

## Sınavlara Gözetmen Atama Probleminin Çözümü İçin Takas Bazlı Bir Algoritma Önerisi

Mehmet ÖZÇALICI\*

Geliş Tarihi (Received): 07.02.2017 – Kabul Tarihi (Accepted): 19.04.2017

### Öz

Modern akademik kurumlar, her dönemde birkaç kere, gözetmenlerin görevli oldukları sınavların hangileri olduğunu içeren çizelgeler hazırlamakla meşgul olmaktadır. Her yükseköğretim kurumunun kendine has yapısının olması, kurumlara özel çözümlerin geliştirilmesini zorunlu kılmaktadır. Sınav çizelgeleri birçok kurumda kağıt üzerinde hazırlanmaktadır. Bu durum, çizelgeyi hazırlamakla görevli kişiler için önemli bir mesai gerektirmektedir. Bu çalışmada, sınavlara gözetmen atama problemini etkin bir şekilde çözecek bir algoritma önerilmektedir. Algoritmanın geliştirilmesinde, yazarın ve çalışma arkadaşlarının kağıt üzerinde gözetmen atadıkları zamanlardaki uygulamaları esas alınmıştır. Algoritma, Kilis 7 Aralık Üniversitesi İktisadi ve İdari Bilimler Fakültesinin 2016-2017 Güz dönemi dönem sonu sınavlarına gözetmen atamak için kullanılmıştır. Sonuçta, 35-40 dakikada tamamlanan atama işlemi, önerilen algoritmanın MATLAB yazılımındaki uygulaması ile birlikte, bir saniyenin altında bir sürede tamamlanabilmiştir. Geliştirilen algoritma, üzerinde bazı değişiklikler gerçekleştirmek suretiyle, diğer kurumların gözetmen atama problemlerinin çözümü için de uygulanabilir. Aynı zamanda esnek bir yapıya sahip olması sayesinde başka ihtiyaçlar doğrultusunda yeniden düzenlenebilir.

**Anahtar Kelimeler:** Gözetmen atama, çizelgeleme, algoritma, optimizasyon, MATLAB.

---

\* Yrd. Doç. Dr., Kilis 7 Aralık Üniversitesi, İktisadi ve İdari Bilimler Fakültesi, e-posta: mozcatici@gmail.com

## **Developing A Swap Based Algorithm For Solving Proctor Assignment Problem**

### **Abstract**

Modern academic institutions are getting busy with preparing examination schedules which includes the information of which proctor is assigned to which exams in each semester. Each institution has its own structure which makes it a necessity to develop institution-specific solutions. In most of the institutions, exam timetables are preparing manually. This requires a considerable time for the responsible team. In this study, a new algorithm is proposed to assign proctors to exams, in an efficient way. The algorithm is based on the ways that are applied by author and his colleagues when they were preparing the examination schedule. The algorithm is applied to assign proctors to final exams in Kilis 7 Aralık University, Winter Semester of 2016-2017. By applying the proposed algorithm in MATLAB environment, it became possible to prepare a schedule which requires 35-40 minutes to prepare manually in a small fraction of a second. The proposed algorithm can be adopted by other institutions by making small changes. Moreover, its flexible content makes it possible to rearrange for different needs.

***Keywords:*** *Proctor assignment, scheduling, algorithm, optimization, MATLAB.*

## Giriş

Modern akademik kurumlar için hangi gözetmenin hangi sınavlarda görev alacağı belirlenmesi, zaman alan ve önemli bir yönetimsel süreçtir. Sınavların sorunsuz bir şekilde tamamlanabilmesi için her dönem gözetmen atama işlemlerinin dikkatle gerçekleştirilmesi gerekmektedir. Yükseköğretim kurumlarında iyi bir gözetmen atama sisteminin yaratılması, problemin karmaşıklığından dolayı, zor bir görevdir. Buna ek olarak, her kurumun kendine ait yapısının olması, kurumlara özel çözümlere ihtiyaç duyulmasına neden olmaktadır.

Sınavlara gözetmen atama probleminin çözümü birkaç açıdan önem arz etmektedir. İlk olarak bir gözetmenin birden fazla sınavda görevlendirilmesi, sınavların aksamasına neden olmaktadır. Diğer taraftan sınav çizelgeleri, sınavlardan en az bir hafta önce ilan edilmelidir. Bu durum, sınavlara gözetmen atama işleminin hızlı bir şekilde gerçekleştirilmesini zorunlu kılmaktadır. Ayrıca bir dönemde, birden fazla sınav için gözetmen atama işlemi gerçekleştirilmektedir (birinci – ikinci ara sınavlar, dönem sonu sınavları, bütünleme sınavları, ek ders sınavları vb). Bu şekilde tekrarlanması ve hızlı çözülmesi gereken bir problem olmasına rağmen, sınav çizelgeleri birçok kurumda kağıt üzerinde ve önemli mesai harcamak suretiyle gerçekleştirilmektedir. Bu şekilde hazırlanan çizelgelerde hata yapma olasılığının yüksek olduğunu söylemek mümkündür. Gözetmen atama probleminin hızlı ve gerektiği gibi çözülmesi, kurumlara önemli zaman tasarrufu sağlayacaktır.

