

GENETİK ALGORİTMALARIN FARKLI ÇAPRAZLAMA TEKNİKLERİYLE İKİ BOYUTLU KESME PROBLEMLERİNE UYGULANIŞI

Alev SÖKE*, Zafer BİNGÜL**

* Kocaeli Üniversitesi, Teknik Eğitim Fakültesi, Elektronik ve Bilgisayar Eğitimi Bölümü
41300 İzmit KOCAELİ

** Kocaeli Üniversitesi, Mühendislik Fakültesi, Mekatronik Mühendisliği
41300 İzmit KOCAELİ

ÖZET

Bu çalışmada, farklı çaprazlama teknikleri kullanan genetik algoritmalar (GA) ve geliştirilmiş aşağı sol (AS) algoritmasının ortak kullanımıyla 2 boyutlu giyotinsiz bir kesme problemine Matlab ortamında çözüm geliştirilmiştir. 200x200 birimlik bir alan ile sınırlanmış bir büyük parça ve yerleşecek 29 adet birbirinden farklı düzgün dikdörtgen parçadan oluşan bir test problemi üzerinde çalışılmıştır. Çalışma sonucunda aynı problem için, farklı çaprazlama tekniklerinin birbirinden çok farklı sonuçlar verdiği görülmüştür. Tüm nesil boyunca her çaprazlama tekniği için elde edilmiş uygunluk değerlerinin aritmetik ortalamalarının ve standart sapmalarının frekansları, en iyi sonucun sıralamaya dayalı çaprazlama tekniği ile, en kötü sonucun ise Stefan Jakobs çaprazlama tekniği ile elde edildiğini göstermektedir.

Anahtar kelimeler : Genetik algoritmalar, Geliştirilmiş aşağı sol algoritması, Çaprazlama teknikleri, Kesme problemleri.

APPLYING GENETIC ALGORITHMS WITH DIFFERENT CROSSOVER TECHNIQUES TO TWO DIMENSIONAL CUTTING PROBLEMS

ABSTRACT

In this study, a solution was developed for the two dimensional non - guillotine a cutting problem by using both genetic algorithms (GAs) with different crossover techniques and improved bottom left (BL) algorithm in Matlab environment. A test problem which consists of a large piece that is limited with 200x200 unit field and 29 regular individual rectangle pieces to place in. At the end of this study, it was observed that the different crossover techniques for the same problem produced very different results. The frequencies of arithmetic means and standard deviations of the fitness values obtained for each of the crossover techniques during the whole generation showed that the best result was obtained with the order base crossover technique and the worst one with Stefan Jakobs crossover technique.

Key words: Genetic algorithms; Improved bottom left algorithm; Crossover techniques; Cutting problems.

1.GİRİŞ

İki boyutlu kesme problemleri günümüzde pek çok endüstride farklı koşullar ve amaçlar içeren problemler olarak karşımıza çıkmaktadır. Problemin karmaşıklığı ve çözüm yaklaşımı, yerleştirilecek parçaların geometrisine ve problemin doğasından gelen kısıtlamalara bağlıdır. Ağaç, cam ve kağıt endüstrisi düzgün biçimli parçaların kesilmesiyle ilgilenirken, tekstil, metal ve deri endüstrisi düzgün biçimli olmayan başka bir deyişle bilinen geometrik şekiller içermeyen parçaların kesilmesiyle ilgilenmektedir (1).

İki boyutlu giyotinsiz kesme problemleri, bir büyük parça üzerinde çok sayıda küçük parçanın en elverişli yerleşimini bulmaya yardım eden eniyileme problemleri olarak bilinir. Kesme işlemi büyük parça üzerinde bir uçtan diğer uca kadar yapılmadığından giyotinsiz kesme adını alır (2). Bu problemlerde büyük parçanın alanı, yerleştirilecek parçaların toplam alanına eşit olabileceği gibi sınırsız da kabul edilebilir. Yerleştirme bir kurallar

seti veya kısıtlamalarla tanımlanır. Kesme problemlerinin amacı yerleştirmenin yapılacağı büyük parçanın kullanılabilirliğini arttırmak ve böylelikle kullanılmayan alanı başka bir deyişle firesi en az olan yerleşim planını bulmaktır (3). Yerleşim planı küçük parçaların büyük parça üzerinde yerleşimini gösterir. Yerleşim planında kullanılmayan alana yer almayan parçalara fire denir. Kesme problemleri, yerleştirilecek parça sayısı 10 -15 arasında olduğunda kolay problemler, parça sayısı 25-30 arasında olduğunda zor problemler olarak adlandırılır. Fire değerleri kolay problemler için %0'dan başlayan zor problemler için %8'e kadar artan miktarlarda değiştiğinde kabul edilebilir standartlar arasındadır (2).

Kesme problemleri için en iyi çözümün bulunması, probleme ait oldukça geniş arama uzayında düzenli bir arama gerektirmektedir. Bu tip problemlerin çoğunda arama uzayı içinde yönlendirilmemiş bir arama, arama uzayının geniş olması ve çözümün katsayısal bir modelle gösterileme-

mesi sebebiyle oldukça verimsizdir. Araştırmalar, bu nedenle en iyi çözüme yakın iyi çözümleri verimli bir şekilde bulan yaklaşım teknikleri üzerinde yoğunlaşmaktadır (4). Bu sebeplerden ötürü bu çalışmada, dikdörtgen biçimindeki parçaların iki boyutlu yerleşimini genetik algoritmalar (GA) ile sağlayan bir çalışma gerçekleştirilmiştir.

Bölüm 2’de GA parametrelerinin kesme problemlerine uygulanışı ve permütasyonla dayalı problemlerde kullanılan farklı çaprazlama teknikleri ve Bölüm 3’te de en iyi yerleşim planının oluşturulması için kullanılan geliştirilmiş aşağı sol (AS) algoritması anlatılmaktadır. Bölüm 4’te günümüze kadar kesme problemleri için geliştirilen çözüm yaklaşımları ve bu çalışmadaki kesme problemlerinin çözümü için GA ve geliştirilmiş AS algoritması yardımıyla nasıl bir çözüm yaklaşımı uygulandığı anlatılmaktadır. Bölüm 5’te çalışmada kullanılan 29 parçalı test probleminin farklı çaprazlama teknikleri için elde edilen sonuçlar verilmiştir. Son olarak da Bölüm 6’da çalışmaya dair sonuçlar tartışılmıştır.

