

2017-2018 Güz Yarıyılı
Balıkesir Üniversitesi, Endüstri Mühendisliği Bölümü

EMM4129

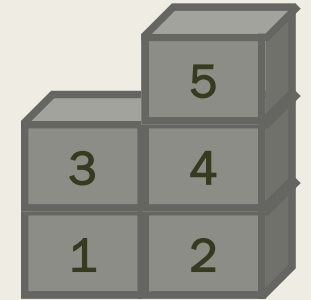
ÇİZELGELEME

(Akış Tipi Atölyelerde Çizelgeleme, Johnson Algoritması, Palmer
Sezgiseli, CDS Algoritması, NEH Algoritması)

Yrd. Doç. Dr. İbrahim KÜÇÜKKOÇ

Web: <http://ikucukkoc.baun.edu.tr>

Email: ikucukkoc@balikesir.edu.tr

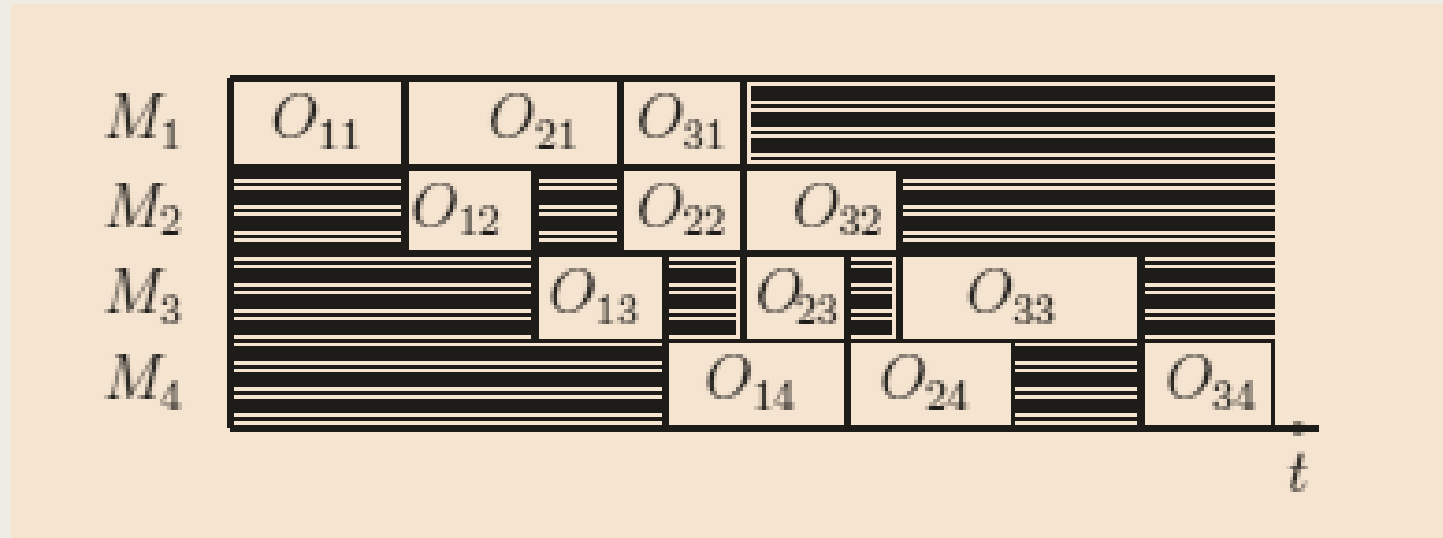


Ders Planı (Vize Sonrası)

- 10. Hafta (K8205): Paralel Makine Çizelgeleme (Öne Geçmeli Durum, Farklı Hızdaki Makineler)
- 11. Hafta (K8205): Akış Tipi Çizelgeleme, Johnson Algoritması, Palmer Sezgiseli, CDS Algoritması, NEH Algoritması
- 12. Hafta (K8205): Atölye Tipi Çizelgeleme
- 13. Hafta (C105): LEKIN Uygulaması – Akış Tipi ve Atölye Tipi Çizelgeleme Örnek Sorular/Çözümler
- 14. Hafta (C105): LEKIN Kullanarak Uygulamalı Sınav
 - *Bir önceki hafta yapılan uygulamaya benzer şekilde, verilecek olan föy üzerinde bulunan akış tipi ve atölye tipi çizelgeleme problemleri LEKIN üzerinden ve/veya manuel olarak çözümlenerek, çizelgeler yorumlanacak ve bazı sorulara cevap aranacaktır.*
 - *Buradan alınacak puanlar, final sınavının 40 puanını oluşturacaktır.*

Akış Tipi Atölyelerde Çizelgeleme

- **Akış tipi** atölyelerde makinelerin, işlerin yapılış biçiminden kaynaklanan, doğal bir sırası vardır.
- Teknolojik baskılar işlerin makineler arasındaki akışını belirler. Bu akış bütün makineler için benzer şekildedir.
- Şöyle ki; eğer J1 işinin M4 makinesinde M3 makinesinden daha önce işlenmesi gerekiyorsa, bu durum diğer bütün işler için de aynen geçerlidir.



Akış Tipi Atölyelerde Çizelgeleme

- $n/m/F/F_{\max}$
- n iş problemi ve m makinede çözüm zordur.
- Akış tipi atölye problemi incelenirse;
 $n/2/F/F_{\max}$ veya $n/2/F/C_{\max}$ problemi çözümlidir.
- $C = r + F$
- $F = \text{Toplam } F_{ij} = W_{ij} + P_{ij}$

Johnson Algoritması – n iş 2 makine

- Johnson algoritması bir grup işin iki makinede (n iş, iki makine) sırasıyla çalışması sisteminde $C_{\max}$ 'ı veya $F_{\max}$ 'ı minimize etmek için optimum çözümü verir.
- Çalışma Prensipleri:
 - *Birinci makineye, n işin içinden birinci makinedeki en kısa süreli olanları tahsis edersek, beklemeyi minimize ederiz.*
 - *İkinci makinedeki en kısa süreli işleri de en sona tahsis etmeye çalışırız.*
- Johnson Sıralama Algoritması olarak da bilinir.
- Tüm işler için r 'ler SIFIR kabul edilir.