Literatürde sınavlara gözetmen atama probleminin çözümünü gerçekleştiren çalışmalar mevcuttur (Al-Yakoob, Sherali, & Al-Jazzaf, 2010; Koide, 2015). Sınavlara gözetmen atama probleminin çözümü için çalışma sayfaları tabanlı bir sistem de önerilmektedir (Özçalıcı, 2016). Bu yöntem bilgisayar tabanlı olduğu için tarafsız bir çözüm sunmakla beraber çok parametreye sahip gerçek hayat problemlerinin çözümü için ek finansal yatırımlara ihtiyaç duyulmaktadır (Microsoft Excel™ eklentisi veya yeni orta-üst düzey yazılımların satın alınması gibi). Yatırım yapılması maliyetli olduğundan yöneticilerin bu çözümü desteklemesi mümkün görünmemektedir. Yatırım yapılsa bile sınav programını hazırlayan personel tarafından öğrenme süreci için zaman veya eğitim için ek mali yükümlülükler gerekebilir.

Bu çalışmada ise çözümü var olan (sınavların aksamadan yürütülmesini sağlayacak yeterli sayıda gözetmenin bulunduğu) atama problemlerinin çözümü için yeni bir algoritma önerilmektedir. Önerilen algoritmanın geliştirilmesinde, yazarın ve çalışma arkadaşlarının kağıt üzerinde gözetmen atarken kullandıkları yöntemler esas alınmıştır. Başka bir ifade ile, problem çözülürken uygulanan adımlar, bir algoritma haline getirilmiştir. Böylelikle 35-40 dakikada

tamamlanan atama işlemi, MATLAB yazılımında geliştirilen kodlar yardımıyla bir saniyenin altında bir zamanda tamamlanabilmiştir. Bu yeni algoritmanın, kurumlara önemli zaman tasarrufu sağlayacağını söylemek mümkündür.

Çalışma altı bölümden oluşmaktadır. Bu giriş bölümünden sonra, birinci bölümde literatür özeti yer almaktadır. İkinci bölümde çizelgeleme problemi hakkında genel bilgiler verilmektedir. Üçüncü bölümde önerilen algoritma tanıtılmaktadır. Dördüncü bölümde algoritmanın gerçek bir probleminin çözümünde uygulaması yer almaktadır. Son olarak sonuç ve tartışma bölümünde değerlendirme ve sonraki çalışmalar için öneriler yer almaktadır.

## 1. Literatür Taraması

Sınav çizelgesi hazırlama problemi literatürde çok fazla çalışılan problemlerden biridir. Bunun esas nedenleri arasında her kurumun ihtiyaçlarının birbirinden farklı olması ve bu ihtiyaçlara göre çözüm üretilmesi zorunluluğu sıralanabilir. Bu konuda yapılmış çalışmalarını özetleyen literatür taramaları da mevcuttur (Lewis, 2008; Pillay, 2016; Qu, Burke, McCollum, Merlot, & Lee, 2009; Schaerf, 1999). Literatürde genellikle sınav çizelgeleme problemi (hangi sınavın hangi gün olacağı) üzerinde durulmakla birlikte, gözetmen atmama probleminin de incelendiği çalışmalara rastlanmaktadır. Literatürde sınav çizelgeleme problemi ile ilgili yapılan çalışmaları yayınladıkları yılları dikkate almak suretiyle, şu şekilde özetlemek mümkündür.

Arbaoui, Boufflet ve Moukrim 2016 yılında yayınladıkları çalışmalarında sınav çizelgeleme probleminin çözümü için meta-sezgisel optimizasyon algoritması içeren bir model geliştirmişlerdir. Bu model değişkenlerin ve kısıtların sayısını çok yüksek oranda azaltmaktadır. Erişilen sonucun geleneksel matematiksel programlama bazlı sezgisel yaklaşımdan daha iyi olduğu vurgulanmaktadır. Woumans, Boeck, Beliën ve Creemers (2016) sınav çizelgeleme probleminin çözümüne öğrenci odaklı bir yaklaşım sergilemişlerdir. Öğrenciler için sınavların mümkün olduğunca geniş aralıklarla sıralanmasını sağlayan bir model önermektedirler.

Koide (2015) ise çalıştığı üniversitede gözetmen atama probleminin çözecek karışık tamsayı programlama modeli geliştirmiştir. Yazar çalışmasında, sistemin kolay kullanılmasını sağlayacak yöntemleri de tartışmıştır.

Al-Betar, Khader ve Doush (2014) çalışmalarında sınav çizelgeleme problemini Armoni Arama (Harmony Search) algoritması ile çözecek bir model üzerinde çalışmışlardır. Modelin

performansını 22 farklı yöntemle ve 12 adet veri seti üzerinde incelemişlerdir. Bu veri setlerinin 2'si hariç diğer veri setlerinin hepsinde, önerdikleri yöntemin, diğer yöntemlerden daha iyi sonuç verdiğini raporlamaktadırlar.

Acar ve Şevkli (2013) çalışmalarında sınav çizelgelemesi için matematiksel bir model önermektedirler. Öğrencilerin sınav çakışması probleminin önemine dikkat çekilmekte ve matematiksel modellemeye dayalı sezgisel bir yöntem geliştirmişlerdir ve yöntemi Fatih Üniversitesi'nde uygulamışlardır.

Kalayci ve Güngör (2012) çalışmalarında Pamukkale Üniversitesi için genetik algoritma tabanlı sınav çizelgeleme modeli geliştirmişlerdir. Çalışmalarında öğrencilerin başarılarına odaklanmışlardır. Zorluk derecelerine göre sınavlar arasında mümkün olduğunca uzun zaman dilimlerinin yer almasına dikkat edilmiştir. Abdullah ve Turabieh (2012) çalışmalarında genetik algoritma ve Tabu arama algoritmalarının birleşiminden oluşan melez bir yöntemle üniversiteler için sınav takvimi hazırlama problemi üzerinde durmuşlardır. Rashidi Komijan ve Nouri Koupaei (2012) çalışmalarında sınav programı hazırlama problemini çözecek bir matematiksel model önermişlerdir. Bu modeli, İran'da bulunan İslami Azad Üniversitesi Endüstri Mühendisliği bölümü için GAMS yazılımını kullanarak uygulamışlardır.

