



İki paralel enjeksiyon makinasının kreyn kısıtı altında çizelgelenmesi

Gülçin Bektur, Tuğba Saraç*

Eskişehir Osmangazi Üniversitesi, Mühendislik Fakültesi, Endüstri Mühendisliği, 26480, Eskişehir, Türkiye

Ö N E Ç I K A N L A R

- Kreyn kısıtlı paralel makine çizelgeleme
- Matematiksel model
- Genetik algoritma

Makale Bilgileri

Geliş: 24.03.2015

Kabul: 22.09.2016

DOI:

10.17341/gazimmfd.278445

Anahtar Kelimeler:

Tek ekipli paralel makine çizelgeleme, sıra bağımlı hazırlık zamanı, makine uygunluk kısıtları, genetik algoritma, matematiksel model

ÖZET

Bu çalışmada iki paralel enjeksiyon makinasının kreyn kısıtı altında çizelgelenmesi problemi ele alınmıştır. Enjeksiyon makinalarıyla üretimin hazırlık aşamasında, üretilecek ürüne ait kalıbın makinaya bağlanması gerekmektedir. Kalıplar ilgili makinalara kreyn ile taşınmaktadır. Makinalar tek bir kreyni paylaştığı için hazırlık zamanları çakışmamalıdır. Ayrıca bir işin hazırlık zamanı kendisinden önceki işe bağlı olarak farklılık göstermektedir ve her iş her makinada üretilememektedir. Amaç, son işin tamamlanma zamanının en küçüklenmesidir. Bu nedenle problem $P_2, S_1 | ST_{sd}, M_j | C_{enb}$ sınıfında yer almaktadır. Ele alınan problem için bir matematiksel model önerilmiştir. Ayrıca problemin NP-zor problemler sınıfında yer alması nedeniyle büyük boyutlu problemlerin çözümü için bir genetik algoritma geliştirilmiştir. Önerilen algoritmanın başarısı rassal türetilen test problemleri kullanılarak matematiksel model ve elde edilen alt sınırlar üzerinden gösterilmiştir.

Two parallel injection machine scheduling under crane constraint

H I G H L I G H T S

- Parallel machine scheduling under crane constraint
- Mathematical model
- Genetic algorithm

Article Info

Received: 24.03.2015

Accepted: 22.09.2016

DOI:

10.17341/gazimmfd.278445

Keywords:

Parallel machine scheduling with one server, sequence dependent setup times, machine eligibility constraints, genetic algorithms, mathematical model

ABSTRACT

In this study, two parallel injection machine scheduling problem under crane constraint is considered. In setup phase of production with injection machine, the mold of the product is fixed to the machine. Molds are transported by a crane to relative machine. The crane is shared by two machines so setup times should not overlap. In addition, the problem has sequence dependent setup times and machine eligibility constraints. The objective is minimization of the makespan. So, class of the problem is $P_2, S_1 | ST_{sd}, M_j | C_{max}$. A mathematical model was proposed for the problem and due to, the problem is in the class of NP-hard problems, a genetic algorithm is developed for solving large size problems. The success of the algorithm is shown by using randomly generated instances through mathematical model and lower bounds.

* Sorumlu Yazar/Corresponding author: tsarac@ogu.edu.tr / Tel: +90 222 239 3750

1. GİRİŞ (INTRODUCTION)

Bu çalışmada, $P_2, S_1 | ST_{sd}, M_j | C_{enb}$ problemi ele alınmıştır. P_2 , 2 adet paralel makina olduğunu, S_1 , makinaların hazırlık aşamasında bir adet kaynak (kreyn) kullandığını, ST_{sd} , sıra bağımlı hazırlık sürelerinin olduğunu ve M_j , makine uygunluk kısıtlarının dikkate alındığını göstermektedir. Probleme, makinalara bağlanacak kalıpların kreyn ile taşınma süreci, hazırlık zamanı olarak düşünülmektedir. Bir adet kreyn olduğu için problem, tek ekipli paralel makine çizelgeleme problemi olarak isimlendirilmektedir ve işlerin hazırlık zamanlarının çakışmaması gerekmektedir. Hazırlık sürelerinin sıra bağımlı olması, bir işin hazırlık zamanının kendisinden önceki işe bağlı olarak farklılık göstermesidir. Amaç fonksiyonu ise işlerin tamamlanma zamanının en küçüklenmesidir. Literatür incelendiğinde tek ekipli paralel makine çizelgeleme problemlerini ele alan çalışmaların çok azında sıra bağımlı hazırlık sürelerinin dikkate alındığı gözlenmiştir. A. Allahverdi vd. [1] hazırlık sürelerinin dikkate alındığı çizelgeleme problemlerini incelemiştir. Çalışmada paralel makine çizelgeleme problemlerinin pek çoğunda sıra bağımlı hazırlık sürelerinin dikkate alınmadığı belirtilmiştir. A. Allahverdi vd. [2] bir başka çalışmalarında sıra bağımlı hazırlık sürelerini dikkate alan ve hazırlık için tek ekip kullanımı durumunda problemin zorluğunun arttığını belirtmişlerdir. Türker ve Sel [3], $P_2, S_1 | ST_{sd} | C_{enb}$ problemini incelemiştir. Problemin çözümü için genetik algoritma ve yasaklı arama yaklaşımlarını birlikte kullanan bir sezgisel önermişlerdir. Önerilen sezgisel yöntemin performansı rassal arama yönteminin sonuçlarıyla karşılaştırarak değerlendirilmiştir. Çalışmada makine uygunluk kısıtları dikkate alınmamıştır ve probleme özgü bir matematiksel model önerilmemiştir. Huang vd. [4], $(PD, S | ST_{sd} | C_{enb})$ çalışmalarında işlerin hangi makinaya atanacağını önceden bilindiği, tek ekipli ve sıra bağımlı hazırlık süresine sahip problem için matematiksel model ve melez bir genetik algoritma önermişlerdir. Kravchenko ve Werner [5], tek ekipli paralel makine çizelgeleme problemi için, bir algoritma önermişlerdir. Ele alınan problemde amaç fonksiyonu tamamlanma zamanının en küçüklenmesidir ve iki paralel makine dikkate alınmıştır. İşlerin hazırlık süreleri eşit olup, bir kabul edilmiştir. Ayrıca iki paralel makine için amaç fonksiyonunun boş zamanların en küçüklenmesi olduğu durumda problemin NP-zor bir problem olduğu gösterilmiştir. Gan vd. [6], $(P_2, S_1 | p_i, s_i | C_{enb})$ tek ekipli paralel makine çizelgeleme probleminde, son işin tamamlanma zamanını en küçükleyecek şekilde 2 farklı karışık tam sayılı model önermişlerdir. Çalışmada dal-sınır algoritması ve sütun türetme yöntemleri bütünlük olarak kullanılmıştır. Kim ve Lee [7], $(P, S_1 | s_j | C_{enb})$ tek ekipli paralel makine çizelgeleme probleminde son işin tamamlanma zamanını en küçükleyecek şekilde tavlama benzetimi ve yasaklı arama algoritmasını kullanan melez bir sezgisel yöntem önermişlerdir. Ayrıca iki farklı matematiksel model geliştirmişlerdir. Bu modellerin farkı, birinde çalışanın bekleme süresinin de dikkate alınması ve bu sürenin en küçüklenmesidir. Bekleme zamanının en küçüklenmesiyle işlerin tamamlama zamanının da en küçüklendiği

