Gazi Üniv. Müh. Mim. Fak. Der.
Cilt 23, No 2, 459-465, 2008

J. Fac. Eng. Arch. Gazi Univ.
Vol 23, No 2, 459-465 , 2008

ZAMANA-BAĞIMLI ÖĞRENME ETKİLİ ÇİZELGELEME PROBLEMİNDE MAKSİMUM GECİKME MİNİMİZASYONU: DOĞRUSAL OLMAYAN PROGRAMLAMA MODELİ

Tamer EREN
Endüstri Mühendisliği Bölümü, Kırıkkale Üniversitesi, 71451 Kırıkkale
teren@kku.edu.tr

(Geliş/Received: 19.07.2007; Kabul/Accepted: 14.01.2008)

ÖZET

Çizelgeleme literatürünün çoğunda işlerin işlem zamanları sabit kabul edilmiştir. Ancak işlerin işlem zamanlarında, başlama zamanı veya pozisyonuna bağlı olarak azalma görülebilmektedir. Bu olgu literatürde öğrenme etkisi olarak bilinmektedir. Bu çalışmada da zamana-bağımlı öğrenme etkili tek makineli çizelgeleme problemi ele alınacaktır. Ele alınan problemin amaç fonksiyonu maksimum gecikme minimizasyonudur. Çalışmada problemin klasik (öğrenme etkisiz) durumunda en iyi çözümü garanti eden EDD (en erken teslim tarihi) kuralının, zamana-bağımlı öğrenme etkili durumda optimal çözümü garanti etmediği gösterilmiştir. Ayrıca problemi çözmek için doğrusal-olmayan programlama modeli geliştirilmiş ve geliştirilen modelle 14 işe kadar optimal çözümler bulunmuştur.

Anahtar kelimeler: Çizelgeleme, maksimum gecikme, zamana-bağımlı öğrenme etkisi, doğrusal-olmayan programlama modeli.

MINIMIZING THE MAXIMUM LATENESS IN A SCHEDULING PROBLEM WITH A TIME-DEPENDENT LEARNING EFFECT: A NON-LINEAR PROGRAMMING MODEL

ABSTRACT

In traditional scheduling problems, most literature assumes that the processing time of a job is fixed. However, there are many situations where the processing time of a job depends on the starting time or the position of the job in a sequence. In such situations, the actual processing time of a job may be more or less than its normal processing time if it is scheduled later. This phenomenon is known as the ‘‘learning effect’’. In this study, we introduce a time-dependent learning effect into a single-machine scheduling problem. The objective function of the problem is minimization of the maximum lateness. This study shows that EDD rule, which guarantees the best solution in classical situation (without learning effect), can not guarantee the best results in the situation with learning effect. In addition, a non-linear programming model is proposed for this problem, and solutions are found for problems which have up to 14 jobs.

Keywords: Scheduling, maximum lateness, time-dependent learning effect, non-linear programming model.

1. GİRİŞ (INTRODUCTION)

Çizelgeleme literatürünü bakıldığında problemler genellikle işlem zamanları sabit kabul edilme varsayımına dayanmaktadır. Halbuki işin işlem zamanı, işin başlama zamanına veya işin pozisyonuna bağlı olarak azalabilmektedir. Bu olgu literatürde öğrenme etkisi olarak bilinmektedir. Literatürde öğrenme etkisi zamana-bağımlı ve pozisyona bağımlı olmak üzere iki grupta ele alınmıştır.

Birinci grupta işin işlem zamanı işin başlama zamanına bağımlı olarak azalma varsayımına dayanırken, diğerinde ise pozisyonuna göre işlem zamanlarının azaldığı kabul edilmiştir[1]. Bu çalışmada da ilk gruba göre tek makineli çizelgeleme probleminde öğrenme etkisi ele alınmıştır. Problemin amaç fonksiyonu maksimum gecikme minimizasyonudur. Problemin klasik durumunda optimal sonucu veren EDD (en erken teslim tarihi) kuralının, zamana bağımlı durumda optimal çözümü garanti etmediği gösterilecektir. Ayrıca problem için doğrusal-olmayan programlama modeli geliştirilmiş ve geliştirilen modelle 14 işe kadar optimal çözümler bulunmuştur.

