

## Genetik Algoritma ile Çözümü Gerçekleştirilen

### Atölye Çizelgeleme Probleminde Ürün Sayısının Etkisi

Serdar Biroğul, Uğur Güvenç

Muğla Üniversitesi, Teknik Eğitim Fakültesi, Elektronik Bilgisayar Eğitimi Bölümü, Kötekli - Muğla  
sbirogul@gazi.edu.tr , uugurguvenç@gazi.edu.tr

**Özet:** Atölye çizelgeleme problemleri, NP (Non - Polinomal) olarak adlandırılan problem sınıfına girmektedir. Bu tür problemlerin belirli kısıtlar altında çözümünün gerçekleştirilmesi klasik matematiksel yöntemlerle mümkün olmamaktadır. Ürün sayısı gibi belirli kısıtlar altında atölye çizelgeleme probleminin en iyi çözümü genetik algoritma (GA) gibi sezgisel arama metodlarıyla sağlanmaktadır. Elde edilen çözüm uzayı kromozom kodlarının çözümüyle görsel bir yapı olarak gantt şemasında gösterilmiştir. Bu çalışmada ürün adedinin dikkate alınmasının hem oluşacak olan gantt şemasına hem de genetik algoritmanın çalışmasına nasıl bir etki yaptığı incelenmiştir.

**Anahtar Kelimeler:** Atölye çizelgeleme, genetik algoritma, gantt şeması, ürün adedi

#### The Effect of Product Number on Solving The Job-Shop Scheduling Problem By Using Genetic Algorithm

**Abstract:** Job-Shop Scheduling Problems (JSSP) are included in the problem group called NP (non-polynomial) and only approximate solutions could be found. The use of heuristic methods as Genetic Algorithms (GA) instead of mathematical rules gives more successful results in JSSP with constraints and products number criteria. Result is shown interface that diagram of gantt scheme. In this paper, it is examined how considering the number of products is affects both the gantt scheme and the genetic algorithm.

**Keywords:** job-shop scheduling, genetic algorithms, gantt scheme, the number of product.

#### 1. Giriş

GA'nın, çizelgeleme yöntemlerinin bir dalı olan atölye çizelgeleme problemlerinde kullanımı ilk defa Davis tarafından gerçekleştirilmiştir [1]. Liepis ise ilk defa genetik algoritma yapısını iki makineli çizelgelemeye uygulamıştır [2]. Biegal ve Daven atölye çizelgelemede genetik algoritmayı bütünleşmiş imalat çevrimi içinde kullanmışlar ve bu yapıyı tek, iki ve çok makineli sistemlere uygulamışlardır [3]. Chen ve arkadaşları genetik algoritmaların literatürdeki diğer sezgisel yaklaşımlardan daha iyi sonuç verdiğini belirtmişlerdir [4]. Gerçek uygulamada atölye çizelgeleme problemlerinde karşılaşılan önemli unsurlardan biri de ürün adedir. Literatüre ba-

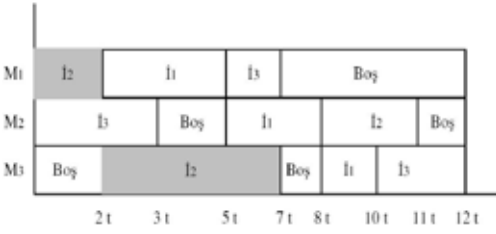
kıldığında bu tür problemlerin çözümünde ürünler tek parçaymış gibi kabul edilmektedir

[5-9]. Böylesi bir yaklaşım problemin gerçek dünyayı ifade etmesini de zorlaştırmaktadır. Bu yüzden çizelgelemenin yapılmasında aynı parçaların birden fazla olduğu gerçeği ihmal edilmemelidir. Bu gerçeklik doğrultusunda ise hem oluşturulacak olan gantt şeması değişmekte hem de genetik algoritmanın performansı etkilenmektedir.

