden fazla dersliğe atanamaz.
3. Bir öğretim üyesi aynı gün ve aynı zaman diliminde birden fazla derslikte bulunamaz.
4. Aynı öğrenci grubuna ait derslerden belirli bir gün ve zaman diliminde birden fazla atama yapılamaz.
5. Dersliklerin kapasitesi, dersliğe atanan öğrenci grubunun sayısından büyük olamaz.

**Esnek Kısıtlar;**

1. Laboratuvar ve amfi dersliklerine sadece bu derslikler için ayrılmış olan dersler atanmalı,
  2. Bir derslikte peş peşe örgün ders olmamalı (çevrim içi dersler dahil değildir),
  3. Bir günde iki farklı saatte (öğleden önce ve öğleden sonra) dersi olan öğretim üyeleri için örgün ders sayısı en fazla 1 olmalı,
  4. Herhangi bir sınıfa ait herhangi bir ders saatindeki ders çevrim içi ise ilgili sınıfın aynı güne atanacak diğer dersi de çevrim içi olmalı. (Çevrim içi ders sayısı 1'den fazla olan sınıflar için)
  5. Aynı ders saatinde koridordaki tüm dersliklere örgün eğitim ataması yapılmamalı (bir koridorda en fazla 3 derslikte örgün ders olmalı),
- Ayrıca Yükseköğretim Kurulu'nun "Yükseköğretim Kurumlarında Uzaktan Öğretime İlişkin Usul ve Esaslar" yönetmeliğinin 6. Maddesi (b) bendine göre; "Aynı dönemde bir ders hem uzaktan hem de örgün öğretim olarak planlanıp öğrencilere tercih imkânı sunulabilir. Bu kapsamda her yarıyılıda eğitim-öğretim dönemi başlamadan en geç dört hafta önce Senato tarafından uygun görülmesi ve Yükseköğretim Kuruluna bildirilmesi halinde, öğretim programlarında her bir yarıyıldaki derslerin (yabancı dil hazırlık dahil) AKTS kredilerine göre en fazla %30'u uzaktan öğretim yoluyla verilebilir." (Yüksek Öğretim Kurumlarında Uzaktan Öğretime İlişkin Usul ve Esaslar Md.6, (b) Bendi) ifadesi göz önünde bulundurulmuştur. Yüzdeler COVID-19 salgını sırasında değiştiği için daha sonra da değiştirilebilir.

**B. AMAÇ FONKSİYONU**

DPÇP için önceki bölümlerde amaç fonksiyonu örnekleri sıralanmaktadır. Bu çalışmada ise, salgın dönemlerinde oluşturulabilecek ders programları çalışma konumuz olduğu için COVID-19 koşullarında oluşturulmuş ders çizelgesinde kısıt ihlalini minimumda tutmak amaç fonksiyonunu oluşturmaktadır. Aynı sınıfa ait derslerin aynı gün, aynı zaman dilimine atanması ve birden fazla ders yükü olan öğretim üyelerinin derslerinin aynı gün ve aynı zaman dilimine atanmasından kaynaklanan çakışmaları önlemek adına çakışmalar büyük bir ceza puanıyla ( $\alpha$ ) çarpılmış ve geri kalan esnek kısıtlar için ihlal sayıları toplanmıştır. Amaç fonksiyonu formülasyonu Eşitlik 14'te gösterilmektedir. Ayrıca, Genetik Algoritmada amaç fonksiyonunu baz alan fitness değerlerinin hesaplanması için izlenen adımlar Algoritma 2'de verilmiştir

$$\text{Amaç Fonksiyonu} = \min[\alpha * (\text{toplam çakışma}) + \text{esnek kısıt ihlal sayıları}] \quad (14)$$

**Algoritma 2 | Fitness Hesabı**

- 1: **procedure** *fitness\_hesapla*( $S_e, p_{ik}, d_i, q_i, a_i, o_i, n_{ij}, m_{lg}, v, \alpha, O, Y$ )
- 2: *cakisma\_ogretim\_uyesi* değerini  $S_e$  ve  $p_{ik}$  kullanarak **hesapla**.
- 3: *cakisma\_ders* değerini  $S_e$  ve  $n_{ij}$  kullanarak **hesapla**.
- 4: *cakisma\_lab\_amfi* değerini  $S_e, a_i$  ve  $o_i$  kullanarak **hesapla**.
- 5: *cakisma\_online* değerini  $S_e, O$  ve  $Y$  kullanarak **hesapla**.
- 6: *cakisma\_online2* değerini  $S_e$  ve  $O$  kullanarak **hesapla**.
- 7: *derslik\_koridor* değerini  $S_e, m_{lg}$  ve  $v$  kullanarak **hesapla**.
- 8: *derslik\_kapasite* değerini  $S_e, d_i$  ve  $q_i$  kullanarak **hesapla**.
- 9:  $f = \alpha * \text{cakisma_ogretim_uyesi} + \alpha * \text{cakisma_ders} + \text{cakisma_lab_amfi} + \text{cakisma_online} + \text{cakisma_online2} + \text{derslik_koridor} + \text{derslik_kapasite}$
- 10: **return**  $f$

**C. UYGULAMA ÖNCESİ GEREKLİ BİLGİLER**

Uygulama öncesinde bölüme ait genel bilgiler Tablo 3'te sunulmaktadır. Pembe olarak işaretlenen dersler çevrim içi olarak yürütülmektedir.



**TABLO 3 | Mevcut Dersler ve Dersi Veren Öğretim Üyeleri**

Departman	Yıl	Ders Kodu	Ders Adı	Ders Süresi	Öğretim Üyesi
IIBF	1	ISL107	İşletme ve İktisat Matematiği I	3	A
IIBF	1	ISL101	İktisada Giriş I	3	B
IIBF	1	ISL103	İşletme Bilimine Giriş I	3	C
IIBF	1	ISL105	Genel Muhasebe	3	D
IIBF	1	TDL101	Türk Dili I	3	E
IIBF	1	YDL101	Yabancı Dil I	3	F
IIBF	2	ATA101	Atatürk İlke ve İnkılap Tarihi	3	G
IIBF	2	ISL203	Yönetim ve Organizasyon	3	H
IIBF	2	ISL215	Envanter ve Bilanço	3	I
IIBF	2	ISL207	İstatistik I	3	i
IIBF	2	ISL219	Anayasa Hukuku	3	J
IIBF	2	ISL213	Pazarlama İlkeleri	3	K
IIBF	2	ISL201	Mikro İktisat	3	L
IIBF	3	ISL345	Yöneylem Araştırması	3	A
IIBF	3	ISL303	Maliyet Muhasebesi	3	M
IIBF	3	ISL305	İşletme Finansı	3	N
IIBF	3	ISL347	Mesleki İngilizce 3	3	Q
IIBF	3	ISL353	Pazarlama Araştırması	3	P
IIBF	3	ISL359	Muhasebe Uygulamaları	3	R
IIBF	3	ISL361	Şirketler Hukuku	3	J
IIBF	4	ISL469	Kamu Maliyesi	3	S
IIBF	4	ISL489	Uluslararası İşletmecilik	3	H
IIBF	4	ISL451	Mali Tablolar Analizi	3	T
IIBF	4	ISL477	Girişimcilik	3	U
IIBF	4	ISL409	Uluslararası Pazarlama	3	C
IIBF	4	ISL447	Vergi Hukuku	3	V
IIBF	4	ISL441	Hizmet Pazarlaması	3	P

**Kaynak:** (Yılmaz, 2023)

Fakültede 2 koridorda yer almak üzere 8 derslik ve 1 laboratuvar ve 1 adet amfi İşletme Bölümü'ne tahsis edilmiştir. Dersliklerin tümü İşletme Bölümü öğretim üyelerinin ofislerinin bulunduğu 2. kat koridorlarında yer almakta ve mevcut öğrenci sayısı herhangi bir sınıfın kapasitesini aşmamaktadır. Laboratuvar ve amfi dersleri sabit tutulmuş olup, ders listesinde gösterilmemiştir. Amfi ve Laboratuvar derslerinin atamalarının hiçbir kısıt ihlaline sebebiyet vermediği varsayılmaktadır.