gösterilmiştir. Zhang ve Wirth [8], $(P_2, S_1 | s_i | C_{enb})$, çizelgeleme için gerekli bilgilerin kısmi olarak bilindiği durumda tek ekipli paralel makine çizelgeleme problemine liste çizelgeleme yöntemi önermişlerdir. İşlerin aynı uzunluğa ve aynı hazırlık süresine sahip olduğu problemler incelenmiştir. Ayrıca iki farklı sezgisel yöntem önerilmiştir. Hasani vd. [9], $(P_2, S_1 | C_{enb})$ tek ekipli paralel makine çizelgeleme problemine matematiksel bir model önermişlerdir. Abdekhodae ve Wirth [10], tek ekipli paralel makine çizelgeleme problemi için en son işin tamamlanma zamanını en küçükleyecek şekilde bir model önermişlerdir. Eşit işlem süreleri ve kısa işlem sürelerinin olduğu test problemleri önerilen model ile çözülmüştür. Ayrıca iki farklı sezgisel yöntem önerilmiş ve performansları test edilmiştir. Jiang vd. [11], işlerin bölünebilir olduğu, tek ekipli paralel makine çizelgeleme problemi için bir çözüm yöntemi önermişlerdir. Önerilen yöntem, eşit işlem sürelerine ve eşit hazırlık sürelerine sahip işler için en iyi çizelgeleri üretmiştir. Hall vd. [12], tek ekipli özdeş paralel makine çizelgeleme probleminde, farklı amaç fonksiyonları için problemin hesaplama karmaşıklığını incelemiştir. Edis vd. [13], makine ve operatörlerin birlikte çizelgelenmesi problemine tamsayı ve kısıt programlama modeli geliştirmişlerdir. Kravchenko ve Werner [5], $P_2, S_1 | s_i = s | C_{enb}$ probleminin NP- zor bir problem olduğunu belirtmişlerdir. Amaç fonksiyonunun son işin tamamlanma zamanının en küçüklenmesi olduğu tek ekipli iki paralel makinanın çizelgelenmesi problemi hazırlık zamanlarının birbirine eşit olduğu durumda bile NP- zordur. Bu çalışmada $P_2, S_1 | s_i = s | C_{enb}$ problemi birbirine eşit olmayan sıra bağımlı hazırlık süreleri ve bazı işlerin her makinada işlenememesi durumunun dikkate alınmasıyla genelleştirilmiştir. Bu yüzden NP- zor bir problemdir. Bu çalışmada literatürden farklı olarak sıra bağımlı hazırlık süreli, tek ekipli paralel makine çizelgeleme probleminde, uygulamada çok sık karşılaşılan, makinaların bazı işleri işlemek için uygun olmama durumu dikkate alınmıştır. Çalışmada ele alınan problem için bir matematiksel model ve GA (Genetik Algoritma) önerilmiştir. GA'nın başarısı matematiksel model ve alt sınır değerleri üzerinden gösterilmiştir. Çalışmanın izleyen bölümünde ele alınan problem tanımlanmıştır. Üçüncü bölümde geliştirilen matematiksel model verilmiştir. Dördüncü bölümde tasarlanan GA ayrıntılarıyla açıklanmıştır. Sayısal sonuçlar beşinci bölümde, sonuç ve öneriler ise son bölümde verilmiştir.

2. PROBLEMİN TANIMLANMASI VE MOTİVASYON

(PROBLEM DEFINITION AND MOTIVATION)

Bu çalışmada, plastik parça üreten bir fabrikanın bünyesinde yer alan iki plastik enjeksiyon makinasının çizelgelenmesi problemi ele alınmıştır. Söz konusu firma, plastik enjeksiyon, boyama, serigrafı ve tampon baskı konularında faaliyet göstermektedir ve yan sanayi konumundadır. Plastik enjeksiyon; granül halinde gelen plastik ham maddenin sıcaklık yardımıyla eritilip bir kalıp içine enjekte edilerek şekillendirilmesi ve soğutularak

kalıptan çıkarılmasını içeren bir üretim yöntemidir. Bu yöntemle en küçük bileşenlerden bahçe mobilyalarına kadar çok çeşitli ebat ve kategorilerde plastik parçalar üretilebilmektedir. En yaygın üretim yöntemlerinden birisi olan plastik enjeksiyon işlemin gerçekleştirildiği makine, plastik enjeksiyon makinesi olarak adlandırılmaktadır. Bir plastik enjeksiyon parçasının üretim süreci aşağıdaki aşamalardan oluşmaktadır:

Kapama: Bir enjeksiyon makinesi üç ana parçadan oluşur. Bunlar kapama ünitesi (mengene), enjeksiyon ünitesi ve kalıptır. Enjeksiyon ünitesinde ham madde haznesi yer alır. Bu ünite ham maddenin gerekli sıcaklığa kadar ısıtılarak eritilmesi ve uygun basınç ve hızda kalıba enjekte edilmesi görevini yerine getirir. Kapama ünitesi, enjeksiyon ve soğutma esnasında kalıbı basınç altında tutan ünite dir.

Ütipleme: Parçanın istenen ölçülerde olması ve görünümünde çöküntü gibi hataların oluşmaması için uygulanan bir işlemdir. Bu aşamada kalıp içine enjekte edilmiş olan plastik eriyiğin, basınç uygulanarak kalıp boşluğunu iyice doldurması sağlanır. Plastik, kalıp içinde katı hâle gelinceye kadar işlem devam eder. Kullanılan ham maddenin özelliğine, parça ebatlarına ve ağırlığına bağlı olarak işlemin süresi değişkenlik gösterir.

Soğutma: Enjekte edilen eriyik ham maddenin, kalıbın içinde katılaşmasını sağlar.

