



## New representation schemes for identical parallel machine scheduling problems with sequence dependent setup times

Arda Takan<sup>ID</sup>, Tuğba Saraç\*<sup>ID</sup>

Department of Industrial Engineering, Faculty of Engineering and Architecture, Eskişehir Osmangazi University, 26480, Eskişehir, Türkiye

### Highlights:

- The parallel machine scheduling problem with sequence-dependent setup time is considered.
- New solution representations have been proposed for the addressed problem.
- Developed representation schemes have been especially successful in large-sized problems.

### Keywords:

- Parallel machine scheduling problem
- Representation scheme
- Metaheuristics
- Genetic Algorithm,

### Article Info:

Research Article  
Received: 01.11.2020  
Accepted: 10.05.2022

### DOI:

10.17341/gazimmfd.815942

### Correspondence:

Author: Tuğba Saraç  
e-mail: tsarac@ogu.edu.tr  
phone: +90 530 315 2581

### Graphical/Tabular Abstract

As can be seen from Table A, when the number of jobs is high, and the number of machines is relatively low, GA<sub>1</sub> and GA<sub>2</sub> generate better results than the GA<sub>K</sub>.

Table A. Test Results

no	n	m	GA <sub>K</sub>		GA <sub>1</sub>		GA <sub>2</sub>		GAMS
			t <sub>K</sub>	z <sub>K</sub>	t <sub>1</sub>	z <sub>1</sub>	t <sub>2</sub>	z <sub>2</sub>	
1	8	3	2,10	286,84	3,03	287,73	3,03	288,98	277*
2	15	5	3,74	258,70	4,07	262,25	4,03	259,01	317
3	15	6	3,53	229,42	4,85	231,99	4,90	230,35	306
4	18	8	4,19	255,19	4,84	231,36	4,94	235,99	251
5	19	8	4,43	272,42	5,17	228,54	5,07	233,11	315
6	20	8	4,56	281,04	5,55	249,76	5,66	249,76	294
7	20	9	4,74	266,24	5,56	230,62	5,67	235,23	316
8	20	10	4,82	252,28	5,76	215,95	5,88	211,63	298
9	25	7	5,29	366,61	5,83	326,81	5,95	330,08	371
10	25	8	5,31	333,49	6,01	291,47	6,01	288,56	442

\* optimum

### Purpose:

This study aims to generate effective representation schemes for identical parallel machine scheduling problem (IPMSP) with sequence-dependent setup times.

### Theory and Methods:

In this study, an IPMSP with sequence-dependent setup times, which is significantly crucial in literature, is studied. There are different heuristics and metaheuristics for the problem in the literature. The representation method used in these studies is usually the permutation type representation, where each number corresponds to a job. The order of these numbers represents the order of the jobs in the machines. In this study, a classical genetic algorithm (GA<sub>K</sub>), new solution representations, and two new genetic algorithms (GA<sub>1</sub> and GA<sub>2</sub>) using the proposed solution representations are presented.

### Results:

The performance of the GA<sub>K</sub>, GA<sub>1</sub> and GA<sub>2</sub> are compared by using randomly generated instances to show the success of the proposed representations. The obtained results are given in Table A. The table consists of four parts. In the first part, problem number (no), job number (n) and machine number (m) are included respectively. In the following sections, the objective function values obtained by GA<sub>K</sub>, GA<sub>1</sub>, GA<sub>2</sub> and GAMS (z<sub>K</sub>, z<sub>1</sub>, z<sub>2</sub>, z<sub>G</sub>) and solution times of genetic algorithms (t<sub>K</sub>, t<sub>1</sub>, t<sub>2</sub>), are given.

### Conclusion:

In this study, new solution representations and two new genetic algorithms using the proposed solution representations are presented. Diversification of the solution space is expanded, and the same results are eliminated with these solution representations. Specifically, when the number of jobs is high, and the number of machines is relatively low, the new algorithms generate better results than the classical genetic algorithm for these complex problems.



## Sıra bağımlı hazırlık süreli özdeş paralel makine çizelgeleme problemi için yeni çözüm gösterimleri

Arda Takan<sup>ID</sup>, Tuğba Saraç\*<sup>ID</sup>

Eskişehir Osmangazi Üniversitesi, Mühendislik Mimarlık Fakültesi, Endüstri Mühendisliği Bölümü, 26480, Odunpazarı, Eskişehir, Türkiye

### Ö N E Ç I K A N L A R

- Sıra bağımlı hazırlık süreli paralel makine çizelgeleme problemi ele alınmıştır
- Ele alınan problem için yeni çözüm gösterimleri önerilmiştir
- Geliştirilen çözüm gösterimleri özellikle büyük boyutlu problemlerde başarılı olmuştur

### Makale Bilgileri

Araştırma Makalesi

Geliş: 01.11.2020

Kabul: 10.05.2022

### DOI:

10.17341/gazimmfd.815942

### Anahtar Kelimeler:

Paralel makine çizelgeleme problemi,  
metasezgisel algoritmalar,  
genetik algoritmalar,  
çözüm gösterimi

### ÖZ

Bu çalışmada, literatürde önemli bir yere sahip olan sıra bağımlı hazırlık süreli özdeş paralel makine çizelgeleme problemi (ÖPMÇP) ele alınmıştır. Problemin çözümüne yönelik olarak literatürde farklı sezgisel ve metasezgisel algoritmalar bulunmaktadır. Bu çalışmalarda kullanılan çözüm gösterimi genellikle her bir sayının bir işe karşı geldiği ve bu sayıların sırasının da işlerin makinalardaki sıralarını temsil ettiği permütasyon tipi gösterimdir. Bu çalışmada, ele alınan problem için yeni çözüm gösterimleri önerilmiştir. Önerilen çözüm gösterimlerinin başarısını gösterebilmek amacıyla ilgili probleme yönelik klasik bir genetik algoritma ile yeni önerilen gösterim şekillerini kullanan iki yeni genetik algoritma rassal türetilen test problemleri kullanılarak karşılaştırılmıştır. Önerilen çözüm gösterimleri kullanıldığında çaprazlama esnasında problemin çözüm uzayı açısından, klasik genetik algoritmaya göre çeşitliliğin arttığı ve aynı anlama gelen kromozom yapılarının elendiği gözlemlenmiştir. Özellikle büyük boyutlu problemlerde, geliştirilen çözüm yaklaşımları ile klasik gösterim kullanılan genetik algoritmadan daha başarılı sonuçlara ulaşılmıştır.

## New representation schemes for identical parallel machine scheduling problems with sequence dependent setup times

### H I G H L I G H T S

- The parallel machine scheduling problem with sequence-dependent setup time is considered
- New solution representations have been proposed for the addressed problem
- Developed representation schemes have been especially successful in large-sized problems

### Article Info

Research Article

Received: 01.11.2020

Accepted: 10.05.2022

### DOI:

10.17341/gazimmfd.815942

### Keywords:

Parallel machine scheduling problem,  
metaheuristic algorithms,  
genetic algorithms,  
representation scheme

### ABSTRACT

In this study, an identical parallel machine scheduling problem (IPMSP) with sequence-dependent setup times, which is significantly crucial in literature, is studied. There are different heuristics and metaheuristics for the problem in the literature. The representation method used in these studies is usually the permutation type representation, where each number corresponds to a job. The order of these numbers represents the order of the jobs in the machines. In this study, new solution representations are presented. A classical genetic algorithm and two new genetic algorithms using the proposed solution representations are compared by using randomly generated instances to show the success of the proposed representations. Diversification of the solution space is expanded, and the same results are eliminated with these solution representations. Specifically, the new algorithms generate better results than the classical genetic algorithm for large sized problems.

## 1. Giriş (Introduction)

Çizelgeleme, üretim ve servis sistemlerinde önemli bir karar verme sürecidir. Temel olarak, üretim sürecindeki işlerin hangi kaynaklarla üretilceğini ve bu işlerin makinalara hangi sıra ile atanacağını belirler. Çizelgelemeyi oluşturan üç temel unsur: faaliyet, kaynak ve zamandır.

Üretim çizelgelemede, üretim olanaklarının en etkin şekilde kullanılması çok önemlidir. Müşterilerin taleplerine olabildiğince hızlı bir şekilde cevap verilmelidir. İşlerin, teslim tarihinde gecikme olmadan tamamlanması önemlidir. Ayrıca, yarı mamul stoklarının enküçüklenmesi ve fazla mesai çalışmalarının enküçüklenmesi önemli olan diğer unsurlardandır. Üretimde çizelgelemenin proje çizelgeleme, atölye çizelgeleme, esnek montaj hattı çizelgeleme, tedarik zincirinde çizelgeleme, araçların rotalanması ve çizelgenmesi gibi pek çok uygulaması mevcuttur. Etkin bir çizelgeleme yaklaşımının kullanılması, siparişlerin geç teslim edilmesi, var olan kapasitenin tam kullanılmaması, üretim sürecinin uzaması, üretimde yaşanan darboğazlar, müşteri memnuniyetsizliği ve müşteri kaybı gibi sorunların çözülmesine yardımcı olur.

Üretim çizelgeleme problemleri literatürde genel olarak dört farklı grupta incelenmektedir. Bunlar; tek makine çizelgeleme, paralel makine çizelgeleme, atölye tipi ve akış tipi çizelgeleme problemleridir. Bu çalışma kapsamında sıra bağımlı hazırlık süreli özdeş paralel makine çizelgeleme problemi (ÖPMÇP) ele alınmıştır. Paralel makine çizelgeleme problemlerinin temel parametreleri hazırlık ve işlem süreleridir. Hazırlık süresi, makinenin iş başlamadan önce üretime hazırlanması sürecidir. Bu süreç, kullanılacak aletlerin temini, malzeme sürecindeki işin pozisyonlanması ve gerekli kalibrelerin kurulumu gibi aşamaları içerebilir. Hazırlık süreleri uzunca bir süre yapılan çalışmalarda göz ardı edilmiştir ya da üretim zamanının bir parçası olarak görülmüştür. Bu durum bazı çizelgeleme problemleri için anlamlı olabilir ancak farklı durumlar için ayrı hazırlık süresi gerekebilir. Hazırlık süreleri iki tür olarak karşımıza çıkmaktadır. Birinci türde, hazırlık süresi sadece yapılacak olan işe bağlıdır bu yüzden sıra bağımsız hazırlık süresi olarak adlandırılır. İkinci türde ise, hazırlıklar hem işlem göreceğe işe hem de bir önceki işe bağlıdır. Bu hazırlık süresi türüne ise sıra bağımlı hazırlık süresi denir [1]. Hazırlık süresinin önemi farklı çalışmalarda ortaya konmuştur [2-4].