Öğrenme etkisi ile ilgili ilk çalışma Biskup [2] tarafından tek makineli çizelgeleme problemleri için yapılmıştır. Biskup [2] çalışmasında toplam akış zamanının SPT (en kısa işlem zamanı) kuralı ile minimize edildiğini göstermiştir. Ayrıca teslim tarihinden minimum sapma probleminin, atama modeli ile $O(n^3)$ zamanda çözüldüğünü göstermiştir. Moshiev [3] yaptığı çalışmada maksimum tamamlanma zamanının yine SPT kuralı ile çözüldüğünü göstermiştir. Araştırmacı çok ölçütlü iki problemi ele almıştır. Bunlardan birincisi tamamlanma zamanı ve tamamlanma zamanınından sapmayı enküçükleme, diğeri ise teslim tarihi atama problemidir. Bu iki problemin atama modeli ile $O(n^3)$ zamanda çözüldüğünü göstermiştir. Ayrıca Moshiev [3] problemi klasik durumda (öğrenme etkisiz) optimal olarak çözen yöntemlerin öğrenme etkili olmadığını, maksimum gecikme için EDD ve minimum geciken iş sayısı problemlerin Moore [4] algoritması ile çözülmesi durumunda optimal çözümü garanti etmediğini göstermiştir. Maksimum gecikme probleminin ise Zhao vd. [5] ve Wu vd. [6] özel durumlarda $O(n \log n)$ zamanda çözüldüğünü göstermişlerdir. Eren ve Güner [7] ise çalışmalarında toplam gecikme problemini ele almışlar ve problem için matematiksel programlama modeli önermişlerdir. Ayrıca büyük boyutlu problemler için tabu arama ve tavlama benzetimi sezgiselleri geliştirmişlerdir. Bu bahsedilen tüm çalışmalar pozisyona bağımlı öğrenme etkisi ile yapılmıştır. Zamana bağımlı öğrenme etkisi ile ilgili ilk çalışma ise Kuo ve Yang [8,9] tarafından yapılmıştır. Araştırmacılar çalışmalarında maksimum tamamlanma zamanı ve toplam tamamlanma zamanı minimizasyonu probleminin SPT kuralıyla optimal olarak çözülebildiğini göstermişlerdir. Ayrıca Kuo ve Yang [10] yaptıkları diğer bir çalışmada, tek makineli grup çizelgeleme probleminde maksimum tamamlanma zamanı ve toplam tamamlanma zamanı problemlerinin yine SPT kuralı ile çözülebileceğini göstermişlerdir.

Çalışmanın ikinci bölümünde ele alınan problem tanımlanacaktır. Üçüncü bölümde problem için önerilen doğrusal-olmayan programlama modeli verilecektir. Deneysel sonuçlar ise dördüncü bölümde sunulacaktır. Son bölümde çalışmanın sonuçları verilecek ve gelecekte yapılabilecek çalışmalar hakkında öneriler sunulacaktır.

2. PROBLEMİN TANIMLANMASI (PROBLEM DESCRIPTION)

Pozisyona bağlı öğrenme etkisinde işlerin işlem zamanları değil de tekrar sayısı dikkate alınmıştır. Eğer öğrenme, işin başlama zamanına bağlı ise zamana-bağımlı öğrenme etkisi olarak ifade edilmektedir. Kuo ve Yang [8] tarafından model şu şekilde tanımlanmıştır: Tek makineli n işli çizelgeleme problemi ele alınmıştır. p_j işinin r. posizyondaki işlem zamanı aşağıdaki şekilde tanımlanmıştır:

$$p_{jr} = \left(1 + p_{[1]} + p_{[2]} ... + p_{[r-1]}\right)^a p_j = \left(1 + \sum_{r=1}^{r-1} p_{[j]}\right)^a p_j$$

Burada p_j, j işinin işlem zamanını göstermektedir. Öğrenme indeksi $a < 0$ dır ve öğrenme oranının iki tabanına göre logaritmasıdır. d_j ve C_j ise j işinin teslim tarihi ve tamamlanma zamanıdır. Maksimum gecikme $L_{\max} = \max_{j=1}^{n} L_j = \max_{j=1}^{n} \left\{0, C_j - d_j\right\}$ şeklinde ifade edilmektedir. İşlerin atandığı pozisyonlara göre işlem zamanları ve gecikmeleri Tablo 1'de gösterilmiştir.

