

T.C.

İstanbul Üniversitesi

Sosyal Bilimler Enstitüsü

Üretim Anabilim Dalı

Yüksek Lisans Tezi

Hücreyel İmalat Sisteminde Hücre ve Yerleşim Düzeni Tasarımı

Ateş ATAMTÜRK

25010960822

TEZ DANIŞMANI

Doç. Dr. Necdet Özçakar

İstanbul 2009

TEZ ONAYI

..... Anabilim- Bilim Dalında
..... Numaralı
.....'ın hazırladığı
“.....” konulu **YÜKSEK LİSANS /
DOKTORA-TEZİ** ile ilgili **TEZ SAVUNMA SINAVI**, Lisansüstü Öğretim
Yönetmeliği'nin 15.Maddesi uyarınca, ...
günü saat/.....'de yapılmış, sorulan sorulara alınan cevaplar sonunda adayın
tezinin'ne* **OYBİRLİĞİ /OYÇOKLUĞUYLA** karar
verilmiştir.

JÜRİ ÜYESİ	KANAATİ (2)	İMZA

HÜCRESEL İMALAT SİSTEMİNDE HÜCRE VE YERLEŞİM TASARIMI

ATEŞ ATAMTÜRK

Yüksek Lisans, Üretim Ana Bilim Dalı

Tez Yöneticisi: Doç. Dr. Necdet Özçakar

Aralık 2009

ÖZ

Hücresel imalat Grup Teknolojisinin bir uygulamasıdır. Hücresel imalata geçişte birinci adım, makinelerin hücelere, parçaların ailelere gruplanmasıdır. Daha önceki çalışmalar hücre oluşturmak için gerçek üretim çevresinde yetersiz olan sıfır bir makine parça matrisini kullanmışlardır.

Bu tezde daha gerçekçi olan operasyon sıra tabanlı hücre oluşturma metotları incelenmiştir. Gerçek bir hücre oluşturma ve yerleşim düzeni problem akış matrisi tabanlı sezgisel algoritma ile analiz edilmiştir.

CELL AND LAYOUT DESIGN IN CELLULAR MANUFACTURING

ATES ATAMTURK

MS, Department of Manufacturing

Supervisor: Doc. Dr. Necdet Ozcakar

December 2009

ABSTRACT

Cellular manufacturing is an application of Group Technology. The first step in the transition to the cellular manufacturing is grouping the machines into cells and parts into families. Early studies used zero one machine part incidence matrix to perform cell formation which is insufficient for the real life manufacturing environment.

In this thesis, operation sequence based cell formation methods were studied which are more realistic. A real life cell formation and layout problem was analyzed by flow matrix based heuristic algorithm.

ÖNSÖZ

Üretim dünyasında artan rekabet koşulları parti üretimi yapan işletmelerde seri üretim avantajlarından yararlanma gereksinimini arttırmaktadır. Bu amaca ulaşmak için en elverişli disiplinlerden biri olan Grup Teknolojisi ve onun üretim sistemlerine uygulanmış biçimi olan hücresel imalat konusu son yıllarda oldukça hızla gelişen bir üretim şekli olarak karşımıza çıkmaktadır. Bu tezde, ilk zamanlar teorik yöntemlerle kısmen üretim ortamına uygulanan hücresel imalatın, giderek gerçek üretim koşullarının kısıtlarını göz önüne alarak gelişimi ve son olarak üretim akış bilgisini kullanan daha gerçekçi hücre oluşturma ve hücre yerleşim düzeni belirleme yöntemleri incelenmiştir. Tezin sonunda ise yapılan uygulama ile üretim akış bilgisini kullanan yöntemlerin üstünlükleri analiz edilmiştir.

Yönlendirmeleriyle tezin şekillenmesini sağlayan ve bilimsel araştırma yöntemleri konusunda yardımlarını esirgemeyen tez danışmanım Sn. Doç. Dr. Necdet Özçakar'a teşekkür ederim.

İÇİNDEKİLER

TEZ ONAYI	i
ÖZ	iii
ABSTRACT	iv
ÖNSÖZ	v
ŞEKİLLER	ix
BİRİNCİ BÖLÜM: ÜRETİM SİSTEMLERİNE GENEL BİR BAKIŞ	3
1.1 Üretim Sistemi	3
1.2 Üretim Sistemlerinin Sınıflandırılması	3
1.2.1 Sürekli Tip Üretim Sistemi	3
1.2.2 Atölye Tipi Üretim Sistemi	5
1.2.3 Proje Tipi Üretim Sistemi	6
İKİNCİ BÖLÜM: GRUP TEKNOLOJİSİ KAVRAMI VE HÜCRESEL İMALAT	9
2 Giriş	9
2.1 Grup Teknolojisi ve Tarihsel Gelişimi	9
2.2 Parça Gruplama	10
2.3 Hücresel İmalat	12
2.4 Hücresel İmalatın uygulanması	16
2.5 Hücre Oluşturma Problemi	18
2.6 Hücre Oluşturma Yöntemlerinin Gelişimi	21
2.7 Hücre Oluşturma Problemlerinde Operasyon Sıra Verisi Kullanan Modeller	30
2.7.1 Matematiksel Programlama	30
2.7.2 Ağ Analizi Metodu	31
2.7.3 Malzeme Akış Analizi Metodu	31
2.7.4 Sezgisel Metot	32
ÜÇÜNCÜ BÖLÜM: HÜCRE OLUŞTURMA METOTLARI	34
3 Giriş	34
3.1 İkili Makine-Parça Matrisi	34
3.1.1 Hiyerarşik Metotlar	35
3.1.2 Hiyerarşik Olmayan Metotlar	43

3.1.3 Dizi Tabanlı (array based) Kümelenendirme Metotları	50
3.2 İkili Makine Parça Matrisini Kullanan Metotlar İçin Genel Değerlendirme	57
3.3 Operasyon Sırasını Kullanan Metotlar	58
3.3.1 Operasyon Sırasını Kullanan Benzerlik Katsayıları	60
3.3.2 Makine Hücrelerinin Gruplanması ve Yerleştirilmesi	60
3.3.3 Tavlama Benzetimi Yaklaşımı (Simulated Annealing Approach)	64
3.3.4 Çoklu Benzer Makinelerle İmalat Hücresi Oluşturma İçin Tavlama Benzetimi Yaklaşımı	67
3.3.5 Hücre Oluşturma için Sıra Verili Bir Kümelenendirme Algoritması (CASE)	80
3.3.6 Genetik Algoritmaları Kullanan Metotlar	88
3.3.7 Operasyon Sıra Verisini Kullanan Melez Sezgisel Algoritmalar	97
3.3.8 Hücre Oluşturma Problemi İçin Dal-Sınır'la İyileştirilmiş Genetik Algoritma	98
3.3.9 Doğrusal Hat Boyunca Hücre Atama Probleminin Grafik Metot Kullanılarak Çözümü	111
DÖRDÜNCÜ BÖLÜM: HÜCRE OLUŞTURMA İÇİN AKIŞ MATRİSİ TEMELLİ SEZGİSEL METOT	118
4 Giriş	118
4.1 Akış Matrisi Girdisi	118
4.2 Hücre Oluşturma ve Yerleşim Tasarımı Algoritması	119
4.3. Hücre Oluşturma ve Yerleşim Düzeni Tasarımı İçin Performans Göstergesi	122
4.3.1. Hücre Hareket İndeksi	122
4.3.2. Hücre kullanım İndeksi	122
4.3.3. Toplam Hareket İndeksi	123
BEŞİNCİ BÖLÜM: AKIŞ MATRİSİ TEMELLİ SEZGİSEL ALGORİTMA İÇİN HÜCRE OLUŞTURMA VE YERLEŞİM DÜZENİ TASARIMI İÇİN BİR UYGULAMA	124
5. Giriş	124
5.1. Ürün Gamı	125

5.2.	Fabrikanın Mevcut Yerleşim Düzeni	126
5.3.	Derece Sıra Algoritmasının Uygulanması.....	129
5.4.	CASE Algoritmasının Uygulanması	129
5.5.	Akış Matrisi Tabanlı Sezgisel Algoritmanın Uygulanması	135
SONUÇ VE DEĞERLENDİRME		139
BİBLİYOGRAFYA.....		142

ŞEKİLLER

Şekil 1. n işlemlili bir sürekli tip üretim sistemi.....	4
Şekil 2. Atölye tipi üretim sistemi.....	6
Şekil 3. Proje tipi üretim sistemi.....	7
Şekil 4. Hücresel imalat sisteminin ana yapısı	13
Şekil 5. Hücresel imalat sistemleri.....	14
Şekil 6. Makinelerin bir hücresel imalat sistemi içinde düzenlenmesi.....	15
Şekil 7. Grup teknolojisi için sınıflandırılmalı gözden geçirme çerçevesi	27
Şekil 8. Hücre oluşturma metotlarının sınıflandırılması	28
Şekil 9. Çözüm yaklaşımlarının sınıflandırılması	28
Şekil 10. Benzerlik katsayılarına göre oluşturulmuş dendogram	37
Şekil 11. Ortalama bağlantılı kümelenendirme örneğine ait dendogram	41
Şekil 12. Makine hücrelerinin gruplanması ve yerleştirilmesi için 1. aşama akış diagramı.....	64
Şekil 13. Permutasyon kodlamanın kullanıldığı kromozom	91
Şekil 14. 6 makine için 3 hücre çözümlü akış çizgesi	91
Şekil 15. Rulet tekerleği seçme operatörü.....	93
Şekil 16. GA'da çaprazlama operatörünün uygulanması	94
Şekil 17 GA'da mutasyon operatörünün uygulanması	94
Şekil 18. Dal Sınır Yönteminde sayım ağacı.....	95
Şekil 19. Tablo 48'de verilen makine parça matrisinin çizgesi.....	102
Şekil 20. Çaprazlama sonucunda oluşan bireyler.....	105
Şekil 21. Yavru 6'da yapılan mutasyon işlemi.....	105
Şekil 22. Dal-sınır iyileştirme.....	106
Şekil 23. Çizgenin alt çevrimlere bölünmesi.....	107
Şekil 24. Çizge için w1 ve w3 alt çevrimlerinin toplamı.....	107
Şekil 25 Dallandırma kuralının gösterimi.....	108
Şekil 26 İş istasyonları ve AS/R makinesinin gösterimi	111
Şekil 27. (a) N=4 iş istasyonlu problem için 1-yer şebeke. (b) iş istasyonu 3'ün atamasının yapılmış olduğu N=4 iş istasyonu için 2-yer şebeke.....	114
Şekil 28. Grafik metodu için 4 makine ve 3 ürün için örnek.....	116

Şekil 29. Grafik metodu sayısal örneğinin 1. adımı.	116
Şekil 30. grafik metodu sayısal örneğinin 2. adımı.	117
Şekil 31. Hücre oluşturma ve yerleşim tasarımı algoritması akışı	121
Şekil 32. Fabrikanın mevcut yerleşim düzeni.....	127
Şekil 33. Uygulama sonrası işletmenin yeni yerleşim düzeni.	138

TABLÖLAR

Tablo 1. Farklı üretim sistemlerinin özelliklerinin karşılaştırması.....	8
Tablo 2. 1990’larda hücre tasarımı içi tek kriterli bazı modeller	26
Tablo 3. Makine parça matrisi	36
Tablo 4. Benzerlik katsayıları matrisi.....	37
Tablo 5. Ortalama bağlantılı kümelendirme algoritması için örnek makine parça matrisi.....	38
Tablo 6. Ortalama bağlantılı kümelendirme algoritması örneği için benzerlik katsayıları matrisi	39
Tablo 7. "1" Eşik değerine göre 1. iterasyon sonucu benzerlik katsayıları matrisi.....	39
Tablo 8. "0,8" Eşik değerine göre 2. iterasyon sonucu benzerlik katsayıları matrisi.....	39
Tablo 9 "0,5" Eşik değerine göre 3. iterasyon sonucu benzerlik katsayıları matrisi.....	40
Tablo 10 "0,4" Eşik değerine göre oluşturulan benzerlik katsayıları matrisi ...	40
Tablo 11 "0,3" Eşik değerine göre oluşturulan benzerlik katsayıları matrisi	40
Tablo 12. Tek bağlantılı, tam bağlantılı ve ortalama bağlantılı kümelendirme metodlarının avantaj ve dezavantaj karşılaştırması	42
Tablo 13. GRAFICS algoritmasına ait makine parça matrisi	46
Tablo 14. GRAFICS algoritması örneğine ait benzerlik katsayıları matrisi. ...	46
Tablo 15. GRAFICS algoritması örneği için uygulanabilir çözüm.....	47
Tablo 16. GRAFICS algoritması örneği için blok diyagonal matris.....	48
Tablo 17. Küme tanımlama algoritması için başlangıç matrisi örneği.....	49
Tablo 18. Küme tanımlama algoritması 1. iterasyon.....	49
Tablo 19. Küme tanımlama algoritması 1. iterasyon sonucu.	49
Tablo 20. Küme tanımlama algoritması 2. iterasyon.....	49
Tablo 21. Küme tanımlama algoritması 3. iterasyon.....	50
Tablo 22. Küme tanımlama algoritması sonucunda düzenlenen makine parça matrisi.....	50

Tablo 23. Derece sıra algoritması için örnek makine parça ve başlangıç matrisi.	52
Tablo 24. Derece sıra algoritması örneği 1. iterasyon sonucu.	52
Tablo 25. Derece sıra algoritması örneği 2. iterasyon sonucu.	53
Tablo 26. Derece sıra algoritması örneği 3. iterasyon sonucu.	53
Tablo 27. Derece sıra algoritması örneği 4. iterasyon sonucu.	53
Tablo 28. Değiştirilmiş derece sıra algoritmasına ait başlangıç makine parça matrisi.	56
Tablo 29. Değiştirilmiş derece sıra algoritması 2. iterasyon sonuç matrisi ve blok arama prosedürünün uygulanma şekli.	56
Tablo 30. Değiştirilmiş derece sıra algoritması 2. aşama sonunda bulunan temel hücreler ve parça aileleri.	56
Tablo 31. Değiştirilmiş derece sıra algoritması 3. aşamaya ait benzerlik katsayıları matrisi.	57
Tablo 32. Değiştirilmiş derece sıra algoritması 3. aşama sonucunda hiyerarşik olarak oluşturulan makine hücreleri ve parça aileleri.	58
Tablo 33. Operasyon sıra verisine göre düzenlenmiş bir makine parça matrisi.	59
Tablo 34. Makine hücrelerinin gruplanması ve yerleştirilmesi yöntemi için operasyon sırasına dayanan makine parça matrisi.	62
Tablo 35. Operasyonlar arası akış matrisi.	73
Tablo 36. Operasyonlar için makine tipi ve kullanım oranı.	74
Tablo 37. Makine tipi 1 için operasyon atama sonucu.	74
Tablo 38. Makine tipi 2 için operasyon atama sonucu.	75
Tablo 39. Makine tipi 3 için operasyon atama sonucu.	75
Tablo 40. Makine tipi 4 için operasyon atama sonucu.	75
Tablo 41. Makineler arası akış matrisi.	76
Tablo 42. İlk iterasyonda atanmamış makinelerle hücreler arasındaki ortalama akış.	77
Tablo 43. 2. iterasyonda atanmamış makinelerle hücreler arasındaki ortalama akış.	77

Tablo 44. Tüm makineler hücrelere atandıktan sonra tavlama benzetimi algoritmasının başlangıç çözümü.....	77
Tablo 45. Başlangıç çözümü için operasyonlar arası akış matrisinin blok diyagonal yapısı.	78
Tablo 46. Makineler arası akış matrisi blok diyagonal yapısı.....	79
Tablo 47. Örnek problem için en iyi çözüm.	79
Tablo 48. CASE algoritması örneği için operasyon sıra verili makine parça matrisi.....	86
Tablo 49. CASE algoritmasında benzerlik katsayıları matrisi.	86
Tablo 50. CASE algoritması örneğinde 1. iterasyon sonucu makine parça matrisi ve grup teknoloji etkinliği eleman değerleri.	88
Tablo 51. 6 makinenin 3 hücreye atandığı çözümün genetik zincirine örnek	90
Tablo 52. 3 hücreli ve 9 kenarlı çizge için ikili kod	92
Tablo 53. Makine-parça matrisine göre parça rotaları.....	101
Tablo 54. Genetik algoritma için rastgele üretilmiş başlangıç popülasyonu..	103
Tablo 55. 27 makine ve 17 iş istasyonundan oluşan operasyon sıra matrisi. .	128
Tablo 56. Uygulamanın Derece Sıra Algoritmasıyla Çözümü.....	129
Tablo 57. Makine parça matrisi için dönüşüm matrisi.	130
Tablo 58. CASE benzerlik katsayıları matrisi.	131
Tablo 59. Parça çekirdeği 1'e göre SC_{kj} değerlerini gösteren tablo.....	132
Tablo 60. Parça çekirdeği 2'ye göre SC_{kj} değerlerini gösteren tablo.....	132
Tablo 61. CASE algoritmasında 1. İterasyonda parçaların hücrelere atanmasından sonra makine parça matrisinin durumu.	133
Tablo 62. İterasyon 2'de parçaların atanmasından sonra makine parça matrisinin durumu.	134
Tablo 63. CASE algoritması 3. iterasyon sonucu makine parça matrisi.	135
Tablo 64. Operasyon akış matrisi.	136
Tablo 65. Akış matrisi tabanlı sezgisel algoritma uygulaması sonrası makine parça matrisi.	137
Tablo 66. Performans hesaplamalarının özeti.	137
Tablo 67. Uygulamada yapılan hücre oluşturma performanslarının karşılaştırılması.	141

Giriş

Üretim yapan işletmelerin temel amacı istenen zamanda, istenen kalitede ve uygun fiyatta üretim yaparak kar elde etmektir. Sanayi devriminden 1920'lere kadar gelinen aşamada işletmeler bu amaca ulaşmak için genellikle emek yoğun metotları benimsediler. Bu metotla eğitilmiş işçiler ve basit ve çok amaçlı araç gereçlerle tüketicinin değişik ihtiyaçlarına cevap verildi. Daha sonraki dönemde Henry Ford ve Alfred Sloan tarafından geliştirilen seri üretim yöntemi sanayide büyük ilgi gördü. Bu metot sayesinde üretilen ürünler bir üretim hattı boyunca bölünmüş olan operasyonlar ve ürüne özgü makine ve teçhizat ile daha kısa zamanda üretilabiliyordu. Ancak çok çeşitli ürünü çok amaçlı makineler kullanarak üretmek zorunda olan işletmeler bu kavramı doğal olarak kendi işletmelerine uygulayamadılar. Üretim şekli gereği, bu işletmeler için makine ayar zamanları, proses içi stoklar, malzeme taşıma maliyetleri, işçilik ve ekipman maliyetleri gibi faktörler minimize edilmesi gereken problemler olarak kaldı.

Parti halinde üretim yapan işletmeler için, seri üretim avantajlarından yararlanma fırsatı ilk defa Mitrofanov tarafından 1966'da ortaya atılan Grup Teknolojisi kavramı ile ortaya çıktı. "Grup teknolojisi, üretilen parçaların tasarım ve üretim karakteristiklerine göre gruplandırmaya çalışan bir yönetim felsefesidir" (Heragu, 1994: 203-215). Üretilen parçaların tasarım ve üretim şekillerinden doğan bu grupta felsefesi ilk defa üretim ortamında uygulanabileceği Wemmerlov ve Hyer tarafından ortaya atıldığında Hücreli İmalat Sistemleri ortaya çıktı. Böylece üretim ortamında birbirinden farklı fonksiyonlara sahip makinelerden oluşan imalat hücrelerine, gruplanan parça aileleri atandığında ayar sürelerinin kısaldığı, parçaların atölyede fazla yol kat etmeden hemen hemen aynı hücre içerisinde operasyonlarını tamamlama olanağı olduğu görüldü. Bu yaklaşım sayesinde çok çeşitli üretim gamına sahip işletmeler seri üretimin avantajlarından yararlanarak esnek bir imalat sistemi içinde üretim yapabilme olanağı buldular.

Önceleri yapılan makine ve parça gruplamada parçaların makine ihtiyaçlarının dikkate alındığı gruplama algoritmaları kullanıldı. Bu metotlar genel olarak sıfır-bir ikili sayı matrisi kullanırlar ve matrisin diyagonalinde makine parça kümelerinin olduğu bloklarla sonlanırlar. Bu türde yapılan kümelendirmelerin operasyon

sıraları, üretim hacimleri, işleme zamanları, hücreler arası hareketler v.b. kısıtları dikkate almaması gerçek üretim şartlarına uygulama zorlukları doğurdu. Bu nedenle son zamanlarda yapılan çalışmalar daha çok aralıklı veri ve sıra verilerini girdi olarak kullanan algoritmalar üzerine yoğunlaşmıştır.

Makine parça kümelenirmede sıra verisinin kullanımı, hücre formasyonu problemi üzerinde çalışan kullanıcıya birçok değerli bilgiyi aynı anda sağlar. Bu bilgiler sayesinde sadece hücrelerin oluşturulması sağlanmaz, aynı zamanda hücre içindeki makinelerin yerleşim düzenleri de tanımlanabilir. Sıra verisi kullanılarak hücrelerin biçimlendirilmesi ilk defa Selvam ve Balasubramanian tarafından 1985’de önerildi (Yin ve Yasuda, 2006: 329-352). Sıra verisini kullanan sezgisel yaklaşımlarla geliştirilen algoritmalar ve özellikle son yıllarda kullanılan genetik algoritmalar maksimum hücre sayısı, üretim hacmi, maksimum hücre büyüklüğü, operasyon süreleri gibi üretim kısıtların dahil edilmesiyle, daha yüksek performansa sahip hücre formasyon çözümlerine ulaşılmıştır.

Bu tezde, grup teknolojisi kavramı ve onun üretim sistemlerine uygulanmasından doğan hücreli imalat sistemlerinin literatür incelemesinden sonra hücre oluşturma probleminin ikili makine parça matrisini kullanan metotlara yer verilmiştir. Daha sonra araştırmaların özellikle daha gerçekçi sonuçlar vermesi nedeniyle operasyon sıra verisini kullanan metotlara yer verilmiş ve hücre oluşturma ve hücre yerleşim düzenini aynı anda kullanan sezgisel bir algoritma incelenmiştir. İncelenen algoritmanın önerilen performans kriterleriyle diğer metotlara göre daha iyi performans gösterdiği belirlenmiştir.

BİRİNCİ BÖLÜM: ÜRETİM SİSTEMLERİNE GENEL BİR BAKIŞ

1.1 Üretim Sistemi

Belirli bir mal veya hizmeti üretmek amacıyla, belirli miktar ve çeşitteki girdi üzerinde, ekonomik değer katacak işlemleri yapmak üzere, malzeme, makine, insan, metot ve çevre elemanlarıyla oluşturulan sistemlere üretim sistemleri denir. Üretim sistemlerinin çıktıları ise yarı mamul veya mamuldür. İşte bu girdi ve çıktı arasında kalan üretim sistemleri, ürettikleri çıktının cinsine, çıktıya olan talebe, sistemde yer alan üretim faktörlerin fiziksel kısıtlarına, diğer ekonomik kısıtlara göre çeşitli şekillerde ortaya çıkar.

1.2 Üretim Sistemlerinin Sınıflandırılması

Birçok üretim yönetimi uzmanı, üretim sistemlerini farklı şekillerde sınıflandırmaktadırlar. Ancak bunların içinde en genel olanı ve en çok kabul göreni aşağıda yapılan sınıflandırmadır (Yamak, 1993: 27-30). Buna göre üretim sistemleri aşağıdaki gibi sınıflandırılmaktadır:

1. Sürekli tip (Flow Shop)
2. Görev tipi (Job Shop)
3. Proje tipi (Project Shop)

Sanayinin her kesiminde bu 3 tür üretim sisteminin örneklerini bulmak olanağı vardır. Dolayısıyla, sektör bazında veya bir üretim dalında, belli bir sistemin egemen olduğunu söylemek mümkün değildir.

1.2.1 Sürekli Tip Üretim Sistemi

Bu tip üretim sistemlerinde tüm üretim araçları üretilen ürün tipi veya tiplerine özgü tasarlanmıştır. Diğer bir deyişle, “sürekli tip üretim sistemi özel amaçlı, tek fonksiyonlu makineleri kullanan ürün akış yerleşim düzeni ile karakterizedir” (Kamrahi ve Parsaei, 1994: 450-464). Üretilen ürün tipi sınırlı kalmak kaydı ile birden fazla olabilir. Ürün tipi birden fazla olduğu durumlarda üretim makine ve ekipmanında küçük ayar değişiklikleri yapmak sureti ile bir ürün tipinin üretiminden diğerine geçmek mümkün olur. Sürekli üretim sisteminde üretilen ürün miktarlarının

çok yüksek olması da bu üretim sisteminin diğer bir özelliğidir. Dolayısıyla bu üretim sistemi uzun süreli, yüksek hacimli ve düzenli bir talebe ihtiyaç gösterir. Üretilen ürünün talebinde büyük bir düşme veya tamamen ortadan kalkması durumunda, tamamen yeni bir ürüne geçiş kısa sürede sağlanamaz. Çünkü bu üretim tipinde kullanılan, makine ve ekipman üretilen ürüne özgü tasarlandığından, esnek bir yapıya sahip değildir. Bununla birlikte sisteme giren girdilerin de süreklilik arz etmesi, sistemdeki akışın devam etmesini sağlar. Girdilerdeki kesiklik direkt olarak sisteme yansır ve sistemde duruşlara neden olur. Aynı şekilde sisteme ait makine veya ekipmandaki bir arıza veya duruş, tüm sistemin tamamen veya kısmen durmasına neden olur.

Sürekli üretim tipinin yukarıda bahsedilen özelliklerinden de anlaşılacağı gibi, sistemin üretim hızı diğer üretim tiplerine oranla çok yüksektir. Bunun dışında yine diğer üretim tiplerine oranla sistemin yönetimi, geliştirilmesi, yapılan değişikliklerin yerleştirilmesi ve izlenmesi de bu üretim tipinin sağladığı avantajlardandır. Sistemde yer alan işlem birimlerini kolay yönetilebilir küçük parçalara bölmek suretiyle departmanlaşma sağlanmış olur. Örnek olarak, tornalama, delme, frezeleme, taşlama şeklinde giden bir işlemler dizisinde yer alan benzer işlemlerin sırasını değiştirmeden ayrı departmanlarda toplanması, ürünün montaj hattına dengeli akmasını sağlar.

Buradan da anlaşılacağı üzere sürekli tip üretim sisteminde üretim hızı diğer üretim tiplerine oranla çok yüksektir. Bunun dışında bu üretim tipinin her açıdan yönetimi kolaydır. Diğer üretim tiplerinde olduğu gibi uzun hazırlık zamanlarına ihtiyaç duymaz. Üretilen ürün hacimleri yüksek olduğundan sistemde bir kez ayar yapıldıktan sonra, sistem uzun süre aynı ürünü üretir. Sistemin daha hızlı ve etkin çalışabilmesi için geliştirme faaliyetleri daha kolay yürütülür. Yapılan işlem dizileri hep aynı olduğundan izlenmesi, incelenmesi ve geliştirilmesi kolaydır. Ayrıca proses içi stoklar ve hammadde stokları çok düşük düzeyde olduğu gibi, tam zamanında üretim uygulaması olan sistemlerde sıfır stok ile çalışılır.



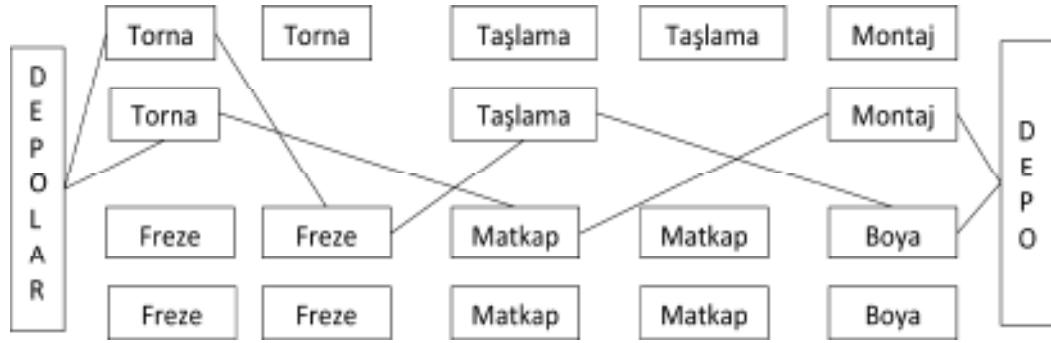
Şekil 1. n işlemlili bir sürekli tip üretim sistemi (Yamak, 1993: 27-30).

1.2.2 Atölye Tipi Üretim Sistemi

Bu tip üretim sistemlerinde kullanılan makine ve ekipman, üniversal olarak da tabir edilen, çok amaçlı, çok çeşitli ürünlerin üretilmesine olanak veren üretim araçlarıdır. Üretilen ürünler birbirinden çoğunlukla farklı ve çok sayıda çeşide sahip, buna karşın üretim adetleri düşüktür. Her bir ürün çeşidine ait bağlama aparatı, kesici takım, teknik bilgi paketi, makine ayarı v.b. birbirinden kısmen veya tamamen farklıdır. Bu farklılık beraberinde kalifiye işgücü gerektirir.

Sürekli tip üretim sisteminde olduğundan farklı olarak, bu üretim sistemi ürettiği ürünlerin talep yapısından etkilenmez. Sistem yeni bir ürünü bünyesine kolaylıkla alabilir. İşletmenin bir üründen diğerine geçişte kapasite kullanım oranının düşmesine karşın, bu üretim tipi daha geniş bir ürün çeşidini pazara sunma avantajını sağlar.

Atölye tipi üretim sisteminde makine ve ekipmanın yerleşim düzeni, fonksiyonel yerleşim düzenine sahip olduğundan, işlem gören malzemelerin atölye içinde kat ettikleri mesafeler çok fazladır. Bu tip üretim sistemine sahip bir talaşlı imalat atölyesinde örneğin tüm tornalar bir yerde, tüm matkaplar ayrı bir yerde v.s. bulunurlar. Böylece tornalanacak olan malzeme önce torna bölümüne gelir, delik delme için matkapların bulunduğu bölüme gider. Bu durum proses içi stokların artmasına da yol açar. Bunun dışında üretilen ürünler birbirinden farklı olduğundan, sırası gelen işlem için makine veya iş merkezine gelen malzeme için yeni baştan makine ayarı, takım ve aparat değişimi v.b. yapılır. Bu durum standart zamanın artmasına neden olur. Hatta bazı durumlarda, teknik bilgi paketinin ilgili departmandan gelmesi bile zaman kaybına neden olabilir. Üretilen ürünlerde standadizasyon düşük oranda olduğundan prosesin hatalı ürün üretme riski vardır. Kalite iyileştirme çalışmaları uzun zaman alır ve etkinliği düşüktür. Atölye tipi üretim sisteminin bu olumsuzluklarını ortadan kaldırmak için hücresel imalata geçiş en etkili yoldur. Zira hücresel imalatın asıl amacı atölye tipi üretimi, sürekli tip üretim sistemine yaklaştırmak olduğundan, bahsi geçen kayıp zamanlar en aza indirilir.



Şekil 2. Atölye tipi üretim sistemi (Kamrahi ve Parsaei, 1994: 450-464).

1.2.3 Proje Tipi Üretim Sistemi

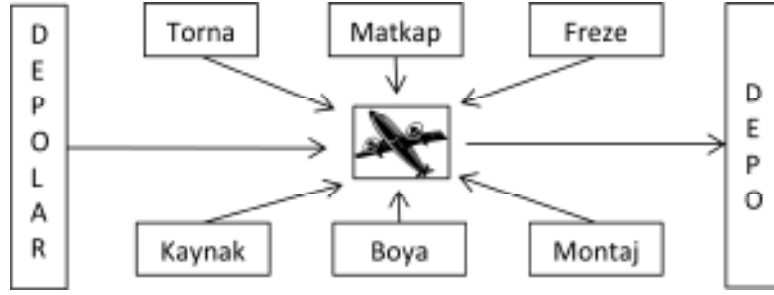
Proje tipi üretim yukarıda verilen sistemlerden tamamen farklıdır. Proje tipi üretim sisteminde üretilen ürün bir proje kapsamında ele alınır. Proje belirli adımlardan oluşur ve bu adımların en erken ve en geç tamamlanma süreleri mevcuttur.

Proje tipi üretimde üretilen ürün çoğunlukla bir kez üretilir ve üretim tamamlandıktan sonra proje ömrünü tamamlar. Hatta bazı durumlarda üretim tesisi tamamen ortadan kalkar. Örneğin bir baraj üretiminde baraj tamamlandıktan sonra üretimde kullanılan tüm tesisler yerlerinden kaldırılır. Bu üretim tipinde çoğunlukla üretim faaliyetlerinde kullanılan makine, ekipman ürünün etrafında yer alır. Gerekliğinde işini tamamlayan makine yerine kaldırılır.

Projenin zaman planlaması bu süreler göz önüne alınarak yapılır. Projelerin planlanmasında CPM, PERT, GANNT gibi tekniklerden yararlanılır.

Proje tipi üretim yukarıda verilen özellikleri nedeniyle diğer üretim sistemlerinden tamamen ayrılır.

Dolayısıyla ne sürekli bir akıştan, ne de kesikli bir üretimden söz edilebilir. Sisteme giren girdiler çok sayıda ve çeşitte olup, bu girdilerin sürekli ve kesikli gelmesi sistemi etkilemez. Ayrıca bu sistem talep yapısından etkilenmez. Çünkü diğer üretim sistemlerinden farklı olarak burada talep bir kerelik oluşur ve sistem bu talebi karşıladıktan sonra mevcudiyetini yitirir. Ayrıca burada yerleşim düzeni üretim araçlarının, üretilen ürün etrafında toplandığı bir yapıdadır (Şekil 3).



Şekil 3. Proje tipi üretim sistemi (Kamrahi ve Parsaei, 1994: 450-464).

Aşağıda verilen Tablo 1’de farklı üretim sistemlerinin makine çeşitleri, proses tasarımı, hazırlık süresi, proses içi stok, parti miktarı, termin süresi kriterleri açısından karşılaştırılması görülmektedir.

Tablo 1. Farklı üretim sistemlerinin özelliklerinin karşılaştırması (Kamrahi ve Parsaei, 1994: 450-464).

ÖZELLİK	Atölye Tipi Job Shop	Akış Tipi Flow Shop	Proje Tipi Project Shop	Hücreyel
Makine Çeşitleri	Esnek, genel amaçlı	Özel ve tek amaçlı	Genel ve mobilize	Esnek ve programlanabilir
Malzeme Taşıma	Elle taşıma	Transfer hattı	Elle ve otomatik	Otomatik taşıma
Hazırlık Süresi	Uzun, değişken	Uzun	Değişken	Tek
Parti Miktarı	Orta ve küçük hacimli parti	Büyük hacimli parti	Küçük hacimli parti	Küçük hacimli parti
Proses İçi Stoklar	Büyük	Büyük	Değişken	Düşük
Kalite Kontrol	Rastgele örnekleme	Rastgele örnekleme	%100 kalite kontrol	%100 kalite kontrol

İKİNCİ BÖLÜM: GRUP TEKNOLOJİSİ KAVRAMI VE HÜCRESEL İMALAT

2 Giriş

Grup teknolojisi, bir imalat sisteminde mevcut olan çok çeşitlilikteki üretim parçalarının, parçaların tasarım (şekil, boyut, tolerans v.b.), ve imalat (üretim için gereken makineler, prosesler v.b.) özelliklerini kullanarak belli sınıflar altında toplamak ve bu sınıflandırmayı imalat ortamında bir takım kazançlar sağlamak için kullanmaya dayalı bir disiplindir. Grup teknolojisinin kullanılarak imalat makinelerinin atölye ortamında gruplanması ve benzer özellikteki parça ailelerinin bu makine gruplarında üretilmesine dayanan üretim şekli ise hücreli imalatıdır.

2.1 Grup Teknolojisi ve Tarihsel Gelişimi

Üretimde Grup Teknolojisi kavramının kullanıldığı ilk tarih 1925'dir (Kamrahi ve Parsaei, 1994: 450-464). Bu tarihte R.E.Flanders Grup Teknolojisine benzer bir imalat kavramı kullanmıştır. Flanders proses, taşıma minimizasyonu ve işin gözle kontrolden çok, ürünün standardizasyonu ve ürüne göre departmanlaşmayı önermiştir. Önerilen bu kavram daha sonra Jones ve Lamson makine şirketinde uygulanmıştır. 1933 yılında Almanya'da Brady ve daha sonra 1938 yılında A. P. Sokolovski Sovyetler Birliği'nde "Parts be classified and parts with similar features be manufactured together" isimli yayınında parçaların gruplanabileceğini ve benzer özelliklere sahip olan bu parça gruplarının birlikte standart prosesle üretilebileceğini ortaya koymuştur. Sokolovsky çalışmasını genişleterek proses tabanlı bir sınıflandırma metodu geliştirmiştir. Yine 1938 yılında J.C. Kerr Üretim Mühendisleri Enstitüsüne sunduğu makalesinde makinelerin gruplanmasına dayanan atölye tipi üretim yerleşim düzenini tartışmıştır. 1949 yılında İsveç'te A. Korling "Group Production" isimli makalesinde seri üretim tekniklerinin parti üretimine adaptasyonundan bahsetmiştir. Hemen hemen 1960'ların sonuna kadar grup teknolojisinin gelişimi Sovyetler Birliği'nde olmuştur. 1953-1965 yılları arasında bu ülkede 750'den fazla işletmede grup teknolojisi uygulanmaya başlamıştır. 1950-57 yılları arasındaki süreçte birden çok parça ailelerine tek aile-makine yaklaşımının gelişmesiyle grup teknolojisinin temelleri ortaya çıktı (Gallagher, Grayson, ve Phillips, 1971: 229-237). imalat sanayinde artan rekabet atölye tipi üretim yapan işletmelerde rekabet gücünün

arttırılması gerekliliğini doğurmuştur. Tüketicilerin artan ihtiyaçları ürün çeşitliliğini arttırmış ve üreticilerin bu ihtiyaçları karşılamak için her geçen gün bünyelerine kattıkları yeni ürünleri üretime almaları üretim ortamında üretim partilerinin karmaşık rotalar izlemesine, makineler arasında proses içi stokların artmasına, bazı makinelerin dar boğaz oluşturup bazılarının boş kalmasına, bir partinin operasyonundan sonra kısa aralıklarla ayar yapma zorunluluğunun ortaya çıkmasına, kalitesizlik maliyetlerinin artmasına neden olmuştur. 1959 yılında S P Mitrofanov Sovyetler Birliğinde “Scientific Principles of Group Technology isimli kitabını yayınlamıştır. Mitrofanov tarafından yayınlanan grup teknolojisi kavramı geniş çeşitlilikteki parçaların tasarım ve imalat özelliklerinin kullanılarak gruplanabileceğini ortaya koydu. Grup teknolojisi, benzer tasarım ve/veya üretim karakteristiklerine sahip ürünlerin gruplandırmasına çalışan yönetim felsefesidir (Heragu, 1994: 203-215). Bu yönetim felsefesine göre artık işletmelerdeki ürün çeşitliliği azaltılmış, üretim planlama, ürün tasarımı gibi konularda birçok kolaylıklar sağlanmıştı. Grup teknolojisinin asıl temellerinin atıldığı bu kitapta Mitrofonov ayrıca Sovyetler Birliğinde 800 fabrikada GT uygulama sonuçlarını vermiştir. Mirofonov’un kitabı 1966 yılında İngilizceye çevrilmiştir. 1960’ların başında Opitz çok çeşitli ürün üreten firmalarda iş parçası istatistikleri üzerinde araştırma yapmış ve parçaları geometrik özelliklerine göre kodlayarak sınıflandıran bir sistem kurmuştur. 1960 yılında Burbidge üretim akış analizi olarak bilinen GT için sistematik planlama yaklaşımını geliştirmiştir. Bundan sonra Grup Teknolojisi özellikle Rusya, doğu ve batı Avrupa ülkelerinde yaygınlaştı. Bu sırada Amerika Birleşik Devletlerinde ise gruplandırma ve kodlama ile ilgili bazı belirtiler görülmüştür. Burbidge’in üretim akış analizinden sonra GT probleminin formülasyonu ve çözümüne gerçek anlamda analitik yaklaşım 1972 yılında Mc Auley tarafından yapılmıştır (Chandrasekharan ve Rajagopalan, 1986: 1221-1233). Mc Auley makine ve parçaları temsil etmek için bir sıfır-bir matrisi ve grupları bulmak için de bir tek bağlantılı kümelenendirme algoritması kullanmıştır. Kümelenendirme Jacard’ın benzerlik katsayısı temelinde yapılmıştır. Mc Auley’in çalışmasından sonra parça ve makine gruplandırma çalışmaları hız kazanmıştır.

2.2 Parça Gruplama

GT ve hücreli imalat yapılandırma çalışmalarında ilk yapılacak iş sistemdeki

tüm parçaların gruplanarak parça ailelerinin oluşturulması ve daha sonra makinelerin gruplanarak imalat hücrelerinin oluşturulmasıdır. Parça ailelerinin oluşturulması genellikle zahmetli bir işlemdir ve dikkatli bir planlama gerektirir. Parçaların imalat ve tasarım benzerliklerinin kullanılarak yapılan gruplama işleminde kullanılan üç değişik metot vardır (Heragu, 1994: 203-215). Bunlardan ilki görsel olarak yapılan gruplamalıdır. Burada parçalar hakkında yeterli bilgi ve tecrübesi olan gruplayıcı parçaların şekil, boyut, çap, malzeme gibi özelliklerini kullanarak gruplama yapar. Ancak bu yöntem yeteri kadar kapsamlı olmadığı ve insan inisiyatifinde olduğundan verimli değildir. İkinci yöntem Burbidge tarafından önerilen üretim akış analizidir. Üretim akış analizi parçaların rota bilgilerinden yararlanılarak yapılır. Parçaların talep ettikleri proses özelliklerine göre yapılan bu gruplamada rota bilgilerinin doğruluğu büyük önem kazanır. Üçüncü gruplandırma yöntemi ise sınıflandırma ve kodlama yöntemidir. Burada parçalar tasarım ve proses özelliklerine göre bağımsız olarak değerlendirilerek sınıflanır ve ardından kodlanır. Literatürde kullanılan üç çeşit kodlama yöntemi vardır. Monokod kodlama yönteminde kod içindeki her bir basamağın tek başına anlamı yoktur, basamaklar önceki basamaklarla birleştirildiğinde anlam kazanırlar. Polikod yönteminde her bir basamak ayrı bir özelliği tanımlar. Her iki metodun birleştirilerek kullanılmasıyla hibrid kodlama sistemi ortaya çıkar. En çok kullanılan bu kodlama sistemi bir monokod yapısı içinde küçük polikod serileri içerir. Literatürde iyi bilinen sınıflama ve kodlama sistemlerine örnek vermek gerekirse BRISCH, CODE, MICLASS, OPITZ, KC-1 en çok kullanılan yöntemlerdir (Ballakur ve Steudel, 1987: 639-665). İyi tasarlanmış bir parça sınıflandırma ve kodlama sisteminin yararları genel olarak aşağıdaki gibidir (Kamrahi ve Parsaei, 1994: 450-464):

- a) Parça ailelerinin ve makine hücrelerinin oluşturulmasına yardımcı olur.
- b) Tasarım, teknik resim ve proses planlarının kolay elde edilmesini sağlar.
- c) Tasarımların yinelenmesine engel olur.
- d) Makine, ekipman ihtiyaçlarının kolay tahmin edilmesini sağlar.
- e) Güvenilir iş parçası istatistikleri sağlar.