Bölümde okutulan dersler için ikinci öğretim programı bulunmaması dolayısıyla dersler 08:15-12:00 ve 13:15-17:00 aralıklarında yer alan oturlara ders saatlerine orantılı şekilde tahsis edilmektedir.

## VI. BULGULAR

GA uygulaması için, popülasyon 100 birimlik kromozoma (Şekil 1) derslerin tesadüfi olarak dağıtılmasıyla oluşturulmuştur. Uygunluk değerlerinin hesaplanmasından sonra, nesil oluşturma işlemi, rulet tekerleğinin oluşturulması ile ebeveyn seçimi, çaprazlama ve mutasyon işlemleri gerçekleştirilmiştir. Zorunlu ve esnek kısıtların sağlandığı ders programı oluşturulmuştur. Uygulama ve oluşturulan program çıktıları aşağıda sunulmaktadır.

Uygulama bölümünde bahsedilen problem örneği için matematiksel modelin yanı sıra genetik algoritma kullanıldığında da en iyi çözüm bulunduğu görülmüştür.

50 birimlik popülasyon ve 100 birimlik popülasyon için işlem gören çizelgelemeye ait bilgiler Tablo 4 ve Tablo 5'te verilmektedir.

**TABLO 4 | 50 Kromozom Yığınının Oluşan Popülasyon Çıktıları**

Popülasyon						
50	Çaprazlama Oranı	Mutasyon Oranı	İterasyon Sayısı	En İyi Çözüm	Süre (Saniye)	En İyi İterasyon
0.30	0.01	50	7	8	49	
			100	1	13	88
			150	2	21	83
0.05	50	8	10	41		
		100	3	14	91	
		150	1	17	122	
0.10	50	9	7	44		
		100	5	13	82	
		150	0	24	85	
0.50	0.01	50	4	11	46	
			100	2	21	76
			150	1	18	140
0.05	50	5	8	42		
		100	1	14	78	
		150	1	22	137	
0.10	50	2	9	48		
		100	0	17	84	
		150	0	27	90	
0.80	0.01	50	2	9	42	
			100	3	19	57
			150	2	23	122
0.05	50	2	9	44		
		100	0	17	97	
		150	0	23	79	
0.10	50	1	10	47		
		100	1	17	86	
		150	0	28	128	

**Kaynak:** (Yılmaz, 2023)

Tablo 4'e göre 50 birimlik kromozomdan oluşan popülasyonda en iyi sonuç 0.8 çaprazlama oranı ve 0.05 mutasyon oranıyla 23 saniyede 150 iterasyon içerisinde 79. iterasyonda elde edilen sonuç olmuştur.

Kromozom sayısı arttıkça evrene daha çok yaklaşılabileceği için kromozom sayısı artırılarak program tekrar çalıştırılmıştır.

**TABLO 5 | 100 Kromozom Yığınından Oluşan Popülasyon Çıktıları**

Popülasyon						
100	Çaprazlama Oranı	Mutasyon Oranı	İterasyon Sayısı	En İyi Çözüm	Süre (Saniye)	En İyi İterasyon
	0.30	0.01	50	6	21	49
			100	6	30	84
			150	0	39	120
		0.05	50	12	31	48
			100	2	58	93
			150	0	37	145
		0.10	50	4	32	41
			100	4	60	87
			150	0	79	116
	0.50	0.01	50	3	21	49
			100	3	37	73
			150	0	50	93
		0.05	50	6	17	44
			100	1	33	86
			150	0	51	111
		0.10	50	3	18	45
			100	1	37	65
			150	0	63	97
	0.80	0.01	50	3	24	44
			100	3	48	47
			150	1	76	84
		0.05	50	4	48	43
			100	0	48	73
			150	0	62	117
		0.10	50	2	37	43
			100	0	47	50
			150	0	71	59

Kaynak: (Yılmaz, 2023)

**TABLO 6 | En İyi Sonuca Ait Program**

	PAZARTESİ	SALI	ÇARŞAMBA	PERŞEMBE	CUMA
201	ISL103 Öğr. Üyesi: C Oturum: 09.15	ISL107 Öğr. Üyesi: A Oturum: 09.15	ISL441 Öğr. Üyesi: P Oturum: 13.15	ISL347 Öğr. Üyesi: Q Oturum: 09.15	ATA101 Öğr. Üyesi: G Oturum: 09.15
				YDL101 Öğr. Üyesi: F 13.15	ISL361 Öğr. Üyesi: J Oturum: 13.15
202	ISL219 Öğr. Üyesi: J Oturum: 09.15	ISL303 Öğr. Üyesi: M Oturum: 09.15	ISL345 Öğr. Üyesi: A Oturum: 13.15		
203			ISL213 Öğr. Üyesi: K Oturum: 09.15	ISL305 Öğr. Üyesi: N Oturum: 13.15	ISL101 Öğr. Üyesi: B Oturum: 13.15
204					
205	ISL353 Öğr. Üyesi: P Oturum: 09.15			TDL101 Öğr. Üyesi: E Oturum: 09.15	ISL451 Öğr. Üyesi: T Oturum: 09.15
				ISL409 Öğr. Üyesi: C Oturum: 13.15	ISL447 Öğr. Üyesi: V Oturum: 13.15
206	ISL207 Öğr. Üyesi: I Oturum: 13.15	ISL105 Öğr. Üyesi: D Oturum: 13.15	ISL359 Öğr. Üyesi: R Oturum: 09.15		ISL215 Öğr. Üyesi: I Oturum: 13.15
207	ISL489 Öğr. Üyesi: H Oturum: 09.15	ISL477 Öğr. Üyesi: U Oturum: 09.15	ISL469 Öğr. Üyesi: S Oturum: 09.15		
208		ISL201 Öğr. Üyesi: L Oturum: 09.15		ISL203 Öğr. Üyesi: H Oturum: 09.15	
LAB.			Lab. dersi Oturum: 13.15		
AMFİ	Amfî dersi Oturum: 09.15				

Kaynak: (Yılmaz, 2023)

Tablo 5'te görüldüğü üzere arttırılan popülasyon ile süreler artmış olsa dahi en iyi çözüm, 100 birimlik popülasyon olmak üzere, 0.8 çaprazlama oranı ve 0.1 mutasyon oranı ile 47 saniyede elde edilen sonuç olmuştur. Yine Tablo 5'te görüldüğü üzere kısıtları ve amaç fonksiyonunu sağlayan başka en iyi sonuçlar da mevcuttur. Ancak en kısa sürede tüm şartları sağlayan en iyi çözüm baz alınmaktadır. En iyi çözümde tüm kısıtlar sağlanmış aynı zamanda en kısa sürede en iyi sonuç elde edilmiştir.

Tablo 6'da yeşil ile gösterilen alanlar online dersleri temsil etmektedir. Bu çıktıya göre dersliklerin arka arkaya örgün eğitimde kullanılmasının minimuma indirildiği, laboratuvar ve amfi dersliklerinin sadece o dersliğe ayrılan derslerde kullanıldığı ve koridorlarda tüm dersliklerin örgün eğitim için kullanımda olmadığı görülmektedir. Çizelgede hem zorunlu hem de esnek kısıtların sağlandığı söylenebilir.

Farklı bir senaryoda program çalışıp çalışmadığı test edilmek üzere sınıfların çevrim içi ders dağılımları değiştirilerek analizler tekrarlanmıştır. Çevrim içi ders sayıları 6, 12 ve 15 olarak test edilmiştir. Bu sayılar total derslerin %20, %40 ve %50'sine karşılık gelmektedir. Her bir senaryoda yüzdelikler ayrı ayrı denenmiş ve her bir senaryo için program 10'ar kez çalıştırılarak ortalama süreler ve başarı oranları hesaplanmıştır. Tablo 7, Tablo 8 ve Tablo 9 incelenen senaryoların sonuçlarını sunmaktadır.