Klasik durumda (öğrenme etkisiz) problemi optimal olarak çözen EDD yönteminin, ele alınan zamana-bağımlı öğrenme etkili çizelgeleme probleminde optimal sonucu garanti etmediği örnek bir problem üzerinde gösterilecektir.

Sayısal örnek:

Tek makineli dört işli çizelgeleme problemi sayısal verileri Tablo 2'de verilmiştir. Ele alınan problem zamana-bağlı öğrenme etkili olduğu durumda EDD kuralı ile çözüldüğünde maksimum gecikme değeri iş sıralaması Tablo 3'de verilmiştir. Tablo 3'de görüldüğü gibi $L_{\max}$ değeri 2.22 saat ve sıralama

Tablo 1. Pozisyonlara göre işlem zamanı ve gecikme

pozisyon	İşlem zamanı	Gecikme
1	p_1	$L_1 = C_1 - d_1 = p_1 - d_1$
2	$\left(1 + p_{[1]}\right)^a p_2$	$L_2 = C_2 - d_2 = C_1 + \left(1 + p_{[1]}\right)^a p_2 - d_2$
3	$\left(1 + p_{[1]} + p_{[2]}\right)^a p_3$	$L_3 = C_3 - d_3 = C_2 + \left(1 + p_{[1]} + p_{[2]}\right)^a p_3 - d_3$
…	…	…
r-1	$\left(1 + p_{[1]} + p_{[2]} ... + p_{[r-2]}\right)^a p_{r-1}$	$L_{r-1} = C_{r-1} - d_{r-1} = C_{r-2} + \left(1 + p_{[1]} + p_{[2]} ... + p_{[r-2]}\right)^a p_{r-1} - d_{r-1}$
r	$\left(1 + p_{[1]} + p_{[2]} ... + p_{[r-1]}\right)^a p_r$	$L_r = C_r - d_r = C_{r-1} + \left(1 + p_{[1]} + p_{[2]} ... + p_{[r-2]} + p_{[r-1]}\right)^a p_r - d_r$

Tablo 2. Sayısal örnek verileri

j	1	2	3	4
p_j	5	8	9	12
d_j	15	12	14	11

4-3-2-1 olarak bulunmuştur. Ele alınan problemin tüm çözüm alternatifleri ise Tablo 4'de verilmiştir. Tablo 4'de de görüldüğü gibi 4 işli bir problem permütasyon tipi için 4!=24 alternatif bulunmakta ve 1-4-2-3 sıralaması ile gecikme olmadığı yani $L_{\max} = 0$ saat olduğu görülmektedir.

Tablo 3. EDD sonucuna göre çözüm sonucu

j	4	2	3	1
p_j	12	8	9	5
d_j	11	12	14	15
C_j	12.00	14.22	16.18	17.10
L_j	1.00	2.22	2.18	2.10

Problemin optimal sıralaması ve gecikme değerleri de Tablo 5'de verilmiştir.

Bu örnekte de görüldüğü gibi EDD kuralı zamana-bağlı öğrenme etkili çizelgelemede optimal çözümü garanti etmemektedir. Ele alınan problemi özel durumda polinom zamanda çözen iki teorem verilecektir. Ayrıca tüm durumlarda optimal çözümü bulmak için kullanabilecek doğrusal-olmayan programlama modeli takip eden bölümde verilecektir.

Teorem: Tek makineli zamana-bağımlı çizelgeleme probleminde tüm işlerin,

i. teslim tarihi aynı ise maksimum gecikme optimal olarak SPT kuralı ile,

ii. işlem zamanları aynı ise maksimum gecikme optimal olarak EDD kuralı ile bulunur.