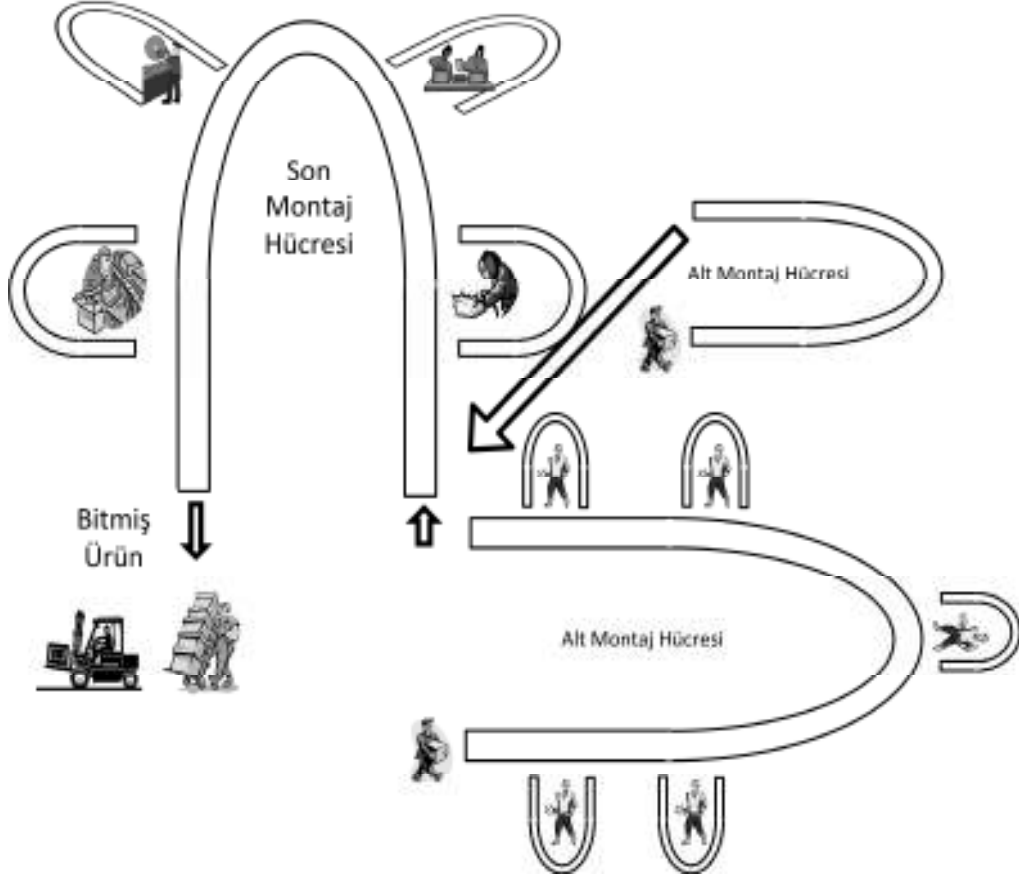
- f) Üretim planlama ve çizelgeleme prosedürlerine yardım eder.
- g) Maliyet tahminini geliştirir ve maliyet muhasebesi prosedürlerinin kurulmasını sağlar.
- h) Daha iyi makine kullanımı ve daha iyi aparat, takım ve insan gücü kullanımı sağlar.

2.3 Hücresel İmalat

Grup teknolojisi felsefesindeki gelişmeler, parça ailelerinin imalat ortamında makine ve teçhizat arasında yapılacak gruplamalarla üretilebileceğini gösterdi. Bu nedenle, hücresel imalat, grup teknolojisinin başarılı bir imalat uygulamasıdır (Mahdavi, Shirazi, ve Paydar, 2008: 943-953). Grup üretim ve aile programlama olarak da bilinen hücresel imalat, parça ailelerini, operatör veya robot tarafından çalıştırılan tek bir hat veya makine hücresi içinde üreten üretim prosesi olarak da tanımlanabilir (Kamrahi ve Parsaei, 1994: 450-464). İdeal olarak bir hücre, bir parçanın başından sonuna kadar tek ve sürekli bir akışta işlenebildiği (geriye akış olmadan), fiziksel olarak yakın bir şekilde konumlandırılmış birbirine benzemeyen makineler grubudur (Ballakur ve Steudel, 1987: 639-665). Burada belirtilen geriye akış olmama durumu açıktır ki akış tipi üretime yaklaşma hedefinin en önemli kriteridir. Bu nedenle tezin son bölümünde incelenecek olan sezgisel metot bu kriterle ilgili bir de performans ölçütü önermektedir. İmalat hücreleri makine ekipman ve işgücü bakımından insanlı ve insansız olarak 2'ye ayrılırlar. İnsanlı hücrelerde operatörler birden fazla makineye nezaret ederler. İnsansız hücrelerde ise operatörlerin yaptığı işler robot, otomatik malzeme taşıma sistemi, programlanabilir makinelerle aktarılır. İnsanlı ve insansız hücrelerin yerleşimine ilişkin bir örnek Şekil 5'de görülebilir. Özellikle insanlı imalat hücrelerinde taşıma hareketlerinin en küçüklenmesi amacıyla genellikle "U" şeklinde yerleşim yapılır. Bu tip yerleşimde işlenecek parçalar U'nun bir bacağından girip sırayla ilgili makinelerle uğradıktan sonra U'nun diğer bacağından hücredeki işlemlerini tamamlamış olarak hücreyi terk ederler (Şekil 4).

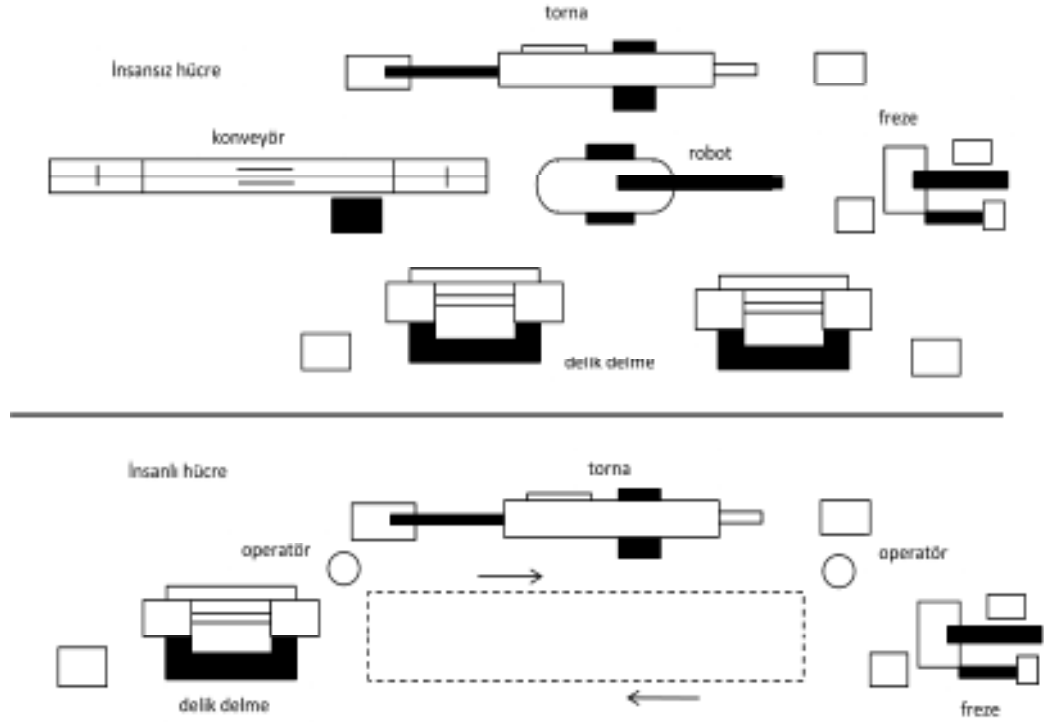
Grup teknolojisi, üretim sistemlerine 3 yolla uygulanabilir (Wemmerlov ve Hyer, 1989: 1512). Parti üretim ortamında ortak olan en basit uygulama ayar

etkinliğini sağlamak için parça benzerliklerini temel almaktır. İkinci uygulama



Şekil 4. Hücresel imalat sisteminin ana yapısı (Wang ve K. B. Wu, 2001: 181-188).

formal parça ailelerini oluşturmak ve ekipmanları bu ailelere atamaktır (ancak ekipmanlar oldukları durumda bırakılır). Grup teknolojisinin en son uygulaması ise imalat hücrelerini oluşturmaktır. Bir hücresel imalat sisteminin amacı düşük talep yapısına sahip geniş çeşitlilikteki ürünleri yüksek verimle üretmektir. Hücresel imalat, genellikle fonksiyon olarak benzer olmayan birkaç makine bir hücre içerisinde gruplanır. Böylece hücreye atanmış olan parça ailesi hemen hemen tüm operasyonlarını hücre içinde tamamlayarak ürüne dönüşür.

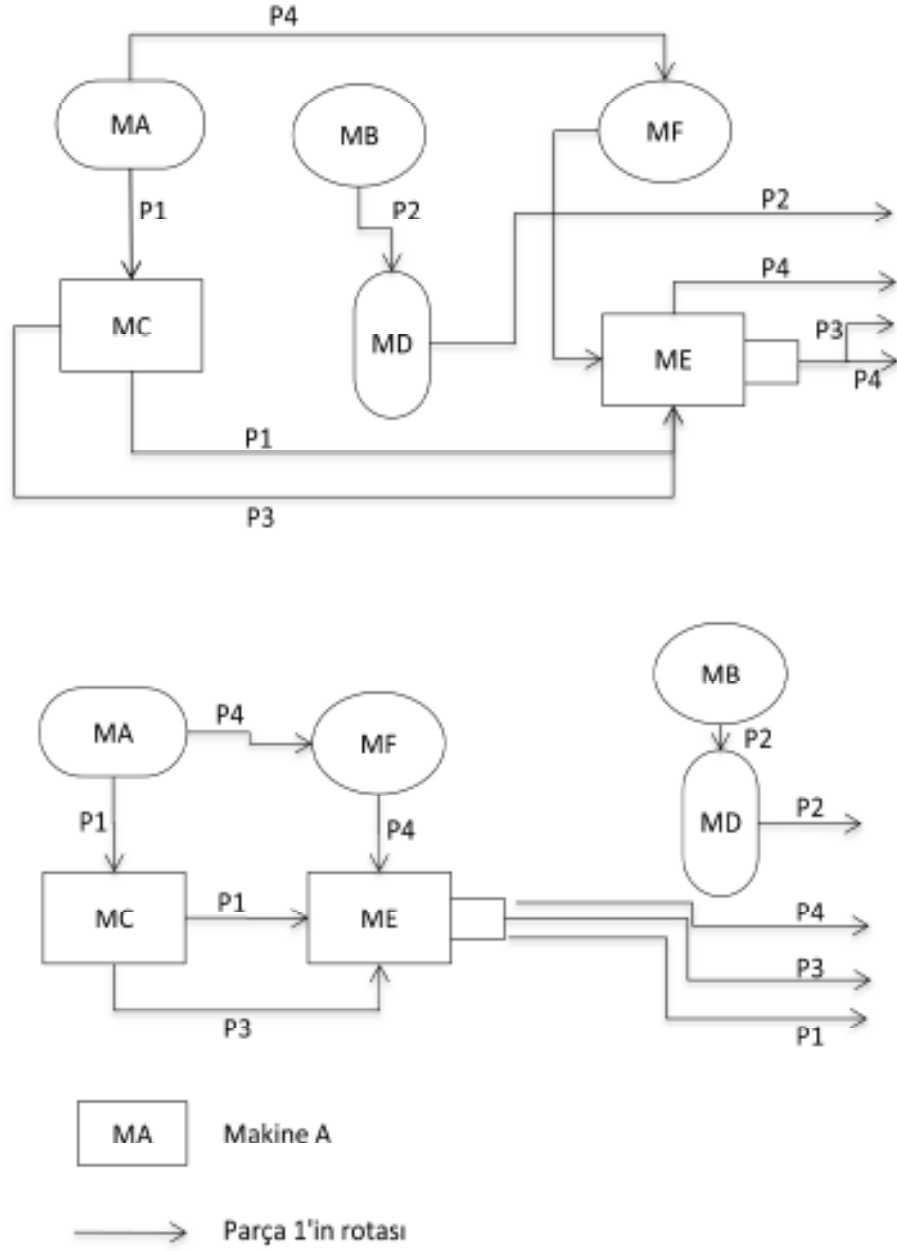


Şekil 5. Hücresel imalat sistemleri (Kamrahi ve Parsaei, 1994: 450-464).

Buradan da anlaşıldığı gibi parti halinde yapılan atölye tipi üretim sürekli üretime yaklaşılarak onun avantajlarından yararlanmaktadır. Mevcut üretimin hücresel imalat şeklinde düzenlenmesi Şekil 6’da verilmiştir.

Buna ilave olarak hücreler, takım çalışmasına yardım eden sosyolojik birimleri temsil eder (Wemmerlov ve Hyer, 1989: 1512). Bu birimler kendi içlerinde daha iyi tanıdıkları üretim çevresinde ürettikleri parçalara ait kalite problemlerinin çözülmesinde, kayıp zamanların azaltılmasında, prosesin iyileştirilmesinde etkin rol oynarlar. Bir anlamda özellikle 1990’larda yaygınlaşmaya başlayan yalın üretim tekniklerinin uygulanmasına olanak verirler.

Özetle hücresel imalatın uygulayıcılara sağladığı tipik avantajlar aşağıdaki gibidir (Kamrahi ve Parsaei, 1994: 450-464), (Heragu, 1994: 203-215) (Burbidge, 1992: 1209-1219):



Şekil 6. Makinelerin bir hücresel imalat sistemi içinde düzenlenmesi (Parsaei, Kolli, ve Hanley, 1997: 181-182).

- Daha az ayar yapma ve ayar zamanlarının azalması
- Üretim sürelerinin kısalması
- Daha fazla imalat esnekliği
- Proses içi stoklarda azalma ve düşük düzeyde stok bulundurma (jit)
- Malzeme taşıma maliyetlerinde azalma

- Ürün maliyetlerinde azalma
- Daha iyi sorumluluk
- Otomasyon
- Yüksek verimlilik
- Ekipman maliyeti ve direk endirek işçilik maliyetinde azalma
- Kalite iyileşmesi
- Makine beklemelerinin azalması ve makine kullanım oranı iyileşmesi
- Makineler etrafında daha az alan ve alan kullanımının iyileşmesi
- Çalışan memnuniyetinin iyileşmesi

2.4 Hücresel İmalatın uygulanması

Hücresel imalatta uygulamaya geçilirken en büyük sorun tüm üretim sistemini hücrelere bölmektir. Çözülmesi gereken diğer konular kavramsal olarak ikiye ayrılabilir. Birincisi makine ve ekipmanların yerleşimi, malzeme akışını ilgilendiren donanımsal konular, ikincisi ise çalışanların eğitimi ve şirket kültüründe yapılacak değişiklikler gibi yönetsel konulardır. Donanımsal konular tasarım ve yatırım sorunudur. Çünkü tüm atölye ortamı yeniden düzenlenecek, bazılarında modifikasyon yapılacak veya yerleri değiştirilecektir. Bazı uygulamalarda yeni makine ve ekipman temin edilmesi de gerekebilir veya gerekliliğin dışında böyle bir yönetsel karar verilebilir. Yönetsel konular ise daha fazla zaman ve üzerinde uğraş gerektiren konulardır. Operatörlerin yeniden eğitilmesi, operatörlere yeni işler verilmesi veya işlerin yeniden tanımlanması gerekebilir. Hücrelere atanmış olan operatörler ideal olarak hücredeki tüm işleri bilir olması gerekir (insanlı imalat hücrelerinde). Bu da operatörlerin önceki duruma göre çok işlevli olmaları demektir. Önceki durumda freze operatörü sadece kendi istasyonuna gelen parçaların freze operasyonundan sorumludur. Ancak hücre içindeki operatörler örneğin aynı anda hem freze hem de tornalama operasyonundan sorumlu olabilirler. Hatta çoğu durumda hücrede üretilen tüm parçaların kalitesinden sorumlu olabilirler. Diğer taraftan yönetim kademesinde bulunan yöneticilerin iş tanımlarında değişiklikler olması kaçınılmazdır. Hücreleri kendi kendine yönetilir olması istenen bir durumdur. Bu

durumda yöneticiler rutin olan yönetsel faaliyetlerden uzaklaşarak daha çok denetleyici ve destekleyici bir rol üstlenmelidirler.

Yukarıda belirtilen donanımsal ve yönetsel konular bir proje yönetimi disiplninde ele alınıp eşzamanlı olarak yürütülmelidir. Aynı zamanda bu iki bacadan biri eksik kalmamalıdır, çünkü hücelere bölünmüş bir atölye, hücelerel imalat yetkinliğine ulaşmamış çalışanlarla eski durumdan daha kötü performans değerlerine götürebilir.

Wemmerlov ve Hyer tarafından A.B.D. işletmelerinde yapılan istatistiksel bir çalışmada tespit edilen, hücelerel imalat uygulamasına geçişte karşılaşılan en büyük zorluklar aşağıda sıralanmıştır (Wemmerlov ve Hyer, 1989: 1512):

- 1) İnsan ilişkili
 - a) Operatör direnci
 - b) Ustabaşı/kısım amiri direnci
 - c) Yönetim direnci
 - d) Takım çalışması yeteneği eksikliği
- 2) Ekipman ilişkili
 - a) Makine arızaları
 - b) Özel takım/avadanlık gereksinimi
- 3) Operasyon ilişkili
 - a) Yükleme ve işgücü dengeleme
 - b) Kapasite dengesizliği
 - c) Hücre çizelgeleme
 - d) Basit akışlara ulaşma

Hücelerel imalat uygulayan işletmelerin edindiği tecrübelerle göre uygulamada dikkate alınması gereken en önemli faktörler aşağıdaki gibidir:

- 1) Hücre tasarımı
 - a) İmalat hücrelerinin planlanmasında grup teknolojisi analizi temeldir.
 - b) Ürün ömür çevrimi dikkate alınmalı.

- c) Hücre içinde karışık akışlardan kaçınılmalı.
- d) Mastarlama ihmal edilmemeli
- e) Hücre içindeki ekipman ve metotlar hızlı bir şekilde yenilenmeli.
- f) Yeni bir tesis tasarımı yapmak daha kolaydır.
- g) Hücrelerin yeniden tasarlanması için birden fazla iterasyon planlanmalı.

2) Çalışan katılımı

- a) Yönetim desteği ile hücre gelişimi aktivitelerine operatörler de katılmalı.
- b) Birim sorumluları/ustabaşları ve üretim çalışanlarının erken aşamalarda katılımı sağlanmalı.
- c) Hücrelerin planlama aşamasında tüm üretim alanları kapsama alınmalıdır.
- d) Çalışanların kabul etmesi küçümsenmemeli.

3) Eğitim

- a) Operatörler için eğitim sağlanmalı, böylelikle hücre içindeki tüm makineleri kullanabilirler.
- b) Hücre sorumlularının eğitimi hücrelere atanmadan önce tamamlanmalı.

4) Uygulama süreci

- a) Bir pilot hücreyle çalışmaya başlanmalı.
- b) Bunun için basit ve başarılı olunma potansiyeli yüksek olan hücre kullanılmalı.
- c) Uygulama yapılan hücre temin edilen ürün için tek kaynak olmamalı.
- d) İnsansız hücreler için en yetenekli çalışanlar seçilmeli.
- e) Şüphe olduğu durumlarda hücreyi devreye almayın.

2.5 Hücre Oluşturma Problemi

Hücresel imalat sistemlerinin tasarımında en önemli problem hücre oluşturma

problemidir. Hücresel imalat sisteminin temel prosedürü üretilecek parçaları özelliklerine göre parça ailelerine bölmek ve parça ailelerinin operasyon ihtiyaçlarına göre makineleri gruplayarak makine hücrelerini oluşturmaktır. Hücre formasyonu olarak tanımlanan bu problem daha genel şekilde Wu ve Salvendy tarafından şu şekilde verilmiştir: “Eğer makinelerin sayısı, tipi, kapasiteleri, üretilecek parçaların tipi, sayısı ve her bir parça için rota bilgileri, ihtiyaç duydukları makine standartları biliniyorsa, hangi makineler ve onlarla ilişkili olan parçalar hücreleri biçimlendirmek için birlikte gruplandırılmalıdır?” (Selim, Askin, ve Vakharia, 1998: 3).

Hücre oluşturma probleminde kullanılan metot ne olursa olsun gerçek üretim ortamında var olan kısıtları dikkate aldığı sürece başarılı olur. Hücre formasyon problemlerinde dikkate alınması gereken en önemli kısıtlar aşağıdaki gibidir (Heragu, 1994: 203-215), (Heragu ve Gupta, 1994: 125-140):

- Makine kapasiteleri: Hücre içinde gruplanan makinelerin kapasiteleri, kendilerine atanan parça ailelerinin taleplerinden büyük olmalıdır. Aksi halde hücreler arası hareketlerde artış meydana gelir.
- Operasyon sıraları
- Güvenlik ve teknolojik faktörler: Hücreler içindeki makineler, üretim ortamının güvenliğini tehlikeye atacak şekilde fiziksel olarak bir araya getirilmemelidir. Örneğin boya ve kaynak makinelerinin birbirine yakın olmaması gerektiği gibi.
- Makine hücre sayısı ve hücre büyüklüğünün üst sınırı: Bazı üretim sistemlerinde operatör kullanım oranının arttırılması için çok fonksiyonlu operatörler belli sayıda hücreye atanabilirler. Bu durum oluşturulacak hücre sayısını etkiler. Bunun gibi, hücre içine bir operatör atandığında hücredeki makine sayısına da sınırlama getirilebilir. Toplam atölye boyutları da oluşturulan hücrelerin boyutlarına sınırlama getirir.
- Hücre içi ve hücreler arası taşıma maliyetlerinin en küçüklenmesi: Daha önce de belirtildiği gibi, hücresel imalat tasarımında hücreler arası taşıma istenmeyen bir durumdur. Hücrelere giren parçalar, aynı hücrede tüm operasyonlarını tamamlayıp ürüne dönüşüp hücreyi terk etmesi ideal

durumdur. Ancak bu pratikte pek gerçekleşmez. Bazı parçalar diğer hücrelerdeki makineleri talep ederler. Dolayısıyla problem hücreler arası hareketlerin minimize edilmesidir. Bunun yanında hücre içi taşıma maliyetleri de başka bir problem olarak karşımıza çıkar. Bu problemin çözümünde de hücre içi makine yerleşim tasarımı önem kazanır. Optimum çözüme ulaşabilmek için hücre oluşturma ve yerleşim düzeni tasarımı eş zamanlı olarak ele alınmalıdır. Bu tezin konusu olan algoritma her iki problemi eş zamanlı olarak ele almaktadır.

- Makine kullanım oranı: Makine kullanım oranları da yapılan hücre oluşturulmasından sonra değerlendirilmesi gereken önemli bir faktördür.
- Makine ve ekipman maliyetleri: Yeni ekipman ihtiyaçları özellikle darboğaz durumlarında ortaya çıkar. Ancak böyle bir maliyet yerleşim düzenin yeniden düzenlenmesi ile örneğin darboğaz makinenin hücreler tarafından ortak kullanılmasıyla aşılabilir. Bunun gibi satın alınacak yardımcı ekipmanlar operasyon maliyetlerinin düşürülmesini sağlayabilir. Diğer yandan işletme ortamında hücresel imalata geçişte temel olarak iki genel yaklaşım mevcuttur. Ya mevcut makine ve ekipman hücreler halinde düzenlenecek, ya da yeni ekipman ve makineler temin edilecektir. Yeni ekipmanlarla düzenlenecek olan hücreler genelde esnek otomasyonun çeşitli şekillerini içerecek şekilde tasarlanırlar. Bu gibi hücreler tipik olarak insansızdır ve hücrelerin işletilmesinde insan rolü parçaların tezgaha yüklenmesi ve tezgahtan alınması, takım değiştirme, bakım ve muayene faaliyetleriyle sınırlandırılmıştır (Selim, Askin, ve Vakharia, 1998: 3). Son yıllardaki gelişmeler sonucunda muayene faaliyetlerinin çoğu da operatör inisiyatifinden çıkmıştır.
- Bağımsız hücrelerde iş çizelgelemesi: İlk olarak Vakharia hücresel imalat sistemi tasarımında çizelgelemenin dikkate alınması gerektiğini belirtmiştir. Hücrelerde sistemin tümüne nazaran daha az iş olması çizelgeleme problemlerinin optimal olarak çözülmesini mümkün hale getirmektedir.
- İşlem hacmi (kapasite) maksimizasyonu: İmalat hücrelerinin kapasite artışı çeşitli yollarla arttırılabilir. Örneği ayar için geliştirilecek bir aparat ayar

zamanını önemli ölçüde azaltabilir. Bunun sonucunda kapasite artışı sağlanabilir.

- Alternatif rotalar

Hücre oluşturmada en basit metot eye-balling olarak bilinen metottur. Bu metotta yeterli tecrübeye sahip bir operatör tarafından makineler uygun hücrelere atanırlar. Bu yöntem tamamen insan tecrübesine dayandığından ve makine sayısının artışına neden olduğundan verimsizdir. Diğer bir metot parça sınıflandırma ve kodlamadır. Tasarım ve imalat özelliklerine göre sınıflanan parçalar parça ailesi olarak gruplanırlar. Gruplanan bu ailelerin imalat özelliklerine göre makine hücreleri oluşturulur. Bu metodun devamında veya bağımsız olarak üretim akış analizi makine hücrelerine atama yapmak için kullanılabilir.

2.6 Hücre Oluşturma Yöntemlerinin Gelişimi

1960'lerden itibaren özellikle Burbidge'nin üretim akış analizini önermesinden sonra, hücre oluşturma ve parça gruplama için birçok model ve çözüm yaklaşımı geliştirilmiştir. Burbidge 1963'te yaptığı sınıflandırmada hücre oluşturma problemlerini 3'e ayırmıştır (Heragu , 1994: 203-215):

- 1) Pratik Kural (Rule of Thumbs Techniques),
- 2) Sınıflandırma ve Kodlama Teknikleri,
- 3) Üretim akış analizi.

Burbidge hücre ve parça ailelerinin birlikte oluşturulmasında parçaların tasarım, şekil veya diğer karakteristiklerinin dikkate alınmasının grup teknolojisinde kullanışlı olmadığını söylemiştir (Burbidge, 1992: 1209-1219).

King ve Nakornchai (1982) yaptıkları geniş kapsamlı gözden geçirme çalışmasında hücre biçimlendirme modellerini dört alt gruba ayırmıştır (Mansouri, Hussein, ve Newman, 2000: 1201-1218):

- 1) Benzerlik katsayısı: Bu sınıftaki metotlar iki makine arasındaki operasyonel benzerlikleri kullanarak, geliştirilen algoritmalarla hücre oluşturmaya ve/veya parça ailesi oluşturmaya çalışır.
- 2) Set teorik: Bu teknikler, bir şebeke kenarları boyunca bir yol olarak

temsil edilen makine ve parça süper kümeleri¹ yapılandırırılar.

- 3) Değerlendirmeli: Bu metotlar üretim akış analizi ile aynıdır.
- 4) Diğer analitik metotlar: Bunlar kümelendirme teknikleridir. Oluşturdukları algoritmalar vasıtasıyla, makine parça matrisindeki satır ve kolonları yeniden düzenleyerek blok diyagonal yapıyı elde etmeye çalışırlar.

Ballakur ve Steudel 1985’de yaptıkları sınıflandırmada parça gruplama ve makine grubu oluşturma yöntemlerini aşağıdaki gibi göstermişlerdir (Ballakur ve Steudel, 1987: 639-665):

- 1) Parça ailesi gruplama
 - a. Sınıflandırma ve kodlama
 - b. Küme analizi
- 2) Makine gruplama
 - a. Algoritmik prosedürler
 - i. Küme analizi
 - ii. Çizge kuramı
 - b. Algoritmik olmayan prosedürler
- 3) Makine-parça gruplama
 - a. Kombinatorial analiz
 - b. Algoritmik teknikler
 - i. Derece sıra kümelendirme
 - ii. Direkt kümelendirme algoritması
 - iii. Bağ enerji algoritması
 - iv. Askın ve Subramaniam yaklaşımı
- 4) Manuel teknikler
 - a. Üretim akış analizi

¹ Eğer A kümesi B kümesinin bir alt kümesi ise, bu durumda B kümesi A kümesinin bir süper kümesidir.

b. Komponent akış analizi

Mosier ve Taube (1985) literatürü dört başlığa ayırmıştır:

- 1) Parça tanımlama,
- 2) Gruplama,
- 3) GT atölyesinde çizelgeleme ve
- 4) GT uygulama.

Wemmerlov ve Hyer (1986) 70’den fazla makaleyi incelemiş ve bunları 4 grupta kategorize etmiştir (Mansouri, Hussein, ve Newman, 2000: 1201-1218):

- 1) Parça özelliklerine göre yapılan gruplandırma
- 2) Parça rotalarına göre yapılan gruplandırma
 - a. Önce makine gruplarını daha sonra parça ailelerini tanımlayan yaklaşımlar,
 - b. Önce parça ailelerini sonra makine gruplarını tanımlayan yaklaşımlar,
 - c. Parça ailelerini ve makine gruplarını eş zamanlı olarak tanımlayan yaklaşımlar.

Han ve Ham 1986’da yayınladıkları “Parça ailesi oluşturma için çok amaçlı kümelendirme analizi” isimli makalede grup teknolojisi algoritmalarını 4 bölüme ayırmışlardır (Heragu, 1994: 203-215):

- 1) Peripatetik ve oküler teknikler: Bu teknikler daha önce açıklanan rule of thumb ve eye balling yöntemleriyle aynıdır.
- 2) Üretim akış analizi teknikleri
- 3) Sınıflandırma ve kodlama teknikleri
- 4) Matematik programlama teknikleri: Bu teknikler bulanık matematik, örüntü tanımlama, küme analizi gibi teknikleri kullanırlar.

Vakharia 1986 yılında yayınladığı “Grup teknolojisinde hücre oluşturma metotları” isimli makalesinde 4 başlık altında sınıflandırma yapmıştır (Heragu, 1994: 203-215):

- 1) Tanımlayıcı prosedürler: Üretim akış analizi ve parça akış analizi tekniklerinden oluşmaktadır. Her iki yöntem de temelde bir çok benzerlikler içermektedir.
- 2) Blok diyagonal teknikler: Bu teknikler yukarıda verilen King ve Nakornchai'nin sınıflandırdığı “diğer analitik teknikler” içinde yer almaktadır.
- 3) Benzerlik katsayısı teknikleri
- 4) Diğer analitik teknikler: Bunların arasında yukarıda geçen set teorik metodu vardır.

Wemmerlov ve Hyer 1986'da yayınladıkları “Hücre sel imalatta parça ailesi/makine grubu tanımlama problemi için prosedürler” isimli makalesinde sınıflandırma çalışmaları üzerinde kapsamlı bir analiz yapmışlardır. Hücre oluşturma sürecini anlamamızda diğer birçok metodun bize yardımcı olmadığını göstermişlerdir. Bu nedenle grup teknolojisi tekniklerini aşağıdaki gibi sınıflandırmışlardır (Heragu, 1994: 203-215):

- 1) Makine rotalarının yardımı olmadan parça ailesi tanımlayan teknikler,
- 2) Rotaları kullanarak parça ailesi tanımlayan teknikler,
- 3) Sadece makine gruplarını tanımlayan teknikler,
- 4) Parça aileleri ve makine gruplarını eş zamanlı olarak tanımlayan teknikler.

Kusiak ve Chow 1988'de yayınladıkları “Üretim sistemlerinin ayrışması” isimli makalede grup teknolojisini aşağıdaki şekilde sınıflandırmışlardır (Heragu, 1994: 203-215):

- 1) Üretim akış analizi teknikleri (PFA)
- 2) Benzerlik katsayısı teknikleri (Similarity Coefficient)
- 3) Sıralama tabanlı teknikler (Sorting based)
- 4) Bağ-enerji teknikleri (Bond energy)
- 5) Maliyet tabanlı teknikler (Cost based)
- 6) Küme tanımlama teknikleri (Cluster identification)

Chu (1989) hücresel imalat için geniş kapsamlı bir kaynakça sağlamış ve literatürü, 2 ana başlık altında toplamıştır (Mansouri, Husseini, ve Newman, 2000: 1201-1218):

- A. Tasarıma yönelik yaklaşımlar,
- B. Üretime yönelik yaklaşımlar,
 - a. Dizi tabanlı (array based)
 - b. Hiyerarşik (hierarchical)
 - c. Hiyerarşik olmayan (non-hierarchical)
 - d. Matematiksel (mathematical)
 - e. Çizge kuramı (graph theoretic)
 - f. Sezgisel (heuristic) yaklaşımlar

1990'lar itibarıyla tek kriter dikkate alan yaklaşımlar Tablo 2'de özetlenmiştir (Mansouri, Husseini, ve Newman, 2000: 1201-1218).

Üretim akış analizi içindeki modeller 1991'e kadar çözüm yaklaşımı, karar değişkenleri, amaç ve kısıtlar temelinde karşılaştırılmıştır.

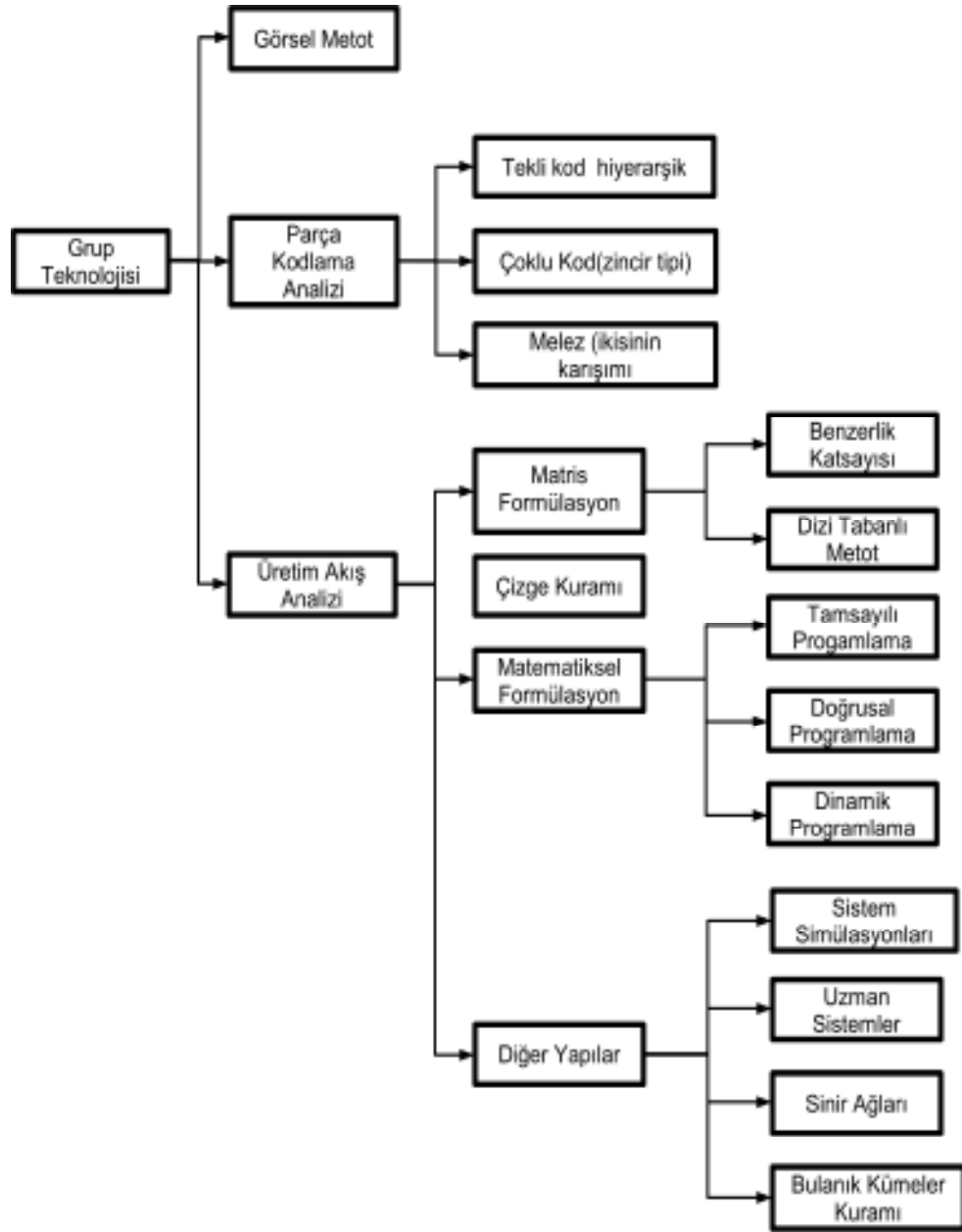
Offodile, Mehrez ve Grznar 1994 yılında yayınladıkları "Grup teknolojisi için sınıflandırılmalı gözden geçirme çerçevesi" isimli makalelerinde detaylı bir sınıflandırma yapmışlardır (Şekil 7).

H. M. Selim ve diğerleri (1998) hücre oluşturma tekniklerini 5 ana başlık altında toplamıştır (Şekil 8) (Selim, Askin, ve Vakharia, 1998: 3). Tanımlayıcı prosedürler kendi aralarında 3'e ayrılır. Bunlardan parça aileleri tanımlama prosedürü (PFI: part family identification) hücre oluşturmaya parça ailelerini tanımlayarak başlar, sonra makineleri parça ailelerine atar. Makine grupları tanımlama (MGI: machine groups identification) önceki metodun adımlarını tersten izler. Diğer metot ise parça aileleri/makine gruplama prosedürüdür. Bu gruptaki prosedürler parça aileleri ve makine gruplarını eş zamanlı olarak tanımlar. Parça ailesi tanımlama prosedürleri ayrıca formal (rule of thumb kuralları, görsel değerlendirme ve diğer kriterler) ve informal (kodlama ve sınıflandırma sistemleri)

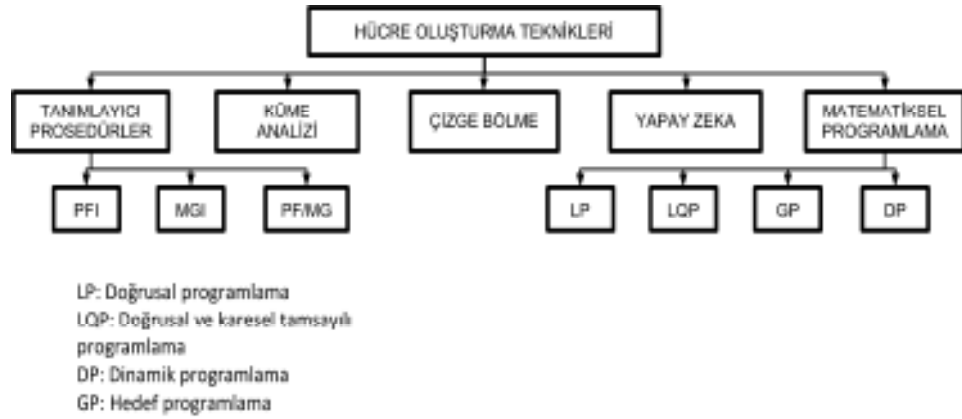
Tablo 2. 1990’larda hücre tasarımı içi tek kriterli bazı modeller (Mansouri, Hussein, ve Newman, 2000: 1201-1218).

Kriter	Referanslar
Hücreler arası hareket minimizasyonu	Klincewics ve Rajan (1994), Joines <i>et al.</i> (1996), Sofianopoulou (1997), Cheng <i>et al.</i> (1996)
Parça ve/veya makine benzerliklerinin (benzemezliliklerinin minimizasyonu) maksimizasyonu	Kao and Moon (1991), Kaparthi and Suresh (1991), Chu (1993), Boctor (1991), Chen ve Srivasata (1994), Askin <i>et al.</i> (1991), Lee ve Carcia-Diaz (1993), Kusiak <i>et al.</i> (1993)
Parça makine matrisinde blok diyagonal biçimin elde edilmesi	Kaparthi and Suresh (1992), Kaparthi <i>et al.</i> (1993), Suresh ve Kaparthi (1994), Mukhopadhyay <i>et al.</i> (1994), Chen ve Irani (1993)
Hücre yük dengesizliklerinin minimizasyonu	Venugopal ve Narendran (1992 b)
İstisna elemanların sayısının minimizasyonu	Song and Hitomi (1992), Amirahmadi ve Choobineh (1996)

olarak alt gruplara ayrılabilir. Literatürün çoğu parça ailelerini tanımlamak için informal yaklaşımları tartışır. Makine gruplama prosedürleri hücre oluşturma problemini iki aşamalı bir süreç olarak ele alır. Birinci aşamada makineler parça rota bilgisinde bulunan bilgiler temelinde gruplanırlar. İkinci aşamada parçalar makine gruplarına atanırlar. B. R. Sarker ve Y. Xu 1998 yılında yaptıkları sınıflandırmada operasyon sıra verisine dayanan metotları matematik programlama, ağ analizi, malzeme akış analizi metodu ve sezgisel olarak 4’e ayırmışlardır (Sarker ve Xu, 1998: 771-783).

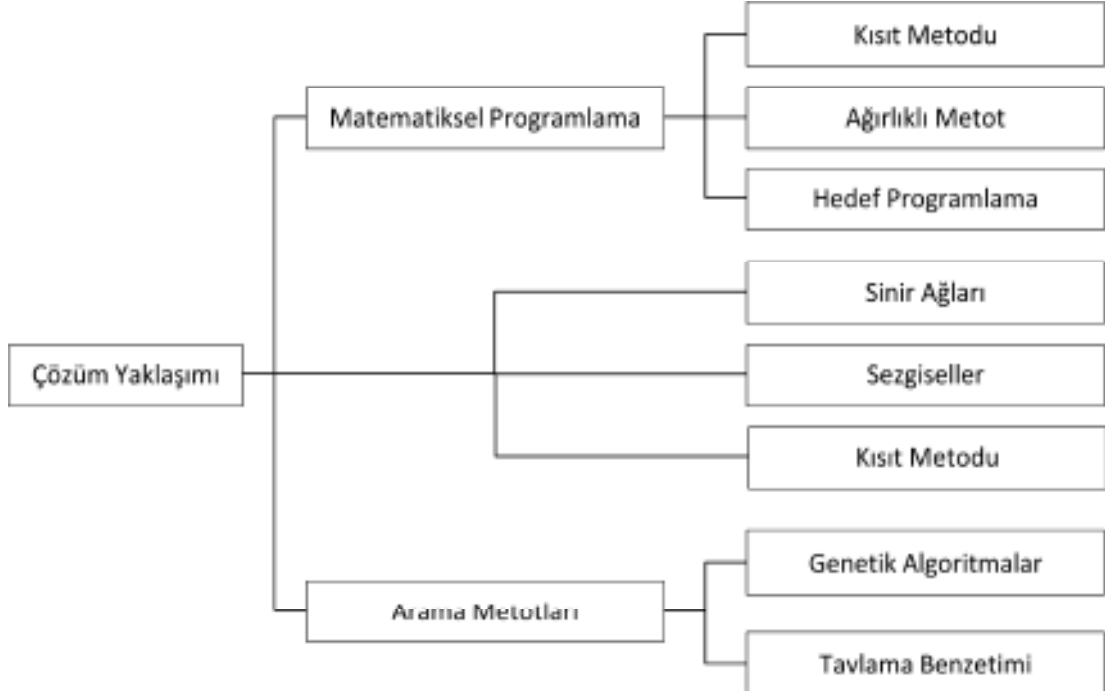


Şekil 7. Grup teknolojisi için sınıflandırılmalı gözden geçirme çerçevesi (Reisman, Kumar, Motwani, ve Cheng, 1997: 508-520).