Bu çalışmanın motivasyonu, sıra bağımlı hazırlık süreli ÖPMÇP'nin daha etkin bir şekilde çözülebilmeye olanak yaratabilecek yeni çözüm gösterimleri geliştirmektir. Bu amaçla, ele alınan problemin çözümüne yönelik iki yeni çözüm gösterimi ve bu gösterimleri kullanan genetik algoritmalar geliştirilmiştir. Önerilen çözüm gösterimlerinin, literatürde daha önce çalışılmamış olması bu çalışmanın özgün yönünü oluşturmaktadır. Literatürde yer alan klasik çözüm gösterimleri ile aynı çözüme karşı gelen birçok farklı kromozom türeyebildiği ve bu durumun arama etkinliğini azalttığı tespit edilmiştir. Tekrarlı çözümleri önleyecek şekilde geliştirilen çözüm gösterimlerini kullanan genetik algoritmaların özellikle büyük boyutlu problemlerde, başarılı olduğu rassal türetilen test problemleri kullanılarak gösterilmiştir. Çalışmanın izleyen bölümünde literatür taramasına yer verilmiştir. Üçüncü bölümde, ele alınan problemin matematiksel modeli sunulmuştur. Dördüncü bölümde sıra bağımlı hazırlık süreli ÖPMÇP'ne yönelik klasik çözüm gösterimi ve klasik bir genetik algoritma (GA<sub>K</sub>), beşinci bölümde yeni bir çözüm gösterim şekli ve GA (GA<sub>1</sub>), altıncı bölümde ise kromozomun ikinci kısmının permutasyon tipi olduğu ikinci bir gösterim biçimi ve GA (GA<sub>2</sub>) verilmiştir. Yedinci bölümde elde edilen deneysel bulgular tartışılmıştır. Son bölümde ise sonuç ve öneriler yer almaktadır.

## 2. Literatür Taraması (Literature Review)

Literatürde ÖPMÇP'yi ele alan pek çok çalışma ([5-27]) mevcuttur. Min ve Cheng [5] son işin tamamlanma zamanının enküçüklenmesini amaçladıkları çalışmalarında, özdeş paralel makine çizelgeleme problemini ele almışlardır ve ele alınan problemin çözümü için bir genetik algoritma geliştirmişlerdir. Tahar vd. [6] sıra bağımlı hazırlık süreli ve iş bölümlü ÖPMÇP'de son işin tamamlanma zamanının enküçüklenmesi için doğrusal programlamaya dayalı bir sezgisel algoritma önermişlerdir. Lee vd. [7] ÖPMÇP'de son işin tamamlanma zamanının enküçüklenmesi için eniyi çözüme yakın çözümler üretebilen bir tavlama benzetimi algoritması önermişlerdir. Biskup vd. [8] ÖPMÇP'de toplam gecikmenin enküçüklenmesi için etkin bir sezgisel algoritma geliştirmişlerdir. Tanaka ve Araki [9] özdeş paralel makine çizelgeleme probleminde toplam gecikmenin enküçüklenmesi için daha sıkı alt sınırlar elde edebilmek amacıyla Lagrange gevşetme yöntemi önermişlerdir. Chang ve Chen [10] son işin tamamlanma zamanının enküçükleyecek şekilde ilişkisiz paralel makine çizelgeleme problemini ele almışlardır. Büyük problemler için çözüm kalitesini daha fazla iyileştirmek için baskınlık özelliklerini genetik algoritma ile entegre ederek yeni bir metasezgisel yöntem önerilmiştir. Wang ve Cheng [11] iki özdeş paralel makine çizelgeleme problemini ele almışlardır. Zamanında biten iş sayısının enbüyüklenmesi amacıyla bir sezgisel algoritma geliştirmişlerdir. Beezão vd. [12] belirli işlem süreli ve araç gereksinimli özdeş paralel makine çizelgeleme problemini araştırmışlardır. Probleme yönelik iki matematiksel model ve uyarlanabilir geniş komşuluk algoritmasını önermişlerdir. Labbi vd. [13] hazırlık kısıtlı özdeş paralel makine çizelgeleme problemini çalışmışlardır. Hazırlık zamanlarının farklı ele alındığı iki durumu incelemişlerdir ve probleme yönelik sezgisel bir algoritma geliştirmişlerdir. Chaudhry ve Elbadawi [14] işlerin gecikmesinin enküçüklenmesini amaçlayan özdeş paralel makine çizelgeleme problemini ele almışlardır. Probleme yönelik bir hesap tablosu tabanlı bir genetik algoritma geliştirmişlerdir. Wang vd. [15] hem toplam enerji tüketiminin hem de son işin tamamlanma zamanının enküçüklenmesini amaçlandığı özdeş paralel makine çizelgeleme problemini ele almışlardır. Küçük boyutlu problemler için genişletilmiş  $\epsilon$ -kısıt yöntemini, orta ve büyük boyutlu problemler için ise yerel arama stratejisi ve NSGA-II algoritmasını önermişlerdir. Lee [16] toplam gecikmenin enküçüklenmesini amaçlandığı özdeş paralel makine çizelgeleme probleminde rasgele yinelenen ağaçlı bir algoritma üzerinde çalışmışlardır. Kim vd. [17] sıra bağımlı hazırlık süreli ve iş bölümlü ÖPMÇP için yeni bir matematiksel model ve metasezgisel yaklaşımlar (tavlama ve genetik algoritma) önermişlerdir. Lee ve Kim [18] Sıra bağımlı hazırlık süreli, iş bölümlü ve sınırlı sayıdaki hazırlık operatörlü ÖPMÇP üzerine çalışmışlardır. Problemin çözümü için bir matematiksel model ve hazırlık sürelerine dayalı iteratif bir sezgisel algoritma geliştirmişlerdir. Silva vd. [19], ortak sunuculu (*common server*) ve sıra bağımlı hazırlık süreli özdeş paralel makine çizelgeleme problemini ele almışlardır. Geliştirilmiş kod çözme algoritmasına dayalı iteratif yerel arama algoritmasını geliştirmişler ve test problemleri üzerinde kıyaslamalar yapmışlardır. Ozer ve Saraç [20], ortak kaynakların kullanıldığı ÖPMÇP'ni ele almışlardır. Tamamlanma zamanının ağırlıklı toplamlarının enküçüklenmesi amacıyla sahip problem için karma tamsayı matematiksel modeller ve model tabanlı bir genetik algoritma önermişlerdir. Izquierdo vd. [21], ÖPMÇP'deki hazırlık aşamasında operatör yorgunluğunun (öğrenme) etkileri üzerine çalışmışlardır. İlk aşamada değişken komşuluk arama algoritması kullanmışlardır daha sonra ise öğrenmenin veya yorgunluğun, elde edilen çözümler üzerindeki etkilerini gerçek hayattan alınan bir problem üzerinde, çok etmenli bir benzetim yaklaşımı kullanarak incelemişlerdir. Ghalami ve Grosu [22] ÖPMÇP için en iyi polinom zamanlı yaklaşım yaklaşımına dayalı bir paralel yaklaşım algoritması önermişlerdir. Laha ve Gupta [23]

ÖPMÇP için guguklu arama algoritması (*cuckoo search algorithm*) önermişlerdir. Branda [24] belirsiz bitiş zamanlı ÖPMÇP üzerine çalışmışlardır. Ağ akışı formülasyonları, probleme yönelik etkili ayrıştırma algoritmaları oluşturmak için kullanılmıştır. Jager [25] öncelik kısıtlı ve hazır olma zamanlı ÖPMÇP'ni araştırmışlardır. İşlerin toplam ağırlıklandırılmış tamamlanma zamanının enküçüklenmesi için zaman indeksli doğrusal programlama gevşeme yöntemini geliştirmişlerdir. Schwerdfeger ve Walter [26] toplam iş yükü dengelenmesi kriterinin enküçüklenmesi amacıyla sahip özdeş paralel makine çizelgeleme problemini ele almışlardır. Problem için

kesin çözüm algoritması ve sezgisel algoritma geliştirilerek çözüm kalitesi ve çözüm süresi açısından geliştirilen yöntemin üstün yönleri belirtilmiştir. Ouazene ve Yalaoui [27] zamana bağlı işlem süreli özdeş paralel makine çizelgeleme problemini çalışmışlardır. Zamana bağlı işlem süreli iki paralel makine problemi için toplam tamamlanma süresini en aza indiren kesin çözüm algoritması önermişlerdir. Ele alınan problem ile ilgili literatür ayrıca Tablo 1'de de verilmiştir. Tablo 1'den de görülebileceği gibi, ÖPMÇP odaklı çalışmaların önemli bir kısmında ele alınan amaç fonksiyonu son işin tamamlanma zamanının enküçüklenmesidir. Genetik algoritmalar