2011 yılında ise, İlkuçar, sınav programı hazırlama probleminin optimizasyonu üzerinde durmuştur. Yazar çalışmasında, C# platformunda hazırladığı modelin, Mehmet Akif Ersoy Üniversitesi Burdur Meslek Yüksekokulu'nun üç dönem boyunca sınav programlarının hazırlanmasında kullanıldığını ve programlarla ilgili hiçbir şikâyetin alınmadığını ifade etmektedir.

Kahar ve Kendall (2010) çalışmalarında Malezya'da bir üniversitede zaman çizelgeleme probleminin çözümü üzerinde durmuşlardır. Çalışmalarında bir sınavın farklı salonlarda bir arada gerçekleşmesi durumunu da göz önünde bulundurmuş ve yeni bir model geliştirmişlerdir. Çalışma sonunda üniversitede halihazırda kullanılan yöntemden daha iyi bir çözümün üretilebileceği ifade edilmektedir. Al-Yakoob vd. (2010) çalışmalarında Kuveyt Üniversitesi'nde sınav çizelgeleme problemini; sınavların hangi gün ve hangi salonda olacağı problemi ve sınavlara gözetmen atama problemi olmak üzere iki bölüme ayırmışlardır. Sınavlara gözetmen atama problemini doğrusal programlama bazlı sezgisel bir algoritma yardımıyla çözmüşlerdir.

Çoruhlu (2007) çalışmasında yaklaşık 4000 öğrencinin öğretime devam ettiği bir yüksek öğretim kurumunda sınav çizelgeleme problemini çözmek için tamsayı programlama tabanlı

bir yöntem önermiştir. Yazar, yaklaşık 150 gözetmenin görev aldığı problemin, önerilen yöntem yardımıyla, kabul edilebilir zaman diliminde çözüldüğünü raporlamaktadır.

MirHassani (2006) çalışmasında üniversiteler için zaman çizelgeleme problemlerinin tamsayı programlama yaklaşımıyla çözülebileceğini ifade etmektedirler. Çalışmada önerilen model bir üniversitede ders programı hazırlamak için uygulanmıştır. 1226 adet değişkene sahip bir problemin çok kısa sürede çözülebileceği raporlanmaktadır.

Dowland ve Thompson (2005) çalışmalarında sınav programını hazırlamak için metasezgisel bir optimizasyon yöntemi olan karınca koloni optimizasyon (ant colony optimization) yöntemini kullanmışlardır. Çalışmalarında geleneksel modele bazı ek amaç ve kısıtlar eklemek suretiyle, çakışma olmayan bir çözüme ulaşmışlardır.

Burke ve Newall (2004) ise çalışmalarında sınav çizelgeleme probleminin sezgisel sıralama adaptasyonu ile çözümlenmesini önermektedirler. Bu yöntemin diğer yöntemlerle benzer veya onlardan daha iyi sonuç üretebildiğini ifade etmektedirler.

## **2. Çizelgeleme Problemi**

Çizelgeleme faaliyeti birçok üretim ve hizmet işletmelerinde süreklilik arz eden bir karar verme sürecidir. Çizelgeleme, kıt olan kaynakların belirli periyotlarda görevlere atanması ile ilgilenmektedir ve bir ya da daha fazla amacın optimize edilmesini hedeflemektedir (Pinedo, 2016, s. 1). Bir organizasyondaki kaynak ve görevler farklı biçimlerde olabilir. Kaynaklar bir atölyede makineler, havaalanında pistler, inşaat alanında işçiler, bilgisayarlarda işlemciler, üniversitelerde öğretim üyeleri veya araştırma görevlileri olabilir. Görevler ise üretim süreci, uçakların hava alanlarına inmesi ve havalanması, inşaat projelerinde çeşitli aşamalar, bilgisayar programlarının yürütülmesi, dersler veya gözetmenlikler olabilir (Brucker & Knust, 2010, s. 1). Çizelgeleme işlemini gerçekleştirmek için her görevin başlama ve bitiş saatlerinin belirli olması gerekir.

Çizelgeleme konusunda yapılan çalışmaların tarihi 1950'lere kadar gitmektedir (Leung, 2004). Tarih boyunca; yönetim, endüstri mühendisliği ve yöneylem araştırması alanlarında çalışan kişiler, kaynakları yönetme problemi ile karşılaşmışlardır. İyi yapılan çizelgeleme faaliyetlerinin kurumlara maliyet ve zaman avantajı sağlayacağı bir gerçektir.

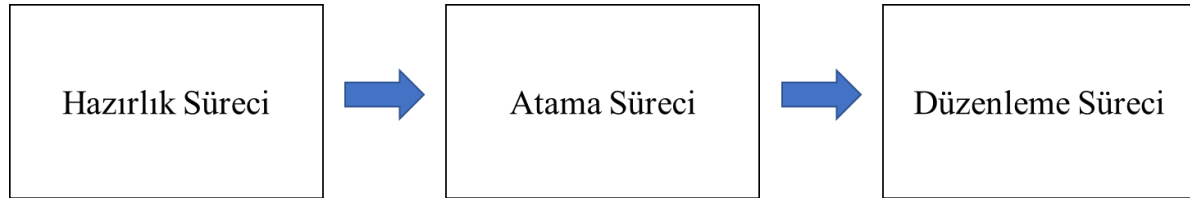
Özel olarak, eğitim alanında da çizelgeleme problemi ile karşılaşılmaktadır. Literatürde yayınlanan çalışmalar göz önünde bulundurulduğunda, eğitim alanında üç ana problem üzerinde çalışıldığı anlaşılmaktadır (Kingston, 2013, s. 91). Bu problemler; liselerde

çizelgeleme, üniversitelerde sınav çizelgeleme ve yine üniversitelerde ders programı hazırlama problemlerdir. Bunun yanı sıra çeşitli alt problemler de mevcuttur.