Johnson Algoritması – n iş 2 makine

- **1. Adım:** Yapılacak işlerin tümü bir listeye konulur. Buna A listesi diyelim. Ve iki tane boş liste oluşturulur. Bunlara da L1 ve L2 diyelim.
- **2. Adım:** Bütün alt işler arasında en işlem süresi en küçük olan iş bulunur.
 - Eğer bu süre 1. makineye aitse, bu iş L1 listesinin sonuna eklenir ve A listesinden çıkarılır.
 - Eğer bu süre 2. makineye aitse, bu iş L2 listesinin başına eklenir ve A listesinden çıkarılır.
- **3. Adım:** A listesinde iş kalmayana kadar 2. adım tekrar edilir.
- **4. Adım:** L1 ve L2 listeleri birleştirilir.

Johnson Algoritması – Örnek 1

■ 7/2/F/F_{max} Problemi

İşlem Sırası: M1->M2

İş	M1	M2
1	6	3
2	2	9
3	4	3
4	1	8
5	7	1
6	4	5
7	7	6

L1 listesi: 4..

L2 listesi: ...



L1 listesi: 4...

L2 listesi: ...5



L1 listesi: 4-2 ...

L2 listesi: ...3-5 veya ...1-5



...



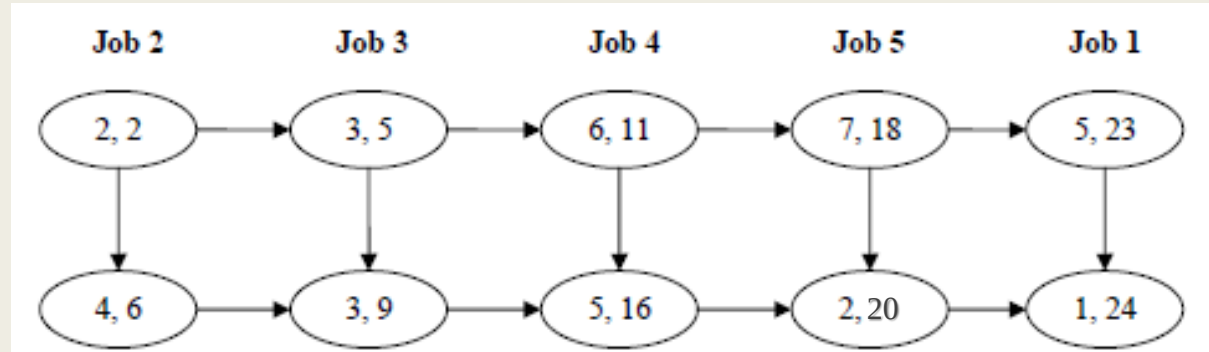
Nihai Sıralama: 4-2-6-7-3-1-5 veya 4-2-6-7-1-3-5

Johnson Algoritması – Örnek 2

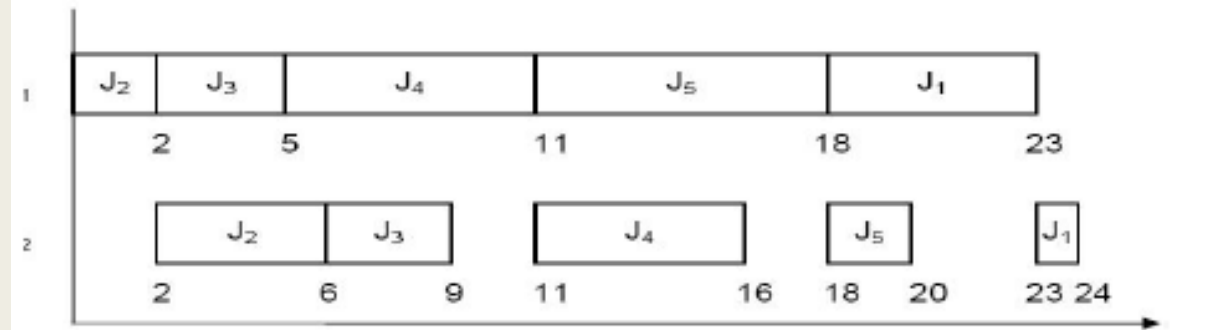
- 5/2/F/Fmax Problemi, İşlem Sırası: M1->M2

Job (j)	j ₁	j ₂	j ₃	j ₄	j ₅
p _{1j}	5	2	3	6	7
p _{2j}	1	4	3	5	2

5 iş 2 makineden oluşan bu problem Johnson algoritmasıyla çözülmüş ve optimum çözüme ait yönlendirilmiş ağ ve Gantt diyagramları yandaki şekilde verilmiştir.



Şekil: J2, J3, J4, J5, J1 sıralaması için yönlendirilmiş ağ



Şekil: J2, J3, J4, J5, J1 sıralaması için Gantt diyagramı

Johnson Algoritması – n iş 3 makine

■ $n/3/F/F_{\max}$ problemi

Özel şartların sağlanması durumunda, n iş 3 makineden oluşan bir akış tipi çizelgeleme probleminde de Johnson algoritması kullanılabilir. Bunun için sağlanması gereken **özel şartlar** şunlardır:

$$\min_{i=1}^n \{P_{i1}\} \geq \max_{i=1}^n \{P_{i2}\} \quad \text{VEYA}$$

$$\min_{i=1}^n \{P_{i3}\} \geq \max_{i=1}^n \{P_{i2}\} \quad \text{olmalı}$$

- Yani Johnson algoritmasının uygulanabilmesi için aslında M2 makinesinin M1 veya M3 makineleri tarafından domine edilmesi (bastırılması) ve darboğaz oluşturmaması gerekiyor.
- Bu şartlar sağlanırsa,
 1. makine işlerinin süreleri, $a_i = P_{i1} + P_{i2}$
 2. makine işlerinin süreleri, $b_i = P_{i3} + P_{i2}$

olarak kabul edilir ve a_i, b_i süreleri kullanılarak Johnson algoritması ile iki kukla makine (M1' ve M2') üzerinden sıralanarak çizelgeleme yapılabilir.

Johnson Algoritması – Örnek 3

■ 6/3/F/Fmax problemi

İş	M1	M2	M3
1	4	1	3
2	6	2	9
3	3	1	2
4	5	3	7
5	8	2	6
6	4	1	1

İki makinaya
indirgeniyor

M1'	M2'
5	4
8	11
4	3
8	10
10	8
5	2

Elde edilen çizelge için Gantt diyagramını çiziniz.

Johnson uygulanır. Sıralama:

2 - 4 - 5 - 1 - 3 - 6

Ya da

4 - 2 - 5 - 1 - 3 - 6

olarak bulunur.

