DAİRESEL KESME PROBLEMLERİNİN ÇÖZÜMÜNDE GENETİK ALGORİTMA TABANLI BİR YAKLAŞIM KULLANILMASI

Hulusi SEYDANOĞLU

YÜKSEK LİSANS TEZİ

Endüstri Mühendisliği Anabilim Dalı

Şubat-2007

A GENETIC ALGORITHM BASED APPROACH
FOR THE CIRCULAR CUTTING PROBLEM

Hulusi SEYDANOĞLU

MASTER OF SCIENCE THESIS

Department of Industrial Engineering

February-2007

DAİRESEL KESME PROBLEMLERİNİN ÇÖZÜMÜNDE
GENETİK ALGORİTMA TABANLI BİR YAKLAŞIM KULLANILMASI

Hulusi SEYDANOĞLU

Eskişehir Osmangazi Üniversitesi
Fen Bilimleri Enstitüsü
Lisansüstü Yönetmeliği Uyarınca
Endüstri Mühendisliği Anabilim Dalı
Endüstri Mühendisliği Bilim Dalında
YÜKSEK LİSANS TEZİ
Olarak Hazırlanmıştır

Danışman: Prof. Dr. A. Attila İŞLİER

Şubat-2007

Hulusi SEYDANOĞLU'nun YÜKSEK LİSANS tezi olarak hazırladığı "Dairesel Kesme Problemlerinin Çözümünde Genetik Algoritma Tabanlı Bir Yaklaşım Kullanılması" başlıklı bu çalışma, jürimizce lisansüstü yönetmeliğinin ilgili maddeleri uyarınca değerlendirilerek kabul edilmiştir.

Üye (Danışman): Prof. Dr. A. Attila İŞLİER

Üye : Prof. Dr. Doğan EROL

Üye : Doç. Dr. Müjgan SAĞIR ÖZDEMİR

Üye : Y. Doç. Dr. Muzaffer KAPANOĞLU

Üye : Y. Doç. Dr. Nil ARAS

Fen Bilimleri Enstitüsü Yönetim Kurulu'nun .... tarih ve .... sayılı kararıyla onaylanmıştır.

Prof. Dr. Abdurrahman KARAMANCIOĞLU

Enstitü Müdürü

ÖZET

Bu çalışmada, dairesel kesme probleminin çözümü amacıyla genetik algoritma tabanlı bir yaklaşım geliştirilmiştir. Bu problemde, farklı yarıçaplardaki bir daireler kümesinin sabit boyutlardaki dikdörtgen bir ana malzemeden kesilmesi gerekmektedir. Problemin karmaşıklığı nedeniyle, konuyla ilgili kaynaklarda çok az çalışma bulunmaktadır ve problemin çözümüne yönelik yaklaşımlar genellikle sezgisel yordamlardır. Amaç, her daire için talep kısıtları göz önünde alınarak, ana malzeme kullanım oranının en büyüklenmesidir. Uyum fonksiyonu, bir çözümde bulunan daireler arasındaki toplam çakışma miktarı olarak belirlenmiştir. Algoritma, önceden belirlenen sayıda dairenin ana malzemeye rastgele yerleştirilmesi ile başlamakta, çakışma olmayan bir çözüm bulununcaya kadar çalıştırılmaktadır. Çakışma olmadan elde edilen en büyük daire sayısına sahip çözüm son çözüm olarak kabul edilmektedir. Algoritmanın etkinliğinin ölçülmesi amacıyla rastgele oluşturulan örnekler üzerinde testler gerçekleştirilmiştir. Ayrıca sonuçlar, kaynaklarda yer alan çözümlerle de karşılaştırılmıştır. Son olarak; geliştirilen algoritma, bir gerçek hayat probleminden alınan veriler kullanılarak test edilmiştir. Elde edilen sonuçlar, geliştirilen yaklaşımın kabul edilebilir sürede iyi çözümler ürettiğini göstermiştir.

Anahtar Kelimeler
Dairesel kesme problemi
Genetik Algoritmalar

SUMMARY

In this work, a genetic algorithm based approach to solve the "circular cutting problem" is developed. In this problem a set of circles with non-identical diameters needs to be cut from a rectangular stock sheet of fixed dimensions. Due to its complexity, there is not much published literature on this subject and the approaches to solve the problem are mainly heuristics. The objective is to maximize the usage of the stock sheet while respecting the upper demand value for each circle type. The fitness function evaluates the amount of overlap among the circles in a solution. Algorithm starts with a solution which consists of a pre-defined number of circles randomly placed inside the rectangle and runs until a solution with non-overlapping circles is found. The solution which consist the maximum number of circles without overlap is considered the final solution. In order to evaluate the algorithms performance some computational tests were performed over a varied set of randomly generated test instances and a comparative study with other methods of the literature is presented. Finally the algorithm is tested on a real test data. Computational results show that the proposed approach produces better results within reasonable computational times.

Key Words:

Circular cutting problem

Genetic Algorithms

TEŞEKKÜR

Çalışmamın her aşamasında yardım ve ilgilerini esirgemeyen, değerli görüşleriyle beni yönlendiren danışman hocam Prof. Dr. A. Attila İŞLİER'e en içten teşekkürlerimi sunarım.

Ayrıca çalışmam sırasında desteğinden dolayı eşime, anlayışından dolayı kızıma teşekkür ederim.

İÇİNDEKİLER

İÇİNDEKİLER (devam)

Sayfa

İÇİNDEKİLER (devam)

ŞEKİLLER DİZİNİ

ÇİZELGELER DİZİNİ

SİMGELER ve KISALTMALAR DİZİNİ

Simgeler	**Açıklama :**
D_i	i. sipariş parçasının talep miktarı
d_{ij}	i. ve j. parçaları arasındaki çakışma miktarı
d_{il}	i. parçanın ana malzemenin kısa kenarıyla olan çakışma miktarı
d_{iw}	i. parçanın ana malzemenin uzun kenarıyla olan çakışma miktarı
F_B	Bileşke kuvvet
f_{ix}	i. parçanın x koordinatındaki eksenel kuvveti
f_{iy}	i. parçanın y koordinatındaki eksenel kuvveti
$F_{sağ}$	Ana malzemenin kısa kenarının itme kuvveti
L_0	Ana malzemenin boyu
L_i	i. dikdörtgenin boyu
MAU_{ij}	i. ve j. parçalarının merkezleri arasındaki uzaklık
n	Küçük parçaların sayısı
S_0	Dikdörtgen ana malzeme
r_0	Dairesel ana malzemenin yarıçapı
r_i	i. parçanın yarıçapı
W_0	Ana malzemenin eni
W_i	i. dikdörtgenin eni
U(K)	Uyum fonksiyonu
K	Kromozom
x_i	i. parçanın x koordinatı
y_i	i. parçanın y koordinatı

Kısaltmalar

CX	Cycle Crossover
ÇM	**Ç**akışma **M**esafesi
GA	**G**enetik **A**lgoritma
G	**G**rup

SİMGELER ve KISALTMALAR DİZİNİ (devam)

Kısaltmalar	**Açıklama**
MAU	**M**erkezler **A**rası **U**zaklık
MaxNSA	**Max**imizing the **N**umber of circles with **S**imulated **A**nnealing
MinOSA	**Min**imizing **O**verlap with **S**imulated **A**nnealing
MKP	**M**alzeme **K**esme **P**roblemi
NP	**N**on-**P**olynomial
OBX	**O**rder-**B**ased **C**rossover
OX	**O**rder **C**rossover
PBX	**P**osition-**B**ased **C**rossover
PMX	**P**artial **M**atching **C**rossover
2-opt	İki parçanın yer değişimi ile elde edilen en iyi
3-opt	Üç parçanın yer değişimi ile elde edilen en iyi
n-opt	N parçanın yer değişimi ile elde edilen en iyi

FORMÜLLER DİZİNİ

Açıklama :	Formül :	
Dikdörtgen kesme modelinde çakışmama kısıtı	$enb(\lvert x_{ip}-x_{jq}\rvert-\alpha_{ij},\lvert y_{ip}-y_{jq}\rvert-\beta_{ij})z_{ip}z_{jq}\geq 0$	(1)
i. parçanın ilk P_i kopyasının S_0'dan kesilme kısıtı	$z_{ip}=1$	(2)
Dikdörtgen kesme modelinde ana malzeme içinde kalma kısıtı	$L_i/2\leq x_{ip}\leq L_0-L_i/2$	(3)
Dikdörtgen kesme modelinde ana malzeme içinde kalma kısıtı	$W_i/2\leq y_{ip}\leq W_0-W_i/2$	(4)
Dikdörtgen kesme modelinde tamsayı kısıtı	$z_{ip}\in(0,1)$	(5)
Dikdörtgen kesme modelinde amaç fonksiyonu	Enb $\sum_{i=1}^{n}\sum_{p=1}^{Q_i}v_i z_{ip}$	(6)
Dairesel kesme modelinde (P1) ana malzeme içinde kalma kısıtı	$r-\frac{L_0}{2}\leq x_i\leq\frac{L_0}{2}-r$	(7)
Dairesel kesme modelinde (P1) ana malzeme içinde kalma kısıtı	$r-\frac{W_0}{2}\leq y_i\leq\frac{W_0}{2}-r$	(8)
Dairesel kesme modellerinde (P1 ve P2) amaç fonksiyonu	Enk $\sum_{i<j}$ Enb$\left(0,(2r\ -MAU_{ij})\right)$	(9;11)
Dairesel kesme modelinde (P2) ana malzeme içinde kalma kısıtı	$MAU_{i0}\leq(r_0-r)$	(10)
Dairesel kesme modelinde (P3) ana malzeme içinde kalma kısıtı	$r_i-\frac{L_0}{2}\leq x_i\leq\frac{L_0}{2}-r_i$	(12)
Dairesel kesme modelinde (P3) ana malzeme içinde kalma kısıtı	$r_i-\frac{W_0}{2}\leq y_i\leq\frac{W_0}{2}-r_i$	(13)
Dairesel kesme modellerinde (P3 ve P4) amaç fonksiyonu	Enk $\sum_{i<j}$ Enb$\left(0,(r_i+r_j-MAU_{ij})\right)$	(14;16)
Dairesel kesme modelinde (P4) ana malzeme içinde kalma kısıtı	$MAU_{i0}\leq(r_0-r_i)$	(15)
Uyum fonksiyonu	U(K) = $\sum_{i=1}^{n}((\sum_{j=i+1}^{n}d_{ij})+d_{il}+d_{iw})$	(17)

FORMÜLLER DİZİNİ (devam)

Açıklama :	Formül :	
i. ve j. parçaları arasındaki çakışma mesafesi	$d_{ij} = \begin{Bmatrix} MAU_{ij} < r_i + r_j \quad ise \quad r_i + r_j - MAU_{ij} \\ aksi\ halde \quad 0 \end{Bmatrix}$	(18)
i. parçanın ana malzemenin kısa kenarıyla olan çakışma mesafesi	$d_{il} = \begin{Bmatrix} \lvert x_i \rvert + r_i > \frac{L_0}{2} \quad ise, \quad \lvert x_i \rvert + r_i - \frac{L_0}{2} \\ Aksi \quad halde, \quad 0 \end{Bmatrix}$	(19)
i. parçanın ana malzemenin uzun kenarıyla olan çakışma mesafesi	$d_{iw} = \begin{Bmatrix} \lvert y_i \rvert + r_i > \frac{W_0}{2} \quad ise, \quad \lvert y_i \rvert + r_i - \frac{W_0}{2}, \\ Aksi \quad halde, \quad 0 \end{Bmatrix}$	(20)
i. ve j. parçaların merkezleri arasındaki uzaklık	$MAU_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$	(21)
ABC ve DEF üçgenlerinin benzerliği ile elde edilen denklem	$\frac{f_x}{x_i - x_j} = \frac{f_y}{y_i - y_j} = \frac{d_{ij}}{MAU_{ij}}$	(22)
Parçalar arasındaki itme kuvvetinin x eksenindeki bileşeni (f_x)	$f_x = \frac{d_{ij}(x_i - x_j)}{MAU_{ij}}$	(23)
Parçalar arasındaki itme kuvvetinin y eksenindeki bileşeni (f_y)	$f_y = \frac{d_{ij}(y_i - y_j)}{MAU_{ij}}$	(24)
Dinamik mutasyon oranı hesaplama formülü	Tam_değer [$\frac{\lvert r_i - r_j \rvert}{r_i + r_j}$ x 30]	(25)

1. GİRİŞ

Ara mamul veya son ürünün elde edilmesi amacıyla ham malzemelerin kesilmesi sürecini içeren her üretim türünde, ham malzemenin en verimli şekilde kullanılması problemi ile karşılaşılmaktadır. Bir yandan malzeme verimliliğini artırmak isteyen sanayi kuruluşlarının talebi; diğer yandan da fire en küçükleme probleminin kuramsal açıdan çekiciliği, bu alanda çok zengin bir kaynak birikimine yol açmıştır. Kesme problemlerini kısa sürede tam olarak çözen bir algoritmanın henüz geliştirilememiş olması, araştırmacıların bu alana olan ilgisini canlı tutmaktadır.

Malzeme kesme problemi, "Belirli boyutlarda temin edilebilen bir malzemeden, daha küçük boyuttaki parçaların kesilmesi problemi" olarak tanımlanabilir. Malzeme kesme problemlerinin en basit şekli olan tek-boyutlu problem bile NP-zor sınıfına girmektedir (Lai ve Chan, 1997). Uygulamada ise, bu problemin fire kaybı (trim loss) ve palet yükleme (pallet loading) gibi daha karmaşık türleriyle daha sık karşılaşılmaktadır. Problemin kuramsal açıdan ilgi çekmesinin nedeni de uygulamanın yaygınlığı yanında kesme-istifleme (cutting and packing) olarak adlandırılan bir dizi problemin prototipini oluşturması ve çözüm güçlüğü olmaktadır.

Değişik türdeki iki-boyutlu kesme problemleri kaynaklarda geniş bir biçimde yer almaktadır. Bu tür problemler; büyük bir dikdörtgen ana malzemeden küçük parçaların kesilmesi problemi olarak düşünülebildiği gibi; küçük parçaların büyük bir dikdörtgen ana malzemeye yerleştirilmesi problemi olarak da değerlendirilebileceği için, daha genel bir isimle kesme-istifleme problemleri olarak da adlandırılmaktadır (Beasley, 2004).

Görünüşte basit olan bu problem tam olarak çözülebilmiş değildir. Örneğin 67 [cm] uzunluğunda ve 44 [cm] genişliğindeki dikdörtgen bir ana malzemeden 6 [cm] uzunluğunda ve 5 [cm] genişliğindeki küçük dikdörtgen parçaların en büyük sayıda kesilmesi probleminin tamsayılı doğrusal programlama yazılımı ile çözümü 105 saatten fazla sürmektedir. Ayrıca, (67x44)/(6x5) = 98 [adet] olmasına karşın çözüm değeri olarak 97 sayısı elde edilmektedir (Lins vd., 2003).

İki-boyutlu kesme problemlerinin bir alt problemi olan dairesel kesme probleminde ise, küçük parçalar aynı veya farklı boyuttaki dairesel şekilli malzemelerden oluşmaktadır. Bu problem, mutfak gereçleri üretimi, tekstil, havacılık sanayi, büro mobilyalarının üretimi, dairesel tabanlı aynı yükseklikteki malzemelerin paletlere yerleştirilmesi gibi birçok endüstri dalında ortaya çıkmaktadır.

Farklı yarıçaplardaki dairesel şekilli malzemelerin bir ana malzemeye yerleşimi problemi çok zor görünmemekte ise de; problemin çözümü ilk bakışta göründüğü kadar kolay değildir. Asıl problem ana malzemenin kesilecek parçalara göre küçük olduğu durumda ortaya çıkmaktadır. Bu durumda en iyi yerleşim için, ana malzeme üzerinde bulunan dairesel malzemelerin konumlarında çok sayıda değişikliklerin denenmesi gerekmektedir. Sonuç olarak çok sayıda farklı kesme planları ortaya çıkmakta, çözüm uzayının büyümesi de çözümü zorlaştırmaktadır (Wang vd., 2002).

Meta sezgisel teknikler, çözümü zor kombinatoryal en iyileme problemlerinin yaklaşık (approximate) çözümü için geliştirilen ve son yıllarda yaygınlaşan araçlardır (Lodi vd., 2002a). Matematiksel programlamaya dayanan ve en iyi çözümü veren algoritmalar, hesaplama maliyetinin çok yüksek olması yüzünden pratik değillerdir. Bu yöntemler, sadece küçük boyutlu problemleri çözebilirler. Uygulamada karşılaşılan problemlerin çözümünde ise genellikle en iyiye yakın ya da iyice çözümler veren sezgisel algoritmalar kullanılmaktadır. Küçük parça sayısı çok ve küçük_parça_alanı / ana_malzeme_alanı oranı küçük olduğunda sezgisel algoritmalar oldukça iyi sonuçlar vermekte, ancak parça sayısı azalıp oran büyüdükçe çözümler de en iyiden uzaklaşmaktadır (Chauny vd., 1991).

Farklı yarıçaplardaki dairesel kesme problemlerinin çözümü için kaynaklarda çok az sayıda çalışma bulunmaktadır. Bunlardan erişilebilenler kurma esaslı sezgisel kurallar, benzetim, dal-sınır tekniği, genetik algoritma, tavlama benzetimi, indirgenmiş eğim (reduced gradient) yöntemlerine dayanan çözüm yolları olarak sıralanabilir (Wang vd., 2002), (Stoyan ve Yaskov, 1998), (WenQi vd., 2001), (Hifi vd., 2004), (Hifi ve M'Hallah, 2004).

Erişilebilen çalışmalardan sadece biri genetik algoritmalarla ilgilidir. Bu yaklaşımda, genetik algoritma kullanılarak parçalar için iyi bir sıralama araştırılmakta, küçük parçalar bu sırayla ana malzemeye yerleştirilip en iyi yerleşim elde edilmeye çalışılmaktadır.

Çaprazlama operatörü bu tür problemlerde çok ciddi tahribatlara sebep olmaktadır. Bu nedenle, bu tez çalışmasında geliştirilen, genetik algoritma tabanlı yeni yaklaşımda çaprazlama operatörü kullanılmamıştır. Algoritma, mutasyon ağırlıklı çalışması yönüyle kaynaklarda yer alan yaklaşımdan farklılık göstermektedir. Mutasyon işlemi, DEĞİŞTİR modülü ile -rastgele seçilen iki çemberin yer değiştirmesi- yapılmakta; bunu izleyen tamir işlemleri KAYDIR modülü ile -çakışan çemberlerin birbirlerini iterek daha iyi yerleşim sağlaması- gerçekleştirilmektedir. KAYDIR modülü, eldeki çözümleri -az da olsa- değişikliğe uğrattığı için, çözüm uzayının farklı bölgelerine atlama imkanı da sağlamaktadır. Geliştirilen yaklaşım Hifi ve M'Hallah'ın (2004) test sonuçları ile karşılaştırılmış ve daha iyi çözümler elde edildiği görülmüştür.

Çalışmanın ikinci bölümünde, malzeme kesme problemi ile ilgili genel kavram ve tanımlar açıklanmaktadır. Kısıtlar ve amaçlar yönüyle problemin yapısı tanıtılıp, genel kesme problemlerinin çeşitleri, kaynaklarda yer alan sınıflandırma yöntemleri ve geliştirilen çözüm yaklaşımlarına yer verilmektedir.

Üçüncü bölümde, iki boyutlu dairesel kesme problemlerinin yapısı, modellenmesi ve çözümüne yönelik olarak bugüne kadar yapılan çalışmalardan ulaşılabilenler genel hatlarıyla ele alınıp tartışılmaktadır.

Dördüncü bölümde, genetik algoritmalarla ilgili temel tanım ve kavramlar hakkında açıklamalarda bulunduktan sonra, dairesel kesme probleminin çözümü için geliştirilen yeni yaklaşıma yer verilmektedir. Problemin nasıl modellendiği, kullanılan algoritmanın yapısı ve karşılaşılan özel durumlar için geliştirilen çözüm metotları bu bölümde irdelenmektedir. Dördüncü bölümün sonunda, geliştirilen yaklaşımın sınanması, önce rastgele türetilmiş verilerle yapılmış, çözümün farklı durumlardaki davranış biçimi irdelendikten sonra, kaynaklardaki problemlerle karşılaştırılıp yeni yöntemin kuramsal olarak kıyaslanması yapılmış, en sonunda da gerçek hayat problemleri ele alınarak işyerinde kabul gören çözümlerle yapılan karşılaştırma sonucunda pratik açıdan bir değerlendirme yapılmaya çalışılmıştır.

Çalışmanın sonuç ve öneriler bölümünde, geliştirilen yaklaşımın genel bir değerlendirilmesi yapılarak, ileride ele alınabilecek çalışmalara yer verilmektedir.

2. MALZEME KESME PROBLEMLERİ

Endüstride kullanılan ham malzemeler genel olarak belirli standart büyüklüklerde temin edilebilmektedir. Bu malzemelerin daha sonra belirli boyutlarda kesilerek üretim sürecine alınması gerekmektedir. Ham malzemenin en verimli biçimde nasıl kullanılacağının belirlenmesine yönelik araştırmalar kaynaklarda genel malzeme kesme problemleri olarak adlandırılmaktadır.

Bu bölümde malzeme kesme problemleri ile ilgili genel tanım ve kavramlar açıklanmaktadır. Malzeme kesme problemlerinde kaynaklarda büyük bir yer tutan dikdörtgen parçaların kesimi tanıtılıp, genel kesme problemleri ile ilgili; boyut kavramının irdelenmesi, kesme problemlerinin sınıflandırılması, kısıtlar ve amaçlar, kesme problemlerinin modellenmesi, geliştirilen çözüm yaklaşımları ve dairesel kesme problemlerinin genel tanımları üzerinde durulmaktadır.

2.1. Malzeme Kesme Problemlerine Genel Bakış

Tekstil, deri, kağıt, cam, kereste ve metal gibi birçok endüstri dalında sık bir biçimde karşılaşılan ***malzeme kesme problemi***, "*Belirli boyutlarda temin edilebilen bir malzemeden, daha küçük boyuttaki parçaların kesilmesi problemi*" olarak tanımlanabilir.

En genel adıyla malzeme kesme ***(cutting)*** ve istif ***(packing)*** problemlerinde küçük parçaların (örn. dairesel şekilli kutu kapakları) büyük nesneler (örn. sac levha) içinde geometrik olarak yerleşimini gösteren planların belirlenmesi amaçlanmaktadır. Bunun bir alt kümesi olan istif problemlerinde büyük nesneler (örn. konteyner, kutu) boş alan olarak tanımlanmakta ve küçük parçalar (örn. koliler) kullanılarak bunların doldurulması gerekmektedir. Diğer alt kümeyi oluşturan kesme problemleri ise, küçük parçalara (örn. iki-boyutlu şekiller) kesilmesi gereken büyük nesnelere (örn. metal levha, rulo) dayanır (Hopper ve Turton, 2001a).

Malzeme kesme problemlerinin çok değişik çeşitleri mevcuttur. Bunlara bir örnek küçük parçaların elde edilmesi için değişik büyüklüklerdeki ana malzemelerden hangisinin /hangilerinin belirleneceği (assortment) problemidir. Başka bir örnek ise dik açılı olmayan (non-orthogonal) kesme planlarının kullanılması, başka bir ifadeyle sipariş parçalarının levha üzerine açılı olarak yerleştirilmesidir. Ayrıca dikdörtgen olmayan şekle sahip parçaların düzenlenmesi problemleriyle de çok sık karşılaşılmaktadır (Dietrich ve Yakowitz, 1991).

Kesilecek malzemeye, ***ana malzeme***, ana malzemeden kesilen küçük parçalara ise ***sipariş parçası*** denilmektedir (Saraç, 2001). ***Kesme planları*** (Pattern), kesme işleminin nasıl yapılacağını, bir başka deyişle parçaların ana malzemeye ne şekilde yerleştirileceğini gösteren geometrik modellerdir (İşlier, 1993). Sipariş listesindeki parçaların tümünün üretilmesini sağlayacak şekilde birden fazla kesme planı hazırlanabilir ve bu planlar kendilerini defalarca tekrarlayabilir. Chauny vd. (1991), tek-boyutlu kesme planını, boylarının toplamı ana malzemenin boyunu aşmayan parçaların bir kombinasyonu olarak tanımlamıştır.

Kesme planına benzer biçimde, bir ***yerleştirme planı***, küçük parçaların ana malzemeye yerleştirileceği sırayı gösteren bir permütasyonla temsil edilebilmektedir (Leung vd., 2003). Bunun için de problemin ya tek-boyutlu olması veya giyotin kesimine karşı gelen ağaç yapısı gibi bir sistematiğin kullanılması gerekmektedir.

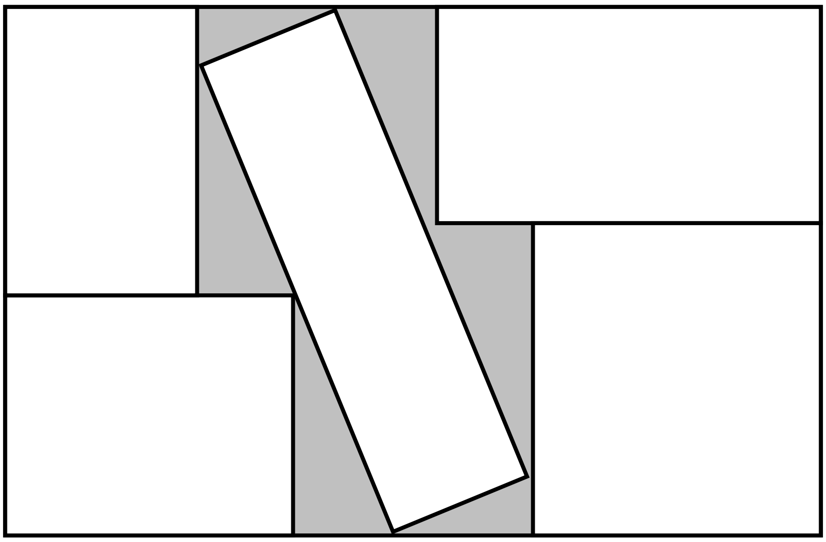
Sipariş listesinde bulunmadığı halde kesme planında yer alan artık malzemelere ***fire*** denilmektedir. Çoğu malzeme kesme probleminde amaç fonksiyonu toplam fire miktarının en küçüklenmesi olarak tanımlanmaktadır.

2.1.1. Dikdörtgen parçalara özgü durumlar

Malzeme kesme problemlerinde dikdörtgen şekilli parçaların kesimi hem kesme teknolojisi, hem de kesme planlarının hazırlanması açılarından özel bir yer tutmaktadır. Dikdörtgenin dört kenarı da doğru olduğu için bunların kesimi kalıp, şablon vb. özel donanım gerektirmeden, daha basit ve ucuz tezgahlarda yapılabilmektedir. Yerleşim planı tasarımında da dışbükey yapı ve düzgün kenarlar, yerleşimi kolaylaştırmakta, düşük fireli planların kolayca düzenlenebilmesini sağlamaktadır.

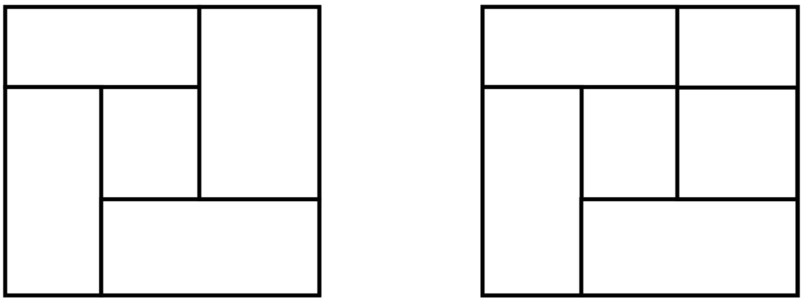
Dikdörtgen parçaların kesme planlarının oluşturulmasında sıklıkla karşılaşılan dik açılılık (orthogonality), giyotinle kesim, kademeli kesme, döndürme ve boyut kavramları sırasıyla aşağıda açıklanmaktadır.

Dik açılılık (orthogonality): Dik açılı (orthogonal) malzeme kesme-istifleme problemlerinde dikdörtgen şeklindeki parçalar kümesi, dikdörtgen ana malzeme üzerine kenarları sırasıyla x ve y eksenlerine paralel olacak şekilde yerleştirilmektedir (Wu vd., 2002). Dik açılı olmayan kesme (Şekil 2.1), bazı özel durumlarda kayıpları azaltabilse bile kesme işleminde kullanılan tezgahlar genelde dik açılı olmayan (non-orthogonal) kesime izin vermemektedir.



Şekil 2.1. Dik açılı olmayan (non-orthogonal) kesme

Giyotinle Kesim: Gerçek veya kavramsal bir giyotin tezgahında her ardışık kesim (birbirini izleyen her kesim) levhanın bir ucundan diğerine uzayacak şekilde yapılmalı, dikdörtgen parçalar bu şekilde kesilmelidir (Şekil 2.2). Sac levhaları uzun bir bıçağın tek bir düşey hareketiyle kesen gerçek giyotin tezgahı da, kesme masasında cam levhaları elmasla çizip, çizginin zayıflattığı yerden kırarak ikiye ayıran kavramsal bir giyotin de, malzemeyi boydan boya kesmektedir.

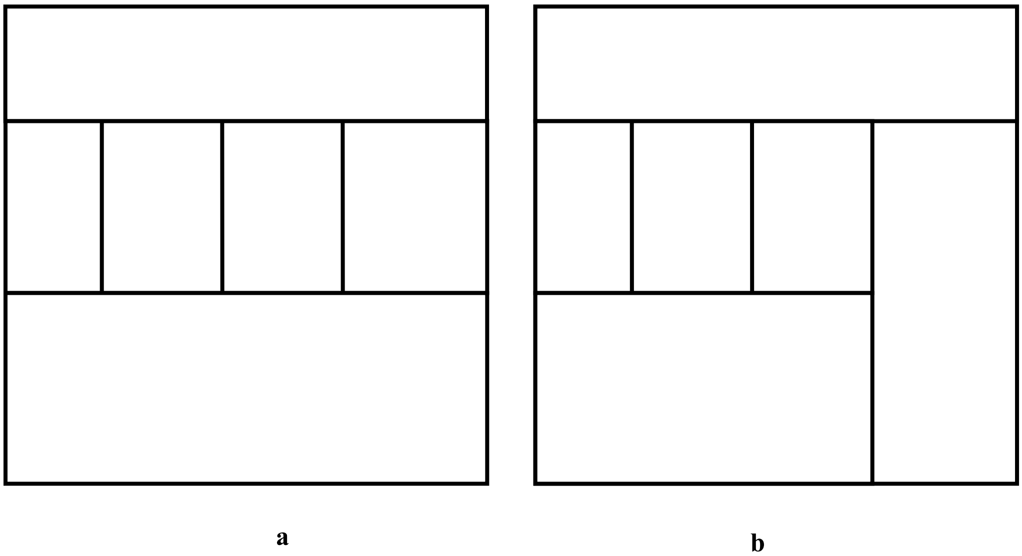


a **b**

Şekil 2.2. Kesme planları: a) giyotin kısıtsız, b) giyotin kısıtlı

Kademeli Kesim: İki-boyutlu malzeme kesme problemlerinde kademeli kesme işleminin modellenmesi Gilmore ve Gomory'nin (1965) çalışmalarıyla başlamıştır (Kenyon ve Remila, 2000). Bu kesme işlemi giyotinle kesmenin özel bir durumudur.

İlk aşamada levha, birkaç alt parçaya ayrılır. İkinci aşamada, bu alt parçalar da zıt yönde kesilip daha alt düzeydeki parçalar elde edilir (Şekil 2.3). Kontrplak kesiminde kullanılan belirli tipteki makinelerde bu tarz kesim yapılır.