**Tablo 1.** ÖPMÇP ile ilgili literatür (Literature for IPMSP)

Makale	Problem	Amaç fonksiyonu	Çözüm yöntemi	Gösterim biçimi
Min ve Cheng [5]	ÖPMÇP	Son işin tamamlanma zamanı	GA	Makine kod tabanlı gösterim
Tahar vd. [6]	Sıra bağımlı hazırlık süreli ve iş bölünmeli ÖPMÇP	Son işin tamamlanma zamanı	Sezgisel algoritma + doğrusal programlama	-
Lee vd. [7]	ÖPMÇP	Son işin tamamlanma zamanı	Tavlama benzetimi	-
Biskup vd. [8]	ÖPMÇP	Toplam gecikme	Sezgisel algoritma	-
Tanaka ve Araki [9]	ÖPMÇP	Toplam gecikme	Lagrange gevşetmesi	-
Chang ve Chen [10]	ÖPMÇP	Son işin tamamlanma zamanı	GA ve tavlama benzetimi	Rassal sıralama
Wang ve Cheng [11]	ÖPMÇP	Zamanında biten iş sayısı	Sezgisel algoritma	-
Beezão vd. [12]	Araç gereksinimli ÖPMÇP	Son işin tamamlanma zamanı	Komşu arama algoritması	-
Labbi vd. [13]	ÖPMÇP	Son işin tamamlanma zamanı	Sezgisel algoritma	-
Chaudhry ve Elbadawi [14]	ÖPMÇP	Toplam gecikme	GA	İlk kısımda işler, ikinci kısımda ise makineler
Wang vd. [15]	ÖPMÇP	Toplam enerji tüketimi ve son işin tamamlanma zamanı	Yerel arama, NSGA-II algoritması	Vektör kromozom
Lee [16]	ÖPMÇP	Toplam gecikme	Açgözlü algoritma	-
Kim vd. [17]	Sıra bağımlı hazırlık süreli ve iş bölünmeli ÖPMÇP	Toplam gecikme	Tavlama benzetimi ve GA	Metin kodlama yöntemi
Lee ve Kim [18]	Sıra bağımlı hazırlık süreli, iş bölünmeli ve tek sunuculu ÖPMÇP	Son işin tamamlanma zamanı	Hazırlık sürelerine dayalı iteratif bir sezgisel algoritma	-
Silva vd. [19]	Ortak sunuculu ve sıra bağımlı hazırlık süreli ÖPMÇP	Son işin tamamlanma zamanı	yerel arama	-
Ozer ve Sarac [20]	Ortak kaynakların kullanıldığı ÖPMÇP	Toplam ağırlıklı tamamlanma zamanı	Matematiksel model tabanlı GA	2 x n-bit tamsayılı matris gösterimi
Izquierdo vd. [21]	ÖPMÇP	Toplam tamamlanma zamanı	Değişken komşuluk arama algoritması	-
Ghalami ve Grosu [22]	ÖPMÇP	Son işin tamamlanma zamanı	Paralel yaklaşım algoritması	-
Laha ve Gupta [23]	ÖPMÇP	Son işin tamamlanma zamanı	Guguklu arama algoritması	-
Branda [24]	Belirsiz bitiş zamanlı ÖPMÇP	Risk altındaki koşullu değer maliyet	Ayrıştırma algoritması	-
Jager [25]	Öncelik kısıtlı ve hazır olma zamanlı ÖPMÇP	İşlerin toplam ağırlıklandırılmış tamamlanma zamanı	Zaman indeksli doğrusal programlama gevşeme yöntemi	-
Schwerdfeger and Walter [26]	ÖPMÇP	Toplam iş yükü dengelenmesi	Kesin çözüm algoritması ve sezgisel algoritma	-
Ouazene ve Yalaoui [27]	Zamana bağlı işlem süreli ÖPMÇP	Son işin tamamlanma zamanı	İki Makine-Optimal algoritması	-

ÖPMÇP'nin çözümünde başarı ile uygulanmıştır [5, 10, 14, 15, 17, 20]. Bu çalışmada kullanılan çözüm gösterimleri genellikle her bir sayının bir işe karşı geldiği ve bu sayıların sırasının da işlerin makinalardaki sıralarını temsil ettiği permütasyon tipi gösterimdir.

Bu çalışma kapsamında, literatürde daha önce kullanılmamış iki yeni çözüm gösterimi ve bu gösterimleri kullanan iki genetik algoritma geliştirilmiştir. Önerilen çözüm gösterimleri tekrarlanan çözümleri engellediğinden, özellikle çözüm uzayının büyük olduğu büyük boyutlu problemlere, klasik çözüm gösterimlerinin kullanıldığı yöntemlere kıyasla daha başarılı çözümlerin bulunması fırsatını sunmaktadır. Bu çalışmada literatüre sağlanan katkı genetik algoritma özelinde gösterilmiş olsa da tüm metasezgisel algoritmalar için aynı avantajı sağlama potansiyeline sahip olması nedeniyle literatürde önemli bir boşluğu doldurmuştur.

### 3. Matematiksel Model (Mathematical Model)

Ele alınan problemde,  $n$  işten her birisi, aynı özelliklere sahip  $m$  makineden birisine atanmalıdır. Her makinenin ilk işi, işe bağlı bir hazırlık süresine, ikinci ve sonraki işler ise kendisine ve kendisinden önceki işe bağlı (sıra bağımlı) bir hazırlık süresine gereksinim duymaktadır. Tüm işler sıfır zamanında hazırır. İşlerin bölünmesine izin verilmemektedir. Her makine aynı anda sadece bir iş gerçekleştirebilmektedir. Amaç, son işin tamamlanma zamanının enküçüklenmesidir. Sıra bağımlı hazırlık süreli özdeş paralel makine çizelgeleme problemine ait matematiksel model aşağıda, bu modelde kullanılan indis, parametre ve karar değişkenlerinin tanımları ise Tablo 2'de verilmiştir.

**Tablo 2.** İndisler, Parametreler ve Karar Değişkenleri (Parameters and Decision Variables)

İndisler	
$i, j, k \in N = \{1, 2, \dots, n\}$	$i$ ve $j$ iş, $k$ sıra indisi
$l \in M = \{1, 2, \dots, m\}$	$l$ makine indisi
Parametreler	
$n$	iş sayısı
$m$	makine sayısı
$p_j$	$j$ . işin işlem süresi
$h_j$	eğer $j$ . iş ilk sıradaysa hazırlık süresi
$s_{ij}$	eğer $i$ . iş $j$ 'den bir önceki iş ise, $j$ . işin sıra bağımlı hazırlık süresi
$M$	yeterince büyük pozitif bir sayı
Karar değişkenleri	
$C_j$	$l$ . makinede $j$ . işin tamamlanma zamanı
$C_{enb}$	son işin tamamlanma zamanı
$y_{jkl}$	eğer $j$ . iş $l$ . makinede $k$ . sıraya atanırsa 1, diğer durumda 0
$z_{ijkl}$	eğer $j$ . iş $l$ . makinede $k$ . sıraya atanırsa ve $i$ . işten hemen sonra ise 1, diğer durumda 0

Amaç fonksiyonu:

$$\text{enk } z = C_{enb} \quad (1)$$

Kısıtlar:

$$C_j \geq p_j \quad \forall j \in N \quad (2)$$

$$C_j + M(1 - y_{j1l}) \geq h_j + p_j \quad \forall i \neq j, k = 1, i, j, k \in N, l \in M \quad (3)$$

$$C_j - C_i + M(1 - z_{ijkl}) \geq s_{ij} + p_j \quad \forall i \neq j, k > 1, i, j, k \in N, l \in M \quad (4)$$

$$1 + z_{ijkl} \geq y_{ik-1l} + y_{jkl} \quad \forall i \neq j, k > 1, i, j, k \in N, l \in M \quad (5)$$

$$\sum_{j \in N} y_{jkl} \leq 1 \quad \forall k \in N, l \in M \quad (6)$$

$$\sum_{k \in N} \sum_{l \in M} y_{jkl} = 1 \quad \forall j \in N \quad (7)$$

$$C_{enb} \geq C_j \quad \forall j \in N, l \in M \quad (8)$$

$$y_{jkl} \in \{0,1\} \quad \forall j, k \in N, l \in M \quad (9)$$

$$z_{ijkl} \in \{0,1\} \quad \forall i, j, k \in N, l \in M \quad (10)$$

$$C_j \geq 0 \quad \forall j \quad (11)$$

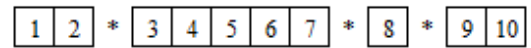
$$C_{enb} \geq 0 \quad (12)$$

Amaç (1) son tamamlanan işin tamamlanma zamanının enküçüklenmesidir. Kısıt (2),  $j$ . işin tamamlanma zamanının  $j$ . işin işlem zamanından büyük ya da eşit olmasını garanti eder. (3) numaralı kısıt, her  $l$  makinesi için ilk sıradaki işin tamamlanma zamanını hesaplar. Kısıt (4),  $l$ . makinenin  $k$ . sırasındaki işin ( $k > 1$ ) tamamlanma zamanını hesaplar. Kısıt (5),  $z_{ijkl}$  ve  $y_{jkl}$  karar değişkenleri arasındaki ilişki kısıtıdır. Kısıt (6)  $l$ . makinedeki her sıraya sadece bir işin atanabileceğini belirtir. Kısıt (7), bir işin bir makinenin sadece bir sırasına atanabileceğini gösterir. Kısıt (8), son işin tamamlanma zamanını hesaplar. Kısıt (9)-(12) işaret kısıtlarıdır.

### 4. Sıra Bağımlı Hazırlık Süreli ÖPMÇP İçin Klasik Bir Çözüm Gösterimi ve Klasik Bir GA (GA<sub>K</sub>) (A Classical Solution Representation and a Classical GA (GA<sub>K</sub>) for IPMSP With Sequence Dependent Setup Times)

GA ilk olarak Holland [28] tarafından önerilmiştir. Genetik algoritmanın güncel formu ise Goldberg [29] tarafından geliştirilmiştir. GA, doğal seçime (kalıtıma) dayalı stokastik arama tekniğidir ve popülasyon tabanlı bir metasezgisel algoritmadır.

*Klasik Çözüm Gösterimi.* Cheng ve Gen [30]'in önerdiği klasik GA'da tanımlanan kromozom gösteriminde  $n$  iş  $m$  makine söz konusu olduğunda kromozom uzunluğu  $n+m-1$  olmaktadır. İş numaraları kromozomda aynen kullanılıyorken makineler \* karakteri ile birbirinden ayrılmaktadır. Birinci \*'a kadar olan işler birinci makineye atanırları, ikinci \*'a kadar olan işler ise ikinci makineye atanırları ve son \*'dan sonra olan işler ise son makineye atan işleri simgeler. On iş, dört makinenin olduğu bir problem için örnek bir kromozom Şekil 1'de gösterilmiştir.

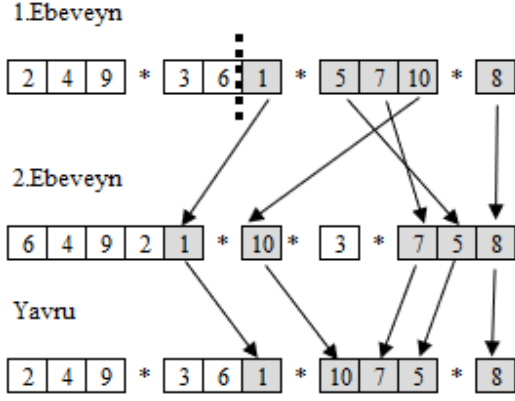


**Şekil 1.** Klasik çözüm gösterimi (classical solution representation)

Şekil 1'de verilen klasik çözüm gösterimine sahip kromozom, birinci makinada sırasıyla iş 1 ve iş 2'nin, ikinci makinada sırasıyla iş 3, iş 4, iş 5, iş 6 ve iş 7'nin, üçüncü makinada iş 8'in ve dördüncü makinada sırasıyla iş 9 ve iş 10'nun işlem göreceği bir çözüme karşı gelmektedir.