2. GENETİK ALGORİTMALAR

Kesme problemleri için bir yerleşim planı, yerleştirilecek küçük parçaların yerleşim sırasına karşılık gelen bir permütasyonla oluşturulmaktadır. Parçalarının yerleşim sırası bilinen bir kesme probleminin yerleşim planı kolaylıkla elde edilebilir. Ancak kesme problemlerinin çözümü oldukça geniş bir arama uzayında arama yapmayı gerektirmekte ve arama uzayının büyük olması nedeniyle de en iyi çözümlerin bulunması çok zaman almaktadır. Diğer taraftan olasılıksal yaklaşımlar en iyi çözümün en kısa zamanda üretimini sağlarlar. Bu yüzden kesme problemleri gibi kombiyasyonel problemlerin eniyileme uygulamalarında tercih edilmektedir. Yerel arama yöntemleri de, arama sürecinde yerel en küçük çözümde takılıp kalabilir ve daha iyi bir çözüm değerine ulaşılmasına engel olabilirler. Arama algoritmaları için bir dezavantaj sayılan bu durum karşısında daha detaylı arama yapabilen yöntemler geliştirilmiştir. (GA) sıklıkla başvurulan bu yöntemlerden birisidir.

GA, ilk olarak Holland (5) tarafından eniyileme problemleri için kullanılmıştır. Bu algoritmalar daha sonra eniyileme problemleri için çok önemli bir araç haline gelmiş ve günümüzde de hala çok değişik uygulamalarda kullanılmaktadır. Bunun nedeni GA’nın problem için tek bir çözüm

üretmek yerine arama uzayı içerisinde farklı çözümlerden oluşan çözüm kümesi üretmesidir. Çözüm kümesi içindeki bütün çözümler birbirinden tamamen bağımsızdır.

GA terminolojisinde nüfus olarak adlandırılan çözüm kümesi vektör, kromozom veya birey adı verilen sayı dizilerinden oluşur. Kesme problemleri için birer permütasyonla gösterilen bu bireylerin her biri ayrı bir yerleşim planını oluşturmaktadır. Aynı zamanda her birey problem için olası bir çözümü temsil etmektedir. Nüfus içerisindeki bu bireyler GA işlemcileri tarafından belirlenir. Çözümü aranan problemin bireyler içindeki kodlanması problemden probleme değişiklik göstermektedir. Problemin kodlanması, GA’nın problemin çözümündeki başarısına karar vermedeki en önemli faktördür. Bazı kodlamalar özelleştirilmiş genetik işlemlerin kullanımını gerektirirler (6). Kesme problemlerinde, problemin bireyler içindeki kodlanması tamsayılar kullanılarak gerçekleştirilmektedir. Bu kodlamanın en büyük avantajı permütasyon içindeki sıralamanın değişimiyle yeni bir yerleşim planının kolayca oluşturulabilmesidir. Uygunluk fonksiyonu, GA’da probleme özel olarak kurulan tek kısımdır. Bu fonksiyon, bireyleri problemin parametreleri haline getirir ve bireylerin uygunluk değerlerine bakarak herhangi bir bireyin problem için ne kadar uygun bir çözüm olacağına karar verir (7). GA’nın problemin çözümündeki başarısı, çoğunlukla bu fonksiyonun etkin kurulmasına bağlıdır. Bu çalışmada kullanılan uygunluk fonksiyonu, büyük parça üzerinde kullanılmayan alanla başka bir deyişle fireyle ters orantılı olarak değişmektedir. GA’da nüfus içindeki bireylere ait genetik bilginin değişimi ve yeni bireylerin üretimi için evrim yasalarındaki seçim, çaprazlama ve mutasyon gibi işlemler bilgisayar ortamında taklit edilmektedir. Bu çalışmada rulet tekerleği seçimi, yer değiştirme mutasyonu ve 6 farklı çaprazlama tekniği kullanılmıştır.

Seçim işlemcisi: Seçim işleminin amacı, uygunluk değerleri yüksek bireylerin nesiller aracılığıyla daha çok üretilmelerine imkan vermektir. Bu çalışmada kullanılan rulet tekerleği yöntemi en yaygın kullanılan yöntemdir. Rastlantısal bir yöntem olan rulet tekerleği yönteminde, bireylerin seçilme olasılıkları sürekli bir çizgi üzerine kesitler halinde yerleştirilir. Bireylerin bu çizgi üzerindeki kesit uzunlukları, uygunluk değerlerine bağlı olarak hesaplanan seçilme olasılıklarına eşittir. Seçim işlemi yapılırken kesitlerin toplam uzunluk

aralığında rasgele bir sayı üretilir. Bu sayı hangi kesit üzerine düşüyorsa o kesitin ait olduğu birey seçilmiş olur.

Mutasyon işlemcisi: Nüfus içerisinde tek tip bireyler varsa çaprazlamalarından elde edilecek bireylerde yine aynı tip olur. Dolayısıyla nüfusu oluşturan bireyler arasında çeşitlilik olmaz. Bunu engellemek için mutasyon işlemcisi kullanılır. Bu çalışmada kullanılan işlemci, yer değiştirme mutasyon işlemcisidir. Bu işlemcide rasgele bir sayı seçilir ve önceden belirli mutasyon oranıyla karşılaştırılır. Eğer seçilen sayı mutasyon oranının

Çocuğun ilk geni babanın ilk geni olur. Babanın ilk geninin annede karşılık gelen geni babadan alınır. Alınan bu gene karşılık gelen annedeki gen yine babadan seçilir. Böylelikle ilk alınan genle karşılaşınca dek bu işlem devam ettirilir. Bu işleme bir devir denir. Çocuk için bu devir kullanılarak babadan alınacak sabit genler belirlenir. Daha sonra çocuğun eksik kalan genleri sırasıyla anneden alınarak çocuk birey oluşturulmuş olur. Şekil 2’de bu çaprazlama tekniğine ait bir örnek verilmiştir. Babanın ilk geni (1) alınır, babadaki bu gen annedeki 9 genine karşılık gelmektedir. Babadaki, 9 geni de annedeki 7 genine karşılık

Baba	1	2	3	4	5	6	7	8	9	10
Çocuk	1	2	3	4	9	10	6	8	7	5
Anne	9	10	1	3	2	4	6	8	7	5

Şekil 1. Sıralamaya dayalı çaprazlama.

dan küçükse, birey içinde rasgele iki genin yeri değiştirilerek yeni bir birey üretilir. Aksi durumda birey değişikliğe uğramadan olduğu gibi kalır.