Johnson Algoritması – Örnek 4

- **4/3/F/C_{max}** problemi için işlerin makinelerdeki süreleri aşağıdaki tabloda verilmiştir.

Job (j)	Process time (M1)	Process time (M2)	Process time (M3)
1	8	2	4
2	5	4	5
3	6	1	3
4	7	3	2

M1 ve M3 makineleri için minimum işlem süreleri bulunur: 5 ve 2.
Bu sürelerin, M2 makinesi için maksimum işlem süresinden daha büyük olması şartı aranır.

Şart sağlandığı için Johnson algoritması ile çözüm aranabilir. Birleştirilmiş işlem süreleri aşağıdaki tabloda verilmiştir:

Job (j)	Process time (M ₁)	Process time (M ₂)
1	10	6
2	9	9
3	7	4
4	10	5

Bu halde optimum sıralama J2, J1, J4, J3 olarak elde edilir.

Bu sıralama için elde edilecek çizelgenin yönlendirilmiş ağ diyagramını çiziniz.

Peki n iş 3 makine için özel şartlar sağlanmazsa?

- Hatırlanacağı üzere Johnson algoritmasının kullanılabilmesi için aşağıdaki özel şartların sağlanması gerekmektedir:

$$\min_{i=1}^n \{P_{i1}\} \geq \max_{i=1}^n \{P_{i2}\} \quad \text{VEYA} \quad \min_{i=1}^n \{P_{i3}\} \geq \max_{i=1}^n \{P_{i2}\}$$

- Johnson algoritması, bu özel şartların sağlanmadığı durumlarda da kullanılabilir ve iyi çözümler verir, fakat optimum çözümü garanti etmez.
- Bu tür genelleştirilmiş problemler için Johnson algoritması kullanılarak iyi bir başlangıç çözümü elde edilebilir ve bu çözüm başka prosedürler yardımıyla iyileştirilerek daha iyi çözümler aranabilir.

$m > 3$ iken C_{max} 'ın minimize edilmesi

- n işin m makinede çizelgelenmesi gereken genel bir akış tipi çizelgeleme problemidir. Johnson algoritmasının geliştirilmesinin üzerinden yarım asırdan fazla zaman geçmiş olmasına karşın hala $m > 3$ durumu için akış tipi çizelgeleme probleminde yayılma zamanını optimize edecek bir kural geliştirilememiştir.
- Bu durum kısmen, çalışılan problemin NP-complete (çözülmesi zor) yapısından kaynaklanmaktadır.
- Aşağıdaki algoritmalar, $m > 3$ iken n işin çizelgelenmesi problemi için önerilmiş etkili algoritmalarlardır.

Palmer Sezgiseli

- Bu sezgisel iki adımda uygulanır:

1. n iş m makineden oluşan bir statik akış tipi atölye sisteminde, her iş için A_j değerini hesaplayınız:

$$A_j = - \sum_{i=1}^m \{m - (2i - 1)\} p_{ij}$$

2. İşleri A_j değerlerine göre azalan şekilde sırala.

Örnek: Tabloda verilen F3 | | Cmax problemini Palmer sezgiselini kullanarak çözünüz.

Machines	j_1	j_2	j_3	j_4
M_1	6	8	3	4
M_2	5	1	5	4
M_3	4	4	4	2

Palmer Sezgiseli

$$A_j = - \sum_{i=1}^m \{m - (2i - 1)\} p_{ij} = - \sum_{i=1}^3 \{3 - (2i - 1)\} p_{ij}$$

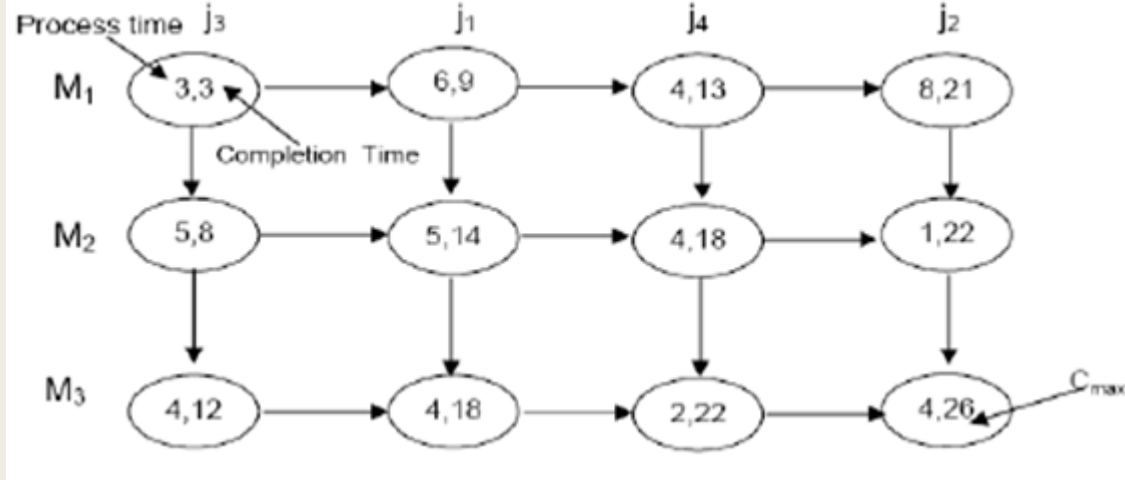
İşlerin A_j değerleri, aşağıdaki şekilde hesaplanır:

	i=1	i=2	i=3	
	$3-(2-1) = 2$	$3-(2 \times 2-1) = 0$	$3-(2 \times 3-1) = -2$	
Job j	p_{1j}	p_{2j}	p_{3j}	A_j
1	6	5	4	-4
2	8	1	4	-8
3	3	5	4	2
4	4	4	2	-4