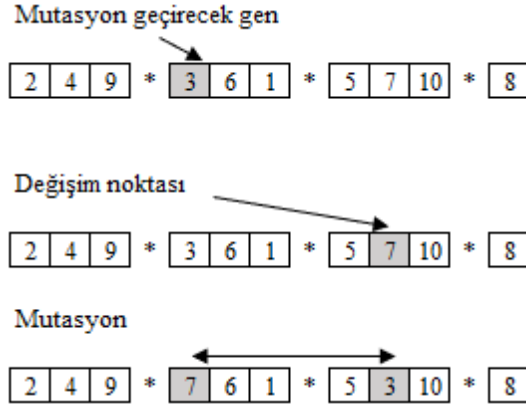
*Seçim.* Seçim için ikili turnuva yöntemi kullanılmıştır. Bu yöntemde popülasyon içerisindeki rassal olarak iki kromozom seçilir ve uyum değerleri kıyaslanır. Uyum değeri daha iyi olan birey çaprazlama için seçilir.

**Çaprazlama.** Cheng ve Gen [30]'in önerdiği klasik GA'da kullanılan çaprazlama operatörü Şekil 2'de gösterilmiştir. Çaprazlama için kromozom üzerinde en küçük 1 en fazla  $n+m-1$  olacak şekilde, rassal bir nokta seçilir. Birinci kromozomda bu noktaya kadar olan değerler aynen alınır. Bu noktadan sonraki işler ise ikinci kromozomdaki sıralarına göre yerleştirilir.



**Şekil 2.** Klasik çaprazlama operatörü  
(Classical crossover operator)

**Mutasyon.** Permütasyon tipi kromozomlarda literatürde mutasyon operatörü olarak genellikle *swap* ve *insert* yöntemleri kullanılmaktadır. Bu çalışma kapsamında her iki yöntem de kullanılmıştır. *Swap* mutasyon operatöründe her gene bir mutasyon şansı verilir. Başarılı olan gen için rassal bir nokta seçilerek bu nokta ile yerleri değiştirilir. Bu operatörlerin nasıl çalıştıkları Şekil 3 ve Şekil 4'de birer örnek kromozom üzerinde gösterilmiştir.



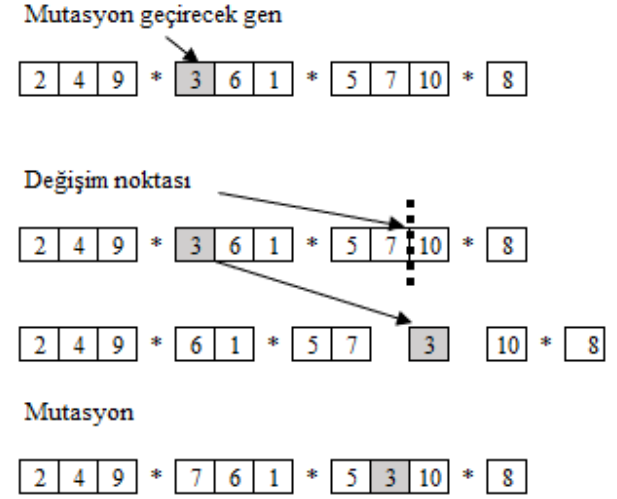
**Şekil 3.** *swap* mutasyonu (*swap* mutation)

*Insert* mutasyon operatöründe her gene bir mutasyon şansı verilir. Başarılı olan gen için rassal bir noktaya taşınarak aradaki genler kaydırılır.

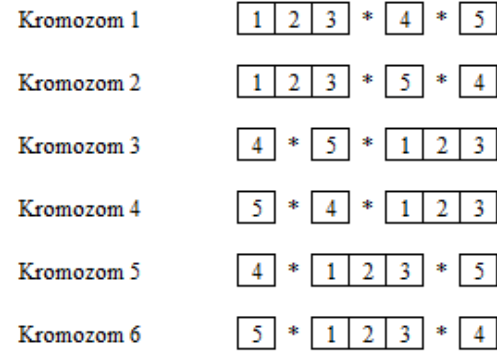
### 5. Sıra Bağımlı Hazırlık Süreli ÖPMÇP İçin Geliştirilen Yeni Bir Çözüm Gösterimi ve GA (GA<sub>1</sub>) (A New Solution Representation and a GA (GA<sub>1</sub>) for IPMSP with Sequence Dependent Setup Times)

Cheng ve Gen [30]'in kullandığı gösterim biçiminde iki kromozomun çaprazlanması esnasında ilk kromozomdaki makinelere atanan işler sabit tutulduğundan yeterli değişiklik sağlanamayabilmektedir. Bu noktada bir iyileştirme yapabilmek adına çözüm gösterimi üzerinde değişiklik yapılabileceği düşünülmüştür. Bu nedenle, her çaprazlama

ve mutasyon sonrası kromozomun uygun çözüm olup olmadığının kontrolünü yapmak ya da özel çaprazlama ve mutasyon operatörleri kullanmak yerine her çaprazlamanın sonucunun uygun bir çözümü tarif ettiği ve çaprazlamaların makine atamalarının üzerinde de değişiklik sağladığı bir gösterim yöntemi geliştirilmiştir. Özdeş paralel makineler ile çalışıldığından işlerin hangi makinaya atandığının bir önemi yoktur işlerin atandıkları makinelerdeki sırası önemlidir. Örneğin  $n=5$   $m=3$  için 1-2-3-4-5 işlerini göz önüne alındığında Şekil 5'te verilmiş olan altı kromozom birbirleriyle eşdeğerdir. Ancak klasik gösterimde bu kromozomların her birisi ayrı ayrı türeyebilmekte ve değerlendirmeye tabi tutulmaktadır.



**Şekil 4.** *insert* mutasyonu (*insert* mutation)



**Şekil 5.** Eşdeğer çözüme sahip kromozomlar  
(Chromosomes with equivalent solution)

**Yeni bir Çözüm Gösterimi.** Bu çalışmada iki parçalı yeni bir çözüm gösterimi geliştirilmiştir. Kromozomun ilk parçası  $m$  adet ikinci parçası  $n$  adet genden oluşmaktadır. İlk parçada her makinaya kaç iş atandığı ikinci parçada ise işlerin nasıl sıralandığı bilgisi yer almaktadır. Üç makine ve beş iş olduğu bir problem için örnek bir kromozom Şekil 6'da verilmiştir. Şekil 6'dan da görülebileceği gibi birinci makinaya 3 iş ikinci ve üçüncü makinalara ise birer iş atanmıştır. Ayrıca birinci makinede iş sırası iş 1, iş 2 ve iş 3 şeklindedir. İkinci makineye iş 4 ve üçüncü makinaya da iş 5 atanmıştır.

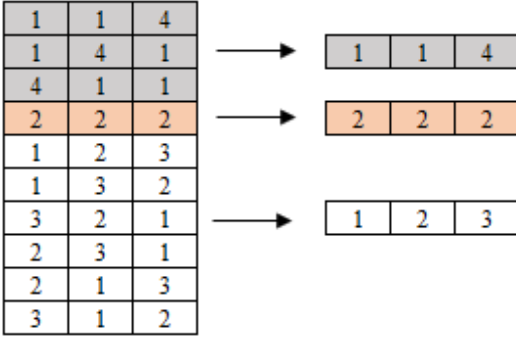


**Şekil 6.** Yeni bir çözüm gösterimi  
(A new solution representation)



Ancak Şekil 6'da verilen yeni çözüm gösterimi yine klasik gösterimde olduğu gibi GA'nın tek noktalı, iki noktalı vb. klasik çaprazlama ve mutasyon operatörlerinin kullanımına uygun değildir. Uygunluğu bozmayacak uygulamalara muhtaçtır. Bu problemi giderebilmek için çözüm gösteriminin her iki parçasının da geliştirilmesi gerekmektedir.

$n$  iş  $m$  makine özdeş paralel makine çizelgeleme probleminde, Şekil 6'da verilen gösterim biçimi kullanıldığında kromozomun ilk parçasının hangi farklı değerleri alabileceği bellidir. Örneğin 6 makine 3 iş için Şekil 7'de gösterildiği gibi makinelere atanan iş sayılarının kombinasyonlarının sayısı 10 adettir. Ancak özdeş makinalar söz konusu olduğunda 1 1 4, 1 4 1 ve 4 1 1, çözümlerinin hepsi eşdeğerdir. Burada hangi makineye 4 hangi makinalara birer iş atandığından ziyade önemli olan bir makineye 4 diğerlerine 1'er iş atanmış olması ve bu işlerin atandıkları makinalarda nasıl sıralandıklarıdır. Bu açıdan bakıldığında görülebileceği gibi Şekil 7'deki ilk üç kombinasyon eşdeğerdir. Benzer şekilde diğer eşdeğer çözümlerde gruplandırıldığında eşdeğer olmayan çözüm sayısı sadece 3'tür. Yeni geliştirilen gösterim şeklinde, sadece bu 3 çözümü değerlendirecek biçimde bir çözüm gösterimi geliştirilmiştir. Böylece aynı anlama gelen ancak farklı gözükten kromozomların tekrar tekrar türetilerek zaman kaybedilmesinin önüne geçilmiş aynı çözüme denk gelen fazladan çözümlerin elenmesiyle daha az sayıda kromozom türeterek çözüm uzayının taranmasına olanak yaratılmıştır.



Şekil 7. 6 iş 3 makine problemi için atama kombinasyonları (Assignment combinations for 6 job-3 machine problem)

Herhangi  $n$  iş  $m$  makine için, sadece eşdeğer olmayan makine iş sayıları kombinasyonlarını bulacak bir algoritma geliştirilmiştir. Geliştirilen algoritmanın adımları aşağıda verilmiştir.

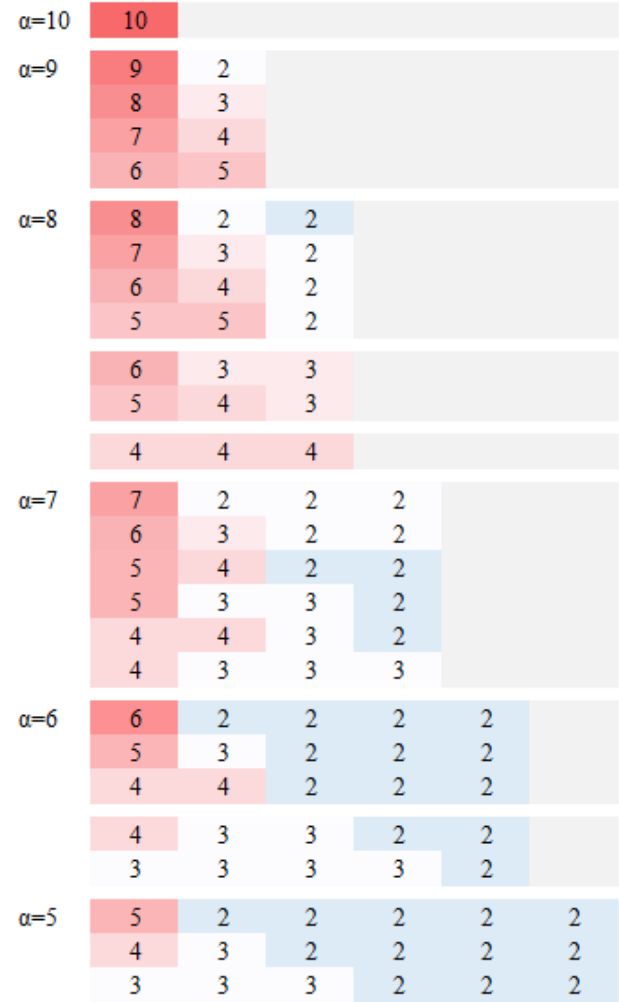
- Adım 1:  $\alpha = n - (m-1)$  olarak hesapla
- Adım 2:  $m$  kadar sütun türet, güncellenecek sütun sayısını 1 olarak belirle
- Adım 3: Güncellenecek sütuna  $\alpha$  değerini ata,  $\alpha$ 'yı 1 azalt
- Adım 4: Eğer artırılan sütun değerinin 1 fazlası, ilk sütun değerinin 1 eksiğinden küçükse arttırabildiğin ilk sütunu 1 arttır.
- Adım 5: Arttıracak sütun yoksa arttırılacak sütunların sayısını 1 arttır,  $\alpha = \alpha - 1$  olarak güncelle, Adım 3'e git.
- Adım 6: Eğer arttırılacak sütun yoksa dur.