Bu alt problemlerden biri de sınavlara gözetmen atama problemidir. Sınavlara gözetmen atama problemi, hangi gözetmenin hangi sınavda görev alması gerektiğine karar verilmesi olarak tanımlanabilir. Her üniversitenin, her fakültenin ihtiyaçları birbirinden farklıdır. Ancak genel anlamda sınavlara gözetmen atama sürecinin Şekil 1'deki gibi üç bölümden meydana geldiğini ifade etmek mümkündür. Hazırlık aşamasında atama işleminin gerçekleştirilmesi için ihtiyaç duyulan bilgiler bir araya getirilmektedir. Bu bilgiler;

- Gözetmen sayısı
- Gözetmenlerin müsait olduğu (veya izinli olduğu) günler (saatler)
- Sınavların hangi gün yapılacağı
- Her bir saatte kaç adet gözetmene ihtiyaç duyulduğu

İhtiyaç duyulan bu bilgiler bir araya getirildikten sonra gözetmenlerin hangi sınavlarda görev alacaklarına karar verilmesi gerekmektedir ve bu karar süreci atama süreci olarak adlandırılabilir. Atama işlemi bittikten sonra, atama sonuçlarının taraflara bildirilecek şekilde düzenlenmesi gerekmektedir.



**Şekil 1.** Sınavlara Gözetmen Atama Süreci

İyi hazırlanmış bir sınav programında olması gerekenler şu şekilde özetlenebilir;

- Gözetmenlere her saatte sadece bir sınıfta görev verilmeli, aynı anda iki görev yazılmamalıdır.
- Görev dağılımları eşit olmalıdır. Toplam görev sayısının gözetmen sayısına tam olarak bölünemediği durumlarda, her gözetmene, hesaplanan alt ve üst sınırlarda görev ataması gerçekleştirilmelidir.
- Her sınava ihtiyaç duyulduğu kadar gözetmen atanmalı fazla veya eksik atama gerçekleştirilmemelidir.

Özellikle yeni kurulan üniversitelerde gözetmen sayısında artma veya azalmalar ortaya çıkmaktadır. Ayrıca gözetmenlerin müsait oldukları günler her dönem değişmektedir. Her sınav döneminin yapısı bu şekilde farklılık arz etmektedir. Bu nedenle de her dönem mutlaka yeniden gözetmen atama işleminin gerçekleştirilmesi gerekmektedir.

### 3. Önerilen Yöntem

Çalışmada, yazar ve çalışma arkadaşlarının sınavlara gözetmen atama problemini kağıt üzerinde çözdükleri zamanlarda uyguladıkları yöntem baz alınmıştır. Yazar ve çalışma arkadaşları, problemi çözerken görev dağılımlarının eşit olmasını sağlamak için, gözetmenler arasında görev değiş tokuşu gerçekleştirmektedirler ve bu işlem her gözetmene adil sayıda görev düşene kadar devam etmektedir. Ancak bir değiş tokuş işlemini bilgisayar desteği olmadan tamamlamak uzun zaman almaktadır ve zahmetlidir.

Önerilen algoritmanın işleyebilmesi için bazı varsayımlara ihtiyaç duyulmaktadır. İlk olarak uygun bir çözümün olduğu varsayılmaktadır. Örneğin bir saatte 10 adet gözetmene ihtiyaç duyuluyorsa ve o saatte sadece 8 adet gözetmen müsait ise bu durumda problem çözülemeyecektir. Bir diğer varsayım ise her sınavın en fazla bir saat sürdüğüdür. Buna rağmen birden fazla teknikle bu varsayım ortadan kaldırılabilir. Algoritmanın çalışması için bir hazırlık sürecine de ihtiyaç duyulmaktadır. Bu süreçte bütün gözetmenlerin müsait olduğu zaman dilimlerinin belirlenmesi ve her saatte kaç adet gözetmene ihtiyaç duyulduğunun belirlenmesi gerekmektedir.

Tablo 1’de üzerinde çalışılan problem özet halinde sunulmuştur. Sütunlarda gözetmenler yer almaktadır. Satırlarda ise sınav sayıları yer almaktadır. Her sınavda kaç adet görevliye ihtiyaç duyulduğunu içeren  $p$  vektörü ise son sütunda görülmektedir. Gözetmenlerin uygun olmadığı zaman dilimleri 9 rakamı ile işaretlenmiştir ve o saatlerde o gözetmenlere görev yazılmaması gerekmektedir. Tabloda sadece boş bırakılan hücrelere atama işlemi gerçekleştirilebilir. Problem hakkında detaylı bilgi ve problemin kısaltılmış bir versiyonunun doğrusal programlama modeli (Özçalıcı, 2016) kaynağında yer almaktadır. Tekrardan kaçınmak adına burada ifade edilmeyecektir. Problemin tanımı için gerekli olan parametreler Şekil 2’de yer almaktadır.