Şekil 8. Hücre oluşturma metotlarının sınıflandırması (Selim, Askin, ve Vakharia, 1998: 3).

Yukarıda alıntılar yapıldığı Mansouri ve diğerleri'nin yapmış olduğu önemli bir sınıflandırma da çözüm yaklaşımı temelli sınıflandırmadır. Özellikle tezin 3. bölümünde yer alan operasyon sıra verisi temelli metotların yer aldığı sınıflandırmayı burada belirtmekte yarar var (Şekil 9).



Şekil 9. Çözüm yaklaşımlarının sınıflandırılması (Mansouri, Hussein, ve Newman, 2000: 1201-1218).

Abraham P. George ve diğerleri (2003) yaptıkları sınıflandırmada problem

girdisi olan verilere göre bir sınıflandırma yaparak ikili ve ikili olmayan veri kullanan metotlar olarak ayırım yapmıştır (George, Rajendran, ve Ghosh, 2003: 125-133). İkili veriyi kullanan metotlar: hiyerarşik, matematik, çizelge teorik, derece sıra kümelenendirme (rank order clustering), küme tanımlama algoritması (cluster identification algorithm), hiyerarşik olmayan kümelenendirme, boolean matrisi, ana komponent analizi (principle component analysis), hücre benzerlik algoritması (cellular similarity algorithm). İkili olmayan veriyi ise sıra-derece (ordinal-level data) ve oran-sıra (ratio level data) verisi olarak ikiye ayırmışlardır. Bunlardan sıra derece verisi operasyon sıra bilgisini kullanır. Makine parça matrisindeki 1'lerin yerini parçanın ilgili makinedeki operasyonun sırası yer alır. Oran sıra verisi ise iş yükü bilgisini kullanır ve buradan modifiye edilmiş makine parça birim matrisini elde eder.

M. Boulif ve K. Atif 2003 yılında genetik algoritma ile hücre oluşturmaya dalsınır tekniği ile iyileştirdikleri çalışmalarında, hücre oluşturma çalışmalarını üç ana grupta toplamışlardır (Boulif ve Atif, 2006: 2219-2245):

1. Parça-makine matrisi temelli metotlar: Makine-parça matrisi de olarak bilinen matris hangi makinede hangi parçanın işlendiği bilgisini gösterir. Genel olarak bu gruptaki metotlar satır ve kolonların yerlerini değiştirerek blok diyagonal yapıyı oluşturmaya çalışarak hücre oluştururlar. Özellikle erken dönemdeki çalışmalar bu grup içindedir. Bu metotlar operasyon sırası veya üretim hacmi gibi bilgileri dikkate almazlar.

2. Benzerlik katsayısı temelli metotlar: Bu metotlar makine parça matrisindeki bilgileri kullanarak makineler arasındaki benzerlik ilişkisini matematiksel bir katsayıya çevirir ve bu benzerliği kullanarak makineleri hücrelere gruplar. Son yıllarda yapılan çalışmalarda bu benzerlik katsayılarına operasyon sırası, üretim hacmi, operasyon zamanı gibi bilgiler de dahil edilmiştir.

3. Meta-sezgisel temelli metotlar: Makine hücresi oluşturma probleminin NP-tam olarak tanımlanmasıyla birlikte, araştırmacılar sezgisel metotlara yönelmişlerdir. Meta-sezgisel olarak adlandırılan genel amaçlı sezgiseller arasında yasak arama (tabu search), tavlama benzetimi (simulated annealing), sinir ağları yaklaşımı (neural network approach), genetik algoritma (genetic algorithm) sayılabilir. Son yapılan

alışmalar, bu disiplinlerin bir veya daha fazlasını birleřtirerek oluřturdukları melez algoritmalarla daha iyi sonular almaktadırlar.

2.7 Hcre Oluřturma Problemlerinde Operasyon Sıra Verisi Kullanan Modeller

zellikle 1990’ların sonundan itibaren yapılan hcre oluřturma algoritmaları giderek daha fazla operasyon sıra verisi kullanmaya bařlamıřsa da, 1990’ların bařından itibaren ilk temeller atılmaya bařlamıřtır. Sarker ve Xi 1998 yılında yayınladıkları makalede operasyon sıra verisine dayanan kmelendirme metodlarını 4 blme ayırmıřlardır (Sarker ve Xu, 1998: 771-783): Matematik programlama, aē analizi metodu, malzeme akıř analizi metodu ve sezgisel metod.

2.7.1 Matematiksel Programlama

Hcre oluřturma problemini deēerlendirme ltlerinin eřitli hedefleriyle birlikte uygun matematik programlama tekniklerini kullanarak modelleyen bir yaklařımdır. Buradaki deēerlendirme ltlerine benzerlik katsayısının en byklenmesi, maliyetlerin en kklenmesi, akıř mesafelerinin en kklenmesi rnek olarak verilebilir. Choobineh’in 1988 yılında yayınladıēı “Hcresel imalat tasarımı iin bir ereve” isimli makalesinde hcre oluřturma iin iki prosedrl bir metod nermiřtir. Metodun ilk ařamasında operasyon benzerlik katsayıları hesaplanır. İkinci ařmada hcre sayısı, her bir hcredeki makine tipi ve sayısını, malzeme tařıma maliyetlerinin en kklenmesini dikkate alan bir tamsayılı programlama nerir. Hcre oluřturma iin Aktrk (1996) ve Aktrk ve Balkse (1996) ok amalı matematiksel model nermiřlerdir. Bu model hem imalat zelliklerini hem de operasyon sıralarını dikkate almıřtır ve operasyon sırasına dayanan bir benzerlik katsayısı nermiřtir. Bu metod, hcreleri oluřturduēu gibi, eř zamanlı olarak her bir hcredeki makinelerin yerleřimini de yapabilmektedir. Bunun dıřında 1998 yılında Broering, Leep, Parsaei ve Wong makine arızası durumunda alternatif rotaları dikkate alan bir benzerlik katsayısına dayanan para ailesi oluřturma algoritması geliřtirdiler. 2006 yılında Defersha ve Chen operasyon sıra verisini kullanan kapsamlı bir matematik model nerdiler (Mahdavi, Shirazi, ve Paydar, 2008: 943-953).

2.7.2 Ağ Analizi Metodu

Ağ analizi metodunun temeli hücre oluşturma modelini yönlü ve yönsüz grafik veya ağ olarak modellemektir ve ağ optimizasyon tekniklerini kullanarak hücreler arası veya hücre içi akışları minimize etmektir.

Hücre oluşturma probleminin ilk adımında yapılması gereken temel çalışmalardan birinin makine ve parçalar arasındaki ilişkilerin incelenmesi olmasından yola çıkarak araştırmacılar çizge teorisi kullanılması olasılığını araştırmışlar ve çizge teorisini hücre oluşturma problemlerine başarılı bir şekilde adapte etmişlerdir. Çizge teorisine göre hücre oluşturma probleminde makineler kümesi çizgenin düğümleri ve parça rotalarından kaynaklanan makineler arasındaki ilişkiler ise düğümleri birbirine bağlayan kenarlar şeklinde temsil edilebilir. Çizge teorisi ve uygulamaları ile ilgili daha ayrıntılı bilgi Harary (1968), Busacker ve Saaty (1965) ve Deo (1973)'de bulunabilir (Rajagopalan ve Batra, 1975: 567-579).

Wu ve Salvedy (1993) hücre oluşturma problemi için operasyon sıra verisini kullanan bir yönsüz grafik modeli önermiştir. Tam'ın 1988 yılında geliştirdiği operasyon sıra tabanlı benzerlik katsayısını kullanarak Kiang 1995 yılında makine hücrelerini belirlemek için sinir ağları tabanlı bir algoritma geliştirmiştir. 2001 yılında S-D Lee ve C-P Chiang, iki yönlü doğrusal malzeme akışlı yerleşimde hücreler arası akış maliyetini çok uçlu şebeke akış modeli kullanarak en küçükmeye çalışmıştır (Mahdavi, Shirazi, ve Paydar, 2008: 943-953). Boulif ve Atıf 2006 yılında çizge bölme temeline dayanan dal-sınır metoduyla iyileştirilmiş genetik algoritmayı açıklamıştır. Algoritmalarında operasyon sırası, maksimum hücre sayısı, maksimum hücre büyüklüğü, makinelerin ortaklığı gibi kısıtları dikkate almışlardır.

2.7.3 Malzeme Akış Analizi Metodu

Bu yaklaşım hücre oluşturma problemini hücre içi malzeme akışlarının etkisini dikkate alarak modelleme yapar. Çeşitli zorluklar nedeniyle çoğu malzeme akış analizi metodu sezgisel yaklaşımları temel alır. Vakharia ve Wemmerlov (1990) kümelenendirme tekniği kullanan ve operasyon sıra verisine dayanan 4 aşamalı bir akış yaklaşımı sunmuşlardır. Bu yaklaşımda parça aileleri operasyon benzerlik katsayısı

kullanarak oluşturulur ve makine yükleri dikkate alınarak makineler parça ailelerine atanır.

2.7.4 Sezgisel Metot

Hücre oluşturma probleminin karmaşıklığı nedeniyle araştırmacılar birçok etkili sezgisel metot geliştirmişlerdir. Ancak sezgisel yaklaşımlardan elde edilen sonuçlar optimum değildir. Diğer taraftan sezgisel yaklaşım makul bir hesaplama süresi içinde kabul edilebilir bir çözüm sağlayabilir. Bu nedenle, bu tür metotlar araştırmacılar tarafından hücre oluşturma problemlerini çözmek için kullanılmıştır. Selvam ve Balasubramanian (1985) malzeme taşımayı ve makine atıl kapasitesini minimize etmek için, yaklaşık olarak aynı operasyon sırasına sahip parçaları gruplayan sezgisel bir algoritma geliştirmişlerdir. Bu çalışmada operasyon benzerlik katsayısı matrisi ve ilgili hesaplama metodu sunulmuş ve benzerlik katsayısı matrisine göre parçalar gruplanmıştır.

Tam (1993) Levenshtein'in iki sıra arasındaki yakınlığı bulan yaklaşımını kullanarak operasyon sırasına dayanan yeni bir benzerlik katsayısı geliştirmiştir. Daha sonra parçalar k en yakın komşu kümeleme metodu kullanılarak gruplanır. Tam ayrıca parça, operasyon ve makine arasındaki ilişkiyi tanımlamış ve hücre oluşturma probleminde operasyon sırasının önemini vurgulamıştır.

Logendran (1991) hücre oluşturma probleminde operasyon sıralarının ve hücrelerin yerleşim düzeninin etkisini göstermiştir. Bunun için operasyon sıralarıyla hücre içi ve hücreler arası hareketleri değerlendirerek hücreleri oluşturan bir model geliştirmiştir. Ayrıca hücrelerin yerleşim düzenini dikkate alarak hücreler arası hareketleri değerlendirmiştir. Bu modelde makine kullanımı dikkate alınmış ve modelin çözümü için bir sezgisel algoritma geliştirilmiştir.

Heragu ve Kakuturi (1997), makine gruplama ve yerleşim düzeni problemlerini entegre etmeye çalıştılar. Bu metotta ilk önce bir sezgisel metotla makine hücreleri oluşturulur. Daha sonra optimale yakın hücre içi ve hücreler arası yerleşim düzenleri melez tavlama benzetimi² temelli algoritma ile oluşturulur.

² Optimizasyon problemlerinin çözümü için katı cisimlerin özellikle metallerin soğuma esnasında atomik dizilişlerini temel alan sezgisel bir arama yöntemidir.

Lee ve Chiang (2001), makine hücrelerinin iki yönlü doğrusal akış yerleşimi boyunca konumlandırıldığı ortak kümeleme-yerleşim düzeni problemini ele aldılar. Bu algoritma hücreler arası hareketleri minimize etmek yerine hücreler arası akış maliyetini minimize etmeye çalışmışlardır.

Chiang ve Lee (2004), iki yönlü doğrusal akış yerleşim düzeni için optimal bölme yaklaşımli bir genetik tabanlı algoritma geliştirdiler. Öncekine benzer şekilde bu algoritma hücreler arası akış maliyetini minimize etmeye çalışır.

Housyar ve McGinnis tesisleri yerleştirmek için grafik yaklaşımli bir sezgisel algoritma uygulamıştır. Bu yaklaşımda proses içi stokların hareket mesafelerinin minimize edilmesi amaçlanmıştır.

ÜÇÜNCÜ BÖLÜM: HÜCRE OLUŞTURMA METOTLARI

3 Giriş

Önceki bölümde incelendiği gibi literatürde araştırmacıların ortaya koyduğu birçok hücre oluşturma metodu mevcuttur. Bu tezin içeriğinde operasyon sırasına dayanan bir algoritma inceleneceğinden hücre oluşturma metotlarını bu bölümde, problemin girdisi olarak kullanılan parça-makine matrisi temelinde inceleyeceğiz.

Hücre oluşturma problemi girdisi olan makine-parça matrisi temelinde bakıldığında ilgili metotlar genel olarak ikili makine-parça matrisi ile makine hücre oluşturma ve sıra verisi ile hücre oluşturma olarak 2'ye ayrılır (Mahdavi, Shirazi, ve Paydar, 2008: 943-953).

3.1 İkili Makine-Parça Matrisi

Üretim akış analizini kullanan kümelenendirme metotlarında makinelerin rota bilgileri kullanılarak 0-1 ikili makine parça matrisi oluşturulduktan sonra çeşitli yöntemlerle gruplandırma yapmak mümkündür. Matriste i . satır ve j . kolonda yer alan 1 bilgisi, j . parçanın i . makinede işlendiğini temsil eder (Kusiak ve Chow, 1988: 457-471).

Makine-parça matrisi, $a_{ij}=1$, eğer i . parça j . makinede işlem görüyorsa, 0, aksi durumda.

İkili makine parça birim matrisi kullanılarak yapılan gruplandırmada amaç, çeşitli yöntemler kullanarak matriste bulunan 1'leri makine ve parça sıralarını yeniden düzenleyerek, matrisin diyagonalinde toplamaktır. Matris diyagonalinde toplanan 1 bilgisi makine ve parça gruplarını tanımlar. Ancak tüm gruplama problemlerinde 1 bilgisi diyagonalde toplanamayabilir. Bu nedenle gruplama problemlerinin asıl amacı gruplama sonucunda ortaya çıkan diyagonal dışında kalan 1 bilgisinin minimize edilmesidir. Ortadan kaldırılamayan bu 1 bilgisi istisnai eleman olarak adlandırılır ve sonuç olarak bu operasyon ya atölye dışında yan sanayide yaptırılır veya ilgili makine duplike edilerek oluşturulan hücrelere dahil edilir. İzleyen bölümlerde ikili makine parça matrisini kullanan hücre oluşturma metotları incelenmiştir.

3.1.1 Hiyerarşik Metotlar

Hiyerarşik metotlar kümeleme problemlerinde sıklıkla kullanılan metotlardır. Bu metotlar ilk olarak her bir makineyi (veya parçayı) küme olarak alırlar ve daha sonraki süreçte bütün elemanla tek bir küme haline gelene kadar grupları kümelandirirler. Çoğu hiyerarşik metotlar kümeleri oluşturmak için benzerlik katsayılarını hesaplar ve kullanır (Srinivasan ve Narendran, 1991: 463-478). Bu metot “dendogram” denilen ve verilen veri veya bilgi setini benzerlikleri temelinde küçük kümeler halinde gruplayarak grafik bir sonuç verir (Kamrahi ve Parsaei, 1994: 450-464). Bu küçük kümeler daha sonra büyük kümeler halinde derecelendirilir ve gruplanır. Hiyerarşik metotlar iki formda yürütülür: toplayan (agglomerative) ve bölen (divisive). Toplayan formlar kümeleri, objeleri sıralı olarak toplayarak oluşturur. Bölen formlar ise bilgi veya veri setlerini daha fazla bölünemeyene kadar ardışık olarak kümelere böler.

3.1.1.1 Tek Bağlantılı Kümelandirme Algoritması

Tek bağlantılı küme analizi metodu Sneath tarafından 1968 yılında geliştirilmiş ve McAuley tarafından sayısal sınıflandırma alanında grup analizi problemine uygulanmıştır (King, 1980). Metot, ikili makine parça matrisi bilgilerini kullanarak hesaplanmış benzerlik katsayılarına dayanan makine gruplama hiyerarşik sürecini içerir. Mc Auley makineler arasındaki benzerlik katsayılarını hesaplamak için Jaccard’ın benzerlik katsayısını kullanmıştır (Won ve Kim, 1997: 207-220). Bir çok benzerlik katsayısı arasından Jaccard benzerlik katsayısı literatürde en çok kullanılan benzerlik katsayısıdır (Yin ve Yasuda, 2005: 471-489).

$$S_{hi} = \frac{N_{hi}}{N_h + N_i - N_{hi}} \quad (1)$$

Burada; S_{hi} : h ve i makineleri arasındaki benzerlik katsayısı; $h, i=1, \dots, m$,

N_h, N_i : sırasıyla h ve i makineleri tarafından işlem gören parça sayısı,

N_{hi} : h ve i makinelerinin ikisinde de işlem gören parça sayısı.

Metodun adımları aşağıdaki gibidir:

Adım 1: Makineler arasındaki benzerlik katsayısını hesapla,

Adım 2: En büyük benzerliğe sahip iki makine ilk makine hücresini oluşturmak için birleştirilir.

Adım 3: Eşik değerini bir sonraki değere düşür ve eşik değerinden daha büyük benzerlik katsayılı tüm makine/makine hücreleri daha büyük gruplar halinde gruplanır.

Adım 4: Tüm makineler tek bir makine hücresi içinde gruplanana kadar adım 3 sürer.

Makine benzerlikleri eşik değerlerinde bağlanan kümeler dendogram üzerinde gösterilir. Dendogramın sol tarafındaki skalada benzerlik katsayıları, üst kısımda yer alan yatay bölümde makine numaraları yer alır. Aşağıdaki örnekte metodun işleyişini görebiliriz (King, 1980: 213-232). Tablo 3 parça rota bilgilerinden elde edilen makine-parça matrisidir. Tek bağlantılı kümelendirme metodunun sunduğu çözümlerin en büyük dezavantajı zincirleme problemidir. Zincirleme düşük benzerlik katsayısına sahip parça ve makinelerin gruplanmasıdır.

Tablo 4 ise makine çiftleri arasındaki hesaplanmış benzerlik katsayılarını ifade etmektedir. (1) eşitliğinden benzerlik katsayısı hesaplama örneği aşağıda verilmiştir:

1 ve 2 nolu makineler için benzerlik katsayısı: $1/(2+2)-1=1/3=0,33$.

Şekil 10'da ise benzerlik katsayıları matrisi kullanılarak elde edilen dendogram görülmektedir. İlk olarak en büyük benzerlik katsayısı olan 0,67 eşik değeri olarak seçilmiş ve makine 1,4 birleştirilerek bir küme elde edilmiştir. Daha sonra aynı eşik değerinde makine 2,4 ve 3,5 birleştirilerek kümelendirilmiştir. Eşik değeri bundan sonra 0,33'e düşürülmüş ve 1 ve 2 nolu makinelerin olduğu gruplar birleştirilmiştir.

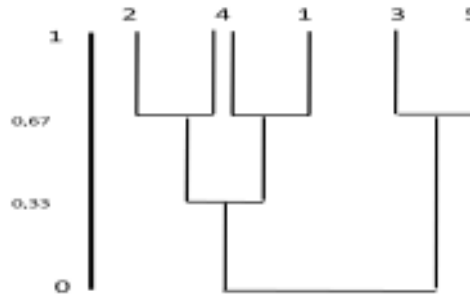
Tablo 3. Makine parça matrisi (King, 1980: 213-232).

		Parçalar					
		1	2	3	4	5	6
Makineler	1			1		1	
	2		1	1			
	3	1			1	1	
	4		1	1		1	
	5	1	1		1		1

Tek bağlantılı kümелendirme metodunun sunduğu çözümlerin en büyük dezavantajı zincirleme problemidir. Zincirleme düşük benzerlik katsayısına sahip parça ve makinelerin gruplanmasıdır.

Tablo 4. Benzerlik katsayıları matrisi.

	Makineler				
	1	2	3	4	5
1		0,33	0	0,67	0
2			0	0,67	0
3				0	0,67
4					0
5					



Şekil 10. Benzerlik katsayılarına göre oluşturulmuş dendrogram (King, 1980: 213-232).

3.1.1.2 Ortalama Bağlantılı Kümелendirme

Ortalama bağlantılı kümелendirmenin en büyük avantajı zincirlemenin üstesinden gelmesidir (Kusiak ve Chow, Decomposition of manufacturing systems, 1988: 457-471). Metodun adımları aşağıdaki gibidir (Yin ve Yasuda, 2005: 471-489):

Adım 1: Makine çiftleri arasındaki benzerlik katsayılarını hesapla ve makine benzerlik katsayısı matrisini oluştur.

Adım 2: En büyük benzerlik katsayısını eşik değeri olarak seç.

Adım 3: Eşik değerine kadar olan makineleri gruplandır. Makineler arasındaki benzerlik katsayılarının ortalamalarını alarak yeni benzerlik katsayısı matrisini oluştur.

Adım 4: Eşik değerini bir sonraki değere düşür.

Adım 5: Eşik değeri daha fazla düşürülemeyene kadar adım 3'e dön.

Aşağıda Tablo 5'de ortalama bağlantılı kümelenendirme algoritması için 11 ayrı makinede işlem gören 22 parça için sayısal bir örnek verilmiştir.

Adım 1'e göre öncelikle makine çiftleri arasındaki benzerlik katsayıları hesaplanarak benzerlik katsayıları matrisi oluşturulur (Tablo 6).

Tablo 5. Ortalama bağlantılı kümelenendirme algoritması için örnek makine parça matrisi.

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22
A	1	1	1								1				1	1				1	1	1
B					1						1	1										
C					1	1						1	1						1			
D	1	1	1								1				1	1				1	1	1
E	1	1	1				1								1	1				1	1	1
F					1			1				1					1	1	1			
G				1		1			1					1			1	1	1			
H					1	1			1	1		1	1	1			1	1				
I				1					1					1			1	1	1			
K	1	1	1		1		1	1			1	1			1	1		1				
L				1						1							1	1				

En büyük eşik değeri olan “1” değerine göre (A,D), (B,F), (G,I) makineleri gruplanır. Şekil 11'deki dendogramda yapılan gruplama görülebilir. 1. İterasyon sonucunda yapılan gruplamaya göre benzerlik katsayıları matrisi, makinelerin çiftli olarak benzerlik katsayılarının ortalamaları alınarak güncellenir (Tablo 7).

2. İterasyon sonucunda elde edilen benzerlik katsayıları matrisi Tablo 8'dedir.

0,5 eşik değerine göre oluşturulan benzerlik katsayıları matrisi Tablo 9'da görülebilir.

Tablo 6. Ortalama bağlantılı kümelendirme algoritması örneği için benzerlik katsayıları matrisi

	A	B	C	D	E	F	G	H	I	K	L
A	-	0,09	0	1	0,8	0	0	0	0	0,43	0
B		-	0,8	0,09	0	0,285	0	0,2	0	0,27	0
C			-	0	0	0,38	0,2	0,4	0,1	0,14	0
D				-	0,8	0	0	0	0	0,43	0
E					-	0	0	0	0	0,18	0
F						-	0,25	0,36	0	0,31	0
G							-	0,45	1	0	0,38
H								-	0,36	0,18	0,3
I									-	0,06	0,43
K										-	0,07
L											-

Tablo 7. "1" Eşik değerine göre 1. iterasyon sonucu benzerlik katsayıları matrisi

	(A,D)	B	C	E	F	(G,I)	H	K	L
(A,D)	-	0,09	0	0,8	0	1	0	0,43	0
B		-	0,8	0	0,285	0	0,2	0,27	0
C			-	0	0,38	0,15	0,4	0,14	0
E				-	0	0	0	0,18	0
F					-	0,125	0,26	0,31	0
(G,I)						-	0,4	0,03	0,405
H							-	0,18	0,3
K								-	0,07
L									-

Tablo 8. "0,8" Eşik değerine göre 2. iterasyon sonucu benzerlik katsayıları matrisi

	(A,D,E)	(B,C)	F	(G,I)	H	K	L
(A,D,E)	-	0,8	0	0,5	0	0,43	0,31
(B,C)		-	0,333	0,08	0,3	0,2	0
F			-	0,125	0,26	0,31	0
(G,I)				-	0,4	0,03	0,41
H					-	0,18	0,3
K						-	0,07
L							-

Tablo 9 "0,5" Eşik değerine göre 3. iterasyon sonucu benzerlik katsayıları matrisi

	(A,D,E,G,I)	(B,C)	F	H	K	L
(A,D,E,G,I)	-	0,4	0,1	0,2	0,2	0,2
(B,C)		-	0,333	0,3	0,2	0
F			-	0,26	0,31	0
H				-	0,18	0,3
K					-	0,07
L						-

0,4 eşik değerine göre oluşturulan benzerlik katsayıları matrisi Tablo 10’da görülebilir.

Tablo 10 "0,4" Eşik değerine göre oluşturulan benzerlik katsayıları matrisi

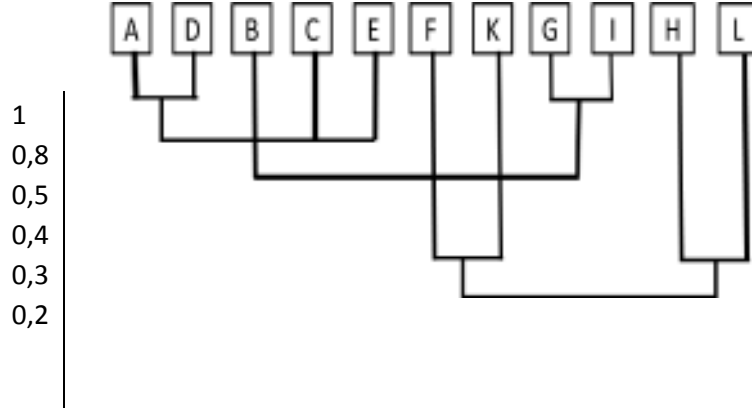
	(A,D,E,G,I,B,C)	F	H	K	L
(A,D,E,G,I,B,C)	-	0,2	0,25	0,2	0,1
F		-	0,26	0,31	0
H			-	0,18	0,3
K				-	0,07
L					-

0,3 eşik değerine göre oluşturulan benzerlik katsayıları matrisi Tablo 11’de görülebilir.

Buradan sonra üretim ortamının diğer kısıtlarına göre iterasyonlar sonlandırılır veya devam ettirilebilir. Şekil 11, yapılan gruplandırmalara ait dendogramı göstermektedir.

Tablo 11 "0,3" Eşik değerine göre oluşturulan benzerlik katsayıları matrisi

	(A,D,E,G,I,B,C)	(F,K)	(H,L)
(A,D,E,G,I,B,C)	-	0,2	0,18
(F,K)		-	0,17
(H,L)			-



Şekil 11. Ortalama bağlantılı kümelenendirme örneğine ait dendogram

Yukarıda verilen tek bağlantılı kümelenendirme, ortalama bağlantılı kümelenendirme ve ilave olarak tam bağlantılı kümelenendirme metotları hücre oluşturma problemine uygulanan en bilinen kümelenendirme metotlarıdır. Bu metotlarla ilgili, literatürde yapılan karşılaştırma çalışmalarının özeti Tablo 12’de verilmiştir.

3.1.1.3 Makine Parça Hücre Oluşturma (MACE)

Özellikle King’in kullandığı derece sıra algoritmasının makine parça matrisindeki elemanların pozisyonlarına bağlı olması ve istisnai elemanları minimize edememesi nedeniyle Waghodekar ve Sahu MACE adını verdikleri hiyerarşik kümelenendirme algoritmasını geliştirmişlerdir.

Bu metot aşağıdaki kavramlara dayanır (Waghodekar ve Sahu, 1984: 937-948):

- 1) Hücre oluşturma yakın benzerliğe sahip tüm makineleri bir hücre altında bir araya toplamayı amaçlar.
- 2) Makine ve parçaların herhangi bir sıra veya numaralandırmalarına bağlı olmaksızın 2 makine arasındaki karşılıklı (benzerlik katsayısı bağlamında) ilişki değişmez kalır.
- 3) i ve j makineleri arasındaki benzerlik katsayısı aşağıdaki yollardan biriyle açıklanabilir:

Tablo 12. Tek bağlantılı, tam bağlantılı ve ortalama bağlantılı kümelenendirme metotlarının avantaj ve dezavantaj karşılaştırması (Yin ve Yasuda, 2005: 471-489).

Prosedür	Avantaj	Dezavantaj
Tek bağlantılı kümelenendirme	Az miktarda hesaplama gerekliliği.	Zincirlemeye en fazla eğilim, hücrelerde az yoğunluğa yol açar, en yüksek derecede tek parçalı hücreler.
Tam bağlantılı kümelenendirme	Az miktarda hesaplama gerekliliği.	En kötü prosedür olarak performans göstermiştir.
Ortalama bağlantılı kümelenendirme	En iyi prosedür olarak performans göstermiştir: en az derecede zincirleme oluşturmuştur, en yüksek hücre yoğunluğu sağlamıştır, benzerlik katsayısı seçiminde farksız, az sayıda tek parçalı hücre.	En yüksek derecede makine çiftlemeye ihtiyaç duyar, daha fazla hesaplama gereksinim duyar.

- Eklenti tipi benzerlik katsayısı (Carrie 1973, Rajagopalan ve Batra 1975) $SC_{ij} = NCC_{ij} / (TNC_i + TNC_j - NCC_{ij})$
- Her bir i ve j makinesinde işlenen toplam parça sayısına dayanan ürün tipi benzerlik katsayısı $PSC_{ij} = (NCC_{ij} \times NCC_{ij}) / (TNC_i \times TNC_j)$
- Bir makine tarafından işlenen ortak parçaların toplam akışına dayanan benzerlik katsayısı $SCTF_{ij} = (NCC_{ij} \times NCC_{ij}) / (TFC_i \times TFC_j)$

NCC_{ij} : i ve j makinelerinin her ikisini birden kullanan ortak parça sayısı.

TNC_i : i makinesini kullanan toplam parça sayısı.

N: Makine sayısı.

TFC_i : Tüm kalan makinelere göre i makinesinde işlenen parçaların toplam akışı.

$$TFC_i = \sum_{j=1}^N NCC_{ij} \quad (2)$$

Algoritmanın akışı aşağıdaki gibidir:

1. Aşama: Benzerlik katsayısı temelinde makine gruplarının belirlenmesi.
 - a) MCI_{ij} makine parça girdi matrisini oku.
 - b) MI_{ij} (machine incidence matrix)
 - c) TNC_i ve TFC_i 'yi hesapla.
 - d) S_{ij} benzerlik katsayı matrisini hesapla.
 - e) CELFO'yu çağır.
2. Aşama: Hücreler arası akışların belirlenmesi ve hücreler arası akış benzerlik katsayısı temelinde hücrelerin gruplanması.
 - f) Hücreler arası akışı hesapla.
 - g) S_{ij} hücreler arası akış benzerlik matrisini hesapla.
 - h) CELFO'yu çağır.
1. Aşama: h adımıyla hesaplanan makine sıralarına göre parça yerleştirme.
 - i) h adımıyla hesaplanan makine sıralarına göre oluşan parça yerleştirmesini yap.
 - j) MCO_{ij} makine parça çıktı matrisi.

3.1.2 Hiyerarşik Olmayan Metotlar

Paylaştırma kümelenme olarak da bilinen hiyerarşik olmayan metotlar, küme içindeki objelerin birbirleriyle benzer, diğer kümelerdeki objelerden farklı olduğu çeşitli kümeler yaratırlar. Literatürde en çok bilinen hiyerarşik olmayan metotlar Chandrasekharan ve Rajagopalan'ın 1987'de geliştirdikleri ZODIAC ve Srinivasan ve Narendran'ın 1991'de geliştirdikleri GRAFICS algoritmalarıdır.

3.1.2.1 ZODIAC

Bu metot Chandrasekharan ve Rajagopalan'ın 1986 yılında geliştirdikleri ideal çekirdek kümeleme algoritmasının (Ideal seed clustering algorithm) gelişmiş versiyonudur. ZODIAC (zero-one data: ideal seed algorithm for clustering) metodunda, ilk aşamada çekirdek seçiminin rastgele seçilmesi kısıtlanmamıştır. Çekirdek seçimleri suni, temsil eden ve doğal çekirdekler olabilir. İdeal çekirdek sadece ZODIAC'ın son aşamasında kullanılır. Bu metodun diğer bir özelliği, veri

yapısına ve küme üyeliğine bağlı olarak gruplama etkinliği için genellikle 1'den küçük olan bir limitin olmasıdır.

ZODIAC algoritması 0-1 makine parça matrisinde blok diyagonal yapının oluşturulmasına odaklanır. Bazı durumlarda veri setinin içinde blok diyagonal yapının olmamasına rağmen, algoritma en iyi blok diyagonal yapının oluşturulmasına imkan verir. Ayrıca hücre içindeki makine sayısında herhangi bir kısıtlama olmadan makine parça matrisini blok diyagonal yapıya sokar (Srinivasan ve Narendran, 1991). Bununla birlikte G. Srinivasan ve T. T. Narendran tarafından bu algoritmanın dezavantajları aşağıdaki gibi tanımlanmıştır (Srinivasan ve Narendran, , 1991: 463-478):

1. Çekirdeklerin başlangıç adedi ve çekirdek noktalarının seçimi dikkatli yapılmalıdır. Yapay çekirdeklerin uygun olmayan seçimi başarısız bir yapıya ya da çok fazla sayıda singleton'un oluşmasına neden olur. Veri kümesinin karakteristikleri temelinde üretilen doğal çekirdekler, eğer parametre değerlerinin bazıları dikkatli seçilmezse, ihtiyaç duyulandan daha az çekirdeklerin oluşmasına neden olur.
2. En az "şehir blok uzaklığı" temelindeki kümelenendirme kriteri, parçalar tarafından ihtiyaç duyulan işleme kapsamını yansıtmaz. $S_1=[1100000000]$ ve $S_2=[0000011111]$ gibi iki çekirdek ve $V=[0000110000]$ gibi bir vektör düşünelim. V ve S_1 arasındaki şehir blok mesafesi 4 ve V ve S_2 arasındaki 5'dir. Eğer şehir blok mesafesi temelinde kümelenendirirsek, ortak 1'leri olmadığı halde V , S_2 'ye değil S_1 'e kümelenendirilmelidir. Bu durum doğal, temsili ve ideal çekirdek kümelenendirme yapıldığında genellikle hasta yapılı matrisler ortaya çıkarır. Bu tür açık hata kaynaklarından kaçınmak için kümelenendirme için temel seçiminde dikkatli olunmalıdır.

3.1.2.2 GRAFICS (grouping using assignment method for initial cluster seeds) Metodu

Bu metodun temelinde, hücreleri oluşturmak için benzer makinelerin birlikte gruplandığı ve makineler arası benzerliğin maksimize edildiği makine ataması probleminin çözümü için başlangıç makine hücreleri seçilir. Makine hücreleri

böylece bir ideal şekil kazanır. Verilen makine hücresi seti parçaları kümelemek için çekirdekler sağlaması ve verilen parça ailesi seti makineleri kümelendirmek için çekirdekler üretmek için kullanılmasıyla birlikte bir hiyerarşik olmayan kümeleme algoritması için bir ideal başlangıç noktası elde edilir. Algoritmanın adımları aşağıdaki gibidir (Srinivasan ve Narendran, 1991: 463-478):

- Adım 1: Satır vektörleri için benzerlik matrisini hesapla.
- Adım 2: Atama problemini toplam benzerliği maksimize edecek şekilde satır vektörlerini atayarak çöz.
- Adım 3: Çözümdeki subtour'ların sayısını say. Bu sayıyı k_r olarak belirle. Bu sayı satır küme sayısına eşittir.
- Adım 4: Kolonları kümelemek için satır kümelerinden çekirdekler üret.
- Adım 5: Kendilerine en az bir üye çeken çekirdeklere atayarak singleton kümeleri eleterek, maksimum yoğunluk kuralına göre kolonları kümele.
- Adım 6: Kolon küme sayısını k_c ile belirt. Eğer $k_c=k_r$ ise adım 10'a git.
- Adım 7: Satırları kümelemek için kolon kümelerinden çekirdekler üret.
- Adım 8: Maksimum yoğunluk kuralına göre satırları kümele.
- Adım 9: 4'den 9'a kadar olan adımları küme sayısı $k_c=k_r$ olana kadar tekrarla.

Benzerlik katsayıları matrisini elde etmek için Andrew Kusiak'ın önerdiği benzerlik katsayısı kullanılır (Kusiak, 1987: 561-569).

Herhangi iki 0-1 vektörü için

$$P_i = [a_{i1}, a_{i2}, \dots, a_{ik}, \dots, a_{im}]$$

ve

$$P_j = [a_{j1}, a_{j2}, \dots, a_{jk}, \dots, a_{jm}]$$

Benzerlik katsayısı aşağıdaki gibidir:

$$s_{ij} = \sum_{k=1}^m \delta(a_{ik}, a_{jk}) \quad (3)$$

Burada;

$$\delta(a_{ik}, a_{jk}) = 1 \text{ eğer } a_{ik} = a_{jk}$$

0 aksi halde.

(3) eşitliği kullanılarak hesaplanan benzerlik katsayıları matrisi en büyükleme amaçlı makine atama problemi için girdi oluşturur. Daha sonra bu atamadan kapalı alt döngüler tanımlanır (Srinivasan, Naredran, ve Mahadevan, 1990: 145-152). Bu alt döngüler kolonların kümelenendirilerek makine hücrelerinin oluşturulmasında çekirdek olarak kullanılır.

Sayısal Örnek:

Aşağıdaki makine parça matrisi Chandrasekharan ve Rajagopalan tarafından kullanılmıştır (Tablo 13). Makine parça matrisine ait hesaplanmış benzerlik katsayıları matrisi Tablo 14’de verilmiştir.

Makine benzerlik katsayıları matrisi atama çözümü;

$$X_{13}=X_{24}=X_{31}=X_{42}=X_{56}=X_{65}=X_{78}=X_{87}=1$$

Tablo 13. GRAFICS algoritmasına ait makine parça matrisi.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1		1	1					1	1		1		1	1		1	1		1	
2			1	1		1	1							1				1		1
3		1						1	1		1		1	1		1	1		1	
4			1	1		1	1			1								1		1
5	1				1	1				1		1			1		1			
6	1				1				1	1		1			1					1
7			1	1		1	1				1	1						1		1
8			1	1		1	1											1		1

Tablo 14. GRAFICS algoritması örneğine ait benzerlik katsayıları matrisi.

	1	2	3	4	5	6	7	8
1	0	7	19	5	5	5	6	6
2	7	0	6	18	8	8	17	19
3	19	6	0	4	6	6	5	5
4	5	18	4	0	10	10	17	19
5	5	8	6	10	0	16	9	9
6	5	8	6	10	16	0	9	9
7	6	17	5	17	9	9	0	18
8	6	19	5	19	9	9	18	0

Bu atamada, örneğin 1 nolu makinenin 3 nolu makineye atanması (ve tersi) 1 ve 3 nolu makineler bir alt döngü oluşturur.

Çözümünden 4 alt döngü 1-3-1, 2-4-2, 5-6-5, 7-8-7 tanımlanabilir.

Bunlar 4 makine grubuna karşılık gelir {1,3}, {2,4}, {5,6}, {7,8}.

Bu makine grupları parçaları kümeleyen 4 çekirdeğe neden olur: $S_1=[10100000]$, $S_2=[01010000]$, $S_3=[00001100]$ ve $S_4=[00000011]$

Bu çekirdekleri kullanarak ve parçaları maksimum yoğunluk kuralına göre kümelendirilerek izleyen parça grupları elde edilir: {2,8,9,11,13,14,16,17,19}, {3,4,6,7,18,20}, {1,5,10,12,15}.

4 çekirdek noktası olmasına rağmen S_4 'ün herhangi bir parça çekememesi nedeniyle parça grup sayısı 3'e inmiştir. Bu 3 gruptan, satırları kümelemek için 3 çekirdek elde edilir: $S_5=[01000001101011011010]$, $S_6=[00110110000000000101]$, $S_7=[10001000010100100000]$.

S_5 'den S_7 'ye kadar olan çekirdekleri kullanarak, maksimum yoğunluk kuralına göre makine kümeleri elde edilir. Elde edilen kümeler: {1,3}, {2,4,7,8} ve {5,6}'dır. Böylece uygulanabilir bir çözüm oluşturan eşit sayıda makine ve parça grubu elde edilmiş olur (Tablo 15). Çözüm 9 istisna elemana ve %95,83 gruplama etkinliğine sahiptir.

Son olarak satırlar kümelendiğinden, 3 satır kümesi, parçaları kümelemek için izleyen çekirdekleri üretmek için kullanılır: $S_8=[10100000]$, $S_9=[01010011]$, $S_{10}=[00001100]$.

Yukarıdaki 3 çekirdek kullanılarak kolonlar kümelendirir. Böylece izleyen kolon kümeleri elde edilir: {2,8,9,11,13,14,16,17,19}, {3,4,6,7,18,20}, {1,5,10,12,15}.

Eşit sayıda makine ve parça sayısına sahip olduğundan uygulanabilir çözümdeki istisna eleman ve boşluk sayısı hesaplanır. Aynı çözüm elde edildiğinden dolayı prosedür durdurulur. Blok diyagonal yapı Tablo 16'da gösterilmiştir.

Tablo 15. GRAFICS algoritması örneği için uygulanabilir çözüm.

No	Makineler	Parçalar
1	1,3	2,8,9,11,13,14,16,17,19
2	2,4,7,8	3,4,6,7,18,20
3	5,6	1,5,10,12,15

Tablo 16. GRAFICS algoritması örneği için blok diyagonal matris.

	2	8	9	11	13	14	16	17	19	3	4	6	7	18	20	1	5	10	12	15
1	1	1	1	1	1	1	1	1	1	1										
3	1	1	1	1	1	1	1	1	1											
2						1				1	1	1	1	1	1					
4										1	1	1	1	1	1			1		
7				1						1	1	1	1	1	1				1	
8										1	1	1	1	1	1					
5								1				1				1	1	1	1	1
6			1												1	1	1	1	1	1

3.1.2.3 Küme Tanımlama Algoritması

Küme tanımlama algoritmasının adımları aşağıdaki gibidir:

Adım 1: İterasyon sayısı k 'yı 1'e eşitle.

Adım 2: Makine parça matrisinin $[a_{ij}](k)$ i . satırını seç ve yatay bir çizgiyle satırı kapat

Adım 3: Satırdaki her bir 1 girdisi üzerinden yatay çizgiyi kesen dikey çizgi ile kolonları kapat.

Adım 4: Dikey çizgi tarafından bir kere kesilmiş olan 1 girdileri için ilgili satırda yatay çizgi çiz.

Adım 5: Makine parça matrisinde 2 kez kesilmiş olan bir girdileri satır ve kolon kümelerini oluşturur.

Adım 6: Makine parça matrisinden 1'den 4'e kadar olan adımlarla çizilen satır ve kolonları çıkararak $[a_{ij}](k+1)$ matrisini oluştur.

Adım 7: Eğer $[a_{ij}](k+1)=0$ ise iterasyonu durdur. Aksi halde $k=k+1$ ve 1. adıma dön.

Sayısal Örnek: Aşağıdaki örnekte 7 makine ve 8 parçadan oluşan bir makine parça matrisi verilmiştir.