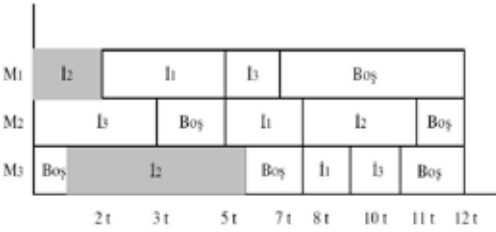
#### 2. Problemin Tanımı

GA ile çözümü gerçekleştirilen atölye çizelgeleme problemlerinde en iyi çözümü daha

kolay elde etmek için işler (ürünler) bir adet olarak kabul edilmektedir. Şekil 1.'de farklı siparişe ait aynı parçadan işlerin ürün adetleri önemsenmemiş durumdaki gantt şeması gösterilmiştir. Buna göre İ2 işi M1 makinesindeki işlemi bittikten sonra M3 makinesinde devam etmektedir. Bu şekil üzerinde şu varsayımlarda da bulunabilir; İ2 işinin sipariş sayısı bir veya birden fazla olabilir veya farklı siparişlere ait de olabilir. Ancak bu durum çizelgelemede bilinmez. Böyle bir yaklaşım hem tamamlanma süresinin en azlanmasını engellemekte hem de makinelerin aylak (boş) zamanlarının fazla olmasına neden olmaktadır [6]. Bu sorun işletme için hem teslim tarihlerini geciktirmekte hem de makinelerin verimli çalışmamasından dolayı maddi zarara sebep olmaktadır.



Şekil 1 . Ürün sayısının dikkate alınmadığı durumdaki gantt şeması



Şekil 2. Ürün sayısının dikkate alındığı durumdaki gantt şeması

Yukarıda anlatılan bu durumu düzeltmek için şekil 2.'de çizilen gantt şemasında da görüldüğü gibi işler parça sayıları dikkate alınarak gerçekleştirilmiştir. M3 makinesindeki İ2 işi M1 makinesindeki sürenin bir kısmında işleme dahil olmaktadır. Bu durum işlerin çakıştığı anlamına gelmemektedir. Sadece M1 makinesindeki farklı partiye ait İ2 işinin işlem süresinin bittiğini ve M3 makinesinde işlemine devam

ettiğini gösterir. M3 makinesinde İ2 işi devam ederken başka bir partiye ait olan İ2 işi ise M1 makinesinde işlemine devam etmektedir. Şekil1. ve şekil2. deki gantt şemaları karşılaştırıldığında şekil2. deki gantt şemasının işlem bitim süresi daha kısa olmaktadır.

### 3. Kısıtlı En İyileme Problemlerinde GA Kullanımı

Atölye çizelgeleme problemleri kısıtlı en iyileme problemleri olarak değerlendirilmektedir. GA'nın kullanıldığı kısıtlı en iyileme problemlerinde karşılaşılan en önemli zorluk ise klasik genetik operatörlerin uygulanmasıyla kısıtları sağlamayan dizi yapılarının elde edilmesidir. Bu sorunu ortadan kaldırmak için bu çalışmada ceza fonksiyonu yaklaşımı kullanılmıştır.

#### 3.1 Genetik Algoritma (GA)

Goldberg'e göre GA rastlantısal arama tekniklerini kullanarak çözüm bulmaya çalışan, değişken kodlama esasına dayanan sezgisel bir arama tekniğidir. GA'nın en temel özellikleri şunlardır; GA, değişken kodlarıyla uğraşır. GA, çözümü çözümler uzayında (yığın) arar. GA, ne yaptığını değil nasıl yaptığını bilir. GA amaç işlevini kullanır, sapma değerleri veya diğer hata faktörlerini kullanmaz. GA'nın uygulanmasında kullanılan operatörler rastlantısal yöntemlere dayanır, belirli ve kesin yöntemler kullanılmaz [9,10]. Bu çalışmada kullanılan GA'nın akış diyagramını gösteren pseudo-code aşağıda verilmiştir.