İlk senaryoda çevrim içi ders sayısı 6, bir başka deyişle %20 olarak belirlenmiş ve program çalıştırılmıştır. Bu senaryoya göre sonuçlar Tablo 7'de gösterilmektedir.

**TABLO 7 | Senaryo 1 için Program Çıktıları**

Çevrim içi ders sayısı oranı	Çevrim içi ders sayısının sınıflara dağılımları	En İyi Çözüm	Süre (saniye)
20%	2,2,1,1		
	1	0	48,86
	2	0	44,96
	3	0	55,04
	4	0	57,03
	5	0	47,74
	6	0	50,84
	7	0	48,83
	8	0	51,73
	9	0	55,02
	10	0	64,67
	<b>En iyi</b>	<b>0</b>	
	<b>En kötü</b>	<b>0</b>	
	<b>Başarı oranı:</b>	<b>100%</b>	
	<b>Ortalama süre:</b>	<b>52,472</b>	

Kaynak: (Yılmaz, 2023)

Tablo 7'ye göre, derslerin %20'sinin çevrim içi yürütülmesi durumunda ortalama 52.472 saniyede %100 başarıyla program elde edilmektedir.

İkinci senaryoda çevrim içi ders sayısının 12 olduğu, bir başka deyişle derslerin % 40'ının çevrim içi yürütüldüğü farz edilmiştir. Tablo 8'de program çıktıları açıklanmaktadır.

**TABLO 8 | Senaryo 2 için Program Çıktıları**

Çevrim içi ders sayısı oranı	Çevrim içi ders sayısının sınıflara dağılımları	En İyi Çözüm	Süre (saniye)
40%	3,3,3,3		
	1	0	55,98
	2	0	59,86
	3	1	51,69
	4	0	56,85
	5	0	55,25
	6	0	54,29
	7	0	53,97
	8	0	49,19
	9	0	56,49
	10	1	54,76
	<b>En iyi</b>	<b>0</b>	
	<b>En kötü</b>	<b>1</b>	
	<b>Başarı oranı:</b>	<b>80%</b>	
	<b>Ortalama süre:</b>	<b>54,833</b>	

**Kaynak:** (Yılmaz, 2023)

Tablo 8'e göre derslerin %40'ının çevrim içi yürütülmesi durumunda iki optimale yakın sonuçla başarı oranı %80'e gerilerken ortalama süre ise 54.833'e yükselmektedir.

Son senaryoda ise çevrim içi ders sayısı toplamının 15, yani %50 olduğu varsayılmaktadır. Bu senaryo için uygulanan program çıktıları Tablo 9'da sunulmaktadır.

**TABLO 9 | Senaryo 3 için Program Çıktıları**

Çevrim içi ders sayısı oranı	Çevrim içi ders sayısının sınıflara dağılımları	En İyi Çözüm	Süre (saniye)
50%	4,3,4,4		
	1	0	53,15
	2	0	53,90
	3	0	56,93
	4	0	79,19
	5	0	60,38
	6	0	57,10
	7	2	60,12
	8	0	56,24
	9	0	72,19
	10	1	58,63
	<b>En iyi</b>	<b>0</b>	
	<b>En kötü</b>	<b>2</b>	
	<b>Başarı oranı:</b>	<b>80%</b>	
	<b>Ortalama süre:</b>	<b>60,783</b>	

**Kaynak:** (Yılmaz, 2023)

Tablo 9'da ise derslerinin oransal olarak yarısının çevrim içi yürütülmesi durumunda iki optimale yakın sonuçla başarı oranı %80 olurken ortalama süre 60.783'e yükselmektedir. Bu durum göstermektedir ki başarı oranı %80'in altına düşmezken program her orandaki çevrim içi ders sayısına koordine olarak çalışmaktadır. Çalıştırılan üç farklı senaryoya göre program sadece belirlenen kısıtlar üzere değil, kısıtlardaki değişimlere bütünlük şeklinde çalışmaktadır. Burada bahsedilen durum derslerin kaç tanesinin örgün ya da çevrim içi

olduğu program için sağlanmak zorunda olan bir koşul olmamaktadır. Bu sayede herhangi bir sebepten hibrit eğitime geçilmesi durumunda, program herhangi bir kurumda uygulanabilecektir. Modelin geçerliliğini test etmek üzere farklı bir çaprazlama tekniği ve farklı çevrim içi ders sayılarının olduğu senaryolar uygulanmıştır. Bu senaryolara ait sonuçlar aşağıda sunulmaktadır. Öncelikle sıralı çaprazlama uygulanarak modelin geçerliliği test edilmiştir.

**TABLO 10 | 50 Birimlik Popülasyon için Sıralı Çaprazlama**

Popülasyon						
50	Çaprazlama Oranı	Mutasyon Oranı	İterasyon Sayısı	En İyi Çözüm	Süre (Saniye)	En İyi İterasyon
	30	0.01	50	5	12	42
			100	2	20	71
			150	1	22	102
		0.05	50	7	11	38
			100	4	22	96
			150	2	19	105
		0.10	50	8	7	49
			100	5	13	98
			150	0	19	116
	0.50	0.01	50	6	8	50
			100	2	16	91
			150	1	24	140
		0.05	50	5	9	42
			100	1	19	99
			150	0	43	81
		0.10	50	2	16	47
			100	1	14	81
			150	0	23	123
	0.80	0.01	50	2	11	46
			100	0	22	57
			150	0	34	98
		0.05	50	2	14	49
			100	0	37	82
			150	0	60	62
		0.1	50	2	9	37
			100	1	21	71
			150	0	34	64

**Kaynak:** (Yılmaz, 2023)

Tablo 10'da 50 birimlik popülasyon için 0.3, 0.5 ve 0.8 çaprazlama oranlarında; 0.01, 0.05 ve 0.1 mutasyon oranları uygulanmış ve modelin tutarlı olduğu tespit edilmiştir. En iyi çözüm 0.80 çaprazlama oranı, 0.05 mutasyon oranı ve 150 iterasyonda 60 saniyede 62. iterasyonda elde edilmiştir. Aynı işlemler farklı popülasyon miktarında da test edilmek istenerek popülasyon sayısı yükseltmiştir.

**TABLO 11 | 100 Birimlik Popülasyon için Sıralı Çaprazlama**

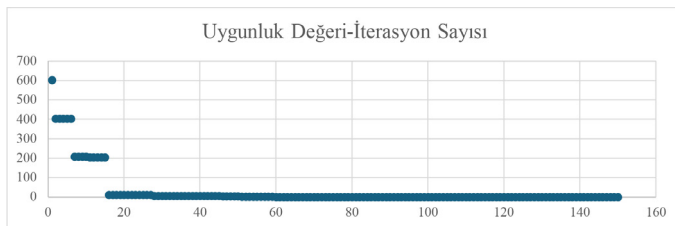
Popülasyon						
100	Çaprazlama Oranı	Mutasyon Oranı	İterasyon Sayısı	En İyi Çözüm	Süre (Saniye)	En İyi İterasyon
0.30	0.01	0.01	50	6	17	28
			100	4	49	82
			150	1	45	125
	0.05	0.05	50	10	17	42
			100	2	43	83
			150	1	85	110
	0.10	0.10	50	4	15	46
			100	2	29	82
			150	0	55	100
0.50	0.01	0.01	50	6	18	47
			100	1	51	91
			150	0	51	135
	0.05	0.05	50	6	26	47
			100	0	54	90
			150	0	97	96
	0.10	0.10	50	4	41	45
			100	2	36	75
			150	0	58	91
0.80	0.01	0.01	50	4	42	41
			100	0	91	61
			150	1	85	79
	0.05	0.05	50	1	29	45
			100	0	64	82
			150	0	84	96
	0.10	0.10	50	2	25	49
			100	0	57	70
			150	0	100	60

**Kaynak:** (Yılmaz, 2023)

Tablo 11’de 100 birimlik popülasyon için 0.3, 0.5 ve 0.8 çaprazlama oranlarında; 0.01, 0.05 ve 0.1 mutasyon oranları uygulanmış ve modelin tutarlılığı farklı popülasyon sayıları için doğrulanmıştır. En iyi çözüm 0.80 çaprazlama oranında, 0.10 mutasyon oranı ve 150 iterasyonda 100 saniyede 60. iterasyonda elde edilmiştir.