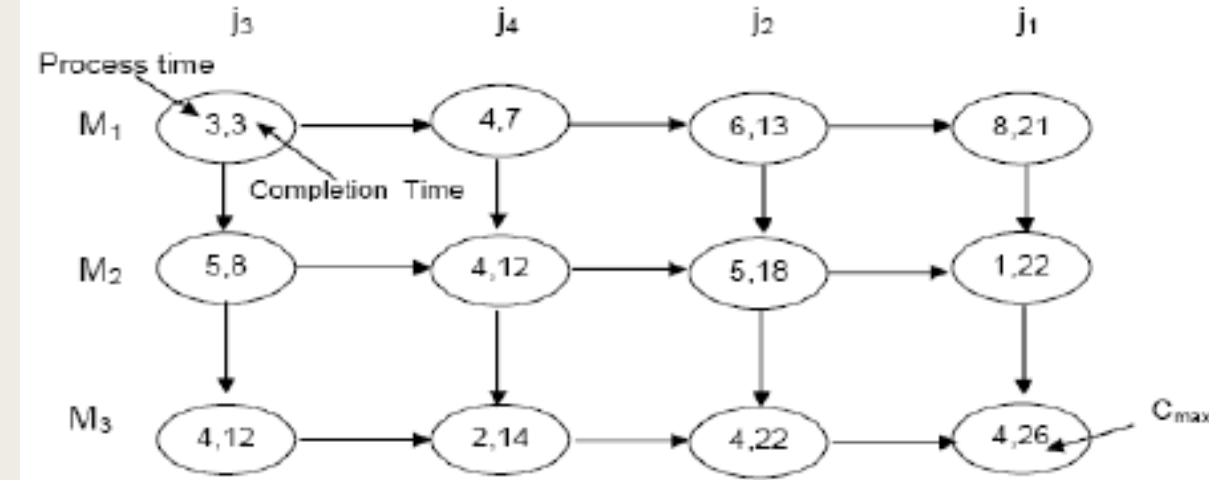
A_j değerleri azalan şekilde sıralanırsa, aşağıdaki sıralamalar elde edilir:

Sıralama 1: {J3, J1, J4, J2} Sıralama 2: {J3, J4, J1, J2}

Palmer Sezgiseli



İlk sıralama için, $\{J_3, J_1, J_4, J_2\}$, elde edilen yönlendirilmiş ağ.



İkinci sıralama için, $\{J_3, J_4, J_1, J_2\}$, elde edilen yönlendirilmiş ağ.

Görüleceği üzere her iki çözüm için de elde edilen yayılma zamanı, $C_{max}=26$ olmaktadır.

Campbell, Dudek ve Smith (CDS) Algoritması

- Bu algoritma, n iş m makineden ($m > 2$) oluşan bir problemi, p adet n iş alt-problemine dönüştürür ($p = m - 1$). Her bir problem Johnson algoritması kullanılarak çözülür ve $C_{\max}$ değerleri bulunur.
- Alt problemlerden minimum $C_{\max}$ değerini veren çizelge, genel çizelge olarak kabul edilir.
- Aşağıdaki 4 iş – 3 makine problemini dikkate alırsak:

Jobs	M_1	M_2	M_3
j_1	P_{11}	P_{21}	P_{31}
j_2	P_{12}	P_{22}	P_{32}
j_3	P_{13}	P_{23}	P_{33}
j_4	P_{14}	P_{24}	P_{34}

2 adet alt problem oluşacaktır.

Campbell, Dudek ve Smith (CDS) Algoritması

- İlk problemde, M1 makinesi M1' kukla makinesi gibi, M3 makinesi ise M2' kukla makinesi gibi düşünülerek Johnson algoritması ile çözülür.

Jobs	$M_1' = M_1$	$M_2' = M_3$
j ₁	P ₁₁	P ₃₁
j ₂	P ₁₂	P ₃₂
j ₃	P ₁₃	P ₃₃
j ₄	P ₁₄	P ₃₄

Campbell, Dudek ve Smith (CDS) Algoritması

- İkinci alt problem ise M_1+M_2 'nin M_1' , M_2+M_3 'ün M_2' gibi dikkate alınması ile oluşturulur.

Jobs	$M_1' = M_1+M_2$	$M_2' = M_2+M_3$
j_1	$P_{11} + P_{21}$	$P_{21} + P_{31}$
j_2	$P_{12} + P_{22}$	$P_{22} + P_{32}$
j_3	$P_{13} + P_{23}$	$P_{23} + P_{33}$
j_4	$P_{14} + P_{24}$	$P_{24} + P_{34}$

CDS algoritması için genelleştirilmiş formül:

This implies that the surrogate problems data will be in generated as follows:

For $k = 1, \dots, m-1$ and $j = 1, \dots, n$ then,

$$M_1' = \sum_{i=1}^k P_{ij} \text{ and } M_2' = \sum_{i=m-k+1}^m P_{ij}$$

Where:

M_1' = the processing time for the first machine

M_2' = the processing time for the second machine

	M_1	M_2	$\dots$	M_m
S_1	M_1			M_m
S_2	M_1+M_2			$M_{m-1}+M_m$
S_3	$M_1+M_2+M_3$			$M_{m-2}+M_{m-1}+M_m$
S_{m-1}	$M_1+M_2+\dots+M_{m-1}$			$M_2+M_3+\dots+M_m$

Campbell, Dudek ve Smith (CDS) Algoritması

Örnek: Aşağıda verilen 4-iş 3-makineden oluşan akış tipi çizelgeleme problemini CDS algoritmasını kullanarak Cmax'ı minimize etmek amacıyla çözünüz.

	j_1	j_2	j_3	j_4
M_1	6	8	3	4
M_2	5	1	5	4
M_3	4	4	4	2

3 adet makine olduğu için $m-1=2$ adet alt problem oluşacaktır. CDS ile çözüm aşağıdaki gibi yapılır.

- **Alt Problem-1:** M_1 makinesi M_1' kukla makinesi, M_3 makinesi de M_2' kukla makinesi olarak düşünülür.