İspat: Bu teoremin ispatı için bitişik iş çiftlerinin yerdeğiştirmesi yöntemi kullanılacaktır. S_1 çizelgesi iş sıralaması $(\pi_1, J_h, J_i, J_j, \pi_2)$ Burada π_1 r-2 ve ondan önceki pozisyondaki işlerin sıralamasını J_h

Tablo 5. Optimal çözüm sonucu

j	1	4	2	3
p_j	5	12	8	9
d_j	15	11	12	14
C_j	5.00	9.90	11.78	13.55
L_j	0	0	0	0

J_i ve J_j ise sırasyla r-1, r ve r+1. pozisyondaki işleri, π_2 ise r+2 ve ondan sonraki pozisyondaki işlerin sıralamasını göstermektedir. S_2 çizelgesinin S_1 sçizelgesinden farkı i ve j işleri yer değiştirmiştir ve sıralama $(\pi_1, J_h, J_j, J_i, \pi_2)$ şeklindedir. İşin r. pozisyondaki tamamlanma zamanı ve gecikmesi

$$C_r = C_{r-1} + \left(1 + \sum_{k=1}^{r-1} p_{[k]}\right)^a p_r, \quad \text{ve}$$

$$L_r = C_r - d_r = C_{r-1} + \left(1 + \sum_{k=1}^{r-1} p_{[k]}\right)^a p_r - d_r \quad \text{dir.}$$

Maksimum gecikme ise $L_{\max} = \max_{r=1}^{n} L_r$ dır.

i. İşlerin teslim tarihleri birbirine eşit $d_i = d_j = d$ ve $p_i \le p_j$ dir.

S_1 çizelgesindeki i. ve j. işlerin gecikmeleri denklem (1) ve (2) de S_2 çizelgesindeki j ve i. ve işlerin gecikmeleri ise denklem (3) ve (4) de verilmiştir.

$$L_i(S_1) = C_i(S_1) - d_i$$
$$= C_h(S_1) + p_i\left(1 + \sum_{k=1}^{r-1} p_{[k]}\right)^a - d \qquad (1)$$

$$L_j(S_1) = C_j(S_1) - d_j = C_h(S_1)$$
$$+ p_i\left(1 + \sum_{k=1}^{r-1} p_{[k]}\right)^a + p_j\left(1 + p_i + \sum_{k=1}^{r-1} p_{[k]}\right)^a - d \qquad (2)$$

Tablo 4. Sayısal örneğin tüm alternatif sonuçları

Alternatif	Sıralama	Lmax değeri (sa)	Alternatif	Sıralama	Lmax değeri (sa)
1	1-2-3-4	2.17	13	3-1-2-4	4.15
2	1-2-4-3	0.47	14	3-1-4-2	3.22
3	1-3-2-4	2.24	15	3-2-1-4	4.21
4	1-3-4-2	1.31	16	3-2-4-1	3.36
5	**1-4-2-3**	**0.00**	17	3-4-1-2	3.40
6	1-4-3-2	1.56	18	3-4-2-1	2.50
7	2-1-3-4	3.57	19	4-1-2-3	3.27
8	2-1-4-3	1.87	20	4-1-3-2	5.05
9	2-3-1-4	3.68	21	4-2-1-3	3.07
10	2-3-4-1	2.83	22	4-2-3-1	2.22
11	2-4-1-3	1.00	23	4-3-1-2	5.10
12	2-4-3-1	1.00	24	4-3-2-1	4.20

$$L_j(S_2) = C_j(S_2) - d_j$$
$$= C_h(S_2) + p_j\left(1+\sum_{k=1}^{r-1} p_{[k]}\right)^a - d \quad (3)$$
$$L_i(S_2) = C_i(S_2) - d_i = C_h(S_2)$$
$$+ p_j\left(1+\sum_{k=1}^{r-1} p_{[k]}\right)^a + p_i\left(1+p_j+\sum_{k=1}^{r-1} p_{[k]}\right)^a - d \quad (4)$$

Denklem (1) ve (2)'den $L_i(S_1) > L_j(S_1)$ dir. Çünkü

$$p_j\left(1+p_i+\sum_{k=1}^{r-1} p_{[k]}\right)^a > 0$$

Denklem (3) ve (4)'den ise $L_i(S_2) > L_j(S_2)$ dir.