Uygulanan oranlar sabit kalsa da farklı bir çaprazlama tekniğinin uygulanması sonucunda model herhangi bir çaprazlama tekniğinde, farklı popülasyon sayılarında; bir başka deyişle farklı bir kombinasyonda da tutarlı sonuçlar vermektedir. Bu durum modelin uygulanabilirliğinin bir başka kanıtı olmaktadır.

İterasyon sayıları ve uygunluk değerlerinin ayrıntılı grafiği ise Tablo 12’de verilmektedir. Tablo 12’ye göre 602 ile başlayan uygunluk değeri kayda değer azalmalarla 60. iterasyonda minimum değere ulaşmaktadır.

**TABLO 12 | Sıralı çaprazlama için uygunluk değerleri ve iterasyon sayıları**

**Kaynak:** (Yılmaz, 2023)

Elde edilen sonuçlar çevrim içi derslerin sayısı ile ilgili bir çıkarımda bulunmaya yeterli olmadığı için, çevrim içi ders sayısının %20, %40 ve %50 olarak varsayıldığı farklı senaryolar uygulanmıştır. Bu senaryolara göre;

**TABLO 13 | Çevrim içi Ders Sayısının Arttırıldığı Senaryolar için Analiz Sonuçları**

Çevrim içi ders sayısı oranı	Çevrim içi ders sayısının sınıflara dağılımları	En İyi Çözüm	Süre (saniye)
20%	2,2,1,1		
	1	0	79,28
	2	0	70,92
	3	0	62,54
	4	0	68,29
	5	0	62,75
	6	0	60,55
	7	0	59,45
	8	0	65,73
	9	0	64,90
	10	0	63,73
<b>En iyi</b>		<b>0</b>	
<b>En kötü</b>		<b>0</b>	
<b>Başarı oranı:</b>		<b>100%</b>	
		<b>Ortalama süre:</b>	<b>65,814</b>
40%	3,3,3,3		
	1	0	80,46
	2	0	79,04
	3	0	75,28
	4	0	92,93
	5	0	82,32
	6	0	78,44
	7	0	82,05
	8	0	88,01
	9	0	78,01
	10	0	86,54
<b>En iyi</b>		<b>0</b>	
<b>En kötü</b>		<b>0</b>	
<b>Başarı oranı:</b>		<b>100%</b>	
		<b>Ortalama süre:</b>	<b>82,308</b>
50%	4,3,4,4		
	1	0	97,81
	2	0	85,18
	3	0	88,52
	4	0	86,42
	5	0	87,30
	6	0	85,02
	7	0	88,46
	8	0	90,45
	9	0	89,08
	10	0	90,56
<b>En iyi</b>		<b>0</b>	
<b>En kötü</b>		<b>0</b>	
<b>Başarı oranı:</b>		<b>100%</b>	
		<b>Ortalama süre:</b>	<b>88,88</b>

**Kaynak:** (Yılmaz, 2023)

Tablo 13’teki veriler oluşturulurken çevrim içi ders sayıları 6 (%20), 12 (%40) ve 15 (%50) varsayımı kullanılmıştır. Sıralı çaprazlama kullanılan bu analiz için çaprazlama oranı 0.8, mutasyon oranı ise 0.1

olarak alınmıştır. Ayrıca popülasyon sayısı 100 ve iterasyon sayısı 150 olarak işleme alınmıştır. Bu oranların seçilmesinin nedeni ise en hızlı çözümü veren kombinasyonlar olması ve bu kombinasyonlar üzerinden sürelerin karşılaştırılmasıdır.

Her üç senaryo için de başarı oranı %100'dür ancak ortalama süreler farklılık göstermektedir. Çevrim içi derslerin sayısının artması kayda değer bir süre artışına sebep olmaktadır.

## SONUÇ VE ÖNERİLER

Bu çalışmadaki DPÇP, herhangi bir salgın sonucunda bulaşın önüne geçilmesi, temasın azaltılması, bina içi dolaşımın azaltılması gibi durumları göz önünde bulundurarak herhangi bir pandemi ve salgın döneminde uygulanmak üzere tasarlanmıştır. Bu yaklaşım için zorunlu kısıtlar çıkışmaların engellendiği ve sınıf kapasitelerinin aşılmadığı kısıtlarla çalışılmıştır. Çalışmanın özgünlüğü ise esnek kısıtlardan kaynaklanmaktadır. Esnek kısıtlarda ise laboratuvar ve amfi derslerinin bahsi geçen dersliklere atanması kısıtı ile derslerin amacına uygun dersliklere atanması, bir derslikte peş peşe örgün ders olmaması kısıtı ile dersliklerin dezenfekte edilmesi için uygun zaman tanınması, bir günde iki dersi olan öğretim üyelerinin örgün derslerinin aynı gün olmaması kısıtıyla salgın döneminde uygulanan evde kal politikasının izlenmesi ve temasın azaltılması, bir sınıfa ait derslerin oturumlarından birinin çevrim içi ise diğerinin de çevrim içi olması kısıtı ile öğrencilerin yine evde kal politikası ile bina içerisinde dolaşımının azaltılması ve bir koridorda aynı anda en fazla üç dersliğin örgün eğitim için kullanılması ile yine bina içi dolaşımın azaltılması amaçlanmaktadır. Bu minvalde, amaç fonksiyonu ile zorunlu ve esnek kısıtlar sağlanmaya çalışılmıştır. Çalışmanın belirlenen amaç fonksiyonu ve kısıtlarını göz önüne alarak bir matematiksel model oluşturulmuştur. Matematiksel model ile tanımlanan problemin çözümü GA kullanılarak gerçekleştirilmiştir.

Bölümde amfi ve laboratuvar sabit dersleri ile beraber 29 adet ders bulunmaktadır. Bölümdeki 27 ders 22 öğretim üyesi tarafından okutulmaktadır. Bölümün kullanımına tahsis edilmiş bir adet amfi, bir adet laboratuvar bulunmaktadır. 201, 202, 203 ve 204 numaralı derslikler ilk koridorda, 205, 206, 207 ve 208 numaralı derslikler ikinci koridorda olmak üzere toplam 8 derslik bulunmaktadır. "Yükseköğretim Kurumlarında Uzaktan Öğretime İlişkin Usul ve Esaslar" yönetmeliğinin 6. Maddesi (b) bendine göre; "Aynı dönemde bir ders hem uzaktan hem de örgün öğretim olarak planlanıp öğrencilere tercih imkânı sunulabilir. Bu kapsamda her yarıyıldaki eğitim-öğretim dönemi başlamadan en geç dört hafta önce Senato tarafından uygun programlarında her bir yarıyıldaki derslerin (yabancı dil hazırlık dahil) AKTS kredilerine göre en fazla %30'u uzaktan öğretim yoluyla verilebilir." (Yüksek Öğretim Kurumlarında Uzaktan Öğretime İlişkin Usul ve Esaslar Md.6, (b) Bendi) kapsamında bölümde derslerin %30'u çevrim içi olarak verilmektedir.

Çalışmada Genetik Algoritma kullanılmıştır. GA literatürde çok sık kullanılan ve iyi sonuçlar verdiği bilinen bir sezgisel yöntem olduğu için tercih edilmiştir. Kromozom yapısı derslikler, derslik oturumları ve eğitim verilen günlerin çarpımıyla oluşturulmuştur. Bu çarpımla 1x20'lik diziler eğitim verilen her bir gün için derslik ve ders oturumlarını oluştururken, 1x100'lük dizi ise kromozom yapısını oluşturmaktadır. Kromozomdaki genler her derslik için ders oturumlarını temsil etmektedir. GA uygulamasında popülasyon rastgele olarak türetilmiştir. Ebeveyn havuzu rulet tekerleği yöntemi ile oluşturulmuştur. Çalışmada iki farklı çaprazlama yöntemi (tek nokta çaprazlama ve sıralı çaprazlama) kullanılmıştır. Bu sayede modelin tutarlılığı test edilmiştir. Mutasyon ise kromozom uzunluğu içerisinde rastsal olarak seçilen iki genin birbirleriyle yer değiştirilmesiyle oluşturulmuştur.