	j_1	j_2	j_3	j_4
$M_1' = M_1$	6	8	3	4
$M_2' = M_3$	4	4	4	2

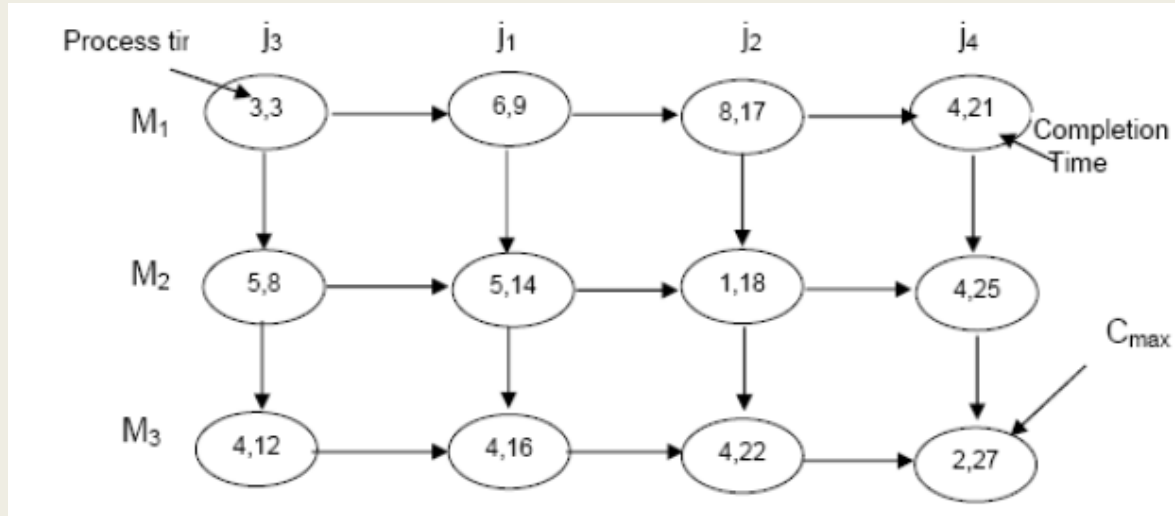
Johnson kuralı uygulanarak sıralama $J_3-J_1-J_2-J_4$ veya $J_3-J_2-J_1-J_4$ olarak elde edilir.

Campbell, Dudek ve Smith (CDS) Algoritması

İlk sıralamaya göre aşağıdaki gibi bir tablo elde edilir:

	j_3	j_1	j_2	j_4
$M_1' = M_1$	3	6	8	4
$M_2' = M_3$	4	4	4	2

Yönlendirilmiş ağ ile ilk sıralamanın çizelge olarak gösterilmesi aşağıdaki gibi olur:

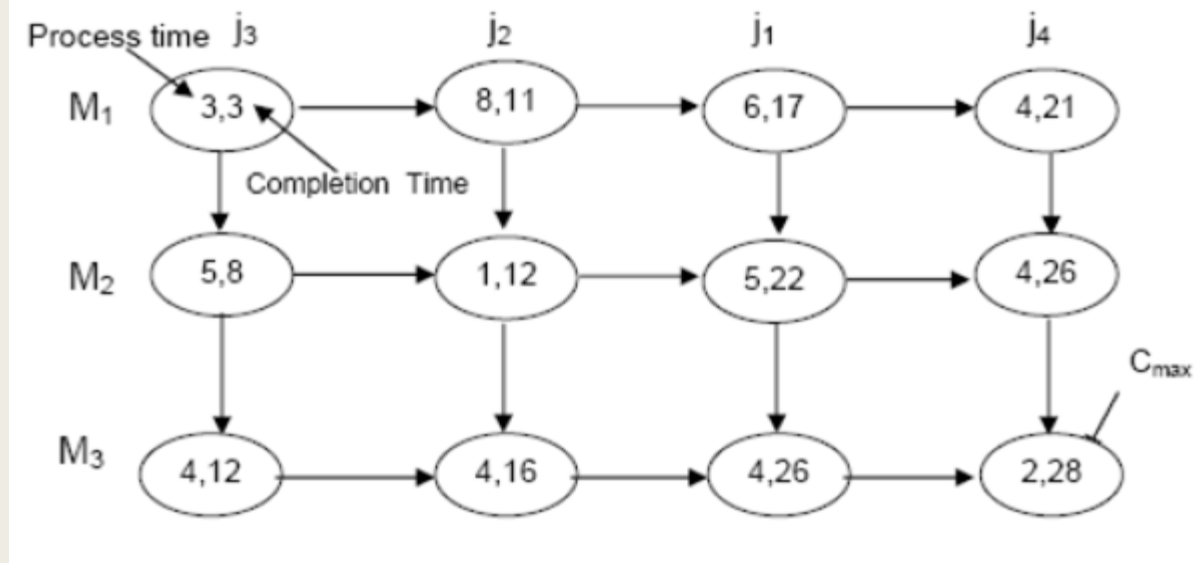


Campbell, Dudek ve Smith (CDS) Algoritması

İkinci sıralamaya göre aşağıdaki gibi bir tablo elde edilir:

	J_3	j_2	j_1	j_4
$M_1' = M_1$	3	8	6	4
$M_2' = M_3$	4	4	4	2

Yönlendirilmiş ağ ile ikinci sıralamanın çizelge olarak gösterilmesi ise aşağıdaki gibi olur:



Campbell, Dudek ve Smith (CDS) Algoritması

- **Alt problem-2:** Orijinal problemdeki veriler kullanılarak ikinci alt problem aşağıdaki gibi elde edilir:

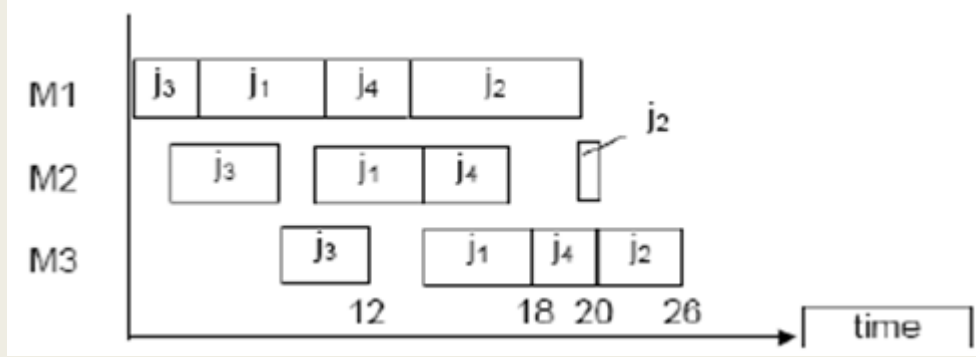
	j_1	j_2	j_3	j_4
$M_1' = M_1 + M_2$	11	9	8	8
$M_2' = M_2 + M_3$	9	5	9	6

Bu durumda Johnson kuralı ile elde edilecek olan sıralama ise J3-J1-J4-J2 olur. $C_{\max}$ ise tabloda gösterildiği gibi 26 olarak elde edilir.