Kalıp Açma: Kapama ünitesi kalıbın iki tarafını ayrılacak şekilde açılır.

Çıkarma: Kalıp tipine göre maçalar veya iticiler aracılığıyla bitmiş parça kalıptan çıkarılır.

Mengene bölümünün üzerinde hareket ettiği silindireler arasındaki mesafeye kolon aralığı denir. Bir kalıbın enjeksiyon makinesine bağlanabilmesi için kalıp en ve boyunun makinenin kolon aralığından küçük olması gereklidir. Yine bir kalıbın enjeksiyon makinesine bağlanabilmesi için kalıbın derinliğinin kapama aralığıyla uyumlu olması gereklidir. Sonuç olarak; bir kalıp, sadece teknik kısıtları sağlayan makinelere bağlanabilmektedir [14]. İşletmede bulunan enjeksiyon makinelerinin farklı teknik özelliklere sahip olmaları sebebiyle her ürün her makinede üretilememektedir. Bu da makine uygunluk kısıtlarının dikkate alınmasını zorunlu kılar. Bazı makinelerde ürünlerin işleme girme sırasına göre hazırlık süreleri farklılık göstermektedir, diğer bir deyişle enjeksiyon makinelerindeki hazırlık süreleri sıra bağımlıdır. Plastik enjeksiyon makinelerinde kullanılan kalıpların makinelere bağlanması ve sökülmesi için kreyn kullanılmaktadır. İşletmede kreyn bir adet olduğu için işlerin hazırlık süreleri çakışmamalıdır. Ele alınan problemin amacı ise son işin tamamlanma zamanını en küçükleme olarak belirlenmiştir. Böylece hazırlık sürelerinin azaltılması da sağlanacaktır. Sonuçta $P_2, S_1 | ST_{sd}, M_j |$ Cenb problemi dikkate alınmıştır.

3. ÖNERİLEN MATEMATİKSEL MODEL (PROPOSED MATHEMATICAL MODEL)

Ele alınan problem için tam sayılı doğrusal bir matematiksel model geliştirilmiştir.

$N = \{1, 2, \dots, n\}$ iş kümesi

$M = \{1, 2\}$ makine kümesi

İndisler

i, j ve $k \in N$

$l \in M$

Parametreler

p_j : j . işin işlem süresi

h_j : j . iş birinci sırada üretilecekse j . işe ait kalıbın makinaya bağlanması için gereken hazırlık süresi

s_j : i . işe ait kalıbın makinadan sökülmesi ve j . işe ait kalıbın bağlanması için gereken hazırlık süresi

M : çok büyük pozitif sayı

$b_{jl} = \begin{cases} 1; & j. \text{ iş } l. \text{ makinede işlenebiliyorsa} \\ 0; & \text{d. d} \end{cases}$

Karar Değişkenleri

C_j : j işinin tamamlanma zamanı

a_j : j işinin başlama zamanı

w_j : j işinin kreyn bekleme süresi

z_j : j . işin kreyni kullanmayı tamamlama zamanı

$x_{jkl} = \begin{cases} 1; & \text{Eğer } j. \text{ iş } k. \text{ sırada } l. \text{ makinede çizelgelendiyse} \\ 0; & \text{d. d} \end{cases}$

e_{lj} : Kreyn kullanımına dair kısıtlarda kullanılan 0-1 karar değişkenleri

C_{enb} : Son işin tamamlanma zaman

$$C_j + M(1 - x_{jkl}) \geq p_j + w_j + h_j \quad \forall j, k, l, k=1 \quad (1)$$

$$C_j - M(1 - x_{jkl}) \leq p_j + w_j + h_j \quad \forall j, k, l, k=1 \quad (2)$$

$$C_j + M(2 - x_{i, k-1, l} - x_{jkl}) \geq s_{ij} + p_j + C_i + w_j \\ \forall i, j, k, l, i \neq j, k > 1 \quad (3)$$

$$C_j - M(2 - x_{i, k-1, l} - x_{jkl}) \leq s_{ij} + p_j + C_i + w_j \\ \forall i, j, k, l, i \neq j, k > 1 \quad (4)$$

$$\sum_j x_{jkl} - \sum_i x_{i, k-1, l} \leq 0 \quad \forall k, l, k > 1 \quad (5)$$

$$\sum_j x_{jkl} \leq 1 \quad \forall k, l \quad (6)$$

$$\sum_k \sum_l x_{jkl} = 1 \quad \forall j \quad (7)$$

$$a_j \geq C_i + w_j - M * (2 - x_{jkl} - x_{ik-1l})$$

$$\forall i, j, k, l, \quad i \neq j, k > 1 \quad (8)$$

$$a_j \leq C_i + w_j + M * (2 - x_{jkl} - x_{ik-1l})$$

$$\forall i, j, k, l, \quad i \neq j, k > 1 \quad (9)$$

$$a_j \leq w_j + M * (1 - x_{jkl}) \quad \forall j, k, l, \quad k = 1 \quad (10)$$

$$a_j \geq w_j - M * (1 - x_{jkl}) \quad \forall j, k, l, \quad k = 1 \quad (11)$$

$$z_j \geq a_j + n_j - M(1 - x_{jkl}) \quad \forall j, k, l, \quad k = 1 \quad (12)$$

$$z_j \leq a_j + n_j + M(1 - x_{jkl}) \quad \forall j, k, l, \quad k = 1 \quad (13)$$

$$z_j \geq a_j + g_{ij} - M(2 - x_{ik-1l} - x_{jkl}) \quad \forall i, j, k, l, \quad i \neq j, k > 1 \quad (14)$$

$$z_j \leq a_j + n_j + M(2 - x_{ik-1l} - x_{jkl}) \quad \forall i, j, k, l, \quad i \neq j, k > 1 \quad (15)$$