Çünkü $p_i\left(1+p_j+\sum_{k=1}^{r-1} p_{[k]}\right)^a > 0$

Denklem (1) den Denklem (3) çıkarıldığında $L_i(S_1) - L_j(S_2) = (p_i - p_j)\left(1+\sum_{k=1}^{r-1} p_{[k]}\right)^a < 0$ ifadesi bulunur. Çünkü $p_i \le p_j$ olduğundan dolayı $L_j(S_2) > L_i(S_1)$ dir.

Denklem (1) den Denklem (4) çıkarıldığında ise,

$$L_i(S_1) - L_i(S_2) = (p_i - p_j)\left(1+\sum_{k=1}^{r-1} p_{[k]}\right)^a$$
$$- p_i\left(1+p_j+\sum_{k=1}^{r-1} p_{[k]}\right)^a < 0$$

dır. Çünkü

$p_i \le p_j$ olduğundan dolayı $L_i(S_2) > L_i(S_1)$

$$L_j(S_1) - L_i(S_2) = p_i\left(1+\sum_{k=1}^{r-1} p_{[k]}\right)^a$$
$$+ p_j\left(1+p_i+\sum_{k=1}^{r-1} p_{[k]}\right)^a - p_j\left(1+\sum_{k=1}^{r-1} p_{[k]}\right)^a$$
$$- p_i\left(1+p_j+\sum_{k=1}^{r-1} p_{[k]}\right)^a$$
$$L_j(S_1) - L_i(S_2) = (p_i - p_j)\left(1+\sum_{k=1}^{r-1} p_{[k]}\right)^a$$
$$+ p_j\left(1+p_i+\sum_{k=1}^{r-1} p_{[k]}\right)^a - p_i\left(1+p_j+\sum_{k=1}^{r-1} p_{[k]}\right)^a$$
$$L_j(S_1) - L_i(S_2) = (p_i - p_j)(x)^a + p_j(p_i + x)^a$$
$$- p_i(p_j + x)^a$$
$$\frac{L_j(S_1) - L_i(S_2)}{p_i} = (1-\lambda)(x)^a + \lambda(p_i + x)^a$$
$$-(p_j + x)^a$$

$r = 1$

$$L_j(S_1) - L_i(S_2) = (p_j - p_i) + p_i(1+p_j)^a$$
$$- p_j(1+p_i)^a$$

$r \ge 2$

$$L_j(S_1) - L_i(S_2) = (p_j - p_i)\left(1+\sum_{k=1}^{r-1} p_{[k]}\right)^a$$
$$+ p_i\left(1+p_j+\sum_{k=1}^{r-1} p_{[k]}\right)^a - p_j\left(1+p_i+\sum_{k=1}^{r-1} p_{[k]}\right)^a$$

Durum 1. $r = 1$

$\lambda = p_j / p_i$ eşitliği tekrar yazarsak,

$$L_i - L_j = \lambda p_i\left(1-(1+p_i)^a\right) - p_i\left(1-(1+\lambda p_i)^a\right)$$

$t = p_i > 0$ yazılır ve $\lambda = p_j / p_i \ge 1$ olduğundan dolayı

$L_i - L_j = \lambda t\left(1-(1+t)^a\right) - p_i\left(1-(1+\lambda t)^a\right) \ge 0$ (bknz. [8]).

Sonuç olarak $p_i \le p_j$ sırasıyla elde edilir.