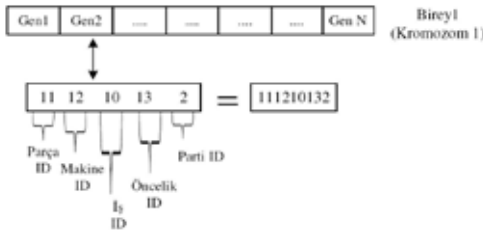
```
Genetik Algoritma {  
    Başlangıç popülasyonunu oluştur  
    Kriterleri belirle  
    do  
    {  
        Kromozomları uygunluk  
fonksiyonuna göre değerlendir  
        Kromozomları eşleştirme havuzuna  
gönder  
        Kromozomlara çaprazlama  
operatörünü uygula  
        Kromozomlara değişim operatörünü  
uygula
```

```
Popülasyondaki tüm kromozomlara  
tamir operatörünü uygula  
Her bir kromozomun uygunluk  
değerine göre Roulett tekerleği  
yöntemini uygula  
Yeni popülasyonu bir önceki  
popülasyondan oluştur  
} while (!(iterasyon sonu mu) veya  
!(iyileşme durdu mu))  
En iyi kromozomu sonuç olarak al  
}
```

### 3.2 GA'da Genetik Operatörler

GA'nın temel işleyişini oluşturan ve yürütücülüğünü belirleyen kısım ise genetik operatörlerin uygulanmasıdır. Bu operatörler, değişken kodlama operatörü, üreme operatörü, çaprazlama operatörü ve değişim operatörü, uygunluk fonksiyonu ve seçim operatörüdür. Bunlara ilaveten kısıtlı eniyileme problemlerinde mutlak suretle kullanılması gereken ve probleme özgü olarak geliştirilen diğer bir operatörde tamir (düzeltme) operatörüdür.

GA'nın uygulanmasında ilk adım, problem için arama uzayını en iyi temsil eden kodlama yapısının seçilmiş olmasıdır. Bu çalışmada her bir genin içeriği onluk tabandaki sayılarla oluşturulmuştur. GA'da kullanılan kromozom ve gen yapısı şekil 3'de gösterilmiştir.



Şekil3. Kromozom ve gen yapısı.

GA'nın başlangıcında uzay büyüklüğünü ve bu uzay içindeki bireylerin belirleneceği başlangıç popülasyonunun oluşturulması gerekmektedir. Başlangıç popülasyonunun büyüklüğü ise GA'nın etkinliği açısından önemlidir. Popülasyon büyüklüğü genellikle 30-100 arasında se-

çilmektedir [1-9]. Popülasyon büyüklüğünün küçük değerde olması çözüm uzayının küçük olmasını bu da aranan en iyi çözüm değerlerine ulaşılamamasına neden olmaktadır. Çözüm uzayının çok büyük değerde olması ise hem GA'nın etkinliğini azaltmakta hem de çözümün farklı noktalarda aranmasına neden olmaktadır. Bu çalışmada popülasyon büyüklüğü 40 olarak alınmıştır.

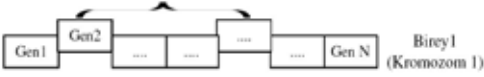
Genetik operatör olarak çaprazlama ve değişim (mutasyon) operatörlerinin uygulanma yüzdeliği ve biçimi problemin tipine ve büyüklüğüne göre değişmektedir. Çaprazlama operatörü, yan yana gelen iki kromozomda (anne ve baba birey) karşılıklı gen yapılarının değişimini sağlamaktadır. Çaprazlama oranı genellikle %50 ile %95 aralığında seçilmektedir [6-12]. Bu çalışmada çaprazlama oranı %90 olarak seçilmiş ve uygulama yöntemlerinden biri olan çok noktalı çaprazlama yöntemi kullanılmıştır. Bunun sebebi ise dizi yapılarının kodlamadan ve makine-iş bağıntısından dolayı çok büyük gen sayısına sahip olmasıdır.