Çaprazlama işlemcisi: GA içinde çaprazlama işlemi bireyler arasındaki bilgi değişimini gerçekleştirerek daha iyi bireylerin üretilmesini sağlar. Çaprazlama yapılırken iki bireyin belirli genleri karşılıklı olarak yer değiştirir. Böylelikle iki yeni birey başka bir deyişle iki yeni olası çözüm üretilmiş olur.

Permütasyona dayalı çaprazlama teknikleri:

Sıralamaya dayalı çaprazlama (Order based crossover - OBX): Başlangıçta rasgele bir kesim noktası belirlenir. Çaprazlamanın yapılacağı baba olarak isimlendirilen ilk bireyden kesim noktasına kadar olan genler yeni bireyin diğer bir deyişle çocuğun ilk genlerini oluşturur. Çocuğun diğer genleri annenin kullanılmayan genlerinden aynı sırayla alınarak tamamlanır. Bu çaprazlama tekniği için bir örnek Şekil 1’de görülmektedir. Kesim noktası 4 seçtiğimizde babanın ilk 4 geni (1, 2, 3, 4) çocuğa kopyalanır ve çocukta eksik kalan diğer genler de (9, 10, 6, 8, 7, 5) anneden alınarak çocuk üretilir (2).

Devirli çaprazlama (Cycle crossover - CX): Bu çaprazlamaya babanın ilk geninden baş-

gelmektedir. Bu devir takip edildiğinde babadan alınacak genler (1, 9, 7, 6, 4, 3) sırasıyla belirlenir. Bu genler çocuk içerisine babadaki konumuna göre yerleştirilir. Eksik kalan genler de anneden tamamlanır (8).

Doğrusal çaprazlama (Linear crossover-LX): Rasgele iki kesim noktası belirlenir. Babadan bu kesim noktaları arasında kalan genler çocuk içindeki aynı konuma kopyalanır. Daha sonra babadan alınmayan diğer genler anneden sırasıyla alınarak çocuk oluşturulmuş olur. Bu çaprazlama tekniği Şekil 3’te bir örnekle gösterilmektedir. Birinci kesim noktasını 3, ikinci kesim noktasını 6 seçtiğimizde çocuk için babadan alınan genler (4, 5, 6) olarak belirlenir. Bu genler çocukta aynı konuma yerleştirilir ve çocuğun babadan alınmayan diğer genleri (9,10,1 3, 2, 8, 7) sırasıyla anneden alınarak tamamlanır (9).

Kısmi eşleştirmeli çaprazlama (Partially matched crossover - PMX): Rasgele iki kesim noktası belirlenir. Babadaki bu iki kesim noktası arasında kalan genler çocuktaki aynı konuma kopyalanır. Çocuktaki eksik kalan diğer genler başlangıçta annenin bu iki kesim noktası dışındaki genlerinden aynı sırayla alınarak tamamlanır. Fakat burada çocuk içinde tekrarlanan genler olabilir ve

Baba	1	2	3	4	5	6	7	8	9	10
Çocuk	9	2	3	1	5	6	4	8	7	10
Anne	9	10	1	3	2	4	6	8	7	5

Şekil 5. Düzenli çaprazlama

bu birey içinde istenmeyen bir durumdur. Bu durumda tekrarlanan gen kesim noktaları arasındaki babanın annede karşılık gelen geniyle değiştirilir. Şekil 4'te bu çaprazlama tekniği bir örnekle gösterilmektedir. 1. kesim noktasını 2, 2. kesim noktasını 6 seçtiğimizde çocuk için babadan alınan genler (3, 4, 5, 6) olarak belirlenmektedir. Bu genler çocuktaki aynı konuma kopyalanır ve çocuğun diğer genleri anneden sırayla alınarak tamamlanır (9, 10, 6, 8, 7, 5). Yalnız burada 6 ve 5 genleri tekrar yapmak-tadır. Bunu engellemek için tekrar yapan bu genler sırasıyla babanın annede karşılık gelen 1 ve 2 genleriyle yer değiştirerek çocuğun üretilmesi sağlanır (2,8).

Düzenli çaprazlama (Uniform crossover – UX): Öncelikle babanın tüm genleri çocuğa kopyalanır. Boyutu birey boyutunun bir eksiği kadar, değerleri 0 ve 1 olmak üzere rasgele değişen bir vektör oluşturulur. Oluşturulan bu vektörün 1 değerine sahip elemanlarının indisleri babanın bir kopyası olan çocuktaki değiştirilecek genleri göstermek için kullanılır. Babadaki bu indislere sahip genlerin annede karşılık gelenlerinin indis değerleri alınır ve sıralanır. Daha sonra sıralanan bu indis değerlerindeki genler anneden alınarak çocukta önceden tespit edilmiş değiştirilecek genlerin yerine yazılır. Ve böylece çaprazlama gerçekleştirilmiş olur. Şekil 5'te bu çaprazlama tekniği bir örnekle gösterilmektedir. Çocuğun değişecek (1, 4, 7, 8, 9) konumlarındaki genleri yerine anneden bu genlere karşılık gelen (9, 1, 4, 8, 7) genleri çocuğa kopyalanarak yeni bir birey elde edilmiş olur (9).