**Tablo 1.** Problemden özet bir görüntü

		$G_1$	$G_2$	...	$G_{11}$	$G_{12}$	$p$
02/01/2017	10:00		9	...			2
02/01/2017	11:00		9	...			4
02/01/2017	13:00		9	...			7
02/01/2017	14:00		9	...			8
...	...	...	...	...	...	...	...
12/01/2017	10:00	9		...			8
12/01/2017	11:00	9		...			9
12/01/2017	13:00	9		...			6
12/01/2017	14:00	9		...			4

Parametreler
$n$ : Toplam gözetmen sayısı
$p_i$ : Her saatte gereken gözetmen sayısı (vektör)
$i$ : Satır indeksi (Sınav saatleri)
$j$ : Sütun indeksi (Gözetmenler)
$ub$ : Üst sınır $ub = yukarıyuvarla \left( \frac{\sum(p_i)}{n} \right)$
$lb$ : Alt sınır $lb = aşağıyuvarla \left( \frac{\sum(p_i)}{n} \right)$
$m_j$ : Gözetmenlere düşen toplam görev sayısı (vektör)
$d$ : Karar matrisi

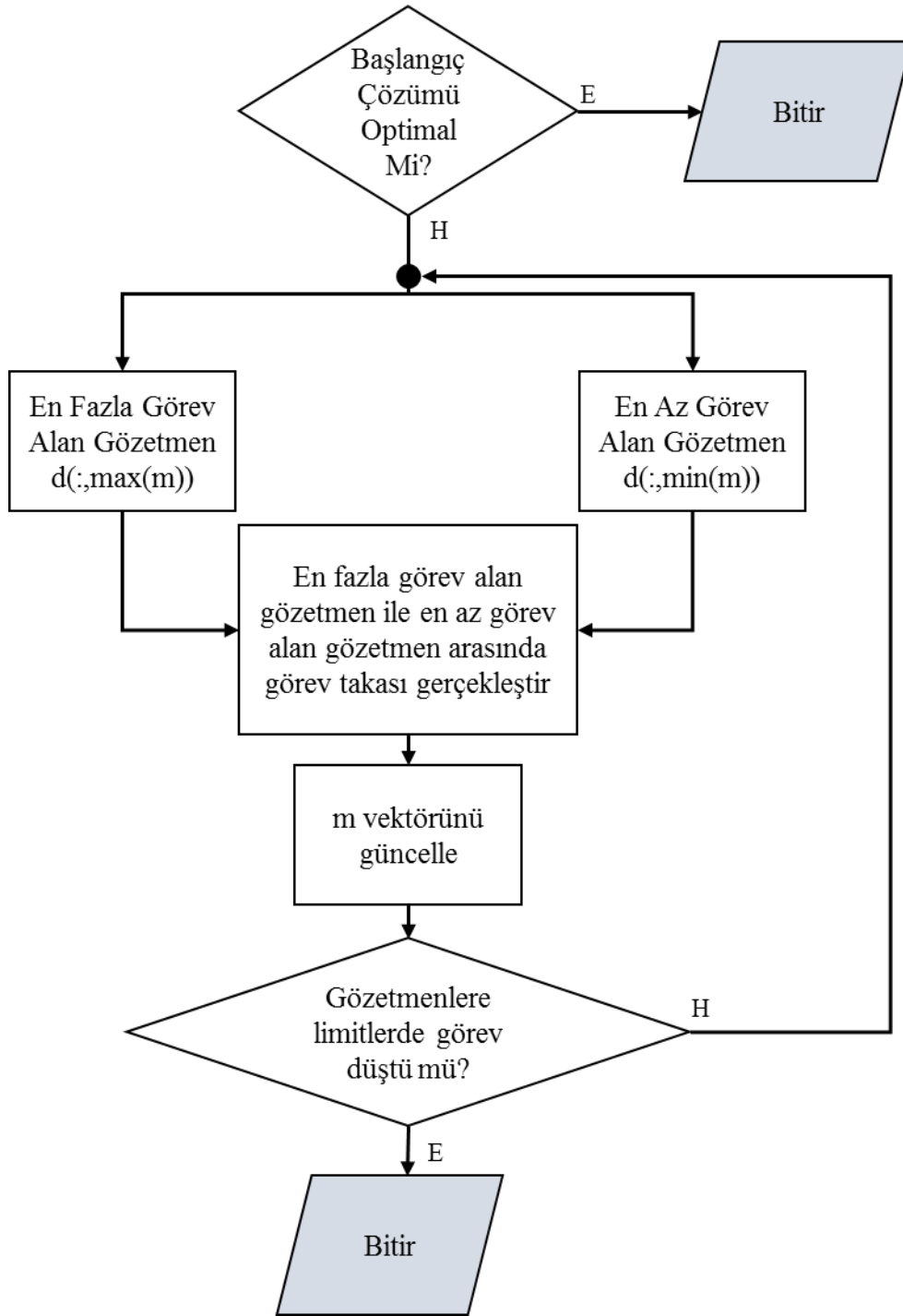
**Şekil 2.** Parametreler

Algoritma iki adımdan oluşmaktadır. İlk aşamada rastgele bir başlangıç çözümü oluşturulmaktadır (Şekil 3). Bu adımda  $p$  vektörü boyunca her sınava ihtiyaç duyulduğu kadar gözetmen, o saatte müsait olan gözetmenler arasından rastgele seçilerek atanmaktadır. Bu başlangıç çözümünde aynı gözetmen birden fazla sınava atanmamakta ve her saatte ihtiyaç duyulduğu kadar gözetmen atanmaktadır. Ancak seçimler rastgele gerçekleştirildiğinden, her gözetmene düşen görev sayısında ( $m$  vektörü) önemli dengesizlikler bulunmaktadır.