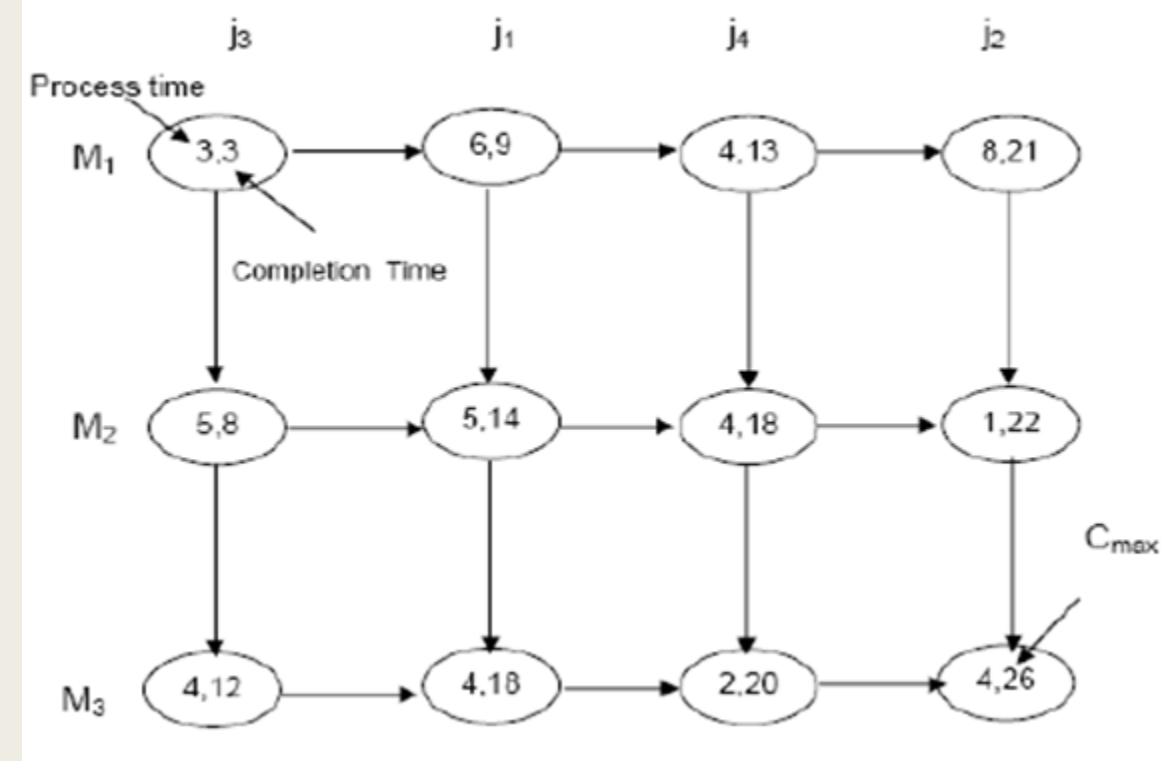
Machine	j_3	j_1	j_4	j_2	C_3	C_1	C_4	C_2	$C_{\max}$
M_1	3	6	4	8	3	9	13	21	
M_2	5	5	4	1	8	14	18	22	
M_3	4	4	2	4	12	18	20	26	26

Campbell, Dudek ve Smith (CDS) Algoritması

- J3-J1-J4-J2 sıralaması alt problemlerde en iyi C_{max} değerini veren sıralama olduğu için bu sıralama en iyi sıralama olarak kabul edilir ve orijinal problem için çizelge oluşturulur.



Sonuç olarak minimum C_{max} değeri J3-J1-J4-J2 sıralaması kullanılarak 26 olarak bulunmuştur.



Nawaz, Enscor ve Ham (NEH) Algoritması

- NEH algoritması iterasyonlar şeklinde ilerler. Algoritmanın çalışma prensibini açıklayan adımlar aşağıdaki gibidir:
- Adım-1: Her bir işin, toplam iş süresi (tüm makinelerde yapılması gereken sürelerin toplamı) bulunur: $T_j = \sum_{i=1}^m p_{ij}$
- Adım-2: İşler T_j değerlerine göre azalan şekilde bir listede sıralanır.
- Adım-3: Listedeki ilk iki iş seçilir ve yerleri değiştirilerek iki alternatif kısmi sıralama oluşturulur. Bu durumda oluşturulan kısmi sıralamaların çizelgelenmesi sonucu oluşacak Cmax değerleri bulunur. Cmax değeri yüksek olan sıralama gözardı edilir. Cmax değeri düşük olan sıralama '*mevcut sıralama*' olarak adlandırılır.
- Adım-4: Listedeki bir sonraki iş seçilir ve mevcut sıralamadaki tüm yerlere koyularak deneme yapılır. Elde edilecek tüm sıralamaların Cmax değerleri hesaplanır.
- Adım-5: Cmax değeri yüksek olan sıralama gözardı edilir. Cmax değeri düşük olan sıralama '*mevcut sıralama*' olarak adlandırılır.
- Adım-6: Listedeki sıralanmamış iş kaldıysa Adım-4'e dön, yoksa dur.

Nawaz, Enscore ve Ham (NEH) Algoritması

Örnek: Aşağıdaki F3 | | Cmax problemini NEH algoritması yardımıyla çözünüz:

	j_1	j_2	j_3	j_4
M_1	6	8	3	4
M_2	5	1	5	4
M_3	4	4	4	2

Çözüm: İşlerin T_j değerleri hesaplanır.

	j_1	j_2	j_3	j_4
M1	6	8	3	4
M2	5	1	5	4
M3	4	4	4	2
T_j	15	13	12	10

T_j değerlerine göre azalan liste: J1-J2-J3-J4

Nawaz, Enscore ve Ham (NEH) Algoritması

- Listenin ilk iki sırasındaki işler (J_1 ve J_2)'den oluşan kısmi sıralamalar: $S_{12**} = \{J_1, J_2, *, *\}$ veya $S_{21**} = \{J_2, J_1, *, *\}$ şeklinde olabilir.
- S_{12**} kısmi sıralaması için oluşan kısmi çizelgenin $C_{\max}$ değeri 19 olur:

	j_1	j_2	C_1	C_2	$C_{\max}$
M_1	6	8	6	14	
M_2	5	1	11	15	
M_3	4	4	15	19	19

- S_{21**} kısmi sıralaması için oluşan kısmi çizelgenin $C_{\max}$ değeri ise 23 olur:

	j_2	j_1	C_1	C_2	$C_{\max}$
M_1	8	6	8	14	
M_2	1	5	9	19	
M_3	4	4	13	23	23

Nawaz, Enscor ve Ham (NEH) Algoritması

- İki sıralama karşılaştırılırsa, $S_{12^{**}}$ sıralamasının diğerine göre üstün olduğu görülür. Dolayısıyla bundan sonraki araştırmalar $S_{12^{**}}$ sıralaması üzerinden devam edecek, $S_{21^{**}}$ sıralaması göz ardı edilecektir.