Çaprazlama operatörünün her nesile uygulanmasından sonra ilerleyen nesillerde birbirinin aynı gen yapılarına sahip diziler oluşmaktadır. Bu durumu ortadan kaldırmak için değişim (mutasyon) operatörü uygulanır. Bu operatörün amacı birbiri tekrarı olan ve çeşitliliğin azaldığı durumda, aynı kromozomdaki genlerin yerinin değiştirilmesidir. Ancak bu operatörün uygulanma yüzdesinin fazla olması istenen sonuç değerlerinden çok uzaklaşmaya, çözüm uzayının farklı noktalara kaymasına neden olmaktadır. Bu nedenle değişim operatörünün uygulanma yüzdeliği %0.5 ile %15 arasında belirlenmektedir[8-12]. Burada bu oran %1 olarak seçilmiş ve değişim operatörü rastgele seçilen iki genin değişimi kuralından yararlanılarak gerçekleştirilmiştir.

Çaprazlama operatörüne ait yapı şekil 4'de değişim operatörüne ait yapı şekil 5'de gösterilmiştir.

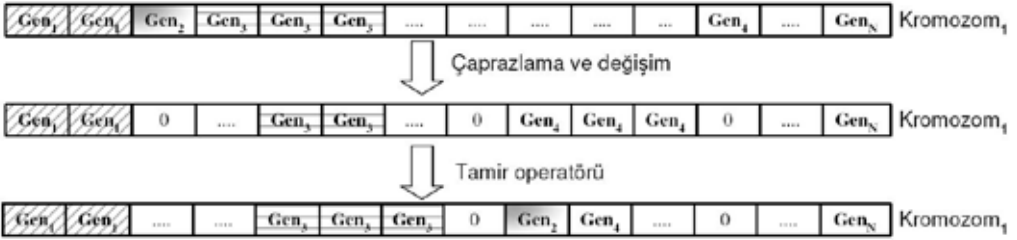


Şekil 4. Çok noktalı çaprazlama işlemi



Şekil 5. Rastgele iki gen değişimi

Tamir operatörü (düzenleyici algoritma) genetik operatörlerin uygulanmasından sonra uygunluğu bozulan kromozomları istenilen bilgilere uygun duruma getirmek için probleme özgü olarak tasarlanan algoritmadır. Çaprazlama ve değişim operatörlerinden sonra oluşan yeni dizideki gen yapıları gerçekte olmayan yapılara dönüşmektedir. Döngü boyunca bu işlem tekrarlandığında ise GA düzgün çalışmayacak ve sonuçlar çözüm uzayında farklı noktalara giderek hem çözümden uzaklaşılacak hem de çözümler yok olacaktır. Tamir operatörüne ait yapı şekil 6'da gösterilmiştir.



Şekil 6. Genetik işlem sonrası dizi durumu

Uygunluk fonksiyonu, kromozomların çözümde gösterdikleri başarı derecesini belirleyen bir değerlendirme işlevidir. Hangi kromozomların bir sonraki nesile aktarılacağı ve hangi kromozomların yok olacağı uygunluk değerlerinin büyüklüğüne göre karar verilir. GA'da kullanılan değerlendirme işlevi veya uygunluk fonksiyonu problemin amaç işlevini oluşturmaktadır. Uygunluk fonksiyonun formül olarak gösterimi aşağıda yapılmıştır.

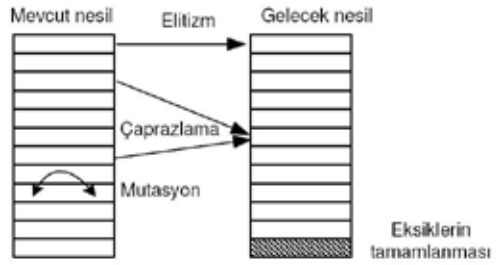
Tüm kısıtlar için bir puanları toplamı;

$$\sum_{k=0}^t \sum_{i=0}^n \sum_{j=0}^l P_k * C_{ij}$$

Uygunluk fonksiyonu;

$$f(i) = \frac{1}{1 + \left( \sum_{k=0}^t \sum_{i=0}^n \sum_{j=0}^l P_k * C_{ij} \right)}$$

l= kromozomdaki gen sayısı, n= popülasyondaki kromozom sayısı, t= kısıt sayısı, Cij = i. kromozomdaki ceza puanına sahip j. gen, Pk = k. kısıta ait ceza puanı.