Tablo 17. Küme tanımlama algoritması için başlangıç matrisi örneği.

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

Tablo 18. Küme tanımlama algoritması 1. iterasyon.

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

Tablo 19. Küme tanımlama algoritması 1. iterasyon sonucu.

	1	4	6	7
2	1		1	
3		1		1
4	1		1	
6		1		

Tablo 20. Küme tanımlama algoritması 2. iterasyon.

	1	4	6	7
2	1		1	
3		1		1
4	1		1	
6		1		

Tablo 21. Küme tanımlama algoritması 3. iterasyon

		4	7
3	1	1	
6	1		

Tablo 22. Küme tanımlama algoritması sonucunda düzenlenen makine parça matrisi.

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

3.1.3 Dizi Tabanlı (array based) Kümelenendirme Metotları

3.1.3.1 Derece Sıra Kümelenendirme Algoritması

Derece sıra algoritması (literatürde bilinen adıyla Rank Order Clustering Algorithm ROC), makine satırlarını ve parça kolonlarını yeniden düzenleyerek blok diyagonal matris elde etmeye çalışan hiyerarşik olmayan bir kümelenendirme metodudur. Kusiak ve Chow tarafından sıralama temelli kümelenendirme algoritmaları sınıfına sokulan yöntem (Kusiak ve Chow, 1988: 457-471), J. R. King tarafından 1980 yılında geliştirilen yöntem üzerinde daha sonra King ve Nakorchai tarafından 1982 yılında, Chandrasekharan ve Rajagopalan tarafından 1986 yılında (MODROC), Chan ve Milner tarafından 1982 yılında (direct clustering algorithm) ve Kusiak tarafından 1985 yılında bazı yenilikler ve iyileştirmeler tanımlanmıştır (Srinivasan ve Narendran, 1991: 463-478). Hiyerarşik metotların aksine benzerlik katsayıları kullanılmadan makine parça matrisi üzerindeki satır ve kolonların ondalık sayı sistemine karşılık gelen değerleri tespit edilir ve bu değerlere göre yeniden sıralama yapılarak makine parça matrisi düzenlenir. İterasyonlar halinde devam ettirilen

algoritma mümkün olduğunca çok sayıda ve dolayısıyla küçük gruplar elde etmeye çalışır.

Derece sıra algoritmasının akış prosedürü aşağıdaki gibidir (King, 1980: 213-232):

Adım 1: Makine parça matrisinin her satırı için “1” girdilerini ikili sistemde oku. Satırları azalan ikili değer sırasında derecele. Aynı değerde satırlar için dereceleme satırlar arasında uygun biçimde yap.

i. satır için: $\sum_{k=1}^n a_{ik} 2^{n-k}$ (Kusiak ve Chow, 1988: 457-471)

Adım 2: Oluşan matris satır sırası, adım 1’de belirlenen derece sırasının aynısı ise algoritmayı durdur. Aksi durumda adım 3’e git.

Adım 3: Azalan derece sırasında satırları düzenleyerek makine parça matrisini yeniden oluştur. Matrisin her kolonu için “1” girdilerini ikili sistemde oku. Kolonların derece sıralarını belirle.

j. kolon için: $\sum_{k=1}^m a_{kj} 2^{n-k}$ (Kusiak ve Chow, 1988: 457-471)

Adım 4: Oluşan matris kolon sırası adım 3’ün aynısı ise algoritmayı durdur. Aksi durumda adım 5’e git.

Adım 5: Azalan derece sırasında kolonları yeniden düzenleyerek makine parça matrisini yeniden oluştur. Adım 1’e geri dön.

Aşağıdaki örnekte 10 ayrı makinede işlem gören 8 parçaya ait bir makineye ait makine parça matrisi verilmiştir.

Adım 1: Tablo 23’de görüldüğü gibi satırlara ait ikili bilgi okutulur ve satırların karşısına ondalık eşdeğerleri yazılır. Ondalık eşdeğerlere göre yapılan sıralama ise derece sırası kolonuna yazılır. Örnek olarak 1. Satırın ikili bilgisi 10100101 şeklindedir. Ondalık eşdeğer ise $2^7+2^5+2^2+2^0=165$ olarak hesaplanır.

Adım 2: Satırların belirlenen derece sırası, halihazırdaki satır derecesinden farklıdır.

Adım 3: Derece sıra kolonuna göre satırların yeniden sıralandırılmasıyla Tablo 24’deki matris elde edilir. Kolonların ikili sistemdeki ondalık eşdeğerleri hesaplanarak her bir kolonun altına kaydedilir ve derece sıraları belirlenir.

Tablo 23. Derece sıra algoritması için örnek makine parça ve başlangıç matrisi.

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0		
	1	2	3	4	5	6	7	8	Ondalık Eşdeğer	Derece Sırası
A	1		1			1		1	165	1
B	1					1			132	4
C		1			1			1	73	5
D	1		1			1			164	2
E				1			1		18	8
F		1			1			1	73	6
G					1			1	9	10
H	1		1			1			164	3
I				1			1		18	9
J		1					1		66	7

Tablo 24. Derece sıra algoritması örneği 1. iterasyon sonucu.

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
	1	2	3	4	5	6	7	8	İkili Ağırlık
A	1		1			1		1	2^9
D	1		1			1			2^8
H	1		1			1			2^7
B	1					1			2^6
C		1			1			1	2^5
F		1			1			1	2^4
J		1					1		2^3
E				1			1		2^2
I				1			1		2^1
G					1			1	2^0
Ondalık eşdeğer	960	56	896	6	49	960	14	560	
Derece sırası	1	5	3	8	6	2	7	4	

Adım 4: Kolonların belirlenen derece sıraları halihazırdaki derecesinden farklı olduğundan algoritma devam ettirilir.

Adım 5: Kolonların derece sıralarına göre yeniden düzenlenmiş şekilde görülmektedir.

Adım 2: Satırların belirlenen derece sırası, şu anki sırasının aynısı olduğundan iterasyon durdurulur. Elde edilen matris nihai makine parça matrisidir ve makine hücreleri (A,D,H,B), (C,F,G) ve (J,E,I) şeklinde gruplanır. İstisnai elemanları inceleyecek olursak; 8 nolu parça 1. hücreyi ve 2. hücreyi, 2 nolu parça 2. ve 3. hücreyi kullanmaktadır. Bu parçalar istenmeyen hücreler arası harekete neden

olmaktadır ve ilave karar gerektirmektedir.

Tablo 25. Derece sıra algoritması örneği 2. iterasyon sonucu.

	1	6	3	8	2	5	7	4	Ondalık	Derece
A	1	1	1	1					240	1
D	1	1	1						224	2
H	1	1	1						224	3
B	1	1							192	4
C				1	1	1			28	5
F				1	1	1			28	6
J					1		1		10	8
E							1	1	3	9
I							1	1	3	10
G				1		1			20	7

Tablo 26. Derece sıra algoritması örneği 3. iterasyon sonucu.

	1	6	3	8	2	5	7	4	İkili
A	1	1	1	1					2^9
D	1	1	1						2^8
H	1	1	1						2^7
B	1	1							2^6
C				1	1	1			2^5
F				1	1	1			2^4
G				1		1	1		2^3
J					1		1		2^2
E							1	1	2^1
I							1	1	2^0
Ondalık	960	960	896	568	52	56	7	3	
Derece	1	2	3	4	5	6	7	8	

Tablo 27. Derece sıra algoritması örneği 4. iterasyon sonucu.

	1	6	3	8	5	2	7	4	Ondalık eşdeğer	Derece sırası
A	1	1	1	1					240	1
D	1	1	1						224	2
H	1	1	1						224	3
B	1	1							192	4
C				1	1	1			28	5
F				1	1	1			28	6
G				1	1				24	7
J						1	1		6	8
E							1	1	3	9
I							1	1	3	10

Chandrasekharan ve Rajagopalan 1986 yayınladıkları makalede derece sıra algoritması üzerinde yaptıkları çalışmada algoritmanın aşağıdaki dezavantajlarını tespit etmişlerdir (Chandrasekharan ve Rajagopalan, 1986: 1221-1233).

- 1) Derece sıra algoritması, iyi yapılandırılmış matrislerde bile kesin blok diyagonal matrisi sonuç olarak vermez.
- 2) Son kararlı çözüm, istenen çözüm değildir. Başka bir deyişle ara iterasyonlarda ortaya çıkan makine parça matrisi istenen blok diyagonal matrise sahip olmasına rağmen, yapı daha sonraki iterasyonlarda bozulabilir veya istisnai elemanlar ortaya çıkabilir.
- 3) Algoritma kullanılarak elde edilen sonuç güçlü bir şekilde matrisin başlangıç düzenine bağlıdır. Bu durum darboğaz makinelerin rastgele olarak belirlenmesine neden olur.
- 4) 1 bilgisinin sol üst köşeye toplanması eğilimi olduğundan, matrisin kalan bölümü düzensiz hale gelir. Bu nedenle sadece bir hücre ve ilgili parça ailesi oluşma riski vardır.

3.1.3.2 Değiştirilmiş Derece Sıra Kümelenendirme Algoritması (Modified Rank Order Clustering Algorithm)

Derece sıra kümelenendirme algoritmasının yukarıda verilen dezavantajları nedeniyle Chandrasekharan ve Rajagopalan tarafından algoritma üzerine değişiklik yapılarak “değiştirilmiş derece sıra kümelenendirme algoritması” geliştirilmiştir. Algoritma üç aşamadan meydana gelir.

Birinci aşamada başlangıç makine parça matrisine iki iterasyon derece sıra kümelenendirme algoritması uygulanır. Oluşan sıralanmış matriste eğer i . satırda ilk k elemanlar bir ise, $(i-1)$. Satırda bulunan k elemanlar birdir. Bu, ayrıca kolonlar için de geçerlidir. Sıralanmış matrisin bu özelliği, algoritmanın 2. aşamasında bir blok tanımlanmasında kullanılır.

İkinci aşamada sol üst köşede oluşan kare veya dikdörtgen matrisin sınırları belirlenir. Sınır arama prosedürü a_{11} elemanından başlanılarak a_{22} diyagonalı üzerinden bir “0” elemanıya karşılaşılan kadar devam ettirilir. i değeri $(i-1)$ ’e

düşürülerek $a_{(i-1)j}$ elemanına dönülür. İlgili kolon üzerinde “0” elemanına rastlanana kadar kolon üzerinde yukarı devam edilir ($a_{(i-1),(j-1)}$). Belirlenen bloğa ait kolonlar matristen çıkarılır ve ilgili parça ailesi ve makine hücresi algoritmanın ikinci aşamasının çıktısı olarak kaydedilir. Yukarıda verilen birinci aşama matrisin kalan kısmına uygulanır ve prosedür tüm makine hücreleri belirlenene kadar tekrarlanır.

Üçüncü aşamada oluşan bloklar arasındaki benzerlik katsayıları matrisi hesaplanır. En yüksek benzerlik katsayısına sahip hücreler ve parça aileleri birleştirilir. Benzerlik katsayıları yeniden hesaplanır ve en yüksek katsayı belirlenir. Eğer en yüksek katsayı 0 ise grup sayısı 1 olana kadar yukarıdaki prosedür tekrarlanır.

Sayısal Örnek:

Aşağıda algoritmanın işleyişiyle ilgili sayısal bir örnek verilmiştir. Başlangıç matrisi 8 makine ve 20 parçadan oluşmaktadır (Tablo 28). Başlangıç makine parça matrisine derece sıra algoritması 2 iterasyon uygulandığında ortaya çıkan satır ve kolon sıralamaları Tablo 29’da görülmektedir. Böylece algoritmanın birinci aşaması tamamlanmış olur. Daha sonra sol üst köşede bulunan bloğu tanımlamaya sıra gelir. İkinci aşamada uygulanan blok arama prosedürünün uygulanması Tablo 29’da gösterilmiştir. Blok arama sonucunda bulunan sağ üst köşedeki en geniş blok matristen çıkarılır ve 1. aşamaya dönülerek matrisin kalanına 2 iterasyon derece sıra algoritması uygulanır ve tüm kolonlar gruplanana kadar buraya kadar olan prosedür tekrarlanır.

İkinci aşama sonucunda bulunan temel makine hücreleri ve parça aileleri

Tablo 30’da verilmiştir. Daha sonra üçüncü aşamada daha önce tespit edilmiş olan hücrelerin benzerlik katsayı matrisi hesaplanır.

$$s_{ij} = \frac{n(C_i \cap C_j)}{n(\text{Min}[C_i, C_j])}$$

(4)

Burada görüldüğü gibi benzerlik katsayıları iki temel hücre arasındaki benzerlik, ortak eleman sayısının, en küçük hücredeki eleman sayısına oranı şeklinde ifade edilmiştir.

Tablo 28. Değiştirilmiş derece sıra algoritmasına ait başlangıç makine parça matrisi (Chandrasekharan ve Rajagopalan, 1986: 1221-1233).

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	1		1	1					1	1				1	1	1		1	1	
2		1	1	1		1	1		1		1							1		1
3					1	1	1	1			1	1	1			1	1		1	1
4			1	1			1	1	1	1			1	1	1		1	1	1	1
5	1	1	1			1				1		1	1		1	1	1		1	1
6	1	1			1		1	1		1	1	1		1	1	1		1	1	
7					1	1	1	1			1	1	1			1	1		1	1
8	1	1	1	1	1			1	1	1			1	1		1	1			

Tablo 29. Değiştirilmiş derece sıra algoritması 2. iterasyon sonuç matrisi ve blok arama prosedürünün uygulanma şekli.

	10	16	1	2	3	17	13	14	8	5	4	9	19	15	12	20	6	18	7	11
8	1	1	1	1	1	1	1	1	1	1	1	1								
5	1	1	1	1	1	1	1						1	1	1	1	1			
6	1	1	1	1				1	1	1			1	1	1			1	1	1
1	1	1	1	1	1			1			1	1	1	1				1		
4	1				1	1	1	1	1		1	1	1	1		1		1	1	
3		1				1	1		1	1			1		1	1	1		1	1
7		1				1	1		1	1			1		1	1	1		1	1
2				1	1						1	1				1	1	1	1	1

Tablo 30. Değiştirilmiş derece sıra algoritması 2. aşama sonunda bulunan temel hücreler ve parça aileleri (Chandrasekharan ve Rajagopalan, 1986: 1221-1233).

No	Makineler	Parçalar
1	8,5,6	10,16,1,2
2	8,4,5	3,17,13
3	8,4,1	9,4,14
4	6,3,7	8,5,19,7,12,11
5	4,6,1	15,18
6	2,3,7,5	20,6

Üçüncü aşamada hesaplanan temel hücrelerin benzerlik katsayıları matrisi Tablo 31’de verilmiştir.

Tablo 31. Değiştirilmiş derece sıra algoritması 3. aşamaya ait benzerlik katsayıları matrisi (Chandrasekharan ve Rajagopalan, 1986: 1221-1233).

2	0,67				
3	0,33	0,67			
4	0,33	0	0		
5	0,33	0,33	0,67	0,33	
6	0,33	0,33	0	0,67	0
	1	2	3	4	5

En yüksek benzerlik katsayısı olan 0,67'den (2,1 konumunda yer alan) başlayarak yapılan hiyerarşik kümelendirme Tablo 32'de verilmiştir. Burada I 0 benzerlik, II 0,33 benzerlik, III 0,67 benzerlik seviyesidir. Bundan sonra kullanıcı hiyerarşi seviyesini seçerek algoritmayı burada durdurur.

3.2 İkili Makine Parça Matrisini Kullanan Metotlar İçin Genel Değerlendirme

Buraya kadar verilen hücre oluşturma metotları ikili makine parça işleme bilgisini girdi olarak kullanan metotlardı. Bu metotlar kısıtlı veri kullanımı ve bazı kısıtların probleme dahil edilememesi nedeniyle gerçek üretim ortamındaki problemlerin çözülmesinde yetersiz kalmaktadırlar. Bu metotların genel dezavantajları aşağıda özetlenmiştir (Nair ve Narendran, 1998: 157-179):

1. İmalat operasyonlarının sırası. Operasyon sırası malzeme akışı üzerinde bir etkiye sahiptir. Bir parçanın orta sıradaki operasyonlarının hücre dışında yapılması iki adet hücreler arası harekete neden olur. Eğer ardışıklarsa malzeme taşıma açısından hücre dışı birden fazla operasyonlar önem taşımazlar.
2. Aynı makinede ardışık olmayan operasyonlar. İkili makine parça matrisi yaklaşımında aynı parça tarafından bir makineye yapılan ardışık olmayan ziyaretler dikkate alınmaz.
3. Hücreler arası hareket hacmi. Parti miktarları genelde eşit büyüklükte değildir. Diyagonal blok dışındaki bir "1" verisi, parçanın operasyon tipi ve hacmine bağlı olarak birden fazla hücreler arası hareketi gösterebilir. Hücreler arası hareket hacminin en küçükleme azaltılmış malzeme taşımayı sağlar.

4. Malzeme taşıma maliyeti. Malzeme taşıma hücreler arası hareketin hacmine, parça boyutuna ve ağırlığına bağlıdır. Hücre içi taşımanın hücreler arası taşımadan daha hızlı olduğu unutulmamalıdır.

Tablo 32. Değiştirilmiş derece sıra algoritması 3. aşama sonucunda hiyerarşik olarak oluşturulan makine hücreleri ve parça aileleri (Chandrasekharan ve Rajagopalan, 1986: 1221-1233).

Hiyerarşik Kümelendirme	Grup No	Makineler	Parçalar
I	1	8,5,6,4	10,16,1,2,3,17,13
	2	8,4,1	9,4,14
	3	6,3,7	8,5,19,7,12,11
	4	4,6,1	15,18
	5	2,3,7,5	20,6
II	1	8,5,6,4	10,16,1,2,3,17,13
	2	2,3,7,5	20,6
	3	8,4,1,6	9,4,14,15,18
	4	6,3,7	8,5,19,7,12,11
III	1	8,5,6,4,1	10,16,1,2,3,17,13,9,4,14,15,18
	2	2,3,7,5	20,6
	3	6,3,7	8,5,19,7,12,11
IV	1	8,5,6,4,1	10,16,1,2,3,17,13,9,4,14,15,18
	2	2,3,7,5,6	20,6,8,5,19,7,12,11
V	1	8,5,6,4,1,2,3,7	10,16,1,2,3,17,13,9,4,14,15,18,20,6,8,5,19,7,12,11

3.3 Operasyon Sırasını Kullanan Metotlar

Yukarıda verilen metotların sadece parçaların makine ihtiyaçlarını dikkate alması ve malzeme akış şeklini yansıtmaması araştırmacıları operasyon sıra verisini kullanmaya yöneltmiştir. Sıra verisi, imalat ayarında işlenecek işlerin sırasını tanımlar. Bu nedenle operasyon sıra verisinin kullanımı parça aileleri ve makine gruplarının oluşturulmasını sağlamakla kalmayıp, her bir hücre içindeki baskın malzeme akışlarına göre hücre yerleşimini de belirler (Mahdavi, Shirazi, ve Paydar, 2008: 943-953). Dolayısıyla parçaların işlem gördükleri makinelerin bilgisine dayanan ikili matris girdisini kullanan metotlar genellikle sadece parça gruplaması

veya sadece makine gruplaması yaparlar. Oysa kullanıcıya operasyon sırası, akış şekli, akış miktarı gibi ilave bilgi sağlayan operasyon sırasını gösteren matris girdisini temel alan algoritmalar makine ve parça gruplamasının yanı sıra hücre ve hücre içi makine yerleşimini de tasarlamak mümkündür. Tipik bir operasyon sıra verisini gösteren makine parça matrisi aşağıdaki gibi ifade edilebilir.

$$X_{ij} = \begin{cases} k & \text{eğer } i \text{ parçası } k. \text{ operasyon için } j \\ & \text{makinesine uğrarsa} \\ 0 & \text{aksi durumda} \end{cases}$$

Yukarıdaki ifadeye göre operasyon sırasını gösteren bir makine parça matrisi örneği Tablo 33’de gösterilmiştir. Matrisi incelediğimizde örneğin 2. satır 6. kolonda bulunan 4 ifadesi, 2 nolu parçanın 4 nolu operasyonunun 6 nolu makinede gerçekleştirildiğini gösterir. Ancak bir makine eğer iki veya daha fazla ardışık olmayan operasyonu gerçekleştirdiği durumda ilgili matrisin yetersiz kalacağı açıktır. Bu durumda matristeki operasyonlar vektör olarak gösterilir. Örneğin 2 nolu parça 1 ve 3 nolu operasyonlarını 2 nolu makinede gerçekleştiriyorsa, matrisin 2. satır ve 2. kolonundaki ifade (1,3) olarak gösterilir.

Sonraki bölümde de gösterileceği gibi bazı uygulamalarda matrise eklenecek ilave kolonlarla da ifade edilmektedir. Yukarıdaki örneği ele alırsak matris kolonlarına eklenecek ilave bir 2 nolu kolon ile 2 nolu parçanın 3. operasyonu burada gösterilir.

Tablo 33. Operasyon sıra verisine göre düzenlenmiş bir makine parça matrisi.

		Makine							
		1	2	3	4	5	6	7	8
Parça	1			1			3		2
	2		3		1		4		2
	3		3		1		2	4	
	4	3		1			4		2
	5		3	1			2		
	6	3			1		4		2
	7	3	2	1			4		5
	8		2		1				
	9		3			1	2		
	10	3	2			1		4	5

3.3.1 Operasyon Sırasını Kullanan Benzerlik Katsayıları

3.3.2 Makine Hücrelerinin Gruplanması ve Yerleştirilmesi

Şimdiye kadar incelenen makine hücresi ve ilgili parça ailelerinin oluşturulması yöntemlerinde gerçek imalat ortamının önemli kısıtlarının dikkate kısmen veya tamamen dikkate alınmadığı görülmektedir. Ayrıca bu metotlarda hücre içi malzeme akış maliyetlerini minimize eden yerleşim düzeni tasarımı da tamamen kullanıcının sezgisel inisiyatifine bırakılmıştır. Bu tezin konusu olan hücre ve yerleşim düzeni tasarımında her iki problemin yani hem makine parça gruplama, hem de yerleşim düzeni probleminin birleştirilmesi Heragu, ve Kakaturi tarafından denenmiştir (Heragu ve Kakaturi, 1997: 561-571). Algoritma, S. S. Heragu ve Y. P. Gupta'nın 1994 yılında yayınladıkları "A heuristic for designing cellular manufacturing facilities" isimli makalelerinde önerdikleri algoritmanın geliştirilmiş şeklidir. Bu algoritma ikili makine parça matrisini girdi olarak kullandığı gibi hücre içi makine ve hücre yerleşim düzeni problemleriyle ilgilenmez. Bunun üzerine yapılandırılan algoritmada operasyon sıra verisi kullanılarak makine hücrelerinin gruplanması ve yerleşim düzeni tasarlanırken malzeme akış hacmi göz önüne alınmıştır. Bu algoritmanın diğer bir özelliği ise emniyet ve teknolojik gerekliliklerin karşılanması ve önceden belirlenmiş olan hücre büyüklüğü ve hücre sayısının aşılmaması gibi iki önemli kısıtın dikkate alınmasıdır. Aslında adı geçen kısıtlar Heragu ve Gupta'nın 1994 tarihinde sundukları "Hücresel imalat tesislerinin tasarımı için bir sezgisel yaklaşım" adındaki çalışmada da dikkate alınmıştır (Heragu ve Gupta, 1994: 125-140), ancak bu çalışmada ikili makine parça matrisi girdi olarak kullanılmış ve yerleşim düzeni tasarımı problemine girilmemiştir. Bu bölümde incelenen algoritma yukarıda adı geçen çalışmaya dayanmaktadır.

Metot, makine gruplama ve ilgili parça ailelerinin oluşturulması problemini operasyon sırası bilgisini içeren makine parça matrisi girdisini temel alarak çözer (Tablo 34). Makine parça matrisinde alternatif rotalar ve aynı makinede gerçekleştirilen birden fazla operasyonun bilgisi kolaylıkla elde edilebilmektedir. Örneğin 1 nolu makinenin 4 nolu operasyonu hem 9 nolu, hem de 10 nolu makinede yapılabilmektedir. Aynı makinede gerçekleştirilebilen birden fazla operasyon ise matris makine kolonlarına ilave edilen kolonlar vasıtasıyla gösterilmiştir. Örneğin 2

nolu parça için rota 1-4-7 şeklinde devam ettikten sonra, işlenecek parça tekrar 4 nolu makineye geri dönerek 4 nolu operasyonu görür ve 8 nolu makinede işlendikten sonra tekrar 7 nolu makineye dönerek 6 nolu operasyonu görür. Operasyon sırasına dayanan makine parça matrisi üzerinde aynı zamanda dikkate alınacak olan aylık ve yıllık talep ve imalata giren partilerin büyüklüğü bilgisi de yer almaktadır.

Önemli kısıtlardan biri olan teknolojik ve emniyet kısıtları ile ilgili bilgi için makine ilişki matrisine başvurulur. Bu matrisin oluşturulabilmesi için önceden makinelere ait teknolojik ve emniyet bilgilerinin elde edilmesi gerekir. Örneğin kaynak makinesi ve boya makinesinin aynı hücrede bulunması emniyet açısından sakıncalıdır. Burada aynı hücrede bulunma zorunluluğu olan makineler A ilişkisine sahip, aynı hücrede bulunmaması gereken makine X ilişkisine sahiptir şeklinde tanımlanmıştır³. Bu kısıt daha sonraki bölümlerde yer alan genetik algoritmada da dikkate alınmıştır.

Metot, taşıma maliyetlerinin minimize edilebilmesi için malzeme taşıma maliyet matrisini kullanır. Matristeki bilgi, bir i ve j makinesi arasındaki birim mesafede yapılan birim malzeme yükleme maliyeti şeklindedir.

Makine kullanım oranları da dikkate alınan diğer önemli bir husustur. Bu matris alternatif proses rotasına sahip parçaların, alternatif makineleri kullanma olasılıklarını gösterir. Alternatif makineye sahip olmayan operasyonlar için girdi bilgisi 1'dir.

Daha önce de belirtildiği gibi algoritmanın kullandığı en önemli kısıtlardan biri de hücre içinde izin verilen makine sayısıdır. Bu bilgi analizci tarafından daha önceden belirlenir.

Diğer bir önemli kısıt ise hücre sayısı üst limitidir. Hücre sayısı üst limiti makine sayısı, belli bir periyotta parça talepleri ki bu bilgi makine parça matrisinde verilir, alternatif makine olasılık matrisinde bulunan olasılık bilgileri, parça parti büyüklüğü ve operasyon sıralarına bağlıdır.

³ Makineler arasındaki emniyet nedeniyle ayrı hücrelere yerleştirilme ve aynı hücrede bulunmaları ile ilgili kısıtı gösteren makine ilişki matrisi için bkz. A heuristic for designing cellular manufacturing facilities, S. S. Heragu, Y. P. Gupta, International Journal of Production Research, 1994, cilt 32, no 1, 133.

Tablo 34 Makine hücrelerinin gruplanması ve yerleştirilmesi yöntemi için operasyon sırasına dayanan makine parça matrisi (Heragu ve Kakuturi, 1997: 561-571).

		MAKİNELER															Parça Talebi	Parti Büyüklüğü	
PARÇALAR		1	2	3	4	5	6	7	8	9	10	11	12	4	7	11			
	1	1			2				3	4	4						2	1	
	2	1			2			3	5					4	6		3	1	
	3	1	2		3			4	5	6							1	1	
	4	1			2			3	3	4							3	1	
	5	1					2	4		5	3						2	1	
	6						1	3	4	5	2						1	1	
	7				2		1		3	4							2	1	
	8		3	1	5	2	4		6	7							1	1	
	9			1	4	2	3		5	6							1	1	
	10	1			1			2	4						3			2	1
	11						1											3	1
	12							2					1	3				1	1
	13												1	2				1	1
	14								2			3	1					3	1
	15	1							2			4	3	6			5	1	1
	16	1							2			4	3	6			5	2	1
	17								2				1	3				1	1
	18						1	2			3							3	1
	19													1				2	1

Aşağıdaki f_{ij} eşitliği i ve j makineleri arasındaki malzeme taşıma ekipmanı tarafından yapılan hareket sayısını hesaplar. Bu şekilde oluşturulan akış matrisi makine gruplarının belirlenmesinde kullanılan tek girdidir.

$$f_{ij} = \sum_{k=1}^m F_{ij}^k \quad (5)$$

$$F_{ij}^k = \begin{cases} (D_k P_{ki} P_{kj} / B_k) & \text{Eğer } |R_{ki} - R_{kj}| = 1, \\ 0 & \text{aksi halde;} \end{cases} \quad (6)$$

m =parça sayısı.

D_k =verilen periyotta k parçası için talep.

P_{ki} = k parçasının i makinesinde işlenme olasılığı.

B_k = k parçası parti büyüklüğü.

R_{ki} = i makinesinin kullanılarak k parçası üzerinde yapılan operasyonun numarası.

F_{ij}^k sadece ardışık operasyonlar için belirlenir.

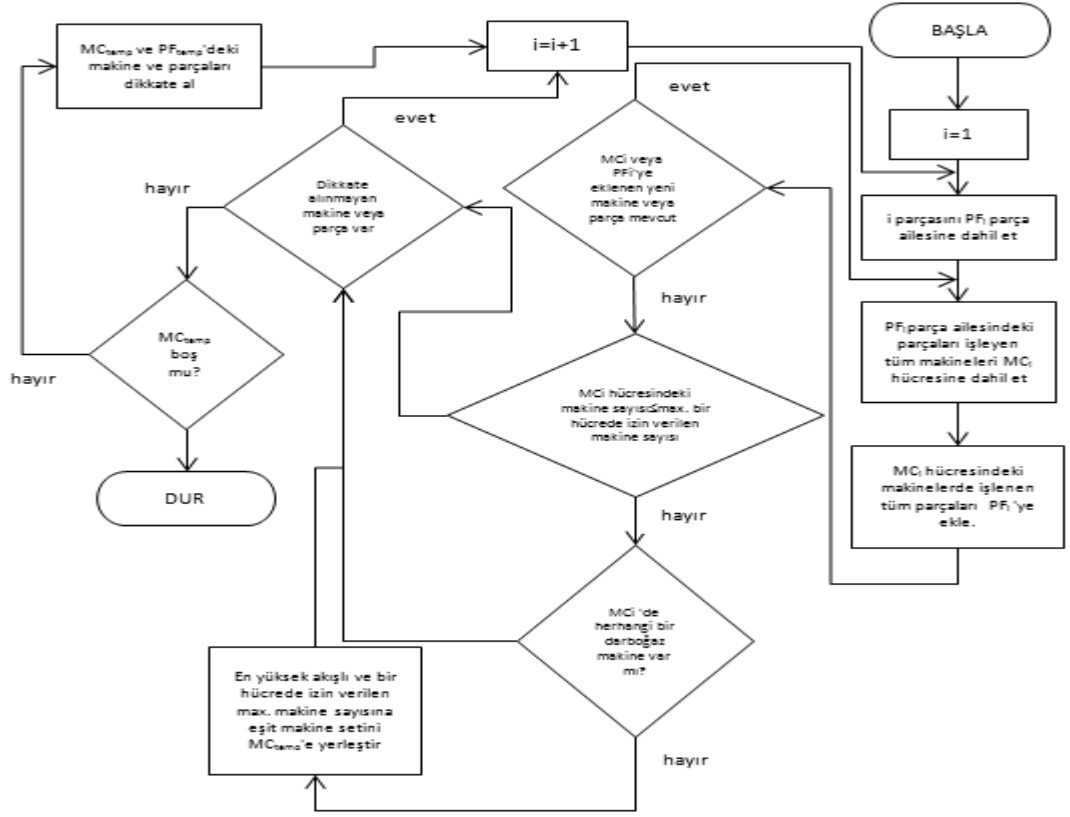
Algoritma 3 aşamadan oluşur. 1. aşamada makineler hücreler halinde gruplanır, 2. aşamada hücre içindeki makinelerin yerleşim düzeni belirlenir, 3. aşamada ise hücrelerin yerleşim düzenleri belirlenir.

1. Aşama: İlk olarak makinelerin hücrelere gruplanması gerçekleştirilir. Bunun için kullanılan algoritmanın akışı Şekil 12’de gösterildiği gibidir.

2. Aşama (hücrelerde makine yerleşimi): Hücrelerin yerleşim düzeni için 3 temel yaklaşım vardır. Bunlar; karesel atama problemi yaklaşımı, düzlemsel grafik yaklaşımları ve karesel olmayan matematiksel modelleme yaklaşımıdır. Bu metotta orantısız tesis yerleşimi problemlerinin modellenmesi avantajları nedeniyle son yaklaşım kullanılmıştır. Yapılan modellemeye her bir makine çifti için akış zamanları, mesafe zamanları, taşıma maliyetleri ve birim makine yükleme bilgileri de eklenir.

3. Aşama (hücrelerin yerleşimi): Hücreler içindeki yerleşim tamamlandıktan sonra her bir hücrenin boyutları ve hücreler arası akış hesaplanır. Burada da 2. aşamada kullanılan algoritma kullanılır.

Hücre yerleşiminde kullanılan tavlama benzetimi yöntemi sonraki bölümde detaylı olarak incelenecektir.



Şekil 12. Makine hücrelerinin gruplanması ve yerleştirilmesi için 1. aşama akış diyagramı (Heragu ve Kakuturi, 1997: 561-571).

3.3.3 Tavlama Benzetimi Yaklaşımı (Simulated Annealing Approach)

Tavlama benzetimi yaklaşımı kombinyonel en iyileme problemlerini⁴ katıların tavlama prosesine benzeyen bir algoritma ile çözer (Wang ve K. B. Wu, 2001: 181-188). Bir tavlama benzetimi algoritması genel olarak bir başlangıç çözümü, komşu çözümler oluşturma yöntemi ve bir tavlama programından oluşmaktadır. Tavlama sürecinin işlevi, yeterince yüksek bir sıcaklıktaki çözümünden başlayıp sıcaklığı aşamalı olarak iyi ve kötü çözümler arasında dolaşmak ve en sonunda en iyi çözüme ulaşmaktır (Ayan, 2009: 101-117). Metot ilk defa S. Kirkpatrick, C. D. Gelatt ve M. P. Vecchi tarafından 1983 yılında yayınlanan “Tavlama Benzetimi ile En İyileme” (Optimizasyon by simulated annealing) isimli makalede tanıtılmıştır. Metodun temeli ısı işlem gören katı cisimlerin ısı enerjisi ile

⁴ Kombinyonel en iyileme bir optimizasyon dalıdır. Uygulama alanı uygulanabilir çözümler kümesinin ayrık veya bir ayrığa indirgenebilir olduğu en iyileme problemleridir.

kararsızlaşan yapılarının, bu enerjiyi kaybettikçe kristalleşme ile kararlı hale gelen yapılarına dayanmaktadır. Çözüm arama sürecinde tavlama benzetimi sadece en iyiyi kabul etmez, belli bir olasılıkla kötü olan komşu çözümleri de kabul eder.

3.3.3.1 Tavlama Benzetiminin Genel Prosedürü

Genel tavlama benzetimi prosedürü aşağıdaki gibidir (Xambre ve Vilarinho, 2003: 434-446):

Başlangıç sıcaklığı seç $T_0 > 0$;

S_0 başlangıç çözümünü seç, bunu mevcut çözüm S ve en iyi mevcut çözüm S^* yap.

tekrarla

tekrarlama sayacı $n=1$;

tekrar

S komşuluğunda S_n çözümünü üret

$\Delta = f(S_n) - f(S)$;

eğer $(\Delta \leq 0)$ ise $S = S_n$;

değilse $S = S_n$ $p = e^{-\Delta/T}$ olasılığı ile;

eğer $(f(S_n) < f(S^*))$ ise $S^* = S_n$;

$n = n + 1$

$n < \text{her bir sıcaklık seviyesinde } (L) \text{ izin verilen tekrar sayısı}$

T sıcaklığını azalt

Durma kriteri doğru olana kadar.

Algoritma bir S_0 başlangıç çözümüyle başlar. Yukarıda verilen akışta iç döngüde n sayacı L 'ye ulaşana kadar mevcut çözümün S_n komşu çözümü üretilir. Eğer S_n , S^* 'den daha iyiyse $(\Delta \leq 0)$ üretilen çözüm mevcut çözümün yerini alır, aksi halde belli bir olasılıkla ($p = e^{-\Delta/T}$) kabul edilir. Algoritmanın dış çevrimine göre sıcaklık derecesi T her iterasyonda azaltılır ve böylelikle kötü çözümlerin kabul edilme olasılığı düşer.

3.3.3.2 Tavlama Benzetiminde Kontrol Parametreleri

3.3.3.2.1 Başlangıç Sıcaklığı

Kirkpatrick ve diğerleri başlangıç sıcaklığı T_0 'ın seçimi için tavlama prosesine başlamadan önce bir deneme koşumu yapmayı önermişlerdir. Buna göre, bu denemede kabul edilmiş olan kötü çözümlerin toplam çözümlere oranı önceden belirlenmiş olan başlangıç kabul olasılığına (P_0) eşit olmalıdır. Başlangıç sıcaklığı, katının içindeki parçacıkların rassal olarak düzenlendiği sıvı faza geçecek kadar büyük olmalıdır. Bunun anlamı başlangıçta, tavlama prosesinin sıcaklığı sistemin tüm mümkün durumlara geçebilecek kadar yüksek olmasıdır. Bu şekilde algoritma, başlangıç konfigürasyonuna kuvvetli bir şekilde bağlı olmayan bir çözüm bulabilir. Hesaplama T_0 aşağıdaki gibi hesaplanır:

$$T_0 \approx \frac{\bar{\Delta}}{\ln P_0^{-1}} \quad (7)$$

3.3.3.2.2 Her bir sıcaklıktaki iterasyon sayısı (L)

Her bir sıcaklık seviyesinin (L) uzunluğu belli bir sıcaklıkta (T) üretilen çözümlerin sayısını belirler. Mevcut çözümün kendisine komşu olan çözümle yer değiştirmesi geçiş (transition) olarak adlandırılmaktadır. Herhangi bir sıcaklıkta geçişlere karar vermek için yapılacak ikili karşılaştırmaların sayısı sınırlandırılmalıdır. Aksi halde tavlama prosesi sınırsız bir döngüye dönüşebilir. L 'nin değeri problem hacminin bir oranı olarak belirlenebilir. Bir başka yol L 'yi belli bir çözümün komşu çözümlerinin sayısının bir oranı olarak belirlemektir. Oluşturulabilecek tüm komşu çözümlerin sayısının önceden biliniyor olması durumunda L toplam deneme sayısının bir oranı olarak da belirlenebilir.

3.3.3.2.3 Sıcaklığı düşürme kuralları

Sıcaklık düşürme Kirkpatrick ve diğerleri (1983) tarafından geliştirilmiş olan sabit oranlı soğutma fonksiyonu ile azaltılmaktadır. Kontrol parametresi sıcaklık azaltma oranı (r), genellikle küçük sıcaklık değişimleri için seçilir ve 0'la 1 arasında önceden belirlenmesi gereken soğutma oranıdır. Tipik olarak $0,7 \leq r \leq 0,95$ kullanılır. Azaltma kuralının fonksiyonu aşağıdaki gibidir:

$$T_k = rT_{k-1}, \quad (0 < r < 1), \quad k=1,2,\dots$$

3.3.3.2.4 Durdurma şartı

Durdurma şartı istenen enerji seviyesine ne zaman ulaşılabileceğini belirler. Tavlama prosesi, iterasyon sayısı önceden belirlenmiş bir değere ulaştığında veya toplam deneme sayısı önceden belirlenmiş bir değere ulaştığında durdurulur. Bunların dışında diğer kurallar hesaplama süresini sınırlamak, sıcaklık değerinin başlangıçta belirlenmiş bir son sıcaklığın T_{min} altına düşmesi şeklinde verilebilir.

Tavlama benzetimi probleminde önceden belirlenmesi gereken kriterler aşağıda verilmiştir:

1. Üretilen çözümlerin toplam sayısı,
2. Ulaşılabilecek istenen enerji seviyesinin sıcaklığı (donma sıcaklığı),
3. Kabul oranı (kabul edilen çözümlerin sayısı ile üretilen çözümlerin sayısı arasındaki oran)

Yukarıda verilen tavlama benzetimi algoritması spesifik problemlere uygulanması sırasında kontrol parametreleri uğraşılacak probleme uygun seçilmelidir. Diğer belirlenmesi gereken iki önemli konu ise başlangıç ve komşu çözümleri üretecek olan probleme özgü prosedürlerdir.

3. Aşama (hücre yerleşimi): Hücre içi yerleşim düzeni belirlendikten sonra 3. aşamada hücrelerin yerleşimi yapılır. Bunun için öncelikle her bir hücrenin büyüklüğü ve aralarındaki akış hesaplanır ve ikinci aşamada kullanılan tavlama benzetimi algoritması tekrar uygulanır.

Çözüm algoritmasının getirdiği hesaplama yükü nedeniyle Heragu ve Kakaturi tarafından bir bilgisayar yazılımı geliştirilmiştir ve bu yazılım özellikle 2. ve 3. aşamalarındaki hücre içi ve hücreler arası yerleşimde kullanıcıya birçok esneklik sağlayarak işletmenin fiziksel olanaklarına göre çeşitli alternatifleri deneme olanağı vermiştir.

3.3.4 Çoklu Benzer Makinelerle İmalat Hücresi Oluşturma İçin Tavlama Benzetimi Yaklaşımı

İmalat hücresi oluşturma problemini operasyon sıra verisi ve akış hacimlerini dikkate alan ve problem için bir başlangıç çözümü oluşturduktan sonra tavlama

benzetimi algoritmasıyla çözen bir yaklaşım Ana R. Xambre ve Pedro M. Vilarinho tarafından (2003) önerilmiştir. Diğer sıra verisini kullanan metotlarda hangi parçanın hangi operasyonunun hangi makinede gerçekleştirileceği önceden belirlenir ve bu bilgi parça makine matrisine girdi teşkil eder. Ancak gerçek üretim ortamında elimizde örneğin torna makinelerinden oluşan bir grup, matkap makinelerinden oluşan bir grup v.b. vardır ve parçalar operasyonlarını bu makinelerde boşluk bulduklarında gerçekleştirirler. Bu metotta amaç, makine kapasite ihtiyaçları ile operasyonlar arasındaki akış hacmini dikkate alarak hücreler arası akışı en küçükmektir. Önerilen yaklaşım operasyon sıra verisini kullanan metotlardan iki yönden ayrılır. İlk olarak operasyon ve makinelerin hücrelere atanmasını sağlar ve böylece bir başlangıç çözümü oluşturulur. İkinci olarak bir operasyonun benzer makinelere bölünebileceğini dikkate alır (bunun anlamı aynı parçanın iki farklı partisinin farklı rotalar izleyebileceğidir).

3.3.4.1 Matematik Programlama Modeli

Varsayımlar;

- i. Zaman içerisindeki iş hacmi karardır.
- ii. Toplam üretim hacmi, verilen bir planlama periyodunda önceden belirlenmiş miktarlarda üretilen parçaların belli bir kümesi olarak temsil edilebilir.
- iii. Her bir parçanın operasyon sırası bilinmekte ve bu nedenle (i,j) operasyon çifti arasındaki f_{ij} akışı bilinmektedir. f_{ij} akışı, parça hacimlerini hareket ettirmek için ihtiyaç duyulan i ve j operasyonları arasındaki malzeme taşıma hareketlerini temsil etmektedir.
- iv. O operasyonlarının her biri farklı parçalar üzerinde gerçekleştirilir.
- v. Üretim $(t=1,...,T)$ makine tipi kümesi şeklinde sınıflandırılan makinelerde gerçekleştirilecektir.
- vi. Her bir operasyon fonksiyonel olarak benzer olan makineler kümesinde $(m=1,...,M_t)$ gerçekleştirilir.
- vii. Her bir i operasyonunun kullanım oranı her bir t makine tipindeki kullanılabilir kapasitenin yüzdesi olarak ifade edilir ve α_{it} ile gösterilir.