Stefan Jakobs çaprazlama (Stefan Jakobs crossover – SJX): Öncelikle rasgele 2 sayı p ve q belirlenir. Bunlardan biri (p) çaprazlamaya hangi noktadan başlanacağını diğeri de (q) p noktasından itibaren kaç gen alınacağını gösterir. Babanın p.geninden itibaren q adet gen alınır ve bu genler çocuğun ilk genlerini oluşturur. Çocuğun babadan alınmayan diğer genleri de anneden sırayla alınarak çaprazlama tamamlanır. Şekil 6'da bu çaprazlama tekniği bir örnekle gösterilmektedir. p ve q sayısı sırasıyla 2 ve 4 olarak alınır. Babanın 3.geninden itibaren 4 gen alınarak çocuğun ilk genleri olarak kopyalanır. Çocuğun eksik kalan diğer genleri de (9, 10, 1, 2, 8, 7) anneden sırayla alınarak tamamlanır (2, 11).

Baba	1	2	3	4	5	6	7	8	9	10
Çocuk	3	4	5	6	9	10	1	2	8	7
Anne	9	10	1	3	2	4	6	8	7	5

Şekil 6. Stefan Jakobs çaprazlama.

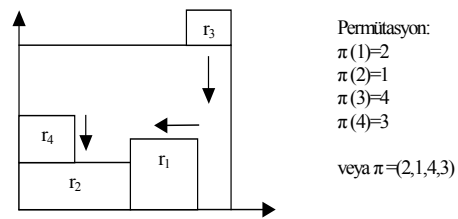
3. GELİŞTİRİLMİŞ AŞAĞI SOL ALGORİT-MASI

Geliştirilmiş AS algoritması, yerleştirme problemlerinde küçük parçaları yerleştirmek için kullanılan ve kaydırma tekniği üzerine kurulmuş bir algoritmadır (11). Bu algoritmanın işleyişi, yerleştirilecek küçük parçanın büyük parçanın sağ üst köşesine yerleştirilmesiyle başlar. Parça öncelikle mümkün olduğunca aşağıya doğru, sonra da altında kalan parçanın üst kenarları boyunca sola doğru bir köşeyle karşılaşınca kadar hareket ettirilir. Bir köşeyle karşılaşıldığında parça yine aşağıya doğru hareket ettirilir. Daha sonra mümkünse parça yine sola doğru hareket ettirilir. Bu işlemler aşağıya hareket öncelikli olmak üzere aşağıya ve sola doğru bir parça yada kenarla karşılaşınca kadar devam ettirilir. Parça sabit konumuna, çakışma olmaksızın aşağıya ve sola doğru hareketi artık gerçekleştiremez konuma geldiğinde ulaşır. Parçanın tamamının büyük parça içine yerleştirilemediği durumlarda, parça çıkartılır diğer bir deyişle yerleştirilemez (2). Şekil 7'de geliştirilmiş AS algoritmasının parça yerleştirilirken nasıl çalıştığı bir örnek üzerinde gösterilmiştir. Oklar en iyi yerleşim için r_3 parçasının hareket yönünü göstermektedir.

Yerleşim planı bir π permütasyonu ile gösterilebilir.

$\pi = (i_1, \dots, i_n)$ – Permütasyon

i: dikkörtgen parça sırası (r_i)



Şekil 7. Geliştirilmiş AS algoritmasının gösterimi.

AS algoritması ve yukarıda çalışması anlatılan geliştirilmiş AS algoritmasının çalışma karakteristikleri birbirinden farklıdır (11). AS algo-

ritmasında herhangi bir koşul olmadan parçalar büyük parça üzerinde aşağı ve sol hareketleriyle kaydırılırken geliştirilmiş AS algoritmasında parçalar için aşağı hareket her zaman önceliklidir.

4. ÇÖZÜM YAKLAŞIMI

Kesme problemleri için bir yerleşim planı, yerleştirilecek küçük parçaların yerleşim sırasına karşılık gelen bir permütasyonla oluşturulmaktadır. Yerleşim planını oluşturmak için de farklı yerleşim algoritmaları kullanılmaktadır. Jakobs (1996) çalışmasında genişliği sınırlı, yüksekliği sınırsız kabul edilmiş dikdörtgen şeklindeki büyük parçanın içine küçük dikdörtgen parçaların yerleştirilmesi problemini ele almıştır (10). En küçük fire değerine sahip permütasyonu bulmak için GA'ı, yerleşim sırası bilinen parçaları yerleştirmek için AS algoritmasını kullanmıştır. Liu ve Teng (1999) çalışmalarında sınırları önceden belirlenmiş dikdörtgen şeklindeki büyük parça içine sırasıyla 25 ve 50 adet küçük dikdörtgen parçanın yerleştirilmesi problemleriyle ilgilenmişlerdir (11). Parçaların yerleşim sırasını bulmak için GA'ı, yerleşim planını oluşturmak için geliştirilmiş AS algoritmasını kullanmıştır. Her iki araştırmacının da kullandığı çaprazlama tekniği Stefan Jakobs çaprazlama tekniğidir. Leung (2001) sınırları önceden belirlenmiş dikdörtgen şeklindeki büyük parça üzerinde çok sayıda küçük parçanın yerleşimini ele alan 8 test problemi üzerinde çalışmıştır (2). En elverişli yerleşim planını bulmak için GA'ı, yerleşim planının oluşturulması için geliştirilmiş AS algoritmasını kullanmıştır. Diğer iki çalışmadan farklı olarak bu çalışmada 5 farklı çaprazlama tekniği kullanılmıştır. Ancak çaprazlama teknikleri ve bunların çözüm üzerine etkileri detaylı olarak incelenmemiştir.

Bu çalışmada, 200x200 birimlik bir alan ile sınırlandırılmış bir büyük parça ve yerleşecek 29 adet birbirinden farklı düzgün dikdörtgen parçadan oluşan bir test problemi üzerinde çalışılmıştır. Bu test problemi özel olarak seçildiğinden en iyi çözüm sıfır firenin elde edildiği çözümdür.

Test probleminin çözümünde kullanılan program, Matlab ortamında yazılmıştır. Program M-dosyaları kullanılarak oluşturulan fonksiyonlardan oluşmaktadır. Matlab, bilinen matematiksel gösterimlerle tanımlanan problemler ve çözümleri için hesaplama, görsellik ve programlamanın kolaylıkla bir arada kullanıldığı bir yazılım paketidir. Çoğunlukla bilimsel alanda, yüksek performans gerektiren hesaplamaların yapılması için kullanılır.