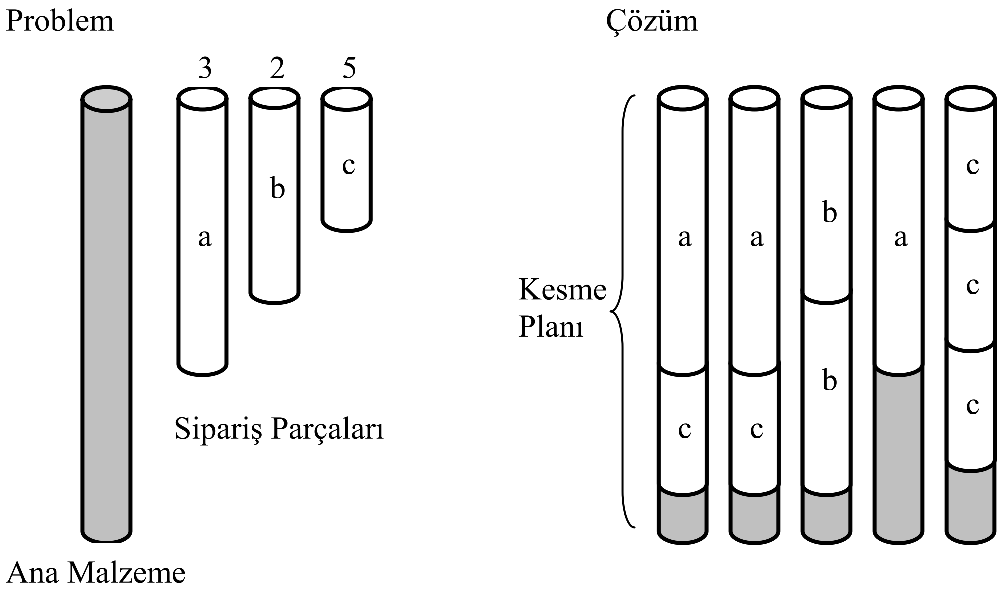
Şekil 2.3. Kesme planları: a) iki aşamalı, b) dört aşamalı

Döndürme: Bir parçanın kesiminde yön önemliyse (örneğin ahşap malzemede liflerin yönü, kumaşta desen), parçanın döndürülmesine izin verilmeyebilir. Döndürmeye izin verildiğinde, alt parça doksan derece döndürülür ve bunun diğer kenarı, tezgahın tahdit çıkıntılarına dayanarak yerleştirilir. İzleyen kesme bu döndürme işleminden sonra yapılır.

2.1.2. Kesme probleminde boyut durumu

Malzeme kesiminde, ana malzemenin blok, rulo, levha veya çubuk olmasına bağlı olarak tek, birbuçuk, iki veya üç-boyutlu kesmeden söz edilebilmektedir.

Tek-boyutlu kesme: Demir çubuklar ve kağıt rulolara uygulanan bu çeşit kesimde, ikinci ve üçüncü boyutların anlamı kalmamaktadır (Şekil 2.4). Gilmore ve Gomory (1961, 1963) kağıt rulolarda tek-boyutlu malzeme kesme problemini ele almış ve bir malzeme kesme problemine en iyi sonucu veren ilk yöntemi geliştirmiştir (Chauny vd., 1991).



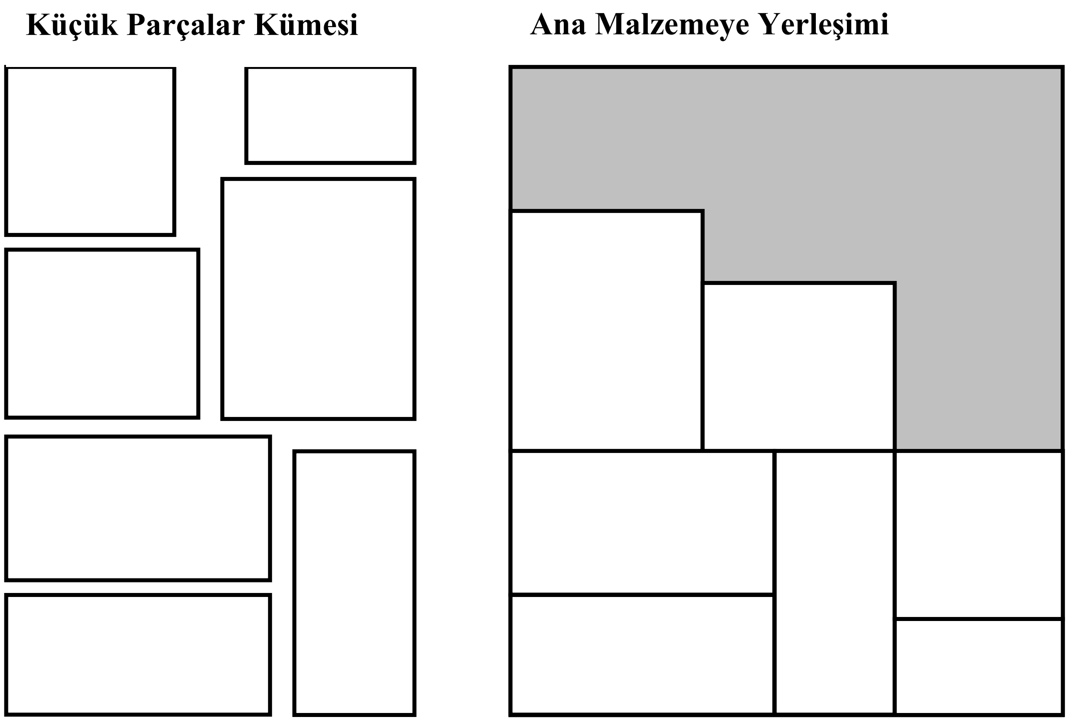
Şekil 2.4. Tek-boyutlu malzeme kesme problemi

Birbuçuk-boyutlu kesme: Bu kesme, levha uzunluğunun -pratik olarak- sonsuz varsayıldığı iki-boyutlu problemin özel bir durumudur. Dikdörtgen parçalar çok uzun bir rulo malzeme üzerine yerleştirileceği zaman, bu çeşit problem ortaya çıkar.

İki-boyutlu kesme: İki-boyutlu malzeme kesme problemi (Şekil 2.5), sac, cam, kontrplak, kumaş ve deri gibi levhalardan kesilecek parçaların bu levhalar üzerine, kullanılan levha sayısını -veya toplam fireyi- en küçükleyecek şekilde dizilmesini ele alır.

Birbuçuk ve iki boyutta düzensiz parçalar: Tek-boyutlu kesme probleminde, kesilecek malzemenin diğer boyutları ve bu boyutlardaki şeklinin (profil) kesme planlarının türetilmesine bir etkisi bulunmamaktadır.

Ancak birbuçuk ve iki boyutta, düzenli-düzensiz şekil ayırımı önem kazanmaktadır. Düzenli şekil kapsamında genelde dikdörtgen kastedilse de, düzgün altıgen ve daire gibi merkeze göre simetrik parçalar da bu grupta ele alınabilir.

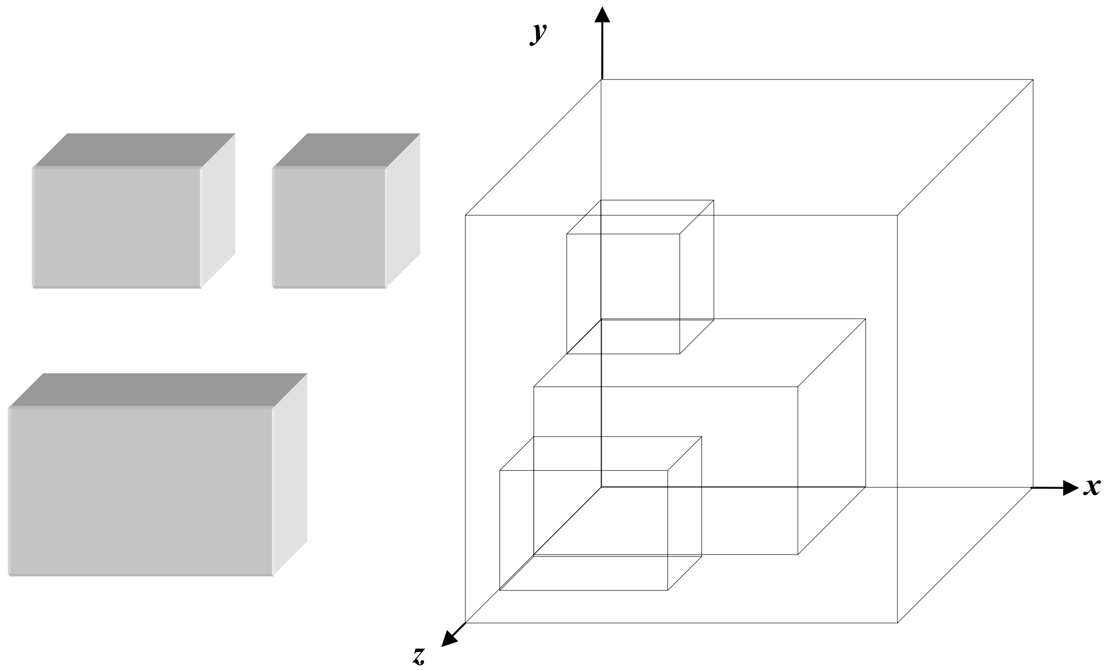


Şekil 2.5. İki-boyutlu malzeme kesme problemi

Üç-boyutlu Problemler: Üç-boyutlu malzeme yerleştirme problemlerinde (Şekil 2.6) aynı/farklı biçim ve ölçülerdeki n adet kutunun sabit bir biçim ve hacme sahip olan bir konteynere toplam hacmi en küçükleyecek şekilde yerleştirilmesi amaçlanmaktadır (Li vd., 2003).

Gerçek hayat uygulamalarında, üç-boyutlu istif problemlerinin birçoğu iki-boyutluya indirgenebilir. Örneğin, süt kapları (şişeleri) sadece şişe dipleri aşağıda kalacak şekilde yerleştirilebilir (Daniels ve Ghandforoush, 1990).

Üç-boyutlu istif problemlerinde düzensiz şekillerle karşılaşılsa bile, üç-boyutlu kesme problemini konu edinen kaynaklarda düzensiz parçaların kesimine yönelik çalışmalarla karşılaşılmamıştır. Zaten kesme teknolojisi açısından ana malzemeden, önce biçimsiz şekilleri kapsayan prizmatik zarfların kesilmesi, elde edilecek parçaların şekillendirilmesinin ise bu prizmatik parçalara bir çeşit oyma işlemi (carving) uygulanmasıyla mümkün olacaktır.



Şekil 2.6. Üç-boyutlu malzeme kesme problemi

2.2. Kesme Problemlerinin Sınıflandırılması

Gerçek hayat şartlarında karşılaşılan özel durumların bir sonucu olarak kesme problemlerinin çeşitli türleri ortaya çıkmıştır. Benzer mantıksal yapıdaki bu malzeme kesme ve istif problemleri kaynaklarda -aşağıda sıralanan- farklı adlarla yer alabilmektedir (Faina, 1999).

- Malzeme kesme (cutting stock) ve fire kaybı (trim loss) problemleri,
- Kutu yerleştirme (bin packing), şerit yerleştirme (strip packing), vektör yerleştirme (vector packing) ve sırt çantası (knapsack) problemleri,
- Araç yükleme (vehicles loading), palet yükleme (pallet loading) ve konteyner yükleme (container loading) problemleri,
- Ana malzeme seçimi (assortment), yuvalama (nesting) ve bölümlendirme (partitioning) problemleri,

- Sermaye bütçelemesi (capital budgeting) ve para üstü verme (change making) problemleri,
- Hafıza tahsisi (memory allocation) ve çok-işlemci çizelgeleme (multiprocessor scheduling) problemleri.

Değişik türdeki iki-boyutlu kesme problemleri kaynaklarda geniş bir yer tutmaktadır. Bu tür problemler; büyük bir dikdörtgen ana malzemeden küçük parçaların kesilmesi problemi olarak, veya küçük parçaların büyük bir dikdörtgen ana malzemeye yerleştirilmesi problemi olarak değerlendirilebileceği için yerleştirme problemleri olarak da adlandırılmaktadır (Beasley, 2004).

Dyckhoff (1990), malzeme ve boşluk arasındaki ikil ilişkiden (duality) yola çıkarak, kesme ve yerleştirme problemleri arasındaki güçlü bağa dikkat çekmiştir. Bu bakış açısıyla, yerleştirme problemlerinin, küçük parçaların büyük nesneye yerleştirilmesi olarak düşünülebileceğini; tersi durumda ise malzeme kesme problemlerinin büyük nesnelerin daha küçük nesnelere bölünmesi olarak tanımlanabileceğini ifade etmiştir (Hopper ve Turton, 2001a).

Birçok kesme probleminde sadece ***giyotinle kesime*** izin verilmektedir. Giyotinle kesme kısıtı malzemenin teknolojik ihtiyaçlar nedeniyle bir baştan diğerine kesilmesi gerektiği durumlarda (giyotin makasta kesmek gibi) ortaya çıkmaktadır. Kavramsal bir giyotin kısıtı yerleştirme problemlerinde de söz konusu olabilmektedir (Tam, 1991).

Malzeme kesme problemlerinin karmaşıklığı, kesilecek küçük parçaların geometrik özellikleri ile yakından ilişkilidir. Malzeme kesme problemleri bu yönüyle düzenli ve düzensiz şekilli malzeme kesme problemleri olarak ikiye ayrılmaktadır. Dikdörtgen ve daire gibi düzenli şekilli kesme problemlerinde küçük boyuttaki malzemeler bir veya iki parametreyle (yarıçap, en, boy vb.) tanımlanabilirken, düzensiz şekilli kesme problemlerinde -özellikle asimetrik veya içbükey malzemelerde- bu tanımlamanın yapılabilmesi çok daha güç olmaktadır.

2.2.1. Boyutlarına göre sınıflandırılma

Malzeme kesme problemlerinin en önemli ayırıcı özelliği boyutudur. Ana malzeme üzerinde kesme işlemi uygulanan doğrultu sayısı, kesme probleminin boyutunu ifade etmektedir (Erol, 1990). Kesme probleminin ağırlık veya zaman gibi boyutlar içermesi durumunda problem üçten daha çok boyut kazanabilmektedir. Malzeme kesme problemleri boyutlarına göre tek-boyutlu, birbuçuk-boyutlu, iki-boyutlu ve üç-boyutlu malzeme kesme problemleri olarak sınıflandırılmaktadır. Gerçek hayat problemlerinden bazı örnekler aşağıda verilmektedir:

Tek-boyutlu:

- Klasik sırt çantası,
- Demir çubukların kesilmesi,
- Montaj hattı dengeleme,
- Bellek tahsisi.

Birbuçuk-boyutlu:

- Buzdolabı saclarının kesilmesi,
- Konfeksiyon sanayinde kumaşların kesilmesi.

İki-boyutlu:

- Gemi yapımında çelik levhaların yerleştirilmesi,
- Mobilya yapımında kerestelerin kesilmesi,
- Makale, haber ve reklamların gazete sayfalarına yerleştirilmesi,
- Kutu yapımında mukavva ve kartonların kesilmesi.

Üç-boyutlu:

- Konteyner yükleme,
- Ambalaj kutularına yerleştirme yapılması.

2.2.2. Talep türüne göre sınıflandırma

Her parçaya ait talep, o parça için kesilecek toplam miktarla (talep miktarı) sınırlandırılmakta ise ***kısıtlı (constrained)*** tersi durum için ise problem ***kısıtsız (unconstrained)*** malzeme kesme problemi olarak adlandırılmaktadır. Kısıtlı malzeme kesme problemlerinde bazı parçalara ait talep miktarları, o parçanın bir levhadan kesilebilecek toplam parça sayısından küçük olabilmektedir.

2.2.3. Ana malzemenin türüne göre sınıflandırma

İki-boyutlu malzeme kesme problemlerinde ana malzeme tek bir levhadan oluşabileceği gibi, birden çok levha veya uzunluğu sonsuz varsayılan şerit şeklinde de olabilmektedir.

İki-boyutlu ***malzeme kesme (sırt çantası)*** problemleri, verilen dikdörtgen şeklindeki küçük parçaların toplam faydayı en büyükleyecek şekilde sabit boyutlardaki dikdörtgen bir ana malzemeden kesilmesi olarak tanımlanmaktadır (Hifi, 2004).

İki-boyutlu ***şerit yerleştirme (strip packing)*** probleminde; her biri W_i eninde ve L_i boyunda n adet dikdörtgen parçanın, W_0 genişliğinde ve sınırsız uzunlukta bir şeride tüm sipariş parçalarının kullanılan şerit boyunu en küçükleyecek şekilde yerleştirilmesi amaçlanmaktadır (Martello vd., 2003).

İki-boyutlu ***kutu yerleştirme (bin packing)*** probleminde; W_i eninde ve L_i boyunda n adet dikdörtgen parçanın $i \mid i = \{ 1, \ldots, n \}$, W_0 eninde ve L_0 boyunda sınırsız sayıdaki levhalara, parçalar üst üste gelmeyecek ve parça kenarları levha kenarlarına paralel bir biçimde, toplam levha sayısını en küçükleyecek şekilde yerleştirilmesi amaçlanmaktadır. $W_i = W_0$, olduğu özel durum tek-boyutlu kutu yerleştirme problemini temsil etmektedir (Lodi vd, 2002b).

2.2.4. Küçük malzemelerin türüne göre sınıflandırma

Her sipariş parçasına ait faydanın o parçanın alanına eşit olması durumunda problem ***ağırlıksız (unweighted)***, sipariş parçasına ait faydanın o parçanın alanından bağımsız olması durumunda ise problem ***ağırlıklı (weighted)*** olarak sınıflandırılmaktadır.

2.2.5. Kesim işlemine göre sınıflandırma

Bazı durumlarda ana malzemenin boydan boya kesilmesi gerekebilmektedir. Bu yönüyle malzeme kesme problemleri ***giyotin kısıtlı/kısıtsız*** olarak sınıflandırılmaktadır.

İki-boyutlu giyotin kısıtsız kesme probleminin özel bir hali palet yükleme problemidir. Bu problem tüm parçalar aynı büyüklükte olduğunda ortaya çıkar ve ismini, ahşap bir palet üzerine yerleştirilebilecek aynı büyüklükteki kutu sayısının en büyüklenmesi olarak tanımlanan pratik problemden almaktadır (Beasley, 2004).

2.2.6. Dyckhoff'un sınıflandırması

Dyckhoff (1988), kesme-istifleme problemlerini simgesel olarak ifade edebilecek bir gösterim önermektedir. Bu gösterimde şu simgeleri kullanmaktadır:

α : Problemin boyutunu, $\alpha \in \{ 1, 2, 3, N \}$,

B : Mevcut bir ana malzeme veya mevcut herhangi bir kapasite için küçük parçaların seçiminin yapılması gerektiğini,

V : Mevcut küçük parçalar veya istenen parçalar için ana malzeme veya gerekli kapasite büyüklüğü seçiminin yapılması gerektiğini,

β : Problemin atama türünü $\beta \in \{ B, V \}$,

O : Bir büyük parçayı,

I : Boyutsal ve geometrik olarak benzer parçaları,

D : Boyutsal ve geometrik olarak farklı parçaları,

γ : Büyük parçaların türünü, $\gamma \in \{ O, I, D \}$,

C : Şekil olarak benzeyen parçaları,

F : Az sayıdaki şekil ve boyut olarak değişik parçaları, M : Çok sayıdaki şekil ve boyut olarak değişik parçaları,

R : Şekil ve boyut olarak az farklılık gösteren çok sayıda parçaları,

δ : Küçük parçaların miktar ve türünü, $\delta \in \{ C, F, M, R \}$ göstermektedir.

Herhangi bir kesme problemi bu simgeleri kullanarak, $\alpha/\beta/\gamma/\delta$ biçimde gösterilebilmektedir. Uygulamada karşılaşılan bazı kesme problemlerinin simgesel gösterimi Çizelge 2.1'de verilmiştir.

Çizelge 2.1. Uygulamada karşılaşılan bazı kesme problemlerinin simgesel gösterimi (Sevük, 1996).

Problem	**Simgesel Gösterim**
Klasik sırt çantası	1 / B / O /
Palet yükleme	2 / B / O / C
Konteyner yükleme	3 / V / I /
Montaj hattı dengeleme	1 / V / I / M
Bellek tahsisi	1 / V / I / M
Çok dönemli sermaye bütçeleme	n / B / O /
Genel malzeme kesme veya kesme kaybı	n / / /

2.3. Kesme Problemlerinin Kısıtları

Malzeme kesme problemleri talep türüne göre kısıtsız ya da kısıtlı olarak adlandırılabilmektedir. İlk türdeki problemlerde sipariş parçaları için talep miktarı sınırlı olmayıp, mümkün olan en büyük sayıda parçanın kesilmesi amaçlanmaktadır. Kısıtlı olan ikinci problem türünde ise sipariş parçalarına ait talep miktarları bellidir ve bu miktarlardan daha fazla kesim işlemine izin verilmemektedir.

Malzeme kesme problemlerinde genellikle iki tür kısıttan söz edilmektedir (Gun vd., 2003):

- Giyotin kısıtı: Her kesme işlemi sonunda iki dikdörtgen parça elde etme gereği (Birçok araştırmada giyotin kısıtsız kesme problemleri yuvalama problemleri olarak adlandırılmaktadır).

- Her parçanın yönünün sabit olması: l uzunluğundaki ve w genişliğindeki bir parçanın w uzunluğunda ve l genişliğindeki parçadan farklı olması ($l \neq$ w) gereği (ana malzemedeki desen, damar veya kusurların etkisi)

Giyotinle kesme kısıtına, cam, çelik ve ahşap gibi birçok endüstri dalında, kesme işleminin teknolojik yönüyle ihtiyaç duyulmaktadır. Parçanın döndürülmesine izin verilmemesi kısıtı ise örneğin parçanın desenli bir yapısı olmasından kaynaklanmaktadır (Valdes vd., 2002). Birçok gerçek hayat probleminde, düz malzemelerin kesilmesinde ve çoğu yerleştirme işleminde, daha iyi yerleşimler elde edilmesi için parçaların döndürülmesine -genellikle 90 derece- izin verilebilir. Ancak, örneğin, parçalar gazeteye yerleştirilecek haberlerse veya parçalar desenli bir ana malzemeden kesilecekse döndürme işlemine izin verilemez. Giyotin kısıtı her zaman karşılaşılan bir kısıt değildir. Giyotinle kesmede, tüm parçaların elde edilmesi için ihtiyaç duyulan (döndürme içeren) aşamaların sayısına bir üst sınır -genellikle iki veya üç- getirilebilmektedir.

2.4. Kesme Problemlerinde Benimsenebilir Amaçlar

Malzeme kesme problemlerinde her ana malzeme ve sipariş parçasına bir değer atanmaktadır. Temel olarak fire miktarının en küçüklenmesi amaçlandığı için bu değer parçanın alanına eşit olarak belirlenmektedir. Ancak uzun süre depoda bekleyen bir ana malzemenin değerinin azaltılması veya acil öncelikli bir parçanın değerinin artırılması durumlarını modele yansıtmak amacıyla bu değerin bir katsayıyla (ağırlık) çarpılması kullanıcılar tarafından talep edilebilmektedir (Vanderbeck, 2001).

Malzeme kesme problemlerinde toplam firenin en küçüklenmesi olan amaç fonksiyonu, kullanılan her parçadan elde edilecek faydanın o parçanın alanıyla orantılı olarak belirlenmesi durumunda bir en büyükleme problemine dönüştürülebilir (Boschetti vd., 2002).

Bazı uygulamalarda, daha sonra kullanılmak üzere, mümkün olduğu kadar büyük bir fire elde etmek amacıyla bir kutudaki kullanılmayan alanların en büyüklenmesi olarak adlandırılan ikinci bir amaç dikkat çekicidir (Lodi vd., 2002b).

2.5. Kesme Problemlerinin Modellenmesi

Endüstride kullanılan ham malzemeler genel olarak belirli standart büyüklüklerde temin edilebilmektedir. Bu malzemelerin çoğu durumda üretim sürecinde kullanılacağı aşamadan önce ihtiyaç duyulan boyutlarda kesilmesi gerekmektedir. Kesim safhasının amacı ise ham malzemenin en verimli biçimde kullanılması olarak tanımlanmaktadır. Gilmore ve Gomory'nin (1961, 1963) tek-boyutlu kesme problemleri üzerindeki yayınları bu alandaki ilk çalışmalar arasında yer almaktadır. Aynı yazarlar, 1965 yılındaki çalışmalarını iki-boyutlu kesme problemlerinin çözümüne yönelik olarak ilerletmiştir. Malzeme kesme problemleri ile ilgili çalışmaların bu tarihten itibaren hızla yaygınlaşmasına karşın problemin karmaşıklığı nedeniyle bu türdeki problemleri çözen her durumda geçerli bir çözüm henüz bulunamamıştır (Gomez ve Fuente, 2000).

Küçük dikdörtgen parçaların dikdörtgen bir ana malzemeden kesilmesi problemi için Beasley (2004) tarafından önerilen matematiksel model aşağıda yer almaktadır:

S_0 : Dikdörtgen ana malzeme,
L_0 : Dikdörtgen ana malzemenin boyu,
W_0 : Dikdörtgen ana malzemenin eni,
L_i : i. dikdörtgenin boyu,
W_i : i. dikdörtgenin eni,
n : Sipariş parçalarının sayısı,
v_i : i. dikdörtgenin fayda değeri,

P_i : i. dikdörtgenin S_0 'dan kesilebilecek en küçük sayısı,

Q_i : i. dikdörtgenin S_0 'dan kesilebilecek en büyük sayısı,

z_{ip} : Eğer i. dikdörtgenin p. kopyası (p = 1,…,Q_i) S_0'dan kesilirse 1'e, aksi durumda sıfıra eşittir.

x_{ip} : i. dikdörtgenin p. kopyasının merkezinin x koordinatı,

y_{ip} : i. dikdörtgenin p. kopyasının merkezinin y koordinatı,

α_{ij} : i. ve j. dikdörtgenlerin boyları toplamının yarısı ($\alpha_{ij} = \dfrac{L_i + L_j}{2}$),

β_{ij} : i. ve j. dikdörtgenlerin enleri toplamının yarısı ($\beta_{ij} = \dfrac{W_i + W_j}{2}$),

Olmak üzere karar modeli:

$$enb(|x_{ip} - x_{jq}| - \alpha_{ij}, |y_{ip} - y_{jq}| - \beta_{ij}) z_{ip} z_{jq} \geq 0, \quad i, j = 1,\ldots,n;\ p, q = 1,\ldots, Q_i \quad (i \neq j \text{ veya } p \neq q), \qquad \textbf{(1)}$$

$$z_{ip} = 1, \qquad i = 1,\ldots,n;\ P_i > 0;\ p = 1,\ldots, P_i, \qquad \textbf{(2)}$$

$$L_i / 2 \leq x_{ip} \leq L_0 - L_i / 2, \qquad i = 1,\ldots,n;\ p = 1,\ldots, Q_i, \qquad \textbf{(3)}$$

$$W_i / 2 \leq y_{ip} \leq W_0 - W_i / 2, \qquad i = 1,\ldots,n;\ p = 1,\ldots, Q_i, \qquad \textbf{(4)}$$

$$z_{ip} \in (0,1), \qquad i = 1,\ldots,n;\ p = 1,\ldots, Q_i, \qquad \textbf{(5)}$$

k.a.

$$\text{Enb} \sum_{i=1}^{n} \sum_{p=1}^{Q_i} v_i z_{ip} \qquad \textbf{(6)}$$

şeklinde belirtilmektedir. Amaç fonksiyonu toplam faydayı en büyüklemektir.

Kısıt 1, S_0'dan kesilmek üzere seçilen herhangi iki parçanın birbirleriyle çakışmamalarını sağlamaktadır.

Kısıt 2, i. parçanın ilk P_i kopyasının S_0'dan kesilmek üzere keyfi olarak seçildiğini göstermektedir.

Kısıt 3 ve 4, herhangi bir i parçasının p. kopyasının, ana malzeme sınırlarını içinde kalmasını sağlamaktadır.

Son olarak kısıt 5, modelde yer alan z_{ip} değişkenine sadece 0 veya 1 değeri atanabileceğini göstermektedir.

2.6. Geleneksel Kesme Problemlerindeki Çözüm Yöntemleri

Çelik, kağıt, cam ve ahşap gibi birçok endüstri alanında uygulanabilirliği nedeniyle malzeme kesme problemleri ile ilgili son yirmi yılda birçok araştırma yapılmıştır. Konuyla ilgili çalışmalara yönelik değerlendirmeler ve problemin değişik türlerine ilişkin sınıflandırmalar, Hinxman ve Dyckhoff'un çalışmalarında ayrıntılı olarak yer almaktadır (Parada vd., 2000).

Bu problemlerin çözümü amacıyla, matematiksel teknikler (örn. tamsayılı programlama, doğrusal programlama ve dinamik programlama), problem-bağımlı yordamlar ve modern yordamlar (örn. tavlama benzetimi, sinir ağları ve genetik algoritmalar) gibi değişik teknikler kullanılmıştır (Babu ve Babu, 1999).

Doğrusal programlamaya dayanan yöntemler kesme problemleri için en iyi çözümlerin bulunmasında veya iyi yordamsal çözümlerin yapılandırılmasında kullanılabilirler (Nonas ve Thorstenson, 2000). Malzeme kesme probleminin tamsayılı programlama problemi olarak ele alınması durumunda olurlu kesme planı sayısının büyüklüğü hesaplamayı imkansız kılmaktadır. Ancak malzeme kesme probleminin doğrusal programlama ile gösteriminde tamsayılı değişkenlerden arındırılması, olası kesme planı sayısının çözüme olumsuz yöndeki etkisini azalmaktadır (Suliman, 2001).

Malzeme kesme problemlerinin ilk modellerinden biri Gilmore ve Gomory (1961, 1963, 1965, 1966) tarafından yapılandırılmış ve geliştirilmiştir. Bu yaklaşım, tamsayılı programlama probleminin bir sütun üretme tekniği ile birlikte kullanılması ilkesine dayanmaktadır. Her sütun olurlu bir kesme planını temsil etmekte ve her tekrarda en büyük marjinal getirili sütunun hesaplanması gerekmektedir. Bu sütun ise tek-boyutlu bir sırt çantası probleminin çözümü ile elde edilmektedir. Bu yordamın büyük boyutlu problemlerin çözümü için pratik olmaması nedeniyle bazı ek algoritmalar önerilmiştir. Bu algoritmalardan bazıları önceki yöntem temel alınarak oluşturulmuş, bir tamsayılı problemin sütunlarını üretmek için çok iyi kesme planlarının geliştirilmesi amaçlanmıştır. Çoğu yaklaşım özellikle kesme planlarının geliştirilmesinde farklılık göstermektedir. Örneğin bir kesme planı, dinamik programlama ile veya ardı ardına giyotinle kesme işlemiyle üretilebilmektedir (Benati, 1997).

Klasik sırt çantası problemini içermeleri nedeniyle kesme problemleri NP-zor problemler sınıfında yer almaktadır. Parça sayısının artmasıyla farklı kesme planlarının sayısı üstel olarak artmaktadır. Bu nedenle kaynaklarda yer alan orta büyüklükteki problemlerin çözümünde genellikle yordama dayanan algoritmalar kullanılmıştır (Scheithauer ve Sommerweis, 1998).

Sol-alt (bottom-left) yerleştirme yordamı dikdörtgen parçaların ana malzeme üzerine yerleştirilmesi için kullanılan oldukça yaygın bir stratejidir. Söz konusu yordam, bir dikdörtgeni mümkün olan en aşağı seviyeye ve olabildiğince sola gelecek şekilde ana malzeme üzerinde uygun bir yere yerleştirir. Bu strateji genelde giyotinle kesme kısıtına uygun yerleşim planları üretemese de stratejinin hızlı yapısı onu kullanışlı bir araç yapmaktadır (Healy vd., 1999).

Kaynaklarda yer alan çalışmaların büyük çoğunluğunu, bir genetik algoritmanın bir yerleştirme yordamı (heuristic placement routine) ile birleştirildiği melez algoritmalar oluşturmaktadır. İki aşamalı olan bu yaklaşımda bir genetik algoritma küçük parçaların yerleştirileceği sıralamanın belirlenmesi amacıyla kullanılmaktadır. Daha sonra bu sıralamanın ana malzeme üzerine nasıl yerleştirileceğinin belirlenmesi için ikinci bir algoritmaya ihtiyaç duyulmaktadır (Hopper ve Turton, 2001b).