Modelin tutarlılığının test edilmesi için operatörlerde farklı teknikler kullanılarak, kurumdan kuruma fark etmeksizin uygulanabilirliğini ölçmek içinse çevrim içi ders sayıları, popülasyon sayısı ve GA operatörlerinin oranları değiştirilerek farklı senaryolar denenmiş ve modelin tutarlılığı ve geçerliliği test edilmiştir.

İlk olarak 50 birimlik popülasyon ile çizelge oluşturulmuş ve 50 birimlik kromozomdan oluşan popülasyonda en iyi sonuç 0.8 çaprazlama oranı ve 0.05 mutasyon oranıyla 23 saniyede 150 iterasyon içerisinde 79. iterasyonda elde edilen sonuç olmuştur. Daha sonra

100 birimlik popülasyon ile çalışılmış, 0.8 çaprazlama oranı ve 0.1 mutasyon oranı ile 47 saniyede elde edilen sonuç en iyi sonuç olmuştur. En iyi çözümde tüm kısıtlar sağlanmış aynı zamanda kabul edilebilir kısıtlarda bir sürede en iyi sonuç elde edilmiştir. 100 iterasyonluk çözüm sürecinde 50. iterasyonda en iyi çözüme ulaşılmıştır.

Herhangi bir salgın veya kapanma durumunda çevrim içi ders yüzdeliğinin değişmesi durumu göz önünde tutularak %30 olarak hesaplanan oran %20 üzerinden tekrar işleme tabi tutulmuştur. Uygulamaya konu olan İşletme Bölümü için bu oran 6 dersin çevrim içi yürütülmesine karşılık gelmektedir. 6 dersin çevrim içi yürütüldüğü durumda %100 başarı oranı ile en iyi çözüm ortalama 52.472 saniyede elde edilmektedir. %40 oranında çevrim içi ders olması durumunda ise başarı oranı iki optimale yakın sonuçla %80'e gerilerken ortalama süre ise ortalama 54.833'e yükselmektedir. Derslerin yarısının çevrim içi yürütülmesi senaryosu test edildiğinde ise iki optimale yakın sonuçla başarı oranı %80 olurken süre ortalama 54.833'e yükselmektedir. Bu durum başarı oranı %80'in altına düşmezken, programın her orandaki çevrim içi ders sayısına koordine olarak çalıştığını göstermektedir. Çalıştırılan üç farklı senaryoya göre GA sadece belirlenen kısıtlar üzere değil, kısıtlardaki değişimlere entegre şekilde çalışmaktadır.

Modelin tutarlılığını test etmek üzere çaprazlama tekniği değiştirilmiştir. 50 birimlik popülasyon için sıralı çaprazlama kullanıldığında, en iyi sonuç 0.80 çaprazlama oranı, 0.05 mutasyon oranı ve 150 iterasyon ile 60 saniyede elde edilmiştir. 100 birimlik popülasyon ile algoritma tekrar çalıştırılmış ve en iyi çözüm 0.80 çaprazlama oranı, 0.10 mutasyon oranı ve 150 iterasyon ile 100 saniyede elde edilmiştir. Daha sonra online ders yükleri değiştirilerek 3 farklı senaryo sıralı çaprazlama kullanılan modele uygulanmıştır. %20 çevrim içi ders olması halinde %100 başarı oranı ile ortalama 65 saniyede, %40 çevrim içi ders olması halinde %100 başarı ile ortalama 82 saniyede ve %50 çevrim içi ders olması halinde ise %100 başarı ile ortalama 88 saniyede en iyi çözüme ulaşılmaktadır. Çaprazlama türlerini karşılaştırdığımızda ise sıralı çaprazlamanın süreleri tek noktalı çaprazlamaya göre arttırdığı ancak başarı oranlarının sıralı çaprazlamada daha yüksek olduğu söylenebilir.

Bu çalışma, salgınların Dünya ve Türkiye genelinde etkin olduğu düzende, herhangi bir kapanma durumunda ders programının amaç fonksiyonunun literatürde geçen verimlilik, etkinlik, öğrenci ya da öğretim üyeleri memnuniyeti yerine derslikleri baz alarak en az temas, en az bulaş sağlama noktasında literatüre katkı sağlamaktadır. Böylece eğitim öğretim herhangi bir aksaklığa uğramadan devamı esas alınabilecektir.

Gelecekteki çalışmalarda araştırmacılar, salgın dönemindeki teması azaltan, aynı zamanda verimlilik ya da memnuniyet artırılabilir bir DPÇ oluşturabilir. Problem yapısı öğretim üyelerinin veya öğrencilerin dersi anlatmak ya da almak istedikleri gün tercihleriyle genişletilebilir. Sayısal/sözel, zorunlu/seçmeli, mesleki/temel bilim gibi ayrımlara göre ders planı oluşturulması modele eklenebilir. Problem fakülte bazında ders programı oluşturacak şekilde genişletilebilir. Pandemi kaynaklı farklı koşullar (örneğin derslerin küçük gruplar halinde yapılması) modele eklenebilir. Çözüm yaklaşımlarında ise, farklı metasezgisel algoritmalar (örneğin karınca kolonisi, parçacık sürü optimizasyonu gibi) ve bu algoritmaların hibrit varyantları bu çalışmada önerilen probleme özgü durumlar dikkate alınacak şekilde geliştirilebilir.

## KAYNAKÇA

- [1] Abduljabbar, I. A., & Abdullah, S. M. (2022). An evolutionary algorithm for solving academic courses timetable scheduling problem. *Baghdad Science Journal*, 19(2), (s. 399-408). <https://doi.org/10.21123/bsj.2022.19.2.0399>
- [2] Ahmad, I. R., Sufahani, S., Ali, M., & Razali, S. N. (2018). A heuristics approach for classroom scheduling using. *Journal of Physics: Conference Series*, 995(1).
- [3] Alnowaini, G., & Aljomai, A. A. (2021). Genetic algorithm for solving university course timetabling problem using dynamic chromosomes. In *2021 International Conference of Technology, Science and Administration* (s. 1-6). IEEE. <https://doi.org/10.1109/ICTSA52017.2021.9406539>
- [4] Altunay, H., & Eren, T. (2017). Ders programı çizelgeleme problemi için bir literatür taraması. *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, 23(1), (s. 55-70). <https://doi.org/10.5505/pajes.2016.37233>
- [5] Alvarez-Valdes, R., Crespo, E., & Tamarit, J. M. (2002). Design and implementation of a course scheduling system using tabu search. *European Journal of Operational Research*, 137(3), (s. 512-523). [https://doi.org/10.1016/S0377-2217\(01\)00091-1](https://doi.org/10.1016/S0377-2217(01)00091-1)