Schedule	$C_{\max}$
$S_{21^{**}}: (j_2, j_1, *, *)$	23
$S_{12^{**}}: (j_1, j_2, *, *)$	19

Nawaz, Enscor ve Ham (NEH) Algoritması

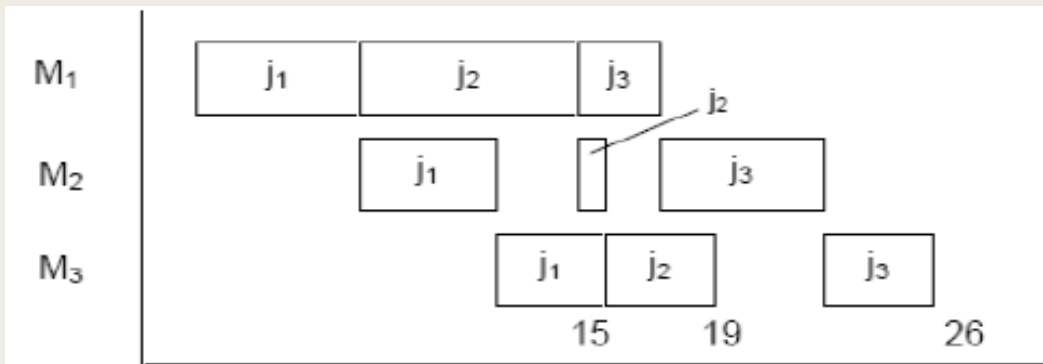
- Şimdi, listede J1 ve J2 işlerinden sonra gelen, J3 işi için denemeler yapılacaktır.
- J3 işi, mevcut sıralamada 3 farklı şekilde yerleştirilebilir:
 - *J1 işinden önce: Yeni kısmi sıralama $S_{312*} = \{J_3, J_1, J_2, *\}$*
 - *J1 işinden sonra: Yeni kısmi sıralama $S_{132*} = \{J_1, J_3, J_2, *\}$*
 - *J2 işinden sonra: Yeni kısmi sıralama $S_{123*} = \{J_1, J_2, J_3, *\}$*

Nawaz, Enscore ve Ham (NEH) Algoritması

$S_{123*} = \{J_1, J_2, J_3, *\}$ kısmi sıralaması için Cmax hesaplaması aşağıdaki gibidir:

	j_1	j_2	j_3	C_1	C_2	C_3	C_{max}
M_1	6	8	3	6	14	17	
M_2	5	1	5	11	15	22	
M_3	4	4	4	15	19	26	26

$S_{123*} = \{J_1, J_2, J_3, *\}$ kısmi sıralaması için Gantt şeması:



Nawaz, Enscore ve Ham (NEH) Algoritması

$S_{312*} = \{J_3, J_1, J_2, *\}$ kısmi sıralaması için Cmax hesaplaması aşağıdaki gibidir:

	j_3	j_1	j_2	C_3	C_1	C_2	$C_{\max}$
M_1	3	6	8	3	9	17	
M_2	5	5	1	8	14	18	
M_3	4	4	4	12	18	22	22

Son olarak, $S_{132*} = \{J_1, J_3, J_2, *\}$ kısmi sıralaması için Cmax hesaplaması aşağıdaki gibidir:

	j_1	j_3	j_2	C_1	C_3	C_2	$C_{\max}$
M_1	6	3	8	6	9	17	
M_2	5	5	1	11	16	18	
M_3	4	4	4	15	20	24	24

Nawaz, Enscore ve Ham (NEH) Algoritması

- Elde edilen kısmi çizelgeler kıyaslandığında, $S_{312*} = \{J_3, J_1, J_2, *\}$ kısmi çizelgesinin minimum $C_{\max}$ oluşturduğu gözlenir.

Partial Schedule	$C_{\max}$
$S_{123*} (j_1, j_2, j_3, *)$	26
$S_{312*} (j_3, j_1, j_2, *)$	22
$S_{132*} (j_1, j_3, j_2, *)$	24

- Dolayısıyla araştırmaya $S_{312*} = \{J_3, J_1, J_2, *\}$ kısmi çizelgesi üzerinden devam edilir.

Nawaz, Enscor ve Ham (NEH) Algoritması

- Şimdi, listede J1, J2 ve J3 işlerinden sonra gelen, son iş olan J4 için denemeler yapılacaktır.
- J4 işi, mevcut sıralamada 4 farklı şekilde yerleştirilebilir:
 - J3 işinden önce: Yeni kısmi sıralama $S_{4312} = \{J_4, J_3, J_1, J_2\}$
 - J3 işinden sonra: Yeni kısmi sıralama $S_{3412} = \{J_3, J_4, J_1, J_2\}$
 - J2 işinden önce: Yeni kısmi sıralama $S_{3142} = \{J_3, J_1, J_4, J_2\}$
 - J2 işinden sonra: Yeni kısmi sıralama $S_{3124} = \{J_3, J_1, J_2, J_4\}$

$S_{4312} = \{J_4, J_3, J_1, J_2\}$ kısmi sıralaması için Cmax hesaplaması aşağıdaki gibidir.

	j ₄	j ₃	j ₁	j ₂	C ₄	C ₃	C ₁	C ₂	C _{max}
M ₁	4	3	6	8	4	7	13	21	
M ₂	4	5	5	1	8	13	18	22	
M ₃	2	4	4	4	10	17	22	26	26

Nawaz, Enscore ve Ham (NEH) Algoritması

$S_{3412} = \{J_3, J_4, J_1, J_2\}$ kısmi sıralaması için Cmax hesaplaması aşağıdaki gibidir.

	j_3	j_4	j_1	j_2	C_3	C_4	C_1	C_2	$C_{\max}$
M_1	3	4	6	8	3	7	13	21	
M_2	5	4	5	1	8	12	18	22	
M_3	4	2	4	4	12	14	22	26	26

$S_{3142} = \{J_3, J_1, J_4, J_2\}$ kısmi sıralaması için Cmax hesaplaması aşağıdaki gibidir.

	j_3	j_1	j_4	j_2	C_3	C_1	C_4	C_2	$C_{\max}$
M_1	3	6	4	8	3	9	13	21	
M_2	5	5	4	1	8	14	18	22	
M_3	4	4	2	4	12	18	20	26	26

Nawaz, Enscore ve Ham (NEH) Algoritması

Son olarak, $S_{3124} = \{J_3, J_1, J_2, J_4\}$ kısmi sıralaması için $C_{\max}$ hesaplaması aşağıdaki gibidir.

	j_3	j_1	j_2	j_4	C_3	C_1	C_2	C_4	$C_{\max}$
M_1	3	6	8	4	3	9	17	21	
M_2	5	5	1	4	8	14	18	25	
M_3	4	4	4	2	12	18	19	27	27

Elde edilen kısmi çizelgeler kıyaslandığında, üç alternatif sıralama için aynı $C_{\max}$ değerinin (26 zb) elde edildiği görülür.