%% Birinci Bölüm: Başlangıç Çözümünün Oluşturulması
<pre> <b>for</b> idx = 1:length(p)     uygun olan gözetmenleri belirle     rastgele seçim yap ve görev ata <b>end</b> </pre>
%% İkinci Bölüm: Görev Takaslarının Gerçekleştirilmesi
<pre> <i>m</i> vektörünü hesapla <i>isoptimal</i> = herhangi bir <math>m_j &gt; ub</math>? <b>while</b> <i>isoptimal</i> = 0     <i>enfazla</i> = <math>d(:, \max(m))</math>;     <i>enaz</i> = <math>d(:, \min(m))</math>;     <b>for</b> kdx = 1:length(p)         <b>if</b> <i>enfazla</i>(kdx) = 1 ve <i>enaz</i>(kdx) = 0             <i>enfazla</i>(kdx) = 0;             <i>enaz</i>(kdx) = 1;             döngüden çık         <b>endif</b>     <b>endfor</b>     yeni <i>enfazla</i> ve <i>enaz</i> vektörlerini <i>d</i> matrisinde yerine yerleştir     <i>m</i> vektörünü güncelleştir     <i>isoptimal</i> değerini güncelleştir <b>endwhile</b> </pre>

Şekil 3. Algoritmanın Sahte Kodu (pseudocode)

İkinci aşamada ise oluşturulan başlangıç çözümü iterasyonlar (yinelemeler) yardımıyla optimize edilmektedir (Şekil 3). Her iterasyonda en fazla ve en az sayıda görev alan gözetmenler belirlenmektedir. Her iki gözetmenin aldığı görevler  $p$  vektörü boyunca taranmaktadır. En fazla sayıda görev alan gözetmenin görev aldığı ve en az sayıda görev alan gözetmenin görev almadığı ilk saat (indeks) belirlenmektedir. Bu saatte görev takası gerçekleştirilmektedir. Başka bir ifade ile en fazla sayıda görev alan gözetmenden bir görev en az sayıda görev alan gözetmene devredilmektedir. Daha sonra çözümün optimal olup olmadığı kontrol edilmektedir. Herhangi bir gözetmene üst sınırdan ( $ub$ ) daha fazla sayıda görev düştüğü durumda, çözüm optimal değildir ve iterasyonlara devam edilir. İterasyonlar Şekil 4’de akış grafiği şeklinde ifade edilmiştir.



**Şekil 4.** Algoritmanın Görev Değişikliği İşlemlerini Gerçekleştiren Bölümünün Akış Diyagramı

#### 4. Örnek Uygulama

Algoritma, Kilis 7 Aralık İktisadi ve İdari Bilimler Fakültesi 2016 – 2017 Güz Dönemi dönem sonu sınav çizelgelemesi için uygulanmıştır. Söz konusu dönemde sekiz aktif bölüm,

ikinci öĖretimler dahil 20 adet sınıfta toplam 1184 öĖrenci bulunmaktadır. Ayrıca gözetmenlik yapabilecek 12 adet gözetmen bulunmaktadır. DiĖer taraftan farklı büyüklüklerde 8 adet salon bulunmaktadır ve bazılarında bir bazılarında ise iki adet gözetmenin görevlendirilmesi gerekmektedir. İkinci öĖretimlerle birlikte toplam 78 adet ders ve atanması gereken 298 adet gözetmenlik görevi bulunmaktadır.

Algoritma MATLAB yazılımında çalışacak şekilde tasarlanmıştır ve süreç standart bir dizüstü bilgisayarda bir saniyenin altında bir sürede tamamlanmıştır. İki adet gözetmen 24 adet sınavda görevlendirilmiş ve geriye kalan on adet gözetmen 25'er adet sınavda görevlendirilmiştir. Tablo 2'de algoritmanın ürettiĖi sonuçlar yer almaktadır.

**Tablo 2.** Algoritmanın oluşturduĖu nihai çözüm

Tarih	Gün	Saat	G <sub>1</sub>	G <sub>2</sub>	G <sub>3</sub>	G <sub>4</sub>	G <sub>5</sub>	G <sub>6</sub>	G <sub>7</sub>	G <sub>8</sub>	G <sub>9</sub>	G <sub>10</sub>	G <sub>11</sub>	G <sub>12</sub>	p
2.01.2017	Pazartesi	10:00	1			1	0	0	0	0			0	0	2
2.01.2017	Pazartesi	11:00	1			1	1	0	1	0			0	0	4
2.01.2017	Pazartesi	13:00	1			1	1	1	1	1			1	0	7
2.01.2017	Pazartesi	14:00	1			1	1	1	1	1			1	1	8
2.01.2017	Pazartesi	15:00	1			0	0	0	0	0			0	1	2
2.01.2017	Pazartesi	16:00	1			0	0	0	1	0			1	1	4
2.01.2017	Pazartesi	17:00	1			1	1	1	1	0			1	1	7
3.01.2017	Salı	10:00	1	1		1	1	1	1	1	1		1	1	10
3.01.2017	Salı	13:00	1	1		0	0	0	0	0	0		0	0	2
3.01.2017	Salı	14:00	1	1		1	0	0	0	1	1		0	1	6
3.01.2017	Salı	15:00	1	1		1	1	0	0	1	1		1	1	8
3.01.2017	Salı	16:00	1	0		1	0	0	1	1	0		0	0	4
3.01.2017	Salı	17:00	1	1		1	0	1	0	0	1		1	0	6
4.01.2017	Çarşamba	10:00	1	1		1	1	1	1	1	1		1	1	10
4.01.2017	Çarşamba	11:00	1	1		1	1	0	1	0	0		1	1	7
4.01.2017	Çarşamba	13:00	1	1		0	0	1	1	1	0		0	1	6
4.01.2017	Çarşamba	14:00	1	1		1	0	1	0	1	1		0	0	6
4.01.2017	Çarşamba	15:00	1	1		1	1	1	1	1	1		1	1	10
4.01.2017	Çarşamba	16:00	0	0		1	0	0	1	0	0		0	0	2
5.01.2017	Perşembe	10:00	0	0	1	1	0	0	0	1	0	1	1	0	5
5.01.2017	Perşembe	11:00	0	1	1	0	0	0	0	1	1	1	1	0	6
5.01.2017	Perşembe	13:00	1	0	1	1	0	0	1	1	0	1	0	0	6
5.01.2017	Perşembe	14:00	1	1	1	1	1	0	1	1	1	1	1	1	11
5.01.2017	Perşembe	15:00	1	1	1	1	1	0	0	1	0	1	1	1	9
5.01.2017	Perşembe	16:00	1	0	1	0	1	0	0	0	1	0	0	0	4
6.01.2017	Cuma	10:00	0	0	1		0	0	1	0	1	1	0	0	4
6.01.2017	Cuma	11:00	1	0	1		0	1	1	0	0	1	0	0	5
6.01.2017	Cuma	14:00	1	1	1		1	0	1	1	1	1	0	1	9
6.01.2017	Cuma	15:00	0	0	0		0	0	0	1	1	1	0	1	4
6.01.2017	Cuma	16:00	0	0	1		1	0	0	1	0	1	1	0	5
6.01.2017	Cuma	17:00	1	0	1		0	1	0	0	0	1	0	0	4
9.01.2017	Pazartesi	10:00		1	1	1	1	1	1	1	1	1	1	1	11