- [6] Aki, O. (2020). University course timetabling using genetic algorithms. In International Scientific Conference (UNITECH), (s. Vol. 1, p. 390).
- [7] Amrulloh, A., & Sela, E. (2021). Optimization of course scheduling using genetic algorithm and tabu search. Doctoral dissertation, Universitas Teknologi Yogyakarta.
- [8] Ansari, R., & Saubari, N. (2020). Application of genetic algorithm concept on course scheduling. In IOP Conference Series: Materials Science and Engineering (Vol. 821, No. 1, p. 012043). IOP Publishing.
- [9] Bagley, J. D., The Behavior of Adaptive Systems Which Emplly Genetic and Correlation Algorithms, Doktora Tezi, University of Michigan, 1967.
- [10] Behrenk, A. B., Güçlükol Ergin, S., & Toy, A. Ö. (2022, Ekim). Course scheduling problem and real-life implementation. In *The International Symposium for Production Research* (s. 749-758). Cham: Springer International Publishing.
- [11] Bosworth, J. L., Foo, N. Y., & Zeigler, B. P. (1972). Comparison of genetic algorithms with conjugate gradient methods (No. NASA-CR-2093). NASA.
- [12] Burke, E., & Petrovic, S. (2002). Recent research directions in automated timetabling. *European Journal of Operational Research*, 140(2) (s. 266-280). [https://doi.org/10.1016/S0377-2217\(02\)00069-3](https://doi.org/10.1016/S0377-2217(02)00069-3)
- [13] Cavicchio, D. J. (1970). Adaptive search using simulated evolution.
- [14] Chaouachi, J., & Harrabi, O. (2022). Toward artificial intelligence tools for solving the real world problems: effective hybrid genetic algorithms proposal. In *Advances in Selected Artificial Intelligence Areas: World Outstanding Women in Artificial Intelligence* (s. 231-254). Cham: Springer International Publishing.
- [15] Chen, X., Yue, X. G., Li, R., Zhumadillayeva, A., & Liu, R. (2021). Design and application of an improved genetic algorithm to a class scheduling system. *International Journal of Emerging Technologies in Learning*, 16(1), (s. 44-59). <https://doi.org/10.3991/ijet.v16i01.18225>
- [16] Cruz-Rosales, M. H., Cruz-Chávez, M. A., Alonso-Pecina, F., Peralta-Abarca, J. D., Ávila-Melgar, E. Y., Martínez-Bahena, B., & Enríquez-Urbano, J. (2022). Metaheuristic with cooperative processes for the university course timetabling problem. *Applied Sciences*, 12(2), 542.
- [17] Çolak, R., & Yiğit, T. (2021). Üniversite ders çizelgeleme probleminin genetik algoritma ile optimizasyonu. *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, 9(6), (s.150-166). <https://doi.org/10.29130/dubited.1012132>
- [18] Dele, O. A. (2019). An ant colony algorithm based system for allocating course timetable in federal polytechnic Bali, Taraba State, Nigeria. *Bakundi Journal of Technology, Agriculture and Entrepreneurship*, 1(1).
- [19] Duan, Y., & Lu, W. (2021). Automatic course scheduling system in universities based on hybrid genetic-ant colony algorithm. In *Journal of Physics: Conference Series*. 2066(1), (s. 012079). IOP Publishing. <https://doi.org/10.1088/1742-6596/2066/1/012079>
- [20] Eren, T., Taş, C., & Bedir, N. (2018). 0-1 tamsayılı programlama ile ders programı çizelgeleme probleminin çözümü: bir yükseköğretim kurumunda uygulama. *Harran Üniversitesi Mühendislik Dergisi*: 3(3), (s. 166-175).
- [21] Ewi, E. I., & Radiles, H. (2023). Mitigasi premature convergence pada genetic algorithm menggunakan metoda dynamics growth population dalam kasus university course scheduling. *JEKIN-Jurnal Teknik Informatika*, 3(1), (s. 33-44). <https://doi.org/10.58794/jekin.v3i1.486>
- [22] Fedkin, E., Denissova, N., Krak, I., & Dyomina, I. (2021). Automation of scheduling training sessions in educational institutions using genetic algorithms. In *2021 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, (s. 278-283). IEEE. <https://doi.org/10.1109/IDAACS53288.2021.9660939>
- [23] Frantz, D. R. (1972). Non-linearities in genetic adaptive search. Doktora Tezi, University of Michigan.
- [24] Goldberg, D. E. (1989). *Genetic algorithms in search, optimization and machine learning*. Addison-Wesley Longman Publishing.
- [25] Gozali, A. A., Kurniawan, B., Weng, W., & Fujimura, S. (2020). Solving University course Timetabling Problem Using Localized Island Model Genetic Algorithm with Dual Dynamic Migration Policy. *Transactions on Electrical and Electronic Engineering*, 15, 389-400.
- [26] Hollstien, R. (1971). *Artificial genetic adaptation in computer control systems*. University of Michigan.
- [27] Hossain, S. I., Akhand, M. A., Shuvo, M. I., Siddique, N., & Adeli, H. (2019). Optimization of University course scheduling problem using particle swarm optimization with selective search. *Expert Systems with Applications*, 127, 9-24.
- [28] Hu, Y., Wang, S., & Duan, Z. (2021). Using dynamic search mandatory genetic algorithm to solve the university course timetabling problem considering walking distance. In *Proceedings of the Fifteenth International Conference on Management Science and Engineering Management*, (s. 34-45). Springer, Cham. [https://doi.org/10.1007/978-3-030-79203-9\\_4](https://doi.org/10.1007/978-3-030-79203-9_4)
- [29] Huang, Q. & Wang, Y. (2022). Application of Genetic Algorithm in University Teaching Management System. In *Innovative Computing*, (613-620). Springer, Singapore.
- [30] Jiang, C. B., & Liu, H. (2019). A course scheduling algorithm based on improved genetic algorithm with multi-objective constrains. In 2019 Eleventh International Conference on Advanced Computational Intelligence (ICACI) (s. 202-206). IEEE.
- [31] Junjun, Z., Hexia, Y., Oyam, D. M., & Yi, W. (2022). Design and implementation of intelligent course scheduling system for deep integration of education and teaching. *Frontiers in Educational Research*, 5(19), (s. 81-89). <https://doi.org/10.25236/FER.2022.051915>.
- [32] Kakkur, M. K., Singla, J., Garg, N., Gupta, G., Srivasta, P., & Kumar, A. (2021, Ağustos). Class schedule generation using evolutionary algorithms. In *Journal of Physics: Conference Series* 1950(1), (s. 012067). IOP Publishing. <https://doi.org/10.1088/1742-6596/1950/1/012067>
- [33] Kamaşlı Öztürk, Z., Kasımbeyli, N., Özdemir, M. S., Acar, M. S., & Özçetin, E. (2015). Kullanıcı tercihlerinin dikkate alınması durumunda üniversite ders çizelgeleme problemi. *Endüstri Mühendisliği Dergisi*, 27(1), (s. 2-18).
- [34] Kaynar, O., & Yurtsal, A. (2019). Ders programı çizelgeleme probleminin genetik algoritma ile optimizasyonu. *Journal of Information Systems and Management Research*, 9-14.
- [35] Kristiadi, D., & Hartanto, R. (2019). Genetic algorithm for lecturing schedule optimization. *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)* 13(1), 83-94.
- [36] Li, T., Xie, Q., & Zhang, H. (2022). Design of college scheduling algorithm based on improved genetic ant colony hybrid optimization. *Security and Communication Networks*. <https://doi.org/10.1155/2022/2565639>
- [37] Luo, X., Sun, Y., Liu, X., Gao, Y., Sun, H., & Liu, Y. (2022). Course timetable optimization for a university teaching building considering the building energy efficiency and time-varying thermal perception of students. *Building and Environment*, 219, 109175. <https://doi.org/10.1016/j.buildenv.2022.109175>
- [38] Mahlous, A. R., & Mahlous, H. (2023). Student timetabling genetic algorithm accounting for student preferences. *PeerJ Computer Science*, 9, e1200. <https://doi.org/10.7717/peerj-cs.1200>.
- [39] Mahmud, A. (2021). Highly Constrained University Class Scheduling using Ant Colony Optimization. *International Journal of Computer Science & Information Technology*, 13.
- [40] Martin, V. F., & Peluffo-Ordóñez, H. (2022). Virtual and face-to-face course timetabling using multiobjective genetic algorithms based on dynamic gene spaces. *Universidad Internacional de Valencia*.
- [41] Mauluddin, S., Ikbali, I., & Nursikuwagus, A. (2020). Complexity and performance comparison of genetic algorithm and ant colony for best solution timetable class. *Journal of Engineering Science and Technology*, 15(1), 278-292.
- [42] Modibbo, U. M., Umar, I., Mijinyawa, M., & Hafisu, R. (2019). Genetic algorithm for solving university timetabling problem. *Amity Journal of Computational Sciences (AJCS)*, 3(1), 43-50.
- [43] Muklason, A., Irianti, R. G., & Marom, A. (2019). Automated course timetabling optimization using tabu-variable neighborhood search based hyper-heuristic algorithm. *Procedia Computer Science*, 161, 656-664.
- [44] Nasien, D. & Andi, A. (2022). Optimization of genetic algorithm in courses scheduling. *IT Journal Research and Development (ITJRD)*, 6(2), (s. 151-161). <https://doi.org/10.25299/itjrd.2022.7896>
- [45] Nugroho, A. K., Permadi, I., & Yasifa, A. R. (2022). Optimizing course scheduling of engineering unsoed using genetic algorithms. *JITK (Jurnal Ilmu Pengetahuan dan Teknologi Komputer)*, 7(2), (s. 91-98). <https://doi.org/10.33480/jitk.v7i2.2262>
- [46] Özyandı, G. (2010). *Ders çizelgeleme probleminin 0-1 tamsayılı programlama tabanlı uygulaması*. Gazi Üniversitesi Yüksek Lisans Tezi.
- [47] Pérez, E. C., Rios, O. M., Bautista, D. P., Sanchez, S. S., & Acevedo, F. A. (2021). A genetic algorithm solution for scheduling problem. In *2021 XVII International Engineering Congress (CONIIN)* (s. 1-10). IEEE. <https://doi.org/10.1109/CONIIN54356.2021.9634725>
- [48] Pinedo, M. (2008). *Scheduling: Theory, Algorithms, and Systems*. New York: Prentice Hall: 3rd Edition.
- [49] Pongcharoen, P., Promtet, W., Yenradee, P., & Hicks, C. (2008). Stochastic optimisation timetabling tool for university course scheduling. *International Journal of Production Economics*, 2 (112), (s. 903-918).
- [50] Ren, X., & Li, C. (2022). Computer intelligent course scheduling system based on deep learning. In *2022 International Conference on Knowledge Engineering and Communication Systems (ICKES)*, (s. 1-5). Chickballapur, India. <https://doi.org/10.1109/ICKES56523.2022.10060177>
- [51] Rezaeipannah, A., Matoori, S. S., & Ahmadi, G. (2021). A hybrid algorithm for the university course timetabling problem using the improved parallel genetic algorithm and local search. *Applied Intelligence: The International Journal of Research on Intelligent Systems for Real Life Complex Problems*, 51(1), (s. 467-492). <https://doi.org/10.1007/s10489-020-01833-x>
- [52] Rodprasert, N., Taetragee, U., & Akkarajitsakul, K. (2023). Online/offline course and multiple lecturers scheduling using meta-heuristic approaches. In *Proceedings of the 2023 9th International Conference on Computer Technology Applications*, (s. 166-171). <https://doi.org/10.1145/3605423.3605440>
- [53] Rosenberg, R.S. 1967. Simulation of genetic populations with biochemical properties. Doktora Tezi, University of Michigan, Ann Harbor Michigan.
- [54] Sakal, J., Fieldsend, J. E., & Keedwell, E. (2021). Learning assignment order in an ant colony optimiser for the university course timetabling problem. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, (s. 77-78).
- [55] Sari, R., Ramdhanian, K. F., & Pumomo, R. (2022). Team-teaching-based course scheduling using genetic algorithm. *PIKSEL: Penelitian Ilmu Komputer Sistem Embedded and Logic*, 10(1), (s. 55-66). <https://doi.org/10.33558/piksel.v10i1.4416>
- [56] Shuai, C. J. (2021). Design of automatic course arrangement system for electronic engineering teaching based on monte carlo genetic algorithm. *Security and Communication Networks*. (s. 1-11). <https://doi.org/10.1155/2021/3564722>