$$z_q \leq a_j + M e_{1jq} \quad \forall j, q \quad j < q \quad (16)$$

$$z_j \leq a_q + M(1 - e_{1jq}) \quad \forall j, q \quad j < q \quad (17)$$

$$x_{jkl} \leq b_{jl} \quad \forall j, k, l \quad (18)$$

$$C_{max} \geq C_j \quad \forall j \quad (19)$$

$$x_{jkl} \in \{0, 1\} \quad \forall j, k, l \quad (20)$$

$$C_j \geq 0 \quad \forall j \quad (21)$$

$$z_j \geq 0 \quad \forall j \quad (22)$$

$$a_j \geq 0 \quad \forall j \quad (23)$$

$$C_{max} \geq 0 \quad (24)$$

$$w_j \geq 0 \quad \forall j \quad (25)$$

kısıtları altında,

$$enk \ z = C_{enb} \quad (26)$$

(1) ve (2) numaralı kısıt grupları birinci sıraya atanan işlerin tamamlanma zamanını hesaplamaktadır. (3) ve (4) numaralı kısıtlar birinci sıra dışında bir sıraya atanan işlerin tamamlanma zamanını hesaplamaktadır. (5) numaralı kısıt işlerin sıra atlamadan atanmasını sağlamaktadır. (6) numaralı kısıt her makinada belirli bir sıraya en fazla bir işin atanmasını garanti etmektedir. (7) numaralı kısıt her işin mutlaka bir makinanın bir sırasına atanmasını sağlamaktadır. (8)-(11) numaralı kısıtlar her işin başlama zamanını hesaplamaktadır. (12)-(15) numaralı kısıtlar her işin kreyni kullanmayı tamamlama zamanının hesaplanmasını sağlamaktadır. (16)-(17) numaralı kısıtlar kreynin aynı anda birden çok iş tarafından kullanımını engellemektedir. (18) numaralı kısıt işin uygun olmayan

makinaya atanmasını engellemektedir. (19) numaralı kısıt son işin tamamlanma zamanını hesaplamaktadır. (20)-(25) numaralı kısıtlar işaret kısıtlarıdır. Amaç (26) ise son işin tamamlanma zamanının en küçüklenmesidir.

4. ÖNERİLEN GENETİK ALGORİTMA (PROPOSED GENETIC ALGORITHM)

GA'nın genel yapısı aşağıdaki gibidir [3]:

Adım 1: Başlangıç popülasyonu rassal olarak oluştur.

Adım 2: Popülasyondaki her bireyin uygunluk değerini hesapla.

Adım 3: Belirlenen nesil sayısı boyunca bireylerden iyi uygunluk değerine sahip olanların büyük seçim şansına sahip olduğu yeniden üretim mekanizmasıyla yeni popülasyonu oluştur.

Adım 4: Çaprazlama operatörünü uygula.

Adım 5: Mutasyon operatörünü uygula.

Adım 6: Durdurma kriteri sağlanmışsa algoritmayı sonlandır ve ilgili popülasyondaki en iyi uygunluk değerine sahip bireyi raporla, aksi halde Adım 2' ye git.

4.1. Kromozom Yapısı (Structure of the Chromosomes)

Kromozomlardaki iş sıralarının belirlenmesinde rassal anahtar yöntemi kullanılmıştır. Yöntem ilk defa Bean (1994) [15], tarafından önerilmiştir. Bu yöntemin kullanımıyla her bir gen için rassal sayı türetilmektedir. Makinaya atanan işlerin sırası türetilen rassal sayıların ondalıklı kısmı dikkate alınarak belirlenir. Rassal anahtar yöntemi özellikle sıralama problemlerinde çok sık kullanılmaktadır. Bu yöntemle ilgili ayrıntılı bilgiye [15] numaralı kaynaktan ulaşılabilir. Önerilen GA' da kromozomların oluşturulmasında [1, 3] olacak şekilde her iş için sürekli rassal sayı türetilir. Türetilen rassal sayının tamsayı kısmı atandığı makinayı, ondalıklı kısmı ise sırasını belirlemektedir. İşlerin sırasını belirlemek için aynı makinaya atanmış işlere ait rassal sayıların ondalıklı kısımları küçükten büyüğe sıralanmaktadır. Aşağıda örnek bir kromozom verilmiştir. 7 iş 2 makine için örnek kromozom:

1	2	3	4	5	6	7
1,3	2,9	2,7	2,1	1,8	2,3	1,2

Örnekte 1. makinaya, 1. 5. ve 7. işler atanmıştır (tam sayılı kısım dikkate alınır).

2. makinaya, 2. 3. 4. ve 6. işler atanmıştır. 1. makinaya atanan işlerin ondalık değerlerini küçükten büyüğe sıraladığımızda işlerin 1. makinaya atanma sırası elde edilir. Bu durumda 1. makinadaki sıra 7, 1, 5 dir. 2. makinada ise sırasıyla 4. 6. 3. ve 2. işler işlem görür. Örnek kromozomun karşı geldiği çizelge aşağıda verilmiştir.

Makine 1	7	1	5	
Makine 2	4	6	3	2

4.2. İlk Neslin Türetilmesi (Generating of First Generation)

Önerilen kromozom yapısı, her işin bir makinaya atanması ve her makinadaki sıraya sadece bir işin atanması kısıtlarının sağlanmasını garanti etmektedir. Ancak makine uygunluk kısıtlarının sağlanması garanti edilememektedir. Bu nedenle ilk nesil türetilirken işlerin atandığı makinalar kontrol edilmektedir. Eğer herhangi bir iş atanmaması gereken bir makinaya atandıysa kromozom, uygunluk kısıtlarının sağlanacağı şekilde düzeltilmektedir.

4.3. Uygunluk Değerinin Hesaplanması (Calculation of the Fitness Values)

Önerilen kromozomda yer alan bilgiler ile işlerin hangi makinalara hangi sırada atanacağı belirlenmektedir. Ancak hazırlık sürelerinin aynı ana denk gelmesi durumunda hangi hazırlığın daha önce yapılacağı bilgisi bulunmamaktadır. Hazırlık sürelerinin çakışmayacağı şekilde son işin tamamlanma zamanının belirlenebilmesi için aşağıdaki uygunluk değerlerinin hesaplanması algoritması geliştirilmiştir:

Adım 1: Her makine için atanan işlerin sıra bağımlı hazırlık zamanları ve işlem sürelerini topla. Daha büyük değere sahip makinadaki ilk işi 0. zamanda başlat. Küçük değere sahip makinadaki ilk işi, diğer makinadaki ilk işin hazırlık zamanı bittikten sonra başlat. (Böylece makinalara atanan ilk işlerin hazırlık zamanlarının çakışması önlenecek şekilde her iki makinadaki ilk işlerin tamamlanma zamanları bulunur.)

Adım 2: L . makine için bir sonraki kreyn gereken zamanı ($C(L)$) (ilgili makinaya en son atanan işin hazırlık zamanı ve işlem zamanının toplamıyla bulunur) ve kreynin boş kalma zamanını ($SEBZ$) (kreyni en son kullanan işin hazırlığının tamamlanma zamanını) belirle.