Burada  $\alpha$ , referans değer,  $n$ , iş sayısı ve  $m$ , makine sayısıdır.

Şekil 8'de 15 iş 6 makine problemi için algoritmanın türettiği makine iş sayısı kombinasyonları gösterilmiştir.

Şekil 9'da iş sıralarının aynı kaldığı düşünüldüğünde 15 iş 6 makine için 26 adet makine iş sayısı kombinasyonu vardır. 26 bitlik verinin tamamını taşıyıp, gereksiz veri barındırmayacak ikilik sistemde bir kromozom tasarımı söz konusu değildir. İkilik sistemde oluşturulan kromozomlar 2'nin kuvvetlerine eşit bit değerine sahip çözümleri tamamen temsil etmektedir. Şekil 9'da olduğu gibi bu duruma

uymayan değerlerin de ikilik sistemde bir kromozomla temsil edilebilmesi için öncelikle çözümlerin eşit temsiliyetini bozmamak adına büyük boyutlu bir kromozom seçilmiştir. Bu kromozom  $2^0$  ile  $2^{12}$  arasındaki bütün değerleri içermektedir. 1-26 arasındaki çözüm değerlerine dönüştürülmesi için aşağıdaki kromozom temsiliyet algoritması oluşturulmuştur.



Şekil 8. 15 iş 6 makine problemi için her  $\alpha$  değeri için türetilen makine iş sayısı kombinasyonları (combinations of machine job numbers generated for each  $\alpha$  value for 15 job-6 machine problem)

Kromozomun temsil edileceği ikilik sistem dönüşümü için bir algoritma geliştirilmiştir. Geliştirilen algoritmanın yapısı aşağıda verilmiştir.

- Adım 1: Olası makine iş sayısı kombinasyonlarının sayısını bul
- Adım 2: İkilik sistemde eşit temsiliyeti bozmayacak bir boyut seç
- Adım 3: Bu boyuttaki kromozomun alabileceği en fazla değeri olası çözüm sayısına böl
- Adım 4: Rassal olarak kromozomlar oluşturulmaya başlandığında çıkan değeri Adım 3'te hesaplanan değere böl, yukarı yuvarla
- Adım 5: Adım 4'te elde edilen değer karşılığında kromozomun ilgili makine atamasını bul.

Bu algoritma sayesinde her zaman kromozom ile elde edilecek değer, istenen aralıkta olacaktır. Çözümlerin bu yöntemle eşit temsil edilip edilmediğini kontrol etmek amacı ile 1-26 arasında 20704 adet rassal

sayı türetilmiştir. Bu sayılar kromozom olarak düşünüldüğünde her benzersiz çözüm başına 796,31 kromozom düşmelidir. Dönüştürmeli türetimde ise 1-4095 arasında sayılar türetilip  $4095/26 = 157,5$  sayısına bölünmüş ve yukarı yuvarlanmıştır. Tablo 3'de rassal (normal) türetim ve dönüştürmeli türetim ile elde edilen 20704 adet çözüm, ortalamaları ve standart sapmaları Şekil 10'da ise kromozom temsiliyet dağılım grafiği verilmiştir. Bu tabloya göre rassal olarak değerlerin eşit temsil edildiği söylenebilir.

1	10	1	1	1	1	1
2	9	2	1	1	1	1
3	8	3	1	1	1	1
4	7	4	1	1	1	1
5	6	5	1	1	1	1
6	8	2	2	1	1	1
7	7	3	2	1	1	1
8	6	4	2	1	1	1
9	5	5	2	1	1	1
10	6	3	3	1	1	1
11	5	4	3	1	1	1
12	4	4	4	1	1	1
13	7	2	2	2	1	1
14	6	3	2	2	1	1
15	5	4	2	2	1	1
16	5	3	3	2	1	1
17	4	4	3	2	1	1
18	4	3	3	3	1	1
19	6	2	2	2	2	1
20	5	3	2	2	2	1
21	4	4	2	2	2	1
22	4	3	3	2	2	1
23	3	3	3	3	2	1
24	5	2	2	2	2	2
25	4	3	2	2	2	2
26	3	3	3	2	2	2

**Şekil 9.** Türetilen Tüm Makine İş Sayısı Kombinasyonları  
(Generated Whole Machine Job Combinations)

Kromozomun ikinci kısmında çaprazlama işleminin bozulmaya yol açmasını önlemek amacıyla rassal anahtar gösterimi tercih edilmiştir [31]. Bu gösterimde  $[0,1]$  arasında iş sayısı kadar rassal sayı türetilir. Örneğin 6 iş için rassal değerler türetildiğinde ve küçükten büyüğe sıralandığında en küçük iş 1 numaralı iş ve en büyük iş 6 nolu iş olacak biçimde Tablo 4'deki gibi sıralanır. Sıralama işlemi Hoare [32] tarafından geliştirilen hızlı sıralama algoritması (*Quick Sort*) kullanılarak yapılmıştır.

Böylece kromozomun uyum fonksiyonu değeri Şekil 11'de gösterildiği gibi hesaplanır. Öncelikle kromozomun ilk kısmından ikilik sistemin, onluk sistemdeki karşılığı hesaplanır. Dönüştürme algoritmasına göre karşı gelen sıra değeri bulunur. Bu sıra değeri ile, problemin ilk aşamasında kullanılan kombinasyon bulma algoritmasının sonuçlarından ilgili sıra no karşılığı olan makine kombinasyonu elde edilir. Bu kombinasyon ve işler kullanılarak amaç fonksiyonu değeri hesaplanır.

**Seçim.** Seçim operatörü olarak klasik genetik algoritmada olduğu gibi ikili turnuva yöntemi kullanılmıştır.

**Tablo 3.** Temsiliyet frekans tablosu (Representation frequency table)

Çözüm No	Normal Türetim	Dönüştürmeli Türetim
1	812	822
2	792	767
3	793	742
4	817	790
5	779	736
6	783	825
7	816	804
8	819	752
9	787	792
10	820	819
11	762	818
12	827	781
13	783	832
14	827	791
15	802	785
16	754	798
17	868	820
18	794	800
19	792	790
20	774	801
21	803	841
22	830	772
23	790	824
24	727	819
25	809	777
26	744	806
<i>Standart Sapma</i>	29,33	26,77
<i>Ortalama</i>	796,31	796,31

**Çaprazlama.** Kromozom yeni gösterimi ile permütasyon tipi bir kromozom olmadığı için herhangi bir düzeltme işlemine gereksinim duyulmadan doğrudan klasik tek noktalı ya da iki noktalı çaprazlama operatörleri kullanılabilir. Bu çalışmada tek noktalı çaprazlama operatörü kullanılmıştır. Örnek bir kromozom için çaprazlama operatörünün nasıl çalıştığı Şekil 12 ve 14'de gösterilmektedir. Şekillerde görüleceği gibi, kromozomun ilk kısmında makine sıralarını simgeleyen ikilik sistemdeki sıralar, ikinci kısımda işleri simgeleyen rassal anahtar yöntemi ile oluşturulmuş sıra yer almaktadır. Şekil 12'de çaprazlama noktası ilk makine parçasına denk gelmekte Şekil 14'de ise iş sıralarına denk gelmektedir. Ancak her iki durumda da kesilme noktasından oluşan kromozom uygun bir çözümü sağlamaktadır. Çaprazlama noktasının kromozomun ilk kısmına denk geldiği durum Şekil 12'de gösterilmektedir. Şekil 13 ve Şekil 15'te ise çaprazlamalar sonucu oluşan çocuk kromozomlar verilmiştir.

**Tablo 4.** Rassal anahtar dönüşümü (Random key conversion)

Rassal Anahtar	0,61	0,54	0,02	0,76	0,95	0,32
Karşı Gelen İş No	4	3	1	5	6	2

Bu çaprazlama sonucunda oluşan çocuk kromozom Şekil 13'de gösterilmiştir.

Çaprazlama için kesim noktasının kromozomun iş dizisi kısmına denk geldiği durum Şekil 14'de gösterilmektedir.

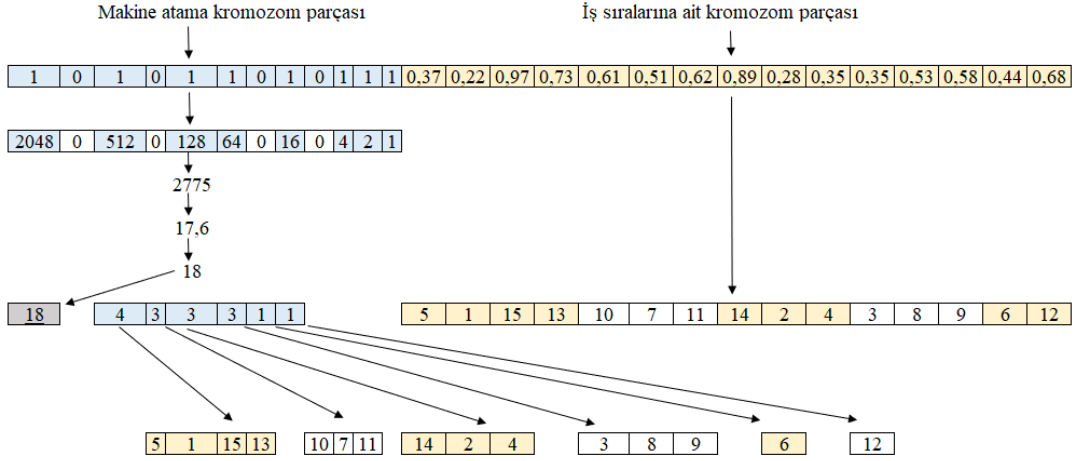
Bu çaprazlama sonucunda oluşan çocuk kromozom Şekil 15'de gösterilmektedir.