Şekil 7. Üretim operatörü

Üretim operatörü ise mevcut yığından bir nesil sonraki nesile aktarılacak olan dizilerin seçilme işlemidir. Taşınan diziler, genetik olarak mevcut nesilde (yığında) en uygun yapıya sahip olan dizilerdir. Bu işlem belirlenen uygunluk değerlerine sahip iyi bireylerin bir sonraki nesile aktarılmasını sağlar. Bu seçimi yapacak olan en basit ve kullanışlı olan seçim mekanizması rulet tekerleği seçimidir. Her nesilde en iyi bireyin saklanarak bir sonraki nesile aktarılmasıyla bir sonraki nesilin kötüye gitmesi engellenmelidir. Bu işleme elitizm denmekte



| Durum   | Ürün sayısı kısıtı | Ürün sayısı | Kromozom uzunluğu | Toplam popülasyon uzunluğu | Döngü | Gantt şemasının bitim süresi (t zaman) | Programın gerçek çalışma zamanı (dakika) |
|---------|--------------------|-------------|-------------------|----------------------------|-------|--|--|
| D1: 3x5 | Önemli değil       | Tekli       | 170               | 6800                       | 2512  | 32                                     | 9 dk.                                    |
| D2: 3x5 | Önemli değil       | Çoklu       | 220               | 8800                       | 17426 | 51                                     | 145 dk.                                  |
| D3: 3x5 | Önemli             | Çoklu       | 220               | 8800                       | 18890 | 39                                     | 157 dk.                                  |

**Tablo 1.** Atölye ortamını ve buna bağlı olarak programın sonuç değerlerini gösteren tablo

Atölye çizelgeleme probleminin çözümü gerçekleştirilirken kullanılan kısıtlar; öncelik kısıtı, parçaların çakışmaması, işlerin bir arada tutulması, ürün sayısı (sipariş sınıflandırması) kısıtı ve boş durma (aylak) zamanı kısıtlarına bağımlı kalınmıştır. Şekil 9,10 ve 11’de örnek olarak 3x5 (3 iş-5 makine) durumuna ait atölye ortamlarının ürün sayısı tek veya çoklu ve kısıtının sağlanmadığı veya ürün sayısı kısıtının sağlandığı durumları gösteren 3 ayrı duruma (D1, D2, D3) ait gantt şemaları gösterilmiştir. Tablo1’de atölye ortamını, GA’daki kromozom ve popülasyon büyüklükleri ve sistemin gerçek zamanlı olarak çalışma sürelerini içeren bilgiler verilmiştir.

D1 durumunda ürün sayıları tek kabul edilmiş dolayısıyla ürün sayısı kısıtına uyulmamıştır. Bu durum gerçek atölye ortamının ifade edilmesini engellemektedir. D2 ve D3 durumlarında ise ürünler çoklu olarak nitelendirilmiş ve bu iki atölye ortamını arasında ürün sayısı kısıtı durumuna göre değerlendirmeler yapılmıştır. D2’ye ait gantt şeması incelendiğinde bu tür atölye ortamında işler farklı partiye ait olsa (kodun son rakamına göre) bile aynı kodlu işler aynı anda farklı makinelerde işlem göremezler. Yani bir makinedeki işler farklı partilere ait olsa bile aynı kodlu (ilk iki kod) işler bir makinede tamamının bitmesinden sonra sıradaki makinede işlem görmektedir. Kırmızı olarak renklendirilmiş kısımlar iş çakışmalarının olduğunu göstermektedir. Tabi ki böylesi durum atölye çizelgelemede istenmemektedir. Mevcut süre içinde GA, D1,D2 ve D3 durumları tablo1 de belirtilen döngü değerinde ve D3’de ürün sayısı kısıtı dahil edilerek çözüm gerçekleştirilmiştir.

D2 ve D3 durumları birbirleriyle karşılaştırıldığında D3’e ait gantt şemasının bitim süresi D2’ye göre çok daha kısa ve makinelerin aylak zamanları çok daha az olduğu görülmektedir. Gantt şemasındaki işlere ait kodların son rakamı o işin kaçınıcı partiye ait olduğunu göstermektedir. Dolayısıyla iş çakışmaları varmış gibi gözükse de koddaki son rakam bu işlerin farklı partilere ait olduğunu belirtmektedir. D3’deki iş çakışması sadece 12t zamanında olmuştur.