Test probleminin çözümü için yürütülen işlem adımları aşağıdaki 2 adımla kısaca özetlenebilir:

1. İlk olarak, en az fire değerini veren ve aynı zamanda parça yerleşim sırasını gösteren permütasyon GA kullanılarak elde edilir.
2. Elde edilen en iyi permütasyon kullanılarak geliştirilmiş AS algoritması ile büyük parça üzerinde parçaların yerleşimi, gerçekleştirilir.

Yerleştirme işlemi aşağıdaki kabuller göz önünde bulundurularak yapılmıştır (2):

1. Bütün parçalar sabit bir yöndedir. Uzunluğu l , genişliği w olan parça eğer parçanın boyutları birbirine eşit değilse uzunluğu w , genişliği l olan parçadan farklıdır.
2. Küçük parçalar, büyük parça üzerinde dik olarak yerleştirilmektedir. Böylece küçük parçaların kenarları büyük parçanın kenarına paralel olur.
3. Hiçbir parçanın eni ve boyu, yerleştirileceği büyük parçanın boyutlarını aşamaz.
4. Büyük parça ve küçük parçaların boyutları tamsayıdır.
5. Büyük parça üzerindeki bütün kesimler çok küçük bir kalınlıktadır. Yerleştirilecek küçük parçaların kenar kalınlıkları yer tutmaz.
6. Her küçük parça, büyük parça üzerinde herhangi bir yere yerleştirilebilir. İki parçanın yan yana gelmemesi gibi bir sınırlama yoktur.

5. BULGULAR

Yerleştirme simülasyonları, GA için 6 farklı çaprazlama tekniği kullanılarak yapılmış ve çaprazlama tekniklerinin bu problemlerin çözümleri üzerindeki etkileri incelenmiştir.

GA'da kullanılan işlemciler ve parametre değerleri:

Nüfus büyüklüğü	: 80
Nesil	: 1000
Mutasyon oranı	: 0.7
Mutasyon işlemcisi	: Yer değiştirme mutasyonu
Gösterim	: Tamsayı kodlama.
Seçim işlemcisi	: Rulet tekerleği seçimi.

Çaprazlama işlemcileri : En küçük fire değerinin elde edildiği çaprazlamadan başlayarak sırasıyla:

- Sıralamaya dayalı çaprazlama (Order based crossover - OBX)
- Devirli çaprazlama (Cycle crossover – CX)
- Doğrusal çaprazlama (Linear crossover - LX)
- Kısmi eşleştirmeli çaprazlama (Partially matched crossover - PMX)
- Düzenli çaprazlama (Uniform crossover – UX)
- Stefan Jakobs çaprazlama (Stefan Jakobs crossover – SJX)

İki boyutlu giyotinsiz kesme problemlerinin çözümü için GA ve geliştirilmiş AS algoritması kullanılarak yapılan önceki çalışmalarda mutasyon oranı artışlarının problemin çözümü üzerinde etkili olmadığı buna karşılık nüfus büyüklüğünü arttırdıkça daha iyi çözümlere ulaşıldığı görülmüştür (12). Bu nedenle çalışmada mutasyon işlemcisi yer değiştirme mutasyonu, mutasyon oranı 0.7 ve nüfus büyüklüğü 80 olarak belirlenmiştir.

Yerleştirme simülasyonları sonucunda:

29 adet parçanın yerleştirilmesi için sözü edilen GA işlemcileri ve parametreleri kullanılarak yapılan çalışmalardan elde edilen fire değerleri Tablo 1’de gösterilmektedir. Aynı problem için sadece çaprazlama tekniklerinin değiştirilmesiyle yapılan çalışmalarda birbirinden farklı fire değerleri elde edilmiştir.Tablodan görüldüğü gibi en küçük fire değeri OBX tekniği kullanıldığında, en büyük fire değeri SJX tekniği kullanıldığında elde

edilmiştir. Firenin en küçük olması yerleşim planının en iyi şekilde gerçekleştiğini göstermektedir. Tablodaki diğer çaprazlama teknikleri için elde edilen bu fire değerlerine bakılarak GA’da çaprazlama işlemcisinin çözüm üzerinde önemli derecede etkisi olduğu söylenebilir.

Tablo 1. 6 farklı çaprazlama tekniği için yapılmış çalışmalar sonucunda elde edilen fire değerleri.

Çalışma No	Çaprazlama Teknikleri	Fire Değerleri
1	OBX	0.0598
2	CX	0.0642
3	LX	0.0740
4	PMX	0.0762
5	UX	0.0910
6	SJX	0.0992

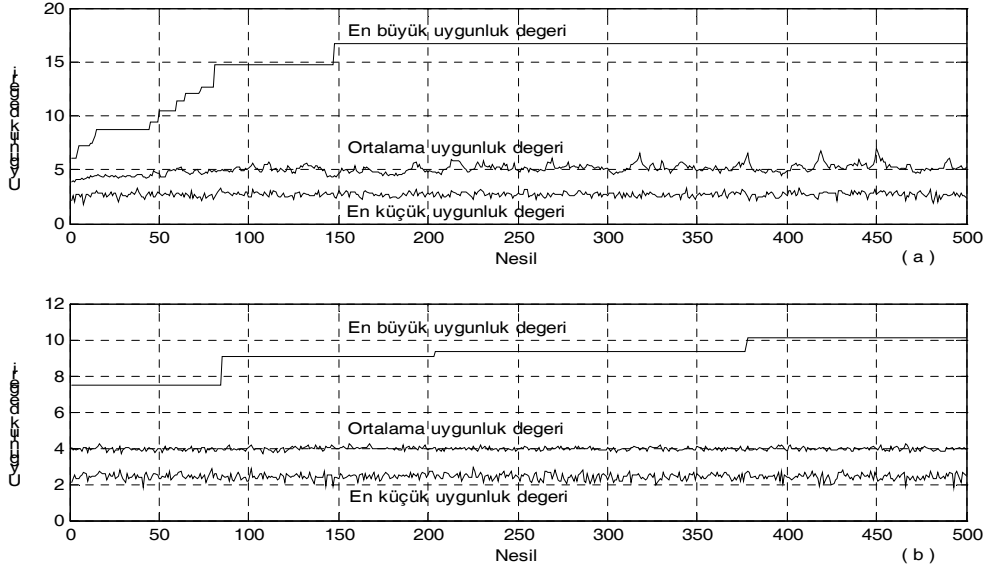
OBX ve SJX tekniklerinin kullanılmasıyla yapılan çalışmalardaki uygunluk değerleri için hesaplanan olasılıksal değerler Tablo 2’de gösterilmektedir. Hesaplanan bu değerler seçilmiş nesillerdeki (1, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000) her bir nüfusa diğer bir deyişle 80 bireye ait olasılıksal değerlerdir. OBX tekniği için hesaplanan uygunluk değerlerine ait aritmetik ortalama değerlerinin, SJX tekniği için hesaplanan değerlerden daha büyük olduğu görülmektedir. Ayrıca tablodan bakıldığında OBX tekniği için hesaplanmış standart sapma değerleri, SJX tekniği için hesaplananlara göre daha büyük değerler arasında değişim göstermektedir. OBX tekniği kullanılarak yapılan çalışmada uygunluk değerlerinin nesiller boyunca artış gösterdiği görülmektedir. Tüm bu sonuçlar OBX tekniğinin nüfus içinde