**Mutasyon.** Mutasyon işlemi, genlerin değerlerinin rassal olarak değiştirilmesiyle sağlanmaktadır. Bu nedenle bu mutasyon rassallaştırma mutasyonu olarak adlandırılmıştır. Eğer mutasyona uğrayacak gen makinelere atanan iş sayılarının verildiği kromozomun

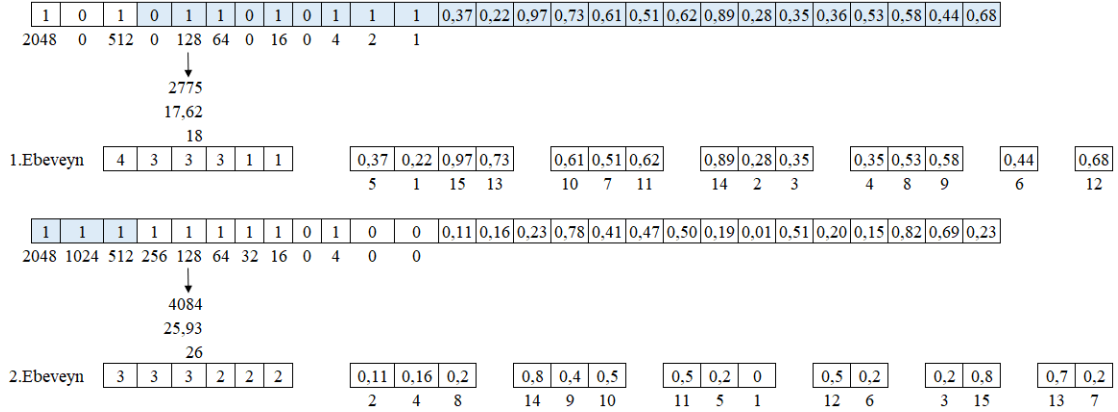




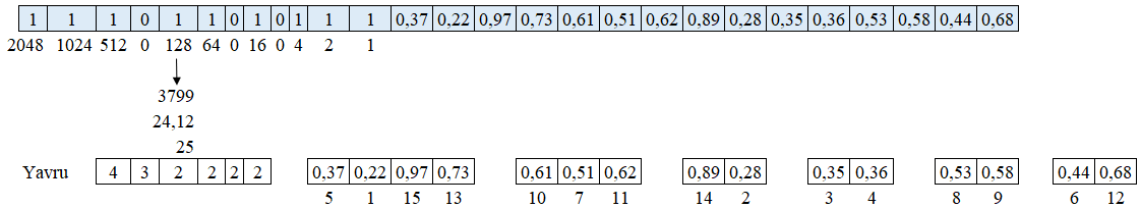
Şekil 10. Kromozom Temsiliyet Dağılım Grafiği (Chromosome Representation Distribution Plot)



Şekil 11. Önerilen yeni çözüm gösterimi (Proposed new solution representation)



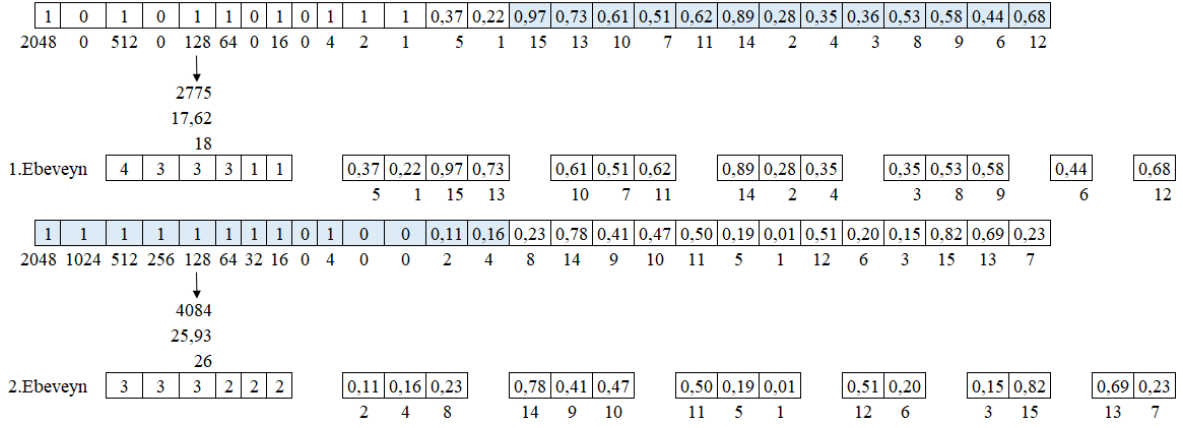
Şekil 12. Çaprazlama noktasının kromozomun ilk kısmında olduğu durum (The situation where the crossover point is in the first part of the chromosome)



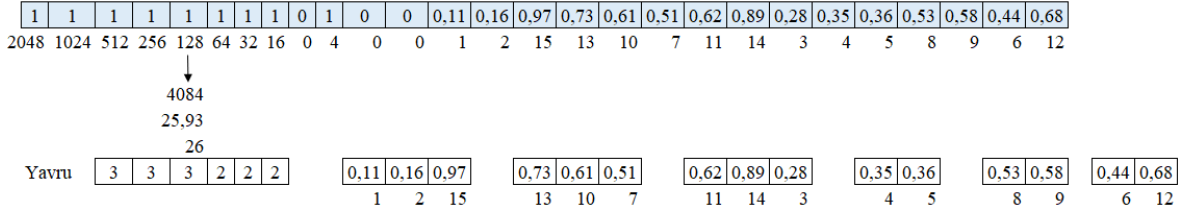
Şekil 13. Yeni gösterim şekli ve tek noktalı çaprazlama kullanılarak elde edilen çocuk kromozom (Child chromosome obtained using new representation and single point crossover)

ilk parçasında yer alıyorsa, değeri 0 ise 1, 1 ise 0 olarak değiştirilir. Eğer mutasyona uğrayacak gen kromozomun ikinci kısmında yer alıyorsa ilgili genin değeri, [0,1] aralığında türetilen yeni bir rassal sayı ile değiştirilir. Şekil 16 ve 17’de kromozomun her iki parçasında

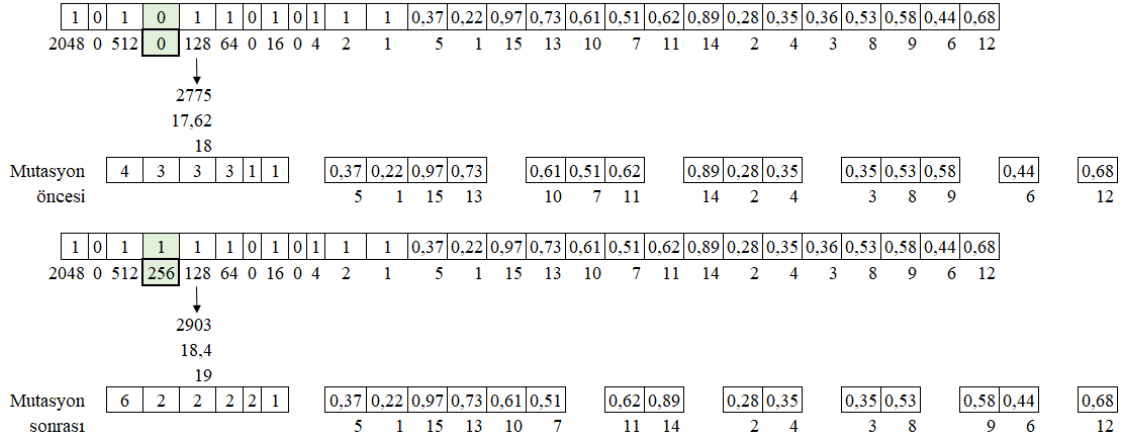
yer alan genlerin rassallaştırma mutasyona nasıl uğradığı gösterilmiştir. Rassallaştırma mutasyonuna ek olarak, iş atamalarındaki değişkenliği arttırmak için *swap* operatörü de kullanılmıştır. Şekil 18’de bu operatöre ait örnek verilmiştir.



Şekil 14. Çaprazlama noktasının kromozomun ikinci kısmında olduğu durum  
(The situation where the crossover point is in the second part of the chromosome)



Şekil 15. Yeni gösterim şekli ve tek noktali çaprazlama kullanılarak elde edilen çocuk kromozom  
(Child chromosome obtained using new solution representation and single point crossover)



Şekil 16. GA<sub>1</sub>'de kromozomun ilk kısmında rassallaştırma mutasyonu (Randomization mutation in the first part of the chromosome in GA<sub>1</sub>)

## 6. Kromozomun İkinci Kısmının Permütasyon Tipi Olduğu İkinci Bir Gösterim Biçimi ve GA (GA<sub>2</sub>) (A Second Solution Representation Where the Second Part of the Chromosome is The Permutation Type and A Ga (Ga<sub>2</sub>))

*İkinci Çözüm Gösterim Şekli.* Önerilen ikinci çözüm gösterimi de bölüm 4'te önerilen çözüm gösteriminde olduğu gibi iki parçalıdır ve makinalara atanan iş sayılarının yer aldığı ilk kısım her iki çözüm gösteriminde de aynıdır. Bu gösterimde işlerin makinalarda hangi sırada işlem göreceğini gösteren kromozomun ikinci kısmı permütasyon tipidir.