- viii. Hücrelerdeki ($c=1,\dots,C$) en fazla izin verilen makine sayısı MC 'dir. Burada en fazla hücre sayısı (C) ve hücrelerdeki en fazla makine sayısı (MC) analizci tarafından belirlenir.

Kontrol edilebilir değişkenler aşağıdaki gibi ifade edilir,

$$x_{mtc} = \begin{cases} 1, & t \text{ tipinin } m. \text{ makinesi } c \text{ hücresine tahsis edilmişse,} \\ 0, & \text{aksi halde} \end{cases}$$

$y_{imt} = t$ tipinin m . makinesinde işlenen i operasyonunun yüzdesi.

Çoklu benzer makinelerle hücre oluşturma problemi aşağıdaki gibi modellenebilir:

$$MinZ = \sum_{i=1}^O \sum_{j=1}^O \sum_{t,u=1}^T \sum_{m,n=1}^{M_t} f_{ij} \cdot y_{imt} \cdot y_{jnu} \cdot \left(1 - \sum_{c=1}^C x_{mtc} \cdot x_{nuc} \right) \quad (8)$$

Kısıtlar;

$$\sum_{i=1}^O y_{imt} \cdot \alpha_{it} \leq 1 \quad (m=1,\dots,M_t; t=1,\dots,T), \quad (9)$$

$$\sum_{t=1}^T \sum_{m=1}^{M_t} x_{mtc} \leq MC \quad (c=1,\dots,C), \quad (10)$$

$$\sum_{m=1}^{M_t} y_{imt} = 1 \quad (i=1,\dots,O; t=1,\dots,T), \quad (11)$$

$$\sum_{c=1}^C x_{mtc} = 1 \quad (m=1,\dots,M_t; t=1,\dots,T), \quad (12)$$

$$x_{mtc} \in \{0,1\}, y_{imt} \in [0,1]. \quad (13)$$

Amaç fonksiyonu (8) hücreler arası akışı en küçükler. Kısıtlar aşağıdaki gibi açıklanabilir:

(9) Her bir makinenin kapasitesinin aşılmasını sağlar,

(10) Her bir hücredeki makine sayısını kısıtlar,

(11) Her bir operasyonun toplam olarak bir veya daha fazla makineye tahsis edilmesini sağlar,

(12) Her bir deęiřkenin kapsamını tanımlar.

3.3.4.2 Önerilen Tavlama Benzetimi Algoritmasının Uygulanması

Önerilen algoritma üretim prosesi detaylarından ve planlama dönemi içindeki talep bilgisinden saęlanan ařaęıdaki bilgilere ihtiyaç duyar:

1. Operasyonlar arası akıř matrisi: Operasyon sırası ve her bir parçanın talep miktarından elde edilir.
2. Her bir operasyon tarafından talep edilen makine tipi.
3. Operasyon başına makine tipi kullanım oranı: Planlama döneminde operasyonu gerçekleřtirmek için gerekli olan operasyon zamanının, ilgili makinenin zamansal kapasitesine oranı.

Analizci ayrıca ařaęıdaki parametreleri tanımlar:

1. Hücrelerde izin verilen en fazla makine sayısı (MC).
2. En fazla hücre sayısı (C).

Başlangıç çözümünü elde etmek için 2 aşama vardır:

1. Aşama: Operasyonlar kullanım oranlarına göre azaltılmış sırada makinelere tahsis edilir.
2. Aşama: Makineler operasyonlara ve operasyonlar arasındaki akıřa göre hücrelere gruplanırlar. Makine gruplama prosedürü, çekirdek makinelerin (her hücre için 1 adet) belirlenmesiyle başlar. Çekirdek makinelerin belirlenmesi kalan makinelerin gruplanması saęlar.

1. Operasyonların Tahsisi:

Her bir makine tipi için, $t=1, \dots, T$:

Adım 1: İhtiyaç duyulan birimler sayısını hesapla

$$M_t = \left\lceil \sum_{i=1}^o \alpha_{it} \right\rceil$$

Adım 2: Operasyonları α_{it} azalan sırasında sırala.

Adım 3: Önceki adımda tanımlanan sırayı izleyerek operasyonları sırasıyla, operasyon için yeterli kapasitesi olan makinelere tahsis et.

Adım 4: Eğer bir operasyon yeterli kapasitesi olmayan makinelerden dolayı tahsis edilemezse iki veya daha fazla makineye bölünür ve en fazla kapasitesi olandan başlanarak tahsis edilir.

2. Makine Gruplama:

Adım 5: Diğer makinelerle en fazla akışa sahip olan makineyi seç ve bir hücreye ata.

Adım 6: Hücrelere atanmış olan makinelerle en düşük akışa sahip olan makineyi seç ve bunu yeni bir hücreye ata. Beraberlik durumunda hücrelere atanmamış makineler arasından bütün makinelerle toplam en yüksek akışa sahip olan makineyi seç.

Adım 7: Önceki adımı tüm C hücreleri kendilerine atanmış 1 adet çekirdek makineye sahip olana kadar tekrarla.

Adım 8: Kalan makinelerin her biri arasında en yüksek ortalama akışa sahip olan hücreye atanırlar. Bir makine ile hücre arasındaki ortalama akış makineden hücrelere atanmış tüm makinelere olan akışla hücredeki makine sayısı oranıdır. Beraberlik durumunda makine rastgele seçilir.

Komşu çözümleri üretecek olan prosedür hücreler arası akış üreten operasyonları seçer ve onları toplam hücreler arası akışı azaltmak için diğer hücreye aktarmaya çalışır. Bu adım, bu operasyonların daha fazla diğer hücreye bağlanması mümkün olmayana kadar tekrarlanır. Bağlanmama nedeni ya kullanılabilir kapasitesi olmayan ilgili tipteki makineler diğer hücreye yerleştirilmiştir, ya da diğer hücrede ihtiyaç duyulan tipte makine yoktur. Bu duruma ulaşıldığında prosedür hücreler arası akış yaratan makineleri farklı hücrelere aktarmaya çalışır. Bu prosedürde yer alan adımlar aşağıdaki gibidir:

Adım 1: En yüksek hücreler arası akış yaratan operasyon çiftini seç.

Adım 2: Önceki adımda seçilen operasyon çiftinden daha az kullanılabilir kapasitesi olan makineye tahsis edilmiş operasyonu seç.

Adım 3: Seçilmiş olan operasyonun en yüksek akışa sahip olduğu makinelerin bulunduğu hücreyi seç. Eğer bu hücrede operasyonun ihtiyaç duyduğu tipte makine ve makinenin işlem yapmak için yeterli kapasitesi varsa operasyonu bu hücreye aktar. Bu gereklilikleri karşılayan birden fazla makine varsa en fazla kapasitesi olanı

seç. Eğer bu değişikliği yürütmek mümkün değilse bu adımı adım 2'den kalan operasyonla tekrarlar.

Adım 4: Komşu çözüm bulunana kadar adım 2 ve 3'ü ürettiği hücreler arası akışın azalan sırası ile adım 2'de seçilen operasyon çiftini seçerek tekrarlar.

Adım 5: Eğer hücreler arası akış üreten tüm operasyon çiftleri test edildiyse ve bir komşu çözüm üretmek mümkün değilse Adım 6'ya git.

Adım 6: En yüksek hücreler arası akışı üreten makineyi seç.

Adım 7: Önceki adımda seçilen makineyi en fazla akışa sahip olduğu makinenin bulunduğu hücreye aktar. Eğer bu aktarma her bir hücredeki en fazla makine (MC) kısıtıyla çatışmazsa;

Adım 8: Eğer Adım 7 başarısız olursa, ürettiği akışın azalan sırası ile makine seçerek komşu çözüm bulunana kadar tekrarlar.

Adım 9: Hücreler arası akış yaratan tüm makineler test edildiyse ve bir komşu çözüm de üretileniyorsa Adım 10'a git.

Adım 10: Başlangıç çözümün üretilmesinde kullanılan aynı prosedürü kullanarak ve ilk çekirdek makineyi rastgele seçerek bir komşu çözüm üret.

Tavlama benzetimi algoritmasının uygulanması için kontrol parametreleri aşağıdaki gibi belirlenmiştir.

1. Başlangıç sıcaklığı (T_0): Hesaplama deneyimleri akış etkisizliğinin iki komşu çözüm arasında %10'dan fazla değişmediğini göstermiştir. Bu nedenle başlangıç sıcaklığını 50 almak alt çözümlerin en az %80'inin kabul edilmesini garanti etmektedir.

2. Sıcaklık azaltma fonksiyonu: Geometrik fonksiyon ($T_i = 0,9T_{i-1}$) kullanılmıştır.

3. Her bir sıcaklık seviyesinin (L) uzunluğu: Her sıcaklık seviyesinde aranacak çözüm sayısı $K \times O$ olarak belirlenmiştir. Burada K analizci tarafından belirlenen sabittir ($K=1$ önerilen değerdir).

4. Durma Kriteri: İki alternatif kriter belirlenmiştir. Birincisinde donma sıcaklığı olarak 10 belirlenmiştir. Bu durumda 16 sıcaklık seviyesi kullanılır ($T_0, a_i^{15} = 50(0,9^{15}) = 10,29$). İkincisinde eğer beş ardışık sıcaklık seviyesinde

üretileen çözümlerin %85'i reddedilirse bulunan en iyi çözümlün yer değıştirme olasılığı çok küçüktür şeklinde bir kabul yapılmıştır ve prosedür durdurulur.

3.3.4.3 Algoritma İçin Bir Örnek

Algoritmanın açıklanacağı örnek 4 makine tipi ve 16 operasyonun bulunduğı bir imalat sistemi üzerine kurulmuştur. Hücrelerde izin verilen en fazla makine sayısı 4 ve en fazla hücre sayısı 3'tür.

$$T=4 \quad O=16$$

$$MC=4 \quad C=3$$

Probleme ait operasyonlar arası akış matrisi Tablo 35'de verilmiştir.

Tablo 36'da her bir operasyonun ihtiyaç duyduğu makine tipi ve kullanım oranları verilmiştir.

Başlangıç çözümünü elde etmek için prosedür, operasyonların makinelere atanmasıyla başlar.

Tablo 35. Operasyonlar arası akış matrisi.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1								112								
2					224											
3														123		
4											137					
5												190				
6			182													
7								227								
8												56				
9				113												
10							222									
11						85										
12													101			
13					209											
14									237							
15										59						
16									246							

Tablo 36. Operasyonlar için makine tipi ve kullanım oranı.

Operasyon	Kullanım Oranı α_{it} (%)	Makine Tipi t
1	22,3	4
2	63,8	2
3	18,5	4
4	67,9	3
5	68,9	1
6	38,5	2
7	38,1	2
8	74,8	3
9	66,5	1
10	10,9	2
11	53,2	4
12	66,4	1
13	26,9	4
14	69,7	2
15	16,4	3
16	39	3

Aşağıda operasyon atamaları gösterilmiştir.

Makine tipi 1 için:

Gereken toplam kapasite=%68,9+%66,5+%66,4=%201,8 ($\sum_{i=1}^O \alpha_{i1}$) olduğundan 3 makine gereksinimi vardır. En yüksek kullanım oranından başlayarak azalan sırada operasyonlar makinelere atanır. Makine tipi 1 için atama sonucu ve makinelerde kalan kapasiteler Tablo 37’de gösterilmiştir.

Tablo 37. Makine tipi 1 için operasyon atama sonucu.

1. Makine			2. Makine			3. Makine		
Operasyon	Kullanılan kapasite	Kalan kapasite	Operasyon	Kullanılan kapasite	Kalan kapasite	Operasyon	Kullanılan kapasite	Kalan kapasite
5	68,9	31,1	9	66,5	33,5	12	66,4	33,6

Makine tipi 2 için:

Gereken toplam kapasite=%63,8+%38,5+%38,1+%10,9+%69,7=%221 ($\sum_{i=1}^O \alpha_{i2}$) olduğundan 3 makine gereksinimi vardır. En yüksek kullanım oranından

başlayarak azalan sırada operasyonlar atandıktan sonra 7. operasyonda makinelerin kalan kapasitelerinden en büyük olan makineye atama yapılır (Tablo 38).

Tablo 38. Makine tipi 2 için operasyon atama sonucu.

1. Makine			2. Makine			3. Makine		
Operasyon	Kullanılan kapasite	Kalan kapasite	Operasyon	Kullanılan kapasite	Kalan kapasite	Operasyon	Kullanılan kapasite	Kalan kapasite
14	69,7	30,3	2	63,8	36,2	6	38,5	61,5
			10	10,9	25,3	7	38,1	23,4

Makine tipi 3 için:

Gereken toplam kapasite=%67,9+%74,8+%16,4+%39=%198,1 ($\sum_{i=1}^O \alpha_{i3}$) olduğundan 2 makine gereksinimi vardır. 8 ve 4 nolu operasyonlar atandıktan sonra başka makine olmadığından 16 nolu operasyon ikiye bölünür. 2 nolu makinenin kapasitesi tamamen dolduktan sonra geri kalan %6,9'luk kullanım oranı 1 nolu makineye aktararak atama tamamlanır (Tablo 39).

Makine tipi 4 için:

Gereken toplam kapasite=%22,3+%18,5+%53,2+%26,9=%120,9 ($\sum_{i=1}^O \alpha_{i4}$) olduğundan 2 makine gereksinimi vardır (Tablo 40).

Tablo 39. Makine tipi 3 için operasyon atama sonucu.

1. Makine			2. Makine		
Operasyon	Kullanılan kapasite	Kalan kapasite	Operasyon	Kullanılan kapasite	Kalan kapasite
8	74,8	25,2	4	67,9	32,1
16	6,9	18,3	16	32,1	0
15	16,4	1,9			

Tablo 40. Makine tipi 4 için operasyon atama sonucu.

1. Makine			2. Makine		
Operasyon	Kullanılan kapasite	Kalan kapasite	Operasyon	Kullanılan kapasite	Kalan kapasite
11	53,2	46,8	13	26,9	73,1
			1	22,3	50,8
			3	18,5	32,3

Makine gruplama prosedürü:

Operasyonlar makinelere atandıktan sonra makine gruplama prosedürüne başlanır. Gruplama prosedürü için operasyonlar arası akış matrisi ve önceki prosedürde gerçekleştirilen atama bilgisi kullanılır (Tablo 41).

Tablo 41. Makineler arası akış matrisi.

	1,1	1,2	1,3	2,1	2,2	2,3	3,1	3,2	4,1	4,2	Toplam
1,1			190		224					209	623
1,2				237			43	316			596
1,3	190						56			101	347
2,1		237								123	361
2,2	224					222	59				505
2,3					222		227		85	182	715
3,1		43	56		59	227				112	497
3,2		316							137		453
4,1						85		137			221
4,2	209		101	123		182	112				<u>727</u>

Burada 16 nolu operasyon 2 makine arasında bölündüğünden akış miktarları da makineler arası akış matrisinde orantılı olarak 3,1 ve 3,2 arasında bölünmüştür.

Diğer tüm makinelerle en yüksek akışa (727) sahip olan 4,2 makinesi 1. hücrenin çekirdek makinesi olarak belirlenir. 4,2 makinesi ile en düşük akışa sahip olan makineler 1,2; 2,2; 3,2 ve 4,1 2. hücre için çekirdek makine adaylarıdır. Beraberlik olduğundan diğer makinelerle en yüksek akışa sahip olan (596) 1,2 seçilir ve çekirdek makine olarak 2. hücreye atanır. 3. hücre için bir sonraki en yüksek akışa sahip (505) olan makine olan 2,2 makinesi çekirdek makine olarak seçilir.

Çekirdek makineler atandıktan sonra kalan makinelerin, hücrelerle olan en yüksek ortalama akışlarına göre hücrelere atanırlar (Tablo 42).

Makine 3,2 en yüksek ortalama akışa sahiptir ve bu nedenle 2. hücreye atanır. Geriye kalan atanmamış makineler içinden tekrar atama yapabilmek için Tablo 41'e göre ortalama akışların tekrar hesaplanması gereklidir (Tablo 43). En yüksek ortalama akışa (224) sahip makine 1,1 olduğundan 3. hücreye atanır.

Tablo 42. İlk iterasyonda atanmamış makinelerle hücreler arasındaki ortalama akış.

Atanmamış Makine	Hücre		
	4,2	1,2	2,2
	1	2	3
1,1	209	0	224
1,3	101	0	0
2,1	123	237	0
2,3	182	0	222
3,1	112	43	59
3,2	0	316	0
4,1	0	0	0

Tablo 43. 2. iterasyonda atanmamış makinelerle hücreler arasındaki ortalama akış.

Atanmamış Makine	Hücre		
	4,2	1,2; 3,2	2,2
	1	2	3
1,1	209	0	224
1,3	101	0	0
2,1	123	119	0
2,3	182	0	222
3,1	112	22	59
4,1	0	69	0

Tüm makineler hücrelere atanan kadar bu adım tekrarlanır. Bu aşamanın sonucu ve tavlama benzetimi algoritmasının başlangıç çözümü Tablo 44’de verilmiştir.

Tablo 44. Tüm makineler hücrelere atandıktan sonra tavlama benzetimi algoritmasının başlangıç çözümü.

Hücre	Makineler			
1	4,2	2,3	3,1	
2	1,2	3,2	2,1	4,1
3	2,2	1,1	1,3	

Komşu çözümü hesaplamak için operasyon hücreler arası akış hesaplanmalıdır. Bunun için blok diyagonal formda bir hücreler arası akış matrisi yapılandırılır (Tablo 45).

Tablo 45. Başlangıç çözümü için operasyonlar arası akış matrisinin blok diyagonal yapısı.

Hücre	1								2				3				
Makine	4,2			2,3		3,1			1,2	3,2	2,1	4,1	2,2	1,1	1,3		
Kapasite	32,4			23,4		1,9			34	0	30	47	25,3	31	34		
Operasyon	1	3	13	6	7	8	15	16	9	4	16	14	11	2	10	5	12
1						112											
3												123					
13																209	
6		182															
7						227											
8																	56
15															59		
16									43								
9										113							
4													137				
16									203								
14									237								
11				85													
2																224	
10					222												
5																	190
12			101														

En yüksek hücreler arası akışı (222) yaratan operasyon çifti 10,7'dir (blok diyagonal formun dışında yani hücreler arası akış). Bu akışı ortadan kaldırmak için 2,3'de gerçekleştirilen 7. Operasyon 3. hücreye aktarılır. Çünkü makine 2,2'de 2,3'den daha fazla kapasite vardır. Ancak 7. operasyon %38,1 kapasiteye ihtiyaç duyduğundan 7. operasyon 3. hücreye aktarılamaz. Bu nedenle 10. Operasyonun 1. hücreye aktarılması denir. 10. operasyon 2. tipteki makinede %10,9'luk bir kapasiteye ihtiyaç duyar ve 1. Hücredeki 2,3'ün de %23,4'lük kapasitesi vardır. 10. operasyon 1. hücreye aktarılır ve komşu çözüm bulunmuş olur.

Komşu çözümlerin hesaplanmasında bütün hücreler arası akış üreten operasyon çiftlerinin test edilmiş olduğu ve herhangi bir operasyonun diğer bir hücreye

aktarılmasının mümkün olmadığı bir duruma ulaşılabilir. Böyle bir durumda prosedür mevcut çözümün makine hücreler arası akışını hesaplar ve bir makineyi farklı bir hücreye aktararak toplam hücreler arası akışı azaltmayı dener (üretilen hücreler arası akışın azalan sırasında ve bir hücrede izin verilen en fazla makine sayısını aşmadan). Tablo 46’da bunun gibi bir durum için makine hücreler arası akış matrisi verilmiştir.

Makine 2,3 ve 4,1 en yüksek hücreler arası akışı yaratır. 2,3 makinesi 2. hücreye hücredeki en fazla makine sayısını aşacağından aktarılamaz. Ancak 4,1 makinesi 1. hücreye aktarılabilir. Bu değişiklik gerçekleştirilir ve yeni komşu çözüm oluşturulur. Bulunacak bir sonraki komşu çözüm makinelerden çok tekrar operasyonları değiştirmeye çalışacaktır.

Tavlama benzetimi uygulanarak örnek problem için bulunan en iyi çözüm Tablo 47’de verilmiştir.

Tablo 46. Makineler arası akış matrisi blok diyagonal yapısı.

Hücreler	1		2				3			
Makineler	2,3	3,1	1,2	2,1	3,2	4,1	1,1	1,3	2,2	4,2
2,3		286				267				
3,1	286		44					56		112
1,2		44		237	315					
2,1			237			123				
3,2			315			137				
4,1	267			123	137					
1,1								190	224	209
1,3		56					190			101
2,2							224			
4,2		112					209	101		

Tablo 47. Örnek problem için en iyi çözüm.

Hücre	1					2						
Makine	3,1			2,3		4,1			2,1		3,2	1,2
Operasyon	8	15	16	10	15	1	3	11	6	14	16	9
% Operasyon	100	100	18	100	100	100	100	100	100	100	82	100
	3											
Makine	1,1	1,3	2,2	4,2								
Operasyon	5	12	2	13								
% Operasyon	100	100	100	100								

3.3.5 Hücre Oluşturma için Sıra Verili Bir Kümelendirme Algoritması (CASE)

Daha önce 0-1 makine parça matrisini kullanan hiyerarşik olmayan metotlardan GRAFICS metodunu incelemiştik. İlgili yöntem kolon ve satırlardan çekirdek oluşturarak blok diyagonal yapıya ulaşarak kümelendirme yapmaktaydı. Bu bölümde ise yine çekirdek algoritmasını kullanarak bu sefer operasyon sıra verisini probleme dahil eden CASE algoritmasını inceleyeceğiz.

Harhalakis, Nagi ve Proth 1990 yılında yayınladıkları “grup teknolojisi uygulamaları için imalat hücresi oluşturmada etkili bir sezgisel” (An efficient heuristic in manufacturing cell formation for group technology applications) isimli makalelerinde sıra gösteren parça makine etki matrisini girdi olarak kullanmışlar ve iki aşamalı bir prosedür önererek hücreler arası hareketleri minimize etmeye çalışmışlardır. Ancak bu yöntemin bazı sorunları vardır. Öncelikle yöntem hücre içindeki makine ve hücre sayısında bir üst sınır spesifikasyon öncülüne ihtiyaç duyar. Halbuki kümelendirme felsefesinin temeli doğal olarak zaten var olan grupları bulup ortaya çıkarmaktır. Algoritma tersinmezlik gibi katı bir kısıta sahiptir. Herhangi bir aşamada iki makine gruplandığında, bu durum vasat bir kümelendirmeye yol açsa da adımlar tekrar edilemez. Bu da sonraki aşamalarda daha iyi grupların oluşmasını engeller. Ayrıca bu yöntemde makine hücreleri ve parça aileleri aynı anda oluşturulmaz.

3.3.5.1 Tanımlar ve Açıklamalar

Aşağıda algoritmada kullanılan elemanların açıklamaları verilmiştir.

m	makine sayısı (satırlar)
n	parça sayısı (kolonlar)
B	$[b_{jip}]$ parça makine matrisi girdisi
n_{ji}	j . parçanın i . makineyi ziyaret etme sayısı ($N_{ij} \geq 0$)
b_{jip}	operasyon sıra numarası eğer j . parça ($1 \leq j \leq n$) i . makineyi ($1 \leq i \leq m$) p . kez ($0 \leq p \leq n_{ji}$) ziyaret ediyorsa; aksi halde 0
w_j	j parçasının ağırlığı
r_j	j parçası için en büyük operasyon sayısı
$s(i,l)$	i ve l makinesi arasındaki benzerlik

c	en büyük makine hücresi sayısı
$TOTOP_k$	k. hücrede toplam operasyon sayısı
NOP_k	k. hücrede toplam operasyonsuzluk sayısı
β	bağ etkinliği

3.3.5.2 Ağırlıklı Makine Sıra Benzerliği

Önceki benzerlik katsayılarından farklı olarak Nair ve Narendran makine sıralarını temel alan yeni bir makine benzerliği tanımlamıştır. Katsayı i ve l makineleri arasında ortak olan hareket sayısı toplamının, i ve l makinelerine ve makinelerinden olan toplam hareket sayına oranı şeklindedir.

$$s(i, l) = \frac{c_i + c_l}{t_i + t_l} \quad (14)$$

$$t_i = \sum_{j=1}^n \sum_{p=1}^{n_{ji}} w_j t_{jip} \quad (15)$$

$$c_i = \sum_{j=1}^n \sum_{p=1}^{n_{ji}} w_j c_{jip} \quad (16)$$

Burada;

$t_{jip}=0$ eğer $b_{jip}=0$

=1 =1 veya r_j

=2 aksi halde

$c_{jip}=0$ eğer $b_{jip}=0$ veya $b_{jlp}=0$

=1 =1 veya r_j

=2 aksi halde

w_j , j parçasının ilave faktörlerin de kapsandığı ağırlık faktörüdür. (15) eşitliği parçaların yaptığı i makinesine ve makinesinden olan hareketlerin toplam sayısını ifade eder. (16) eşitliği i ve l makinelerini ziyaret eden tüm parçaların i makinesinden

ve i makinesine olan hareketlerin toplam sayısını ifade eder. (14) eşitliği i ve l makineleri için ortak olan hareketlerinin toplamının i ve l makinelerinden ve makinelerine olan hareketlerin toplamına oranı olarak benzerlik katsayısını verir.

3.3.5.3 Bağ Etkinliği

Kümelendirme probleminin çözüm kalitesi iki kritere bağlıdır: hücre içi kompaktlık (hücre üyeleri yüksek benzerliğe sahiptir ve en büyüklenmesi gerekir) ve hücreler arası hareket en küçükleme. Dolayısıyla elde edilen çözümün etkinliğini değerlendirmek için her bir hücrenin kompaktlığı aşağıdaki eşitlik yardımıyla değerlendirilir.

$$Kompaktlık = \frac{\sum_{k=1}^c TOTOP_k}{\sum_{k=1}^c (TOTOP_k + NOP_k)} \quad (17)$$

Mükemmel bir diyagonal blok için

$NOP_k=0$ iken $Kompaktlık=1$ 'dir.

0-1 makine parça verisini kullanan problemlerde kullanılan performans ölçümleri, sıra verisini kullanan problemler için uygun olmadığından Harhalakis ve diğerleri sıra verisi kullanan problemler için “grup teknoloji etkinliği” adı altında bir performans kriteri geliştirmişlerdir (Harhalakis, Nagi, ve Proth, 1990: 185-198). Grup teknoloji etkinliği en büyük mümkün olan hücreler arası hareket sayısı ile sistem tarafından ihtiyaç duyulan güncel hücreler arası hareket sayısı farkının, en büyük mümkün olan hücreler arası hareket sayısına oranıdır.

En büyük mümkün olan hücreler arası hareket sayısı:

$$I = \sum_{j=1}^n (r_j - 1) \quad (18)$$

Sistem tarafından ihtiyaç duyulan güncel hücreler arası hareket sayısı:

$$U = \sum_{j=1}^n \sum_{k=1}^{(r_j-1)} xl_{jk} \quad (19)$$

$x_{lk}=0$ eğer $k, k+1$ operasyonları aynı hücrede gerçekleşiyorsa
 $=1$ aksi halde

$$\text{Grup Teknoloji Etkinliđi} = \frac{(I - U)}{I} \quad (20)$$

Hücreler arası hareket grup teknoloji etkinliđi kullanılarak hesaplanabilir. Hücre oluřturmada tamamen bağımsız hücreler elde etmek sıklıkla birincil amaçtır. Eđer tamamen bağımsız hücreler mümkün deđilse, kısıtlar arasında en büyük bağımsızlık durumu aranır. Bu nedenle, bađ etkinliđi (β), kompaktlık ve grup teknoloji etkinliđinin ağırlıklı ortalaması olarak tanımlanır.

$$\beta = \frac{1(I-U)}{I} + \frac{(1-q) \sum_{k=1}^c TOTOP_k}{\sum_{k=1}^c (TOTOP_k + NOP_k)} \quad (21)$$

Ağırlık faktörü q , problemin spesifik gerekliliklerine göre analizciye kompaktlıđı veya hücreler arası hareketin kuvvetini deđiřtirmeye imkan verir.

3.3.5.4 CASE Algoritmasının Açıklaması

Algoritmanın üç temel aşaması vardır:

1. Objeler (burada makineler) arasındaki benzerlik ölçüsünü tanımlamak,
2. Kümelenendirme için çekirdekleri tanımlama,
3. Kümelenendirme için kriter.

Hiyerarřik olmayan kümelenendirme metotları veri kümesi içinde dođal olarak bulunan kümeleri bulmayı amaçlar. Bunu makine kümelenendirme anlamında ifade edersek bazı makineler işledikleri parçalar açısından yüksek bir benzerliđe sahiptir. Bazı makinelerse tamamen birbirinden farklıdır dolayısıyla işledikleri parçalar tamamen farklıdır. İşte bu tür makineler kümelenendirme için potansiyel çekirdek makinelerdir ve centroid olarak adlandırılırlar. Çekirdek makineler tanımlandıktan sonra benzerliklerine göre diđer makineler bu çekirdek makinelerin etrafında kümelenendirilirler.

Yukarıda verilen (14) eřitliđine göre makineler arası benzerlik katsayıları hesaplanır. Bu aşamada benzerlik eřik deđerı 0'a eřitlenir. Yukarıda belirtildiđi gibi,

potansiyel çekirdek makineler aralarında benzerliğin en az olduğu makinelerdir. Bu benzerlik eşiğinde çekirdek makineler aranır. Birden fazla aynı benzerlik değerine sahip makine bulunursa, bunlardan biri rastgele seçilir. Bundan sonra en büyük benzerlik değerine sahip makinelerin çekirdek makinelere atanmasıyla hücreler oluşturulur. Eğer herhangi bir çekirdek makine kendi hücrelerine daha fazla makine çekemezse (singleton) aşağıdaki prosedür benimsenir.

1. Singleton ve her bir küme arasındaki ortalama benzerlik hesaplanır.
2. Singleton yukarıdaki değerlerden en büyüğüne sahip olan kümeye atanır.

Böylece elde edilen her makine hücresi, parçaları kümelendirmek için bir çekirdek sağlar. Örnek olarak, eğer 5 makineli bir problem 2 makine hücresi oluşturuyorsa: $MC_1=\{1,4\}$ ve $MC_2=\{2,3,5\}$ ise parça çekirdekleri $CS_1=\{10010\}$ ve $CS_2=\{01101\}$. Böylece c makine hücrelerinin oluşturulması c m-boyutlu parça-çekirdek vektörlerini oluşturur. Her parça bir m boyutlu uzayda vektör olarak görülür. Parçalar en küçük hücreler arası taşımayı sağlama kriteri temelinde makine hücrelerine (parça çekirdekleri) atanır. Bundan dolayı bir parça kümeleme kriteri aşağıdaki gibi tanımlanır:

1. Her bir parça vektörü ($P_j=\{P_{1j}P_{2j}\dots P_{mj}\}$), $1\leq j\leq n$, ile parça çekirdek vektörü ($CS_k=\{CS_{1k}CS_{2k}\dots CS_{ik}\dots CS_{mk}\}$) $1\leq k\leq c$ hareket sayısı (sc_{jk}) hesaplanır.

$$sc_{kj} = w_j \sum_{i=1}^n X_i$$

(22)

Burada

$X_i = 0$, (eğer P_{ij} veya CS_{ik} sıfır ise)

$= 1$, (eğer $P_{ij} = 1$ veya r_j , ve $CS_{ik} = 1$ ise)

$= 2$, (eğer $P_{ij} = 1$ değil veya r_j , ve $CS_{ik} = 1$ ise)

2. En fazla hareket sayısına sahip parça vektörünü parça çekirdeğine ata. Eşitlik durumunda en küçük 1'lerin sayısına sahip çekirdek lehinde çöz.

Bu çözüm, kendisinin bağ etkinliği ile değerlendirilir. Prosedür eşik değerinin benzerlik katsayıları matrisindeki bir sonraki en küçük değere arttırılmasıyla tekrar ettirilir. Böylece ilave centroid'ler tanımlanabilir. Bu centroid'ler mevcut kümeye eklenir. Prosedür, ardışık çözümlerin bağ etkinliğinin aynı kalması veya azalması durumuna kadar devam ettirilir. Aşağıda algoritmanın akışı verilmiştir.

Adım 0: Veriyi oku.

Adım 1: $TAL=0$, $\beta=0$.

Adım 2: Her bir i ve l makine çifti arasındaki $s(i,l)$ benzerliklerini hesapla ve benzerlik matrisini oluştur.

Adım 3: Centroid'leri tanımla.

Adım 4: Makineleri kümelendirerek hücreleri oluştur.

Adım 5: Parçaları makine hücrelerine atayarak parça ailelerini oluştur.

Adım 6: Çözümü depola ve bağ etkinliğini hesapla.

Adım 7: Eğer yeni çözümün bağ enerjisi mevcut en iyiden daha büyükse eşik değerini bir sonraki yüksek benzerlik değerine arttır ve 3. Adıma git.

Adım 8: En iyi çözümü ve ilgili bağ etkinliğini yazdır.

Adım 9: Dur

3.3.5.5 Sıra Verili Hücre Oluşturma İçin Bir Uygulama

Algoritma 8 makine ve 20 parçadan oluşan bir örnek üzerinde gösterilmiştir (Tablo 48). Parçalara verilen ağırlıklar eşit ve ilgili makine parça matrisinde ayrı bir kolon olarak verilmiştir.

Eşitlik (14)'e göre hesaplanan makineler arası benzerlik matrisi Tablo 49'daki gibidir.

Benzerlik katsayıları hesaplandıktan sonra benzerlik katsayı eşiği 0'a eşitlenir. Bu eşik değerinde matris taranır. Bu değerde makine 4-3 ve makine 8-3 vardır. İki makine çiftinden biri rastgele seçilir. Makine 4 ve 3'ü centroid olarak belirleyelim. Bundan sonra 4 ve 3 centroid'lerine en büyük benzerlik katsayısına sahip makineler bağlanır.

Tablo 48. CASE algoritması örneği için operasyon sıra verili makine parça matrisi.

		MAKİNELER								AĞIRLIK
		1	2	3	4	5	6	7	8	
PARÇALAR	1					2	1			1
	2	1		2						1
	3	2	1		5			3	4	1
	4		1		2			3	4	1
	5					2	1			1
	6		1		2	5		3	4	1
	7		4		2			3	1	1
	8	1		2						1
	9	1		3			2			1
	10				2	3	1			1
	11	3		2				1		1
	12					1	3	2		1
	13	1		2						1
	14	1	2	3						1
	15				1	2				1
	16	1		2						1
	17	3		1		2				1
	18		2		1			4	3	1
	19	1		2						1
	20		2		1		3	4	5	1

Tablo 49. CASE algoritmasında benzerlik katsayıları matrisi.

	1	2	3	4	5	6	7
2	0,286						
3	0,905	0,150					
4	0,130	0,773	0,000				
5	0,158	0,111	0,167	0,400			
6	0,158	0,222	0,167	0,300	0,500		
7	0,240	0,792	0,125	0,769	0,273	0,318	
8	0,200	0,895	0,000	0,714	0,176	0,176	0,870

Makine 3 → 0,905 benzerlikle makine 1.

Makine 4 $\rightarrow$ 0,9 benzerlikle makine 8, 0,81 benzerlikle makine 2, 0,8 benzerlikle makine 7, 0,316 benzerlikle makine 5 ve 0,3 benzerlikle makine 6.

Makine hücreleri $MC_1=\{1,3\}$ ve $MC_2=\{2,4,5,6,7,8\}$ olduğuna göre, parça çekirdekleri $CS_1=\{10100000\}$ ve $CS_2=\{01011111\}$ 'dir.

Bundan sonra her parçanın oluşan hücelere göre sc_{kj} hareket sayısı hesaplanır.

1. hücre 1. parça için sc_{11} :

$P_1 = \{0\ 0\ 0\ 0\ 2\ 1\ 0\ 0\}$ parça vektörü

$CS_1 = \{1\ 0\ 1\ 0\ 0\ 0\ 0\ 0\}$ parça çekirdek vektörü

$sc_{11} = 0+0+0+0+0+0+0+0$

$=0$

$$\left\{ \begin{array}{l} X_i = 0, \text{ (eğer } P_{ij} \text{ veya } CS_{ik} \text{ sıfır ise)} \\ = 1, \text{ (eğer } P_{ij} \text{ 1 veya } r_j, \text{ ve } CS_{ik} \text{ 1 ise)} \\ = 2, \text{ (eğer } P_{ij} \text{ 1 değil veya } r_j, \text{ ve } CS_{ik} \\ \text{1 ise)} \end{array} \right.$$

2. hücre 1. parça için sc_{21} :

$P_1 = \{0\ 0\ 0\ 0\ 2\ 1\ 0\ 0\}$ parça vektörü

$CS_2 = \{0\ 1\ 0\ 1\ 1\ 1\ 1\ 1\}$ parça çekirdek vektörü

$sc_{21} = 0+0+0+0+1+1+0+0$

$=2$

$Sc_{11} \leq sc_{21}$ olduğundan 1 nolu parça 2 nolu hücreye atanır.

Sistemdeki tüm parçaların ataması gerçekleştirildikten sonra bağ etkinliği hesaplanır.

1. iterasyondan sonra makine parça matrisinin durumu Tablo 50'de görülmektedir.

1. iterasyon sonucu grup teknoloji etkinliği (20) eşitliğinden:

Grup teknoloji etkinliği $= (41-9)/41 = 0,78$ 'dir.

$TOTOP_1 = 19$, $TOTOP_2 = 42$, $NOP_1 = 8$, $NOP_2 = 19$

(17) nolu eşitlikten kompaktlık $= 0,92$ olarak hesaplanır.

Bağ etkinliği (β) (21) eşitliğinden q 'a bir değer verilerek hesaplanır.

Tablo 50. CASE algoritması örneğinde 1. iterasyon sonucu makine parça matrisi ve grup teknoloji etkinliği eleman değerleri.

		Makineler								r_{j-1}	U_j	
		3	1	4	2	5	6	7	8			
Parçalar	2	2	1							1	0	
	8	2	1							1	0	
	9	3	1				2			2	2	
	11	2	3					1		2	1	
	13	2	1							1	0	
	14	3	1		2					2	2	
	16	2	1							1	0	
	17	1	3			2				2	2	
	19	2	1							1	0	
	1					2	1			1	0	
	3		2	5	1			3	4	4	2	
	4			2	1			3	4	3	0	
	5					2	1			1	0	
	6			2	1	5		3	4	4	0	
	7			2	4			3	1	3	0	
	10			2		3	1			2	0	
	12					1	3	2		2	0	
	15					1	2			1	0	
	18			1	2			4	3	3	0	
	20			1	2		3	4	5	4	0	
									I=	41	9	=U

3.3.6 Genetik Algoritmaları Kullanan Metotlar

3.3.6.1 Genetik Algoritma

Genetik algoritmalar (GA), doğada yaşayan bireylerdeki evrimsel süreci taklit eden bir arama ve en iyileme yöntemidir. Bilindiği gibi bu evrimsel süreçte iyi özelliklere sahip olan bireyler hayatta kalırken, kötü özelliklere sahip olan bireyler elenirler. İşte bu sürecin özellikle matematiksel modelleme yapılamayan veya kesin çözümün olmadığı problemlere uygulanarak en iyi çözümlerden oluşan bir küme oluşturularak en iyiye yakın çözümün bulunacağı fikri ilk defa 1975 yılında John Holland tarafından yayınlanan “Doğal ve Yapay Sistemlerde Uyum” isimli kitapta ortaya kondu. Holland evrimden ve canlılardaki bu süreçten yararlanarak, makine

öğrenmesi üzerinde yoğunlaştırmıştır. Bu süreci bilgisayar ortamına taşıyarak tek bir mekanik yapının öğrenme yeteneğini geliştirmek yerine böyle yapılardan oluşan topluluğun çiftleşme, çoğalma ve değişim gibi genetik süreçlerden geçerek başarılı yeni bireyler oluşturabildiğini görmüştür. Goldberg, GA konusunun temeli sayılan bir kitap yayınlayarak GA'nın dünyanın her yerinde farklı konularda kullanılabileceğini göstermiştir (Elmas, 2007: 388-401).

GA için söz konusu olan süreç, genetik yapıdaki doğal seçim sürecidir. GA'nın temel prensibi rastgele belirlenmiş büyüklükte, rastgele olarak üretilen bir popülasyondan başlayarak, genetik operatörleri kullanarak yeni bir popülasyon elde etmektir. Bu süreç tatmin edici sonuç elde edilene kadar tekrarlanır (Al-Sultan ve Fedjki, 1997: 788-796).

Genetik algoritmalar, temelde bir çözüm uzayında arama yapan sezgisel bir yaklaşımdır. Özellikle tek bir noktadan değil rassal olarak türetilmiş olan bir grup çözüm ile birçok noktadan arama yaptığı için yerel en iyilere takılma riski çok düşüktür. En iyi çözüm garanti edilmese de hızlı bir şekilde en iyiye çok yakın çözümlere ulaşılabilir (Saraç ve Özçelik, 2007: 22-36). Bu durum P tipi olmayan yani polinomal zamanda çözülemeyen problemler için en iyiye yakın çözümü kısa zamanda aramada etkindir.

3.3.6.2 Genetik Algoritmanın Genel Prensipleri

GA uygulamasında öncelikle problemin çözümünü temsil eden bir kromozom yapısı tanımlanır. Bu yapı kullanılarak rastgele ya da verilen bir sezgisel yöntemle başlangıç çözüm popülasyonu üretilir. Daha sonra popülasyon üyeleri uygunluk fonksiyonu olarak adlandırılan değerlendirme fonksiyonu temelinde seçilir. Uygunluk değeri en yüksek olan üyeler seçilmek için daha fazla şansa sahiptirler. Böylece daha az uygunluğa sahip bireyler, daha iyi performansla sahip bireylerle yer değiştirirler. Daha sonra seçilen bireylere, yeni popülasyon neslini üretmek için genetik operatörler uygulanır. Bu süreç belli sayıda iterasyona ulaşılan kadar tekrarlanır.

GA'nın yürütülmesi aşağıdaki faktörlerin tanımlanmasını gerektirir (Boulif ve Atif, 2006: 2219-2245):

1. Çözümleri temsil eden genetik kod yapısı.

2. Başlangıç popülasyonunu üretmek için metot.
3. Popülasyonun bireylerinin uygunluğunu değerlendirmek için uygunluk fonksiyonu.
4. Yeni nesli üretmek için kullanılacak genetik operatörler.
5. Kesin kontrol parametre değerleri (popülasyon büyüklüğü, iterasyon sayısı, genetik operatör olasılıkları v.b.)