9.01.2017	Pazartesi	11:00	1	1	1	0	1	1	0	1	1	1	1	9
9.01.2017	Pazartesi	13:00	0	1	1	0	1	0	1	0	0	0	0	4
9.01.2017	Pazartesi	16:00	0	1	0	0	1	0	0	0	1	1	0	4
10.01.2017	Salı	13:00	0	1	1	1	0	0	0	0	1	0	0	4
10.01.2017	Salı	14:00	1	1	1	1	1	1	0	1	1	1	1	9
10.01.2017	Salı	15:00	0	0	0	0	0	0	0	1	1	1	1	3
10.01.2017	Salı	16:00	1	1	1	0	1	1	1	1	1	1	0	8
11.01.2017	Çarşamba	10:00	0	1	0	0	0	0	0	0	0	0	1	2
11.01.2017	Çarşamba	11:00	1	0	0	0	1	0	1	1	0	1	1	6
11.01.2017	Çarşamba	13:00	1	1	0	1	1	1	0	1	0	0	0	6
11.01.2017	Çarşamba	14:00	1	1	0	1	1	0	0	0	1	1	0	6
11.01.2017	Çarşamba	15:00	1	1	0	1	1	0	1	1	1	0	0	7
11.01.2017	Çarşamba	16:00	0	1	0	1	1	1	1	1	1	1	1	9
12.01.2017	Perşembe	10:00	1	1	1	1	1	1	0	0	1	1	1	8
12.01.2017	Perşembe	11:00	1	1	1	1	1	0	1	1	1	1	1	9
12.01.2017	Perşembe	13:00	1	0	1	0	0	0	1	1	1	1	1	6
12.01.2017	Perşembe	14:00	0	0	1	1	1	0	1	0	0	0	0	4
			25	25	25	25	25	25	25	25	24	24	25	298

## Sonuç ve Tartışma

Üniversitelerde hangi gözetmenin hangi sınavda görevlendirileceği, genellikle sorumlu araştırma görevlileri tarafından kağıt üzerinde hazırlanmaktadır. Ancak bu durum hem çok uzun zaman almaktadır hem de bu durumda hata yapma olasılığı yüksektir. Sınavlara gözetmen atama işleminin süreklilik arz eden bir karar verme süreci olması, otomasyona gidildiğinde sağlanacak küçük zaman tasarruflarını bile önemli hale getirmektedir.

Çalışmada sınavlara gözetmen atama probleminin çözümü için bir algoritma önerilmiştir. Algoritma, kâğıt üzerinde gözetmen atama sürecinden ilham alınmak suretiyle geliştirilmiştir. Sonuçta, çok hızlı bir şekilde, her sınava ihtiyaç duyulduğu kadar gözetmen atanmıştır. Ayrıca her gözetmene limitlerde (adil sayıda) görev yazılmıştır.

Önerilen yöntemin bazı avantajları ve dezavantajları mevcuttur. Yöntem, büyük çaptaki problemlerin çözümünü gerçekleştirecek düzeydedir. Çözüm, bilgisayar tarafından gerçekleştirildiği için tarafsızdır. Gözetmenlerin aynı anda iki sınavda görevlendirilmesi (görev çakışması) gibi bir durum söz konusu değildir. Bazı değişiklikler yardımıyla farklı fakültelerde ve üniversitelerde de uygulanabilir. Bu avantajlarının yanı sıra, algoritmanın çalışabilmesi için her saat kaç adet gözetmene ihtiyaç duyulduğunun belirlenmesi gerekmektedir. Buna ek olarak, algoritma çıktılarının taraflara belge haline getirilmesi ek bir çaba ile olmaktadır. Başka bir

ifade ile algoritmanın çalışması için hazırlık ve atama işleminden sonra düzenleme sürecine ihtiyaç duyulmaktadır. Sınav sürelerinin bir saatten fazla sürdüğü durumlarda da ek düzenlemelere ihtiyaç duyulmaktadır.

İlerleyen çalışmalarda hazırlık ve atama sonrası süreçler için de algoritmaların geliştirilmesi düşünülebilir. Sınavların hangi gün ve hangi salonda olacağını da hazırlayacak algoritmaların geliştirilmesi ile birlikte sınav çizelgeleme işleminde önemli ölçüde zaman tasarrufu sağlanabilecektir.