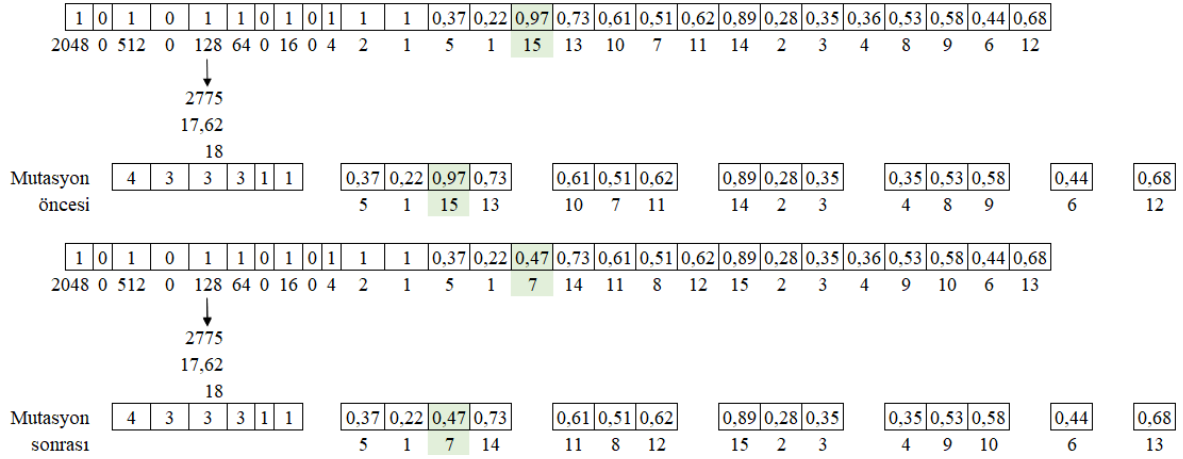
*Seçim.* Seçim operatörü olarak ikili turnuva yöntemi kullanılmıştır.

*Çaprazlama.* Kromozomun ikinci parçası permütasyon tipinde olduğundan, Cheng ve Gen [30]'in önerdiği klasik GA' da açıklanan çaprazlama operatörü kullanılmıştır. Çaprazlama için kromozom

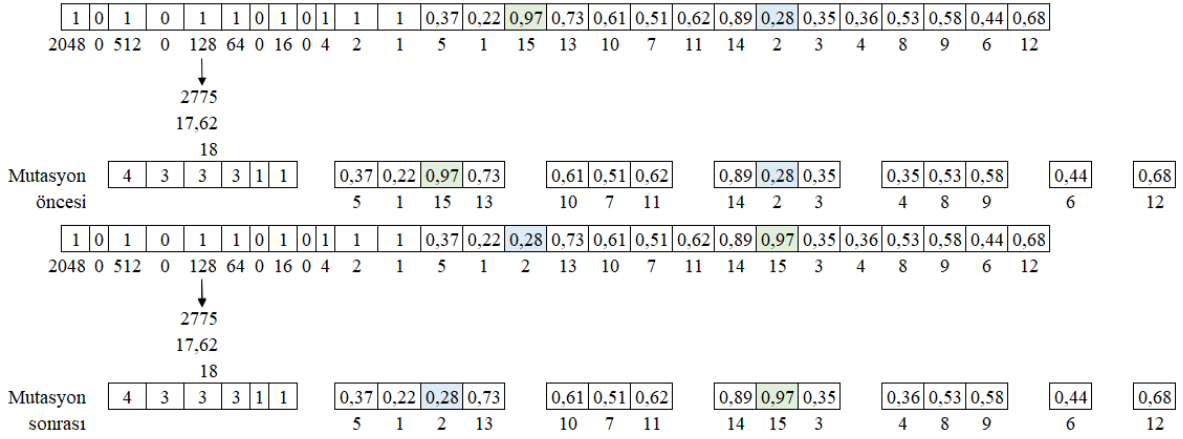
üzerinde en küçük 1 en fazla  $n+m-1$  olacak şekilde, rassal bir nokta seçilir. Birinci kromozomda bu noktaya kadar olan değerler aynen alınır. Bu noktadan sonraki işler ise ikinci kromozomdaki sırası ile yerleştirilir. Önerilen çaprazlama yöntemine örnek Şekil 20'de ve çaprazlama sonucu oluşan yeni kromozom Şekil 21'de verilmiştir.

*Mutasyon.* Mutasyon operatörü olarak klasik genetik algorithmada olduğu gibi *swap* mutasyon ve *insert* mutasyon yöntemleri kullanılmaktadır. Swap işlemi yalnızca kromozomun iki bölgesinden birinde yapılabilir. Yani makine sıraları ve iş sıraları arasında bir *swap* işlemi yapılamaz. Her iki mutasyona örnekler sırasıyla Şekil 22 ve 23'de verilmiştir.

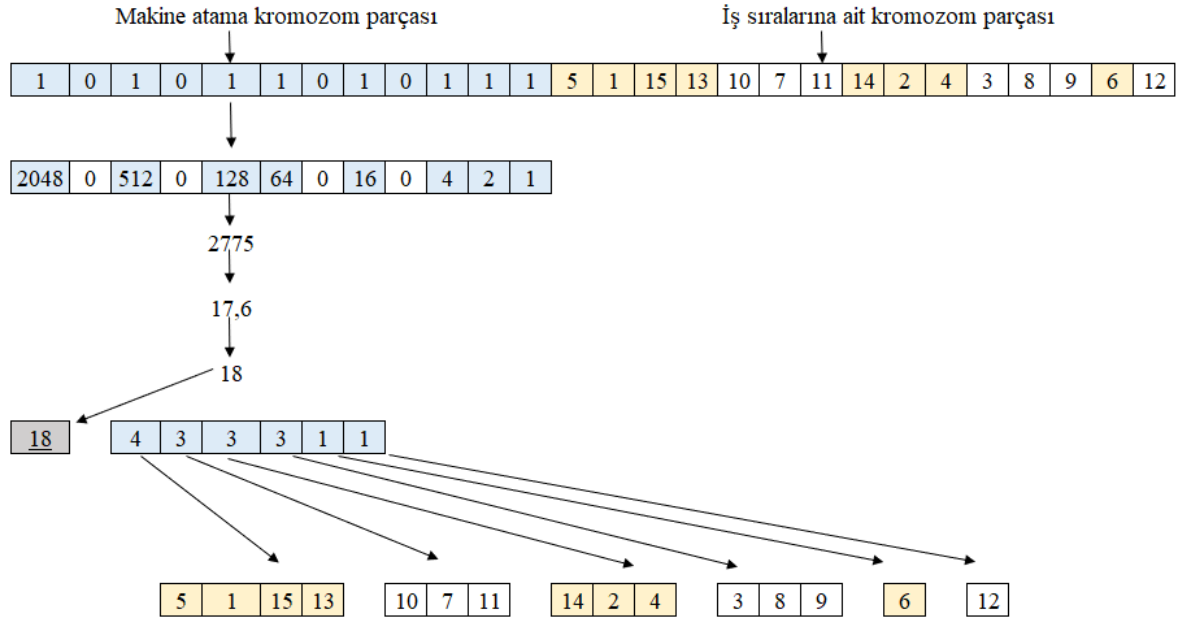
*Insert* işlemi de yalnızca kromozomun iki bölgesinden birinde yapılabilir. Yani makine sıraları ve iş sıraları arasında bir *insert* işlemi yapılamaz.



Şekil 17. GA<sub>1</sub>'de kromozomun ikinci kısmında rassallaştırma mutasyonu (Randomization mutation in the second part of the chromosome in GA<sub>1</sub>)



Şekil 18. GA<sub>1</sub>'de kromozomun ikinci kısmında swap mutasyonu (Swap mutation in the second part of the chromosome in GA<sub>1</sub>)



Şekil 19. İkinci çözüm gösterimi için örnek bir kromozom (An example chromosome for second solution representation)



**Tablo 5.** Test sonuçları (Test results)

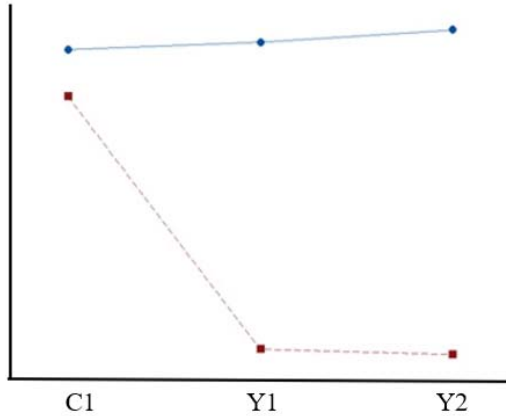
no	n	m	GA <sub>K</sub>		GA <sub>1</sub>		GA <sub>2</sub>		GAMS
			t <sub>K</sub>	z <sub>K</sub>	t <sub>1</sub>	z <sub>1</sub>	t <sub>2</sub>	z <sub>2</sub>	z <sub>G</sub>
1	8	3	2,10	286,84	3,03	287,73	3,03	288,98	277*
2	15	5	3,74	258,70*	4,07	262,25	4,03	259,01	317
3	15	6	3,53	229,42*	4,85	231,99	4,90	230,35	306
4	18	8	4,19	255,19	4,84	231,36*	4,94	235,99	251
5	19	8	4,43	272,42	5,17	228,54*	5,07	233,11	315
6	20	8	4,56	281,04	5,55	249,76*	5,66	249,76*	294
7	20	9	4,74	266,24	5,56	230,62*	5,67	235,23	316
8	20	10	4,82	252,28	5,76	215,95	5,88	211,63*	298
9	25	7	5,29	366,61	5,83	326,81*	5,95	330,08	371
10	25	8	5,31	333,49	6,01	291,47	6,01	288,56*	442

\*eniye çözüm

son sütunda ise GAMS ile elde edilen amaç fonksiyonu değeri (z<sub>G</sub>) verilmiştir. Örneğin 9 nolu problem GA<sub>K</sub> ile ortalama 5,29 saniyede çözülmüş ve ortalama 366,61 değerine ulaşılmıştır. GA<sub>1</sub> ile ortalama 5,83 saniye de ortalama 326,81, GA<sub>2</sub> ile ise ortalama 5,95 saniyede ortalama 330,08 değeri elde edilmiştir. GAMS ise süre limiti içinde ancak 371 değerine ulaşılabilmiştir. Her bir problem için elde edilen en başarılı amaç fonksiyonu değerleri "\*" ile işaretlenmiştir. Tablo 5'ten de görülebileceği gibi, GAMS/Cplex ile sadece 1. problem için, GA<sub>K</sub> ile de sadece 2. ve 3. problemler için başarılı amaç fonksiyonu değerleri elde edilebilmiştir. Problem boyutu arttıkça bir başka değişle çözüm uzayı büyüdükçe önerilen çözüm gösterimlerinin tekrarlayan çözümleri ayıklaması, daha başarılı çözümlere ulaşılabilmesini mümkün kılmıştır. 18 iş, 8 makine ve daha büyük boyutlu problemler dikkate alındığında önerilen çözüm gösterimlerini kullanan yaklaşımlar, GA<sub>K</sub>'dan ortalama %14 daha başarılı amaç fonksiyonu değerleri elde etmişlerdir. Minitab programı ile veriler analiz edilerek kullanılan yöntemlerin ve problem boyutunun sonuca olan etkisi araştırılarak varyans analizi yapılmıştır. Elde edilen varyans analizi tablosu Tablo 6'da verilmiştir. Güven seviyesi %95 olduğunda ( $\alpha = 0.05$ ) yöntemin, boyutun ve yöntemle-boyutun etkileşiminin (Yöntem\*Boyut) etkileri incelenmiştir. Bu üç faktörün  $p$  değeri  $\alpha = 0.05$  değerinden küçük olduğundan aralarındaki fark anlamlıdır.