- [57] Subagio, R. T., Putri, T. E., Sokibi, P., & Harahap, S. Z. (2021). Application of genetic algorithm to optimize lecture scheduling based on lecturers' teaching day willingness. *In Journal of Physics: Conference Series*. 1842(1) 012007. IOP Publishing. <https://doi.org/10.1088/1742-6596/1842/1/012007>
- [58] Sun, Y., Luo, X., & Liu, X. (2021). Optimization of a university timetable considering building energy efficiency: an approach based on the building controls virtual test bed platform using a genetic algorithm. *Journal of Building Engineering*, 35, 102095. <https://doi.org/10.1016/j.jobee.2020.102095>.
- [59] Sun, G., & Li, Y. (2020). Research and analysis of course arrangement based on genetic algorithm. *Journal of Physics: Conference Series*, 1650 (3), 032050. IOP Publishing.
- [60] Susan, S., & Bhutani, A. (2019). A novel memetic algorithm incorporating greedy stochastic local search mutation for course scheduling. *CSE/EUC*, 254-259.
- [61] Szea, S. N., Kuan, H., Chiewa, K. L., Tionga, W. K., & Hengb, C. S. (2020). Heuristic Algorithm for Multi-Location Lecture Timetabling. *Advanced Science Engineering Information Technology*, 10(2), 455-460.
- [62] Şen, Z. (2004). *Genetik algoritmalar ve eniyileme yöntemleri*. İstanbul: Su Vakfı.
- [63] Şimşek, A. B. (2021). A course timetabling formulation under circumstances of online education. *Journal of Turkish Operations Management*, (5)2, (s. 781-791).
- [64] Tan, J. S., Goh, S. L., Sura, S., Kendall, G., & Sabar, N. R. (2021). Hybrid particle swarm optimization with particle elimination for the high school timetabling problem. *Evolutionary Intelligence*, 14(4), 1915-1930.
- [65] Taşkın, Ç., & Emel, G. G. (2009). *Sayısal yöntemlerde genetik algoritmalar*. Bursa: Alfa Aktüel.
- [66] Thakare, S., Nikam, T., & Patil, M. (2020). Automated Timetable Generation using Genetic Algorithm. *International Journal of Engineering Research & Technology (IJERT)*, 9 (07), 1425-1427.
- [67] Thang, H. Q., Giang, V. T., Son, N. T., & Anh, B. N. (2023). Teaching assignment based on nash equilibrium and genetic algorithm. *In 2023 IEEE Symposium on Industrial Electronics & Applications (ISIEA)* (s. 1-7). IEEE. <https://doi.org/10.1109/ISIEA58478.2023.10212338>
- [68] Tian, R., Si, H., Guo, Z., Zhao, X., & Feng Y. (2021). Realization of course scheduling system based on improved genetic algorithm. *In 2021 16th International Conference on Computer Science & Education (ICCSE)*. (s. 1072-1077). IEEE. <https://doi.org/10.1109/ICCSE51940.2021.9569620>
- [69] Topcu, İ., & Kabak, Ö. (2021). Yöneyim araştırması ders notları. *İstanbul Teknik Üniversitesi*, 6-7. Erişim adresi <https://web.itu.edu.tr/topcuil/ya/END331.pdf>
- [70] Trenggonowati, D. L., Herlina, L., Febianti, E., Ilhami, M. A., Muharni, Y., Kurniawan, B., & Irman, A. (2022). Bibliometric analysis of university timetabling using publish and perish. *In Conference on Broad Exposure to Science and Technology 2021 (BEST 2021)* (s. 307-311). Atlantis Press. <https://doi.org/10.2991/aer.k.220131.047>
- [71] Tung Ngo, S., Jafreezal, J., Hoang Nguyen, G., & Ngoc Bui, A. (2021). A genetic algorithm for multi-objective optimization in complex course timetabling. *In 2021 10th International Conference on Software and Computer Applications*, (s. 229-237). <https://doi.org/10.1145/3457784.3457821>
- [72] Wang, Y., & Lei, A. (2018). Design and research of course arranging system based on niche improved genetic algorithm. *International Journal of New Developments in Engineering and Society*, 2(1), 33-38.
- [73] Wang, P., & Huang, J. (2023, Şubat). Research on multi-objective course scheduling method in colleges based on epidemic prevention and control. *In 2023 IEEE 6th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*. 6, (s. 683-687). IEEE. <https://doi.org/10.1109/ITNEC56291.2023.10082461>
- [74] Weinberg, R., & Berkus, M. (1971). Computer simulation of a living cell: Part I. *International Journal of Bio-Medical Computing*, 2(2), (s. 95-120).
- [75] Wong, C. H., Goh, S. L., & Likoh, J. (2022). A genetic algorithm for the real-world university course timetabling problem. *In 2022 IEEE 18th International Colloquium on Signal Processing & Applications (CSPA)* (s. 46-50). IEEE. <https://doi.org/10.1109/CSPA55076.2022.9781907>
- [76] Wren, A. (1995). Scheduling, timetabling and rostering - a special relationship?. *In International conference on the practice and theory of automated timetabling* (s. 46-75). Berlin: Springer.
- [77] Xu, J. (2021). Improved genetic algorithm to solve the scheduling problem of college English courses. *Complexity*. (s. 1-11). <https://doi.org/10.1155/2021/7252719>
- [78] Yang, Y., Gao, W., & Gao, Y. (2017). Mathematical modeling and system design of timetabling problem based on improved GA. *In 2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)* (s. 214-220). IEEE.
- [79] Yang, Z. (2021). Application of multidirectional mutation genetic algorithm and its optimization neural network in intelligent optimization of english teaching courses. *Computational Intelligence and Neuroscience*. <https://doi.org/10.1155/2021/4297600>
- [80] Yang, Z. (2022). Research on college english classroom teaching model based on adaptive genetic algorithm. *Computational Intelligence and Neuroscience*. <https://doi.org/10.1155/2022/9527070>
- [81] Yılmaz, T. N. (2023). Ders programı çizelgeleme probleminin genetik algoritma ile çözümü: bir uygulama (Yayımlanmamış yüksek lisans tezi). *Osmaniye Korkut Ata Üniversitesi Lisansüstü Eğitim Enstitüsü*.
- [82] Yüksek Öğretim Kurulu. Yüksek Öğretim Kurumlarında Uzaktan Öğretime İlişkin Usul ve Esaslar Md.6, (b) Bendi. Erişim Tarihi: 07.01.2022 Erişim Adresi: [https://www.yok.gov.tr/Documents/Kurumsal/egitim\\_ogretim\\_daiesi/Uzaktan\\_ogretim/yuksekogretim-kurumlarinda-uzaktan-ogretime-iliskin-usul-ve-esaslar.pdf](https://www.yok.gov.tr/Documents/Kurumsal/egitim_ogretim_daiesi/Uzaktan_ogretim/yuksekogretim-kurumlarinda-uzaktan-ogretime-iliskin-usul-ve-esaslar.pdf)
- [83] Zaulir, Z. M., Abdülaziz, N. L., & Aizam, N. A. H. (2022). A general mathematical model for university courses timetabling: implementation to a public university in malaysia. *Malaysian Journal of Fundamental and Applied Sciences*, 18(1), (s. 82-94). <https://doi.org/10.11113/mjfas.v18n1.2408>
- [84] Zhang, Q. (2022). An optimized solution to the course scheduling problem in universities under an improved genetic algorithm. *Journal of Intelligent Systems*, 31(1), (s. 1065-1073). <https://doi.org/10.1515/jisys-2022-0114>
- [85] Zhang, Y., Li, C., & Zhang, Y. (2021). Intelligent course scheduling scheme in high school for elective course system in college entrance examination. *In 2021 International Conference on Education, Information Management and Service Science (EIMSS)* (s. 324-332). IEEE. <https://doi.org/10.1109/EIMSS53851.2021.00077>
- [86] Zheng, H., Peng, Y., Guo, J., & Chen, Y. C. (2022). Course scheduling algorithm based on improved binary cuckoo search. *The Journal of Supercomputing*, 78(9), (s. 11895-11920). <https://doi.org/10.1007/s11227-022-04341-6>