Durum 2. $r \ge 2$

$$x = 1+\sum_{k=1}^{r-1} p_{[k]}$$
$$L_j(S_1) - L_i(S_2) = (p_j - p_i)x^a + p_i(p_j + x)^a$$
$$- p_j(p_i + x)^a$$
$$\frac{L_j(S_1) - L_i(S_2)}{x^a} = (p_j - p_i) + p_i\left(1+\frac{p_j}{x}\right)^a$$
$$- p_j\left(1+\frac{p_i}{x}\right)^a$$

$x = 1+\sum_{k=1}^{r-1} p_{[k]} > 0$ olduğu bilenmektedir. $\lambda = p_j / p_i$ yazılırsa

$$\frac{L_j(S_1) - L_i(S_2)}{x^a} = \lambda p_i\left[1-\left(1+\frac{p_i}{x}\right)^a\right]$$
$$- p_i\left[1-\left(1+\frac{\lambda p_i}{x}\right)^a\right]$$

$t = p_i / x > 0$ yazılır ve $\lambda = p_j / p_i \ge 1$ olduğundan dolayı

$$\frac{L_j(S_1) - L_i(S_2)}{x^a}$$
$$= p_i\left[\lambda\left(1-(1+t)^a\right) - \left(1-(1+\lambda t)^a\right)\right] \ge 0$$

(bknz. [8]).

Sonuç olarak $L_j(S_1) \ge L_i(S_2)$ dir.

$$L_{\max}(S_1) < L_{\max}(S_2)$$
$$\max\{L_i(S_1), L_j(S_1)\} < \max\{L_i(S_2), L_j(S_2)\}$$

$L_i(S_1) < L_i(S_2)$ SPT kuralı ile optimal sıralama bulunur.

ii. İşlerin işlem zamanları birbirine eşit $p_i = p_j = p$ ve $d_i \le d_j$ dir.

S_1 çizelgesindeki *i*. ve *j*. işlerin gecikmeleri denklem (5) ve (6) de S_2 çizelgesindeki *j* ve *i*. ve işlerin gecikmeleri ise denklem (7) ve (8) de verilmiştir.

$$L_i(S_1)=C_i(S_1)-d_i = C_h(S_1)+p(1+p(r-1))^a-d_i \quad (5)$$

$$L_j(S_1)=C_j(S_1)-d_j = C_h(S_1) + p(1+p(r-1))^a+p(1+pr)^a-d_j \quad (6)$$

$$L_j(S_2)=C_j(S_2)-d_j = C_h(S_2)+p(1+p(r-1))^a-d_j \quad (7)$$

$$L_i(S_2)=C_i(S_2)-d_i = C_h(S_2) + p(1+p(r-1))^a+p(1+pr)^a-d_i \quad (8)$$

Denklem (8)'dan Denklem (7) çıkarılırsa,

$L_i(S_2)-L_j(S_2)=p(1+pr)^a+d_j-d_i>0$

$p(1+pr)^a>0$ ve $d_i \le d_j$ olduğundan dolayı $L_i(S_2)>L_j(S_2)$ dir.

Denklem (7)'dan Denklem (5) çıkarılırsa,

$L_j(S_2)-L_i(S_1)=d_i-d_j<0$ olduğundan $L_i(S_1)>L_j(S_2)$ dir

Denklem (8)'dan Denklem (5) çıkarılırsa,

$L_i(S_2)-L_i(S_1)=p(1+pr)^a>0$ olduğundan dolayı $L_i(S_2)>L_i(S_1)$ dir.

Denklem (7)'den Denklem (6) çıkarılırsa,

$L_j(S_2)-L_j(S_1)=-p(1+pr))^a<0$ olduğundan dolayı $L_j(S_1)>L_j(S_2)$

Denklem (8)'den Denklem (6) çıkarılırsa,

$L_i(S_2)-L_j(S_1)=d_j-d_i>0$ olduğundan dolayı $L_i(S_2)>L_j(S_1)$ dir.

$L_{\max}(S_1)<L_{\max}(S_2)$

$\max\{L_i(S_1),L_j(S_1)\}<\max\{L_i(S_2),L_j(S_2)\}$

$L_j(S_1)<L_i(S_2)$ EDD kuralı ile optimal sıralama bulunur.