Meta-yordamlar, istenmeyen yerel en iyilere yakalanılan gizli tuzaklardan kaçarak en iyiye yakın çözüm arayan en iyileme yöntemleridir. Yüksek kısıtlı ve kesikli yapıdaki kesme problemlerinin çözümünde kısıtlarla başa çıkabilecek başarılı yöntemlerin bulunması büyük önem taşımaktadır. Bu problemlerde meta-yordamsal teknikler kullanıldığında, olurluluğun devam etmesi en önemli problem olarak ortaya çıkmaktadır (Amaral ve Wright, 2001). Bu kısıtlarla başa çıkmak için temel olarak üç strateji kullanılmaktadır:

- Çözüm uzayında olursuz çözümlere izin verilmesi ve bunlara ceza uygulanması,
- Aramanın sadece olurlu çözümlerle sırlandırılması,
- Bazı olursuz çözümlere izin verilmesi ancak olursuzluğun sınırlandırılması.

Bu bölümde genel malzeme kesme problemi yeri, anlamı, yapısı ve çözüm yaklaşımları yönüyle -ana hatlarıyla- tanıtılmıştır. Bu çalışmanın konusu olan dairesel malzeme kesme problemleri ise iki-boyutlu malzeme kesme problemlerinin bir alt kümesini oluşturmaktadır. Dyckhoff'un (1988), sınıflandırması kullanılarak dairesel malzeme kesme problemi, 2 / B / I / C simgeleriyle gösterilebilmektedir. Ele alınan problemde, dikdörtgen bir ana malzemeden farklı boyutlardaki dairesel şekilli sipariş parçalarının malzeme kullanımını en büyükleyecek şekilde kesilmesi amaçlanmaktadır.

Yaygın bir uygulama alanı olmasına karşın dairesel malzeme kesme problemleri kaynaklarda çok az bir yer tutmaktadır. Bu çalışmaların büyük çoğunluğunda ise sipariş parçalarının aynı boyutta olduğu durum ele alınmıştır. İzleyen bölüm, daha özele inerek dairesel kesme problemini ele almaktadır.

3. DAİRESEL KESME PROBLEMİ ve MEVCUT ÇÖZÜM YAKLAŞIMLARI

Bu bölümde iki-boyutlu kesme problemlerinin bir alt sınıfı olan dairesel malzeme kesme problemlerinin yapısı, modellenmesi ve çözümüne yönelik olarak bugüne kadar yapılan çalışmalardan ulaşılabilenler genel hatlarıyla ele alınıp tartışılmaktadır.

3.1. Problemin Modellenmesi ve Çözüm Yaklaşımlarının Sınıflandırılması

Dairesel malzeme kesme problemi; (L_0,W_0) boyutlarındaki bir dikdörtgen levhadan farklı boyutlardaki dairesel şekilli parçaların mümkün olan en çok sayıda kesilmesi olarak tanımlanmaktadır. Dairesel şekilli parçalar, yarıçapları r_i $(i = 1, \ldots, n)$ olacak şekilde n farklı tiptedir. Ancak kaynaklarda yer alan bazı çalışmalarda dairesel şekilli parçaların aynı boyutlarda olduğu durumlar da ele alınmıştır. Her i parçasına ait D_i maksimum talep miktarı söz konusu olduğunda bu problem kısıtlı dairesel kesme problemi olarak adlandırılmaktadır.

(L_0,W_0) boyutlarındaki bir dikdörtgen levhaya r yarıçapındaki n adet parçanın herhangi i ve j parça çiftleri arasındaki çakışmanın en fazla bir noktada olacak şekilde yerleştirilmesi probleminin çözümü için doğrusal olmayan bir karar modeli P1:

$$r - \frac{L_0}{2} \le x_i \le \frac{L_0}{2} - r \text{ ve,} \qquad \textbf{(7)}$$

$$r - \frac{W_0}{2} \le y_i \le \frac{W_0}{2} - r \text{ , i = 1, . . . , n.} \qquad \textbf{(8)}$$

k.a.

$$\text{Enk} \sum_{i<j} \text{Enb}\big(0, (2r \;\; - MAU_{ij})\big) \qquad \textbf{(9)}$$

olarak tanımlanmaktadır. Modelde yer alan x_i ve y_i değişkenleri, sırasıyla dairesel şekilli parçaların merkezlerine ait x ve y koordinatlarını temsil etmektedir. Kısıt 7 ve 8 parçaların ana malzeme sınırları içinde kalmalarını sağlamaktadır. MAU_{ij} değişkeni i ve j parçalarının merkezleri arasındaki mesafeyi göstermektedir. Amaç fonksiyonu parçalar arasındaki çakışmaların toplamı olarak tanımlanmakta, çakışma olmaması durumunda sıfır değerini almaktadır. Bu problemin en iyileme yöntemleriyle çözümü sonucunda amaç fonksiyonunun sıfır olması durumunda karar probleminin cevabı; verilen sipariş parçalarının dikdörtgen levhaya yerleşimi probleminin olurlu bir çözümünün bulunduğu, diğer durumlarda ise olurlu bir çözümün mevcut olmadığı şeklindedir.

Geliştirilen yaklaşımın aynı yarıçaplı parçaların r_0 yarıçaplı dairesel bir ana malzemeye yerleştirilmesi problemi için genişletilmesi durumunda karar modeli P2:

$$MAU_{i0} \le (r_0 - r) \text{ , i = 1,..., n.} \qquad \textbf{(10)}$$

k.a.

$$\text{Enk} \sum_{i<j} \text{Enb}\big(0, (2r - MAU_{ij})\big) \qquad \textbf{(11)}$$

olarak tanımlanmaktadır. Modelde yer alan MAU_{i0} değişkeni, i parçası ile dairesel ana malzemenin merkezleri arasındaki mesafeyi göstermektedir.

(9) ila (11)'de yer alan karar modellerinin r_1, r_2,…, r_n yarıçaplarında n adet sipariş parçasının yerleşimi probleminin modellenmesi amacıyla tekrar düzenlenmesi durumunda, farklı yarıçaplardaki dairesel parçaların dikdörtgen bir ana malzemeye yerleştirilmesi probleminin karar modeli P3:

$$r_i - \frac{L_0}{2} \le x_i \le \frac{L_0}{2} - r_i \text{ ve,} \tag{12}$$

$$r_i - \frac{W_0}{2} \le y_i \le \frac{W_0}{2} - r_i \text{, i = 1,..., n.} \tag{13}$$

k.a.

$$\text{Enk} \sum_{i<j} \text{Enb}\left(0, (r_i + r_j - MAU_{ij})\right) \tag{14}$$

olarak düzenlenebilir. Farklı yarıçaplardaki dairesel parçaların dairesel bir ana malzemeye yerleştirilmesi probleminin karar modeli P4 ise:

$$MAU_{i0} \le (r_0 - r_i) \text{ , i = 1,..., n.} \tag{15}$$

k.a.

$$\text{Enk} \sum_{i<j} \text{Enb}\left(0, (r_i + r_j - MAU_{ij})\right) \tag{16}$$

olarak tanımlanmaktadır.

İncelenen çalışmaların bir kısmı, daire dışında dikdörtgen parçaları da kabul eden daha genel kapsamlı yaklaşımlardır. Bir kısmı da, daire kesimine özgü olarak geliştirilmiş yöntemleri tanıtmaktadır. Bu gruba giren çalışmalardan bazıları sadece eşit çaplı dairesel şekilli parçaların kesimiyle ilgilenmekte (P1 ve P2), diğerleri ise farklı çaplı parçaları da kabul etmektedir (P3 ve P4).

Çalışmaların çoğunda -parçalar daire olmasına karşın- ana malzeme dikdörtgendir. Bunların dışında, ana malzeme olarak da daire kullanan çalışmalar (P2 ve P4) ayrı bir kategori olarak ele alınmıştır. Bu kategoride geliştirilen çözümler -bir tünele boruların yerleştirilmesi şeklinde örneklenebilecek olan- ikil (dual) istif problemi için daha anlamlıdır.

3.2. Dairesel Kesmenin Genel Amaçlı Problemlerde Ele Alınışı

Düzgün olmayan parçalar için geliştirilmiş yöntemler doğal olarak daire şeklindeki parçaların kesiminde de kullanılabilir. Burada özel olarak dikdörtgen parçaların yanında daireleri de kabul eden çalışmalardan söz edilecektir.

Stoyan ve Yaskov (1998), dikdörtgen ve daireleri malzeme kullanımını en iyileyecek şekilde yerleştirmek amacıyla bir matematik modelin geliştirilmesi ve bu modelin çözüm yönteminin belirlenmesi üzerinde çalışmıştır. Problemin bir matematiksel modeli geliştirilmiş, problemin özellikleri, uygun çözüm alanı ve uç noktalarda doğrusal bir amaç fonksiyonunun en iyi değeri araştırılmıştır. Problemin çözümü amacıyla dal ve sınır tekniği ile indirgenmiş eğim yönteminin bileşimine dayanan bir çözüm yaklaşımı kullanılmıştır. Uygun çözüm alanındaki tüm uç noktaların sıralanması için bir arama ağacı oluşturulmuş, ayrıca bir uç noktadan diğerine artan amaç fonksiyonu ile geçiş sağlanmıştır. Yazarlar geliştirdikleri yöntemi iki örnek problem üzerinde denemiştir. 100 adet dairenin büyük bir dikdörtgen içine yerleştirildiği birinci örnek problemde çözüm süresinin bir saat, 50 dikdörtgenin yerleştirildiği ikinci problemin çözüm süresinin ise 20 dakika olduğu belirtilmiştir. Ancak elde edilen sonuçların etkin olup olmadığından söz edilmemekte, kaynaklarda mevcut diğer yaklaşımlara göre avantaj ve dezavantajlarına değinilmemektedir.

Singh ve Sekhon (1998), dairesel ve dairesel olmayan parçaların bilgisayar destekli modellenmesi, basit kesme kalıpları için yerleşim seçeneklerinin oluşturulması ve bunların grafik gösterimi amacıyla düşük maliyetli bir sistem geliştirmiştir. Geliştirilen sistem, şerit kenarlarına ve kenar boşluklarına göre değişen şerit boyutlarını, köprü uzunluğunu, eğimi ve parçaların döndürülmesini göz önüne alabilmektedir. Önerilen yazılım basit kesme kalıpları için dairesel ve dairesel olmayan parçaların şerit üzerindeki farklı yerleşim düzenlemelerini araştırmak amacıyla ileri düzeydeki Auto-CAD ve AutoLISP komutlarını kullanmaktadır. Singh ve Sekhon, yordamın küçük ve orta büyüklükteki işletmeler için uygun olacağını belirtmiştir.

3.3. Dairesel Kesmeye Özgü Çalışmalar

Sadece dairesel şekilli parçalara yönelik çalışmalarda, özdeş dairelerin yerleştirilmesi (P1), belirli bir kurala göre yapılabildiği ve farklı çaptaki daireleri kabul eden (P3) yöntemlere göre daha sade modellenip, daha kolay çözümler verebildiği için, bu iki yaklaşım arasında önemli farklılıklar bulunmaktadır. Ayrıca, daireleri dikdörtgen yerine, daha büyük bir daire içine yerleştirmeye (P2 ve P4) yönelik çalışmalar, ilk ikisinden de farklı olmaktadır. Bu nedenle bu üç tür probleme yönelik çalışmalar burada ayrı ayrı ele alınacaktır.

3.3.1. Aynı büyüklükteki dairelerin yerleştirilmesi (P1)

Graham ve Lubachevsky (1996), benzetim kullanarak belirli bir sayıdaki n adet diskin üst üste gelmeyecek şekilde bir karenin içine yerleştirilmesi problemini incelemiştir. Kullanılan algoritma bilardo toplarının hareketlerinin benzetimine dayanmaktadır. Bu yaklaşımda topların boyutu, hareketlerin sona ermesine kadar yavaş yavaş artırılmaktadır.

Nurmela ve Ostergard (1997), verilen bir dikdörtgenin içine yerleştirilebilecek n adet eşit dairenin yarıçapının en büyüklenmesi problemini araştırmıştır. Bu yaklaşım, daireler arasındaki uzaklıklardan elde edilen enerji fonksiyonunun en küçüklenmesine dayanmaktadır. Ele alınan problem çok yaygın bir geometrik problemdir. Bu problemin bir benzeri ise, aralarındaki uzaklıklar en az d uzunluğunda olacak şekilde n adet noktanın bir kareye yerleştirilmesi için en büyük d sayısının bulunması problemidir. Yakın zamana kadar yapılan çalışmalarda, en çok 20 dairenin kare içerisine yerleştirilmesine yönelik tüm yerleşimler ele alınmıştır. Yazarlar, 20'den fazla dairenin iyi yerleşimlerine ilişkin hesaplama yöntemlerini inceleyerek; 50 daireye kadar bulunan en iyi yerleşimleri sunmuştur.

Nurmela ve Ostergard (1999), eşit büyüklükteki n adet dairenin üst üste gelmeyecek şekilde bir kare içine yerleştirilmesi durumunda, en büyük yarıçapın bulunması problemini ele almıştır. Kaynaklarda yer alan önceki çalışmalarda 20 ve daha küçük daire sayıları için en iyi yerleşimlerin belirlendiği, bu yerleşimlerin doğrulanması için ise yoğun bir biçimde bilgisayar desteğinden yararlanıldığı ancak elle kontrolün mümkün olmadığı belirtilmiştir. Bu çalışmada, eski yaklaşımların doğrulanması yapılarak mevcut yaklaşım geliştirilmiş ve daire sayısı 27'ye çıkarılmıştır. Bilgisayar destekli doğrulamalarda yaygın olduğu üzere bu çalışmadaki yerleştirmelerin en iyiliğinin doğrulanmasında da otomatikleştirilmiş ve nispeten basit adımlardan oluşan çok uzun zincirler kullanılmıştır. Daire sayısının 20 ve daha az olduğu önceki çalışmalarla ilgili ve daire sayısının 20 ila 27 arasında olduğu önerilen yaklaşımla ilgili bilgisayar sonuçlarına yönelik bazı istatistik veriler sunularak daire sayısının 20 ila 27 arasında olduğu yerleşimlerin şekilsel gösterimine yer verilmiştir.

Correia vd. (2000), aynı çaptaki dairesel tabanlı malzemelerin (tüpler, rulolar) dikdörtgen bir zemine (palet) yüklenmesi problemini tavlama benzetimi ile ele almıştır. Pratik kullanılabilirlik açısından tüm malzemeler aynı yükseklikte kabul edilmiştir. Ele alınan problem, dairesel tabanlı malzemelerin dikdörtgen palet üzerindeki yerleştirme planlarının belirlenmesi ve aynı zamanda palet üzerine yerleştirilen malzeme sayısının en büyüklenmesini içermektedir. Problemin çözümü amacıyla iki adet tavlama benzetim algoritması geliştirilmiştir (MinOSA ve MaxNSA). Bu iki algoritma kullanılan amaç fonksiyonları ve komşuluk (neighborhood) yapıları yönüyle farklılık göstermektedir. MinOSA algoritması, verilen sabit sayıdaki dairenin bir dikdörtgenin içerisine, daireler arasındaki çakışmaları en küçükleyecek bir biçimde yerleştirilmesine olanak sağlamaktadır. Algoritma, verilen dairelerin palet üzerine rastgele yerleştirilmesiyle başlar. Rastgele seçilen bir dairenin palet üzerinde sekiz doğrultudan biri yönünde hareket ettirilmesiyle komşuluk sağlanır. Bu komşuluk yapısı palet kenarlarıyla sınırlandırılmıştır. MinOSA algoritmasında, amaç fonksiyonu dairelerin toplam çakışma miktarı olarak tanımlanmıştır. Ancak bu fonksiyonun hesaplanmasındaki güçlük nedeniyle, daireler yerine her daireyi içine alan en küçük kare kullanılmıştır. MaxNSA algoritmasında ise daire sayıları sabit değildir. Algoritma, dikdörtgen malzemeye yerleştirilen daire sayısının artırılmasını ve buna paralel olarak dairelerin birbirleriyle olan çakışmalarının son çözümde en aza indirilmesini amaçlamaktadır. Algoritmanın işleyişi MinOSA ile benzerlik teşkil eder ancak; ilave olarak rastgele seçilen bir dairenin palet üzerinden çıkarılması veya rastgele seçilen bir dairenin palete yerleştirilmesi şeklinde iki ilave komşuluk ilişkisi kullanmaktadır. Her iki algoritma farklı türdeki problemler üzerinde test edilmiş, her iki algoritmanın da performansının iyi olmasına karşın, Palet_Alanı/Daire_Alanı oranı arttıkça algoritmaların etkinliğinin azaldığı ifade edilmiştir.

Correia vd. (2001), eşit çaplardaki dairelerin bir dikdörtgenin içine yerleştirilmesi problemi için yeni bir üst sınır önermektedir. Bu üst sınırın hesaplanması, bir en iyi çözümde bulunan önlenemeyen boş alanların miktarının belirlenmesi prensibine dayanmaktadır. İdeal olarak, yeni üst sınırın etkinliğinin belirlenebilmesi için bu yeni üst sınırın en iyi çözümlerle kıyaslanması gerekmektedir. Ancak probleme ilişkin kaynaklardan alınan matematiksel modelin doğrusal olmayan yazılım paketlerini içeren yöntemlerle çözümünün denendiği fakat başarısızlıkla sonuçlandığı belirtilmiştir. Bu nedenle, elde edilen sonuçların kalitesi, tavlama benzetimi uygulanarak elde edilen çözümlerle doğrulanmıştır.

Cui (2005), büyük elektrik jeneratörlerinin stator ve rotor saclarının imalinde kullanılan T-kanalı içeren dairesel parçaların kesme planlarının hazırlanması amacıyla bir algoritma önermektedir. Kesme planlarında, bir şerit içinde özdeş boşluklardan oluşan birden fazla satır yer alabilmektedir. Bir şerit, x doğrultusu ve y doğrultusu olarak adlandırılan iki dik doğrultudan birinde yer almaktadır. Algoritma, kesme planlarındaki şeritlerde yer alan boşluk satırlarının, şerit sayılarının ve yönlerinin en iyi kombinasyonlarını belirlemek amacıyla sırt çantası algoritması ve bir tür gizli sayımlama yöntemi kullanmaktadır. Geliştirilen yaklaşım, büyük elektrik jeneratörleri üreten bir firmanın gerçek malzeme kesme verilerine uygulanmıştır. Elde edilen sonuçların, hesaplama süresi ve malzeme kullanımı yönüyle, geliştirilen algoritmanın etkin olduğunu gösterdiği ifade edilmektedir.

Birgin vd. (2005), aynı yarıçaptaki silindirlerin daha büyük bir konteynere yerleşimi probleminin çözümü amacıyla doğrusal olmayan karar problemine dayalı bir yordam geliştirmiştir. Bu yordam, silindir merkezlerinin konteyner tabanı ile elde edilen dikdörtgenin içinde (kenarlardan en az bir yarıçap mesafede) yer alması gerektiği ve birbirlerinden en az bir çap mesafede olmaları durumuna dayanmaktadır. Bu temel varsayım kullanılarak geliştirilen yordam, bu kısıtları sağlayan silindir merkezlerinin en büyük sayısını bulmaya çalışmaktadır. Problemin sürekli yapısının bu çalışmaya başlanmasına olumlu yönde etki eden sebeplerden biri olduğu yazarlar tarafından ifade edilmektedir. Çalışma sonucunda, önerilen yaklaşım kaynaklarda yer alan diğer yöntemlerle karşılaştırılmış olup, daha iyi sonuçlar elde edildiği belirtilmektedir.

3.3.2. Farklı büyüklükteki dairelerin yerleştirilmesi (P3)

Fraser ve George (1994), kağıt üreticisi olan bir firma için bir "konteyner yükleme yazılım paketi" geliştirmiştir. Program girdi olarak, kağıt ürünlerine ilişkin siparişleri levha veya rulo şeklinde alarak, kesin sipariş miktarlarını ve nakliye için istifleme planlarını üretmektedir. Ruloların konteynere yerleştirilmesi için iki ayrı algoritma kullanılmaktadır. İlk algoritma, ruloların konteyner tabanına yerleştirilmesi için sayımlama yöntemiyle iki boyutlu bir plan oluşturmaktadır. Ruloların konteyner içinde düşey olarak birden fazla farklı şekilde yerleştirilmesinin mümkün olması nedeniyle ikinci algoritma, konteyner yüksekliğini etkin bir şekilde kullanmak amacıyla, ruloların üst üste istiflenmesini sağlamaktadır. Bu algoritmada, sırasıyla her istif için yerleştirilmemiş rulolardan en büyüğü seçilmekte, boş kalan hacim ise kalan rulolar arasından sayımlama ile en etkin şekilde doldurulmaktadır. Sipariş büyüklükleri genellikle bir konteynerden büyük olduğu için tüm algoritmalar çoklu-konteyner durumu için tasarlanmıştır. Yazarlar, hazırladıkları programı konteyner sayıları iki ile sekiz arasında değişen 100 problem üzerinde başarıyla test etmiştir. Uygun rulo çapları, etkin istifleme planları ve maliyet etkin nakliyeye uygun üretim miktarları gibi çeşitli karar durumları açısından programın etkin olduğu, ayrıca istifleme ve nakliye dokümanlarının hazırlanması için ihtiyaç duyulan süreyi de azalttığı iddia edilmektedir.

Hifi vd. (2004), kısıtlı ve kısıtsız dairesel kesme problemi için tavlama benzetimine dayanan bir yordam önermektedir. Bu çalışmada, bir enerji fonksiyonu belirlenmiş ve küçük değerleri için başlangıç dikdörtgeninin sol alt köşesinde dairesel parçalar için iyi bir yoğunluk sağlanmıştır. Her parçaya ait fayda olarak o parçanın alanı alınmıştır. Böylece herhangi bir çözümün kalitesi o çözümün dikdörtgen ana malzeme üzerinde kapladığı alanla ölçülmektedir. Başlangıç çözümü olarak büyük miktarda parça dikdörtgen ana malzemeye rastgele yerleştirilmektedir. En büyük sayıda parçanın dikdörtgen içerisine yerleştirilmesini sağlamak amacıyla parçaların dikdörtgenin sol alt köşesine doğru yoğunlaşmaları sağlanmakta ve başlangıç çözümünde parça sayısı dikdörtgenin yüzeyini kaplayacak kadar büyük seçilmektedir. Parçaların birbirleriyle ve dikdörtgenle kesişmesi nedeniyle, bazı parçaların çözümden çıkarılmasıyla kesişmelerin önlenmesine çalışılmaktadır. Kullanılan yöntemin az da olsa olursuz çözümler ürettiği, ancak bu çözümlerin de son bir işlemden geçirilerek tüm çözümlerde olurluluğun sağlanabildiği belirtilmektedir. Kullanılan yöntemin kaynaklarda yer alan diğer yaklaşımlarla karşılaştırıldığında iyi sonuçlar ürettiği ifade edilmiştir.

Hifi ve M'Hallah (2004), L_0 ve W_0 boyutlarındaki bir dikdörtgen levhanın mümkün en çok sayıda dairesel şekilli parçaya kesilmesi problemini ele almıştır. Dairesel parçalar yarıçapları r_i (i = 1, . . . , n) olacak şekilde n farklı türdedir. D_i'nin i. parçaya ait en büyük talep olduğu kısıtlı dairesel kesme problemi, iki yordam kullanarak çözülmüştür (bir kurma esaslı yordam ve bir genetik algoritma tabanlı yordam). Bu yaklaşımların her ikisi de parçalara ait iyi bir sıralama araştırır ve bu sıralama kümelerinin en iyi yerleşimini bulmak için "en iyi yerel konum" yordamının (Hifi ve M'Hallah, 2002), değişik bir versiyonunu kullanır. Bu yerleştirme yordamı dairesel kesme problemleri için özel olarak geliştirilmiş olup, kısıtlı problemler için genetik algoritmanın mutasyon işlemcilerinden biri gibi davranır. Her iki yaklaşım, kaynaklardan alınan problemler kullanılarak mevcut algoritmalarla karşılaştırılmıştır. Hesaplama sonuçları, önerilen yaklaşımların uygun sürelerde yüksek kalitede çözümler ürettiğini göstermiştir. Genetik algoritma tabanlı yordamın kolaylıkla paralelleştirilebilecek olmasının yakın gelecekte araştırılması gereken önemli bir özelliği olduğu ifade edilmektedir.

Zhang vd. (2004), farklı yarıçaplardaki dairelerin dikdörtgen bir konteynere yerleştirilmesi problemini incelemiştir. Bu problem doğrusal olmayan bir en iyileme problemi olarak yapılandırılmış ve çözümü için bir tavlama benzetimi yordamı geliştirilmiştir. Kullanılan yöntem, dairelerin fiziksel hareket etme sürecinin benzetimine ve çakışan dairelerin birbirleriyle olan kesişmelerinin en küçük potansiyel enerji kuralıyla ilişkilendirilmesine dayanmaktadır. Doğadan elde edilen bu stratejiler hızlı yakınsama sağlamaktadır. Algoritma kaynaklardan alınan diğer algoritmalarla karşılaştırılmış ve beş kat daha hızlı sonuç verdiği belirtilmiştir.

Stoyan ve Yaskov (2004), farklı yarıçaplardaki dairelerin bir şerit üzerine yerleştirilmesi problemini ele almıştır. Problemin çözümü amacıyla, bir matematik model geliştirilerek bu modelin çeşitli özellikleri incelenmiştir. Bu özellikler dikkate alınarak, bir yerel en iyiden diğerine geçiş yapılması olarak tanımlanan özgün bir yöntem önerilmektedir. Problem boyutunun arttırılması fikrine ve indirgenmiş eğim yöntemine dayanan bu yaklaşım çeşitli problemler üzerinde test edilmiş olup, daire sayısının 15 ile 35 arasında olması durumunda yeterince etkin olduğu, kaynaklarda yer alan bir dal ve sınır tekniği ile karşılaştırıldığında ise daha üstün olduğu ifade edilmektedir. Ayrıca önerilen yöntemde paralel hesaplamaların kolaylıkla kullanılabilecek olması nedeniyle yöntemin etkinliğinin daha da artırılabileceği iddia edilmektedir.

3.3.3. Daire içine dairelerin yerleştirilmesi (P2, P4)

WenQi vd. (2001), farklı çaptaki dairelerin, dairesel bir konteynere yerleştirilmesi problemini çözmek amacıyla fiziksel bir modele dayanan ve yerel en iyilerden kaçınmak için toplumsal davranışlardan esinlenen bir strateji kullanan yerel bir arama algoritması geliştirmiştir. Bu yaklaşımda, öncelikle orijinal matematiksel problemden başlanarak orijinal probleme eş değer doğal olaylar araştırılmış ve doğal olayların gözlenmesi ve benzetimi ile orijinal problemin çözümüne yönelik bir algoritma geliştirilmiştir. Böylece dairesel yerleştirilme problemi sürekli bir en iyileme problemine dönüştürülmüştür. Bu problemin çözümü için indirgenmiş eğim yöntemi kullanılmıştır. Yerel arama yöntemi çoğu zaman yerel en iyilere takılmaktadır. Kalabalık bir otobüste en çok sıkışan insanlar her zaman yerlerini değiştirerek daha iyi bir yere geçmeye çalışır. Bu toplumsal davranış kullanılarak, -en çok sıkışan parçanın konteyner içerisinde rastgele başka bir yere yerleştirilmesi olarak tanımlanan- yerel en iyilerden kaçınacak bir strateji geliştirilmiştir. Yazarlar, kaynaklarda konuyla ilgili örnek bulunmaması nedeniyle problem örneklerini kendileri türetmiştir. Yerleştirilecek dairelere oranla konteynerin büyük olduğu kolay örnekler için algoritmanın çok hızlı sonuç verdiği, yerleştirilecek daireler için konteynerde sınırlı yer bulunan, daire büyüklüklerinin çok farklılık gösterdiği ve daire sayısının büyük olduğu zor problemler için çözüm süresinin farklılık gösterdiği belirtilmiştir.

Farklı yarıçaplardaki dairelerin daha büyük bir daire içerisine yerleştirilmesi problemi üzerinde çalışan Wang vd. (2002), benzetim esaslı bir yaklaşımdan yararlanmıştır. Bu yaklaşımda, problemin eş değeri olan bir fiziksel gerçekliği temel almışlardır. Buna göre; daireler birim uzunlukta fakat farklı yarıçaplarda silindirler olarak düşünülmüş ve ana malzemenin ise aynı uzunlukta ve ortasında yuvarlak bir boşluk olduğu varsayılmıştır. Hem silindirler hem de ana malzeme üzerindeki boşluk düz, pürüzsüz ve esnektir. Tüm silindirlerin ana malzeme üzerindeki boşluğa paralel bir biçimde yerleştirilmesi durumunda tüm nesneler kendi doğal şekillerine ve büyüklüklerine geri dönme eğiliminde olacağından silindirler arasında sıkışma ve çarpışmalar olacaktır. Bir seri hareketten sonra ise sistem kararlı bir duruma dönecek ve sıkışma ve çarpışmalar ortadan kalkacaktır. Buradan yola çıkarak, silindirlerin fiziksel hareketlerinin matematiksel modellenmesi ve bu modele gradyant yöntemi uygulanması ile ana problemin çözümüne katkı sağlanmıştır. Ancak bazı durumlarda silindirler halen sıkışma durumunda olmasına rağmen hareket edememektedirler. Bu durum izleyen tekrarlarda da değişmemekte ve problemin çözümü olmasına rağmen bir çözüm elde edilememektedir (yerel en iyiye sıkışma). Karşılaşılan bu sorunun önlenebilmesi amacıyla insan tecrübesinden ve vizyonundan esinlenilmiştir. Bir toplulukta insanların rahatsız olduklarında en çok rahatsız olanların yerlerini değiştirdiği düşünülerek bu davranış biçimi dairelere uygulanmıştır. Fiziksel model ve insan davranışları birleştirilerek yerel en iyi ile karşılaşıldığında en çok fiziksel kuvvete maruz kalan dairenin yeri rastgele değiştirilerek tekrara devam edilmiş, böylece bütünsel en iyi çözüm elde edilebilmiştir. Algoritmanın etkinliği sayısal örnekler üzerinde denenmiş ve elde edilen sonuçların, kullanılan yöntemin etkinliğini gösterdiği, ancak kaynaklarda bulunan problemlerde yeterli veri olmaması nedeniyle çözümün hızı ve kalitesi yönüyle bir karşılaştırma yapılamadığı belirtilmiş, kullanılan yöntem ile özellikle zor problemler göz önüne alındığında çözüm süresinde önemli bir azalma elde edildiği ifade edilmiştir.

3.4. Mevcut Çözüm Yaklaşımlarının Tartışılması

Makine imalatında, rondela, silindirik kutu kapağı, elektrik makinelerinin rotor ve stator sacları gibi çok çeşitli dairesel şekilli parçaların sac levhalardan kesilmesi gerekmektedir. Sac dışında cam, lastik, kösele, kumaş gibi pek çok farklı malzemede de zaman zaman dairesel kesme yapılmaktadır. Uygulamadaki bu yaygın kullanıma karşın, dairesel kesme alanındaki çalışmaların kesme problemi konusundaki yüzlerce yayın arasında çok küçük bir yer tutması dikkati çekmektedir. Çalışmaların daha çok dikdörtgen parçaların kesimi alanında yoğunlaşmış olması, bu problemin daha sade yapısı ve çözüm kolaylığı yanında, dairesel kesmenin daha çok "düzgün olmayan şekillerin kesilmesi problemi" içinde düşünülmüş olmasıyla da açıklanabilir.

Oysa, özel geometrik yapısı dolayısıyla dairesel kesme problemine, düzgün olmayan şekillere göre, daha verimli çözümler getirilebilir. Bu nedenle, dairesel kesme problemi, hem kuramsal hem de pratik açılardan ilgilenilmeye değer görülmektedir. Bu çalışmanın hareket noktası da bu görüş olmuştur.