Tablo 2. OBX ve SJX teknikleri kullanılarak elde edilen uygunluk değerlerine ait olasılıksal değerler.

Çaprazlama teknikleri	Nesil no	Uygunluk değerlerine ait olasılıksal değerler					
		Aritmetik ortalama	Medyan	Geometrik ortalama	Standart sapma	En büyük u.değeri	En küçük u.değeri
OBX	1	3,8502	3,6746	3,7784	0,7633	6,0698	2,1674
OBX	100	5,1112	4,4472	4,7629	2,3518	14,6843	2,5332
OBX	200	4,7759	4,4682	4,5419	1,9611	16,7224	2,8821
OBX	300	5,1914	4,9460	4,9306	1,9822	16,7224	2,6508
OBX	400	4,8772	4,1258	4,5538	2,3809	16,7224	2,6513
OBX	500	5,2050	4,5132	4,8517	2,3501	16,7224	2,3770
OBX	600	5,5520	5,1494	5,1161	2,8012	16,7224	2,7012
OBX	700	5,4375	5,0697	5,1243	2,1983	16,7224	2,6804
OBX	800	7,0810	5,3211	6,1763	4,0552	16,7224	2,8129
OBX	900	5,0124	4,4746	4,6806	2,4559	16,7224	2,7461
OBX	1000	4,9849	4,7308	4,7653	1,8333	16,7224	2,3369
SJX	1	4,0224	3,9926	3,9244	0,9173	7,4724	2,1324
SJX	100	4,0420	3,9390	3,9478	0,9444	9,0888	2,1755
SJX	200	3,8810	3,6234	3,7944	0,9322	9,0888	2,5259
SJX	300	4,0643	3,8560	3,9311	1,1348	9,3262	2,2231
SJX	400	3,8250	3,7152	3,7303	0,9926	10,0781	2,3802
SJX	500	3,9130	3,7817	3,8114	1,0299	10,0781	2,4480
SJX	600	4,0759	3,9313	3,9673	1,0644	10,0781	2,5487
SJX	700	4,1131	3,8954	3,9926	1,1183	10,0781	2,4392
SJX	800	3,9283	3,7856	3,8087	1,0937	10,0781	2,4066
SJX	900	3,9259	3,6839	3,8119	1,0852	10,0781	2,2873
SJX	1000	4,0701	3,7880	3,9542	1,0985	10,0781	2,4350

farklı bireyler üretmede başarılı olduğunu göstermektedir.

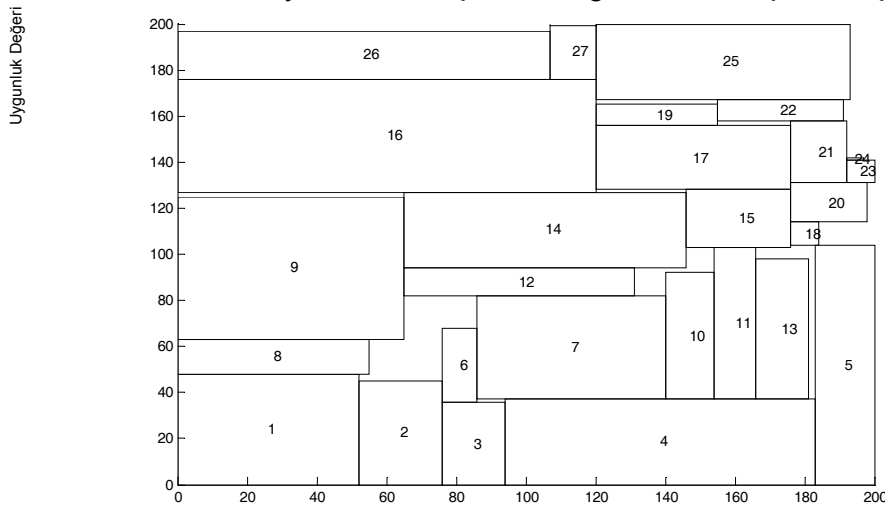
kullanılan GA'da nüfus içerisindeki bireyler birbirinden ne kadar farklı olursa o oranda farklı bi-



Şekil 8. (a) OBX tekniğinin kullanılmasıyla elde edilen en büyük, ortalama ve en küçük uygunluk değerleri.
(b) SJX tekniğinin kullanılmasıyla elde edilen en büyük, ortalama ve en küçük uygunluk değerleri