**Tablo 6.** Varyans analizi (Analysis of variance)

Kaynak	DF	Adj SS	Adj MS	F-Değeri	P-Değeri
Yöntem	2	29899	14950	79,80	0,000
Boyut	1	117488	117488	627,12	0,000
Yöntem*Boyut	2	37040	18520	98,85	0,000
Hata	594	111284	187		
Toplam	599	295711			

**Şekil 24.** Yöntem, boyut ve sonuç ilişkisi (Method, dimension and result relationship)

Yöntem, boyut ve sonuç ilişkisini gösterebilmek amacıyla Şekil 24'de ilişki grafiği verilmiştir. Burada C1, GA<sub>K</sub>'yı, Y1, GA<sub>1</sub>'i, Y2 ise GA<sub>2</sub>'yi ifade etmektedir. Grafikten de görülebileceği gibi yeni çözüm gösterimlerinin kullanan GA<sub>1</sub> ve GA<sub>2</sub> algoritmaları klasik çözüm gösteriminin kullandığı GA<sub>K</sub>'dan özellikle büyük boyutlu problemlerde oldukça başarılıdır.

## 8. Sonuçlar (Conclusions)

Özdeş paralel makine çizelgeleme problemleri endüstride ve gerçek hayatta sıkça karşılaşılan bir problemdir. Literatürde bu probleme yönelik farklı çalışmalar bulunmaktadır. Bu çalışmanın motivasyonu literatürde sıklıkla kullanılan metasezgisel algoritmaların performanslarını arttıracak şekilde yeni çözüm gösterimleri geliştirilmesidir. Geliştirilen çözüm gösterimlerinin başarısını gösterebilmek için literatürde sıklıkla kullanılmış olan klasik çözüm gösterimi ve klasik bir genetik algoritma ile bu çalışmada geliştirilen iki yeni çözüm gösterimini içeren iki yeni genetik algoritma karşılaştırılmıştır. Genelde klasik kromozom yapısında aynı anlama gelen ifadeler de algoritma içerisinde değerlendirilmektedir. Problem özdeş makinaları içerdiğinden aslında yapılan bu işlemin gereksiz olduğu ve aynı anlama gelen kromozomların elenmesinin sağlandığı bu yeni çözüm gösterimleri ile daha başarılı çözümlere erişilebileceği gösterilmiştir. Önerilen çözüm gösterimleri özellikle büyük boyutlu problemlerde, klasik çözüm gösteriminin kullandığı genetik algoritmaya kıyasla amaç fonksiyonu değerini ortalama %14 iyileştirmiştir. Her ne kadar karşılaştırma genetik algoritma özelinde yapıldıysa da önerilen çözüm gösterimi özdeş paralel makinalar için geliştirilecek her türlü sezgisel ve metasezgisel algoritmada kullanılabilir. Bu çalışma, özdeş paralel makine çizelgeleme probleminin metasezgisel algoritmalarla çözümleri aranırken tekrarlayan/aynı anlama gelen çözümleri eleyerek çözüm uzayında daha etkin bir arama gerçekleştirilmesini sağladığından literatüre önemli bir katkı sağlamaktadır. Gelecek çalışmalarda, farklı sezgisel ve metasezgisel algoritmaların önerilen çözüm gösterimini kullandıklarında performanslarına olan etkisi incelenebilir. Ya da bu çalışma kapsamında ele alınan problem farklı kısıtlar ya da farklı amaç fonksiyonları eklenerek daha karmaşık ve çok amaçlı versiyonlarında da yeni çözüm gösteriminin problemin çözüm performansına etkileri araştırılabilir.

## Kaynaklar (References)

1. Allahverdi A., Gupta J., Aldowaisan T., A review of scheduling involving setup considerations, Omega The International Journal of Management Science, 27, 219-239, 1999.
2. Xi Y., Jang J., Scheduling jobs on identical parallel machines with unequal future ready time and sequence dependent setup: An experimental study, International Journal of Production Economics, 137 (1), 1-10, 2012.
3. Yang W.H., Liao C.J., Survey of scheduling research involving setup times, International Journal of Systems Science, 30, 143-155, 1999.

4. Lee Y., Pinedo M., Scheduling Jobs on Parallel Machines with Sequence Dependent Setup Times, *European Journal of Operational Research*, 100 (3), 464-474, 1997.
5. Min L., Cheng W., A genetic algorithm for minimizing the makespan in the case of scheduling identical parallel machines, *Artificial Intelligence in Engineering*, 13 (4), 399-403, 1999.
6. Tahar D.N., Yalaoui F., Chu C., Amodeo L., A Linear Programming Approach for Identical Parallel Machine Scheduling with job Splitting and Sequence-Dependent Setup Times, *International Journal of Production Economics*, 99 (1), 63-73, 2006.
7. Lee W.C., Wu C.C., Chen P., A Simulated Annealing Approach to Makespan Minimization on Identical Parallel Machines, *The International Journal of Advanced Manufacturing Technology*, 31 (3-4), 328-334, 2006.
8. Biskup D., Herrmann J., Gupta J.N.D., Scheduling Identical Parallel Machines to Minimize Total Tardiness, *International Journal of Production Economics*, 115 (1), 134-142, 2008.
9. Tanaka S., Araki M., A Branch-and-Bound Algorithm with Lagrangian Relaxation to Minimize Total Tardiness on Identical Parallel Machines, *International Journal of Production Economics*, 113 (1), 446-458, 2008.
10. Chang P.C., Chen S.H., Integrating Dominance Properties with Genetic Algorithms for Parallel Machine Scheduling Problems with Setup Times, *Applied Soft Computing*, 11 (1), 1263-1274, 2011.
11. Wang X., Cheng T.C.E., A Heuristic for Scheduling Jobs on two Identical Parallel Machines with a Machine Availability Constraint, *International Journal of Production Economics*, 161, 74-82, 2015.
12. Beezão A. C., Cordeau J.F., Laporte G. Yanasee H.H., Scheduling Identical Parallel Machines with Tooling Constraints, *European Journal of Operational Research*, 257 (3), 834-844, 2017.
13. Labbi W., Boudhar M., Oulamara A., Scheduling Two Identical Parallel Machines with Preparation Constraints, *International Journal of Production Research*, 55 (6), 1531-1548, 2017.
14. Chaudhry I.A., Elbadawi I.A., Minimisation of total tardiness for identical parallel machine scheduling using genetic algorithm, *Sadhana*, 42 (1), 11-21, 2017.
15. Wang S., Wang X., Yu J., Ma S., Liu M., Bi-objective identical parallel machine scheduling to minimize total energy consumption and makespan, *Journal of Cleaner Production*, 193, 424-440, 2018.
16. Lee C.H., A Dispatching Rule and a Random Iterated Greedy Metaheuristic for Identical Parallel Machine Scheduling to Minimize Total Tardiness, *International Journal of Production Research*, 56 (6), 2292-2308, 2018.
17. Kim J.G., Song S., Jeong B., Minimising total tardiness for the identical parallel machine scheduling problem with splitting jobs and sequence-dependent setup times, *International Journal of Production Research*, 5 (6), 1628-1643, 2020.
18. Lee, J.H., Kim, H.J., A heuristic algorithm for identical parallel machine scheduling: splitting jobs, sequence-dependent setup times, and limited setup operators, *Flexible Services and Manufacturing Journal*, DOI: 10.1007/s10696-020-09400-9, 2018.
19. Silva, J.M.P., Teixeira, E., Subramanian, A., Exact and metaheuristic approaches for identical parallel machine scheduling with a common server and sequence-dependent setup times, *Journal of The Operational Research Society*, 72 (2), 444-457, 2021.
20. Ozer, E.A., Sarac, T., MIP models and a matheuristic algorithm for an identical parallel machine scheduling problem under multiple copies of shared resources constraints, *TOP*, 27, 94-124, 2019.
21. Izquierdo, C.E., Bello, F.A., Batista, B.M., Alvarez, A., Báez, S., A metaheuristic algorithm and simulation to study the effect of learning or tiredness on sequence-dependent setup times in a parallel machine scheduling problem, *Expert Systems with Applications*, 117, 62-74, 2019.
22. Ghalami, L., Grosu, D., Scheduling parallel identical machines to minimize makespan: A parallel approximation algorithm, *Journal of Parallel and Distributed Computing*, 133, 221-231, 2019.
23. Laha, D., Gupta, J.N.D., An improved cuckoo search algorithm for scheduling jobs on identical parallel machines, *Computers & Industrial Engineering*, 126, 348-360, 2018.
24. Branda, M., Distributionally robust fixed interval scheduling on parallel identical machines under uncertain finishing times, *Computers and Operations Research*, 98, 231-239, 2018.
25. Jager, S., Approximating total weighted completion time on identical parallel machines with precedence constraints and release dates, *Operations Research Letters*, 46 (5), 505-509, 2018.
26. Schwerdfeger, S., Walter, R., Improved algorithms to minimize workload balancing criteria on identical parallel machines, *Computers & Operations Research*, 93, 123-134, 2018.
27. Ouazene, Y., Yalaoui, F., Identical parallel machine scheduling with time-dependent processing times, *Theoretical Computer Science*, 721, 70-77, 2018.
28. Holland J. H., *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*, Ann Arbor, MIT Press Cambridge, 1992.
29. Goldberg D.E., *Genetic algorithms in search, optimization, and machine learning*, Reading, MA: Addison-Wesley, 1989.
30. Cheng R., Gen M., Minmax earliness/tardiness scheduling in identical parallel machine system using genetic algorithm, *International Journal of Computers and Industrial Engineering*, 29 (1-4), 513-517, 1995.
31. Bean J., Genetic algorithms and random keys for sequencing and optimization, *ORSA Journal on Computing*, 6(2), 154-160, 1994.
32. Hoare C., Algorithm 64: Quicksort, *Communications of the ACM*, 4 (7), 321, 1961.
33. Saraç T., Tutumlu B., A bi-objective mathematical model for an unrelated parallel machine scheduling problem with job-splitting, *Journal of the Faculty of Engineering and Architecture of Gazi University*, 37 (4), 2293-2308, 2022.
34. Saraç T., Tutumlu B., A mix integer programming model and solution approach to determine the optimum machine number in the unrelated parallel machine scheduling problem. *Journal of the Faculty of Engineering and Architecture of Gazi University*, 37 (1), 329-345, 2022.