Adım 3: $C(L)$ değeri M olmayan ve en küçük $C(L)$ değerine sahip makinaya sıradaki işi ata. Bu işin hazırlığının başlama zamanı $C(L)$ ve $SEBZ$ değerlerinden büyük olanına eşittir. $C(L)$ ve $SEBZ$ değerlerini her atama sonrası güncelle. Eğer $C(L) \geq C_{enb}$ ise $C_{enb} = C(L)$ olarak güncelle. Eğer bir makinaya atanacak iş kalmadıysa, bu makinanın $C(L)$ değerini M (çok büyük bir sayı) yap.

Adım 4: Tüm makinaların $C(L)$ değerleri M 'ye eşit ise algoritmayı sonlandır. Aksi halde Adım 3'e git.

4.4. Genetik Algoritma Operatörleri (Genetic Algorithm Operators)

Önerilen GA'da kullanılan operatörler; seçim, çaprazlama, mutasyon ve elitizmdir. *Seçim operatörü* olarak, ikili turnuva yöntemi kullanılmıştır. Bu yöntemde göre popülasyondan rassal olarak iki birey seçilmektedir ve uyum değeri daha iyi olan birey bir sonraki nesle aktarılmaktadır. Bu seçim popülasyondaki birey sayısı kadar tekrar edilmektedir. Bu yaklaşım, uyum değeri yüksek olan bireylerin sonraki nesillerde yer alma

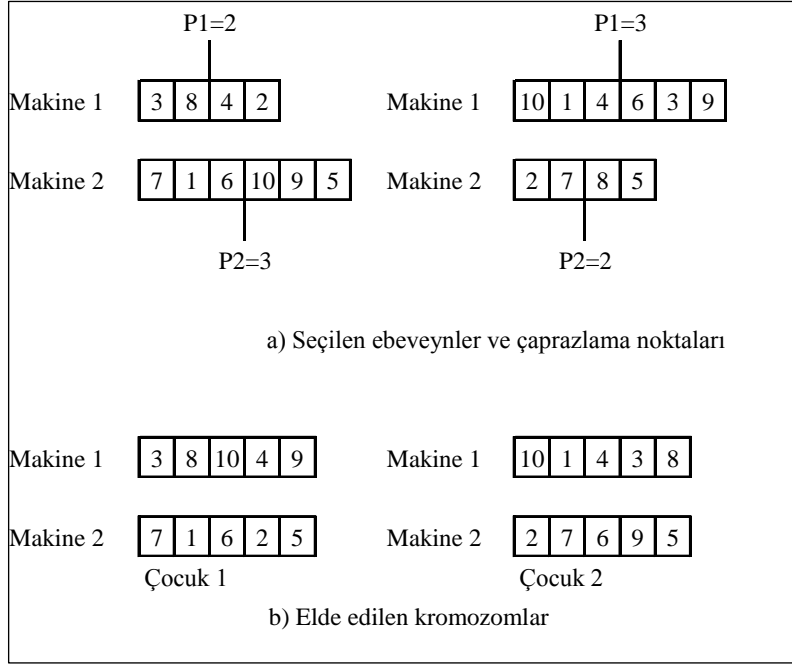
olasılığının da büyük olmasını sağlamaktadır. *Çaprazlama operatörü* yardımıyla bireylerin sahip olduğu farklı özelliklerin bir araya gelmesi sağlanmaktadır. Bu çalışmada kullanılan çaprazlama operatörü çizelge biçimine dönüştürülmüş bireylere uygulanmaktadır ve iki (ebeveyn) bireyden, iki yeni (çocuk) birey oluşmaktadır. Çizelge biçimine dönüştürülmüş bireyler her biri bir makinanın iş sıralarına karşı gelen iki satırdan oluşmaktadır. Çaprazlanacak iki ebeveyn, popülasyondan rassal olarak (çaprazlama oranına göre) seçilmektedir. Daha sonra ebeveynlerin çaprazlama noktaları (p değerleri) yine rassal olarak belirlenmektedir. İlk ebeveyndeki her makina için karşı gelen çaprazlama noktasına kadar olan gen ilk çocuğa aktarılmakta, ilk çocuğun kalan genleri ise diğer ebeveyn bireyden aktarılmaktadır. Benzer kural ikinci ebeveyn ve ikinci çocuk için de geçerlidir. Önerilen çaprazlama operatörü Şekil 1'de bir örnek üzerinde gösterilmiştir. Buna göre ilk ebeveyn için p değerleri 2 ve 3 olarak belirlenmiştir. Böylece birinci makinadan 3 ve 8 ikinci makinadan ise 7, 1 ve 6 işleri ilk çocuğa aktarılmıştır. Ebeveyn 1' den çocuk 1'e aktarılan değerler ebeveyn 2' de silinmiş ve kalan gen değerleri ebeveyn 2' deki atamalar ve sıra dikkate alınarak çocuk 1'e eklenmiştir. Önerilen çaprazlama yaklaşımı makine uygunluk kısıtlarını sağlayan çocuklar oluşturulmasını garanti ettiği için önemli bir avantaja sahiptir.

Mutasyon operatörü, genetik algoritmanın yerel en iyilere takılmasını engellemek ve çözüm uzayında farklı noktalarda arama yapabilmesini sağlamak amacıyla belirli oranlarda uygulanmalıdır. Çalışmada kullanılan mutasyon operatörü işlerin makinalara dengeli bir şekilde dağılmasına yardımcı olmaktadır. Operatör, en fazla işin atandığı makinadaki işlerden birisini rasgele seçilip, makine uygunluk kısıtını da dikkate alarak farklı bir makinada rassal bir sıraya atamaktadır.

Elitizm stratejisinin kullanılmasıyla mevcut neslin en başarılı bireyinin doğrudan bir sonraki nesile aktarılması sağlanarak seçim aşamasında elenmesi ya da operatör işlemleri ile bozulması engellenmektedir. Önerilen GA belirlenen nesil sayısı sağlanana kadar çalışmaktadır.

4.5. Parametrelerin Belirlenmesi (Determining Parameters)

Önerilen GA'nın nesil sayısı, popülasyon büyüklüğü, çaprazlama oranı ve mutasyon oranı parametreleri belirlenirken çeşitli değerler için GA çalıştırılmıştır. Bu değerler Tablo 1'de verilmiştir. Toplam 4 parametre ve her bir parametre için 3 farklı değer olduğundan 81 (3^4) alternatif parametre değeri söz konusu olup, her bir alternatif için problem no 1-1 (2 makine, 10 iş ve sıra bağımlı hazırlık süresinin $U [5, 25]$ olduğu problem) üç kez çalıştırılmış ve elde edilen amaç fonksiyonu ve geçen sürenin 3 koşturum sonucu ortalama değerleri dikkate alınarak parametre değerleri belirlenmiştir. Toplamda genetik algoritma belirtilen problem için 243 defa çalıştırılmıştır. Tablo 1'de * ile belirtilen parametre değerleri seçilmiştir.