Erişilen kaynakların bir kısmında, dairesel kesme probleminin genel iki-boyutlu kesme problemi içinde ele alındığı, geriye kalanında ise konuya tamamen dairesel kesmeye yönelik olarak yaklaşıldığı gözlenmektedir. Daire kesmeye özgü çalışmalarda da dairelerin eşit çaplı olup olmamasına göre amaçlar, kısıtlar yönüyle çözüm yaklaşımları önemli farklılıklar göstermektedir.

Daireleri dikdörtgen içine yerleştirenler yanında (ana malzeme daha çok dikdörtgendir) daha büyük daireler içine de yerleştirmeye çalışan araştırmacılar da vardır. Bunun gibi, daire içine dikdörtgenler yerleştirmek için geliştirilecek çözümler de kerestenin tomruklardan en az fireyle elde edilmesi problemine katkı sağlayabilir.

Bazı çalışmalar ağrılıklı olarak uygulamaya yönelik olduğu için, kesme planları türetecek bir paket programın hazırlanmasını veya çözümün AutoCAD gibi standart bir çizim paketiyle bütünleştirilmesini hedeflemişlerdir. Bu tür çalışmalar ayrıca, sipariş ve malzeme takibi gibi işletmeye yönelik problemleri de çözebilecek şekilde bir tür veri tabanı ve karar destek sistemi ile de ilişkilendirilmişlerdir. Bazı çalışmaların da elektrik makinelerinin silisli sacları gibi özel parçalara yönelik yapıldığı görülmektedir.

Çalışmalarda firenin, fire oranının en küçüklenmesi veya kesilen parça sayısının, malzemeden yararlanma oranının, sağlanan faydanın en büyüklenmesi gibi amaçlar güdülmektedir. En önemli kısıt türünün, geometrik kısıtlar olduğu görülmektedir. İki çeşit geometrik kısıttan söz edilebilir: kesilecek parçanın ana malzemeye sığması ve parçaların üst üste binmemesi.

Küçük problemler, matematik modellerle çözülebildiği için, burada dal sınır tekniğinin ve sırt çantası probleminin değişik versiyonları kullanılabilmiştir. Gradyant esaslı yöntemlerden de sıkça yararlanıldığı görülmektedir. Ancak büyük problemler için sezgisel veya meta sezgisel yöntemlerin kullanılması kaçınılmaz olmaktadır.

Sezgisel yöntemlerde kurma veya iyileştirme esaslı yaklaşımlardan yararlanılmaktadır. Kurma esaslı yaklaşımlar, bir atama sırasının yapılmasını gerektirmektedir. Bu tür sıralama problemleri de meta sezgisellerle çözüm için son derece elverişlidir. Ancak kesme planı oluşturmanın ikinci adımı, sırası gelen parçayı ana malzemede bir yere yerleştirmek olduğundan, bu iş için de ayrı bir yordam, bir kural veya yapı geliştirmek gerekmektedir. Eşit çaplı dairelerin yerleşimi bir dereceye kadar kolay olmaktaysa da, farklı çaptaki daireleri yerleştirmek için değişik yöntemler geliştirmek gerekmektedir.

İyileştirme esaslı algoritmalar ise daha çok ikili yer değişimi düşüncesine dayanmaktadır. Bu karşılıklı yer değişimi sonucunda parçalar birbirine daha çok yaklaşabilmektedir. İyileştirme algoritmaları bu değişikliği ya rassal olarak yapmakta (tavlama benzetiminde böyle olmaktadır) ya da tüm değişim olanaklarını sınadıktan sonra, en büyük kazancı sağlayacak değişimi yapmaktadır (en dik iniş). Bu ikili değişimlerin, makul sayıda tekrarıyla en iyiye yakın (çok zaman da en iyi) çözümlere ulaşılabilmektedir.

İkili yer değişimiyle elde edilen çözümler, bazı kaynaklarda 2-opt (2'li değişimlerle elde edilen en iyi) olarak anılmaktadır. Benzer şekilde üç parçanın yer değişimleri ile elde edilen 3-opt, n parçanın (hepsinin) yer değişimi ile elde edilen çözüm de n-opt çözümdür. En iyi çözüme ulaşmak, ancak 2'li, 3'lü, … ; n'li çözümlerin tümü yapıldığında garanti edilebilmektedir (sayımlama). Problemin kombinatorik yapısı ise, ortaya çıkan çok büyük değişim sayısı dolayısıyla bu noktaya ulaşmaya izin vermemekte, bu nedenle çok zaman ikili değişimlerle yetinilmektedir.

Meta sezgisel yöntem olarak en çok tavlama benzetiminin kullanıldığı görülmüş, genetik algoritmaya dayanan sadece bir çalışmaya ulaşılabilmiştir. Meta sezgisel yöntemlere dayanan ve bu çalışma sırasında incelenen makalelerin isimleri ile nitelik ve yaklaşım yönüyle karşılaştırılmaları toplu olarak Çizelge 3.1'de verilmektedir.

Çizelge 3.1. Dairesel kesme konusunda yapılmış çalışmaların karşılaştırılması

No	Makale İsmi	Yazarlar ve Yayın Yılları	Problemin Niteliği	Çözüm Yaklaşımı
1	Integrated container loading software for pulp and paper industry.	Fraser ve George, (1994)	Dikdörtgen ana malzeme, farklı boyutlarda dairesel parçalar	Belirlenen bir yordama göre mümkün kesme planlarının sayımlanması
2	Repeated patterns of dense packing of equal disks in a square.	Graham ve Lubachevsky, (1996)	Kare şeklinde ana malzeme, aynı boyutlarda dairesel parçalar	Bilardo toplarının hareketlerinin benzetimi
3	Packing up to 50 equal circles in a square.	Nurmela ve Ostergard, (1997)	Dikdörtgen ana malzeme, aynı boyutlarda dairesel parçalar	Çemberler arasındaki uzaklıklardan elde edilen enerji fonksiyonunun en küçüklenmesi
4	Mathematical model and solution method of optimization problem of placement of rectangles and circles taking into account special constraints.	Stoyan ve Yaskov, (1998)	Dikdörtgen Ana malzeme, farklı boyutlarda dikdörtgen ve dairesel parçalar	Dal ve sınır tekniği ile eğim azaltma yönteminin bileşimi
5	A low-cost modeler for two-dimensional metal stamping layouts.	Singh ve Sekhon, (1998)	Şerit biçiminde dikdörtgen ana malzeme, aynı boyutlarda dairesel ve dairesel olmayan parçalar	İleri düzeydeki Auto-CAD ve AutoLISP komutları
6	More optimal packing of equal circles in a square.	Nurmela ve Ostergard, (1999)	Kare şeklinde ana malzeme, aynı boyutlarda dairesel parçalar	Sayımlama ve en iyi olmayan çözümlerin elenmesi
7	Cylinder packing by simulated annealing.	Correia vd., (2000)	Dikdörtgen ana malzeme, aynı boyutlarda dairesel parçalar	Tavlama benzetimi

No	Makale İsmi	Yazarlar ve Yayın Yılları	Problemin Niteliği	Çözüm Yaklaşımı
8	Local search based on a physical model for solving a circle packing problem.	WenQi vd., (2001)	Dairesel ana malzeme, farklı boyutlarda dairesel parçalar	Sürekli bir en iyileme problemi biçiminde modelleme, azalan eğim yöntemi
9	A new upper bound for the cylinder packing problem.	Correia vd., (2001)	Dikdörtgen ana malzeme, aynı boyutlarda dairesel parçalar	Bir en iyi çözümde bulunan önlenemeyen boş alanların miktarının belirlenmesi
10	An improved algorithm for the packing of unequal circles within a larger containing circle.	Wang vd., (2002)	Dairesel ana malzeme, farklı boyutlarda dairesel parçalar	Benzetim ile modelleme ve bu modele gradyant yöntemi uygulanması
11	A simulated annealing approach for the circular cutting problem.	Hifi vd., (2004)	Dikdörtgen ana malzeme, farklı boyutlarda dairesel parçalar	Tavlama benzetimi
12	Approximate algorithms for constrained circular cutting problems.	Hifi ve M'Hallah, (2004)	Dikdörtgen ana malzeme, farklı boyutlarda dairesel parçalar	Bir kurma esaslı yordam ve bir genetik algoritma tabanlı yordam
13	Packing different-sized circles into a rectangular container using simulated annealing algorithm.	Zhang vd., (2004)	Dikdörtgen ana malzeme, farklı boyutlarda dairesel parçalar	Doğrusal olmayan modelleme ve tavlama benzetimi
14	A mathematical model and a solution method for the problem of placing various-sized circles into a strip.	Stoyan ve Yaskov, (2004)	Şerit biçiminde dikdörtgen ana malzeme, farklı boyutlarda dairesel parçalar	Problem boyutunun arttırılması ve indirgenmiş eğim yöntemi
15	A cutting stock problem and its solution in the manufacturing industry of large electric generators.	Cui, (2005)	Dikdörtgen ana malzeme, aynı boyutlarda dairesel ve dairesel olmayan parçalar	Sırt çantası algoritması ve sayımlama yöntemi
16	Optimizing the packing of cylinders into a rectangular container: A nonlinear approach.	Birgin vd., (2005)	Dikdörtgen ana malzeme, aynı boyutlarda dairesel parçalar	Doğrusal olmayan karar problemine dayalı bir yordam

Kaynak taraması, dairesel kesme probleminin çözümünde genetik algoritmalardan yeterince yararlanılmadığını göstermektedir. Oysa genetik algoritmalar (GA), kombinatorik problemlerin çözümünde çok güçlü bir araç olarak tanınmaktadır. Bu nedenle, bu çalışmada dairesel kesme probleminin çözümü için GA tabanlı bir yöntemin geliştirilmesi hedeflenmiştir. GA'ların ana hatlarıyla tanıtılması ve bu çalışmada geliştirilen GA tabanlı özgün bir çözüm yöntemi dördüncü bölümde ele alınmaktadır.

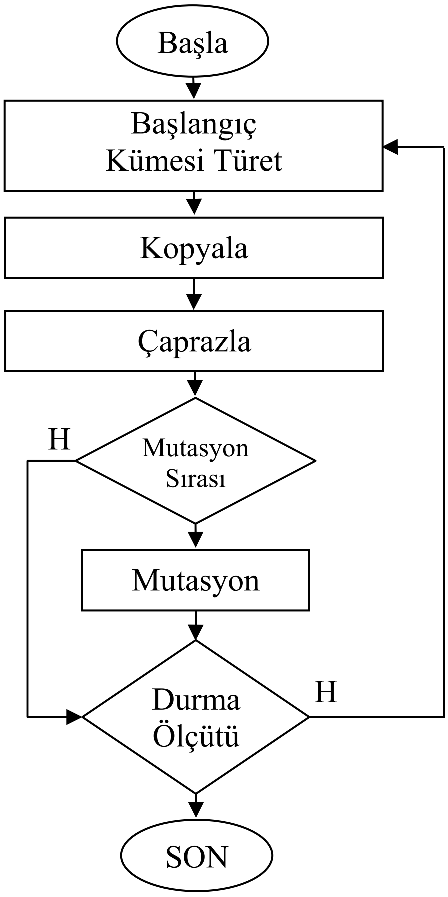
4. GENETİK ALGORİTMALAR ve DAİRESEL KESME PROBLEMLERİNİN ÇÖZÜMÜ İÇİN GELİŞTİRİLEN YAKLAŞIM

Bu çalışmada, dairesel kesme probleminin çözümü için genetik algoritma tabanlı bir yaklaşım geliştirilmiştir. İzleyen kesim; genetik algoritmaların ana hatlarıyla tanıtılması, geliştirilen yaklaşım, parametrelerin belirlenmesi ve geliştirilen yaklaşımın sınanmasını içermektedir.

4.1. Genetik Algoritmalar

Özellikle bir dizi şeklinde modellenebilen problemlerin çözümü için çok elverişli olan genetik algoritmalar, doğada gözlenen evrim sürecine benzer bir biçimde çalışan arama ve en iyileme yordamlarıdır. Arama işlemi “en iyi olanın hayatta kalması ilkesi” kullanılarak daha iyi çözümlere yönlendirilmektedir. Bu ise, bir çözümdeki en çok arzulanan özelliklerin, artırılarak ve birleştirilerek yeni neslin oluşturulması ile elde edilmektedir. Her çözüm kalite yönüyle değerlendirilip üstün bireyler seçilerek üreme işlemine alınmaktadır. Bu işlemin belirli bir sayıda nesle uygulanması sonucunda ise en iyi veya en iyiye yakın çözümler elde edilmektedir (Hopper ve Turton, 2001b).

Genetik algoritmalar, esasta problem bağımlı sezgisel yordamlar olduklarından, kromozom yapıları, operatörler ve stratejiler problemden probleme değişmektedir. Ancak kopyalama, çaprazlama ve mutasyon operatörlerinin değişik versiyonları ile seçim amacıyla rulet kuralı, turnuva gibi tekniklerin ve en iyi çözümü kaybetmemek için elitist (seçkinci) stratejilerin yaygın olarak kullanıldığı gözlenmektedir. Tipik bir genetik algoritma ana hatlarıyla izleyen akış şemasındaki yapıyla gösterilebilir.



Şekil 4.1. Tipik bir genetik algoritma

4.1.1. Olurlu çözümlerin türetilmesi, değerlendirilmesi ve iyileştirilmesi

Bir genetik algoritma tasarımında ilk adım, olurlu çözümlerin kromozom yapılarına karşı gelen karakter dizileriyle nasıl ifade edilebileceğinin düşünülmesi olmaktadır. Malzeme kesme probleminde de kesme planını modelleyecek şablon tasarlandıktan sonra, başlangıç popülasyonunu oluşturacak kromozomlar, bu kalıba uyacak şekilde rassal olarak türetilirler. Bunların ne kadar başarılı olduğunu belirlemek için bir de başarı ölçütünün (uyum fonksiyonu) tanımlanması gerekmektedir. Malzeme kesme problemlerinde, fire oranı bir başarı göstergesi olarak kullanılabilir.

Bu çözümlere genetik operatörler uygulandıkça, uyum gücü daha yüksek kuşaklar elde edilmeye başlanacaktır. Bu iyileşme önce hızlı olmakta, sonra da yavaşlayıp durmaktadır. Daha fazla iyileşme olmayacağı anlaşıldıktan sonra, algoritmanın durdurulması gerekir. Bu amaçla da önceden belirlenmiş bir durma ölçütü kullanılır.

Bir kesme planının doğal gösterimi, her bir parçanın ana malzeme üzerine yerleştirildiği koordinatlara dayanmaktadır. Bu gösterimin avantajı kesme planının tekrar düzenlenmesinin kolay olmasıdır. Ancak koordinatlarda küçük değişiklikler yapılması durumunda, yeniden düzenlenen kesme planında parçaların üst üste gelmesi durumu ortaya çıkabilmektedir. Doğal gösterimin bu özelliği ise genetik algoritmalar için uygun değildir. Bu nedenle daha kullanışlı bir veri yapısına ihtiyaç duyulmaktadır (Liu ve Teng, 1999).

Kaynaklarda yer alan çalışmalarda genellikle kesme planları küçük parçaların ana malzeme üzerine yerleştirileceği sıraya karşı gelen permütasyonlarla gösterilmektedir. Ancak bu durumda her bir sıralamanın ana malzeme üzerindeki yerleşiminin belirlenmesi için ikinci bir yordama ihtiyaç duyulur. Bu gösterimin avantajı, genetik algoritmanın çaprazlama ve mutasyon operatörleri tarafından sıralamanın değiştirilmesi ile yeni permütasyonların çok kolay bir şekilde elde edilebilmesidir.

Genetik yaklaşımda kromozom yapısının belirlenmesinden sonra verilecek karar, başlangıç popülasyonunun büyüklüğünün ne olacağı ve bunun ne şekilde oluşturulacağıdır. Türetme, genellikle olurlu çözümlerin rastgele üretilmesi şeklinde olmaktadır. Başlangıç popülasyonunun büyüklüğü problemin son aşamaya yakınsamasına kuvvetli bir biçimde etki etmektedir. Küçük bir popülasyon, problemin arama uzayını azaltırken bunun tersine büyük bir popülasyon ise hesaplama maliyetini artırmaktadır (Reeves, 1993: Babu ve Babu'dan, 1999). Bu nedenle, yöntemin başarısı popülasyon büyüklüğünün uygunluğuna bağlı olmaktadır.

Uyum fonksiyonu, popülasyonda bulunan bireylerin kalitesinin belirlenmesi amacıyla kullanılmaktadır. En yüksek uyum değerine sahip bireylerin daha yüksek olasılıkla üreme işlemine alınması ve yeni neslin oluşturulmasında bu bireylerin kullanılması genetik algoritmaların en iyi çözüme yakınsamasında büyük rol oynamaktadır. Bir malzeme kesme probleminde kullanılabilecek uyum fonksiyonu, "etkin bir biçimde kullanılan alanın toplam alana bölünmesi" olarak tanımlanabilir (malzemeden yararlanma oranı).

Başlangıç nüfusunun oluşturulması, çaprazlama ve mutasyon işlemleri sonucunda üretilen çocuk kromozomlar ile ebeveyn kromozomlar yeni neslin oluşturulmasında ana unsurlardır. Yeni neslin üretimi için kullanılacak bireyler, bir önceki nesilde bulunan çocuk ve ebeveyn kromozomlar arasından seçim tekniklerinden biri kullanılarak uygunluk değerlerine göre belirlenir ve üreme havuzuna atılır. Yeni nesli oluşturacak elemanlar bunlar arasından uyum değerleri ile orantılı olarak seçilmektedir.

Seçim işlemi ile yeni bir popülasyon oluşturulurken eski popülasyondan sadece çocuk kromozomlar kullanılabileceği gibi tüm ebeveynler ve çocuk kromozomlar veya bunların bir karışımı da kullanılabilmektedir. Bu işlem örnekleme uzayının (sampling space) belirlenmesi olarak tanımlanmaktadır. Örnekleme yöntemi (sampling mechanism) ise örnekleme uzayından kromozomların nasıl seçileceği ile ilgilidir. Bu amaçla rassal, deterministik ve karma (mixed) olmak üzere üç temel yaklaşım kullanılmaktadır.

Rassal örnekleme yöntemlerinden en iyi bilineni oransal seçim (proportional selection) veya rulet seçimi (roulette wheel selection) olarak adlandırılmaktadır. Bu yöntemde çark üzerindeki merkez açılarının büyüklükleri bireylerin uygunluk değerlerini temsil etmektedir. Bir sonraki nesilde yer alacak bireyin seçimi için çark rastgele döndürülür. Çark üzerinde büyük temsil edilen yüksek uyum değerli bireyler seçilmek için daha çok şansa sahiptir. Deterministik yöntemlerden bazıları budama (truncation) seçimi, elitist seçimi ve kuşak değişimi (generational replacement) olarak tanımlanmaktadır. Kuşak değişiminde yeni nesil sadece çocuk kromozomlardan oluşmaktadır. Karma örnekleme yönteminin çok iyi bilinen bir örneği ise rassal turnuva seçimi işlemidir. Bu yöntemde rassal olarak seçilen belirli sayıdaki birey arasından en yüksek uyum değerine sahip olanlar yeni nesle aktarılmaktadır.

Bir kromozomun seçilme olasılığının (selection probability) uyum değeri ile orantılı olarak belirlenmesi durumunda bazı istenmeyen durumlarla karşılaşılabilmektedir. Örneğin GA'nın ilk nesillerinde üstün özellikteki kromozomlar seçim işleminde çok fazla baskın olabilmekte (selection pressure) bu ise erken yakınsamaya (premature convergence) sebep olmaktadır. Bunun yanı sıra GA'nın son safhalarında ise kromozomların uyum değerlerinin birbirine yakın olması nedeniyle üstün olanların diğerleri ile rekabet etmesi daha güç olmaktadır. Seçim olasılığının belirlenmesinde karşılaşılan bu sorunların önlenmesi amacıyla geliştirilen ölçeklendirme (scaling) ve sıralama (ranking) tekniklerinden bazıları doğrusal ölçeklendirme, dinamik doğrusal ölçeklendirme ve sigma budama olarak adlandırılmaktadır.

Genetik algoritmalar en iyi çözümün elde edilebileceğini garanti etmemektedir. Bu nedenle genetik algoritmaların tasarım aşamasında, arama işlemini sona erdirmek için bir "durma ölçütü"nün belirlenmesi gerekmektedir. Durma ölçütü belirli bir çözüm değeri olabileceği gibi algoritmanın belirli sayıda nesil sonunda sona erdirilmesi olarak da tanımlanabilmektedir.

4.1.2. Genetik algoritma operatörleri

İlk olurlu çözümler (başlangıç popülasyonu) türetildikten sonra, genetik operatörlerle, bunlardan daha iyi çözümler türetilmeye çalışılır. Başlıca genetik operatörler, kopyalama, çaprazlama ve mutasyondur. Araştırmacılar, bu operatörlerin çeşitli versiyonlarını geliştirmişlerdir. Geliştirilen algoritmalar, "problem bağımlı" olduğu için, daha farklı versiyonların ve operatörlerin de geliştirilmesi gerekebilmektedir.

4.1.2.1. Kopyalama

Kopyalama işlemi, başlangıç popülasyonu ile aynı sayıda birey içeren ve yardımcı popülasyon (auxiliary population) olarak adlandırılan yeni popülasyonun elde edilmesi amacıyla, bir başlangıç popülasyonunun örneklenmesinden oluşmaktadır. Bu aşamanın amacı popülasyonun geliştirilmesi olup, yüksek uyum fonksiyonuna sahip en iyi bireylere yardımcı popülasyonun üretilmesinde öncelik verilmesi ile sağlanmaktadır (Gomez ve Fuente, 2000).

4.1.2.2. Çaprazlama operatörü

Çaprazlama operatörü en önemli genetik operatördür. Bu operatör aynı anda iki kromozom üzerinde işlem yapar ve her iki kromozomun özelliklerini bir araya getirerek çocuk kromozomlar üretir. Çaprazlama işleminin gerçekleştirilmesinde basit bir yol, bir kesme noktasının belirlenmesi ve ebeveyn kromozomlardan ilkinin kesme noktasının solunda kalan genlerinin diğer ebeveyn kromozomun kesme noktasının sağında kalan genleriyle birleştirilmesi ile çocuk kromozomun üretilmesidir. Bu metot ikili gösterimde daha kullanışlıdır. Genetik algoritmaların performansları büyük ölçüde kullanılan çaprazlama operatörünün probleme uygunluğuna bağlıdır (Gen ve Cheng, 1996).

Çaprazlama operatörü ele alınan problem türünde çok ciddi tahribatlara sebep olmaktadır. Dolayısıyla, bu çalışmada çaprazlama operatörü kullanılmamıştır. Geliştirilen yaklaşım bu yönüyle genetik algoritmalardan farklılık göstermektedir. Genetik algoritmalarda yaygın olarak kullanılan çaprazlama operatörleri ise aşağıdaki sıralanabilir:

- Tek noktalı çaprazlama,
- İki veya çok noktalı çaprazlama,
- PMX çaprazlama (Kısmi eşleştirmeli çaprazlama - partial matching crossover),

- OX çaprazlama (Sıralı çaprazlama - order crossover),
- Sıralamaya dayalı çaprazlama (Order based crossover - OBX),
- Devirli çaprazlama (Cycle crossover - CX),
- Konum-tabanlı çaprazlama (Position-based crossover).

4.1.2.3. Mutasyon operatörü

Çaprazlama işlemi tekrarlandıkça, başarılı genler giderek daha baskın hale gelirler, giderek popülasyonun tamamı tek tip kromozomdan oluşmaya (veya kromozomlar birbirine çok benzemeye) başlar. Bu durumda iki kromozomun çaprazlama işlemi sonucunda farklı bireyler üretilmesi artık mümkün olmaz. Bu senaryodan kaçınmak amacıyla, mutasyon operatörünün kullanılması gerekebilmektedir. Mutasyon operatörü bir kromozomda bulunan genlerden birinin rastgele seçilmesi ve bu genin değiştirilmesi ile yeni bir birey üretmektedir. Daha sonra elde edilen yeni birey popülasyondaki en kötü kromozomla yer değiştirmektedir (Leung vd., 2001). Genetik algoritmalarda kullanılan bazı mutasyon operatörü türleri aşağıda yer almaktadır:

- İkili mutasyon (Binary mutation),
- Ters mutasyonu (Inversion mutation),
- Kaydırma mutasyonu (Shift mutation),
- Bitişik mutasyon (Adjacent mutation).

Genetik algoritmaların yapısı, elemanları ve kullanımını açıklayan bu kesimden sonra, izleyen kesimde dairesel kesme probleminin çözümü için geliştirilmiş GA tabanlı yaklaşım tanıtılacaktır.

4.2. Dairesel Kesme Problemlerinin Çözümü İçin Geliştirilen Yaklaşım

Genetik algoritmalar, diğer kesme problemleri gibi dairesel kesme probleminin çözümü için çok elverişli olmasına karşın, kaynaklarda bu amaçla yapılmış sadece bir çalışmaya rastlanmıştır. Bu çalışma da kurma esaslı bir algoritma önermektedir. Oysa, kurma esaslı algoritmalar genelde -çok iyi olması gerekmeyen- bir başlangıç çözümü oluşturmak amacıyla kullanılmakta, bu başlangıç çözümleri daha sonra iyileştirme algoritmalarına girdi olarak verilip, daha da iyi sonuçlara gidilmektedir. Bu çalışmada ise, genetik algoritma tabanlı bir iyileştirme esaslı algoritmanın oluşturulmasına çalışılmıştır.

İzleyen kesimlerde geliştirilen algoritmada kullanılan kromozom yapısı, uyum fonksiyonu, algoritmanın genel akışı, algoritmada yer alan yordamlar ve yordamların çalışması sırasında ortaya çıkan özel durumlarla bu özel durumların giderilmesi ele alınacaktır.

4.2.1. Kromozomların oluşturulması

Olurlu ve etkin bir GA geliştirmede ilk adım, probleme özgü bir kromozom yapısı tasarımıdır. Tasarlanacak kromozom, problemin olurlu çözümlerinden herhangi birini tam olarak modelleyebilmeli, mümkünse genetik işlemlerin uygulanmasından sonra da bozulmayacak bir yapıda olmalıdır.

Bu çalışmada, ele alınan problemi çözmek için tasarlanmış olan kromozomlar, dairesel şekilli parçaların dikdörtgen levha üzerindeki konumlarını göstermektedir. Parçaların merkezlerine ilişkin koordinat değerleri gerçek sayılar kullanılarak temsil edilmektedir. Her bir kromozomda her parça için x ve y koordinatlarına karşı gelen iki değer bulunmaktadır (Şekil 4.2). Hazırlanan programda sipariş parçalarının levhaya atanma sırası değişmediği için, her bir parçanın kromozom üzerindeki yeri de sabittir (levha üzerindeki yerleri değişebilmektedir).

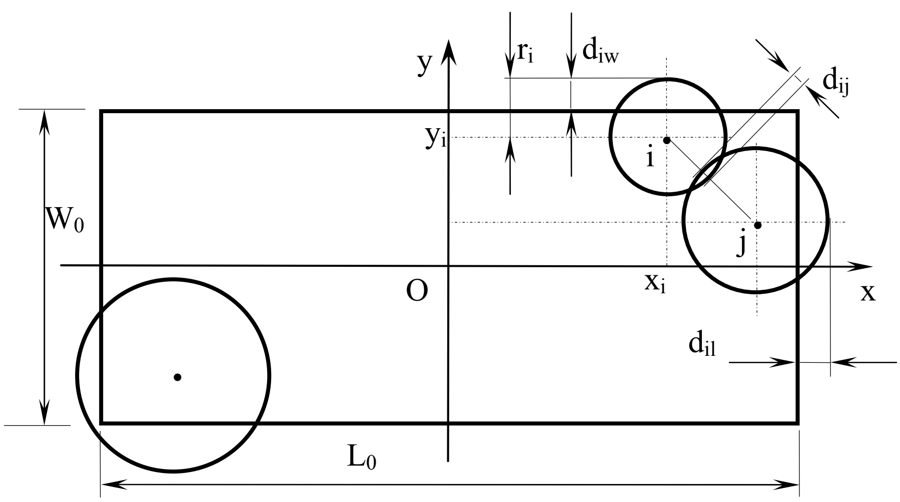
Programın "DEĞİŞTİR" modülünde, yer değişimi işlemi, parçaların dizilişi değiştirilmeksizin ele alınan parça çiftlerinin merkez koordinatlarının (x ve y değerleri) karşılıklı yer değiştirilmesiyle yapılmaktadır. Parçaların yarıçap bilgileri programda ayrı bir yerde tutulmakta, parçaların dizilişi değişmediğinden, bu bilgi de programın çalışması süresince değişmemekte, gerektiğinde alınıp kullanılmaktadır.

1.Parça		2.Parça		3.Parça		…		N.Parça	
x_1	y_1	x_2	y_2	x_3	y_3	…	…	x_n	y_n

Şekil 4.2. Kromozom yapısı

4.2.2. Uyum fonksiyonu

Her kuşakta başarılı bireyleri belirleyip, bunlara yeni kuşağa aktarılma şansı verebilmek için, her şeyden önce başarının tanımlanması ve ölçülebilmesi gerekmektedir. Başarının ölçülüp değerlendirilmesinde kullanılan araç, uyum fonksiyonudur. Geliştirilen yaklaşımda uyum fonksiyonu, sipariş parçalarının kendi aralarında ve ana malzeme ile olan çakışma miktarlarının toplamı olarak tanımlanmıştır. Programın çalışması sırasında bu değerin azaltılmasına çalışılmakta, sıfır yapılması hedeflenmektedir. Şekil 4.3'te gösterildiği üzere, ana malzemenin merkezi (0,0) olarak alınmakta, herhangi bir i parçası da (x_i, y_i) merkez noktası ve, r_i yarıçapı ile temsil edilmektedir.



Şekil 4.3. Ana malzeme, dairesel parçalar, koordinat sistemi ve çakışma durumları

Başlangıçta dairesel parçalar ana malzeme üzerine rastgele yerleştirildiği için olursuz bir çözüm elde edilmektedir. Algoritmanın tekrarlarında bu olursuzluk giderek azalmakta ve en iyi ya da iyice çözüm elde edilmesi durumunda aynı zamanda olurluluk da karşılanmaktadır. En iyi ya da iyice çözümün elde edilebilmesinin ilk koşulu ise uyum fonksiyonu sıfır olan bir kromozomun bulunmasıdır. N parça için olurlu çözüm elde edildikten sonra, sisteme yeni bir dairesel parça daha eklenmekte, bu kez de n+1 parça için yeni bir olurlu çözüm elde edilmeye çalışılmaktadır. Bu ekleme işlemi, sistem duruncaya kadar tekrarlanmakta, böylece olurlu olduğu gibi malzemeden yararlanma oranını da en büyükleyen bir çözüme ulaşılmaktadır.

Burada olurlu terimi, sipariş parçaları ve ana malzeme sınırları arasında çakışma olmadığı için tüm parçaların eksiksiz tam daire şeklinde kesilmesine izin veren kesme planı anlamında kullanılmaktadır. Çakışma olması halinde, kesilecek parçaların uçlarında eksiklikler olması kaçınılmazdır. Şekli bozuk parçalar bir işe yaramayacağı için, bunları türeten kesme planlarından kaçınılmaktadır.

Bir kesme planına ne kadar çok parça sığdırılabilirse, malzeme verimliliği de o kadar artacaktır. Algoritmanın tekrarlanarak çalıştırılması da bu durumu sağlamak içindir.

Popülasyonda yer alan herhangi bir K, K=($x_1, y_1, ..., x_i, y_i, ..., x_n, y_n$), $x_i, y_i \in R$, kromozomuna ait uyum fonksiyonu, parçalar arası çakışmalar ile ana malzeme dışına taşmaların toplamı olarak tanımlanmakta ve:

$$U(K) = \sum_{i=1}^{n}((\sum_{j=i+1}^{n} d_{ij}) + d_{il} + d_{iw}) \tag{17}$$

eşitliği kullanılarak hesaplanmaktadır.