3.3.6.3 Genetik Kod Yapısı

Problemde bireyleri temsil edecek olan kromozomları kodlama şeklinin belirlenmesi problemin çözümüne direk etkisi olan önemli bir kriterdir. Bireyleri ifade eden kromozomlar genlerden oluşur. Diğer bir deyişle genler, kromozomun yapısında kendi başına birer genetik bilgi taşıyan en ufak yapı birimidir. Genlerin bir araya gelmesiyle de bütün bir çözüm kümesini oluşturan kromozom meydana gelir. Bir genin içerdiği bilgi sadece ikili tabandaki sayıları içerebileceği gibi onluk ve onaltılık tabandaki bilgileri de içerebilir. Problemin çözümünü temsil edecek bireylerin kromozom yapılarının belirlenmesinde çeşitli yöntemler olmasına karşın bunlardan en çok kullanılanı tamsayı kod yapısıdır. Kromozom içindeki genlerden her birinin bir makineyi temsil ettiğini varsayalım. Genler ise makinenin atandığı hücreyi gösterir. Örneğin 6 makine ve 3 hücreden oluşan bir çözüme ait kromozom zinciri aşağıdaki şekilde gösterilmiştir. Burada M_1 , M_3 ve M_5 makineleri 1. hücreye, M_2 , ve M_6 makineleri 2. hücreye ve M_4 makinesinin ise 3. hücreye atandığı görülmektedir.

Tablo 51. 6 makinenin 3 hücreye atandığı çözümün genetik zincirine örnek (Al-Sultan ve Fedjki, 1997: 788-796).

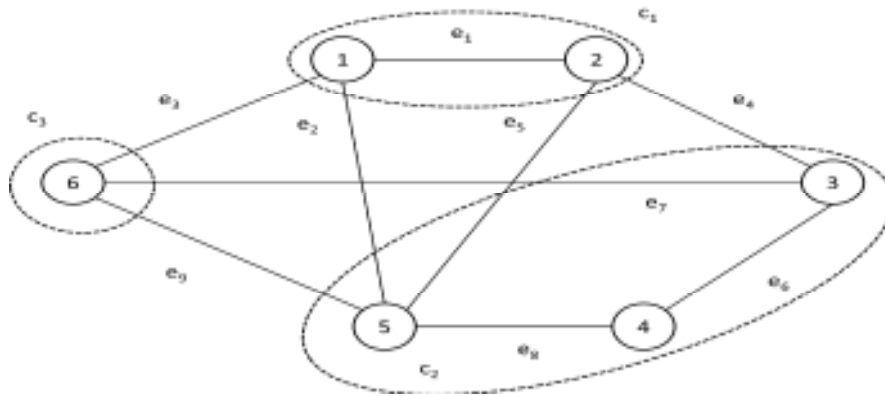
M_1	M_2	M_3	M_4	M_5	M_6
1	2	1	3	1	2

Diğer bir kromozom kodlama yapısı ise permütasyon kodlamadır. T. Kesintürk ve B. Küçük “Karışık modellenli montaj hatlarının genetik algoritma kullanılarak dengelenmesi” isimli makalelerinde her bir genin bir işi, kromozomun ise iş istasyonlarını ve bu istasyonlarda yer alan işlerin tamamını temsil ettiği bir kodlama yapısı kullanmışlardır (Şekil 13).

1. İST	1	6	4	8	9	10	11	12	15	16	17	21	22	23	24	25	26	27	32	43	20	30
2. İST	28	41	13	14	42	2	3	39	31	33	37	18	38	45	46	47	19	44	7			
3. İST	82	85	86	89	90	92	93	94	83	95	96	97	98	99	100	105	84	87	101	20	40	102
4. İST	49	68	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66			
5. İST	67	69	70	71	72	73	74	75	76	77	78	79	80	81								

Şekil 13. Permutasyon kodlamanın kullanıldığı kromozom (Keskindürk ve Küçük, 2006: 52-63).

Boulif ve Atif “İmalat hücresi oluşturma problemi için dal sınır geliştirilmiş yeni bir genetik algoritma” (A new branch & bound enhanced genetic algorithm for the manufacturing cell formation problem) isimli makalelerinde ikili kod kullanmışlardır. Ancak problem modeli çizge bölme kuramı temeline dayandığından kromozom içindeki genler bir $G=(M,E)$ akış çizgesi içindeki E kenarlara ayrılmıştır. Genler ise çizge bölme çözümüne göre bölme içinde kalan kenarlar sıfırla, bölme dışında kalan kenarlar birle gösterilir. Diğer bir deyişle hücre içi malzeme akış trafiği sıfırla, hücreler arası malzeme akış trafiği birle gösterilir. Örneğin bir 6 makine için bir akış çizgesi ve bunun 3 hücreli çizge bölme çözümü olsun.



Şekil 14. 6 makine için 3 hücre çözümlü akış çizgesi (Boulif ve Atif, 2006: 2219-2245).

Tablo 52. 3 hücreli ve 9 kenarlı çizge için ikili kod (Boulif ve Atif, 2006: 2219-2245).

e ₁	e ₂	e ₃	e ₄	e ₅	e ₆	e ₇	e ₈	e ₉
0	1	1	1	1	0	1	0	1

İkili kodun en büyük avantajı GA'nın azaltılmış karakterlere olumlu yönde duyarlı olmasıdır. İkili karakterlerle GA'nın bireylere ait kodların iyi bloklarını tespit etmesi kolaydır. Diğer bir avantaj ise popülasyon kromozomlarını temsil edilebilmesi için boolean değişkenlerinin kullanılabilmesi nedeniyle algoritmanın yürütülmesinde göreceli olarak daha az hafıza gereksinimidir.

3.3.6.4 Başlangıç Popülasyonunun oluşturulması

GA'nın rastgele seçilmiş kromozomlardan oluşan başlangıç popülasyonlarında bile iyi çözümler sağladığı ispatlanmıştır (Boulif ve Atif, 2006: 2219-2245). Bu nedenle GA algoritmalarında genellikle başlangıç popülasyonu rastgele olarak oluşturulur. Yine de rastgele seçim bazı kısıtlı problemlerde aramanın uygun olmayan çözümlere doğru yönelmesine sebep olabilir. Bu durumu ortadan kaldırmak için probleme özgü çeşitli sezgisel yöntemler geliştirilebilir.

Popülasyonda bulunacak birey sayısı önemlidir. Bu sayının belirlenmesinde herhangi bir yöntem yoktur ancak problem çözücünün buradaki deneyimi ön plana çıkar. Popülasyon büyüklüğü arama uzayında iyi bir arama yapılabilmesini sağlayacak kadar büyük, hesaplama zamanını fazla arttırmayacak kadar küçük olmalıdır.

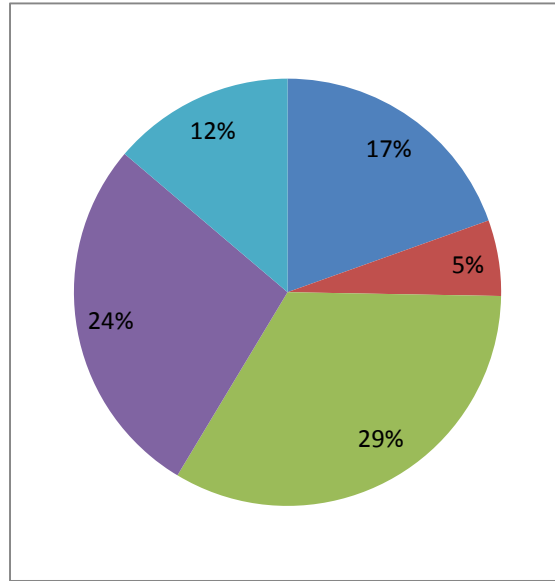
3.3.6.5 Uygunluk Değeri

Oluşturulan popülasyonda eşleşecek olan bireylerin başarı derecelerini gösteren değerlendirme fonksiyonudur. Bir sonraki nesile aktarılabilecek olan bireyler ve yok olacak bireyler bu değerlendirme fonksiyonuna göre yapılır. Literatürde bu değerlendirmeler genellikle ilgili modelde geliştirilen amaç fonksiyonu kullanılarak yapılır. Bununla birlikte GA operatörlerinin uygulanması nedeniyle uygulanabilir olmayan bir çözümle karşılaşıldığında ya uygulanabilir olması için yeniden ayarlanır, ya da basitçe popülasyondan çıkarılır.

3.3.6.6 Genetik Operatörler

Başlangıç popülasyonundan yeni bireyler elde etmek, popülasyona bir takım genetik operatörlerin uygulanmasıyla mümkün olur. Aşağıda bu genetik operatörler verilmiştir.

Seçim (reproducing): Bir sonraki nesile aktarılacak olan bireyleri belirler. Literatürde en çok kullanılan seçim yöntemi rulet tekerleğidir. Uygunluk değerleri hesaplanan bireylerin aldıkları değerlere göre bir rulet tekerleği oluşturulur ve tekerlek bu değerlerin büyüklüğüne göre bölünür. Seçim için rulet tekerleği çevrilir ve bunun için sıfırla toplam uygunluk arasında rastgele bir değer üretilerek bu sayının tekerleğin hangi bölümüne karşılık geldiğine bakılarak birey seçilir. Literatürde rulet tekerleği yönteminin bazı olumsuz yanlarını ortadan kaldırmak için diğer bazı metotlar geliştirilmiştir. Bunlar: orantılı seçim, sıralı seçim, turnuva seçim, denge durumu seçim yöntemleridir (Elmas, 2007: 388-401).



Şekil 15. Rulet tekerleği seçme operatörü.

Çaprazlama (crossover): Seçilen bireyler bundan sonra çaprazlama işlemine tabi tutulurlar. Bunun için her bir kromozomun genleri rastgele olarak bölünür ve baş ve kafa bölümü oluşturulur. Daha sonra kromozomların kuyruk bölümleri kendi içlerinde yer değiştirilir (Şekil 16).



Şekil 16. GA'da çaprazlama operatörünün uygulanması (Al-Sultan ve Fedjki, 1997: 788-796).

Mutasyon: Bu operatör ebeveyn kromozomlarında rastgele seçilen genlerin kromozom içinde yer değiştirmesi ile yapılır.



Şekil 17 GA'da mutasyon operatörünün uygulanması

3.3.6.7 Dal Sınır Yöntemleri

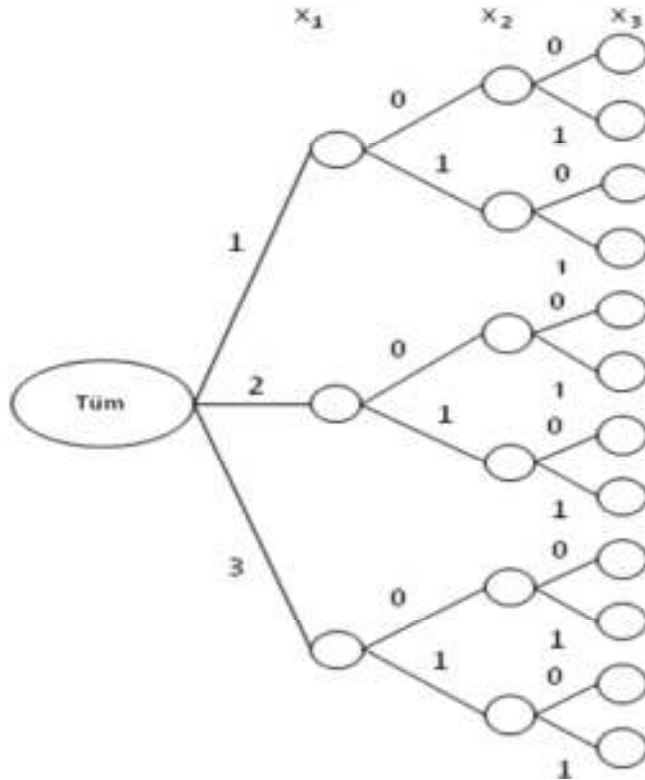
Sezgisel arama yöntemlerinden dal sınır yöntemleri 1950’li yıllarda A. H. Land ve A. G. Doig tarafından ilk kez tam ve karma tamsayılı programlama için geliştirilmiştir ve matematiksel programlama problemlerinin en iyi çözümlerinin belirlenmesi ile ilgili birçok yöneylem araştırması problemine uygulanabilmektedir. Dal sınır yöntemi bir sorunla ilgili türlü aşamaları sistemli bir şekilde analiz ederek en iyi çözümü araştırır (Sezen, 2007: 77-111).

NP optimizasyon problemlerin çözümlerinin aranmasında kullanılan dal sınır yöntemi en iyi çözümü bulmak için tüm çözüm uzayını arayan etkili bir yöntemdir. Yöntem, ağaç yapısına sahip tamsayılı çözümler sayımın gözlenmesi temeline dayanır. Örneğin genel tamsayılı değişken olan x_1 ve ikili değişken olan x_2, x_3 ’e sahip bir modelin toplam sayımını düşünelim. Burada değişkelerin aralıkları $1 \leq x_1 \leq 3$, $0 \leq x_2 \leq 1$ ve $0 \leq x_3 \leq 1$ olsun. Bu durumda modeldeki diğer kısıtlar nedeniyle değişkenler için çözümlerin toplam sayımı Şekil 18’deki gibi olur. Görüldüğü gibi çözüm aramada kullanılan yapı ağaca benzetilmektedir. Sol taraftaki “tüm çözümler” ağacın gövdesini, sağ taraftaki alt küme gösterimleri de ağacın dallarına benzetilebilir. Dallardaki düğümler güncel sayılmış çözümlerdir. 12 adet çözüm vardır, bunlardan her bir dalda x_1 için 3 mümkün değer, x_2 için 2 mümkün değer ve x_3 için 2 mümkün değer sayılmıştır. Örneğin sağ üst köşedeki düğüm $x_1=1, x_2=0$ ve $x_3=0$ çözümünü

temsil eder. Diğer düğümler de mümkün çözümler kümelerinin temsili olarak düşünülebilir.

Dal sınır yönteminde iki temel noktaya dikkat edilmesi gerekir. İterasyonlar boyunca ağacın tüm dalları dallandırılmaz çünkü gerçek problemler için ağaç çok büyük boyutta olabilir. Bunun yerine çözüme en yakın dalların iterasyonlar boyunca ilerletilmesi yapılır. İkinci konu ise budama fonksiyonudur. Bu da uygulanabilir veya optimal olmayan dallara uygulanır ve bu dallar kesilerek daha fazla ilerletilmesine izin verilmez.

Çözüm ağacındaki filiz düğümler uygulanabilir veya uygulanabilir olmayan kısmi bir çözümdür ve daha fazla dallanmaya adaydır. Sınırlama fonksiyonu ise en iyileme tahmincisidir. Amaç fonksiyonunun en iyi değerinin tahmin edilmesi metodu, bir filiz düğümün daha fazla ilerletilmesiyle elde edilebilir. Sadece filiz düğümler sınırlama fonksiyonu ile birleştirilmiştir. Dallardaki düğümler güncel değerler olan ve tahmin olmayan amaç fonksiyonu değerlerine sahiptir.



Şekil 18. Dal Sınır Yönteminde sayım ağacı.

Ağacın dallandırılması sonraki değişkenin her mümkün değeri için bir çocuk düğüm yaratılarak yapılır. Örneğin bir sonraki değişken ikili değer ise, bir sıfır değerli ve bir 1 değerli iki çocuk düğüm yaratılacaktır.

Düğüm seçme politikası, dallanma için bir sonraki düğümün nasıl seçileceğini belirler. Düğüm seçimi için 3 popüler politika vardır:

1. En iyi-birinci veya global-en iyi düğüm seçimi: Dal sınır ağacının herhangi bir yerinde sınırlama fonksiyonunun en iyi değerine sahip filiz düğümün seçilmesidir. En küçükleme yapıyorsak bunun anlamı sınırlama fonksiyonunun en küçük değerine sahip filiz düğümü seçmektir. En büyüklemede ise bunun tersidir.
2. Derin-birinci: Sadece yaratılmış olan filiz düğümler seti arasından seçim yapma. Sınırlama fonksiyonunun en iyi değerli filiz düğümü seçilir. Derin-birinci düğüm seçimi her iterasyonda dal sınır ağacının bir adım derinine götürür. Böylece dallardaki düğümlere çabuk bir şekilde ulaşılır.
3. Genişlik-birinci: Yaratılmış olan filiz düğümler aynı sırada genişletilir.

Benzer şekilde dallanma için bir filiz düğüm seçildiğinde, bu filizin çocuk düğümlerinin yaratılması için kullanılacak olan bir sonraki değişkenin seçim yöntemi de önemlidir. Bu seçimi sağlayan ise değişken seçim politikasıdır. Bunun için birkaç standart seçim politikası vardır. Değişkenler genellikle kendi doğal sıralarında seçilirler.

Bir diğer kural ise budama kuralıdır. Bir düğümü budamak için iki ana neden vardır: çıkacak dalın uygulanabilir olmadığını göstermek ya da optimal olmadığını göstermektir.

Çıkacak dalın optimal olmadığını göstermek standarttır. Eğer filiz düğüm sınırlama fonksiyon değeri incumbent için amaç fonksiyon değerinden daha kötü ise, filiz düğüm budanabilir. Bunun nedeni sınırlama fonksiyonunun değeri iyimser tahminci olmasıdır. En büyükleme yaptığınızı varsayın ve incumbent amaç fonksiyon değeri 87. Eğer iyimser sınırlama fonksiyonu sadece 79 değerine sahip filiz düğümü için abartılıysa filiz düğümden çıkan dallar 79 değerini geçmeyeceğini bilirsiniz ve böylece filiz düğüm budanır.

Son olarak, dallandırma prosedürü, incumbent çözümünün amaç fonksiyon değeri tüm filiz düğümlerin sınırlama fonksiyonu değerine eşit veya daha iyi ise durdurulur. Bu, filiz düğümlerin hiçbiri, elimizdeki uygulanabilir çözümden daha iyi bir çözüm geliştiremeyeceği anlamına gelir.

3.3.7 Operasyon Sıra Verisini Kullanan Melez Sezgisel Algoritmalar

3.3.7.1 NP Tam Problemler

Polinomal (P) denklemler incelenmesi kolay olan problemlerdir ve kısa sürede sorunu çözen yöntemleri mevcuttur. Eğer bir denklem veya sistem polinomal değilse (NP; nonpolynomially bounded) çözümlemesi zor sistemlerdir (Elmas, 2007: 388-401). P tipi problemler etkin şekilde çözülebilirken, NP problemler algoritmik olarak pratik bir şekilde çözülemezler çünkü çözümü bulmak için herhangi bir algoritma katlanarak artan bir uzun süreye ihtiyaç duyar ve problemi çözemez. Burada NP, P olamayan demek değildir. Eğer problem için önerilen çözümün gerçekten de çözüm olup olmadığını polinom zamanda kontrol edebiliyorsak o zaman bu NP tipi problemdir. Bu, çözümü polinom zamanda bulmaya kıyasla daha daha gevşek bir koşuldur. Örneğin yapboz tipi bulmacayı çözmek çok zor olabilir ama eğer birisi yapbozu çözdüğünü iddia ederse bunun doğru olup olmadığını kontrol etmek kolaydır. Çalışma zamanına ilişkin sayısal fikir edinmek isterseniz sırayla her parçaya bakın ve etrafındaki sınırlı sayıda komşusu ile uyuyup uyuşmadığını kontrol edin. Bunu yapmanız için gerekli olan hesaplama sayısı kabaca elinizdeki parça sayısı ile doğru orantılıdır, dolayısıyla çözümün kontrolü polinom zamanda gerçekleşir. Ancak yapbozun kendisi bu şekilde çözülmez. Her çözümü sıra ile deneyemezsiniz çünkü potansiyel çözümlerin sayısı yapbozu oluşturan parçaların sayısının sabit herhangi bir kuvvetinden çok daha hızlı artar.

Yukarıdaki açıklamalardan da anlaşılacağı gibi NP problemlerin çözümünde kesin sonuç yerine yakın çözümler tercih edilir. Bu tip problemlerin kesin çözümlerine makul sürelerde ulaşamadığından; yerel arama ve rastlantısal arama ile yaklaşık çözümler elde edilir. GA, sezgisel bir yöntem olduğundan dolayı verilen bir problem için kesin sonucu bulamayabilir, ancak bilinen yöntemlerle çözülemeyen yada çözüm zamanı problemin büyüklüğü ile üstel olarak artan problemlerde kesin

sonuca çok yakın çözümler verebilir. Çözüm arama tekniklerinden GA dışındaki rastgele arama tekniklerine örnek olarak tavlama benzetimi (simulated annealing), yasaklı arama (tabu search) ve karınca kolonileri (ant colony) verilebilir. Hücresel imalat uygulamaları dışında GA, gezgin satıcı, karesel atama, yerleşim, atölye çizelgeleme gibi optimizasyon problemlerine uygulandığı gibi, otomatik programlama ve bilgi sistemleri, mekanik öğrenme ve finans ve pazarlama alanlarına da uygulamaktadır.

3.3.8 Hücre Oluşturma Problemi İçin Dal-Sınır'la İyileştirilmiş Genetik Algoritma

M. Boulif ve K. Atif'in 2006 yılında yaptıkları çalışmada, NP tam olan hücre oluşturma probleminin çözümü için kullanılan meta-sezgisel metotların birlikte kullanılmasının daha iyi sonuçlar vermesinden yola çıkarak, problemi çizge bölme temelinde modellemişler ve hücreler arası trafiği dal-sınır yöntemi ile iyileştirilmiş bir genetik algoritma ile en küçükmeye çalışmışlardır. Bunun için ilk olarak problem ikili kodlama sistemi kullanılarak genetik algoritma ile çözülmüştür. Daha sonra genetik algoritma performansı, dal-sınır metodu ile iyileştirilmiştir.

3.3.8.1 Problemin Tanımı

Makine hücresi oluşturma problemi bilindiği gibi makinelerin hücelere ve bu parçalarında parça ailelerine kümelenmesinden oluşur. Üretim maliyetini en küçükmek için bu kümelendirme aşağıdaki kısıtları gerektirir:

1. Hücre büyüklüğü için üst sınır,
2. Hücre sayısı için alt ve üst sınırlar,
3. Aynı hücreye yerleştirilmesi zorunlu olan makine kısıtı,
4. Farklı hücelere yerleştirilmesi zorunlu olan makineler kısıtı.

Problem modelinde hücreler arası hareketler öncelikle gözönünde bulundurulmuştur. Problem girdileri arasında operasyon sıralarının yanısıra üretim hacimleri de kullanılmıştır.

3.3.8.2 Matematik Model

Problemin modellenmesinde daha önce belirtildiği gibi çizge teorisi temelli model kullanılmıştır.

Girdi Verisi:

1. $M=\{M_1, M_2, \dots, M_m\}$ makineler kümesi ve $P=\{P_1, P_2, \dots, P_p\}$ parça tipleri kümesidir. P parçaları M makinelerinde işlem görürler.

2. Her bir P_k ($k=1, \dots, p$) parça tipi için;

(i) Parça tarafından ziyaret edilen makinelerin tekli bir sırası (veya rota)

$R_k = (M_{k,1}, M_{k,2}, \dots, M_{k,s_k})$, burada $M_{k,j} \in M$ ($j=1, 2, \dots, s_k$) ve s_k , P_k rotasındaki makine sayısıdır.

(ii) r_k : P_k parça tipinin birim zamandaki ortalama üretim oranıdır.

3. Her bir $(P_k, M_i, M_j) \in P \times M \times M$, $v_{k,i,j}$, P_k rotasında M_i 'nin M_j 'yi ($i, j=1, \dots, m$) (veya tersi) izleme sayısı olarak ifade edilir.

4. Her bir çift $(M_i, M_j) \in M \times M$ için, M_i, M_j ($i, j=1, \dots, m$) hücreler arası trafiği aşağıdaki şekilde ifade edilir:

$$t_{ij} = \sum_{k=1}^p r_k x v_{k,i,j} \quad (23)$$

Daha sonra bir $G=(M,E)$ yönsüz çizge tanımlanır⁵. Burada M kümesi makineler kümesi ve E kümesi pozitif trafikle (akış) bağlı sıralanmamış makine çiftleri kümesidir.

$$E=\{e_{i,j}/(M_i, M_j) \in M \times M, i, j=1, \dots, m; i \neq j \text{ ve } t_{ij} \neq 0\}$$

5. $C=\{C_1, C_2, \dots, C_J\}$ J hücrelerinde M bölüntüsü olsun, öyleki:

$$\bigcup_{i=1}^J C_i = M \text{ ve } C_i \cap C_j = \emptyset \quad \forall i, j \in \{1, 2, \dots, J\}, i \neq j$$

6. $E(C)$, C bölüntüsüyle ilgili hücreler arası trafiğin alt kümesi olarak tanımlanır:

$$E(C)=\{e_{kl} \in E/(M_k, M_l) \in C_i \times C_j; i, j=1, 2, \dots, J; i \neq j \text{ ve } k, l=1, 2, \dots, m\}.$$

⁵ Hesaplama karmaşıklığı nedeniyle çözüm metodlarının çoğu temel olarak yaklaşıklık temelli prosedürlerdir. Özellikle düğümlerin makineleri temsil ettiği ve iki düğümün bir kenarla bağlı olduğu çizge yaklaşımları, hücre oluşturma problemlerini çözmek için önerilmişlerdir (Lee & Chiang, 2001: 3491-3512).

7. Bir kenar ağırlık fonksiyonu W , iki makine arasındaki trafiği göstermek için tanımlanır: $W(e_{ij})=t_{ij}$, burada $i,j=1,\dots,m$ 'dir.

8. C bölüntüsünün toplam hücreler arası trafiği:

$$T(C) = \sum_{e_{kl} \in E(C)} W(e_{kl}) \quad (24)$$

Kısıtlar:

C, yukarıda verilen kısıtlara da uyan bir M bölüntüsü olsun.

1. Bir hücre içinde izin verilen en fazla makine sayısı N,

$$\forall C_i \in C (i = 1, 2, \dots, J), \text{Card}(C_i) \leq N,$$

Burada Card(C_i), C_i hücresinin kardinalidir (C_i atanmış makinelerin sayısı).

2. Verilen bir üst ($J_{\max}$) ve alt ($J_{\min}$) izin verilen hücre sayısı sınırı olmak üzere,

J aşağıdaki koşulu doğrulamalıdır:

$$J_{\min} \leq J \leq J_{\max}$$

3. Verilen bir aynı hücrede bulunması gereken makineler kümesi SC,

$$\forall (M_k, M_l) \in SC, \exists C_i \in C: M_k, M_l \in C_i, \text{ burada } k, l \in \{1, \dots, m\} \text{ ve } i \in \{1, \dots, J\}$$

4. Verilen farklı hücrelerde bulunması gereken makineler kümesi SN,

$$\forall (M_k, M_l) \in SN, \exists C_i, C_j \in C: M_k \in C_i \text{ ve } M_l \in C_j, \text{ burada } k, l \in \{1, \dots, m\} \text{ ve } i, j \in \{1, \dots, J\}, i \neq j.$$

Amaç Fonksiyonu:

S önceki kısıtlara uyan bölüntüler kümesi olsun. Problem M 'in $C^* \in S$ öyleki,

$$T(C^*) = \min_{C \in S} T(C) \quad (25)$$

Bunun anlamı tüm kısıtlara uyan ve en az miktarda hücreler arası trafiğe sahip bölüntüyü aramaktır.

3.3.8.3 Genetik Algoritma

Problemin genetik algoritma ile kodlanmasında klasik tamsayı kod yerine ikili kodlama sistemi kullanılmıştır. Metodun kullandığı kod sistemi 3.3.6.3'de de anlatılan çizge teorisi temelli bir ikili kod yapısıdır. Kromozom uzunluğu Card(E)

uzunluğunda yani $G=(M,E)$ çizgesinde bulunan kenar sayısıdır. Her bir gen bir kenarı temsil eder ve eğer kenar hücreler arası bir kenar ise 1, hücre içi bir kenar ise 0 değerini alır. İkili kodun tercih edilmesinin nedeni GA'nın azaltılmış karakterlerden oluşan kod yapısına duyarlı olması, iyi blokları kolay bulması, daha az hafıza gerektirmesidir.

Örnek problem olarak 3.3.5'de kullanılan operasyon sırası verili 8 makine ve 20 parçadan oluşan makine-parça matrisi kullanılmıştır (Tablo 48).

Çizge teorisi temelli problem modeline göre problemin çizgesi Şekil 19'da verilmiştir.

Makine parça matrisine göre her bir $P_k(k=1,...,20)$ parça tipinin R_k rotaları Tablo 53'de verilmiştir. Buna göre örneğin $R_1=\{6,5\}$ ve $s_1=2$ 'dir.

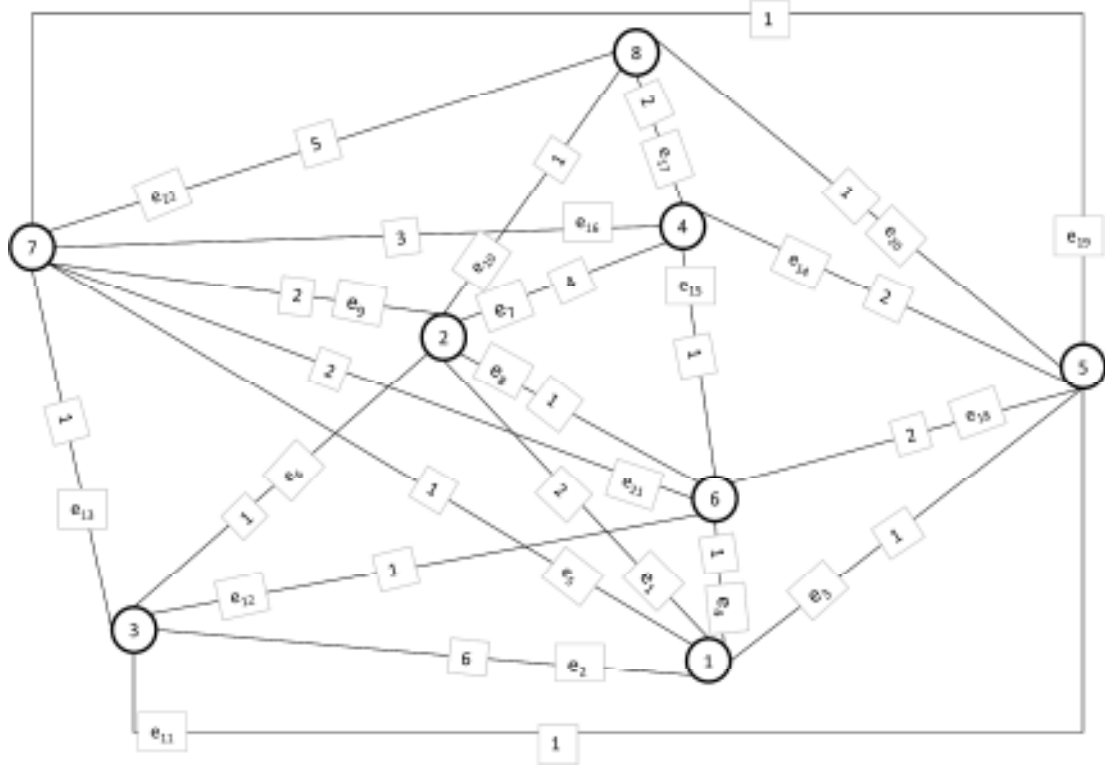
Her bir parça için r_k değeri sabit olup 1 olarak belirlenmiştir.

Her bir $v_{k,i,j}$ değeri rota tablosuna göre hesaplanmış olup Şekil 19'da e kenarları üzerine yazılmıştır.

Tablo 53. Makine-parça matrisine göre parça rotaları.

Rota (R_k)	Makine No				
R_1	6	5			
R_2	1	3			
R_3	2	1	7	8	4
R_4	2	4	7	8	
R_5	6	5			
R_6	2	4	7	8	5
R_7	8	4	7	2	
R_8	1	3			
R_9	1	6	3		
R_{10}	6	4	5		
R_{11}	7	3	1		
R_{12}	5	7	6		
R_{13}	1	3			
R_{14}	1	2	3		
R_{15}	4	5			
R_{16}	1	3			
R_{17}	3	5	1		
R_{18}	4	2	8	7	
R_{19}	1	3			
R_{20}	4	2	6	7	8

r_k değerleri tüm parçalar için 1 olduğundan makineler arası trafik (t_{ij}), $v_{k,ij}$ değerlerine eşittir.



Şekil 19. Tablo 48'de verilen makine parça matrisinin çizgesi.

3.3.8.4 Başlangıç Popülasyonu

Başlangıç popülasyonunun rastgele belirlenmesinin iyi sonuçlar elde etmeye yeterli olmasından hareketle burada da başlangıç popülasyonu rastgele belirlenmiştir. Ancak tek hücreli bir çözümden kaçınmak için kromozomlardaki 0 ve 1 genlerinin belirlenme olasılıklarının farklı bir şekilde belirlenmesi gerekmektedir. Eğer P_1 0 gen, P_2 'de 1 gen olasılığı olarak tanımlanırsa, olasılıklar aşağıdaki gibi belirlenebilir:

$$P_1 = (m-1)/\text{Card}(E) \text{ ve } P_2 = 1 - P_1$$

Buna göre örneğimizde $P_1=0,32$ ve $P_2=0,68$ olacaktır.

Aşağıda örneğimiz için dört adet rastgele oluşturulmuş kromozom verilmiştir.

Tablo 54. Genetik algoritma için rastgele üretilmiş başlangıç popülasyonu

	e ₁	e ₂	e ₃	e ₄	e ₅	e ₆	e ₇	e ₈	e ₉	e ₁₀	e ₁₁	e ₁₂	e ₁₃	e ₁₄	e ₁₅	e ₁₆	e ₁₇	e ₁₈	e ₁₉	e ₂₀	e ₂₁	e ₂₂
1. birey	1	0	1	1	1	1	0	0	1	1	1	0	1	1	1	0	1	1	1	1	0	0
2. birey	1	1	0	1	0	1	0	1	1	1	1	1	1	1	0	1	0	1	0	1	1	1
3. birey	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0	0	1	1	0	1	0	1
4. birey	0	1	1	1	1	0	1	1	0	1	1	1	1	0	1	1	1	1	1	1	1	1

3.3.8.5 Uygunluk Fonksiyonu

Amaç fonksiyonumuz toplam hücreler arası trafiğin en küçüklemesiydi. Önce en küçükleme problemi en büyükleme problemine dönüştürülür. $T'(C) = B - T(C)$ burada $T(C)$, C çözümü için hücreler arası trafiği ifade eden amaç fonksiyonu değeridir. Bu nedenle T' en büyükleme, T en küçükleme eşittir. Eğer B kümesi toplam trafiğe eşitse, $T'(C)$, C çözümünün hücre içi trafiğine eşit olacaktır. İkinci olarak elde edilen değer $Z(C) = T'(C) + (u - v(C))xB$ formülü kullanılarak çevrilir. Burada u kısıt sayısı ve $v(C)$, C 'nin doğrulanmamış kısıt sayısıdır. Bu çeviri çözümlerin B uzunluğunda $u+1$ ardışık aralıklarla incelenmesini sağlar. Başka bir deyişle ilk aşamadan sonra çözüm fonksiyonunun kısıtlar açısından değerlendirilip ince ayar yapılmasını sağlar. Buna göre Tablo 54'de verilen başlangıç popülasyonuna ait uygunluk fonksiyonu değeri aşağıdaki gibidir.

1	0	1	1	1	1	0	0	1	1	1	0	1	1	1	0	1	1	1	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

$$T'(C) = B - T(C) \text{ ise } T'(C) = 22 - 15 = 7 \text{ (hücre içi hareket sayısı)}$$

Daha sonra bu çözüm problemin matematik modelinde verilen kısıtlar açısından da incelenir ve nihai değer tespit edilir.

3.3.8.6 Genetik Operatörler

Seçim:

Bireylerin seçiminde rulet tekerleği yöntemi kullanılmıştır. Rulet tekerleğinde her bir bireyin sahip olacağı yuva aşağıdaki eşitlik kullanılarak tespit edilir.

$$P_r(C_k) = \frac{F(C_k)}{\sum_{t=1}^s F(C_t)} \quad \text{burada } s \text{ GA popülasyon büyüklüğü, } C_k, \text{ GA}$$

popülasyonundan k . çözüm ($k \in \{1, 2, \dots, s\}$) ve $F(C_k)$, C_k çözümünün uygunluğudur.

sxP_3 kez rastgele 0 ve 1 arasında bir sayı seçilir. Burada P_3 , 0 ve 1 arasında bir değer olan seçim prosedüründe seçilen birey oranıdır.

Her seferinde üretilen sayının alanına giren birey seçilir. Aşağıda Tablo 54’de verilen bireylerin uygunluk değerleri ve rulet tekerleği üzerinde kaplayacakları alanlar verilmiştir.

1. birey: $7/22=0,32$

2. birey: $6/22=0,27$

3. birey: $5/22=0,22$

4. birey: $4/22=0,18$

Buna göre rulet tekerleğinin aralıkları: 0,32-0,59-0,81-1’dir. $s=4$ ve P_3 rastgele olarak 0,86 seçildiğinden 3 kez rastgele sayı üretilir: 0,06-0,98-0,73. Üretilen rastgele sayılara göre 1, 4 ve 3 nolu bireyler seçilmiştir.

Çaprazlama:

Ebeveyn olarak adlandırılan birey çiftleri popülasyondan rastgele olarak seçilir. Ebeveynlerin her ikisi de rastgele belirlenen bir noktadan bölünerek kromozomlarında bir baş ve bir kuyruk oluşturulur. Kuyruk bölümleri yer değiştirilerek yeni yavrular oluşturulur ve ebeveynler yerlerini oluşturan yavrulara bırakırlar. Bu prosedür diğer ebeveyn ile sxP_4 kez devam ettirilir. Burada P_4 , 0 ve 1 arasında bir değerdir. Şekil 20’de $P_4=0,73$ seçildikten sonra 3 kez yapılan çaprazlama görülmektedir.

Mutasyon:

Rastgele seçilen bireylerin rastgele genleri değiştirilmek suretiyle mutasyona uğratılırlar. Mutasyona uğrayacak olan bireylerin sayısı sxP_5 kadardır. Burada P_5 0 ve 1 arasında bir değerdir. Şekil 21’de 6. yavru üzerinde yapılan mutasyon işlemi görülmektedir.

Genetik operatörlerin uygulanmasından sonra ortaya çıkan nesildeki bireyler popülasyondaki daha az uygunluğa sahip bireylerle yer değiştirirler. Böylece yeni nesilin son hali belirlenmiş olur. Yukarıda verilen genetik operatörlerin uygulanması prosedürü, önceden belirlenen iterasyon sayısı (i_{max}) kadar tekrarlanır.

1. ebeveyn	1	0	1	1	1	1	0	0	1	1	1	0	1	1	1	0	1	1	1	1	0	0
3. ebeveyn	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	0	0	1	1	0	1	0

1. yavru	1	0	1	1	1	1	0	0	1	1	1	0	1	1	0	0	1	1	0	1	0	1
2. yavru	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	0	0

3. ebeveyn	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0	0	1	1	0	1	0	1
4. ebeveyn	0	1	1	1	1	0	1	1	0	1	1	1	1	0	1	1	1	1	1	1	1	1

3. yavru	1	1	1	1	0	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1
4. yavru	0	1	1	1	1	0	1	1	0	1	1	1	1	1	0	0	1	1	0	1	0	1

1. ebeveyn	1	0	1	1	1	1	0	0	1	1	1	0	1	1	1	0	1	1	1	1	0	0
4. ebeveyn	0	1	1	1	1	0	1	1	0	1	1	1	1	0	1	1	1	1	1	1	1	1
5. yavru	1	0	1	1	1	1	0	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1
6. yavru	0	1	1	1	1	0	1	1	0	1	1	0	1	1	1	0	1	1	1	1	0	0

Şekil 20. Çaprazlama sonucunda oluşan bireyler.

6. yavru	0	1	1	1	1	0	1	1	0	1	1	0	1	1	1	0	1	1	1	1	0	0
6. yavru	0	1	1	1	1	1	1	1	0	1	1	0	1	1	0	0	1	1	1	1	0	0

Şekil 21. Yavru 6'da yapılan mutasyon işlemi.

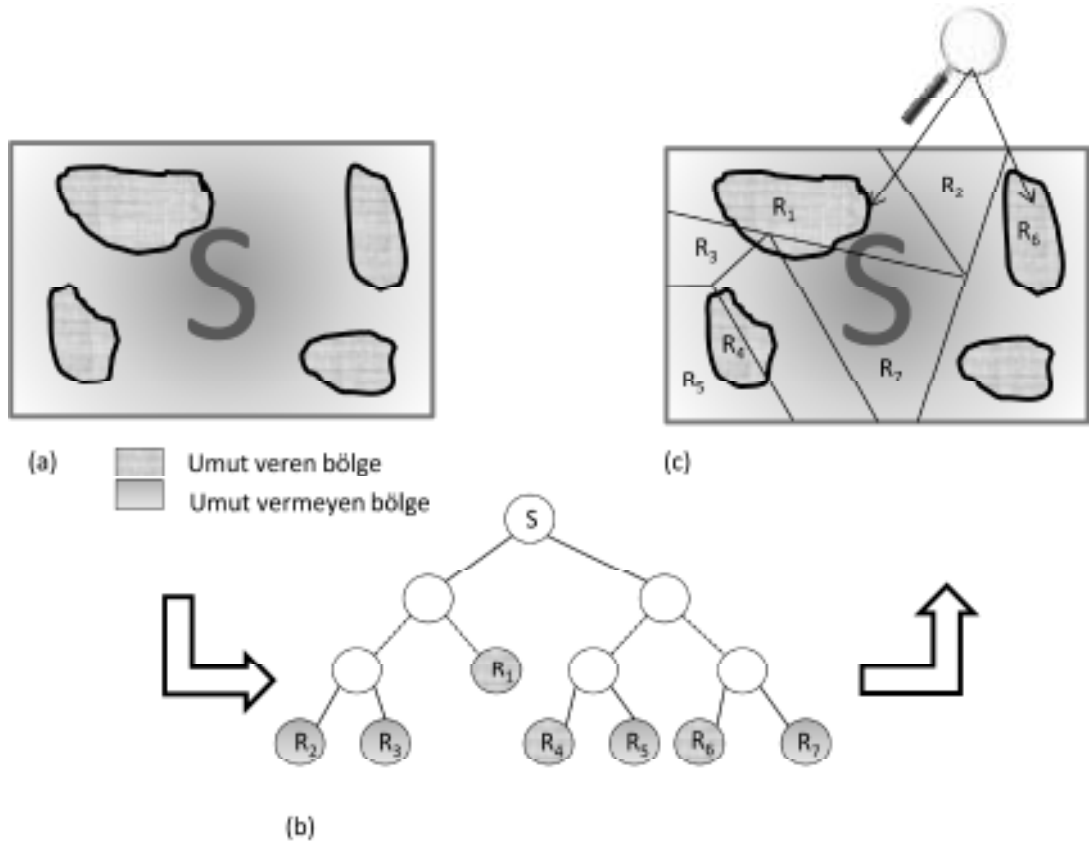
Yukarıda yürütülen genetik operatörlerin uygulanması prosedürü sırasında belirlenen kontrol parametrelerinin seçimi GA performansını direkt olarak etkiler. Popülasyon sayısı s arama uzayını iyi bir şekilde arama yapacak büyüklükte seçilmelidir. Çok büyük seçilmesi hesaplama zamanını arttırır. P_3 seçim oranı, çaprazlama sırasında en uygun bireyleri üretecek iyi bireylerin çoğaltılmasına imkan verecek şekilde seçilmelidir. P_4 çaprazlamaya girecek popülasyon yüzdesi de tüm bireylerin kalıtsal bilgilerinin devreye girmesini sağlayacak büyüklükte olmalıdır. P_5 mutasyona uğrayacak birey yüzdesi bireylerin birbirine benzemesinden kaçınmak için mümkün olduğu kadar küçük seçilmelidir. Bu parametrelerin belirlenmesi genellikle deneysel tecrübeyle yapılır.

İkili GA genellikle kötü çözümlerle genişletilmiş çözüm uzayında iyi bir arama gerçekleştirmesine rağmen, özellikle gerçek büyüklükteki problemlerle uğraşırken en çok umut veren bölgelere odaklanması daha kullanışlı olacaktır. Bu nedenle ikili GA'nın dal sınır tekniği ile iyileştirilmesine karar verilmiştir.