Ancak ürün sayısının çoklu olarak değerlendirilmesi ve GA’nın çalıştırılması hem kromozom ve popülasyon büyüklüğünü hem de sistemin gerçek çözüm zamanını arttırmaktadır. D1 ise atölye çizelgeleme problemi tek adet ürün olarak kabul edilerek gerçekleştirilirse bu atölye ortamı gerçeği tam olarak yansıtmamaktadır.

## 5. Sonuçlar

Atölye çizelgeleme problemlerinin çözümünde genetik algoritmanın kullanımı doğrusal programlamalarda kullanılan klasik metotlara göre daha kısa zamanda ve en iyiye yakın sonucu verdiği görülmektedir. Ayrıca ürün sayısı kısıtının dikkate alınması GA’nın çalışmasını ve oluşturulan gantt şemasını değiştirmektedir. Gerçek uygulamalarda adet kavramının önemli olduğu düşünülürse gantt şemalarının oluşturulmasında bu unsurun ihmal edilmemesi gerektiği vurgulanmıştır. Sipariş sınıflandırmasının yapıldığı durumda gantt şemasının toplam bitiş zamanı bu sınıflandırmanın yapılmadığı gantt şemasına göre daha kısa olmaktadır. Ancak bu özelliğine karşın GA’nın işleyiş süresini ve en

iyiye yakın çözüm değerini bulduğu döngü sayısını arttırmaktadır. GA'nın çalışması esnasında genetik operatörler kadar bu tip problemlerde kullanılan ve atölye çizelgeleme problemine özgü olarak geliştirilen tamir operatörünün de kısıtlı en iyileme problemlerinde mutlak suretle kullanılmaları gerektiği vurgulanmıştır.

## **Kaynaklar**

[1]. Davis L. "Job shop scheduling with genetic algorithm." Proceeding of the first International Conference on Genetic Algorithms 1985; 136-140.

[2]. Liepins G-E, ve Hilliard M-R., "Genetic algorithm: foundations and applications." Annals of Operation Research 1989; 31-38.

[3]. Biegal J-E, ve Davern J-J., "Genetic algorithms and job shop scheduling." Computers and Industrial Engineering, 1990.

[4]. Chen C-L, Nepalli R-V, ve Aljber N., "Genetic algorithms applied to the continuous flow shop problem." Computers and Industrial Engineering 1996; 919-929.

[5]. Murata T., ve Ishibuchi H., "Positive and negative combination effects of crossover and mutation operators in sequencing problems." Evolutionary Computation, Proceedings of IEEE International Conference on 1996; 170175.

[6]. Aggoune R, Mahdi H-A, ve Portman M, "Genetic algorithm for the flow shop scheduling problem with availability constraints." Systems, Man, and Cybernetics, IEEE International Conference on, 2001;4:2546 - 2551.

[7]. Zribi N, Kacem I, El Kamel A, ve Borne P, "Optimization by phases the flexible job shop scheduling problem." 5th Asian Control Conference 2004; 1889-1895.

[8]. Tsujimura Y, Mafune Y, ve Gen M, "Effects of symbiotic evolution in genetic algorithms for job shop scheduling." Proceedings of the 34th Hawaii International Conference on System Science, IEEE International Conference on, 2001; 1-7.

[9]. Goldberg D-E, Genetic Algorithm in Search, Optimization, and Machine Learning. Canada: Addison-Wesley Publishing Company Inc., 1989. p. 741.

[10]. Fang H- Lan, Genetic algorithm in timetabling and scheduling. PhD. Dissertation, University of Edinburgh 1994.

[11]. Sigl B, Golub M, ve Mornar V, "Solving timetable scheduling problem using genetic algorithm." 25th Int. Conf. Information Technology Interfaces, 2003; 519-524.

[12]. Kacem I, "Genetic algorithm for the flexible job shop scheduling problem." Systems, Man, and Cybernetics, IEEE International Conference on, 2003; 4:3464-3469.