En iyi ve en kötü fire değerlerinin elde edildiği OBX ve SJX teknikleri kullanılarak yapılan çalışmalarda en büyük uygunluk, ortalama uygunluk ve en küçük uygunluk değerlerine ait grafikler sırasıyla Şekil 8'de gösterilmektedir. Yerleşim planları 1000 nesil boyunca tekrarlanmaktadır. Bu yüzden grafiklerde sadece 500. nesle kadar elde edilen uygunluk değerleri görülmektedir. Şekil 8a'da OBX tekniğinin kullanılmasıyla elde edilen uygunluk değerleri için çizilmiş grafik görülmektedir. Permütasyon hesabı için

reylar elde edilir. Böylece permütasyon hesabı daha sağlıklı yapılmış olur. Nüfus içerisindeki bu çeşitliliğin bir göstergesi olarak Şekil 8a'da görüldüğü gibi OBX tekniği için ortalama uygunluk değeri nesil ilerledikçe değişim içerisindedir. Şekil 8b'de de SJX tekniğinin kullanılmasıyla elde edilen uygunluk değerleri için çizilmiş grafik görülmektedir. Bu çalışmadaki ortalama uygunluk değeri OBX tekniği ile yapılan çalışmanın aksine bir değişim sergilememektedir. Bu da nüfus içinde farklı bireyler elde edilemediğini dolayısıyla SJX tekniğinin nüfus içindeki çeşitliliği sağlayacak



Şekil 9. OBX tekniği kullanılarak yapılmış çalışmaya ait yerleşim planı.

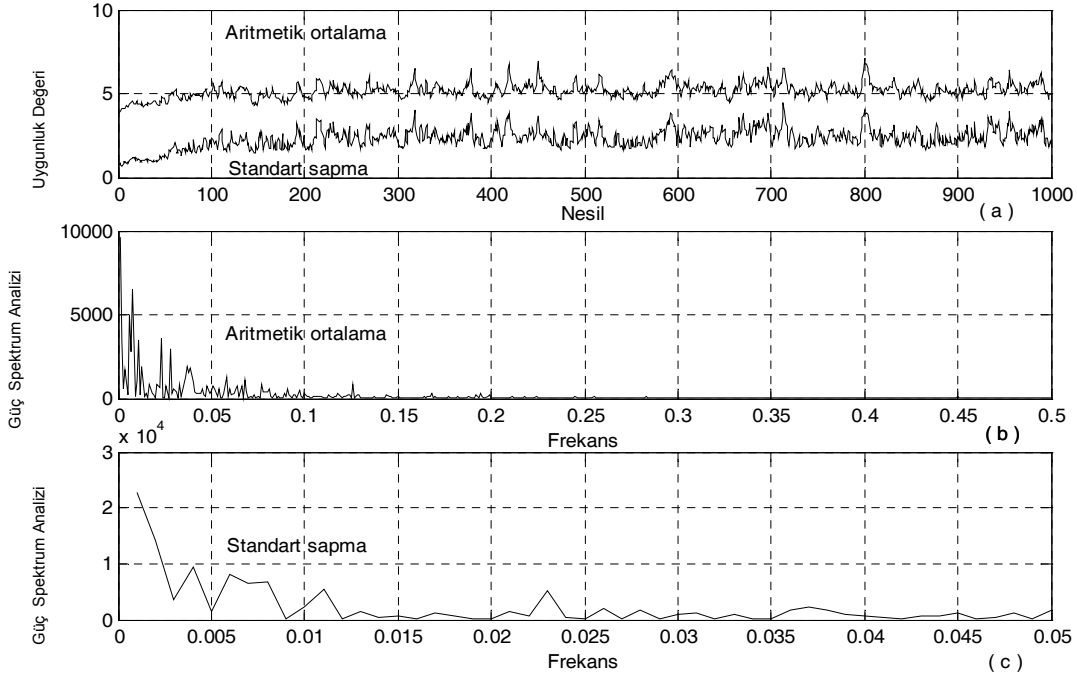
verimli bir çaprazlama tekniği olmadığını göstermektedir.

Şekil 9'da en küçük firenin elde edildiği OBX tekniği kullanılarak yapılmış çalışmaya ait yerleşim planı görülmektedir.

Permütasyon problemlerinde amaç, sonsuz değişimi elde etmektir. Sonsuz değişim, nesiller boyunca her neslin bir önceki nesilden daha iyi nüfus üretmesini ve nüfus içindeki bireylerin birbirinden farklı olmasını gerektirir. Bu açıdan bakıldığında her bir nesilde elde edilen nüfusun uygunluk değerlerinin aritmetik ortalamasının ve standart sapmasının mümkün olduğu kadar değişim göstermesi istenir. Bu sebeple en iyi ve en kötü fire değerlerinin elde edildiği OBX ve SJX teknikleri için, her bir nesildeki aritmetik ortalama ve standart sapmanın güç spektrum analizi yapılmıştır. Şekil 10'da OBX tekniği kullanılarak ya-

dır. Şekil 10 ve Şekil 11'deki grafikleri karşılaştırdığımızda, OBX tekniği için elde edilmiş olan uy-

aritmetik ortalamasının genliği güç spektrum analizinde 0 ile 1×10^4 değerleri arasında değişirken, SJX tekniği için bu değer yaklaşık olarak 0 ile 70 arasında değişmektedir. Aynı şekilde uygunluk değerlerinin standart sapmasının genliği güç spektrum analizinde OBX tekniği için 0 ile 2×10^4 değerleri arasında değişirken SJX tekniği için yaklaşık olarak 0 ile 700 arasında değişmektedir. Bu sonuçlardan görülmektedir ki; OBX tekniği kullanılarak yapılan çalışmada uygunluk değerlerine ait aritmetik ortalama ve standart sapma değerleri nesiller boyunca belli frekanslarda değişim göstermektedir. SJX tekniği kullanılarak yapılan çalışmada ise uygunluk de-