3.3.8.7 Dal-Sınır İyileştirme

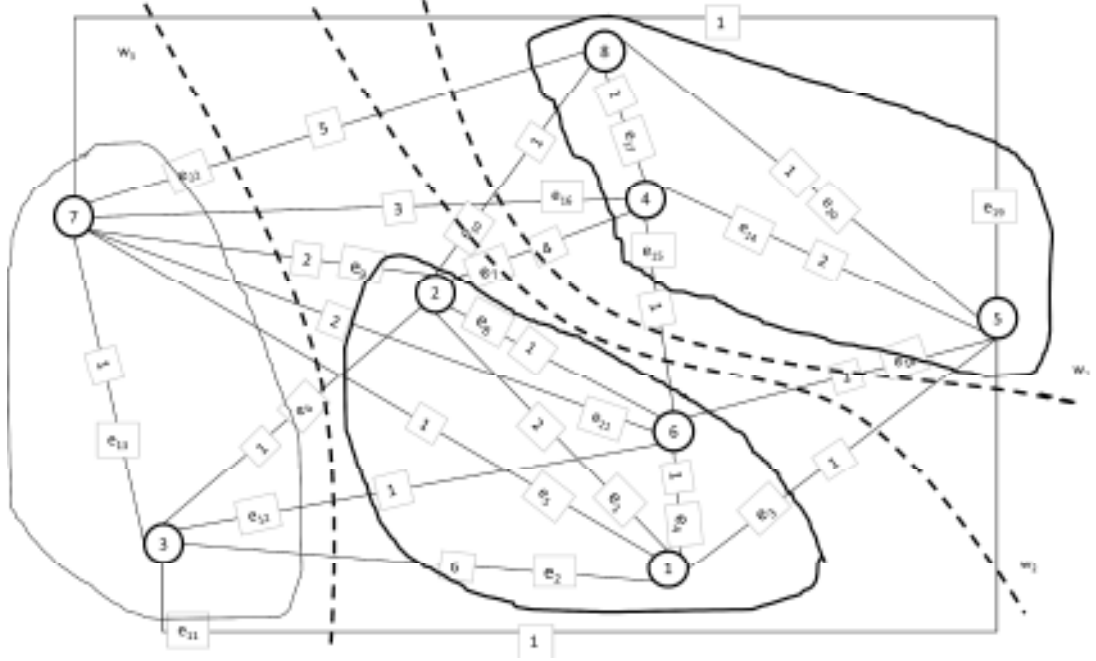
Dal-sınır sayımları çözüm kümesinin dallandırma prosedürü olarak adlandırılan alt kümelere bölünmesiyle yapılır. Daha sonra ortak çözümlerin amaç fonksiyonları için alt sınır kuran bir değerlendirme fonksiyonu her bir alt küme için hesaplanır. Dal-sınır sayımları “zeki” olarak tanımlanır, çünkü v 'ye eşit bir amaç fonksiyonuyla uygulanabilir bir çözüme ulaşıldığında, v 'den yüksek alt sınırı olan tüm dallar aranmaz. Bu prosedürler, en iyi çözümün belirlenmesi için tekrarlanırlar.

GA'yı umut verici olmayan aramalardan korumak için, iyi çözümleri gizleyebilecek bölgeleri belirlemek için bir dal-sınır prosedürü kullanılır. Daha sonra bu bölgeler GA ile aranır (Şekil 22).



Şekil 22. Dal-sınır iyileştirme (Boulif ve Atif, 2006: 2219-2245).

Yukarıdaki durumu formal olarak açıklarsak; daha önce verilen çizge teorisi temelli modeli göz önüne alalım. Çözümün çizge bölme olması nedeni çözüm, bağlantılı çizge olması durumunda, boolean operatörü *VEYA* kullanılarak alt çevrimlerin toplamı (kesikler) olarak temsil edilmesi mümkündür.



Şekil 23. Çizgenin alt çevrimlere bölünmesi.

Şekil 23’de daha önceki örneğin w_1 , w_2 , ve w_3 gibi üç kesikle ayrılmış çözümü görülmektedir. Burada örneğin w_1 ile w_3 ’ün toplamı bize çözümün genetik kodunu verir (Şekil 24).

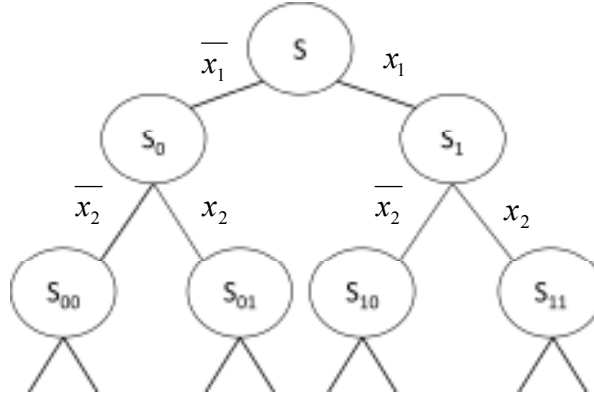
w_1	0	0	1	0	0	0	1	0	0	1	1	0	0	0	1	1	0	1	1	0	0	1
w_3	0	1	0	0	1	1	0	0	1	0	1	1	0	0	0	1	0	0	1	0	1	1
$w_1 \oplus w_3$	0	1	1	0	1	1	1	0	1	1	1	1	0	0	1	1	0	1	1	0	1	1

Şekil 24. Çizge için w_1 ve w_3 alt çevrimlerinin toplamı.

Metodun uygulanması için dallandırma kuralları ve alt sınırlar tanımlanmalıdır.

Dallandırma Kuralları:

$A=\{w_1, w_2, \dots, w_n\}$ G akış çizgesiyle ilişkili n^6 alt çevrim kümesi ve S tüm mümkün çözümler kümesi olsun. S çözüm kümesi önce w_1 'in yapılandırılmalarda kullanılıp kullanılmamasına göre S_0 ve S_1 gibi iki yavru alt kümeye bölünür (S_0 , w_1 tarafından yapılandırılmayan çözümler alt kümesidir ve S_1 , w_1 tarafından yapılandırılan çözümler alt kümesidir). Daha sonra her bir alt küme S_0 ve S_1 , her bir alt kümenin w_2 tarafından yapılandırılıp yapılandırılmadıklarına göre iki yavru alt kümeye bölünür (S_0 , S_{01} ve S_{00} 'a, S_1 , S_{11} ve S_{10} 'a). Dallandırma bu şekilde devam ettirilir.



Şekil 25 Dallandırma kuralının gösterimi.

a. Dallandırma prosedürü: Prosedür her bir w_i alt çevriminin $i \in \{1, \dots, n\}$ bir x_i karar değişkeniyle ilişkilendirilmesiyle tanımlanır. x_i aşağıdaki şekilde tanımlanabilir:

$$x_i = \begin{cases} 1 & \text{eğer } Co_i \text{ yavru alt kümeyi yapılandırır,} \\ 0 & \text{aksi halde} \end{cases}$$

Böylece örneğin S_{00101} alt kümesi $\overline{x_1} \overline{x_2} x_3 \overline{x_4} x_5$ karar sırasıyla tanımlanan yolla ulaşılan alt kümeyi tanımlar. Burada x_1, x_2, x_4 0'a ve x_3, x_5 1'e eşittir. Bu alt küme $w_3 \oplus w_5$ toplamı tarafından yapılandırılan çözümü içerir.

⁶ m düğümlü bir çizgede $n 2^{m-1}-1$ 'e eşittir.

b. Alt çevrim listesinin yapılandırılması: Alt çevrimlerin sayısı n , düğüm sayısı ile üssel olarak büyür ve gerçek örnekler için devasa olabilir. Bu nedenle dallandırma prosedürü, alt çevrim listesi (L) olarak adlandırılan ve rastgele veya bir optimizasyon prosedürü ile belirlenebilen, seçilen alt çevrimler alt kümesi temeline dayanır. Alt çevrim listesi L , metodun etkinliğini etkileyebilir ve bu nedenle elemanları özel dikkatle seçilmelidir. L listesi çok uzun olmadan, çözüm bölgesinin iyi bir şekilde aranmasına izin verecek yeterli uzunlukta olmalı. Çünkü aksi durum hafıza ihtiyacını ve hesaplama zamanını arttıracaktır. İlave olarak, düşük ağırlıklı (alt çevrim ağırlığı kenar ağırlıklarının toplamıdır) alt çevrimleri seçmek iyi çözüm bölgelerini tespit etmek açısından önemlidir.

Yukarıdaki açıklamaların ışığında, alt çevrim listesinin çoğunluğunu belirlemek için bir GA kullanılabilir ve geriye kalan ise rastgele üretilebilir. Çünkü XOR boolean operatörüyle ilişkilendirilmiş alt çevrim kümesi bir vektör uzayı tanımlar, her bir alt çevrim bu vektör uzayında verilen bir tabandan alt çevrimlerin tek doğrusal kombinasyonları ile üretilebilir. GA kodu $m-1$ uzunluğunda (yani düğüm sayısı eksi bire eşit taban boyut) ve her bir genin taban alt çevrimle ilişkili olduğu bir zincirle tanımlanır. Gen zincir tarafından tanımlanan alt çevrimi üretmek için alt çevrimin kullanılıp kullanılmadığı gösterir. Alt çevrimlerin vektör uzayının tabanı örten ağaç metoduyla belirlenebilir.

Alt Sınırlama:

İyi bir alt sınırlama prosedürü birbirine zıt olan iki önemli özelliklerle karakterizedir: çabukluk ve sıkılık. Birincinin anlamı, alt kümelerin değerlendirmelerinin mümkün olduğu kadar çabuk hesaplanma yeteneğidir ve bu genelde düşük karmaşıklık prosedürlerini doğurur. İkincisinin anlamı, bu değerlendirmenin değerlendirilmiş alt kümenin en iyi çözümüne mümkün olduğu kadar yakın olma zorunluluğudur. Bu ikinci özellik engelleme prosesini ve böylece dal-sınır aramanın zekiliğini iyileştirir.

Bu iki özelliğin zıtlığını ortadan kaldırmak, izleyen prosedürle sağlanabilir: pozitif karar değişkenlerine sahip olan alt çevrimlerin toplamı ile tanımlanan çözümün amaç fonksiyonu (24), alt kümenin alt sınırının esas bir değerlendirilmesini oluşturmak için kullanılır.

Bir alt küme, bir w_i alt çevrimine göre iki yavru alt kümeye bölündüğünde, yavrunun temel değerlendirmesi, karar değişkeni x_i 'nin pozitif değeri ile ilgili olarak, ebeveyninin değerlendirmesinden elde edilir. Burada ebeveynin değerlendirmesine, ebeveyninin çözüm alt kümesini oluşturan alt çevrimlere ait olmayan w_i kenarlarının ağırlığı eklenir. x_i 'nin boş değeri ile ilgili ikinci yavru için değerlendirme tamamen aynıdır.

Bu temel değerlendirme, eğer mevcut alt küme ile ilgili çözüm uygulanabilir değilse sıkılaştırılabilir. Aslında, hücre büyüklüğü, hücre sayısı veya birlikte bulunmama kısıtının ürettiği uygulanamazlık, uygulanabilir çözümü elde etmek için en azından bir hücrenin bölünmesini gerektirir. İlk olarak hücre büyüklüğü kısıtı dikkate alınır: eğer çözüm, bir hücrede en fazla izin verilen makine sayısını geçen hücreler içeriyorsa, her bir hücrede hücre içi kenarların en düşük ağırlığının değeri belirlenir. Daha sonra bunların toplamı esas değerlendirmeye eklenir. Aksi halde en az izin verilen hücre sayısı dikkate alınır: eğer bu değere uyulmazsa, hücre içi kenarlar arasından en düşük ağırlığa sahip kenarın ağırlığı eklenir. Son olarak, önceki iki kısıta da uyulursa, birlikte bulunmama kısıtı dikkate alınır: uçları birlikte bulunmayan makineler olan hücre içi kenarların ağırlıklarının toplamı esas değerlendirmeye eklenir.

Bir alt küme için bir kez uygulanabilir çözüm tanımlandıktan sonra daha fazla alt çevrim eklemek uygulanabilir çözümün hücre içi trafiğini veya kendisinin hücre sayısını arttırması nedeniyle aranmayı durdurur. Tüm karar değişkenlerinin 0'a eşit olduğu bu alt kümelerin varsayılan değerlendirmesi için K 'nın toplamı ilk alt kenar ağırlıkları kullanılır. Burada K , bir alt kümeyi tanımlayan en az sayıda kenardır.

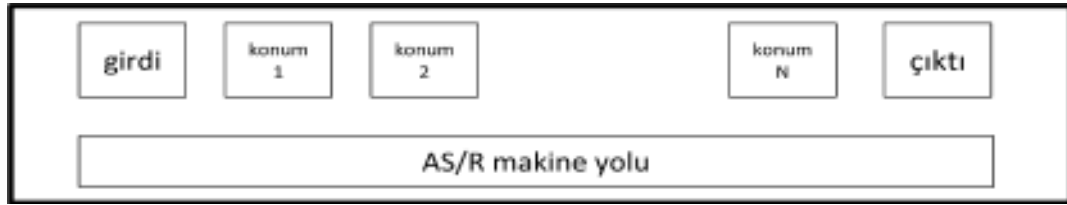
3.3.8.8 GA Arama

Dal-sınır ağacının belli bir derinliğine ulaştıktan sonra, en iyi dallar, GA prosedürü kullanılarak ilgili alt kümeleri araması için seçilir. Arama, bazı GA genlerini 1'e eşitleyerek verilen bir dal alt kümesine sınırlandırılabilir. Bu, çözümdeki bir hücreler arası kenarla bağlı her makine çifti için birlikte bulunmama kısıtı eklenerek yapılır (pozitif karar değişkeni alt çevrimleri toplamı ile yapılandırılan). Belirtmekte yarar var ki, genetik kod yapısı böylece azaltılır çünkü sabitlenmiş (fıkse edilmiş) genler, tüm amaçlara, genetik zincirden çıkarılmıştır.

3.3.9 Doğrusal Hat Boyunca Hücre Atama Probleminin Grafik Metot Kullanılarak Çözümü

Hücreyel imalat sistemlerinde sıklıkla gördüğümüz “U” şeklindeki makine yerleşimi dışında imalat sisteminin özellikleri gereği diğer yerleşim şekilleri de karşımıza çıkmaktadır. Bunlardan biri de doğrusal hat üzerindeki yerleşimlerdir. Doğrusal hat üzerine yerleştirilen bir tesiste, makineler arasında işleri taşıma görevini üstlenen otomatik bir depolama ve geri alma makinesi (AS/R), bu makinenin en az taşıma maliyetine yol açacağı yerleşimi bulmak önem kazanmaktadır. Bu şekilde bir yerleşimde elimizde N adet makine ve bunların atanacağı N adet konum olduğunu ve bu makineler boyunca bir hat üzerinde iş taşıyan bir AS/R düşünürsek, üretim talebinin belirli olduğu bir atölye üretim sistemi ortamında her bir makinenin en iyi konuma atanması problemi “karesel atama problemi⁷” olarak bilinir (Houshyar ve Ginnis, 1990: 1485-1498).

Yukarıda belirtilen üretim sisteminin gösterimi Şekil 26’da görülmektedir.



Şekil 26 İş istasyonları ve AS/R makinesinin gösterimi (Houshyar ve Ginnis, 1990: 1485-1498).

Sistemde işler girdi bölümünden girip operasyon sıralarına göre ilgili konumlardaki makinelerde işlendikten sonra çıkıtı bölümünden sistemi terk etmektedir. Konumlar doğrusal bir hat boyunca yerleşmiş olup, işler konumlar arasında bir AS/R makinesi ile taşınmaktadır. Konumlar eşit uzaklıktadır ve iş istasyonları bu konumlardan herhangi birine yerleştirilebilir. AS/R makinesi her seferinde yalnız bir birim iş taşıyabilir.

Atama vektörü S ’nin kullanımı karesel atama probleminin formülasyonunu çalışmak için kolay hale getirecektir.

⁷ Matematikteki en iyileme veya yöneylem araştırması dalındaki tesis yerleştirme kategorisindeki kombinatoryal en iyileme problemlerinin temellerinden biridir.

$d(i,j)$ i ve j konumu arasındaki mesafe

w_{ij} birim zamanda i iş istasyonundan j iş istasyonuna giden iş miktarı

$s(i)$ i iş istasyonunun atandığı konum

S atama vektörü, $[s(1), s(2), \dots, s(N)]$

S atama kullanılarak işler tarafından birim zamanda kat edilen mesafe:

$$TC(S) = \sum_{i=0}^{N+1} \sum_{j=0}^{N+1} w'_{ij} d[s(i), s(j)] \quad (26)$$

Burada $i=0$ ve $j=N+1$, sırasıyla girdi ve çıktı iş istasyonlarına tekabül etmektedir. $TC(S)$, işler tarafından kat edilen mesafe olduğundan AS/R'nin sadece yüklü hareketi dikkate alınmıştır.

Genel olarak, $W'=(w'_{ij})$ simetrik değildir. Hesaplama uygunluğu nedeniyle W simetrik matrisiyle çalışmak tercih edilmelidir:

$$w_{ij} = \frac{1}{2} (w'_{ij} + w'_{ji}) \quad (27)$$

$s^{-1}(j)$, j konumuna atanmış olan iş istasyonunu temsil etsin. Bu durumda atama maliyeti:

$$TC(S) = \sum_{i=1}^{N+1} \sum_{j=0}^{N+1} w'_{s^{-1}(i), s^{-1}(j)} d(i, j) = \sum_{i=0}^{N+1} \sum_{j=0}^{N+1} w'_{s^{-1}(i, j)} d(i, j) \quad (28)$$

$$TC(S) = 2 \left\{ \sum_{i=0}^N \sum_{j=i+1}^{N+1} w_{s^{-1}(i, j)} d(i, j) \right\} \quad (29)$$

Son olarak, konumların eşit aralıklı olduğu varsayımı daha fazla kolaylık sağlamaktadır.

$$d(i,j)=(j-i)\sigma \quad j>i \quad (30)$$

Burada her bir konumun büyüklüğüdür. Böylece sadece $w_{s^{-1}(i, j)}$ üst diyagonal elemanlarını dikkate alan atama maliyeti aşağıdaki gibi olur:

$$TC(S) = 2 \left\{ \sum_{i=0}^N \sum_{j=i+1}^{N+1} w_{s^{-1}(i,j)} (j - i) \right\} \sigma \quad (31)$$

Şekil 26’da gösterilen problem için çizge teorisinde⁸ kullanılan kesik yaklaşımı uygulanmış ve m halihazırda atanmış olduğu n yeni tesisin konumlandırılması problemi olarak düşünülmüştür. Houshyar ve McGinnis’in önerdiği metot N adımdan oluşur ve k . adımda seçilen iş istasyonlarından biri k . konuma atanır. Böylece N adımda tüm iş istasyonlarının konumlarına atanması tamamlanır.

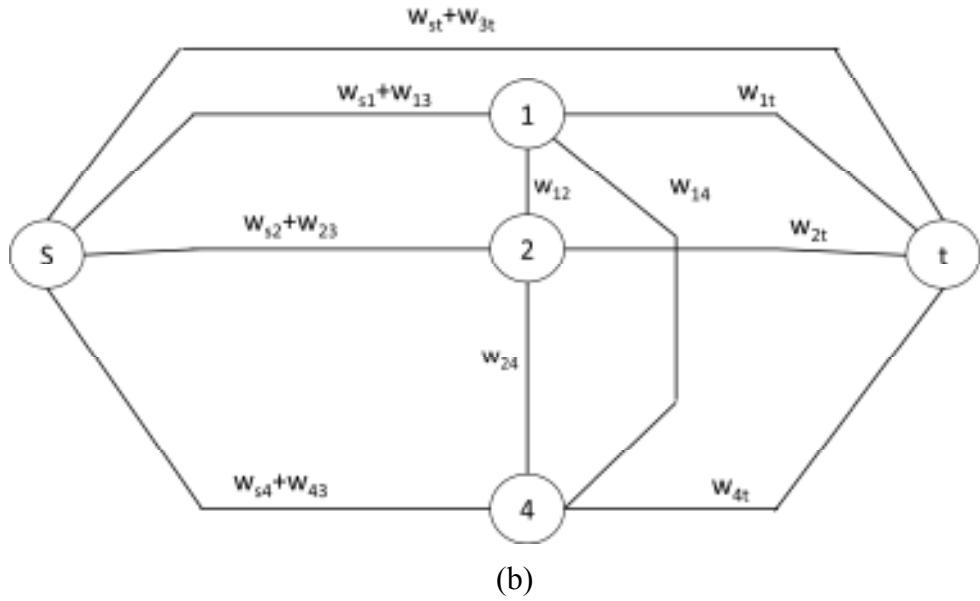
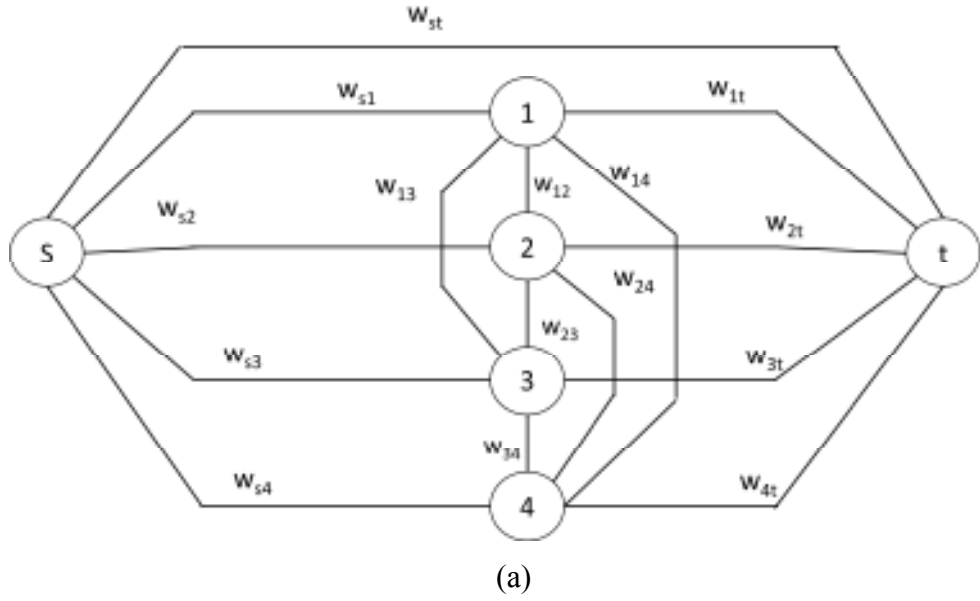
k . adımda, $(k=1, \dots, N)$, $(k-1)$ iş istasyonu hali hazırda konumlandırılmıştır ve $(N-k+1)$ iş istasyonu henüz atanmamıştır. A_k bu konumlandırılmış olan $(k-1)$ iş istasyonunu ifade ediyor olsun. Geriye kalan atanmamış iş istasyonları B_k ile ifade edilsin. Böylece $A_k \cup B_k = \{1, 2, \dots, N\}$. k adımında $(N-k+3)$ düğüme sahip $\{s, B_k, t\}$ şebeke düşünürüz. Burada s ve t sırasıyla girdi ve çıktı iş istasyonlarıdır ve her bir düğüm $j \in B_k$ henüz atanmamış olan bir iş istasyonuna tekabül eder.

$B_k \cup \{t\}$ ’deki herhangi bir j için aşağıdaki kapasiteye sahip yönsüz (s, j) kenarları

$$C(s, j) = \sum_{i \in A_k \cup s} w_{ji}$$

ve $C(t, j) = w_{jt}$ kapasiteli yönsüz (t, j) kenarı tanımlanır. Ayrıca, B_k ’daki tüm i ve j ’ler için $C(i, j) = w_{ij}$ kapasiteli (i, j) yönsüz kenarları tanımlanır ve bu şebeke k -yer şebeke olarak adlandırılır. $N=4$ iş istasyonlu bir problem için 1-yer şebeke ve yine aynı problemin 3 nolu iş istasyonunun 1 konumuna atanmış halinin 2-yer şebekesi Şekil 27’de gösterilmiştir.

⁸ Çizgeleri inceleyen matematik dalıdır. Genel olarak bir çizge $G(V, E)$ ile gösterilir. $V=V(G)$ düğümler, $E=E(G)$ kenarlar kümesidir. Böylece n sayıda V düğümü k sayıda kenar ile birbirine bağlıdır. Hücreli imalat problemlerine uygulandığında düğümler makineleri, makineler arasındaki kenarlar ise makineler arasındaki akışı temsil eder.



Şekil 27. (a) $N=4$ iş istasyonlu problem için 1-yer şebeke. (b) iş istasyonu 3'ün atamasının yapılmış olduğu $N=4$ iş istasyonu için 2-yer şebeke.

Eğer k -yer şebekenin $(N-k+1)$ düğümünden oluşan B_k kümesini sadece v düğümü X_v kümesinde, kalan düğümler $\overline{X_v}$ kümesinde kalacak şekilde X_v ve $\overline{X_v}$ gibi iki ayrı kümeye bölersek, $s \cup X_v$ düğüm kümesi ve $t \cup \overline{X_v}$ düğüm kümesi k -yer şebekesinde bir kesiği tanımlar. Bu kesiğin bir ucu $s \cup X_v$ 'de, diğer ucu ise $t \cup \overline{X_v}$ 'dedir. Kesik $(X_v, \overline{X_v})$ şeklinde gösterilir. $(X_v, \overline{X_v})$ kesiğinin kapasitesi, kesikteki

kenarların toplam kapasitesine eşittir ve $C_k(X_v, \bar{X}_v)$ ile gösterilir. Minimum kapasiteye sahip kesik, minimum kesik olarak adlandırılır.

$$C_k^* = \min C_k(X_v, \bar{X}_v) \text{ tüm } v \in B_k \quad (32)$$

Sezgisel metot aşağıdaki gibi çalışır:

Konum 1 ile başlanır. Bu aşamada tüm iş istasyonları bu konuma atanmak için adaydırlar. Böylece şebeke başlangıçtaki kapasiteleriyle (w_{ij}) , $(N+2)$ düğümünden $\{s, 1, 2, \dots, N, t\}$ oluşur. Burada N farklı kesğin değeri değerlendirilir; her birinde, X_v 'yi oluşturan s düğümü ve 1'den N 'e kadar olan düğümlerden biri vardır ve geriye kalan düğümler X_v 'yi oluştururlar. En küçük değerli kesik seçilir ve $v1$ olarak adlandırılan ilgili iş istasyonu konum 1'e yerleştirilir. Bundan sonra $N+2$ 'den $N+1$ 'e şeklinde düğümlerin azalan sayısı ile şebeke güncellenir. Sonraki değerlendirme için $v1$ çıkarılır. s 'i diğer düğümlere bağlayan kenarların kapasiteleri ilgili $w_{j,v1}$ 'den w_{sj} 'ye ekleyerek güncellenir. Bu prosedür $(N-1)$ kez tekrarlanır ve her adımda 1 adet atanmamış iş istasyonu bir konuma atanır.

3.3.9.1 Grafik Metot İçin Sayısal Bir Örnek

Yukarıda verilen algoritma için Şekil 28'de ilgili verilerinin verildiği 4 makine ve 3 üründen oluşan bir sistem kullanılacaktır. Şekilden de anlaşıldığı gibi sistemde 4 farklı ürün tipi mevcuttur. Her bir ürün tipinin üretim hacmi ve izlediği makine rotaları verilmiştir. Buna göre akış hacmi orijinal matrisi oluşturulmuştur. Örneğin 1 nolu tesisi 1 ve 2 nolu ürün tipleri girdi-1 sırasında kullanmaktadır. Dolayısıyla bunların üretim hacmi toplamı $(29+23=52)$ akış matrisinde girdi-M1 kesişimine yazılır. (27) eşitliği gereği akış matrisi simetrik hale getirilir. Mesafe matrisi, her bir tesis arasında bir birim mesafe olduğu varsayılarak oluşturulmuştur.

$k=1$. adımımda $A_1=\{\emptyset\}$ ve $B_1=\{1,2,3\}$ 'dir. Kenar kapasiteleri orijinal w_{ij} kapasitelerindedir. Böylelikle Şekil 29'daki şebeke elde edilir. B_1 kümesi üç farklı şekilde bölünebilir ve bu bölme sonucunda 89 $(0+26+29+29+2+3)$, 66 $(0+26+29+3+5+3)$ ve 64 $(0+26+2+5+29+2)$ kapasiteleri ortaya çıkar. En küçük kapasiteyi seçeceğimiz için $v1=3$ ve $C_1=64$ olur.

2. adımda güncellenmiş olan şebeke Şekil 30’da gösterilmiştir. Burada $A_2=\{3\}$ ve $B_2=\{1,2\}$ ’dir. B_2 kümesi kapasiteleri 64 ve 89 olan iki farklı yolla bölünebilir. Böylece $v_2=1$ ve $C_2^* = 64$ olur.

s ve t ’nin konumlandırıldığı 31 başlangıç kesiği ve 31 son kesiğin eklenmesiyle toplam maliyet 190’ı elde ederiz.

Ürün Tipi	Üretim Oranı	Operasyon Sıraları
1	29	girdi-1-2-1-3-1-çıkıtı
2	23	girdi-1-çıkıtı
3	6	girdi-3-2-çıkıtı
4	4	girdi-2-3-çıkıtı

Girdi	M1	M2	M3	çıkıtı
g	0	52	4	6
1	0	0	29	29
2	0	29	0	4
3	0	29	6	0
ç	0	0	0	0

Akış hacmi orijinal matrisi

Girdi	M1	M2	M3	çıkıtı
g	0	26	2	3
1	26	0	29	29
2	2	29	0	5
3	3	29	5	0
ç	0	26	3	2

Akış hacmi simetrik matrisi

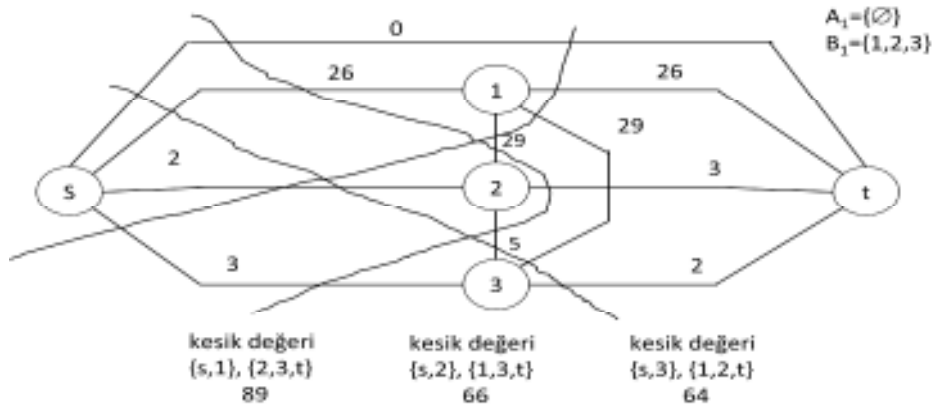
Girdi	M1	M2	M3	çıkıtı
g	x	1	2	3
1	1	x	1	2
2	2	1	x	1
3	3	2	1	x
ç	4	3	2	1

S ataması kullanılarak mesafe matrisi

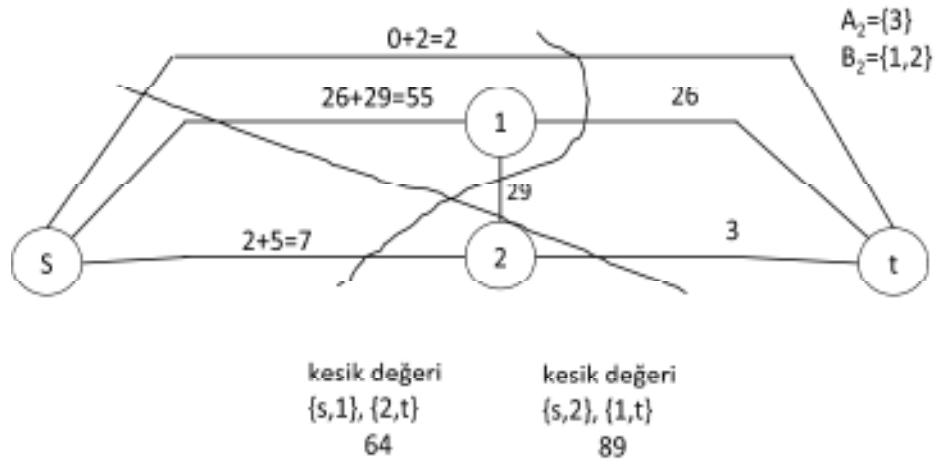
girdi	Tesis 1	Tesis 2	Tesis 3	çıkıtı
-------	---------	---------	---------	--------

[1,2,3] ataması gösterimi
TC(S)=434

Şekil 28. Grafik metodu için 4 makine ve 3 ürün için örnek.



Şekil 29. Grafik metodu sayısal örneğinin 1. adımı.



Şekil 30. grafik metodu sayısal örneğinin 2. adımı.

DÖRDÜNCÜ BÖLÜM: HÜCRE OLUŞTURMA İÇİN AKIŞ MATRİSİ TEMELLİ SEZGİSEL METOT

4 Giriş

Bu bölüme kadar yapılan incelemede hücresel imalatla hücre oluşturma probleminin genel olarak girdisini teşkil eden verileri ve bu veriler üzerinden çalışan yaklaşımları gördük. Buna göre problem modeline sağlanan girdilerin seçilmesi, çözümün gerçekçiliğine katkı sağlaması açısından önemlidir. Aksi takdirde yapılacak çalışmanın gerçek imalat ortamına uygulanması gittikçe zorlaşacaktır. Girdiler açısından bakıldığında hücre oluşturma probleminde ikili makine parça matrisi yerine operasyon veri sıra verisinin kullanılmasının daha gerçekçi sonuçlara ulaşmayı sağlaması nedeniyle bu konuda yapılan son çalışmalar artık bu veriyi kullanmaktadır. Bunun yanında operasyon sıra verisi problemin çözümü esnasında analizciye ek bilgiler de sağlamaktadır. Makineler hücrelere gruplandıktan sonra bu makineler arasındaki akış şekli belli olduğundan, hücre oluşturma işinin ötesine geçerek oluşan hücreler içinde taşıma maliyetlerini en küçükleyecek yerleşim düzenlerini de tespit etmek mümkün olabilmektedir. İşte bu bölümde operasyon sıra verisini girdi olarak kullanarak hücre oluşturan ve eş zamanlı olarak hücre yerleşimlerini de belirleyen sezgisel bir metodu inceleyeceğiz. Metotla birlikte hücre oluşturma performansını denetleyen önemli kriterler de verilerek diğer algoritmalarla göre performans değerlendirmesi yapılacaktır.

4.1 Akış Matrisi Girdisi

Önceki bölümde operasyon sıra verisinin nasıl kullanıldığına dair örnekler verilmişti. Operasyon sıra verisi kullanılarak makinelerin benzerliğinden yararlanmanın yanı sıra, bu verinin direk olarak sezgisel bir algoritmanın içinde kullanıldığı örnekler literatürde mevcuttur. Burada incelenen algoritma sıra verisinden çıkarılan “hareket Katsayısı”nı kullanmaktadır.

Bu metotta oluşturulan akış matrisinde parçaların makineler arasında yaptıkları ileri hareketler temel alınmıştır. Bunun nedeni üretim planlama ve kontrol açısından bakıldığında hücre içindeki geriye hareketlerin ve iş istasyonlarının atlanmasının planlama faaliyetlerinin karmaşıklığını arttırdığından istenmemesidir. Bu nedenle

parçaların makinelerden operasyon sırası taleplerine göre makineler arasındaki ileri hareketleri ortaya çıkaran bir akış matrisi kullanılmaktadır.

Akış matrisinin açıklaması şu şekilde yapılabilir: j ve j' olan iki makine düşünelim. Eğer i parçası hemen ardışık olarak bu iki makineyi ziyaret ediyorsa bu makine çifti arasında bir birim ileri hareketin varlığından söz edilebilir. Hareket katsayısı olarak adlandırabileceğimiz bu ölçütü $f_{jj'}$ ile gösterirsek, $X_{ijj'}$ i parçası için j ve j' makineleri arasındaki birim hareket katsayısıdır ve aşağıdaki gibi ifade edebiliriz:

$$X_{ijj'}=0 \text{ eğer } S_{ij}=0$$

$$X_{ijj'}=0 \text{ eğer } S_{ij}=0$$

$$X_{ijj'}=1 \text{ eğer } S_{ij'}-S_{ij}=1$$

$$X_{ijj'}=0 \text{ aksi takdirde}$$

$$f_{jj'} = \sum_{i=1}^P X_{ijj'}$$

(33)

Tüm parçalar için makinelerin birbirleriyle olan hareket katsayıları yukarıdaki eşitlik kullanılarak hesaplanabilir ve böylece makineler arası akış matrisi ($F=[f_{jj'}]$) oluşturulabilir. F akış matrisi bir $M \times M$ asimetrik matristir.

4.2 Hücre Oluşturma ve Yerleşim Tasarımı Algoritması

Burada önerilen algoritma eş zamanlı olarak makine hücrelerini oluştururken, hücreler içindeki makineleri de sıraya koyarak hücreler içindeki makinelerin de yerleşimini tasarlamaktadır. Algoritma bu nedenle hücreler arası hareketlerden çok, hücre içi malzeme hareketlerine odaklanmıştır. Bu nedenle akış matrisindeki makineler arası en büyük akışı dikkate alarak hücre içindeki hareketleri en küçüklemeye çalışır.

Algoritma en yüksek akış değerine sahip olan makine çiftinin seçilmesiyle başlar. Ortaya çıkan kısmi çözüm daha sonra diğer makinelerin bu hücreye uygun bir şekilde eklenmesiyle geliştirilir. Eğer eklenecek makine bulunamazsa yeni hücre oluşturulur. Prosedür tüm makineler hücrelere atandıktan sonra durdurulur. İlgili parça aileleri tanımlanır. Çözüm sadece makinelerin her bir hücredeki gruplamasını

oluşturmaz, aynı zamanda ileri hareketleri en büyükleyecek sırada hücrelere yerleştirir.

Algoritmanın adımları aşağıdaki gibidir:

Adım 1: S_{ij} matrisini oku.

Adım 2: S_{ij} matrisi yardımıyla akış matrisini yapılandır.

Adım 3: Yeni bir hücre yarat.

Adım 4: Akış matrisinde sıfırdan farklı en büyük girdiyi eşik değeri olarak belirle.

Adım 5: Akış matrisinde eşik değerinin olduğu konumu (j, j') seç. Eşitlik durumunda herhangi birini rastgele seç. Sonra j . ve j' . makineyi mevcut hücreye ata.

Adım 6: j . (j' .) makine için aynı eşik değerinin j . satırda (j' . kolonda) bulunup bulunmadığını kontrol et. Eğer (j', j'') konumda ((j'', j) konumda) varsa j'' . makineyi (j' makineyi) j' . makineden sonra (j . makineden önce) mevcut hücreye ata. Şimdi j' . makine (j . makine) bir sonraki değerlendirme için çıkarılır ve j''' . satırda (j'' . kolonda) güncel eşik değerinin bulunup bulunmadığını kontrol et. Eğer mevcutsa, aynı şekilde devam et.

Adım 7: Eğer tüm makineler atanmışsa, 15. adıma git.

Adım 8: Eğer tüm eşik değerler tükenmişse 10. Adıma git.

Adım 9: Yeni bir hücre yarat ve 5. Adıma git.

Adım 10: Akış matrisindeki bir sonraki en büyük değeri eşik değer olarak al.

Adım 11: Oluşan hücrenin son makinesine (ilk makinesine) uyan satırdaki (kolondaki) konumu seç.

Adım 12: Eğer tüm makineler atanmışsa 15. Adıma git.

Adım 13: Eğer tüm eşik değerler tükenmişse 10. Adıma git.

Adım 14: Yeni hücre yarat ve 5. Adıma git.

Adım 15: Hücre ailesi yarat, operasyonları için en fazla sayıda makineyi talep eden parçayı hücreye ata, eğer eşitlik mevcutsa en iyi CUI'nin elde edildiği parçayı hücreye ata.

4.3. Hücre Oluşturma ve Yerleşim Düzeni Tasarımı İçin Performans Göstergesi

Hücre oluşturma problemlerinde azaltılmaya çalışılan istisnai elemanlar ve dolayısıyla hücreler arası hareket en küçüklemesinin aksine, hücre yerleşim düzeni tasarımının performansının göstergesi hücre içi hareketlerin dolayısıyla malzeme taşıma maliyetlerinin en küçüklemesidir. Parçaların işlem görme sırasında ihtiyaç duyduğu proseslerin mümkün olduğunca birbirine yakın olması ve bu proseslerde daima ileri hareket yaparak işlemlerini tamamlamaları istenen durumdur. Bu nedenle çözümün iyiliğini ölçmek için üç performans ölçütü geliştirilmiştir: hücre hareket indeksi, hücre kullanım indeksi ve toplam hareket indeksi.

4.3.1. Hücre Hareket İndeksi

k hücresi için hücre hareket indeksi CMI_k , ileri hareket sayısının, hücre içindeki toplam hareket sayısına oranı olarak tanımlanır. Ortalama hücre hareket indeksi $ACMI$ ise CMI_k 'nın hücrelerdeki parça sayısı ile ağırlıklandırılmış ortalamasıdır.

$$CMI_k = \frac{k \text{ hücresi içindeki ileri hareket sayısı}}{k \text{ hücresi içindeki hareket sayısı}} = \frac{N_{fk}}{N_{tk}} \quad (34)$$

$$\text{Ortalama hücre hareket indeksi, } ACMI = \frac{\sum_{k=1}^C P_k * CMI_k}{P} \quad (35)$$

Burada, P_k : k hücresindeki parça sayısı,
 $k=(1,...,C)$.

4.3.2. Hücre kullanım İndeksi

Her bir hücre için hücre kullanım indeksi CUI_k , k hücresindeki operasyon sayısının, k hücresinin büyüklüğüne oranı olarak tanımlanır. Ortalama hücre kullanım indeksi $ACUI$, her bir hücredeki hücre kullanım indeksi toplamının, hücre sayısına oranı olarak tanımlanır.

$$CUI_k = \frac{k \text{ hücresindeki operasyon sayısı}}{k \text{ hücresinin büyüklüğü}} = \frac{N_{opk}}{N_{sk}} \quad (36)$$

$$ACUI = \frac{\sum_{k=1}^C CUI_k}{C} \quad (37)$$

Burada C : hücre sayısı,
 $k=(1,...,C)$

4.3.3. Toplam Hareket İndeksi

Diğer kullanılan performans ölçümü toplam hareket indeksidir (OMI). OMI, tüm hücrelerdeki ileri hareket sayısının, sistemdeki tüm parçaları işlemek için gereken toplam hareket sayısına oranı şeklinde tanımlanmıştır.

$$OMI = \frac{\text{Tüm hücrelerdeki ileri hareket sayısı toplamı}}{\text{Sistemdeki toplam hareket sayısı}} = \frac{\sum_{k=1}^C N_{fk}}{N_{tmov}} \quad (38)$$

Performans indekslerini incelersek, ileri hareketlerin fazla olması CMI'nin yüksek olmasına neden olacaktır. CMI, hücre içi hareketler hakkında çözümün kalitesini gösterir. CUI ise hücre içindeki boşlukları göstermektedir. Çünkü hücre kullanım oranı arttıkça hücre içindeki boşlukların azalacağı aşikardır. Diğer taraftan OMI ölçütü hücreler arası hareketlerin kapsamını göstermektedir. Böylelikle hem klasik hücre oluşturma'nın ölçütleri yani hücreler arası hareket ve hücre içindeki boşluklar hem de yerleşim tasarımının performansı ölçülmüş olur.