Uyum fonksiyonunda yer alan d_{ij} değişkeni; herhangi i ve j parçası arasındaki çakışma miktarının ölçüsüdür. Çakışma, yarıçapların toplamı ile merkezler arası uzaklığın farkına eşittir. Merkezler arası uzaklık (MAU_{ij}) yarıçapların toplamından küçük değilse, çakışma yoktur.

$$d_{ij} = \begin{Bmatrix} MAU_{ij} < r_i + r_j \quad ise \quad r_i + r_j - MAU_{ij} \\ aksi\ halde \quad 0 \end{Bmatrix} \tag{18}$$

Uyum fonksiyonundaki d_{il} değişkeni de herhangi bir i parçasının ana malzemenin kısa kenarıyla olan çakışma miktarını göstermektedir:

$$d_{il} = \begin{Bmatrix} |x_i| + r_i > \frac{L_0}{2} \quad ise, \quad |x_i| + r_i - \frac{L_0}{2} \\ Aksi \quad halde, \quad 0 \end{Bmatrix} \tag{19}$$

Benzer şekilde herhangi bir i parçasının ana malzemenin uzun kenarıyla olan çakışma mesafesine karşı gelen d_{iw} değişkeni ise:

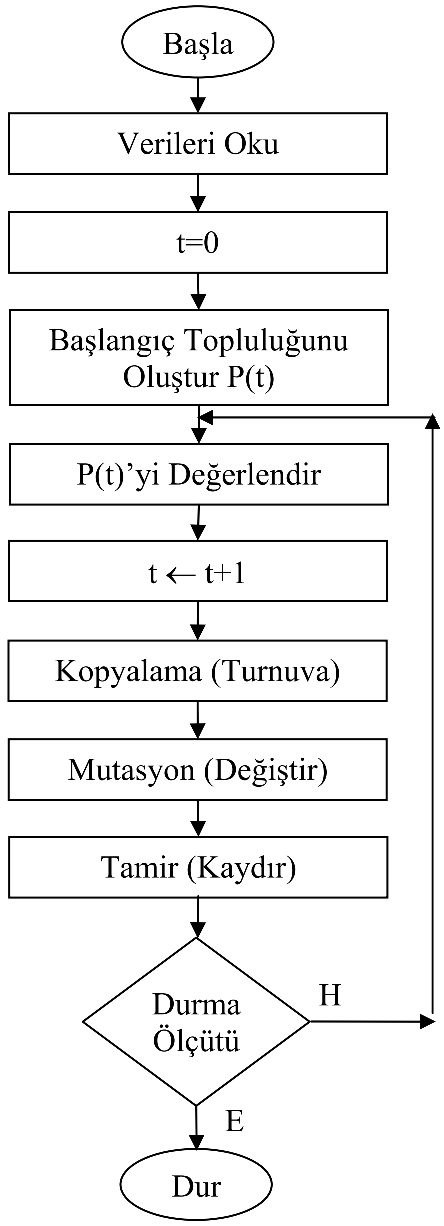
$$d_{iw} = \begin{Bmatrix} |y_i| + r_i > \frac{W_0}{2} \quad ise, \quad |y_i| + r_i - \frac{W_0}{2}, \\ Aksi \quad halde, \quad 0 \end{Bmatrix} \tag{20}$$

formülleri kullanılarak hesaplanmaktadır.

4.2.3. Geliştirilen algoritma

Geliştirilen algoritma temel olarak ana malzemenin tek olduğu ve fireyi en küçükleyecek şekilde sipariş listesinde yer alan malzemelerin ana malzemeden kesilmesi olarak tanımlanan problem türü için geliştirilmiştir. Ancak sipariş listesindeki parçaların sabit boyutlardaki en az sayıda ana malzeme kullanılarak kesilmesi problemine de -küçük bir değişiklikle- uyarlanabilecek bir yapıdadır.

Genel akış şeması Şekil 4.4'te yer alan algoritma, verilerin okunması ile başlamaktadır. Bu veriler esas olarak, ana malzemenin boyutları, dairesel sipariş parçalarının sayısı ve yarıçaplarından oluşmaktadır. Ayrıca popülasyon büyüklüğü, mutasyon oranı, kaydır oranı ve durma ölçütü parametrelerinin de algoritmanın başlangıcında verilmesi gerekmektedir. Genetik operatörlere ilişkin parametrelerin belirlenmesine sonraki kesimlerde değinilecektir.



Şekil 4.4. Geliştirilen algoritmanın genel akış şeması

Ana malzeme üzerine kaç adet sipariş parçasının yerleştirilebileceğinin başlangıçta belirsiz olması nedeniyle, önce ana sipariş listesinden rastgele iki parça alınarak bir "kısmi sipariş listesi" oluşturulmakta ve program çalıştırılarak bu kısmi sipariş listesindeki iki parçanın ana malzemeye olurlu bir şekilde yerleştirmesine çalışılmaktadır. Bunun sağlanması durumunda ana sipariş listesinden rastgele belirlenen üçüncü bir sipariş parçası bu kısmi sipariş listesine eklenmekte ve program tekrar çalışarak üç parça için olurlu bir yerleşim elde edilmeye çalışılmaktadır. Kısmi sipariş listesine, her olurlu çözümden sonra, ana sipariş listesinden bir parça dahil edilerek bu tekrarlı işleme algoritma olursuz çözüm üretene kadar devam edilmektedir. Olursuz çözümden bir önceki çözüm, elde edilen en iyi çözüm olarak kabul edilmektedir.

4.2.4. Genetik operatörler ve tasarlanan modüller

Geliştirilen GA tabanlı yaklaşımda kopyalama işlemi turnuva seçimine dayanmaktadır. Çaprazlama operatörünün ele alınan problem türünde çok ciddi tahribatlara sebep olması nedeniyle, bu çalışmada çaprazlama operatörü kullanılmamıştır. Mutasyon işlemi, DEĞİŞTİR modülü ile yapılmakta, bunu izleyen tamir işlemleri KAYDIR modülü ile olmaktadır. Tüm bu operatörler ve modüller izleyen kısımda ayrıntılı olarak açıklanmaktadır.

4.2.4.1. Mutasyon operatörü (DEĞİŞTİR modülü)

Geliştirilen algoritmada, kesme planlarının oluşturulması işlemi, parçaların atanma sıraları ve merkezlerinin koordinatları olmak üzere iki temel veri kümesine dayandırılmaktadır. Atama sırası ile yarıçap bilgileri arasında birebir ilişki kurulmuştur. Atama sırası değişmeyince, sıraya bağlı olarak tanımlanmış yarıçaplar, kromozomlardaki koordinat bilgileri ile eşleştirilerek kesme planları oluşturulabilmektedir. Böylece iki ayrı veri yapısının eş zamanlı olarak değişmesi ile ortaya çıkacak kaostan kaçınılmış olmaktadır. Bunu sağlamak için de sıralamanın bozulmasına neden olacak çaprazlama türlerinden kaçınılmış, algoritmanın mutasyon ağırlıklı olarak çalışması yaklaşımı benimsenmiştir. Mutasyon operatörünün kullanılması, "DEĞİŞTİR" modülü tarafından yürütülmektedir. İsminden anlaşıldığı üzere bu işlem rastgele seçilen iki parçanın merkez koordinatlarının karşılıklı yer değiştirilmesi ile sağlanmaktadır. Yer değiştirme işlemi her kromozom için sadece bir defa uygulanacak şekilde tasarlanmıştır. Ancak aynı anda birden fazla parça çiftinin yer değiştirmesinin çözüme olan etkisine parametre analizlerinin yer aldığı izleyen kesimlerde ayrıca değinilmektedir. DEĞİŞTİR modülüne ilişkin algoritma aşağıda yer almaktadır.

```
Başla;
i=0;
    tekrarla;
        i=i+1;
        i_sıra_numaralı_kromozomu_seç;
        rastgele_iki_parça_seç;
        seçilen_parçaların_merkezlerini_değiştir;
    tüm_kromozomlar_tamamlanana_kadar;
dur;
```

Aşağıdaki örnekte değiştirme işlemi rastgele belirlenen ikinci ve dördüncü genler arasında gerçekleştirilmektedir.

	1.Parça		2.Parça		3.Parça		4.Parça		5.Parça	
Birey	3	5.2	**0**	**2.1**	-4	2	**-1**	**-2**	5	0.2
Çocuk	3	5.2	**-1**	**-2**	-4	2	**0**	**2.1**	5	0.2

Şekil 4.5. DEĞİŞTİR modülü

4.2.4.2. KAYDIR modülü

GA'ların tasarım sürecinde, kullanılacak kromozomlar yapısal olarak belirlenirken genetik operatörler nedeniyle bir bozulma olmamasına dikkat edilmektedir. Geliştirilen kromozom yapısı, ana malzeme üzerine yerleştirilen dairesel sipariş parçalarının merkez koordinatlarının bir kümesi olarak tanımlanabilir.

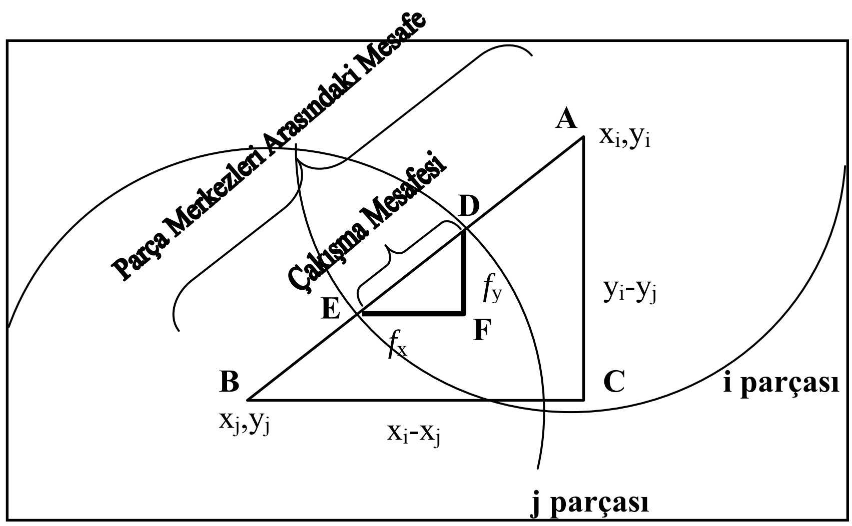
Geliştirilen yaklaşımda, mutasyon işleminden sonra kromozom yapısı bozulmamakla birlikte kromozoma karşı gelen geometrik yapıda bozulma meydana gelmekte, kromozoma ait uyum değeri de aynı etkiye maruz kalmaktadır. Bozulan geometrik yapının tekrar oluşturulması da KAYDIR modülü yardımıyla sağlanmaktadır.

KAYDIR modülü ile, ana malzeme üzerinde bulunan parçalar arasındaki çakışmaların azaltılması amaçlanmaktadır. Bozulmaya uğrayan kromozomlara ait tüm parçalar sırasıyla incelenerek sırası gelen her parçaya, çakışmalar nedeniyle etki eden kuvvetler belirlenmektedir. Gerçekte bunlar fiziki kuvvetler değil, kavramsal kuvvetlerdir. Burada kuvvet terimi, çakışan parçanın hatayı düzeltmek için komşu parça veya ana malzeme sınırını itme eğilimi anlamında kullanılmaktadır. Her parça kendisine etki eden bileşke kuvvet büyüklüğü ve yönünde hareket ettirilerek daha iyi yerleşimler elde edilmektedir. Ancak bu işlem belirli bir tekrar sayısı ile uygulandığında uygun çözümler elde edilmektedir. KAYDIR modülünün kromozomlara hangi tekrar sayısı ile uygulanacağı ise kaydır oranı parametresi ile belirlenmektedir. Bu modülün algoritması aşağıda yer almaktadır:

```
Başla;
i=0;
        tekrarla;
                i=i+1;
                i_nci_parçayı_seç;
                ana_malzemeyle_çakışmaları_hesapla;
                kilit_yönleri_belirle;
                j=0;
                tekrarla;
                        j=j+1;
                        j_nci_parçayı_seç (i≠j);
                        i_ve_j parçaları_arasındaki_mesafeyi_hesapla;
                        mesafe<yarıçap(i)+yarıçap(j) ise_çakışma_yönünü_bul;
                        i_nci_parçaya_etki_eden_bileşke_kuvvetleri_güncelle;
                tüm_parçalar_tamamlanana_kadar;
```

kilit_ yönlere_etki_eden_kuvvetleri_iptal_et;
kalan_kuvvetlerin_bileşke_kuvvetini_bul;
i_nci_parçanın_merkez_koordinatını_bileşke_kuvvet_kadar_değiştir;
değişim_sonucunda_ana_malzeme_ile_çakışmaları_gider;
tüm_parçalar_tamamlanana_kadar;
dur;

KAYDIR modülünde belirlenmesi gereken en önemli parametre parçalara etki eden kuvvetlerin hangi büyüklükte ve hangi yönde olduğudur. Bu parametrelerin belirlenmesi durumunda parçalar bu kuvvetlerin etki ettiği yönde kolaylıkla hareket ettirilebilmektedir. Dairesel parçalara etki eden kuvvetlerin hesaplanma yöntemi Şekil 4.6'da ve izleyen kesimde açıklanmaktadır.



Şekil 4.6. Eksenel kuvvetler

A noktası (x_i, y_i) koordinatlarında yer alan i parçasının merkezini, B noktası ise (x_j, y_j) koordinatlarında yer alan j parçasının merkezini temsil etmektedir. İki parçanın A ve B noktaları arasındaki doğru parçasına karşı gelen merkezleri arasındaki uzaklık (MAU):

$$\text{MAU}_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad \textbf{(21)}$$

İki parçaya ait D ve E noktaları arasındaki doğru parçasına karşı gelen çakışma mesafesi (d_{ij}) ise (18)'de yer alan formül yardımıyla elde edilmektedir.

Bir parçanın diğerini çakışma mesafesi kadar kuvvetle ittiği kabul edilerek, bu kuvvetin eksenel yönlerdeki bileşenleri, f_x ve f_y olarak tanımlanmıştır. Şekil 4.6'da yer alan ABC ve DEF üçgenlerinin benzerliğinden yola çıkılarak,

$$\frac{f_x}{x_i - x_j} = \frac{f_y}{y_i - y_j} = \frac{d_{ij}}{MAU_{ij}} \quad \text{eşitliği yazılabilir.} \tag{22}$$

Yukarıda yer alan eşitlikler kullanılarak f_x kuvveti:

$$f_x = \frac{d_{ij}(x_i - x_j)}{MAU_{ij}} \tag{23}$$

f_y kuvveti ise:

$$f_y = \frac{d_{ij}(y_i - y_j)}{MAU_{ij}} \quad \text{formülleriyle hesaplanmaktadır.} \tag{24}$$

Ana malzeme üzerinde yer alan her parça için f_x ve f_y kuvvetleri belirlenerek, parçaların x ve y koordinatları bu kuvvetler büyüklüğü ve yönünde değiştirilmektedir. Ancak bazı durumlarda yer değişimi sonucunda parçalar ana malzeme dışına çıkabilmektedir. Bu nedenle yer değişimi işleminden sonra yeni koordinatların ana malzeme sınırları içinde olup olmadığı kontrol edilmekte, ana malzeme dışına çıkan parçalara son bir düzeltme işlemi uygulanmaktadır.

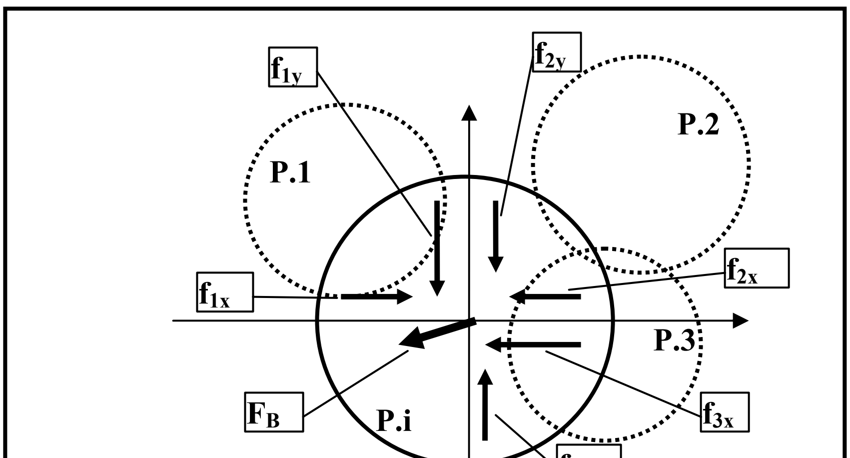
4.2.4.3. KAYDIR modülünde özel durumlar

KAYDIR modülünün yapılandırılması sırasında karşılaşılan bazı özel durumlar izleyen kesimde açıklanmaktadır. Kaynaklarda yer alan GA dışındaki bazı çalışmalarda KAYDIR modülüne benzer yaklaşımlar yer almaktadır. Bilardo toplarının hareketlerinin benzetimi (Graham ve Lubachevsky, 1996), dairelerin birim uzunlukta ve farklı çaplarda esnek silindirler olduğu kabul edilerek bu silindirlerin hareketlerinin benzetimi (Wang vd., 2002) bu çalışmalara örnek olarak gösterilebilir. Ancak, KAYDIR modülü benzeri bu yaklaşımların hiçbirinde karşılaşılan özel durumlara değinilmemektedir.

<u>Özel Durum-1:</u>

KAYDIR modülünde her parça sırasıyla ele alınarak bu parçanın diğer parçalarla ve ana malzeme ile çakışma mesafeleri belirlenmektedir. Çakışmaların azaltılması amacıyla, seçilen parça -diğer tüm parçalarla olan çakışma durumlarının hesaplanması sonucunda- kendisine etki eden kuvvetlerin bileşkesi büyüklüğünde ve yönünde hareket ettirilmektedir. Ancak bir parçayı aynı yönde iten birden fazla parça olması durumunda, ele alınan parça kendisini iten aynı yöndeki kuvvetlerin en büyüğü kullanılarak yer değiştirdiğinde daha küçük kuvvetlerin etkisinden de kurtulmuş olacaktır. Bu nedenle bileşke kuvvet hesaplanırken her yöndeki en büyük itme kuvveti kullanılmaktadır.

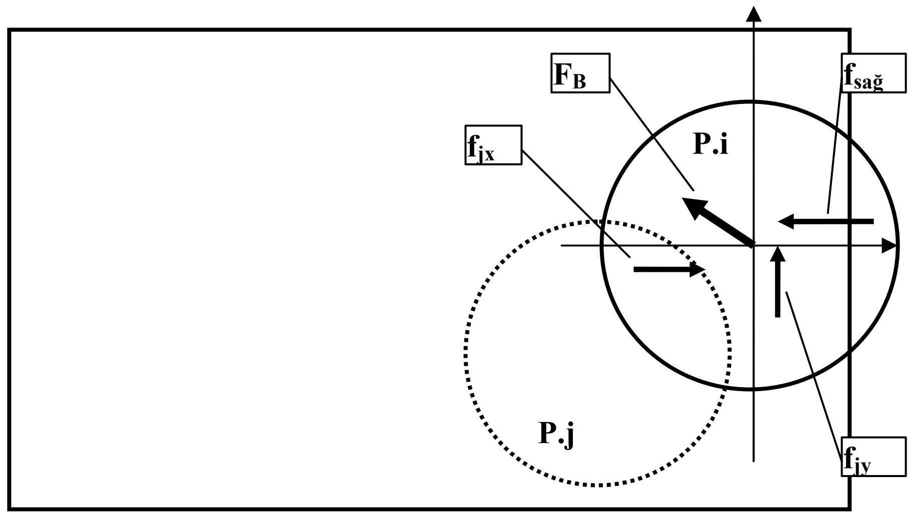
Şekil 4.7'de verildiği gibi seçilen i parçası sadece üç parçayla çakışıyor olsun. Öncelikle bir, iki ve üç numaralı parçaların i parçası ile olan çakışmaları sonucunda oluşan itme kuvvetlerinin f_x ve f_y yönlerinde bileşenleri belirlenir. Aynı doğrultuda yer alan kuvvetlerden en büyük olanlar (f_{1y}, f_{3x}, f_{3y} ve f_{1x}) seçilerek diğer kuvvetler (f_{2y} ve f_{2x}) iptal edilir. Kalan kuvvetlerin bileşkesi (F_B) ise i parçasının hareket edeceği yön ve miktarı göstermektedir.



Şekil 4.7. Özel durum-1

Özel Durum-2:

Programda, sipariş parçalarının ana malzeme sınırları içinde kalması diğer çakışmalara göre daha öncelikli olarak düşünülmüştür. Örneğin, eğer i parçası ana malzeme ile çakışıyor ise (Şekil 4.8), bu durumda ana malzeme ile çakışma yönündeki kuvvete ($f_{Sağ}$) ters yöndeki kuvvetler (f_{jx}) iptal edilmektedir. Böylece gereksiz hareketlerden kaçınılarak uygun çözümler daha kısa sürede elde edilmektedir.



Şekil 4.8. Özel durum-2

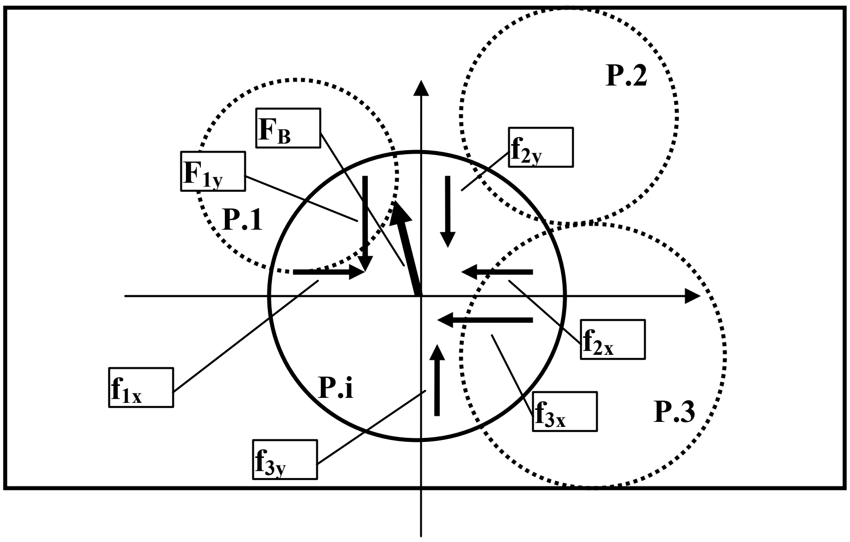
Özel Durum-3:

Programın çalıştırılması sırasında herhangi iki sipariş parçasına ait merkez koordinatları aynı değerleri alabilmektedir ($x_i = x_j$ ve $y_i = y_j$). Bu durumda ise problem ortaya çıkmaktadır. Eksenel kuvvetlerin hesaplanmasında kullanılan (23) ve (24)'de yer alan formüller incelendiğinde "merkezler arası uzaklığın" formüllerde payda kısmında yer aldığı görülmektedir.

Merkezler arası uzaklık ise (21)'de yer alan formül yardımıyla elde edilmektedir. İki parçanın aynı koordinatlarda olması durumunda (23) ve (24)'de yer alan formüllerin payda kısmı sıfır olmaktadır. Sıfıra bölünmeden dolayı ise sonsuz değeri elde edilmesi nedeniyle program hata vermektedir. Bu senaryonun önlenmesi amacıyla bu işlemden önce merkezlerin aynı koordinatlarda olup olmadığı kontrol edilerek eğer aynı koordinatlarda iseler karşı gelen parçaların koordinatlarında küçük bir değişiklik yapılmasıyla sorun giderilmektedir.

Özel Durum-4:

Herhangi bir i parçasına ait çakışmaların hesaplanması sırasında, eğer i parçası ile çakışan parçalar arasında ana malzeme ile komşu olan bir parça mevcut ise (Şekil 4.9), bu durumda, i parçasını bu yöne iten kuvvetler (f_{1y} ve f_{2y}) bileşke kuvvet hesaplanırken dikkate alınmamaktadır. Böylece sipariş parçalarının gereksiz hareketleri önlenerek çözüm süresi kısaltılmaktadır.



Şekil 4.9. Özel durum-4

4.2.5. Yeni neslin oluşturulması

Mutasyon işleminden sonra kromozomlar KAYDIR modülüne alınıp tamir edilmekte, durma ölçütü sağlanmadıysa (uyum fonksiyonunun değeri sıfıra düşmediyse), yeni bir kuşak daha türetilerek algoritmanın çalışması sürdürülmektedir. Yeni neslin oluşturulması amacıyla önce en iyi çözüm kaydedilmekte, daha sonra da popülasyon büyüklüğü sayısında kromozom, turnuva yöntemiyle seçilerek yeni nesle aktarılmaktadır. Son olarak da elde tutulan en iyi çözüm, uyum değeri en düşük olan kromozomun üzerine kopyalanmaktadır. Yeni neslin oluşturulmasına ilişkin algoritma adımları aşağıda verilmektedir.

Başla;

bireyleri başarı sırasına sok, en iyisini “seçkin” olarak belirle;

ilk s kromozomu S kümesi olarak tanımla;

i=0;

tekrarla;

i=i+1;

S kümesinden rastgele bir kromozom seç ve i. bireyin yerine kopyala;

Popülasyon_sayısına_ulaşılıncaya_kadar;

seçkin kromozomu, en kötü kromozom üzerine kopyala:

dur;

4.3. Parametrelerin Belirlenmesi

Parametrelerin belirlenmesi amacıyla, Hifi ve M’Hallah’ın, (2004) çalışmasında yer alan ve en iyi çözümü bilinen altı problemden diğerlerine göre çözümü en zor olan, birinci problem kullanılmıştır. Söz konusu problemde 30 sipariş parçasının 17.491 ve 9.5 birim ölçülerindeki dikdörtgen bir levhaya yerleşiminin en iyi çözümü araştırılmaktadır (Parça yarıçapları EK-1’de yer almaktadır). Diğerleri arasından özellikle bunun seçilmesinin nedeni, altı problem içinde en düşük fire oranına (% 16.86) sahip olması ve yazarlar tarafından geliştirilen yöntemle çözümünde en iyi çözümden çok farklı bir sonuç elde edilmesidir.

Geliştirilen genetik algoritma tabanlı yaklaşımın kullanılabilmesi için popülasyon büyüklüğü, turnuva büyüklüğü, mutasyon oranı ve kaydır oranı parametrelerinin belirlenmesi gerekmektedir. Mutasyon operatörü olan DEĞİŞTİR modülü, rastgele belirlenen parça çiftlerinin yerlerini değiştirmektedir. Programın geliştirilmesi sırasında her kromozom için sadece bir parça çiftinin yer değiştirmesi planlanmış ve başarılı sonuçlar elde edilmiştir. Bu kesimde ise, bir ve iki parça çiftinin yer değiştirilmesi durumlarının çözüme olan katkıları incelenmiştir.

Kaydır oranı parametresi, tamir operatörü olarak görev yapan KAYDIR modülünün kromozomlara kaç defa uygulanacağını göstermektedir. Program süresince çözümlerin olursuz olması ve KAYDIR modülünün ise olursuzluğu azaltan bir işlem olması nedeniyle bu modül tekrarlı bir yapıda tasarlanmıştır.

Her bir deney için program onar kez çalıştırılmış ve elde edilen çözümlerin ortalama değerleri Çizelge 4.1'de verilmiştir (çözümlere ilişkin tüm veriler EK-2'de sunulmaktadır). En iyi çözümün biliniyor olması nedeniyle, parçalar ana malzemeye büyükten küçüğe sıralı olacak şekilde yerleştirilmiştir. İlk aşamada popülasyon büyüklüğü: 10 ve turnuva büyüklüğü: 6 kabul edilerek geliştirilen yaklaşım, mutasyon ve kaydır oranı parametrelerinin farklı değerleri için test edilmiş, daha sonra belirlenen en iyi mutasyon ve kaydır oranları kullanılarak popülasyon ve turnuva büyüklüğü parametrelerinin farklı değerlerinin çözüme etkisi incelenmiştir.

Çizelge 4.1 incelendiğinde; mutasyon oranı: 1 ve kaydır oranı: 10 olduğu durumda iyi çözümlerin kısa sürede elde edildiği görülmektedir. Bu ise KAYDIR modülünün kromozomlara az tekrarla uygulanması nedeniyle erken yakınsamanın önlenmesinden kaynaklanmaktadır.

Yapılan deneyler sonunda mutasyon oranındaki artışın çözümleri iyileştirmediği belirlenmiştir. Bu ise çok fazla parça çiftinin yer değiştirmesinin çözümlerin bozulmasına neden olmasından kaynaklanmaktadır. Mutasyon oranındaki bu bozulma kaydır oranının artırılması ile -az da olsa- azalmaktadır.

Kaydır oranı parametresi, çözümlere büyük oranda etki etmektedir. Bu parametre, kromozomlara KAYDIR modülünün kaç kez uygulanacağını belirlemektedir. DEĞİŞTİR modülü ile rastgele seçilen iki parça yer değiştirmekte, KAYDIR modülü ise bu değişiklik nedeniyle oluşan yerleşim bozukluklarını azaltmaktadır. Bu nedenle yeri değiştirilen parçaların özellikleri KAYDIR modülünün tekrar sayısına etki etmektedir. Eğer yer değiştiren iki parça aynı yarıçaplarda ise yerleşimde bir bozulma olmamaktadır. Parçaların çok farklı boyutlarda olması durumunda bozulmanın daha fazla olması nedeniyle kaydırma işleminin daha fazla tekrarla yapılması daha iyi sonuçlar vermektedir. Bu nedenle, kaydır oranı parametresinin kullanıcı girdisine ihtiyaç duyulmadan program tarafından dinamik olarak hesaplanması amacıyla aşağıda yer alan denklem geliştirilmiştir. Bu denklemde parçaların birbirleri ile olan farklılıklarına göre kaydır oranı parametresi 0 ile 30 arasında farklı değerler almaktadır. Parçalar aynı yarıçaplarda ise kaydır oranı parametresi sıfır değerini, çok farklı olması durumunda ise 30 değerini almaktadır.

$$\text{Kaydır Oranı} = \text{Tam\_Değer} \left[\frac{|r_i - r_j|}{r_i + r_j} \text{ x } 30 \right) \qquad \textbf{(25)}$$

Bu yaklaşım kullanılarak farklı mutasyon oranları için (7 ve 8 nci deneyler) program tekrar çalıştırılmış ve uygun çözümler elde edildiği belirlenmiştir.

Yapılan deneyler sonucunda, mutasyon oranı: 1 ve kaydır oranı: Otomatik olduğu durumda en iyi çözümlerin elde edildiği belirlenmiş ve bu değerler kullanılarak popülasyon büyüklüğü (10, 20 ve 30) ve turnuva büyüklüğü (popülasyon büyüklüğünün 0.2, 0.4 ve 0.6 katları olarak) parametreleri farklı değerleri için incelenmiştir.

Çizelge 4.1. Parametre analizi deney sonuçları (Ortalamalar)

Deney No	Popülasyon Büyüklüğü	Turnuva Büyüklüğü	Mutasyon Oranı	Kaydır Oranı	Süre [dak]	Fire Oranı
1	10	6	1	10	02:22	0.1739
2	10	6	1	20	04:50	0.1768
3	10	6	1	30	06:40	0.1747
4	10	6	2	10	02:40	0.1819
5	10	6	2	20	04:50	0.1812
6	10	6	2	30	07:17	0.1798
7	10	6	1	Otomatik	03:20	0.1751
8	10	6	2	Otomatik	05:26	0.1788
9	10	2	1	Otomatik	03:42	0.1735
10	10	4	1	Otomatik	03:36	0.1752
11	10	6	1	Otomatik	03:20	0.1751
12	20	4	1	Otomatik	06:19	0.1730
13	20	8	1	Otomatik	05:38	0.1746
14	20	12	1	Otomatik	05:48	0.1747
15	30	6	1	Otomatik	07:23	0.1713
16	30	12	1	Otomatik	08:39	0.1729
17	30	18	1	Otomatik	09:16	0.1794

Çizelge 4.1 incelendiğinde 12 ve 15 nci deney sonuçlarının birbirine çok yakın olduğu ve çok başarılı çözümler elde edildiği görülmektedir. 12 ve 15 nci deneylerdeki çözümlerin birbirine çok yakın olması nedeniyle -hesaplama süresi de göz önünde bulundurularak- popülasyon büyüklüğü: 20 ve turnuva büyüklüğü: 4 olarak belirlenmiştir.