Şekil 10. OBX tekniği kullanılarak yapılmış çalışmaya ait uygunluk değerlerinin

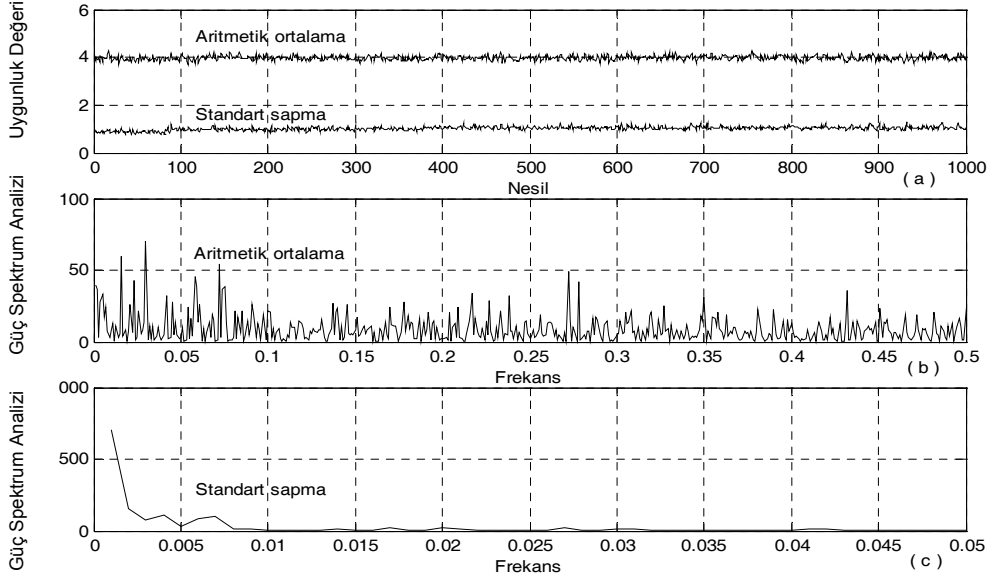
- (a) aritmetik ortalama ve standart sapması,
- (b) aritmetik ortalamalarının güç spektrum analizi,
- (c) standart sapmalarının güç spektrum analizi.

pılmış çalışmaya ait, Şekil 11'de de SJX tekniği kullanılarak yapılmış çalışmaya ait uygunluk değerlerinin aritmetik ortalama ve standart sapması ve bunların güç spektrum analizleri gösterilmektedir. Bu analizler nesiller boyunca uygunluk değerlerinin nasıl bir değişim gösterdiğini açıklamakta-

ğlerine ait aritmetik ortalama ve standart sapma değerleri nesiller boyunca fazla değişim göstermemektedir. Dolayısıyla OBX tekniği kullanılarak elde edilen bireyler, SJX tekniği kullanılarak elde edilen bireylere göre nüfus içinde ve nesiller boyunca daha çok değişim sergilemektedir.

KAYNAKÇA

1. Hopper, E., Two-Dimensional Packing Utilising Evolutionary Algorithms and Other



Şekil 11. SJX tekniği kullanılarak yapılmış çalışmaya ait uygunluk değerlerinin

- (a) aritmetik ortalama ve standart sapması,
- (b) aritmetik ortalamalarının güç spektrum analizi,
- (c) standart sapmalarının güç spektrum analizi.

6. SONUÇ

Bu çalışmada, 29 adet birbirinden farklı dikdörtgen parçayı sınırları önceden belirlenmiş bir büyük parça üzerine yerleştirmeyi amaçlayan iki boyutlu giyotinsiz bir kesme probleminin çözümü üzerinde çalışılmıştır. Permütasyon hesabı için kullanılan GA'da 6 farklı çaprazlama tekniği kullanılmıştır. Problemin çözümünde elde edilen fire değerleri %5 ve %9 arasında değişmektedir. Çalışmaya göre en iyi fire değeri OBX tekniği kullanıldığında elde edilmiştir. Bunun nedeni OBX tekniği kullanılarak yapılan çaprazlama işlemlerinde baskın aile genlerinin yeni üretilen bireyler içinde saklı tutulmasıdır. Elde edilen en iyi fire değerinden en kötü fire değerine göre diğer teknikler CX, LX, PMX, UX, SJX olarak sıralanabilir. OBX tekniği kullanılarak elde edilen bireyler diğer çaprazlama tekniklerine göre nüfus içinde ve nesiller boyunca daha çok değişim sergilediğinden nüfus içerisinde yeterince çeşitlilik sağlamıştır. Böylelikle çözüm için arama uzayında daha geniş bir alan taranmış ve diğer çaprazlama tekniklerine göre daha küçük fire değeri elde edilmiştir.

Meta-Heuristic Methods, PhD Thesis, Cardiff University, UK, 2000.

2. Leung, T.W., Yung, C.H. and Trout, M.D., Applications of Genetic Search and Simulated Annealing to The Two-Dimensional Non-Guillotine Cutting Stock Problem, Computers & Industrial Engineering, vol.40, pp.201-214, 2001.
3. Hopper, E. and Turton, B., Application of Genetic Algorithms to Packing Problems - A Review, Proceedings of the 2nd On-line World Conference on Soft Computing in Engineering Design and Manufacturing, Springer Verlag, pp.279-288, 1997.
4. Callaghan, A. R., Nair A. R. and Lewis, K. E., An Extension of The Orthogonal Packing Problem Through Dimensional Flexibility, Proceedings of DETC'99: 1999 ASME Design Engineering Technical Conferences, vol.199, pp.12-15, 1999.
5. Holland, J.H. Adaption in Natural and Artificial Systems, University of Michigan Pres, Ann Arbor, MI, 1975.

6. Leitch, D.D., A New Genetic Algorithm for The Evolution of Fuzzy Systems, PhD Thesis, University of Oxford, UK, 1995.
7. Beasley, D., Bull, D. R. and Martin, R. R., An Overview of Genetic Algorithms: Part 1, Fundamentals, University Computing, vol.15(2), pp.58-69, 1993.
8. Goldberg, D. E., Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, Reading, MA, 1989.
9. Houck, C.R., Joines, J. and Kay, M., A Genetic Algorithm for Function Optimization: A Matlab Implementation. ACM Transactions on Mathematical Software, 1996
10. Jakobs, S., On Genetic Algorithms for The Packing of Polygons, European Journal of Operational Research, vol.88, pp.165-181, 1996.
11. Liu, D. and Teng, H., An Improved BL-Algorithm for Genetic Algorithm of The Orthogonal Packing of Rectangles, European Journal of Operational Research, vol.112, pp.413-420, 1999.
12. Söke, A. ve Bingül, Z., İki Boyutlu Kesme Problemlerinin Genetik Algoritma Yardımıyla Çözümünün İncelenmesi, XII. International Twelfth Turkish Symposium on Artificial Intelligence and Neural Networks, 2-4 July, 2003, Çanakkale, Turkey