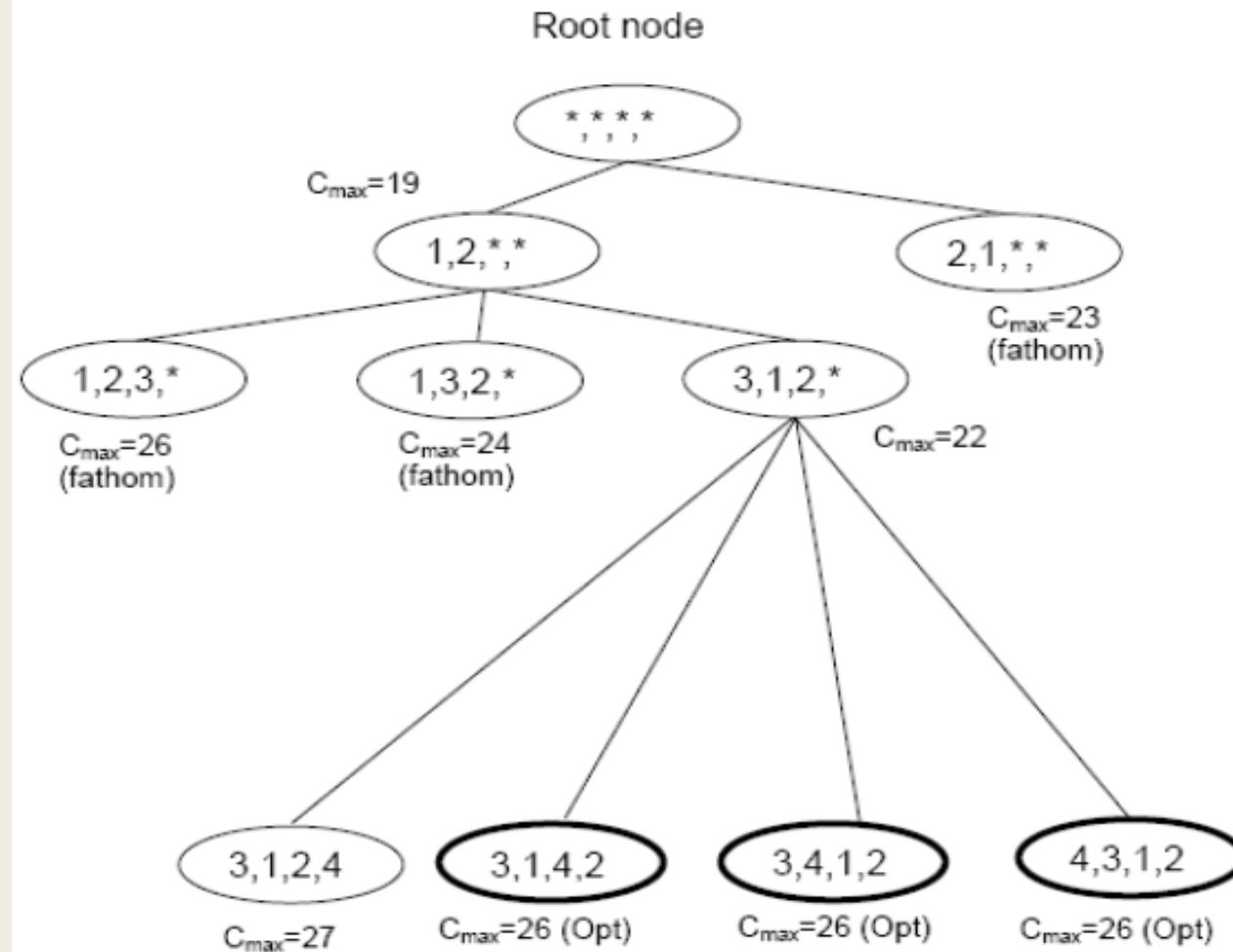
Schedule	$C_{\max}$
$S_{3124} : (j_3, j_1, j_2, j_4)$	27
$S_{3124} : (j_4, j_3, j_1, j_2)$	26
$S_{3412} : (j_3, j_4, j_1, j_2)$	26
$S_{3142} : (j_3, j_1, j_4, j_2)$	26

Bu durumda üç alternatif çözüm vardır:

$\{J_4, J_3, J_1, J_2\}$; $\{J_3, J_4, J_1, J_2\}$ veya $\{J_3, J_1, J_4, J_2\}$

Nawaz, Enscore ve Ham (NEH) Algoritması

- NEH algoritması için elde edilen çözümün aşamalarını gösteren arama ağacı aşağıdaki gibi gösterilebilir.



Kaynaklar

- Principles of Sequencing and Scheduling, Kenneth R. Baker, Dan Trietsch, John Wiley & Sons, New Jersey, 2009.
- Algorithms for Sequencing and Scheduling, Ibrahim M. Alharkan, King Saud University.
- Scheduling: Theory, Algorithms, and Systems, Michael Pinedo, Springer, 2012.
- Üretimde Sıralama ve Çizelgeleme Ders Notları, Yrd.Doç.Dr. A. Ayça Supçiller, Pamukkale Üniversitesi, 2014.
- İş Sıralama ve Çizelgeleme Ders Notları, Prof.Dr. Hüseyin Başlıgil, Yıldız Teknik Üniversitesi, 2013.
- Üretim Çizelgeleme Ders Notları, Yrd.Doç.Dr. Mert Topoyan, Dokuz Eylül Üniversitesi, 2017.
- Üretim ve Servis Sistemlerinde Planlama ve Çizelgeleme, Yad. Doç. Dr. Zehra Kamışlı Öztürk, Anadolu Üniversitesi, 2012