Şekil 1. Çaprazlama operatörü (Crossover operator)

Tablo 1. Parametre değerleri (Parameters values)

Parametreler	Değerler
Nesil sayısı	50*, 100, 200
Populasyon büyüklüğü	50, 100, 200*
Çaprazlama oranı	0,7; 0,8; 0,9*
Mutasyon oranı	0,1; 0,2; 0,3*

Özellikle nesil sayısının artırılması çözüm süresini çok uzattığı için nesil sayısı 50 olarak belirlenmiştir. Denemeler sonucu populasyon büyüklüğü, çaprazlama oranı ve mutasyon oranının tabloda belirtilen değerler seçilmesiyle amaç fonksiyonunun iyileştiği belirlenmiştir.

5. SONUÇLAR VE TARTIŞMALAR (RESULTS AND DISCUSSIONS)

Önerilen GA'nın başarısını gösterebilmek için farklı özelliklere sahip test problemleri rassal olarak türetilmiştir. Dört farklı iş sayısı (10, 20, 30 ve 50) ve sıra bağımlı hazırlık sürelerinin yer alacağı iki farklı aralık ([5, 25] ve [25, 50]) olacak şekilde sekiz farklı problem tipi tasarlanmıştır ve her tip problemde beşer tane olmak üzere toplam 40 adet test problemi türetilmiştir. Tablo 2'de tüm parametrelerin nasıl türetildikleri verilmiştir.

Tablo 2. Parametreler ve türetilme yöntemleri (Parameters and their generation methods)

Parametreler	Dağılım/Değer
n	10 / 20 / 30 / 50
m	2
b_{ij}	%70 olasılıkla makine uygun
s_{ij}/h_j	U [5, 25] ve U [25, 50]
p_j	U [10, 100]

İş sayısı, işlem ve hazırlık süreleri türetilirken Huang vd. [4] tarafından önerilen yöntem dikkate alınmıştır. Tüm test problemlerinde işlem zamanları 10 ile 100 arasında düzgün dağılıma uygun olarak, makine uygunluk parametreleri ise %70 olasılıkla "1" (uygun) olacak şekilde rassal türetilmiştir. Tüm testler "Intel (R) Core (TM) i5- 2410 M CPU@2.30 GHz" işlemcisi, 8 GB belleği ve Windows 7 işletim sistemine sahip bir bilgisayarda yapılmıştır.

Matematiksel model GAMS 24. 1. 3 programında kodlanmış ve Cplex çözücüsüyle çözülmüştür. GA ise MATLAB R2012b programında kodlanmıştır. Test problemleri GAMS Cplex çözücüsü ile çözüm süresi 10000 saniyeyle sınırlandırılarak çözülmüş ve elde edilen çözümler Tablo 3'de verilmiştir.

Her problem için GA ile 3 tekrar yapılmıştır. Matematiksel modele konulan süre kısıtı mecburidir. Çünkü problem NP-zor bir problemdir ve çözüm süresi matematiksel modellerle çok uzun zaman gerektirmektedir. Literatürde NP-zor problemlere önerilen matematiksel modellere her zaman süre kısıtı konulmaktadır. Bu süre genel olarak makalelerde 3600 sn olmaktadır [16, 17]. Modelin eniyiye daha yakın sonuç vermesi istendiğinden süre kısıtı 10000 sn olarak belirlenmiştir.

5.1. Probleme Alt Sınır Bulunması (Finding Lower Bound for Problem)

10000 sn içinde matematiksel modelin bulabildiği eniyi çözüm dikkate alınarak sezgisel yöntemin başarısı gösterilmiştir. Fakat en iyi çözüm elde edilemediği için iki farklı alt sınır (AS) bulma yaklaşımı önerilmiştir.

Tablo 3. Hesaplama süreleri ve kıyaslama sonuçları (Computation times and comparison results)

No	GAMS				GA				Hata	Hata	Hata		
	AS1	AS2	ZGAMS	t	Z1	Z2	Z3	ZGA	tga	GAMS	AS1	AS2	
1-1	364	359	376*	2172,13	376	376	376	376	910,32	0,00	3,30	4,74	
1-2	357	353	368	10000,00	376	376	378	376	843,17	2,17	5,32	6,52	
1-3	262	258	272	10000,00	298	300	277	277	829,97	1,84	5,73	7,36	
1-4	284	282	297	10000,00	301	319	303	301	810,83	1,35	5,99	6,74	
1-5	289	284	294	10000,00	300	325	300	300	850,48	2,04	3,81	5,63	
2-1	397	395	425	10000,00	436	425	431	425	824,78	0,00	7,05	7,59	
2-2	420	417	453	10000,00	462	453	463	453	742,80	0,00	7,86	8,63	
2-3	496	493	518	10000,00	533	527	518	518	761,81	0,00	4,44	5,07	
2-4	541	534	565*	2198,74	611	565	565	565	873,08	0,00	4,44	5,81	
2-5	423	419	446	10000,00	469	446	451	446	749,08	0,00	5,44	6,44	
3-1	-	605	669	10000,00	651	658	655	651	2113,69	-2,69	-	7,60	
3-2	-	632	695	10000,00	688	682	667	667	2198,96	-4,03	-	5,54	
3-3	-	606	667	10000,00	650	650	652	650	2249,91	-2,55	-	7,26	
3-4	-	640	695	10000,00	705	685	688	685	1915,15	-1,44	-	7,03	
3-5	-	665	727	10000,00	729	727	732	727	1640,92	0,00	-	9,32	
4-1	-	840	984	10000,00	957	958	966	957	2133,95	-2,74	-	13,93	
4-2	-	773	884	10000,00	856	910	884	856	2182,61	-3,17	-	10,74	
4-3	-	836	971	10000,00	991	934	947	934	1958,09	-3,81	-	11,72	
4-4	-	720	891	10000,00	848	836	836	836	1911,09	-6,17	-	16,11	
4-5	-	910	1090	10000,00	1001	1046	1030	1001	2079,51	-8,17	-	10,00	
5-1	-	965	1177	10000,00	1056	1056	1066	1056	3132,78	-10,28	-	9,43	
5-2	-	947	1075	10000,00	1031	1057	1061	1031	3179,99	-4,09	-	8,87	
5-3	-	860	1280	10000,00	973	973	976	973	2997,64	-23,98	-	13,14	
5-4	-	882	992	10000,00	968	959	959	959	2639,68	-3,33	-	8,73	
5-5	-	1111	1303	10000,00	1177	1244	1177	1177	2843,29	-9,67	-	5,94	
6-1	-	1234	1672	10000,00	1428	1381	1465	1381	2703,55	-17,40	-	11,91	
6-2	-	1241	1560	10000,00	1446	1454	1396	1396	2821,37	-10,51	-	12,49	
6-3	-	1288	1589	10000,00	1481	1496	1474	1474	3057,12	-7,24	-	14,44	
6-4	-	1187	1840	10000,00	1377	1419	1369	1369	3606,56	-25,60	-	15,33	
6-5	-	1265	1582	10000,00	1595	1481	1470	1470	3238,29	-7,08	-	16,21	
7-1	-	1549	-	10000,00	1705	1725	1718	1705	4926,21	-	-	10,07	
7-2	-	1720	2466	10000,00	1976	1972	1903	1903	4662,78	-22,83	-	10,64	
7-3	-	1466	2733	10000,00	1664	1664	1687	1664	4927,90	-39,11	-	13,51	
7-4	-	1399	2760	10000,00	1585	1649	1617	1585	4487,91	-42,57	-	13,30	
7-5	-	1494	2295	10000,00	1691	1669	1658	1658	4604,08	-27,76	-	10,98	
8-1	-	2041	3542	10000,00	2429	2396	2388	2388	5065,54	-32,58	-	17,00	
8-2	-	2091	2976	10000,00	2434	2431	2475	2431	5238,67	-18,31	-	16,26	
8-3	-	2177	-	10000,00	2508	2508	2449	2449	5310,71	-	-	12,49	
8-4	-	2205	-	10000,00	2571	2563	2540	2540	5222,06	-	-	15,19	
8-5	-	2044	3338	10000,00	2408	2408	2364	2364	5202,47	-29,18	-	15,66	
* En iyi çözüme ulaşıldı										ortalama hata	-9,70	5,34	10,38