4.4. Geliştirilen Yaklaşımın Sınanması

Bu çalışma sonunda geliştirilmiş olan GA tabanlı yaklaşımın sınanması, önce rastgele türetilmiş verilerle yapılmış, çözümün farklı durumlardaki davranış biçimi irdelendikten sonra, kaynaklardaki problemlerle karşılaştırılıp yeni yöntemin kuramsal olarak karşılaştırılması yapılmış, en sonunda da gerçek hayat problemleri ele alınarak işyerinde kabul gören çözümlerle yapılan karşılaştırma sonucunda pratik açıdan bir değerlendirme yapılmaya çalışılmıştır.

4.4.1. Rastgele üretilen problemlerin çözümü

Rastgele problemlerin üretilmesi amacıyla Hifi vd., (2004) çalışmalarında yer alan yöntem kullanılmıştır. Geliştirilen yöntemle üç grup örnek kümesi oluşturulmuştur (G1, G2 ve G3). Her grup beş farklı problemden oluşmaktadır. Her grup için sipariş parçalarının sayısı sırasıyla [20, 30], [30, 70] ve [40, 100] aralıklarından tamsayı olacak şekilde rastgele seçilmektedir. Ana malzemenin boyutları (L_0,W_0), G1 kümesi için [50, 100], G2 kümesi için [100, 150] ve G3 kümesi için [150, 250] aralıklarından rastgele alınmıştır. Her parçanın yarıçapı, $r_i = \gamma_l$ enk $\{L_0,W_0\}$ formülüyle hesaplanmaktadır. Burada yer alan γ_l, [0.05, 0.13] aralığından rastgele seçilmektedir. Her sipariş parçası için talep miktarı ise [1, 10] aralığından rastgele seçilmektedir. Üretilen test problemleri EK-3'te verilmiştir.

Geliştirilen GA tabanlı yaklaşımın rastgele üretilen problemlerde test edilmesi ile elde edilen sonuçlar Çizelge 4.2'de yer almaktadır.

Çizelge 4.2. Geliştirilen yaklaşımın rastgele üretilen problemlerde denenmesi

Problem No	Parça Sayısı	Levha Eni [cm]	Levha Boyu [cm]	Süre [dak.]	Fire Oranı
1-1	28	92.12	97.14	10:20	0.2059
1-2	26	77.78	89.94	05:58	0.2097
1-3	23	60.70	73.84	07:58	0.2027
1-4	23	78.17	94.99	07:48	0.2094
1-5	24	65.91	69.55	05:25	0.1987
2-1	40	101.01	137.44	10:07	0.2178
2-2	62	148.78	149.66	16:39	0.2063
2-3	35	147.28	148.53	09:23	0.2040
2-4	53	133.16	137.46	11:21	0.2116
2-5	61	126.99	141.14	16:28	0.1951
3-1	86	195.88	219.06	24:26	0.1994
3-2	89	204.51	244.62	30:20	0.2010
3-3	92	154.51	230.03	39:11	0.2027
3-4	99	196.05	246.14	24:37	0.1940
3-5	57	211.49	242.22	17:35	0.1942

Çizelge 4.2'de yer alan sonuçlar incelendiğinde, önerilen yaklaşımın tek ana malzeme olduğu durumlar (knapsack) göz önüne alınarak geliştirilmiş olmasına karşın, aynı boyutlarda birden çok ana malzeme olduğu durumlar (bin packing) için de kabul edilebilir fire oranlarının elde edildiği açıkça görülmektedir. Parça sayısının büyük olduğu üçüncü grup problemlerde çözüm süresinin arttığı gözlenmiştir. Ancak, algoritmanın daha hızlı çalışan yüksek programlama dilleriyle kodlanması durumunda, küçük işletmeler için geliştirilen bu yaklaşımın büyük hacimli problemleri daha kısa sürede çözebileceği düşünülmektedir.

4.4.2. Geliştirilen yaklaşımın kaynaklardaki yöntemlerle karşılaştırılması

Geliştirilen yaklaşım, Hifi ve M'Hallah'ın, (2004) çalışmasında yer alan altı problem üzerinde denenmiştir. Her problem için program onar kez çalıştırılmış ve elde edilen sonuçlar Çizelge 4.3'te, bu sonuçların kaynaklarda bulunan çözüm değerleriyle karşılaştırılması da Çizelge 4.4'te sunulmuştur.

Çizelge 4.3. Geliştirilen yaklaşımın kaynaklardan alınan problemlerde denenmesi

Problem No	Parça Sayısı	Levha Eni	Levha Boyu	Deney No	Süre (dak.)	Fire Oranı
1	30	9.500	17.491	1	06:25	0.1738
				2	06:12	**0.1686**
				3	06:22	**0.1686**
				4	06:20	0.1738
				5	06:21	**0.1686**
				6	06:17	**0.1686**
				7	06:22	0.1798
				8	06:21	0.1798
				9	06:19	**0.1686**
				10	06:15	0.1798
				Ortalama:	06:19	0.1730
2	20	8.500	14.895	1	05:40	0.1920
				2	00:48	**0.1840**
				3	01:59	**0.1840**
				4	05:30	0.1920
				5	01:44	**0.1840**
				6	05:55	0.1920
				7	05:54	0.1920
				8	05:50	0.1920
				9	01:18	**0.1840**
				10	05:35	0.1920
				Ortalama:	04:01	0.1888
3	25	9.000	14.93	1	01:48	0.1810
				2	03:29	0.1810
				3	03:06	0.1810
				4	07:12	0.1810
				5	02:48	0.1810
				6	02:27	0.1810
				7	03:45	0.1810
				8	01:17	0.1810
				9	08:25	0.1871
				10	04:42	0.1810
				Ortalama:	03:53	0.1816
4	35	11.000	24.355	1	06:56	**0.1830**
				2	08:12	**0.1830**
				3	05:20	**0.1830**
				4	03:25	**0.1830**
				5	10:22	0.1873
				6	10:30	0.1873
				7	04:49	**0.1830**
				8	03:06	**0.1830**

Problem No	Parça Sayısı	Levha Eni	Levha Boyu	Deney No	Süre (dak.)	Fire Oranı
				9	09:12	**0.1830**
				10	10:34	0.1873
				Ortalama:	07:14	0.1842
5	100	15	38.046	1	13:28	0.1778
				2	35:27	0.1778
				3	51:18	0.1778
				4	24:14	0.1778
				5	29:28	0.1778
				6	16:12	0.1778
				7	24:32	0.1778
				8	35:11	0.1778
				9	18:43	0.1778
				10	46:21	0.1778
				Ortalama:	29:33	0.1778
6	100	19.0	38.647	1	24:07	0.1775
				2	21:44	0.1775
				3	29:23	0.1775
				4	24:34	0.1775
				5	29:01	0.1775
				6	26:12	0.1775
				7	23:25	0.1775
				8	26:12	0.1775
				9	25:45	0.1775
				10	21:26	0.1775
				Ortalama:	25:02	0.1775

Yapılan deneyler sonucunda parça sayısının sırasıyla 30, 20, 25 ve 35 olduğu bir ila dördüncü problemlerde kısa sürede uygun çözümler elde edildiği, parça sayısının 100 olduğu beş ve altıncı problemde ise, tüm deneylerde en iyi çözümün bulunmasına karşın, hesaplama süresinin nispeten daha uzun sürdüğü görülmüştür.

Çizelge 4.4 incelendiğinde kaynaklarda yer alan yönteme oranla geliştirilen yaklaşım ile daha iyi sonuçlar elde edildiği açıkça görülmektedir (ortalama olarak (0.1871 - 0.1787) / 0.1871 = % 4.5). bir, iki ve dördüncü problemlerin kaynaklarda bulunan yöntemle çözümü sonucunda en iyi çözümler bulunamamasına karşın, geliştirilen GA tabalı yaklaşımla, bu problemlerin bilinen en iyi çözümlerine ulaşılmıştır.

Çizelge 4.4 Sonuçların kaynaklardaki fire oranlarıyla karşılaştırılması

Problem No	Problemin En İyi Çözümü	Literatüre Bulunan En İyi Fire Oranları	GA ile Bulunan En İyi Fire oranları	GA ile Bulunan Ortalama Fire oranları	10 Deneyde En İyi Çözümün Bulunma Sayısı
1	0.1686	0.1904	**0.1686**	**0.1730**	5
2	0.1840	0.2015	**0.1840**	**0.1888**	4
3	0.1810	0.1810	0.1810	0.1816	9
4	0.1830	0.1945	**0.1830**	**0.1842**	7
5	0.1778	0.1778	0.1778	0.1778	10
6	0.1775	0.1775	0.1775	0.1775	10
Ortalama	0.1787	0.1871	0.1787	0.1804	7.5

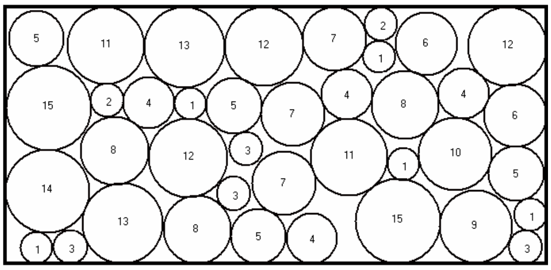
4.4.3. Geliştirilen yaklaşımın gerçek hayat problemlerine uygulanması

Geliştirilen GA tabanlı yaklaşımın gerçek hayat problemlerine uygulanması havacılık sektöründe faaliyet gösteren bir firmadan alınan veriler kullanılarak yapılmıştır. Ana malzeme 255 ve 122 birim ölçülerinde olup, on beş sipariş parçasına ait talep miktarları ise Çizelge 4.5'te yer almaktadır.

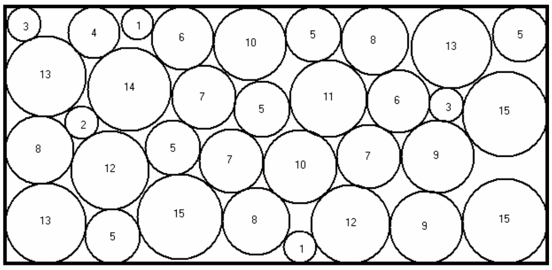
Çizelge 4.5. Gerçek hayat problemi verileri

Parça No	Yarıçap	Talep	Parça No	Yarıçap	Talep
1	7.6	30	9	17.1	35
2	7.8	15	10	17.2	35
3	8.0	25	11	18.1	15
4	12.0	40	12	18.5	11
5	13.0	50	13	19.0	13
6	14.7	32	14	19.7	12
7	15.0	25	15	20.0	9
8	16.0	25			

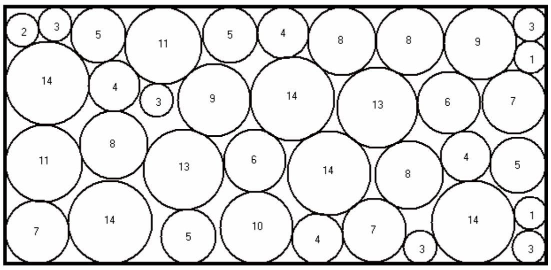
İşyeri şartlarında, EK-4'te verilen çözüm ve 0.26 fire oranı kabul edilebilir görünürken, GA ile bulunan (Şekil 4.10) çözüm bu oranı 0.19'a düşürmüş ve fire oranında (0.26-0.19)/0.26 x 100=% 26.9'luk bir iyileşme sağlandığı görülmüştür.



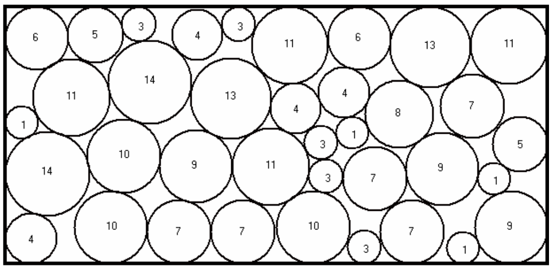
Kesme Planı-1 (Adet:3, Fire Oranı:0.1951)



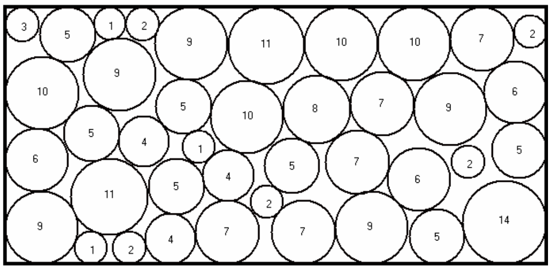
Kesme Planı-2 (Adet:1, Fire Oranı:0.1896)



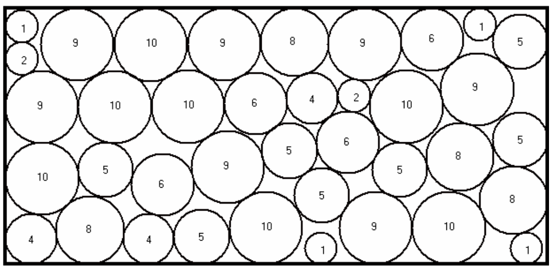
Kesme Planı-3 (Adet:1, Fire Oranı:0.1846)



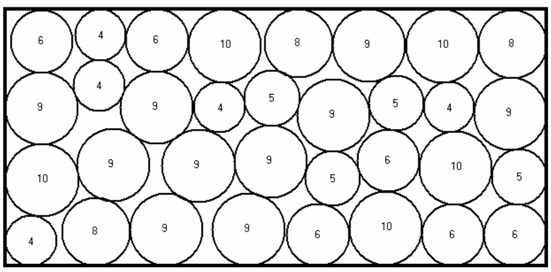
Kesme Planı-4 (Adet:1, Fire Oranı:0.2075)



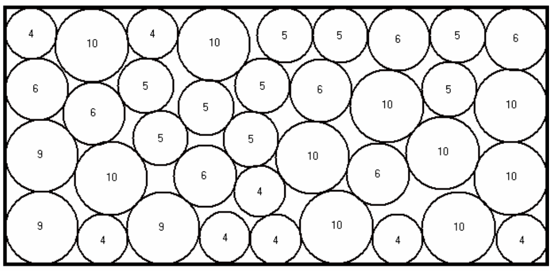
Kesme Planı-5 (Adet:1, Fire Oranı:0.2052)



Kesme Planı-6 (Adet:1, Fire Oranı:0.1951)



Kesme Planı-7 (Adet:1, Fire Oranı:0.2062)



Kesme Planı-8 (Adet:1, Fire Oranı:0.1903)

Şekil 4.10. Geliştirilen yaklaşımın gerçek hayat problemine bulduğu çözümler

5. SONUÇ VE ÖNERİLER

Genetik algoritmaların, kombinatorik problemlerin çözümünde çok güçlü bir araç olarak tanınması nedeniyle, bu çalışmada dairesel kesme probleminin çözümü için GA tabanlı bir yaklaşımın geliştirilmesi hedeflenmiştir.

Kaynak taraması sonucunda, uygulamadaki yaygın kullanıma karşın, dairesel kesme alanındaki çalışmaların kesme problemi konusundaki yüzlerce yayın arasında çok küçük bir yer tuttuğu gözlenmiştir. Dairesel kesme probleminin çözümünde genetik algoritmaya dayanan sadece bir çalışmaya ulaşılabilmiş, bu problemlerin çözümünde genetik algoritmalardan yeterince yararlanılmadığı saptanmıştır.

Genetik algoritma tabanlı yeni yaklaşımda; kromozomlar, parçaların dikdörtgen levha üzerindeki konumlarını gösterecek şekilde yapılandırılmış, uyum fonksiyonu ise, parçaların kendi aralarında ve ana malzeme ile olan çakışma miktarlarının toplamı olarak tanımlanmıştır. Çaprazlama operatörünün bu tür problemlerde çok ciddi tahribatlara sebep olması nedeniyle, bu tez çalışmasında geliştirilen genetik algoritma tabanlı yeni yaklaşımda çaprazlama operatörü kullanılmamıştır. Mutasyon ve tamir işlemlerinin gerçekleştirilmesi amacıyla DEĞİŞTİR ve KAYDIR modülleri oluşturulmuştur. Mutasyon işlemi, DEĞİŞTİR modülü ile (rastgele seçilen iki parçanın yer değiştirmesi) yapılmış; tamir işlemi ise KAYDIR modülü kullanılarak gerçekleştirilmiştir. KAYDIR modülü ile amaç; çakışan parçaların birbirlerini iterek çözüm uzayının farklı bölgelerinin de taranmasıdır.

Erişilebilen kaynaklarda, KAYDIR modülündekine benzer yaklaşımlar kullanan araştırmacıların hiç birinin, burada karşılaşılabilecek özel durumları dikkate almadığı gözlenmiştir. Bu çalışmanın önemli kuramsal katkılarından biri de bu özel durumlara dikkati çekmek ve bunların olumsuz etkilerinin giderilmesi için çözüm teknikleri geliştirmek olarak belirtilebilir.

Geliştirilen GA tabanlı yaklaşımın; popülasyon büyüklüğü, turnuva büyüklüğü, mutasyon oranı ve kaydır oranı parametrelerinin belirlenmesinde Hifi ve M'Hallah'ın, (2004) çalışmasında yer alan birinci problemden yararlanılmıştır. Öncelikle popülasyon büyüklüğü ve turnuva büyüklüğü parametreleri sabit tutularak farklı mutasyon ve kaydır oranları için bir dizi deney yapılmıştır. Yapılan deneyler sonunda mutasyon oranının 1 olduğu durumda (her kromozom için bir parça çiftinin yer değiştirmesi) en iyi çözümlerin elde edildiği belirlenmiştir.

Deney sonuçları, kaydır oranı parametresinin farklı değerlerinin çözümlere büyük oranda etki ettiğini göstermiştir (Bu parametre, kromozomlara yerleşim bozukluklarını azaltan KAYDIR modülünün kaç kez uygulanacağını belirlemektedir). Tekrar sayısının parça boyutlarından etkilendiği düşünülerek, bu parametrenin değişen çaplara bağlı olarak "çevrim içi" belirlenmesini sağlayacak bir kural geliştirilmiş ve yapılan testler sonucunda bu dinamik çözümün kayda değer iyileştirmeler sağladığı görülmüştür. Kaydır oranını kullanıcı girdisine ihtiyaç duyulmadan gerçek zamanlı olarak hesaplayan bu "uyarlama stratejisi" de (adaptive strategy) bu çalışmanın sağladığı bir diğer kuramsal katkı olarak kaydedilebilir.

Mutasyon oranı ve kaydır oranı parametrelerinin belirlenmesinden sonra, bu parametreler sabit alınarak farklı popülasyon büyüklüğü ve turnuva büyüklüğü değerleri için bir dizi deney daha yapılmış ve popülasyon büyüklüğünün 20, turnuva büyüklüğünün de 4 olduğu durumda en iyi çözümlerin elde edildiği görülmüştür.

Parametrelerin belirlenmesinden sonra, geliştirilen yaklaşımın sınanması, önce rastgele türetilmiş verilerle yapılmış, çözümün farklı durumlardaki davranış biçimi irdelendikten sonra, kaynaklardaki problemlerle karşılaştırılıp yeni yöntemin kuramsal olarak karşılaştırılması yapılmış, en sonunda da gerçek hayat problemleri ele alınarak işyerinde kabul gören çözümlerle yapılan karşılaştırma sonucunda pratik açıdan bir değerlendirme yapılmaya çalışılmıştır. Elde edilen sonuçlar incelendiğinde, önerilen yaklaşımın kabul edilebilir sürelerde yüksek kalitede çözümler ürettiği ve kaynaklarda yer alan çözümlerden daha iyi sonuçlar elde edildiği belirlenmiştir.

Yapılan çalışma ile sağlanan iyileştirmeler aşağıdaki şekilde sıralanabilir:

- KAYDIR modülünde; karşılaşılabilecek özel durumların belirlenmesi, bunların olumsuz etkilerinin giderilmesi için çözüm tekniklerinin geliştirilmesi (bölüm 4.2.4.3.'deki özel durum 1, 2, 3 ve 4).
- Kaydır oranının kullanıcı girdisine ihtiyaç duyulmadan program tarafından dinamik olarak hesaplanması amacıyla bir denklem geliştirilmesi (eşitlik 25).
- Geliştirilen yöntemle, kaynaklardan alınan çözümlerden daha iyi sonuçlar elde edilmesi (ortalama % 4.5 iyileşme).
- Geliştirilen yöntemle, gerçek hayat problemleri ele alınarak; işyerinde kabul gören çözümlerden çok daha iyi sonuçlar elde edilmesi (% 26.9 iyileşme).

Dairesel kesme problemleri ile ilgili, gelecekte yapılacak çalışmaların yönünü tayin edebilecek özellikteki bazı öneriler aşağıda yer almaktadır:

- Geliştirilen yaklaşım, dikdörtgen şeklindeki sipariş parçalarını da içerecek şekilde genişletilebilir, ana malzemelerin farklı boyutlarda olduğu durumlar da göz önünde bulundurulabilir.
- Sipariş listesinden parçaların seçim sırasını belirlemek için ayrı bir yordam geliştirilebilir.
- Farklı öncelik kurallarının kullanılmasıyla algoritmanın güçlendirilmesi düşünülebilir.
- Çakıştığı parça sayısı en büyük olan parçaların ana malzeme üzerinde en boş yere yerleştirilmesi benzeri stratejiler algoritmaya dahil edilebilir.

- Oluşturulan kesme planlarından kullanıcının daha rahat yararlanabilmesi için geliştirilmiş olan GA tabanlı yaklaşımın AutoCAD vb. programlarla ilişkilendirilmesinin faydalı olacağı düşünülmektedir.
- Geliştirilen yaklaşım ile elde edilen kesme planlarının, geliştirilecek bir tamsayılı modelin parametrelerini üretmek amacıyla kullanılması durumunda; daha iyi sonuçlar elde edilebileceği değerlendirilmektedir.

KAYNAKLAR DİZİNİ

Amaral, A.R.S. ve Wright, M., 2001, Experiments with a strategic oscillation algorithm for the pallet loading problem, *International Journal of Production Research,* 39, 11, 2341-2351.

Babu, A.R. ve Babu, N.R., 1999, Effective nesting of rectangular parts in multiple rectangular sheets using genetic and heuristic algorithms, *International Journal of Production Research,* 37, 7, 1625-1643.

Beasley, J.E., 2004, A population heuristic for constrained two-dimensional non-guillotine cutting, *European Journal of Operational Research,* 156, 601-627.

Benati, S., 1997, An algorithm for a cutting stock problem on a strip, *Journal of the Operational Research Society,* 48, 288-294.

Birgin, E.B., Martinez, J.M. ve Ronconi, D.P., 2005, Optimizing the packing of cylinders into a rectangular container: A nonlinear approach, *European Journal of Operational Research,* 160, 19-33.

Boschetti, M.A, Mingozzi, A. ve Hadjiconstantinou, E., 2002, New upper bounds for the two-dimensional orthogonal non-guillotine cutting stock problems, *Journal of Management Mathematics,* 13, 95-119.

Chauny, F., Loulou, R., Sadones, S. ve Soumis, F., 1991, A two-phase heuristic for the two-dimensional cutting-stock problem, *Journal of Operational Research Society,* 42, 1, 39-47.

Correia, M.H., Oliveira, J.F. ve Ferreira, J.S., 2000, Cylinder packing by simulated annealing, *Pesquisa Operacional,* 20, 2, 269-286.

Correia, M.H, Oliveira, J.F. ve Ferreira, J.S., 2001, A new upper bound for the cylinder packing problem, *International Transactions in Operational Research,* 8, 571-583.

Cui, Y., 2005, A cutting stock problem and its solution in the manufacturing industry of large electric generators, *Computers & Operations Research,* 32, 1709-1721.

Daniels, J.J., ve Ghandforoush, P., 1990, An improved algorithm for the non-guillotine-constrained cutting-stock problem, *Journal of the Operational Research Society,* 41, 2, 141-149.

Dietrich, R.D., ve Yakowitz, S.J., 1991, A rule-based approach to the trim-loss problem, *International Journal of Production Research,* 29, 2, 401-415.

Dyckhoff, H., 1988, A typology of cutting and packing problems, *EURO/TIMS Conference,* Paris.

KAYNAKLAR DİZİNİ (devam)

Dyckhoff, H., 1990, Typology of cutting and packing problems, *European Journal of Operational Research,* 44, 145-159.

Erol, D., 1990, Giyotinle kesmede iki boyutlu dilme probleminin çözümü için yordamsal bir yaklaşım, Doktora tezi, *Anadolu Üniversitesi, Mühendislik Mimarlık Fakültesi Yayınları,* 127 s.

Faina, L., 1999, An application of simulated annealing to the cutting stock problem, *European Journal of Operational Research,* 114, 542-556.

Fraser, H.J. ve George, J.A., 1994, Integrated container loading software for pulp and paper industry, *European Journal of Operational Research,* 77, 466-474.

Gen, M. ve Cheng, R., 1996, Genetic algorithms and engineering design, *A Wiley-Interscience Publication,* 411 p.

Gilmore, P.C. ve Gomory, R.E., 1961, A linear programming approach to the cutting-stock problem, *Operational Research*, 9, 849-859.

Gilmore, P.C. ve Gomory, R.E., 1963, A linear programming approach to the cutting-stock problem part II, *Operational Research*, 11, 863-888.

Gilmore, P.C. ve Gomory, R.E., 1965, Multistage cutting stock problems of two and more dimensions, *Operational Research*, 13, 94-120.

Gilmore, P.C. ve Gomory, R.E., 1966, The theory and the computation of knapsack functions, *Operational Research*, 14, 1045-1074.

Gomez, A. ve Fuente, D., 2000, Resolution of strip-packing problems with genetic algorithms, *Journal of the Operational Research Society,* 51, 1289-1295.

Graham, R.L. ve Lubachevsky, B.D., 1996, Repeated patterns of dense packing of equal disks in a square, *The Electronic Journal of Combinatorics,* 3, 16, 1-17.

Gun, Y.G., Seong, Y.J. ve Kang, M.K., 2003, A best-first branch and bound algorithm for unconstrained two-dimensional cutting problems, *Operations Research Letters,* 31, 301-307.

Healy, P., Creavin, M. ve Kuusik, A., 1999, An optimal algorithm for rectangle placement, *Operations Research Letters,* 24, 73-80.

Hifi, M., 2004, Dynamic programming and hill-climbing techniques for constrained two-dimensional cutting stock problems, *Journal of Combinatorial Optimization,* 8, 65-84.

KAYNAKLAR DİZİNİ (devam)

Hifi, M. ve M'Hallah, R., 2002, A best-local position procedure-based heuristic for the two-dimensional layout problem, *International Journal on Informatics,* 2, 1, 33-56.

Hifi, M. ve M'Hallah, R., 2004, Approximate algorithms for constrained circular cutting problems, *Computers and Operations Research,* 31, 675-694.

Hifi, M., Paschos, V.T. ve Zissimopoulos, V., 2004, A simulated annealing approach for the circular cutting problem, *European Journal of Operational Research,* 159, 430-448.

Hopper, E. ve Turton, B.C.H., 2001, A review of the application of meta-heuristic algorithms to 2D strip packing problems, *Artificial Intelligence Review,* 16, 257-300.

Hopper, E. ve Turton, B.C.H., 2001, An empirical investigation of meta-heuristic and heuristic algorithms for a 2D packing problem, *European Journal of Operational Research,* 128, 34-57.

İşlier, A., 1993, Malzeme kesme planlarının tasarımında çok ölçütlü karar verme tekniklerinin uygulanması, Doktora tezi, *Osmangazi Üniversitesi Mühendislik Mimarlık Fakültesi,* 116 s.

Kenyon, C. ve Remila, E., 2000, A near optimal solution to a two dimensional cutting stock problem, *Mathematics of Operations Research,* 25, 645-656.

Lai, K.K. ve Chan, J.W.M., 1997, Developing a simulated annealing algorithm for the cutting stock problem, *Computers Industrial Engineering,* 32, 1, 115-127.

Leung, T.W., Yung, C.H. ve Troutt, M.D., 2001, Application of genetic search and simulated annealing to the two-dimensional non-guillotine cutting stock problem, *Computers & Industrial Engineering,* 40, 201-214.

Leung, T.W., Chan, C.K. ve Troutt, M.D., 2003, Application of a mixed simulated annealing-genetic algorithm heuristic for the two-dimensional orthogonal packing problem, *European Journal of Operational Research,* 145, 530-542.

Li, H.L., Tsai, J.F. ve Hu, N.Z., 2003, A distributed global optimization method for packing problems, *Journal of the Operational Research Society,* 54, 419-425.

Lins, L., Lins, S. ve Morabito, R., 2003, An L-approach for packing (c, w)-rectangles into rectangular and L-shaped pieces, *Journal of the Operational Research Society,* 54, 777-789.

KAYNAKLAR DİZİNİ (devam)

Liu, D. ve Teng, H., 1999, An improved BL-algorithm for genetic algorithm of the orthogonal packing of rectangles, *European Journal of Operational Research,* 112, 413-420.

Lodi, A., Martello, S. ve Monaci, M., 2002, Two-dimensional packing problems: A survey, *European Journal of Operational Research,* 141, 241-252.

Lodi, A., Martello, S. ve Vigo, D., 2002, Recent advances on two-dimensional bin packing problems, *Discrete Applied Mathematics,* 123, 379-396.

Martello, S., Monaci, M. ve Vigo, D., 2003, An exact approach to the strip-packing problem, *Informs Journal on Computing,* 15, 3, 310-319.

Nonas, S.L. ve Thorstenson, A., 2000, A combined cutting-stock and lot-sizing problem, *European Journal of Operational Research,* 120, 327-342.

Nurmela, K.J. ve Ostergard, P.R.J., 1997, Packing up to 50 equal circles in a square, *Discrete & Computational Geometry,* 18, 111-120.

Nurmela, K.J. ve Ostergard, P.R.J., 1999, More optimal packing of equal circles in a square, *Discrete & Computational Geometry,* 22, 439-457.

Parada, V., Palma, R., Sales, D. ve Gomes, A., 2000, A comparative numerical analysis for the guillotine two-dimensional cutting problem, *Annals of Operations Research,* 96, 1, 245-254.

Reeves, C.R., 1993, Modern heuristic techniques for combinatorial problems, *Oxford Blackwell Scientific*.

Saraç T., 2001, Bir kutu fabrikası için standart kağıt enlerinin belirlenmesi, Yüksek lisans tezi, *Osmangazi Üniversitesi Mühendislik Mimarlık Fakültesi,* 60 s.

Scheithauer, G. ve Sommerweis, U., 1998, 4-Block heuristic for the rectangle packing problem, *European Journal of Operational Research,* 108, 509-526.

Sevük, N., 1996, Bir boyutlu malzeme kesme problemi için kesme planlarının kombinasyonunda genetik algoritma kullanılması, Yüksek lisans tezi, *Osmangazi Üniversitesi Mühendislik Mimarlık Fakültesi,* 63 s.

Singh, R. ve Sekhon, G.S., 1998, A low-cost modeler for two-dimensional metal stamping layouts, *Journal of Materials Processing Technology,* 84, 79-89.

Stoyan, Y.G. ve Yaskov, G.N., 1998, Mathematical model and solution method of optimization problem of placement of rectangles and circles taking into account special constraints, *International Translation Operational Research,* 5, 1, 45-57.

KAYNAKLAR DİZİNİ (devam)

Stoyan, Y.G., ve Yaskov, G.N., 2004, A mathematical model and a solution method for the problem of placing various-sized circles into a strip, *European Journal of Operational Research,* 156, 590-600.

Suliman, S.M.A., 2001, Pattern generating procedure for the cutting stock problem, *International Journal of Production Economics,* 74, 293-301.

Tam, Y.K., 1991, A Simulated annealing algorithm for allocating space to manufacturing cells, *International Journal of Production Research,* 30, 62-87.

Valdes, R.A., Parajon, A. ve Tamarit, J.M., 2002, A computational study of LP-based heuristic algorithms for two-dimensional guillotine cutting stock problem, *Operational Research Spectrum,* 24, 2, 179-192.