3. MATEMATİKSEL MODEL (MATHEMATICAL MODEL)

3.1. Parametreler (Parameters):

j:	İş sayısı	$j=1,2,...,n$
p_j:	*j* işinin işlem zamanı	$j=1,2,...,n$
d_j:	*j* işinin teslim tarihi	$j=1,2,...,n$
a:	öğrenme indeksi	

3.2. Karar değişkeni (Decision variable):

Z_{jr}: Eğer *j* işi *r*. sırada işlem görmek için çizelgelenmişse 1, aksi halde 0,

$j=1,2,...,n \qquad r=1,2,...,n$

3.3. Doğrusal-olmayan matematiksel model (A Non-linear Mathematical Programming Model):

Amaç fonksiyonu:

$Min \quad L_{\max}$

Kısıtlar:

$$\sum_{j=1}^{n} Z_{jr}=1 \qquad r=1,2,...,n \qquad (1)$$

$$\sum_{r=1}^{n} Z_{jr}=1 \qquad j=1,2,...,n \qquad (2)$$

$$p_{[r]}=\sum_{j=1}^{n} Z_{jr}p_j \qquad r=1,2,...,n \qquad (3)$$

$$d_{[r]}=\sum_{j=1}^{n} Z_{jr}d_j \qquad r=1,2,...,n \qquad (4)$$

$$C_r \ge C_{r-1}+p_{[r]}(1+\sum_{j=1}^{r-1}p_{[j]})^a \qquad r=1,2,...,n \qquad (5)$$

$$L_{\max} \ge C_r-d_{[r]} \qquad r=1,2,...,n \qquad (6)$$

$Z_{jr}:0-1$, $C_0=0$ ve diğer tüm değişkenler pozitif tamsayı

Kısıt (1), *r*. pozisyona sadece bir tek işin atanmasını, Kısıt (2), her bir işin sadece bir kez çizelgelenmesini ifade etmektedir. Kısıt (3) ve Kısıt (4) sırasıyla *r*. pozisyondaki işin işlem zamanı ve teslim tarihini göstermektedir. Kısıt (5), *r*. pozisyondaki işin tamamlanma zamanının bir önceki işin tamamlanma zamanı ve *r*. pozisyondaki işin işlem zamanından büyük veya eşit olmasını göstermektedir. Maksimum gecikmenin *r*. pozisyondaki işin gecikmesinin tamamlanma zamanı ve teslim tarihi arasındaki farktan büyük veya eşit olduğunu da Kısıt (6) tanımlamaktadır.

4. DENEYSEL SONUÇLAR (EXPERIMENTAL RESULTS)

Çalışmada bütün deneysel testler Pentium IV/2 GHz 512 RAM kapasiteli kişisel bilgisayarla yapılmıştır. Ele alınan problemin optimal çözümlerini bulmak için CPLEX 10 yazılım versiyonu kullanılmıştır. İş sayıları 8 ile 14 arasında toplam yedi farklı durumda çözülmüştür. İşlem zamanları (p_j) 1 ile 100, teslim tarihleri (d_j) ise 0 ile $C_{\max}$ arasında düzgün dağılımdan üretilmiştir. Buradaki $C_{\max}$, işlerin SPT kuralı ile sıralandığındaki maksimum tamamlanma zamanını ifade etmektedir. Öğrenme indeksi ise Kuo ve Yang [8]' ın çalışmasında kullandığı -0.50 yanında -0.40 ve -0.60 içinde çözümler bulunmuştur. Her alternatif için 30 problem olmak üzere toplam 630 problem çözülmüştür. Deney seti toplu olarak Tablo 6'da verilmiştir.

Tablo 6. Deney seti

Parametre	Alternatif sayısı	Değer
İş sayısı, *n*	7	8,9,10,11,12,13,14
İşlem zamanı, p_j	1	~U[1,100]
Teslim tarihi, d_j	1	~U[0, $C_{\max}$]
Öğrenme indeksi, *a*	3	-0.40, -0.50, -0.60
Çözülen problem sayısı	30	
Toplam çözülen problem	7×1×1×3×30=630	