Alt Sınır Bulma Yöntemi 1 (n=10 iş olan problemler için)

Alt sınırın mümkün olduğunca en iyi çözüme yakın olması istenilir. Bu yüzden öncelikli olarak $P|ST_{sd}|C_{enb}$ probleminin çözümüyle işlerin sıra bağımlı hazırlık süreleri dikkate alınarak C_{enb} en küçüklenecek şekilde tek makinada işlerin sıralaması yapılmıştır. Elde edilen sonuç makine sayısı olan 2' ye bölünmüştür. Fakat $P|ST_{sd}|C_{enb}$ problemi de NP- zor bir problemdir [18]. İlgili problem farklı boyutlardaki test problemleri için çalıştırılmıştır. Makul sürede (10000 sn) sadece 10 işin olduğu problemler için en iyi çözüm elde edilebilmiştir. Bu yüzden 10 işin olduğu problemler için alt sınır elde edilmesinde öncelikli olarak $P|ST_{sd}|C_{enb}$ problemi çözdürülmüştür. Elde edilen sonucun

yarısı alınarak bir alt sınır bulunmuştur (orijinal problemde paralel 2 makine olmasından dolayı). Fakat işlerin hazırlık süresi çakışabilmekte ve işler bölünebilmektedir. Bu durumlar AS değerinin optimalden uzaklaşmasına neden olmaktadır.

Alt Sınır Bulma Yöntemi 2 (Tüm test problemleri için)

Alt sınır değerleri aşağıdaki şekilde hesaplanmıştır.

$$\text{enküçük}_i\{s_{ij}\} = z_j \quad \forall j$$

$$AS = (\sum_{j=1}^m p_j + z_j) / m$$

Makine sayısı (m) 2' dir.

İşlerin hazırlık süreleri çıkarılabilmekte, işler bölünebilmekte ve işlerin hazırlık süreleri belirlenirken en küçük hazırlık süresi üzerinden hesaplama yapılmaktadır. Bu durumlar AS değerinin en iyi sonuçtan uzaklaşmasına neden olmaktadır. Elde edilen alt sınırlar, matematiksel modelin 10000 sn içinde bulunduğu en iyi değer ve GA'nın sonuçları Tablo 3'de verilmiştir. Tablo 3 sekiz bölümden oluşmaktadır. İlk sütunda problem numarası verilmiştir. Problem numarası 1-1 ve 1-5 aralığındaki problemlerde iş sayısı 10 olup işlerin hazırlık süreleri (s_{ij}/h_j) U [5, 25] dağılımında, 2-1 ve 2-5 aralığındaki problemlerde iş sayısı 10 olup işlerin hazırlık süreleri U [25, 50], 3-1 ve 3-5 aralığındaki problemlerde iş sayısı 20 işlerin hazırlık süreleri U [5, 25], problem numarası 4-1 ve 4-5 aralığındaki problemlerde iş sayısı 20 işlerin hazırlık süreleri U [25, 50], 5-1 ve 5-6 aralığındaki problemlerde iş sayısı 30 işlerin hazırlık süreleri U [5, 25], 6-1 ve 6-5 aralığındaki problemlerde iş sayısı 30 işlerin hazırlık süreleri U [25, 50], 7-1 ve 7-5 aralığındaki problemlerde iş sayısı 50 işlerin hazırlık süreleri U [5, 25], 8-1 ve 8-5 aralığındaki problemlerde ise iş sayısı 50 işlerin hazırlık süreleri U [25, 50] dağılımına uygundur. 2. sütunda alt sınır bulma yöntemi 1 ile elde edilen değerler vardır. 3. sütuna alt sınır bulma yöntemi 2 ile elde edilen çözümler eklenmiştir. 4. Sütunda önerilen matematiksel modelin 10000 sn içinde elde ettiği en iyi çözümler (ZGAMS) ve çözüm süreleri (t) verilmiştir. 5. sütun GA'nın üç kere çalıştırılması sonucu elde edilen amaç fonksiyonu değerleri (Z1, Z2 ve Z3) ve GA'nın üç koşuturum süresinin ortalama değeri (tga) verilmiştir. Ayrıca 3 deneme sonucu GA ile elde edilen en iyi çözüm ZGA olarak verilmiştir. 6. Sütunda (ZGA- ZGAMS)/ ZGAMS*100 formülü kullanılarak GA'nın başarısı GAMS sonuçları üzerinden gösterilmiştir. Genetik algoritma özellikle büyük boyutlu problemlerde süre limiti içinde GAMS'in bulunduğu sonuca göre daha iyi sonuçlar vermiştir. Büyük boyutlu problemlerde GA ile daha küçük bir amaç fonksiyonu değeri elde edildiği için hata yüzdeleri negatif çıkmıştır. Bu negatif sonuçlar GA'nın GAMS sonucuna göre başarısını göstermektedir. Küçük boyutlu problemlerin çözümünde GAMS değeri alt sınır değerlerine daha çok yaklaşmıştır. Önerilen GA iş sayısının 10 olduğu durumda GAMS sonucuna göre en çok %2,17 hata vermiştir. 7. ve 8. sütunlarda GA'nın başarısı sırasıyla AS1 ve AS2 değerleri üzerinden gösterilmiştir. Hata oranı (ZGA- AS)/ AS*100 ile hesaplanmıştır. Alt sınır bulma yöntemi 1 ile elde edilen değerler en iyiye daha yakındır. Fakat yine en iyi değerden küçüktür. Özellikle hazırlık zamanlarının U [25, 50] dağıldığı problemlerde en iyi çözümlerden U [5, 25] dağılımlı problemlere göre daha uzak değerler elde edilmektedir. Bu yüzden hazırlık süresinin U [25, 50] dağıldığı problemlerde hata oranı daha yüksek bir değer çıkmıştır. En iyi çözümden daha küçük değerlerle karşılaştırılma yapılmasına rağmen GA'nın hata oranı AS1'e göre ortalama 5, 34 ve AS2 ye göre ortalama 10,38 bulunmuştur. İki problem için GAMS ile en iyi çözümler elde edilmiştir. Bu problemler 1-1 ve 2-4 problemleridir. Bu problemlerin en iyi sonucu ve AS2 değerleri arasında sırasıyla %4,74 ve %5, 81 hata oranı vardır. Bu değerler önerilen alt sınır bulma yönteminin de başarısını göstermektedir. Problem boyutları arttıkça en iyi