## EXTENDED ABSTRACT

The course scheduling problem has been the focus of researchers' attention since the 1960s. Course scheduling problem is one of the research topics that does not have a single correct result. In this study, the scheduling problem is solved by using Genetic Algorithm and the optimal result is found.

Course scheduling problems consist of the objective function and constraints. The objective function aims to reach the optimal values of the variables in the mathematical model. It is set as maximum or minimum according to the problem structure. Hard constraints are the constraints without which the schedule cannot be formed. Soft constraints, on the other hand, consist of constraints that increase the efficiency of the schedule generated, if they are satisfied.

In this study, the course schedule of Osmaniye Korkut Ata University, Faculty of Economics and Administrative Sciences, Department of Business Administration was generated. The original value of the study is to try to find the optimum solution within the framework of the policies followed in Turkey under the conditions of COVID-19 pandemic. With the pandemic, many new constraints such as the transition to hybrid education, increased indoor circulation, and the lack of suitable slots for disinfection of classrooms have been introduced to the literature. Therefore, the objective function of this study was determined as keeping constraint violations to a minimum. In order to minimize constraint violations, violations of hard constraints were multiplied by a large penalty value and summed with soft constraint violations with coefficient values of 1.

Hard constraints of the study were formulated to prevent overlapping of courses, classrooms, lecturers, student groups and to prevent overcapacity in order to be assigned to certain days and time slots. The soft constraints are that the lectures allocated for laboratories and lecture theatres should only be assigned to these classrooms, there should be no consecutive lectures in a classroom, at least one lecture of the faculty members who have lectures in two different slots in a day should be online, if the lecture of any class in any class hour is online, the other lecture of the relevant class assigned to the same day should also be online (for classes with more than 1 online lecture), and not all classrooms in the corridor should be in formal education during the same class hour.

The Department of Business Administration consists of eight classrooms located in two corridors, one laboratory assigned to the department and one lecture hall. No class exceeds its student capacity. In addition, there are 29 courses in the department; 1 course is taught in the laboratory, 1 course is taught in the lecture hall, and the other 27 courses are taught in classrooms by 22 faculty members.

A mathematical model was established to solve the problem. The established mathematical model was solved with the genetic algorithm. The consistency and validity of the model was tested by experimenting various scenarios in the genetic algorithm application.

The chromosome was formed by multiplying the number of classrooms, class sessions and the number of days of classes. In addition, single point crossover, sequential crossover and random selection mutation were used in the study. The best result in the schedules generated with a population of 50 units was the result obtained at the 79th iteration out of 150 iterations in 23 seconds with a crossover rate of 0.8 and a mutation rate of 0.05. Then, the number of populations was increased to 100 and run again. In this case, the result obtained in 47 seconds with 0.8 crossover rate and 0.1 mutation rate was the best

result. In the solution process of 100 iterations, the best solution was reached at the 50th iteration.

Considering the change in the percentage of online courses in case of any pandemic or closure, the rate calculated as 30% was re-processed over 20%. For the Department of Business Administration, which is the subject of the application, this ratio corresponds to 6 lectures being conducted online. When 6 courses are conducted online, the best solution with 100% success rate is obtained in 52.472 seconds on average. In the case of 40% online courses, the success rate decreases to 80% with two near-optimal results, while the average time increases to 54.833 on average. When the scenario where half of the courses are conducted online is tested, the success rate is 80% with two near-optimal results, while the average time increases to 54.833. This shows that while the success rate does not fall below 80%, the programme works in coordination with the number of online lectures in each ratio. According to the three different scenarios run, the GA works not only according to the constraints but also integrated to the changes in the constraints.

For the 50-unit population, when sequential crossover was used, the best result was obtained in 60 seconds with a crossover rate of 0.80, a mutation rate of 0.05 and 150 iterations. With a population of 100 units, the algorithm was run again and the best solution was obtained in 100 seconds with 0.80 crossover rate, 0.10 mutation rate and 150 iterations. Then, by changing the online course loads, 3 different scenarios were applied to the model using sequential crossover. In case of 20% online courses, the best solution is reached in 65 seconds on average with 100% success rate, in case of 40% online courses, the best solution is reached in 82 seconds on average with 100% success rate and in case of 50% online courses, the best solution is reached in 88 seconds on average with 100% success rate. When the crossover types are compared, it can be said that sequential crossover increases the time compared to single-point crossover, but the success rates are higher with the sequential crossover.

The course scheduling problem in this study is designed to be applied in any pandemic and epidemic period, taking into account situations such as preventing transmission, reducing contact, reducing indoor circulation as a result of any epidemic.