Tablo 7. Deney setinin CPU çözüm sonuçları (saniye)

a	n	ortalama	maksimum	minimum	standart sapma
-0.40	8	7.27	3.18	12.55	3.34
	9	19.04	11.08	28.88	5.43
	10	87.69	45.36	124.08	26.54
	11	165.17	105.14	208.10	28.34
	12	842.87	364.32	1352.16	308.93
	13	3798.05	2556.23	5296.86	835.27
	14	28930.77	11334.85	45106.68	11071.39
-0.50	8	7.24	1.62	12.32	3.41
	9	18.26	8.73	27.93	5.87
	10	75.12	35.78	118.67	26.74
	11	117.26	76.11	177.90	30.52
	12	867.78	271.37	1485.70	403.24
	13	4050.66	2137.56	5690.60	1033.68
	14	26034.31	8269.47	46498.28	10788.74
-0.60	8	7.92	1.41	13.12	3.62
	9	19.14	7.80	28.66	7.32
	10	90.79	35.57	134.57	32.34
	11	162.36	92.32	224.31	42.78
	12	1086.53	408.77	1555.73	359.01
	13	3908.18	2034.85	5625.27	1223.07
	14	31511.95	9034.86	50849.36	11654.31

Problemin ancak 14 işe kadar makul bir zamanda optimal olarak çözüldüğü Tablo 7'de görülmektedir. 14 işli bir problem için çözüm süresi öğrenme indeksi $a = -0.60$ için yaklaşık 9 saattir. Problemin farklı öğrenme etkilerine göre çözümleri incelendiğinde çözüm sürelerinde önemli bir artış gözlenmemiştir.

5. SONUÇLAR (CONCLUSIONS)

Bu çalışmada zamana bağımlı öğrenme etkili tek makineli çizelgeleme probleminde maksimum gecikme minimizasyonu ele alınmıştır. Problem için doğrusal-olmayan programlama modeli geliştirilmiş ve geliştirilen modelle 14 işe kadar optimal çözümler bulunmuştur.

Bu çalışmadan da görüldüğü gibi çok küçük boyutlu problemler optimal olarak çözülebilmektedir. Bundan sonraki çalışmalarda daha büyük boyutlu problemleri çözmek için sezgisel yöntemler geliştirilebilir. Ayrıca çok makineli durumlar incelenebileceği gibi tek ve/veya çok makineli durumlarda çok ölçütlü çalışmalar da yapılabilir.

KAYNAKLAR (REFERENCES)

1. Biskup, D., "A state-of-the-art review on scheduling with learning effects", **European Journal of Operational Research,** baskıda, 2007.
2. Biskup, D., "Single-machine scheduling with learning considerations", **European Journal of Operational Research**, Cilt 115, 173-178, 1999.
3. Mosheiov, G., "Scheduling problems with a learning effect", **European Journal of Operational Research,** Cilt 132, 687-693, 2001.
4. Moore, J.M., "An n job, one machine sequencing algorithm for minimizing the number of tardy jobs", **Management Science,** Cilt 15, 102–109, 1968.
5. Zhao, C.-L., Q.-L. Zhang and H.-Y. Tang, "Machine scheduling problems with learning effects", **Dynamics of Continuous, Discrete and Impulsive Systems, Series A: Mathematical Analysis,** Cilt 11, 741-750, 2004.
6. Wu, C.-C., Lee, W.-C., Chen, T., "Heuristic algorithms for solving the maximum lateness scheduling problem with learning

considerations", **Computers & Industrial Engineering,** Cilt 52, 124-132, 2007.

7. Eren, T., Güner, E., "Minimizing total tardiness in a scheduling problem with a learning effect", Applied Mathematical Modelling, Cilt 31, 1351-1361, 2007.
8. Kuo, W.-H., Yang, D.-L., "Minimizing the total completion time in a singlemachine scheduling problem with a time-dependent learning effect", **European Journal of Operational Research,** Cilt 174, 1184-1190, 2006.
9. Kuo, W.-H., Yang, D.-L., "Minimizing the makespan in a single machine scheduling problem with a time-based learning effect", **Information Processing Letters,** Cilt 97, 64–67, 2006.
10. Kuo, W.-H., Yang, D.-L., "Single-machine group scheduling with a timedependent learning effect", **Computers and Operations Research,** Cilt 33, 2099-2112, 2006.