değer ve AS2 arasındaki hata oranının da artması beklenir. Bu da GA ve AS2 karşılaştırması sonucu elde edilen hata oranlarının mantıklı olduğunu ve önerilen GA'nın başarılı olduğunu göstermektedir.

6. SONUÇLAR (CONCLUSIONS)

Bu çalışmada $P_2, S_1 | ST_{sd}, M_j | C_{enb}$ problemi için bir matematiksel model ve bir GA önerilmiştir. Önerilen GA ile özellikle iş sayısının fazla olduğu problemlerde, matematiksel modelin belirli bir süre içinde elde ettiği sonuca göre daha iyi sonuçlar, daha kısa sürede elde edilmiştir. Ayrıca GAMS Cplex çözücüsü ile çözüm bulunamayan problemler GA ile çözülebilmektedir. GA'nın başarısı ayrıca elde edilen alt sınırlar ile gösterilmiştir. Gelecekteki çalışmalarda ilgili probleme GA'nın dışında tavlama benzetimi, tabu arama algoritması gibi farklı sezgisel yöntemler geliştirilebilir ve bu yöntemlerin başarısı karşılaştırılabilir. Ayrıca probleme hazır olma zamanı, teslim zamanı kısıtları eklenebilir ve ikiden fazla makinenin olduğu daha karmaşık yapılar incelenebilir.

KAYNAKLAR (REFERENCES)

1. Allahverdi A., Gupta J.N., Aldowaisan T., A review of scheduling research involving setup considerations, *Omega*, 27 (2), 219- 239, 1999.
2. Allahverdi A., Mg C.T., Cheng T.C., Kovalyov M.Y., A survey of scheduling problems with setup time sor costs, *European Journal of Operational Research*, 187 (3), 985- 1032, 2008.
3. Türker A.K., Sel Ç., A hybrid approach on single server parallel machines scheduling problem with sequence dependent setup times, *Journal of the Faculty of Engineering and Architecture of Gazi University*, 26 (4) 731- 740, 2011.
4. Huang S., Cai L., Zhang X., Parallel dedicated machine scheduling problem with sequence-dependent setups and a single server, *Computers & Operations Research*, 58 (1), 165- 174, 2010.
5. Kravchenko S.A., Werner F., Parallel Machine Scheduling Problems with a Single Server, *Mathematical and Computer Modelling*, 26 (12), 1- 11, 1997.
6. Gan H., Wirth A., Abdekhodae A., A branch-and-price algorithm for the general case of scheduling parallel machines with a single server, *Computers & Operations Research*, 39 (9), 2242- 2247, 2012.
7. Kim M., Lee Y.H., MIP models and hybrid algorithm for minimizing the makespan of parallel machines scheduling problem with a single server, *Computers & Operations Research*, 39 (11), 2457-2468, 2012.
8. Zhang L., Wirth A., On-line scheduling of two parallel machines with a single server, *Computers & Operations Research*, 36 (5), 1529-1553, 2009.
9. Hasani K., Kravchenko S.A., Werner, F., Block models for scheduling jobs on two parallel machines with a single server, *Computers & Operations Research*, 41, 94- 97, 2014.

10. Abdekhodae A.H., Wirth A., Scheduling parallel machines with a single server: some solvable cases and heuristics, *Computers & Operations Research*, 29 (3), 295- 315, 2002.
11. Jiang Y., Dong J., Ji M., Preemptive scheduling on two parallel machines with a single server, *Computers & Industrial Engineering*, 66 (2), 514- 518, 2013.
12. Hall N.G., Potts C.N., Sriskandarajah C., Parallel machine scheduling with a common server, *Discrete Applied Mathematics*, 102 (3), 223- 243, 2000.
13. Edis E.B., Oğuz C., Özkarahan İ., Solution approaches for simultaneous scheduling of jobs and operators on parallel machines, *Journal of the Faculty of Engineering and Architecture of Gazi University*, 27 (3), 527- 535, 2012.
14. Saraç T., Genelleştirilmiş karesel çoklu sırt çantası problemi için melez bir çözüm yaklaşımı, Doktora Tezi, Eskişehir Osmangazi Üniversitesi, Fen Bilimleri Enstitüsü., Eskişehir, 2007.
15. Bean J.C., Genetic algorithms and random keys for sequencing and optimization, *ORSA J. Comput.* 6, 154-160, 1994.
16. Joo C.M., Kim B.S., Hybrid genetic algorithms with dispatching rules for unrelated parallel machine scheduling with setup time and production availability, *Computers & Industrial Engineering*, 85, 102- 109, 2015.
17. Kim M., Lee Y.H., MIP models and hybrid algorithm for minimizing the makespan of parallel machines scheduling problem with a single server, *Computers & Operations Research*, 39 (11), 2457- 2468, 2012.
18. Pinedo M.L., *Scheduling theory, algorithms and systems*, Springer, New York, 3, 111- 142, 2008.