Vanderbeck, F., 2001, A nested decomposition approach to a three-stage two-dimensional cutting-stock problem, *Management Science,* 47, 6, 864-879.

Wang, H., Huang, W., Zhang, Q ve Xu, D., 2002, An improved algorithm for the packing of unequal circles within a larger containing circle, *European Journal of Operational Research,* 141, 440-453.

WenQi, H., Yu, L. ve RuChu, X., 2001, Local search based on a physical model for solving a circle packing problem, *4th Meta Heuristics International Conference,* 455-459.

Wu, Y.L., Huang, W., Lau, S.C., Wong, C.K. ve Young, G.H., 2002, An effective quasi-human based heuristic for solving the rectangle packing problem, *European Journal of Operational Research,* 141, 341-358.

Zhang, D., Liu, Y. ve Chen, S., 2004, Packing different-sized circles into a rectangular container using simulated annealing algorithm, *International Journal Of Computational Intelligence,* 1, 2, 124-127.

EKLER

EK-1: Kaynaklarda Yer Alan Problemlere Ait Veriler

Problem-1:

Parça Sayısı: 30 Ana Malzemenin Boyu: 17.491 Ana Malzemenin Eni: 9.5

P.No	Y.çap	P.No	Y.çap	P.No	Y.çap	P.No	Y.çap	P.No	Y.çap
1	1.275	11	0.868	21	0.855	31		41	
2	0.760	12	1.404	22	0.527	32		42	
3	1.274	13	1.085	23	0.955	33		43	
4	1.374	14	0.887	24	1.670	34		44	
5	1.281	15	0.853	25	0.564	35		45	
6	1.493	16	1.237	26	0.747	36		46	
7	1.044	17	1.491	27	1.399	37		47	
8	1.484	18	0.592	28	1.292	38		48	
9	1.739	19	2.050	29	0.807	39		49	
10	1.180	20	0.845	30	1.551	40		50	

Problem-2:

Parça Sayısı: 20 Ana Malzemenin Boyu: 14.895 Ana Malzemenin Eni: 8.5

P.No	Y.çap	P.No	Y.çap	P.No	Y.çap	P.No	Y.çap	P.No	Y.çap
1	2.171	11	1.430	21		31		41	
2	0.680	12	0.850	22		32		42	
3	1.229	13	0.612	23		33		43	
4	1.636	14	0.891	24		34		44	
5	1.309	15	1.525	25		35		45	
6	1.819	16	1.325	26		36		46	
7	1.189	17	1.551	27		37		47	
8	1.210	18	0.947	28		38		48	
9	0.934	19	0.961	29		39		49	
10	0.566	20	1.484	30		40		50	

Problem-3:

Parça Sayısı: 25 Ana Malzemenin Boyu: 14.930 Ana Malzemenin Eni: 9

P.No	Y.çap	P.No	Y.çap	P.No	Y.çap	P.No	Y.çap	P.No	Y.çap
1	1.782	11	1.547	21	0.948	31		41	
2	0.833	12	0.509	22	1.044	32		42	
3	2.147	13	1.245	23	0.589	33		43	
4	0.549	14	1.362	24	0.587	34		44	
5	0.997	15	0.576	25	1.273	35		45	
6	0.954	16	0.627	26		36		46	
7	0.837	17	0.595	27		37		47	
8	0.991	18	0.904	28		38		48	
9	1.647	19	0.804	29		39		49	
10	1.945	20	1.812	30		40		50	

EK-1: Kaynaklarda Yer Alan Problemlere Ait Veriler (devam)

Problem-4:

Parça Sayısı: 35 Ana Malzemenin Boyu: 24.355 Ana Malzemenin Eni: 11

P.No	Y.çap	P.No	Y.çap	P.No	Y.çap	P.No	Y.çap	P.No	Y.çap
1	1.886	11	2.166	21	1.658	31	1.123	41	
2	1.500	12	1.478	22	1.542	32	1.406	42	
3	1.489	13	2.178	23	1.007	33	1.540	43	
4	0.687	14	1.285	24	1.314	34	1.921	44	
5	1.273	15	1.947	25	0.716	35	1.151	45	
6	1.409	16	0.823	26	1.117	36		46	
7	0.609	17	1.056	27	1.477	37		47	
8	1.163	18	0.680	28	1.258	38		48	
9	1.153	19	0.606	29	0.701	39		49	
10	1.763	20	1.938	30	1.869	40		50	

Problem-5:

Parça Sayısı: 100 Ana Malzemenin Boyu: 38.0465 Ana Malzemenin Eni: 15

P.No	Y.çap	P.No	Y.çap	P.No	Y.çap	P.No	Y.çap	P.No	Y.çap
1	0.8680	21	1.2038	41	0.9450	61	0.8630	81	2.0793
2	1.6628	22	0.9770	42	2.0147	62	0.6390	82	0.5540
3	0.6564	23	1.1670	43	2.1864	63	0.7414	83	1.7648
4	0.8723	24	1.1880	44	0.9740	64	1.4580	84	0.8970
5	2.1796	25	0.5750	45	1.2570	65	1.0480	85	0.5330
6	0.6920	26	0.8990	46	0.6400	66	1.0900	86	1.2640
7	0.8842	27	1.8498	47	0.6343	67	1.2440	87	1.1200
8	2.1643	28	0.7550	48	0.5420	68	1.1562	88	0.6660
9	1.4673	29	0.6200	49	1.0360	69	1.1620	89	2.0844
10	0.9980	30	1.0620	50	1.2250	70	0.5663	90	1.2395
11	1.8770	31	1.6135	51	1.0250	71	1.6458	91	0.9450
12	0.6450	32	1.0032	52	0.9896	72	0.6490	92	1.1103
13	0.6270	33	0.6598	53	0.7975	73	1.0525	93	2.0691
14	1.0160	34	1.2540	54	0.9454	74	1.2701	94	0.9690
15	1.3940	35	1.2030	55	0.7822	75	0.6820	95	1.4840
16	1.0700	36	1.6934	56	1.9858	76	1.4260	96	0.7652
17	1.2520	37	1.2560	57	0.7340	77	0.6547	97	1.3040
18	0.6730	38	0.7380	58	1.1240	78	0.6500	98	0.5425
19	2.1473	39	1.0020	59	1.2174	79	1.1800	99	2.1847
20	0.8026	40	1.2620	60	1.0780	80	1.5948	100	1.1780

EK-1: Kaynaklarda Yer Alan Problemlere Ait Veriler (devam)

Problem-6:

Parça Sayısı: 100 Ana Malzemenin Boyu: 38.6471 Ana Malzemenin Eni: 19

P.No	Y.çap	P.No	Y.çap	P.No	Y.çap	P.No	Y.çap	P.No	Y.çap
1	1.0134	21	1.6832	41	1.4333	61	0.8510	81	1.1950
2	0.7720	22	0.7312	42	1.0790	62	1.9297	82	0.8130
3	2.0045	23	2.1847	43	2.0300	63	1.1820	83	1.9773
4	0.9216	24	1.8430	44	1.4540	64	1.4435	84	1.0160
5	1.9586	25	1.1910	45	1.3075	65	0.8468	85	0.8290
6	1.8702	26	0.6173	46	1.2890	66	2.0079	86	0.8710
7	0.8723	27	1.4970	47	0.6462	67	1.2190	87	2.1575
8	2.0436	28	1.3007	48	1.2200	68	1.1280	88	1.4990
9	2.0249	29	0.5476	49	1.0710	69	1.6356	89	1.3400
10	1.7053	30	2.1303	50	0.8410	70	1.5489	90	1.1370
11	1.4960	31	1.5115	51	1.4350	71	0.8060	91	0.6887
12	1.3920	32	1.3200	52	1.3942	72	0.9640	92	1.1950
13	1.2259	33	1.4750	53	1.8022	73	0.5510	93	2.0946
14	2.1643	34	1.0865	54	0.9200	74	1.8345	94	0.6700
15	0.8145	35	0.9120	55	0.9437	75	1.7869	95	0.7400
16	1.3449	36	1.2090	56	1.4600	76	0.8150	96	0.7860
17	1.2330	37	1.0559	57	1.3350	77	2.0028	97	0.5090
18	0.5880	38	1.5812	58	0.5748	78	1.3500	98	0.9600
19	0.9199	39	0.9790	59	1.4656	79	1.2980	99	1.9314
20	2.0963	40	1.2320	60	1.6832	80	0.9470	100	2.1286

EK-2: Parametre Analizi Sonuçları

Deney No	Popülasyon Büyüklüğü	Turnuva Büyüklüğü	Mutasyon Oranı	Kaydır Oranı	Süre (dak.)	Fire Oranı
1-1	10	6	1	10	00:50	0.1686
2					02:37	0.1798
3					02:36	0.1686
4					02:37	0.1738
5					02:36	0.1738
6					02:38	0.1798
7					02:36	0.1738
8					02:37	0.1686
9					02:37	0.1864
10					01:58	0.1660
				Ortalama:	02:22	0.1739
2-1	10	6	1	20	04:55	0.1738
2					04:54	0.1798
3					04:53	0.1686
4					04:19	0.1798
5					04:53	0.1738
6					04:54	0.1738
7					04:54	0.1798
8					04:53	0.1798
9					04:53	0.1798
10					04:52	0.1798
				Ortalama:	04:50	0.1768
3-1	10	6	1	30	04:34	0.1686
2					07:18	0.1798
3					07:17	0.1686
4					05:12	0.1686
5					07:19	0.1864
6					05:57	0.1686
7					07:17	0.1798
8					07:18	0.1798
9					07:17	0.1738
10					07:17	0.1738
				Ortalama:	06:40	0.1747
4-1	10	6	2	10	02:39	0.1864
2					02:38	0.1864
3					02:43	0.1864
4					02:44	0.1686
5					02:45	0.1798
6					02:46	0.1864
7					02:45	0.1864
8					02:34	0.1798

EK-2: Parametre Analizi Sonuçları (devam)

Deney No	Popülasyon Büyüklüğü	Turnuva Büyüklüğü	Mutasyon Oranı	Kaydır Oranı	Süre (dak.)	Fire Oranı
9					02:36	0.1798
10					02:35	0.1798
				Ortalama:	02:40	0.1819
5-1	10	6	2	20	04:52	0.1798
2					04:45	0.1798
3					04:46	0.1798
4					04:50	0.1738
5					04:48	0.1864
6					04:55	0.1864
7					04:54	0.1864
8					04:56	0.1738
9					04:56	0.1798
10					04:45	0.1864
				Ortalama:	04:50	0.1812
6-1	10	6	2	30	07:18	0.1864
2					07:19	0.1798
3					07:19	0.1798
4					07:18	0.1798
5					07:15	0.1738
6					07:18	0.1798
7					07:20	0.1798
8					07:20	0.1798
9					07:18	0.1798
10					07:14	0.1798
				Ortalama:	07:17	0.1798
7-1	10	6	1	Otomatik	03:46	0.1798
2					01:30	0.1686
3					02:28	0.1686
4					03:38	0.1798
5					03:50	0.1738
6					03:37	0.1738
7					03:39	0.1738
8					03:37	0.1798
9					03:42	0.1798
10					03:39	0.1738
				Ortalama:	03:20	0.1751
8-1	10	6	2	Otomatik	05:28	0.1864
2					05:26	0.1686
3					05:26	0.1798
4					05:27	0.1798
5					05:28	0.1798

EK-2: Parametre Analizi Sonuçları (devam)

Deney No	Popülasyon Büyüklüğü	Turnuva Büyüklüğü	Mutasyon Oranı	Kaydır Oranı	Süre (dak.)	Fire Oranı
6					05:24	0.1738
7					05:27	0.1798
8					05:29	0.1738
9					05:27	0.1864
10					05:23	0.1798
				Ortalama:	05:26	0.1788
9-1	10	2	1	Otomatik	03:44	0.1798
2					03:09	0.1686
3					03:59	0.1686
4					03:58	0.1738
5					04:12	0.1798
6					03:51	0.1738
7					03:22	0.1686
8					03:42	0.1738
9					03:44	0.1686
10					03:21	0.1798
				Ortalama:	03:42	0.1735
10-1	10	4	1	Otomatik	03:26	0.1864
2					02:41	0.1686
3					03:35	0.1798
4					03:50	0.1798
5					03:53	0.1738
6					03:45	0.1738
7					03:58	0.1738
8					03:44	0.1686
9					03:38	0.1738
10					03:36	0.1738
				Ortalama:	03:36	0.1752
11-1	10	6	1	Otomatik	03:46	0.1798
2					01:30	0.1686
3					02:28	0.1686
4					03:38	0.1798
5					03:50	0.1738
6					03:37	0.1738
7					03:39	0.1738
8					03:37	0.1798
9					03:42	0.1798
10					03:39	0.1738
				Ortalama:	03:20	0.1751
12-1	20	4	1	Otomatik	06:25	0.1738
2					06:12	0.1686

EK-2: Parametre Analizi Sonuçları (devam)

Deney No	Popülasyon Büyüklüğü	Turnuva Büyüklüğü	Mutasyon Oranı	Kaydır Oranı	Süre (dak.)	Fire Oranı
3					06:22	0.1686
4					06:20	0.1738
5					06:21	0.1686
6					06:17	0.1686
7					06:22	0.1798
8					06:21	0.1798
9					06:19	0.1686
10					06:15	0.1798
				Ortalama:	06:19	0.1730
13-1	20	8	1	Otomatik	06:17	0.1798
2					03:54	0.1686
3					06:11	0.1798
4					06:08	0.1738
5					06:28	0.1798
6					06:11	0.1738
7					02:30	0.1686
8					06:11	0.1738
9					06:19	0.1686
10					06:17	0.1798
				Ortalama:	05:38	0.1746
14-1	20	12	1	Otomatik	06:04	0.1798
2					06:18	0.1686
3					06:17	0.1798
4					06:16	0.1738
5					06:16	0.1864
6					03:08	0.1686
7					06:17	0.1686
8					05:13	0.1686
9					06:06	0.1798
10					06:14	0.1738
				Ortalama:	05:48	0.1747
15-1	30	6	1	Otomatik	06:02	0.1686
2					09:44	0.1798
3					09:40	0.1686
4					06:11	0.1686
5					05:28	0.1686
6					09:46	0.1686
7					03:56	0.1686
8					03:39	0.1686
9					09:52	0.1738
10					09:41	0.1798

EK-2: Parametre Analizi Sonuçları (devam)

Deney No	Popülasyon Büyüklüğü	Turnuva Büyüklüğü	Mutasyon Oranı	Kaydır Oranı	Süre (dak.)	Fire Oranı
				Ortalama:	07:23	0.1713
16-1	30	12	1	Otomatik	02:48	0.1686
2					09:58	0.1686
3					09:37	0.1686
4					10:01	0.1738
5					09:58	0.1738
6					04:13	0.1686
7					10:03	0.1798
8					10:04	0.1798
9					10:00	0.1738
10					09:51	0.1738
				Ortalama:	08:39	0.1729
17-1	30	18	1	Otomatik	09:45	0.1798
2					10:04	0.1798
3					09:55	0.1798
4					03:34	0.1686
5					09:54	0.1798
6					10:30	0.1738
7					09:42	0.1798
8					09:41	0.1864
9					09:50	0.1864
10					09:46	0.1798
				Ortalama:	09:16	0.1794

EK-3. Rastgele Üretilen Problemlere Ait Veriler

Problem - 1 - 1:

Parça Sayısı: 28 Ana Malzemenin Boyu: 97.14 Ana Malzemenin Eni: 92.12

P.No	Y.çap	Talep	P.No	Y.çap	Talep	P.No	Y.çap	Talep
1	5.804	5	11	7.372	7	21	4.963	5
2	8.119	6	12	9.686	7	22	6.368	4
3	9.569	7	13	6.548	7	23	9.358	8
4	6.590	8	14	9.510	9	24	4.634	4
5	8.661	8	15	11.195	5	25	5.341	1
6	10.952	4	16	11.657	9	26	6.471	8
7	6.026	3	17	5.040	1	27	10.037	8
8	8.123	7	18	6.912	9	28	8.484	5
9	6.432	7	19	5.523	9			
10	8.775	4	20	9.464	2			

Problem - 1 -2:

Parça Sayısı: 26 Ana Malzemenin Boyu: 89.94 Ana Malzemenin Eni: 77.78

P.No	Y.çap	Talep	P.No	Y.çap	Talep	P.No	Y.çap	Talep
1	6.819	5	11	8.184	3	21	7.476	4
2	10.039	7	12	5.202	3	22	4.510	1
3	8.065	3	13	9.561	1	23	4.661	4
4	7.238	4	14	9.067	3	24	8.038	3
5	5.995	5	15	4.926	5	25	9.870	3
6	3.990	7	16	4.569	4	26	8.951	3
7	9.486	4	17	6.932	4			
8	6.323	5	18	6.806	5			
9	5.113	4	19	9.135	5			
10	6.173	5	20	8.329	1			

Problem - 1 - 3:

Parça Sayısı: 23 Ana Malzemenin Boyu: 73.84 Ana Malzemenin Eni: 60.70

P.No	Y.çap	Talep	P.No	Y.çap	Talep	P.No	Y.çap	Talep
1	6.117	4	11	5.088	9	21	4.244	5
2	7.162	4	12	4.196	1	22	3.854	9
3	3.785	5	13	4.663	5	23	3.693	7
4	6.538	3	14	5.147	4			
5	6.644	5	15	7.617	6			
6	6.534	6	16	5.581	1			
7	3.280	1	17	4.437	7			
8	4.598	9	18	5.493	2			
9	7.764	5	19	7.834	6			
10	4.122	3	20	5.738	3			

EK-3. Rastgele Üretilen Problemlere Ait Veriler (devam)

Problem - 1 - 4:

Parça Sayısı: 23 Ana Malzemenin Boyu: 94.99 Ana Malzemenin Eni: 78.17

P.No	Y.çap	Talep	P.No	Y.çap	Talep	P.No	Y.çap	Talep
1	9.365	1	9	5.104	6	17	6.555	6
2	9.128	6	10	4.060	1	18	7.407	3
3	7.735	3	11	5.278	6	19	4.288	6
4	5.043	9	12	8.593	5	20	7.372	3
5	10.028	7	13	4.027	7	21	5.900	2
6	8.182	5	14	6.610	2	22	7.895	3
7	5.848	8	15	7.102	6	23	9.836	8
8	3.923	4	16	9.994	3			

Problem - 1 - 5:

Parça Sayısı: 24 Ana Malzemenin Boyu: 69.55 Ana Malzemenin Eni: 65.91

P.No	Y.çap	Talep	P.No	Y.çap	Talep	P.No	Y.çap	Talep
1	4.611	3	9	7.920	8	17	5.730	3
2	4.470	4	10	4.433	4	18	5.657	5
3	7.611	3	11	7.872	6	19	6.944	4
4	4.422	5	12	6.782	2	20	7.008	1
5	6.362	1	13	6.600	3	21	6.745	6
6	6.886	1	14	4.583	5	22	4.744	2
7	4.189	2	15	6.788	5	23	8.497	5
8	5.515	5	16	6.836	1	24	3.437	7

Problem - 2 - 1:

Parça Sayısı: 40 Ana Malzemenin Boyu: 137.44 Ana Malzemenin Eni: 101.01

P.No	Y.çap	Talep	P.No	Y.çap	Talep	P.No	Y.çap	Talep
1	12.502	1	16	8.999	1	31	11.519	8
2	10.843	8	17	13.131	1	32	10.916	3
3	9.001	3	18	6.567	3	33	9.067	9
4	9.191	3	19	10.325	4	34	12.576	1
5	12.276	1	20	7.623	4	35	11.279	8
6	7.971	3	21	6.153	8	36	6.776	4
7	7.881	3	22	8.579	7	37	10.552	9
8	11.524	4	23	5.679	1	38	10.948	7
9	7.610	3	24	11.564	6	39	7.212	7
10	9.834	2	25	8.892	6	40	8.853	2
11	7.199	1	26	5.366	4			
12	9.975	3	27	8.812	7			
13	10.693	7	28	13.029	3			
14	12.995	4	29	7.779	9			
15	8.349	5	30	12.179	9			

EK-3. Rastgele Üretilen Problemlere Ait Veriler (devam)

Problem - 2 - 2:

Parça Sayısı: 62 Ana Malzemenin Boyu: 149.66 Ana Malzemenin Eni: 148.78

P.No	Y.çap	Talep	P.No	Y.çap	Talep	P.No	Y.çap	Talep
1	16.82	7	22	13.60	3	43	18.43	3
2	12.70	1	23	8.83	3	44	14.77	9
3	17.30	8	24	12.25	8	45	7.61	7
4	11.97	7	25	13.68	8	46	14.77	6
5	13.47	8	26	18.40	4	47	15.23	7
6	13.54	2	27	17.69	2	48	8.96	8
7	18.42	6	28	8.99	3	49	7.78	8
8	11.59	4	29	11.16	7	50	12.74	3
9	11.24	5	30	11.13	5	51	12.46	2
10	15.94	5	31	7.48	5	52	16.41	1
11	16.11	5	32	10.04	1	53	8.72	2
12	13.45	4	33	12.12	9	54	9.08	5
13	17.63	4	34	14.74	5	55	13.26	5
14	15.91	5	35	18.33	7	56	17.61	3
15	10.14	9	36	7.60	6	57	13.28	5
16	15.51	5	37	17.74	3	58	17.47	3
17	14.27	8	38	15.96	2	59	9.77	1
18	14.85	4	39	13.94	6	60	13.31	7
19	13.12	4	40	8.18	6	61	16.28	8
20	12.20	9	41	15.23	6	62	15.32	7
21	10.52	7	42	18.79	9			

Problem - 2 - 3:

Parça Sayısı: 35 Ana Malzemenin Boyu: 148.53 Ana Malzemenin Eni: 147.28

P.No	Y.çap	Talep	P.No	Y.çap	Talep	P.No	Y.çap	Talep
1	13.65	2	13	9.58	4	25	10.37	5
2	16.52	6	14	17.66	3	26	18.60	2
3	7.52	6	15	15.64	1	27	16.45	1
4	7.87	3	16	14.65	3	28	12.05	3
5	18.80	3	17	13.45	8	29	18.34	7
6	16.19	5	18	18.64	2	30	10.50	1
7	8.37	4	19	14.69	3	31	14.39	8
8	14.72	9	20	12.24	8	32	18.43	4
9	13.83	7	21	11.62	7	33	17.13	6
10	7.53	8	22	15.87	8	34	14.45	5
11	8.29	9	23	18.61	7	35	18.05	2
12	7.55	5	24	12.80	6			

EK-3. Rastgele Üretilen Problemlere Ait Veriler (devam)

Problem - 2 - 4:

Parça Sayısı: 53 Ana Malzemenin Boyu: 137.46 Ana Malzemenin Eni: 133.16

P.No	Y.çap	Talep	P.No	Y.çap	Talep	P.No	Y.çap	Talep
1	16.69	7	19	12.57	9	37	17.01	2
2	11.92	6	20	13.39	7	38	15.36	8
3	10.61	8	21	8.28	1	39	14.58	3
4	16.55	8	22	15.46	7	40	11.63	2
5	10.56	1	23	9.91	8	41	16.71	1
6	13.53	5	24	15.74	6	42	11.12	1
7	13.65	4	25	10.25	5	43	15.97	6
8	15.54	4	26	9.01	2	44	6.82	7
9	12.89	6	27	11.40	3	45	9.13	8
10	8.95	1	28	10.47	9	46	8.33	4
11	9.33	6	29	7.75	1	47	14.67	8
12	9.88	5	30	7.98	7	48	7.39	6
13	16.22	7	31	14.00	7	49	7.64	5
14	16.43	9	32	12.87	9	50	10.06	5
15	12.58	9	33	10.82	6	51	17.13	7
16	16.87	9	34	6.79	4	52	7.52	9
17	16.61	2	35	10.19	5	53	13.12	1
18	11.08	4	36	8.49	3			

Problem - 2 - 5:

Parça Sayısı: 61 Ana Malzemenin Boyu: 141.14 Ana Malzemenin Eni: 126.99

P.No	Y.çap	Talep	P.No	Y.çap	Talep	P.No	Y.çap	Talep
1	10.33	3	22	12.23	1	43	10.07	3
2	12.44	5	23	6.70	1	44	8.71	7
3	14.55	1	24	16.32	1	45	9.15	5
4	8.39	7	25	12.65	7	46	10.19	7
5	16.29	7	26	15.41	7	47	9.00	3
6	11.96	3	27	11.75	3	48	9.24	7
7	10.41	1	28	11.92	8	49	13.62	8
8	13.62	7	29	11.77	5	50	7.45	1
9	9.10	2	30	11.07	7	51	13.40	4
10	7.66	5	31	15.46	6	52	16.10	8
11	7.79	7	32	13.50	5	53	14.48	3
12	10.05	1	33	14.34	2	54	15.63	4
13	14.51	3	34	10.25	3	55	9.74	7
14	7.88	3	35	15.56	2	56	9.14	3
15	10.64	5	36	10.93	9	57	13.20	4
16	12.78	8	37	10.26	3	58	7.35	1
17	15.89	5	38	13.35	7	59	15.76	3
18	15.41	1	39	15.68	5	60	15.99	5
19	13.51	5	40	16.42	5	61	14.57	2

EK-3. Rastgele Üretilen Problemlere Ait Veriler (devam)

P.No	Y.çap	Talep	P.No	Y.çap	Talep	P.No	Y.çap	Talep
20	13.54	2	41	7.99	8			
21	10.13	4	42	15.20	2			

Problem - 3 - 1:

Parça Sayısı: 86 Ana Malzemenin Boyu: 219.06 Ana Malzemenin Eni: 195.88

P.No	Y.çap	Talep	P.No	Y.çap	Talep	P.No	Y.çap	Talep
1	23.40	2	30	21.42	4	59	23.78	6
2	13.45	3	31	18.31	6	60	17.56	8
3	11.96	3	32	16.28	2	61	23.75	4
4	15.39	7	33	20.65	4	62	19.94	9
5	17.57	3	34	20.72	8	63	22.21	7
6	22.08	1	35	20.29	2	64	18.72	4
7	12.23	9	36	14.62	2	65	14.81	2
8	10.60	6	37	18.15	4	66	15.71	7
9	18.81	5	38	16.10	7	67	12.05	3
10	24.22	2	39	14.85	5	68	21.40	1
11	13.36	2	40	25.01	7	69	23.94	6
12	13.97	4	41	18.84	4	70	10.07	5
13	19.40	9	42	17.29	7	71	17.06	2
14	13.21	6	43	11.31	3	72	20.03	5
15	18.04	1	44	21.57	4	73	16.75	7
16	18.44	8	45	24.83	4	74	22.55	7
17	18.74	3	46	11.75	7	75	25.00	4
18	14.95	6	47	22.15	7	76	14.20	3
19	24.32	5	48	22.74	2	77	20.78	8
20	22.14	9	49	13.96	3	78	19.06	2
21	15.49	2	50	19.51	4	79	11.15	2
22	17.38	6	51	23.88	6	80	16.46	5
23	17.76	8	52	20.74	7	81	23.69	6
24	20.48	1	53	10.35	9	82	13.74	9
25	15.42	5	54	22.64	6	83	24.49	5
26	25.35	5	55	11.85	5	84	13.17	2
27	10.61	9	56	20.78	3	85	12.39	3
28	14.01	5	57	13.90	7	86	14.14	3
29	23.54	4	58	20.11	2			

Problem - 3 - 2:

Parça Sayısı: 89 Ana Malzemenin Boyu: 244.62 Ana Malzemenin Eni: 204.51

P.No	Y.çap	Talep	P.No	Y.çap	Talep	P.No	Y.çap	Talep
1	11.16	9	31	22.08	6	61	20.32	8
2	16.66	3	32	13.28	3	62	18.04	2
3	17.61	5	33	20.93	4	63	20.44	6
4	22.78	3	34	12.72	4	64	10.89	2

EK-3. Rastgele Üretilen Problemlere Ait Veriler (devam)

P.No	Y.çap	Talep	P.No	Y.çap	Talep	P.No	Y.çap	Talep
5	14.19	7	35	22.16	5	65	12.83	2
6	21.42	7	36	14.71	3	66	11.48	1
7	21.71	5	37	24.44	4	67	18.28	1
8	22.05	6	38	25.90	4	68	12.65	1
9	13.08	8	39	24.33	9	69	23.47	9
10	11.94	2	40	17.18	1	70	11.10	7
11	22.96	6	41	23.62	4	71	17.00	3
12	10.58	5	42	23.12	7	72	20.90	1
13	16.11	7	43	24.50	4	73	13.22	8
14	13.70	4	44	16.41	2	74	25.75	9
15	21.60	2	45	12.44	1	75	14.55	3
16	23.19	2	46	15.54	7	76	17.01	5
17	25.70	7	47	24.94	5	77	10.82	9
18	15.63	1	48	16.51	3	78	19.56	8
19	16.08	1	49	11.70	1	79	15.25	7
20	13.14	4	50	10.48	9	80	23.52	8
21	13.82	5	51	26.13	5	81	22.78	8
22	10.99	7	52	21.87	1	82	18.26	4
23	11.23	8	53	11.81	7	83	21.62	7
24	23.91	4	54	19.63	2	84	14.14	5
25	12.02	6	55	18.59	9	85	25.16	2
26	21.92	1	56	12.60	6	86	19.21	4
27	26.44	4	57	13.62	4	87	26.23	8
28	17.47	1	58	18.09	9	88	20.10	4
29	21.43	5	59	23.78	5	89	20.87	1
30	25.85	5	60	18.13	2			

Problem - 3 - 3:

Parça Sayısı: 92 Ana Malzemenin Boyu: 230.03 Ana Malzemenin Eni: 154.51

P.No	Y.çap	Talep	P.No	Y.çap	Talep	P.No	Y.çap	Talep
1	17.50	3	32	9.12	8	63	17.60	1
2	18.27	7	33	11.63	8	64	13.49	2
3	8.54	3	34	17.70	1	65	12.87	6
4	11.70	7	35	16.21	9	66	19.31	8
5	9.72	1	36	10.88	6	67	18.82	7
6	9.09	4	37	16.74	2	68	13.98	2
7	15.73	9	38	16.34	4	69	9.62	6
8	19.91	8	39	10.43	3	70	19.99	3
9	8.39	3	40	16.71	2	71	9.72	5
10	11.91	6	41	9.41	4	72	17.82	5
11	14.00	8	42	17.77	4	73	12.14	2
12	19.41	8	43	11.62	2	74	14.23	8
13	15.48	9	44	8.46	5	75	11.13	7
14	19.90	7	45	11.72	8	76	14.58	5
15	15.88	7	46	10.52	1	77	13.14	5

EK-3. Rastgele Üretilen Problemlere Ait Veriler (devam)

P.No	Y.çap	Talep	P.No	Y.çap	Talep	P.No	Y.çap	Talep
16	11.54	1	47	15.30	2	78	10.65	1
17	18.50	5	48	19.94	5	79	13.91	8
18	19.45	4	49	18.89	8	80	8.70	4
19	10.11	5	50	13.40	5	81	15.58	4
20	18.72	4	51	17.96	4	82	11.08	7
21	9.01	5	52	19.47	7	83	9.92	5
22	8.72	7	53	17.85	5	84	13.34	6
23	17.33	9	54	19.97	3	85	14.09	7
24	16.42	4	55	10.85	1	86	16.78	3
25	13.52	1	56	14.99	8	87	10.52	7
26	16.21	2	57	18.10	7	88	19.41	6
27	9.68	2	58	16.99	8	89	13.57	8
28	13.54	3	59	10.94	1	90	9.90	5
29	13.15	4	60	10.74	5	91	15.65	1
30	8.03	5	61	8.50	8	92	14.18	6
31	15.88	7	62	15.38	2			

Problem - 3 - 4:

Parça Sayısı: 99 Ana Malzemenin Boyu: 246.14 Ana Malzemenin Eni: 196.05

P.No	Y.çap	Talep	P.No	Y.çap	Talep	P.No	Y.çap	Talep
1	18.24	6	34	19.62	2	67	22.82	1
2	23.90	9	35	19.33	9	68	21.49	7
3	19.52	1	36	21.27	5	69	12.52	8
4	22.90	7	37	25.12	1	70	13.91	2
5	23.02	1	38	23.21	6	71	20.15	5
6	10.11	8	39	14.56	6	72	12.36	1
7	11.20	3	40	20.82	6	73	18.00	6
8	17.91	6	41	10.71	5	74	20.57	7
9	10.79	1	42	19.54	3	75	10.15	4
10	22.07	1	43	17.37	4	76	14.33	6
11	25.35	8	44	24.31	9	77	16.00	7
12	25.35	1	45	18.41	4	78	13.32	7
13	18.89	4	46	17.32	3	79	10.71	2
14	19.50	1	47	12.29	7	80	15.67	5
15	13.15	3	48	18.03	2	81	10.77	6
16	13.91	1	49	19.16	2	82	14.86	6
17	16.30	5	50	21.63	4	83	22.28	9
18	21.19	6	51	13.51	3	84	20.58	3
19	18.23	1	52	19.92	4	85	24.49	6
20	15.24	3	53	17.47	1	86	20.70	5
21	21.25	5	54	10.33	5	87	20.09	4
22	18.73	8	55	9.97	8	88	9.99	2
23	11.87	1	56	24.63	3	89	10.93	2
24	22.01	6	57	12.21	8	90	22.67	5
25	14.97	4	58	20.34	6	91	24.71	9

EK-3. Rastgele Üretilen Problemlere Ait Veriler (devam)

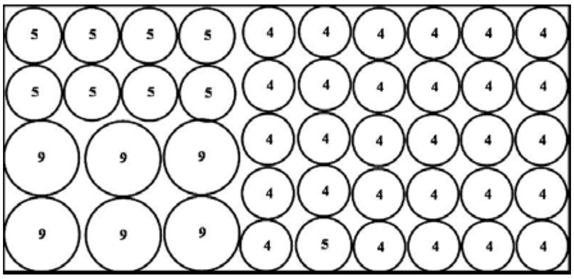
P.No	Y.çap	Talep	P.No	Y.çap	Talep	P.No	Y.çap	Talep
26	15.70	5	59	20.29	3	92	12.12	3
27	10.73	5	60	11.37	2	93	20.18	1
28	11.49	7	61	20.14	9	94	16.76	8
29	12.28	3	62	14.32	1	95	15.77	2
30	19.86	5	63	22.13	3	96	16.22	2
31	21.88	6	64	14.47	2	97	18.29	4
32	16.05	5	65	21.03	6	98	10.98	7
33	22.32	1	66	14.00	7	99	18.25	7

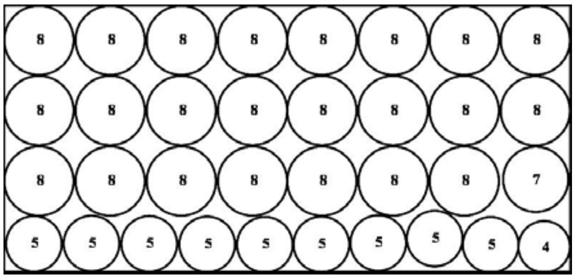
<u>Problem - 3 - 5:</u>

Parça Sayısı: 57 Ana Malzemenin Boyu: 242.22 Ana Malzemenin Eni: 211.49

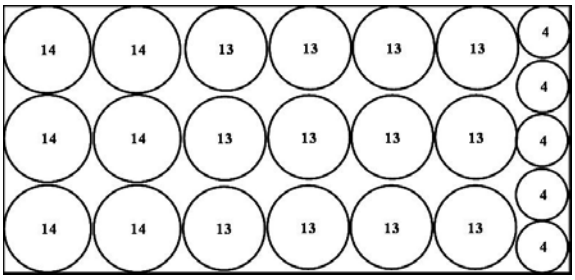
P.No	Y.çap	Talep	P.No	Y.çap	Talep	P.No	Y.çap	Talep
1	14.72	6	21	14.06	3	41	17.25	2
2	17.03	7	22	26.55	8	42	24.75	2
3	15.01	7	23	20.04	7	43	16.76	5
4	19.71	6	24	16.21	3	44	18.88	2
5	23.11	1	25	26.57	2	45	18.36	2
6	16.72	7	26	23.70	4	46	15.01	4
7	23.72	8	27	21.29	4	47	22.14	5
8	16.22	7	28	15.97	1	48	24.53	3
9	16.28	7	29	21.93	7	49	22.85	3
10	23.83	5	30	10.81	4	50	10.81	4
11	11.04	7	31	13.10	1	51	22.87	7
12	23.81	3	32	15.80	8	52	22.69	1
13	12.36	3	33	22.74	4	53	15.52	5
14	15.32	1	34	22.19	2	54	10.76	3
15	23.25	4	35	16.61	5	55	23.45	3
16	19.02	1	36	20.01	3	56	20.06	5
17	20.21	1	37	16.99	9	57	22.69	7
18	20.21	2	38	17.34	7			
19	16.99	3	39	22.59	7			
20	17.78	7	40	13.08	5			

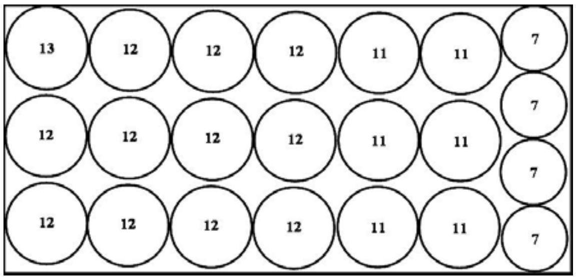
EK-4: İşyeri Şartlarında Oluşturulan Kesme Planları



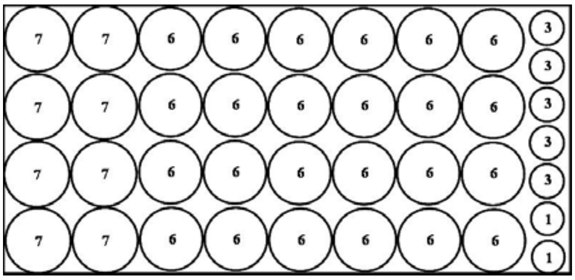


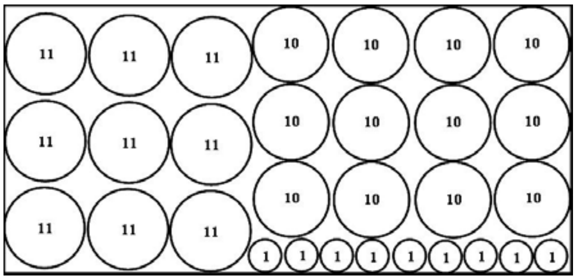
Kesme Planı-1 (Adet:1, Fire Oranı:0.2479) Kesme Planı-2 (Adet:1, Fire Oranı:0.2149)



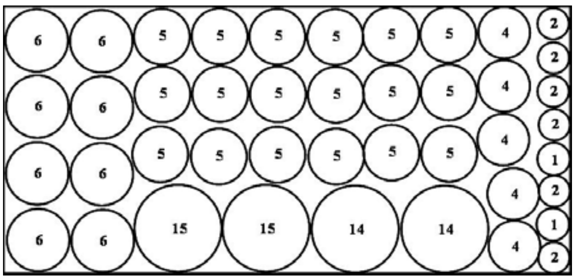


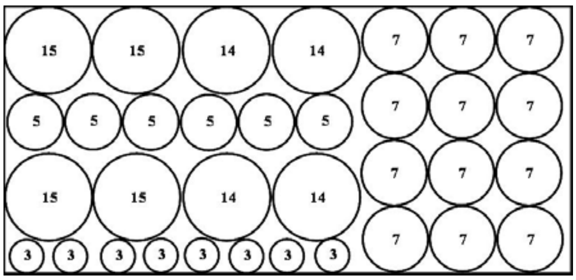
Kesme Planı-3 (Adet:1, Fire Oranı:0.2550) Kesme Planı-4 (Adet:1, Fire Oranı:0.2943)





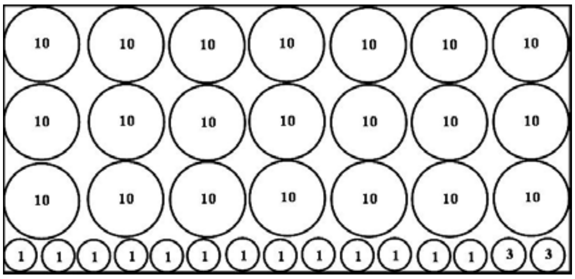
Kesme Planı-5 (Adet:1, Fire Oranı:0.2509) Kesme Planı-6 (Adet:1, Fire Oranı:0.2916)

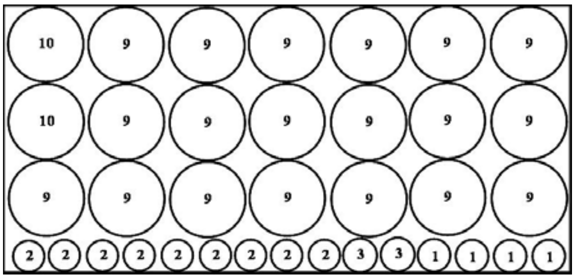




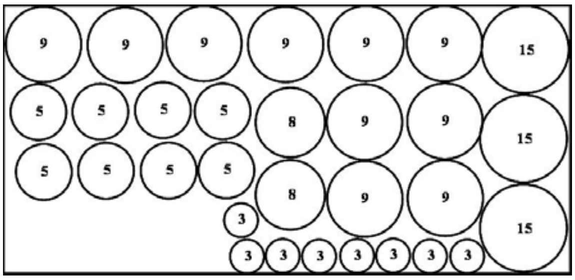
Kesme Planı-7 (Adet:1, Fire Oranı:0.2382) Kesme Planı-8 (Adet:1, Fire Oranı:0.2552)

EK-4: İşyeri Şartlarında Oluşturulan Kesme Planları (devam)





Kesme Planı-9 (Adet:1, Fire Oranı:0.2842) Kesme Planı-10 (Adet:1, Fire Oranı:0.2880)



Kesme Planı-11 (Adet:1, Fire Oranı:0.3439)

EK-5. Bilgisayar Programı

```
Option Explicit
Dim kontrol, kppkon As Boolean
Dim çs, tçç, çno, tur, amalzeme, Klt(), KltYön(1 To 4) As Byte
Dim saat1, saat2, saat3, dakika1, dakika2, dakika3, saniye1, saniye2, saniye3
Dim hs, i, j, jj, k, ks, tks, kpno, tçs, yçs, tkps, pindex, mutoranı, koranı As Integer
Dim mesafe, dssüa, xx, xj, yy, yj, rr, rj, dsağ, dsol, düst, dalt, dij, tdij, ry(101) As Double
Dim oycap, gen, yük, uyum(), uyumYP(), elsts(), elv, itr, maxitr, ls, us, ds, Rs As Double
Dim ras, koor(), koorYP(), tp(), tp1(), kp(), kpts(), kpfo(), ortfo, fsağ, fsol, füst As Double
Dim falt, M2tdijx, M2tdijy, sağboş, solboş, üboş, aboş, elkon, tçalanı As Double
Private Type cember
  no As Integer
  ycap As Double
  talep As Double
End Type
Dim c As cember

Private Sub başlangıç()
   temizle
   saat1 = Hour(Time)
   dakika1 = Minute(Time)
   saniye1 = Second(Time)
   Label10.Caption = "00"
   Label11.Caption = "00"
   Label12.Caption = "00"
   Randomize
   verigirişi
   tks = ks * 2
   ls = -gen / 2
   Rs = gen / 2
   us = yük / 2
   ds = -yük / 2
   Form1.Width = 10380
   Form1.Height = 10380
   Form1.Scale (-Rs - 5, Rs + 1)-(Rs + 5, -Rs - 9)
   ProgressBar1.Visible = True
   ProgressBar1.Min = 0
   ProgressBar1.Max = maxitr
   ReDim uyum(tks)
   ReDim uyumYP(ks)
End Sub

Private Sub Command1_Click()
   başlangıç
   çs = 0
   yçs = 0
   kpno = 1
   Text3 = kpno
   ReDim kp(1 To 100, 1 To 200, 1 To 3)
   ReDim kpts(100)
   ReDim kpfo(100)
```

EK-5. Bilgisayar Programı (devam)

```
Do
    kontrol = False
    Do
        çno = 2 + Int(tçç * Rnd)
        If çno > tçç + 1 Then çno = tçç + 1
        If tp(çno, 2) > 0 Then
            çs = çs + 1
            ry(çs) = tp(çno, 1)
            If çs = 1 Then
                tp(çno, 2) = tp(çno, 2) - 1
            End If
            If çs > 1 Then kontrol = True
        End If
    Loop Until kontrol = True
    ReDim elsts(1 To çs, 1 To 2)
    elv = 1000000
    ReDim koor(tks, 1 To çs, 1 To 2)
    ReDim koorYP(ks, 1 To çs, 1 To 2)
    ReDim Klt(1 To çs, 1 To 4)
    For k = 1 To ks
            k_olustur (k)
    Next
    For itr = 1 To maxitr
        ProgressBar1.Value = itr
        turnuva
        mutasyon
        kaydır
        If elv < oycap Then Exit For
        yenipop
        If itr > maxitr - 5 Then Beep
        If Check1.Value = 0 Then guncelle
        If Check2.Value = 1 Then Exit For
        DoEvents
    Next
    If elv < oycap Then
        If çs = 2 Then
            yçs = yçs + 2
        Else
            yçs = yçs + 1
        End If
        Text1 = yçs
        Txt7 = elv
        tçalanı = 0
        For i = 1 To çs
          tçalanı = tçalanı + 3.14 * ry(i) * ry(i)
        Next
        Text2 = 100 * (gen * yük - tçalanı) / (gen * yük)
        tp(çno, 2) = tp(çno, 2) - 1
        For i = 1 To çs
            kp(kpno, i, 1) = elsts(i, 1)
```

EK-5. Bilgisayar Programı (devam)

```
            kp(kpno, i, 2) = elsts(i, 2)
            kp(kpno, i, 3) = ry(i)
         Next
      End If
      If yçs = tçs Then
         kpfo(kpno) = 100 * (gen * yük - tçalanı) / (gen * yük)
         kpts(kpno) = 1
         Text6 = kpfo(kpno)
         cıkıs
         Exit Do
      End If
      If elv < oycap And çs = tçs Then Exit Do
      If elv > oycap And amalzeme = 1 Then kp_güncelle
   Loop Until elv > oycap And amalzeme = 0
   cıkıs
End Sub

Private Sub kp_güncelle()
         kpts(kpno) = 1
         Text4 = kpts(kpno)
         kpfo(kpno) = 100 * (gen * yük - tçalanı) / (gen * yük)
         kppkon = True
         Do While kppkon = True
            i = 1
            For j = 2 To tçç + 1
               tp1(j, 1) = tp(j, 1)
               tp1(j, 2) = tp(j, 2)
            Next
            Do While kp(kpno, i, 3) <> 0
               For j = 2 To tçç + 1
                  If tp1(j, 1) = kp(kpno, i, 3) Then
                     If tp1(j, 2) <= 0 Then
                        kppkon = False
                        Exit For
                     Else
                        tp1(j, 2) = tp1(j, 2) - 1
                     End If
                  End If
               Next
               If kppkon = False Then Exit Do
               DoEvents
               i = i + 1
            Loop
            If kppkon = True Then
               kpts(kpno) = kpts(kpno) + 1
               Text4 = kpts(kpno)
               i = 1
               Do While kp(kpno, i, 3) <> 0
                  For j = 2 To tçç + 1
                     If tp(j, 1) = kp(kpno, i, 3) Then
```

EK-5. Bilgisayar Programı (devam)

```
                tp(j, 2) = tp(j, 2) - 1
                yçs = yçs + 1
              End If
            Next
            i = i + 1
          Loop
        End If
      Loop
      If yçs = tçs - 1 Then
        cıkıs
        Exit Do
      End If
      kpno = kpno + 1
      Text3 = kpno
      Text4 = 1
      çs = 0
End Sub

Private Sub cıkıs()
  Beep
  If elv > oycap And amalzeme = 0 And Check3.Value = 0 Then
    temizle
    Txt7 = 0
    For j = 1 To çs - 1
      Circle (kp(1, j, 1), kp(1, j, 2)), kp(1, j, 3), 5
    Next
  Else
  End If
  If amalzeme = 1 Then
    temizle
    j = 1
    Do While kp(kpno, j, 3) > 0
      Circle (kp(kpno, j, 1), kp(kpno, j, 2)), kp(kpno, j, 3), 5
      CurrentX = kp(kpno, j, 1) - kp(kpno, j, 3) * 0.2
      CurrentY = kp(kpno, j, 2) + kp(kpno, j, 3) * 0.2
      For i = 2 To tçç + 1
        If kp(kpno, j, 3) = tp(i, 1) Then
          FontSize = 3
          Print i - 1
        End If
      Next
      j = j + 1
    Loop
    ortfo = 0
    tkps = 0
    For i = 1 To kpno
      tkps = tkps + kpts(i)
      ortfo = ortfo + kpfo(i) * kpts(i)
    Next
    ortfo = ortfo / tkps
```

EK-5. Bilgisayar Programı (devam)

```
        Text2 = ortfo
    End If
    ProgressBar1.Visible = False
    Close #1
    kronometre
    Exit Sub
End Sub

Private Sub guncelle()
    temizle
    Txt7 = elv
    Txt7.Refresh
    For j = 1 To çs
        Circle (elsts(j, 1), elsts(j, 2)), ry(j)
    Next
End Sub

Private Sub yenipop()
Dim küçük As Double
Dim küçükinds, elinds As Integer
    For k = 1 To ks
        küçük = 1000000
        For i = 1 To tks
            If uyum(i) < küçük Then
                küçükinds = i
                küçük = uyum(i)
            End If
        Next
        For i = 1 To çs
            koorYP(k, i, 1) = koor(küçükinds, i, 1)
            koorYP(k, i, 2) = koor(küçükinds, i, 2)
        Next
        uyumYP(k) = uyum(küçükinds)
        uyum(küçükinds) = 1000000
    Next
    For k = 1 To ks
        For i = 1 To çs
            koor(k, i, 1) = koorYP(k, i, 1)
            koor(k, i, 2) = koorYP(k, i, 2)
        Next
        uyum(k) = uyumYP(k)
    Next
    elkon = elv
    For k = 1 To ks
        If uyum(k) < elv Then
            elv = uyum(k)
            elinds = k
        End If
    Next
    If elkon <> elv Then
```

EK-5. Bilgisayar Programı (devam)

```
    For k = 1 To çs
      elsts(k, 1) = koor(elinds, k, 1)
      elsts(k, 2) = koor(elinds, k, 2)
    Next
    temizle
    For j = 1 To çs
      Circle (koor(elinds, j, 1), koor(elinds, j, 2)), ry(j)
    Next
    Txt7 = elv
    Txt7.Refresh
  End If
End Sub

Private Sub turnuva()
Dim a, b, c, ksatır As Byte
Dim kazanan As Double
  For a = 1 To ks
    h_uyum (a)
  Next
  If tur > 0 Then
    For a = 1 To ks
      kazanan = 1000000
      For b = 1 To tur
        ras = Int(1 + ks * Rnd)
        If ras > ks Then ras = ks
        If uyum(ras) < kazanan Then
          kazanan = uyum(ras)
          ksatır = ras
        End If
      Next
      uyum(a + ks) = uyum(ksatır)
      For b = 1 To çs
        koor(a + ks, b, 1) = koor(ksatır, b, 1)
        koor(a + ks, b, 2) = koor(ksatır, b, 2)
      Next
    Next
  Else
    kopyala 1, (ks + 1)
  End If
End Sub

Private Sub mutasyon()
  For k = ks + 1 To tks
    değiştir (k)
  Next
End Sub

Private Sub kaydır()
Dim gçm As Byte
Dim mtmp1, mtmp2 As Double
```

EK-5. Bilgisayar Programı (devam)

```
For k = 1 To tks
    gçm = IIf(k > ks, koranı, 1)
    For hs = 1 To gçm
        Txt6 = itr
        Txt6.Refresh
        For j = 1 To çs
            xx = koor(k, j, 1)
            yy = koor(k, j, 2)
            rr = ry(j)
            h_Klt
        Next
        dssüa = 0
        For j = 1 To çs
            If Check2.Value = 1 Then Exit Sub
            DoEvents
            xx = koor(k, j, 1)
            yy = koor(k, j, 2)
            rr = ry(j)
            dç_çakışma
            M2tdijx = 0
            M2tdijy = 0
            fsağ = 0
            fsol = 0
            füst = 0
            falt = 0
            dij = 0
            KltYön(1) = 0
            KltYön(2) = 0
            KltYön(3) = 0
            KltYön(4) = 0
            For jj = 1 To çs
                If jj <> j Then
                    xj = koor(k, jj, 1)
                    yj = koor(k, jj, 2)
                    rj = ry(jj)
                    h_mesafe
                    If mesafe < rr + rj Then
                        dij = rr + rj - mesafe
                        If xx <> xj Or yy <> yj Then
                           mtmp1 = (xx - xj) * dij / mesafe
                           mtmp2 = (yy - yj) * dij / mesafe
                           If xj > xx Then
                               If -mtmp1 > fsağ Then fsağ = -mtmp1
                           End If
                           If xj <= xx Then
                               If mtmp1 > fsol Then fsol = mtmp1
                           End If
                           If yj > yy Then
                               If -mtmp2 > füst Then füst = -mtmp2
                           End If
```

EK-5. Bilgisayar Programı (devam)

```
                If yj <= yy Then
                    If mtmp2 > falt Then falt = mtmp2
                End If
              End If
            Else
              dij = 0
            End If
          End If
        Next
        If fsağ < dsağ Then fsağ = dsağ
        If fsol < dsol Then fsol = dsol
        If füst < düst Then füst = düst
        If falt < dalt Then falt = dalt
        If dsağ > 0 Or KltYön(2) = 1 Then fsol = 0
        If dsol > 0 Or KltYön(1) = 1 Then fsağ = 0
        If düst > 0 Or KltYön(4) = 1 Then falt = 0
        If dalt > 0 Or KltYön(3) = 1 Then füst = 0
        M2tdijx = M2tdijx - fsağ + fsol
        M2tdijy = M2tdijy - füst + falt
        h_boşluk
        If M2tdijx > 0 And M2tdijx > sağboş Then M2tdijx = sağboş
        If M2tdijx < 0 And -M2tdijx > solboş Then M2tdijx = -solboş
        If M2tdijy > 0 And M2tdijy > üboş Then M2tdijy = üboş
        If M2tdijy < 0 And -M2tdijy > aboş Then M2tdijy = -aboş
        koor(k, j, 1) = xx + M2tdijx
        koor(k, j, 2) = yy + M2tdijy
        xx = koor(k, j, 1)
        yy = koor(k, j, 2)
        h_Klt
        düzelt
      Next
      h_uyum (k)
      If uyum(k) < elv Then
        elv = uyum(k)
        For j = 1 To çs
          elsts(j, 1) = koor(k, j, 1)
          elsts(j, 2) = koor(k, j, 2)
        Next
        If Check1.Value = 1 Then
          temizle
          Txt7 = elv
          For j = 1 To çs
            Circle (elsts(j, 1), elsts(j, 2)), ry(j)
          Next
        End If
      End If
      If uyum(k) < oycap Then Exit Sub
    Next
  Next
End Sub
```

EK-5. Bilgisayar Programı (devam)

```
Private Sub değiştir(x As Byte)
Dim geçicix, geçiciy, rasm2, koranıtmp As Double
   koranı = 0
   koranıtmp = 0
   If mutoranı > Int(çs / 2) Then mutoranı = Int(çs / 2)
   For i = 1 To mutoranı
      Do
         rasm2 = 1 + Int(Rnd * çs)
         If rasm2 > çs Then rasm2 = çs
         ras = 1 + Int(Rnd * çs)
         If ras > çs Then ras = çs
         If rasm2 <> ras Then Exit Do
      Loop
      geçicix = koor(x, rasm2, 1)
      geçiciy = koor(x, rasm2, 2)
      koor(x, rasm2, 1) = koor(x, ras, 1)
      koor(x, rasm2, 2) = koor(x, ras, 2)
      koor(x, ras, 1) = geçicix
      koor(x, ras, 2) = geçiciy
      koranıtmp = 10 + Int(Abs(ry(rasm2) - ry(ras)) / (ry(rasm2) + ry(ras)) * 20)
      If koranı < koranıtmp Then koranı = koranıtmp
   Next
End Sub

Private Sub dç_çakışma()
   dsağ = IIf(xx > 0 And xx + rr > Rs, xx + rr - Rs, 0)
   dsol = IIf(xx < 0 And xx - rr < ls, -xx + rr + ls, 0)
   düst = IIf(yy > 0 And yy + rr > us, yy + rr - us, 0)
   dalt = IIf(yy < 0 And yy - rr < ds, -yy + rr + ds, 0)
End Sub

Private Sub h_mesafe()
   mesafe = Sqr((xx - xj) ^ 2 + (yy - yj) ^ 2)
End Sub

Private Sub h_boşluk()
   sağboş = Rs - xx - rr
   solboş = xx - rr + Rs
   üboş = us - yy - rr
   aboş = yy - rr + us
End Sub

Private Sub h_uyum(x As Byte)
Dim tycap As Double
   uyum(x) = 0
   For j = 1 To çs
      xx = koor(x, j, 1)
      yy = koor(x, j, 2)
      rr = ry(j)
```

EK-5. Bilgisayar Programı (devam)

```
        dç_çakışma
        uyum(x) = uyum(x) + dsağ + dsol + düst + dalt
        If j < çs Then
            For jj = j + 1 To çs
                xj = koor(x, jj, 1)
                yj = koor(x, jj, 2)
                rj = ry(jj)
                h_mesafe
                tycap = rr + rj
                If mesafe < tycap Then uyum(x) = uyum(x) + tycap - mesafe
            Next
        End If
    Next
End Sub

Private Sub h_Klt()
   Klt(j, 1) = IIf(xx > 0 And xx + rr >= Rs, 1, 0)
   Klt(j, 2) = IIf(xx < 0 And xx - rr <= ls, 1, 0)
   Klt(j, 3) = IIf(yy > 0 And yy + rr >= us, 1, 0)
   Klt(j, 4) = IIf(yy < 0 And yy - rr <= ds, 1, 0)
End Sub

Private Sub düzelt()
Dim tmp1, tmp2 As Double
    dç_çakışma
    fsağ = 0
    fsol = 0
    füst = 0
    falt = 0
    dij = 0
    For jj = 1 To çs
        If jj <> j Then
            xj = koor(k, jj, 1)
            yj = koor(k, jj, 2)
            rj = ry(jj)
            h_mesafe
            If mesafe < rr + rj Then
                dij = rr + rj - mesafe
                If xx <> xj Or yy <> yj Then
                    tmp1 = (xx - xj) * dij / mesafe
                    tmp2 = (yy - yj) * dij / mesafe
                    If xj > xx And Klt(jj, 1) = 1 Then
                        If -tmp1 > fsağ Then fsağ = -tmp1
                    End If
                    If xj < xx And Klt(jj, 2) = 1 Then
                        If tmp1 > fsol Then fsol = tmp1
                    End If
                    If yj > yy And Klt(jj, 3) = 1 Then
                        If -tmp2 > füst Then füst = -tmp2
                    End If
```

EK-5. Bilgisayar Programı (devam)

```
                If yj < yy And Klt(jj, 4) = 1 Then
                    If tmp2 > falt Then falt = tmp2
                End If
            End If
        Else
            dij = 0
        End If
    End If
Next
koor(k, j, 1) = koor(k, j, 1) - fsağ + fsol
koor(k, j, 2) = koor(k, j, 2) - füst + falt
xx = koor(k, j, 1)
yy = koor(k, j, 2)
h_Klt
End Sub

Private Sub verigirişi()
    çs = Val(Txt1)
    ks = Val(Txt4)
    maxitr = Val(Txt5)
    mutoranı = 1
    tur = Val(Txt9)
    If ks = 1 Then ks = 2
    If amalzeme = 0 Then
        Select Case pindex
            Case 0: Open "c:\hulusi\p11.dat" For Random As 1#
            Case 1: Open "c:\hulusi\p22.dat" For Random As 1#
            Case 2: Open "c:\hulusi\p33.dat" For Random As 1#
            Case 3: Open "c:\hulusi\p44.dat" For Random As 1#
            Case 4: Open "c:\hulusi\p55.dat" For Random As 1#
            Case 5: Open "c:\hulusi\p66.dat" For Random As 1#
        End Select
    Else
        Select Case pindex
            Case 0: Open "c:\hulusi\pr1.dat" For Random As 1#
            Case 1: Open "c:\hulusi\pr2.dat" For Random As 1#
            Case 2: Open "c:\hulusi\pr3.dat" For Random As 1#
            Case 3: Open "c:\hulusi\pr4.dat" For Random As 1#
            Case 4: Open "c:\hulusi\pr5.dat" For Random As 1#
            Case 5: Open "c:\hulusi\pr6.dat" For Random As 1#
            Case 6: Open "c:\hulusi\pr7.dat" For Random As 1#
            Case 7: Open "c:\hulusi\pr8.dat" For Random As 1#
            Case 8: Open "c:\hulusi\pr9.dat" For Random As 1#
            Case 9: Open "c:\hulusi\pr10.dat" For Random As 1#
            Case 10: Open "c:\hulusi\pr11.dat" For Random As 1#
            Case 11: Open "c:\hulusi\pr12.dat" For Random As 1#
            Case 12: Open "c:\hulusi\pr13.dat" For Random As 1#
            Case 13: Open "c:\hulusi\pr14.dat" For Random As 1#
            Case 14: Open "c:\hulusi\pr15.dat" For Random As 1#
            Case 15: Open "c:\hulusi\pgh.dat" For Random As 1#
```

EK-5. Bilgisayar Programı (devam)

```
      End Select
   End If
   tçs = 0
   çs = 0
   tçç = 0
   oycap = 0
   ReDim tp(1 To 101, 1 To 2)
   ReDim tp1(1 To 101, 1 To 2)
   ReDim kp(1 To 1, 1 To 1, 1 To 3)
   For i = 1 To 101
      Get #1, i, c
      If c.ycap <> 0 And c.no > 1 Then
         tçç = tçç + 1
         tçs = tçs + c.talep
         oycap = oycap + c.ycap * c.talep
      End If
      If i = 1 Then
         gen = c.ycap
         yük = c.talep
      End If
      tp(i, 1) = c.ycap
      tp(i, 2) = c.talep
   Next
   oycap = oycap / (tçs * 100)
   Text5 = oycap
   Txt1 = tçs
   Txt2 = gen
   Txt3 = yük
End Sub

Private Sub kopyala(nerden, nereye As Double)
   i = nerden
   For k = nereye To nereye + ks - 1
         For j = 1 To çs
            koor(k, j, 1) = koor(i, j, 1)
            koor(k, j, 2) = koor(i, j, 2)
         Next
         i = i + 1
   Next
End Sub

Private Sub Command3_Click()
   End
End Sub

Private Sub k_olustur(x As Byte)
   For j = 1 To çs
      Randomize
      koor(x, j, 1) = Rs - Rnd * gen
      koor(x, j, 2) = us - Rnd * yük
```

EK-5. Bilgisayar Programı (devam)

```
    Next
End Sub

Private Sub temizle()
    Form1.Cls
    DrawWidth = 4
    Line (ls, us)-(Rs, ds), , B
    DrawWidth = 2
End Sub

Private Sub kronometre()
    saat2 = Hour(Time)
    dakika2 = Minute(Time)
    saniye2 = Second(Time)
    saniye3 = saniye2 - saniye1
    If saniye3 < 0 Then
        saniye3 = saniye3 + 60
        dakika2 = dakika2 - 1
    End If
    dakika3 = dakika2 - dakika1
    If dakika3 < 0 Then
        dakika3 = dakika3 + 60
        saat2 = saat2 - 1
    End If
    saat3 = saat2 - saat1
    Label10.Caption = saat3
    If saat3 = 0 Then Label10.Caption = "00"
    Label11.Caption = dakika3
    If dakika3 = 0 Then Label11.Caption = "00"
    Label12.Caption = saniye3
    If saniye3 = 0 Then Label12.Caption = "00"
End Sub
```