## KAYNAKÇA

- Abdullah, S., & Turabieh, H. (2012). On the use of multi neighbourhood structures within a Tabu-based memetic approach to university timetabling problems. *Information Sciences*, 191, 146–168. <https://doi.org/10.1016/j.ins.2011.12.018>
- Acar, M. F., & Şevkli, M. (2013). Sınav Çizelgelemesi İçin Matematiksel Model Yaklaşımı. *Verimlilik Dergisi*, 2013(1), 75–86.
- Al-Betar, M. A., Khader, A. T., & Doush, I. A. (2014). Memetic techniques for examination timetabling. *Annals of Operations Research*, 218(1), 23–50. <https://doi.org/10.1007/s10479-013-1500-7>
- Al-Yakoob, S. M., Sherali, H. D., & Al-Jazzaf, M. (2010). A mixed-integer mathematical modeling approach to exam timetabling. *Computational Management Science*, 7(1), 19–46. <https://doi.org/10.1007/s10287-007-0066-8>
- Arbaoui, T., Boufflet, J. P., & Moukrim, A. (2016). A matheuristic for exam timetabling. *IFAC-PapersOnLine*, 49(12), 1289–1294. <https://doi.org/10.1016/j.ifacol.2016.07.701>
- Brucker, P., & Knust, S. (2010). On the complexity of scheduling. In Y. Robert & F. Vivien (Eds.), *Introduction to Scheduling* (pp. 1–20). CRC Press Taylor&Francis Group.
- Burke, E. K., & Newall, J. P. (2004). Solving Examination Timetabling Problems through Adaption of Heuristic Orderings. *Annals of Operations Research*, 129, 107–134. <https://doi.org/10.1023/B:ANOR.0000030684.30824.08>
- Çoruhlu, A. (2007). *Sınav Personel Çizelgeleme Modeli*, Yayınlanmamış Yüksek Lisans Tezi, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Ekim 2007.
- Dowland, K. A. & Thompson, J. M. (2005). Ant colony optimization for the examination scheduling problem. *Journal of the Operational Research Society*, 56, 426–438
- İlkuçar, M. (2011). Sınav Gözetmenlik Çizelgeleme Probleminin Optimizasyonu ve Bir Uygulama Yazılımı. In *Akademik Bilişim Konferansı 11* (pp. 413–420).
- Kahar, M. N. M., & Kendall, G. (2010). The examination timetabling problem at Universiti Malaysia Pahang: Comparison of a constructive heuristic with an existing software solution. *European Journal of Operational Research*, 207(2), 557–565. <https://doi.org/10.1016/j.ejor.2010.04.011>

- Kalayci, C. B., & G ng r, A. (2012). A genetic algorithm based examination timetabling model focusing on student success for the case of the college of engineering at Pamukkale University, Turkey. *Gazi University Journal of Science*, 25(1), 137–153.
- Kingston, J. H. (2013). Educational Timetabling. In A.  . Etaner-Uyer, E.  zcan, & N. Urquhart (Eds.), *Automated Scheduling and Planning* (Vol. 505, pp. 91–108). Springer-Verlag. <https://doi.org/10.1007/978-3-642-39304-4>
- Koide, T. (2015). Mixed integer programming approach on examination proctor assignment problem. *Procedia Computer Science*, 60(1), 818–823. <https://doi.org/10.1016/j.procs.2015.08.244>
- Leung, J. Y.-T. (2004). Introduction and Notation. In J. Y.-T. Leung (Ed.), *Handbook of Scheduling Algorithms, Models and Performance Analysis*. Chapman & Hall/CRC.
- Lewis, R. (2008). A survey of metaheuristic-based techniques for University Timetabling problems. *OR Spectrum*, 30(1), 167–190. <https://doi.org/10.1007/s00291-007-0097-0>
- MirHassani, S. A. (2006). A computational approach to enhancing course timetabling with integer programming. *Applied Mathematics and Computation*, 175, 814–822
-  zcalıcı, M. (2016). Sınavlara g zetmen atama problemlerinin alıřma sayfaları ile optimizasyonu. *Niğde  niversitesi İİBF Dergisi*, 9(3), 103–114.
- Pillay, N. (2016). *A review of hyper-heuristics for educational timetabling*. *Annals of Operations Research* (Vol. 239). Springer US. <https://doi.org/10.1007/s10479-014-1688-1>
- Pinedo, M. L. (2016). *Scheduling*. *Scheduling*. Springer Science+Business Media. <https://doi.org/10.1007/978-3-319-26580-3>
- Qu, R., Burke, E. K., McCollum, B., Merlot, L. T. G., & Lee, S. Y. (2009). A survey of search methodologies and automated system development for examination timetabling. *Journal of Scheduling*, 12(1), 55–89. <https://doi.org/10.1007/s10951-008-0077-5>
- Rashidi Komijan, A., & Nouri Koupaei, M. (2012). A new binary model for university examination timetabling: a case study. *Journal of Industrial Engineering International*, 8(1), 28. <https://doi.org/10.1186/2251-712X-8-28>
- Schaerf, A. (1999). Survey of automated timetabling. *Artificial Intelligence Review*, 13(2), 87–127. <https://doi.org/10.1023/A:1006576209967>
- Woumans, G., Boeck, L. De, Beli n, J., & Creemers, S. (2016). A column generation approach for solving the examination-timetabling problem. *European Journal of Operational Research*, 253(1), 17. <https://doi.org/10.1016/j.ejor.2016.01.046>