BEŞİNCİ BÖLÜM: AKIŞ MATRİSİ TEMELLİ SEZGİSEL ALGORİTMA İÇİN HÜCRE OLUŞTURMA VE YERLEŞİM DÜZENİ TASARIMI İÇİN BİR UYGULAMA

5. Giriş

Uygulamanın gerçekleştirildiği firma, 1980 yılında televizyon endüstrisi için yüksek gerilim transformatörü üretimiyle çalışmalarına başlamış 1982’de sac nüveli transformatör üretimi ile ürün yelpazesini 1992 yılına kadar 1VA’dan 400kVA 3 fazlı transformatörlere kadar hızla geliştirmiştir. 1988 yılından sonra kendi geliştirmiş olduğu bilgisayar yazılımı ile transformatör tasarımlarını hazırlamaya başlayan firma, yüksek frekanslı transformatörlerin üretimine başlamıştır. 1989 yılında Almanya’ya ilk ihracatını gerçekleştirmiş, 1991 yılında da Bağcılardaki 2000m²’lik kapalı çalışma alanına sahip olan binasına taşınmıştır.

1997 yılında ISO 9001 belgesini alarak kalite sistemini belgelendirmiş ve artan ürün çeşitleri ile kalite sistemi iyileştirme çalışmalarıyla üretim kapasitesini 2 katına çıkarmıştır. Firma, Eylül 2007 tarihi itibari ile inşaatı tamamlanan, imalat için kullanılabilir alanı 3000m²’den oluşan fabrikasına taşınmıştır.

Müşteri isteğine ve kullanım alanına göre tasarımı ve üretimi özel yapılan ürünler İngiltere, İspanya, Hollanda, İtalya, Polonya, Romanya, Mısır, Suriye, Litvanya, Bulgaristan, Yunanistan, İran ve Dubai gibi ülkelere ihraç edilmekte olup, diğer bir çok ülkelere de sanayi kuruluşlarımız tarafından ihracatı yapılan mamuller içerisinde kullanılmaktadır. Elektrik elektronik sanayi, tekstil, tıp, otomasyon, haberleşme gibi bir çok sektördeki 450 civarında yerli ve yabancı müşteriye binlerce çeşit CE ve VDE sertifikalı ürün ve teknik destek sunmaktadır.

Organizasyon olarak bakıldığında firmanın müşteri isteklerine göre ürün tasarımı bulunan bir ürün geliştirme bölümü bulunmaktadır. Ayrıca özellikle transformatörler için yeni teknoloji imkanları konusunda araştırma yapan bir de ARGE bölümü bulunmaktadır. Ürün gamında bulunan veya bulunmayan tüm müşteri sipariş ve teklif istekleri firmanın satış bölümü tarafından alınmaktadır. Daha sonra standart ürünlerin siparişleri için üretim planlama bölümü tarafından iş emirleri

açılmakta ve üretim bölümü bu iş emirlerine göre üretimi gerçekleştirip mamul depoya teslim etmektedir. Firmanın kalite kontrol bölümü satın alma bölümü tarafından satın alınan malzemeler için giriş kalite kontrolü, devam eden üretim için ara kontrol faaliyetlerini ve bitmiş ürünler için son kontrol faaliyetlerini yürütmektedir.

5.1.Ürün Gami

PCB tipi transformatörler: PCB Tipi Transformatörler izolasyonu dökümle sağlamlaştırılmış emniyet-izolasyon Transformatörleridir.

Kontrol transformatörleri: Kontrol Transformatörleri, çıkış gerilim değerinin her zaman istenen değer aralığında olmasını gerektiren koşullar için tasarlanmış özel transformatörlerdir.

Emniyet-izolasyon transformatörleri: Emniyet-Izolasyon Transformatörleri, insan sağlığının ön planda olduğu uygulamalar için tasarlanmış özel transformatörlerdir.

Üç fazlı transformatörler: Üç Fazlı Transformatörler, bütün üç fazlı sistemlerde kullanılmak için yapılmış genel amaçlı transformatörlerdir.

Motor sürücü giriş reaktörleri: Motor Sürücü Giriş Reaktörleri, üç fazlı motor sürücüleri ile şebeke arasına yerleştirilir.

Motor sürücü çıkış reaktörleri: Motor Sürücü Çıkış Reaktörleri, üç fazlı motor sürücüleri ile motor arasına yerleştirilir.

Harmonik filtre reaktörleri: Harmonik Filtre Reaktörleri, sistemde lineer olmayan yüklerden dolayı oluşan harmonik kirliliğin, kompanzasyon sistemlerine zarar vermesini engeller.

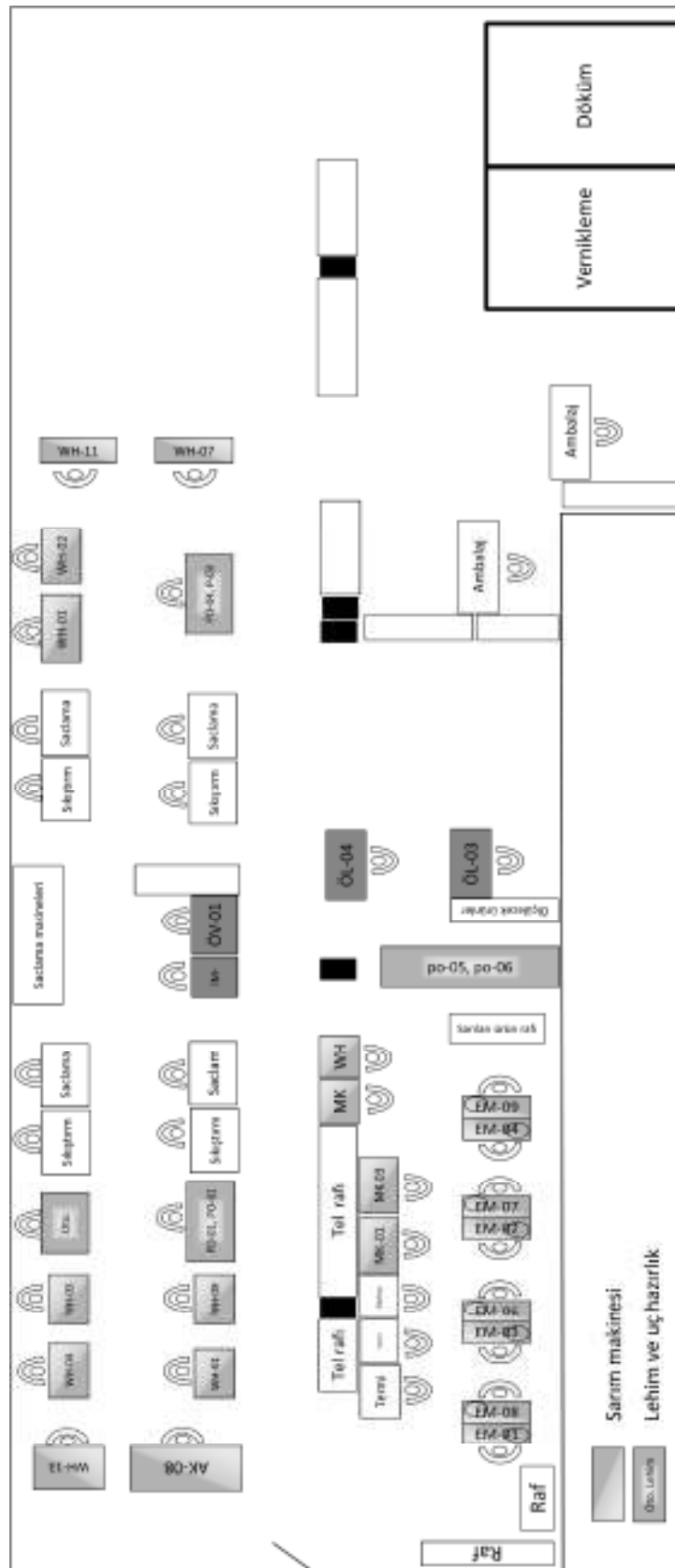
Tek fazlı reaktörler: Tek Fazlı Reaktörler, bir çok amaca hizmet edebilen genel amaçlı endüktans bobinleridir.

Yüksek frekans güç transformatörleri: Anahtarlama Güç Kaynağı (SMPS) transformatörleri, Sürücü Transformatörleri, Darbe Transformatörleri, Akım Transformatörleri.

5.2. Fabrikanın Mevcut Yerleşim Düzeni

Uygulama fabrikanın ferritli ve saclı transformatör üretim bölümünde gerçekleştirilmiştir. Mevcut yerleşim bu iki tip ürün grubuna göre yapılmıştır. Vernikleme ve döküm bölümleri yanıcı ve uçucu kimyasal girdiler kullanması nedeniyle ayrı bir bölümde bulunmaktadır. Hem ferritli hem de saclı ürünler vernik bölümünü ortak olarak kullanmaktadırlar. Mevcut yerleşimin en göze çarpan olumsuz yönü malzeme taşımalarının ve geriye hareketlerin fazlılığıdır. Mevcut yerleşim düzeni Şekil 32’de gösterilmiştir.

Uygulama birbirinden farklı operasyon talepleri ve sırası olan 27 ürün ve bunların operasyonlarını gerçekleştiren 17 iş istasyonu üzerinden yapılmıştır. İş akışları incelenen 27 parçanın operasyon sıralarını gösteren matris aşağıdaki gibidir.



Tablo 55. 27 makine ve 17 iş istasyonundan oluşan operasyon sıra matrisi.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
	2103001035K0206	2104801053E0204	2103201035K0208	2405502106K0140	2401005060E0400	2402000606E0403	2401900506K0134	2403000734M110	2404201506K0121	2404202006K0153	2402500741E0307	2401000834N0406	2401200934M1916	2103501035K0207	210480200600213	2103002316B0422	2104101534M0413	2403301306K0180	2104101519T0109	2402000606E0524	2103001016E0198	2622140834A1600	2401600534E0101	2103061017E3811	2103501516E0166	210520233201067	2103300257E6654
10	1	1		1			1			1	1	1	1	1	1	1			1		1			1	1	1	1
20	2		1												2	2	1				2						
30		2										2	2	2					2					2	2	2	2
40				2	1	1	2	1	1	2	2							1		1		1	1				
50	3		2													3											
60		3												3	3				3		3			3	3	3	3
70				3	2	2	3	2	2	3	3	3	3				2	2		2		2	2				
80	4	4	3											4	4	4	3		4		4			4	4	4	4
90	5	5	4											5	5	5	4		5		5			5	5	5	5
100	6	6	5											6	6	6	5		6							6	6
110	7	7	6											7	7	7	6		7					6		7	7
120				4	3	3	4	3	3	4	4							3		3		3	3				
130		8	7			4		4		5	5							4	8				4	7	7	8	
140	9													8	9	9	7				7						8
150	8	9	8			5		5		6	6				8	8		5	9				5		8	9	
160	10		9																		6				6		
170	11	10	10	5	4	6	5	6	4	7	7	4	4	9	10	10	8	6	10	4	8	4	6	8	9	10	9

Operasyonların açıklamaları aşağıdaki gibidir:

10: Karkas pin çıkartma / takma

20, 30, 40: Tel sarma

50,60,70: Kablo ucu hazırlama ve lehim

80: Saclama

90:Sıkıştırma, çerçeveleme, kutulama

100: İzolasyon testi

110: Voltaj-akım testi

120: Endüktans testi

130: Vernikleme

140: Döküm

150: Kurutma

160: Açık devre kontrolü

170: Etiket, ambalaj

5.3. Derece Sıra Algoritmasının Uygulanması

Tablo 56'da uygulamanın derece sıra algoritması ile çözümü görülmektedir. Makine parça matrisinde de görüldüğü gibi ikili düzendeki makine parça matrisi problemin çözümünde yetersiz kalmaktadır. Blok diyagonal matris tam olarak elde edilememektedir.

Tablo 56. Uygulamanın Derece Sıra Algoritmasıyla Çözümü.

	3	1	16	15	19	26	2	17	14	27	24	25	21	10	11	6	8	18	23	12	13	4	7	5	9	20	22
17	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
8	1	1	1	1	1	1	1	1	1	1	1	1	1														
9	1	1	1	1	1	1	1	1	1	1	1	1	1														
11	1	1	1	1	1	1	1	1	1	1	1																
10	1	1	1	1	1	1	1	1	1	1																	
15	1	1	1	1	1	1	1					1		1	1	1	1	1	1								
2	1	1	1	1				1					1														
5	1	1	1																								
16	1	1										1	1														
13	1				1	1	1				1	1		1	1	1	1	1	1								
1		1	1	1	1	1	1		1	1	1	1	1	1	1					1	1	1	1				
14		1	1	1				1	1	1			1														
6				1	1	1	1		1	1	1	1	1														
3					1	1	1		1	1	1	1								1	1						
7								1							1	1	1	1	1	1	1	1	1	1	1	1	1
4															1	1	1	1	1			1	1	1	1	1	1
12															1	1	1	1	1	1			1	1	1	1	1

5.4. CASE Algoritmasının Uygulanması

Benzerlik katsayı matrisinin oluşturulabilmesi için operasyon sıra matrisinin ara, başlangıç ve bitiş operasyonlarına göre dönüştürmemiz gerekmektedir.

$t_{jip}=0$ eğer $b_{jip}=0$

=1 =1 veya r_j

=2 aksi halde

kuralını uygularsak dönüşüm matrisi aşağıdaki gibi olur.

Tablo 57. Makine parça matrisi için dönüşüm matrisi.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
10	1	1		1			1			1	1	1	1	1	1	1			1		1			1	1	1	1
20	2		1												2	2	1			2							
30		2										2	2	2					2					2	2	2	2
40				2	1	1	2	1	1	2	2							1		1		1	1				
50	2		2													2											
60		2												2	2				2		2			2	2	2	2
70				2	2	2	2	2	2	2	2	2	2				2	2		2		2	2				
80	2	2	2											2	2	2	2		2		2			2	2	2	2
90	2	2	2											2	2	2	2		2		2			2	2	2	2
100	2	2	2											2	2	2	2		2							2	2
110	2	2	2											2	2	2	2		2					2		2	2
120				2	2	2	2	2	2	2	2							2		2		2	2				
130		2	2			2		2		2	2							2	2				2	2	2	2	
140	2													2	2	2	2				2						2
150	2	2	2			2		2		2	2				2	2		2	2				2		2	2	
160	2		2																	2					2		
170	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

(14) eşitliğine göre benzerlik katsayıları matrisi Tablo 58'deki gibi hesaplanmıştır.

1. iterasyonda 0 eşik değerinde 20 ve 30 nolu operasyonlar centroid olarak belirlenmiştir. Bu operasyonlar etrafında en yüksek benzerliğe sahip operasyonlar gruplandığında aşağıda verilen 2 hücre oluşur.

$$MC_1=20, 40, 50, 120, 140, 160, 170$$

$$MC_2=30, 10, 60, 70, 80, 90, 100, 110, 130, 150$$

Buna göre parça çekirdekleri aşağıdaki gibidir:

$$CS_1=\{01011000000101011\}$$

$$CS_2=\{10100111111010100\}$$

(22) eşitliğinden en büyük sc_{kj} değerlerine göre parçaların hücrelere atanmasından sonra makine parça matrisinin durumu Tablo 61'de görülmektedir.

Tablo 58. CASE benzerlik katsayıları matrisi.

	10	20	30	40	50	60	70	80	90	100	110	120	130	140	150	160
20	0,44															
30	0,77	0														
40	0,36	0	0													
50	0,13	0,44	0	0												
60	0,77	0,29	0,78	0	0											
70	0,38	0,08	0,17	0,87	0	0										
80	0,7	0,5	0,64	0	0,38	0,82	0,07									
90	0,7	0,5	0,64	0	0,38	0,82	0,08	1								
100	0,57	0,47	0,53	0	0	0,63	0,08	0,87	0,87							
110	0,62	0,44	0,6	0	0,43	0,7	0,08	0,92	0,92	0,95						
120	0,29	0	0	1	0	0	0,89	0	0	0	0					
130	0,51	0,09	0,48	0,5	0,13	0,48	0,44	0,48	0,48	0,36	0,43	0,5				
140	0,48	0,63	0,25	0	0,4	0,5	0,09	0,7	0,7	0,71	0,67	0	0			
150	0,53	0,29	0,35	0,45	0,35	0,43	0,41	0,59	0,59	0,58	0,56	0,46	0,85	0,29		
160	0,24	0,39	0,15	0	0,57	0,31	0	0,47	0,47	0,29	0,27	0	0,25	0,36	0,33	
170	0,77	0,35	0,6	0,65	0,18	0,6	0,79	0,74	0,74	0,64	0,67	0,71	0,65	0,51	0,76	0,34

1. iterasyonun bağ etkinliği:

Burada;

TOTOP₁=71, NOP₁=57

TOTOP₂=123, NOP₂=17

(17) eşitliğinden Kompaktlık=194/268=0,72

(20) eşitliğinden grup teknoloji etkinliği=(167-67)/167=0,6

Bağ etkinliği $\beta=0,72+0,6=1,32$

2. iterasyonda benzerlik katsayı eşiği bir üst değer olan 0,07'ye yükseltilir. Bu eşik değerinde mevcut centroid kümesine 70 ve 80. Operasyonlar eklendikten sonra oluşan 4 hücrenin operasyon gruplaması aşağıdaki gibi olur:

MC₁=20, 50

MC₂=10, 30, 130

MC₃=40, 70, 120, 170

MC₄=60, 80, 90, 100, 110, 140, 150, 160

Tablo 59. Parça çekirdeği 1'e göre SC_{ki} değerlerini gösteren tablo.

CS_1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	2	0	1	0	0	0	0	0	0	0	0	0	0	0	2	2	1	0	0	0	2	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	2	1	1	2	1	1	2	2	0	0	0	0	0	0	1	0	1	0	1	1	0	0	0	0
1	2	0	2	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	2	2	2	2	2	2	2	0	0	0	0	0	0	0	2	0	2	0	2	2	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	2	0	0	0	0	0	0	0	0	0	0	0	0	2	2	2	2	0	0	0	2	0	0	0	0	0	2
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	2	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	2	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Σx_i	9	1	6	5	4	4	5	4	4	5	5	1	1	3	5	7	4	4	1	4	7	4	4	1	3	1	3

Tablo 60. Parça çekirdeği 2'ye göre SC_{ki} değerlerini gösteren tablo.

CS_2	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
1	1	1	0	1	0	0	1	0	0	1	1	1	1	1	1	1	0	0	1	0	1	0	0	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	2	0	0	0	0	0	0	0	0	0	2	2	2	0	0	0	0	2	0	0	0	0	2	2	2	2
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	2	0	0	0	0	0	0	0	0	0	0	0	2	2	0	0	0	2	0	2	0	0	2	2	2	2
1	0	0	0	2	2	2	2	2	2	2	2	2	2	0	0	0	2	2	0	2	0	2	2	0	0	0	0
1	2	2	2	0	0	0	0	0	0	0	0	0	0	2	2	2	2	0	2	0	2	0	0	2	2	2	2
1	2	2	2	0	0	0	0	0	0	0	0	0	0	2	2	2	2	0	2	0	2	0	0	2	2	2	2
1	2	2	2	0	0	0	0	0	0	0	0	0	0	2	2	2	2	0	2	0	0	0	0	0	0	2	2
1	2	2	2	0	0	0	0	0	0	0	0	0	0	2	2	2	2	0	2	0	0	0	0	0	0	2	2
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	2	2	0	0	2	0	2	0	2	2	0	0	0	0	0	0	2	2	0	0	0	2	2	2	2	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	2	2	2	0	0	2	0	2	0	2	2	0	0	0	2	2	0	2	2	0	0	0	2	0	2	2	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Σx_i	11	17	12	3	2	6	3	6	2	7	7	5	5	13	13	11	10	6	17	2	7	2	6	13	13	17	13

Tablo 61. CASE algoritmasında 1. İterasyonda parçaların hücelere atanmasından sonra makine parça matrisinin durumu.

		4	5	7	9	20	22	1	2	3	6	8	10	11	12	13	14	15	16	17	18	19	21	23	24	25	26	27
M C ₁	20							2		1								2	2	1			2					
	40	2	1	2	1	1	1				1	1	2	2							1			1				
	50							3		2									3									
	120	4	3	4	3	3	3				3	3	4	4							3			3				
	140							9										8	9	9	7			7				8
	160							10		9													6			6		
	170	5	4	5	4	4	4	11	10	10	6	6	7	7	4	4	9	10	10	8	6	10	8	6	8	9	10	9
M C ₂	30								2						2	2	2					2			2	2	2	2
	10	1		1				1	1				1	1	1	1	1	1	1			1	1		1	1	1	1
	60								3								3	3				3	3		3	3	3	3
	70	3	2	3	2	2	2				2	2	3	3	3	3				2	2			2				
	80							4	4	3							4	4	4	3		4	4		4	4	4	4
	90							5	5	4							5	5	5	4		5	5		5	5	5	5
	100							6	6	5							6	6	6	5		6					6	6
	110							7	7	6							7	7	7	6		7			6		7	7
	130							8	7	4	4	5	5								4	8		4	7	7	8	
	150							8	9	8	5	5	6	6				8	8		5	9		5		8	9	
r _{i-1}		4	3	4	3	3	3	10	9	9	5	5	6	6	3	3	8	9	9	7	5	9	7	5	7	8	9	8
U		3	2	3	2	2	2	3	1	2	4	4	5	5	1	1	1	3	3	2	4	1	3	4	1	3	1	1
NOP ₁		2	1	1	1	1	1	5		6	3	3	3	3				6	5	5	3		3	3		2		
NOP ₂		1		1				2			1	1	2	2				1	2		1		1	1		1		
	Σr _{i-1} =	167																										
	ΣU=	67																										
	ΣNOP ₁ =	57																										
	ΣNOP ₂ =	17																										

Buna göre parça çekirdek vektörleri:

$$CS_1=\{0100100000000000\}$$

$$CS_2=\{10100000000010000\}$$

$$CS_3=\{00010010000100001\}$$

$$CS_4=\{00000101111001110\}$$

Parça atamaları yapıldıktan sonra makine parça matrisinin durumu Tablo 62'deki gibidir.

Tablo 62. İterasyon 2'de parçaların atanmasından sonra makine parça matrisinin durumu.

		12	13	4	5	6	7	8	9	10	11	18	20	22	23	1	2	3	14	15	16	17	19	21	24	25	26	27
MC ₁	20															2		1		2	2	1		2				
	50															3		2			3							
MC2	10	1	1	1			1			1	1					1	1		1	1	1		1	1	1	1	1	1
	30	2	2														2		2				2		2	2	2	2
	130					4		4		5	5	4			4		8	7					8		7	7	8	
MC3	40			2	1	1	2	1	1	2	2	1	1	1	1													
	70	3	3	3	2	2	3	2	2	3	3	2	2	2	2							2						
	120			4	3	3	4	3	3	4	4	3	3	3	3													
	170	4	4	5	4	6	5	6	4	7	7	6	4	4	6	11	10	10	9	10	10	8	10	8	8	9	10	9
MC4	60																3		3	3			3	3	3	3	3	3
	80															4	4	3	4	4	4	3	4	4	4	4	4	4
	90															5	5	4	5	5	5	4	5	5	5	5	5	5
	100															6	6	5	6	6	6	5	6				6	6
	110															7	7	6	7	7	7	6	7		6		7	7
	140															9			8	9	9	7		7				8
	150					5		5		6	6	5			5	8	9	8		8	8		9			8	9	
	160															10		9						6		6		
r _{i-1}	3	3	4	3	5	4	5	3	6	6	5	3	3	5	10	9	9	8	9	9	7	9	7	7	8	9	8	167
U	1	1	1		3	1	3		4	4	3			3	3	4	4	2	3	3	3	4	4	3	4	4	2	67
NOP1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
NOP2									3	3						5						5		4	4	5		29
NOP3					2		2		2	2	2			2								5						17
NOP4																1	1						1		1	1	1	6
TOTOP1		9																										
TOTOP2		38																										
TOTOP3		66																										
TOTOP4		81																										
TTOTOP		194																										

2. iterasyonun kompaktlık, grup teknoloji etkinliği ve bağ enerjisi değerleri aşağıdaki gibidir:

$$\text{Kompaktlık} = 194/246 = 0,78$$

$$\text{Grup teknoloji etkinliği} = (167-67)/167 = 0,6 \text{ buradan } \beta = 1,38 \text{ elde edilir.}$$

Bağ etkinliği 1. İterasyondan daha büyük çıktığından iterasyonlara devam edilir.

Tablo 63. CASE algoritması 3. iterasyon sonucu makine parça matrisi.

		1	1							1	1	1	2	2	2	1	3	1	2	1	1	2	2	2	2	1	1	2
		2	3	4	5	6	7	8	9	0	1	8	0	2	3	1	3	6	2	5	9	5	6	1	4	4	7	7
MC 1	20															2	1	2		2				2			1	
	50															3	2	3										
MC 2	10	1	1	1			1			1	1					1		1	1	1	1	1	1	1	1	1		1
	30	2	2																2		2	2	2		2	2		2
	130					4		4		5	5	4			4		7		8		8	7	8		7			
MC 3	40			2	1	1	2	1	1	2	2	1	1	1	1													
	70	3	3	3	2	2	3	2	2	3	3	2	2	2	2												2	
	120			4	3	3	4	3	3	4	4	3	3	3	3													
	170	4	4	5	4	6	5	6	4	7	7	6	4	4	6	1	1	1	1	1	1	1		1				
MC 4	60																		3	3	3	3	3	3	3	3		3
	80															4	3	4	4	4	4	4	4	4	4	4	3	4
	90															5	4	5	5	5	5	5	5	5	5	5	4	5
	150					5		5		6	6	5			5	8	8	8	9	8	9	8	9					
	160															1	0	9					6		6			
MC 5	100															6	5	6	6	6	6		6			6	5	6
	110															7	6	7	7	7	7		7		6	7	6	7
	140															9		9		9				7		8	7	8

3. iterasyonda eşik değeri 0,08'e yükseltilir. Yapılan hesaplamalar sonucunda bağ etkinliği 1,23 elde edildiğinden algoritma bu iterasyonda durdurulmuştur. Elde edilen makine parça matrisi Tablo 63'de görülmektedir.

3. iterasyonun kompaktlık ve grup teknoloji etkinliği değerleri aşağıdaki gibidir:

$$\text{Kompaktlık} = 194/264 = 0,73$$

$$\text{Grup teknoloji etkinliği} = 0,5$$

3. iterasyonda oluşan makine hücreleri aşağıdaki gibidir:

$$MC_1 = 20, 50$$

$$MC_2 = 30, 10, 130$$

$$MC_3 = 70, 40, 120, 130$$

$$MC_4 = 80, 60, 90, 150, 160$$

$$MC_5 = 100, 110, 140$$

5.5. Akış Matrisi Tabanlı Sezgisel Algoritmanın Uygulanması

(33) eşitliğine göre f_{ij} akış matrisi aşağıdaki gibi hesaplanmıştır. 0 değerine sahip alanlar boş bırakılmıştır.

Tablo 64. Operasyon akış matrisi.

	10	20	30	40	50	60	70	80	90	100	110	120	130	140	150	160	170
10		4	9	4													
20					3	2	1										
30						7	2										
40							12										
50								3									
60								9									
70								1				12					2
80									13								
90										10	1					2	
100											10						
110													5	3	3		
120													6				6
130															11		1
140																1	6
150														3		1	10
160													1	1			2
170																	

Algoritmanın işletilmesiyle elde edilen hücreler:

1. Hücre: 60, 80, 90, 100, 110, 140, 160
2. Hücre: 40, 70, 120
3. Hücre: 130, 150, 170
4. Hücre: 10, 30
5. Hücre: 20, 50

Meydana gelen hücrelerin CASE algoritmasında verilen performans ölçümlerine göre performans değerleri aşağıdaki gibidir.

$$\text{Kompaktlık} = (167 - 59) / 167 = 0,64$$

$$\text{Grup teknoloji etkinliği} = 194 / 204 = 0,95$$

Önerilen yöntem sonucunda işletmenin yeni yerleşim düzeni Şekil 33’de görülebilir. Burada vernikleme ve döküm istasyonları patlayıcı ve yanıcı kimyasalların fazlalığı nedeniyle daha önceden sabit olarak tasarlanmış ve birer oda içine yerleştirilmiştir. Bundan dolayı diğer hücreler bu istasyonların etrafına taşınmıştır.

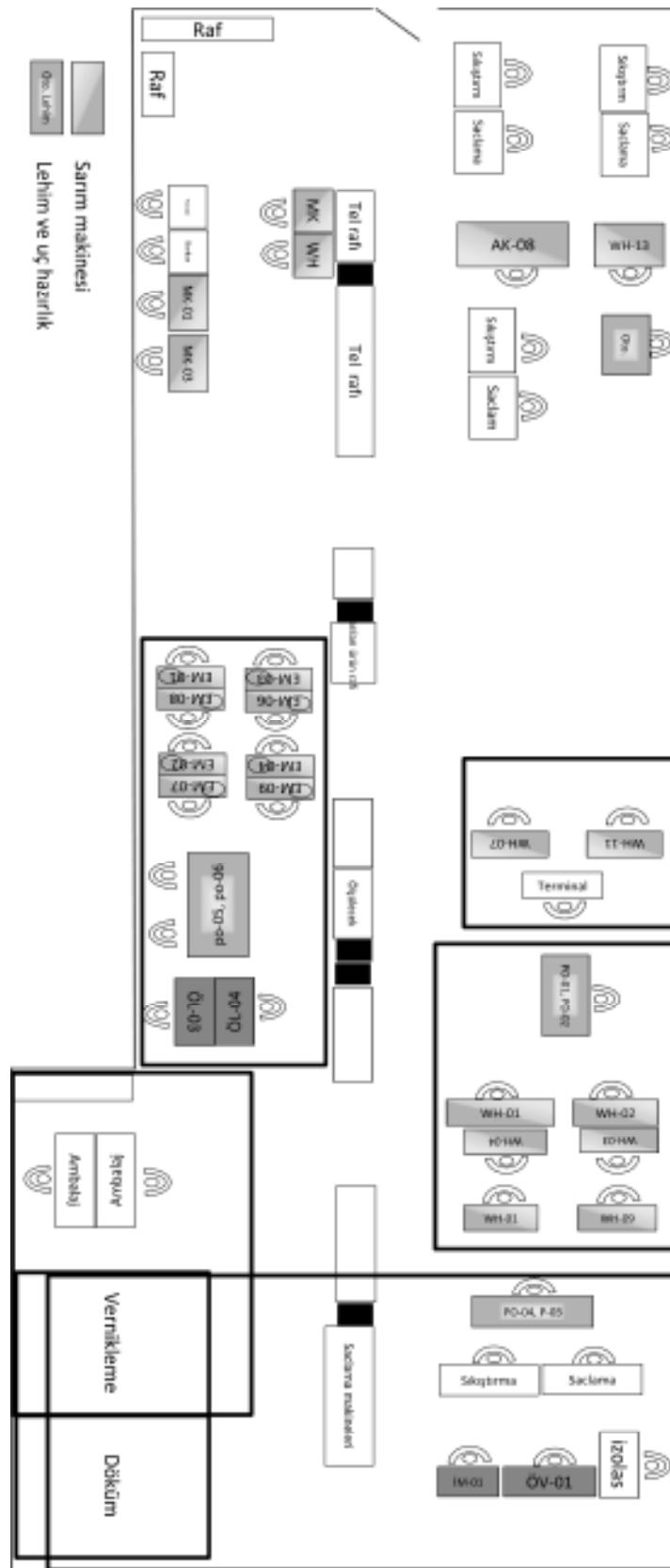
Tablo 65. Akış matrisi tabanlı sezgisel algoritma uygulaması sonrası makine parça matrisi.

	26	27	4	13	5	2	8	12	7	9	10	11	6	1	3	14	15	17	19	23	24	16	18	20	21	22	25	
10	1	1	1	1	1	1	1	1		1	1	1	1	1		1		1						1	1			
30	2	2	2	2		2	2	2			2	2																
60			3	3	3	3	3	3		3	3	3																
80			4	4	4	4	4	4	3	4	4	4	4	4	3													
90			5	5	5	5	5	5	4	5	5	5	5	5	4													
100			6	6	6	6	6	6	5				6	6	5													
110			7	7	7	7	7	7	6		6		7	7	6													
140			8	8	9				7	7			9	9														
160										6		6		10	9													
20					2				1	2			2	2	1													
50													3	3	2													
40																2	1	2	1	1	1	1	1	1	2	2	1	1
70	3	3							2							3	2	3	2	2	2	2	2	2	3	3	2	2
120																4	3	4	3	3	3	3	3	3	4	4	3	3
130						8	8	8			7	7			7								4	4	5	5	4	4
150					8	9	9	9				8	8	8	8								5	5	6	6	5	5
170	4	4	9	9	10	10	10	10	8	8	8	9	10	11	10	5	4	5	4	4	4	4	6	6	7	7	6	6

Performans hesaplamalarının özeti aşağıdaki gibidir.

Tablo 66. Performans hesaplamalarının özeti.

Akış Matrisi Tabanlı Sezgisel Yöntem							
CMI_1	1	P_1	2	$CMI_1 \times P_1$	2	CUI_1	1
CMI_2	0,98	P_2	10	$CMI_2 \times P_2$	9,75	CUI_2	0,729
CMI_3	1	P_3	3	$CMI_3 \times P_3$	3	CUI_3	1
CMI_4	1	P_4	6	$CMI_4 \times P_4$	6	CUI_4	1
CMI_5	1	P_5	6	$CMI_5 \times P_5$	6	CUI_5	1
ACMI=	0,99						
ACUI=	0,95						
OMI=	0,41						



SONUÇ VE DEĞERLENDİRME

Gerek sürekli, gerekse atölye tipi üretim yapan işletmeler, üretimdeki kayıpları azaltmak için çeşitli metotlar kullanmaktadırlar. Atölye tipi üretimde bu kayıpları azaltmanın etkin yollarından biri olan grup teknolojisi ve onun imalata uygulanmış şekli olan hücresel imalat sistemleri de 1960'lı yıllarda Sovyetler Birliği'nde ortaya çıktığından bu yana çok önemli aşamalar kaydetmiştir. Hücresel imalat sistemlerinin en zor problemlerinden olan hücre oluşturma veya daha açık şekliyle makine ve parça gruplama yöntemleri de oldukça geliştirilmiş ve gerçek imalat ortamlarının gereklerini daha etkin biçimde karşılayan bugünkü halini almıştır.

Bu tezin üçüncü bölümünde incelediğimiz 0-1 makine parça verisini kullanan ve daha çok 1980'lerde gelişim gösteren makine parça gruplama yöntemleri, daha çok sistemin içinde doğal olarak bulunan grupları bulmakta daha başarılıdırlar. Bu bölümde de gördüğümüz gibi eğer mevcut sistemde böyle doğal gruplar mevcutsa, 0-1 makine parça verisini kullanan metotlar bu grupları makine parça matrisinin diyagonalinde toplar ve analizcinin ihtiyaç duyduğu grupları gösterir. 0-1 makine parça verisini kullanan metotların, parçaların operasyon ihtiyaçlarına göre makineler arasındaki benzerlik katsayılarını kullanarak geliştirdiği algoritmalar ise yine gerçek üretim ortamlarının kısıtları karşısında yetersiz kalmıştır.

Sistemin içinde doğal olarak bulunmayan makine gruplarının olduğu problemlerde 0-1 makine parça matrisi verisini kullanan metotların çok fazla çözüm üretmediği yapılan uygulamada da görülmüştür. Bu durum, çalışmaların operasyon sıra verisini kullanan metotlara yönelmesine neden olmuştur. Operasyon sıra verisinin kullanımı analizeciye çözümün her aşamasında hücreler arası ve hücre içinde gelişen akışlar hakkında sürekli bilgi sağladığından hücre oluşumuna müdahaleyi kolaylaştırmaktadır. Böylece analizci gerektiğinde çözüme müdahale ederek düzeltmeler yapabilmektedir.

Günümüzde imalat yapan işletmeler yalın üretimi gittikçe daha çok benimsediklerinden, mevcut kayıpları (7 büyük kayıp) sürekli olarak azaltmaya çalışmaktadırlar. Böyle bir imalat sisteminde hücre oluşturma yanı sıra sistemin içindeki kayıpları azaltmak, yani malzeme hareketlerini, insan hareketlerini ve bekleme sürelerini en küçükleme de sistemin verimliliği açısından önemli bir kriter

olduğundan, en uygun yerleşim düzenini tespit etmek, yapılan çalışmanın başarısını arttıran ve hücre oluşturmada sonra dikkate alınarak tamamlanması gereken bir diğer çalışmadır. Yapılan makine gruplama başarılı olduğu ölçüde istisnai elemanları yani hücreler arası hareketleri azaltacaktır. Ancak uygun yerleşim düzeni tespit edilmediğinde özellikle hücreler arası hareketler önemli verim kayıplarına neden olmaktadır. Bunun yanında hücre içindeki yerleşim düzeni doğru yapılmadığı takdirde, gereksiz beklemler ve fazladan malzeme ve insan hareketleri artmakta, bu da çevrim sürelerinin artmasına neden olmaktadır. Bu nedenlerden dolayı hücre oluşturma için geliştirilen yöntemler daha çok hücre oluşturma ve yerleşim düzenini eş zamanlı olarak dikkate alan metotlar üzerinde yoğunlaşmaktadır.

Bu tezin üçüncü bölümünde incelediğimiz operasyon sıra verisini kullanan metotlarda da gördüğümüz gibi, NP zor problemlerin çözümünde kullanılan meta sezgisel metotlardan tavlama benzetimi, genetik algoritmalar, karınca kolonisi, sinir ağları yaklaşımı, yasak arama algoritmaları son dönemde en çok kullanılan hücre oluşturma yöntemleridir. Bu metotların bir arada kullanıldığı hibrit algoritmalar da literatürde önemli bir yer tutmaya başlamışlardır. Ancak bu yöntemlerin çözümünün uzun zaman alması ve hesaplama karmaşıklığı en büyük dezavantajlarıdır. Bu algoritmaların yanında dördüncü bölümde incelediğimiz akış matrisi tabanlı sezgisel metot hücre oluşturma ve yerleşim düzeni tasarımını eş zamanlı olarak ele alan basit ve etkin bir metot olarak karşımıza çıkmaktadır. Metodun operasyon sıra verisini kullanması analizciye oluşturulan çözüm hakkında daha fazla bilgi vermekte ve algoritmanın tamamlanıp hücreler oluşturulduktan sonra gerektiğinde akış verisine göre ince ayar yapma imkanı vermektedir. Beşinci bölümde yapılan uygulama çalışması sonuçlarında da görüldüğü gibi yöntem yakın zamanda geliştirilen sıra verisi kullanan CASE algoritmasıyla kıyaslandığında daha iyi performans gösterdiği görülmektedir.

Aşağıdaki tabloda CASE ve akış matrisi tabanlı sezgisel algoritmayla yapılan makine hücresi oluşturma çalışmasının performans karşılaştırması yer almaktadır.

Tablo 67. Uygulamada yapılan hücre oluşturma performanslarının karşılaştırılması.

Akış Matrisi Tabanlı Sezgisel Yöntem								CASE Yöntemi							
CMI_1	1	P_1	2	$CMI_1 \times P_1$	2	CUI_1	1	$CMI1$	1	$P1$	3	$CMI1 \times P1$	3	$CUI1$	1
CMI_2	0,98	P_2	10	$CMI_1 \times P_2$	9,75	CUI_2	0,73	$CMI2$	1	$P2$	2	$CMI1 \times P2$	2	$CUI2$	0,67
CMI_3	1	P_3	3	$CMI_1 \times P_3$	3	CUI_3	1	$CMI3$	1	$P3$	12	$CMI1 \times P3$	12	$CUI3$	1
CMI_4	1	P_4	6	$CMI_1 \times P_4$	6	CUI_4	1	$CMI4$	1	$P4$	6	$CMI1 \times P4$	6	$CUI4$	0,8
CMI_5	1	P_5	6	$CMI_1 \times P_5$	6	CUI_5	1	$CMI5$	1	$P5$	3	$CMI1 \times P5$	3	$CUI5$	1
ACMI=	0,99							ACMI=	0,96						
ACUI=	0,95							ACUI=	0,89						
OMI=	0,41							OMI=	0,33						

İkili makine parça matrisini kullanan literatürde en çok kullanılan yöntemlerden derece sıra algoritması ile denenen çözümde matrisin blok diyagonalinde belirgin makine grupları oluşturulamamıştır. Bu da sistemin içinde doğal grupların bulunmadığını göstermektedir.

Tablo 67’de de görüldüğü gibi akış tabanlı sezgisel algoritma ile yapılan hücre oluşturma CASE algoritmasıyla karşılaştırınca daha iyi performans göstermektedir. ACMI değerinin yüksek olması hücrelerdeki ileri hareketlerin fazlalığını göstermektedir. CMI değeri hücre içi hareketlere odaklanırken, OMI değeri hücreler arası harekete odaklıdır. OMI indeksinin de yüksek olması akış matrisi tabanlı algoritmanın daha az hücreler arası hareket sağladığını gösterir.

Önerilen yöntem sonuç üzerindeki performansının yanı sıra hesaplama kolaylığı ile de göze çarpmaktadır. Diğer yöntemlerdeki iteratif hesap yoğunluğu önerilen algoritmada yerini basit ve işlevsel hesaplamalara bırakmaktadır.

BİBLİYOGRAFYA

Aktürk, M. S., & Balköse, H. O. (1996). Part-machine grouping using a multi objective cluster analysis. *International Journal of Production Research* , 34 (8), 2299-2315.

Al-Sultan, K. S., & Fedjki, C. A. (1997). A genetic algorithm for the part family formation. *Production Planning & Control* , 8 (8), 788-796.

Ayan, T. Y. (2009). Kaynak kısıtlı çoklu proje programlama problemi için tavlama benzetimi algoritması. *Atatürk Üniversitesi İktisadi ve İdari Bilimler Dergisi* , 23 (2), 101-117.

Ballakur, A., & Steudel, H. (1987). A within-cell utilization based heuristic for designing cellular manufacturing systems. *International Journal of Production Research* , 25 (5), 639-665.

Boulif, M., & Atif, K. (2006). A new branch-&-bound-enhanced genetic algorithm for the manufacturing cell formation problem. *Computers & Operations Research* , 33, 2219-2245.

Burbidge, J. L. (1992). Change to group technology: process organization is obsolete. *International Journal of Production Research* , 30 (5), 1209-1219.

Chan, H. M. (1981). Group technology as a manufacturing philosophy. Birmingham, İngiltere: Aston üniversitesi.

Chandrasekharan, M., & Rajagopalan, R. (1986). MODROC: an extention of rank order clustering for group technology. *International Journal of Production Research* , 24 (5), 1221-1233.

Elmas, P. D. (2007). *Yapay Zeka Uygulamaları*. Ankara: Seçkin Yayıncılık.

Gallagher, C. C., Grayson, T. J., & Phillips, A. (1971). A review of the development and application of group technology at the University of Birmingham. *International of Production Research* , 9 (2), 229-237.

George, A. P., Rajendran, C., & Ghosh, S. (2003). An analytical-iterative clustering algorithm for cell formation in cellular manufacturing systems with ordinal-level and ratio-level data. *International Journal of Advanced Manufacturing Technologies* (22), 125-133.

Harhalakis, G., Nagi, R., & Proth, J. M. (1990). An efficient heuristic in manufacturing cell formation for group technology applications. *International Journal of Production Research* , 28 (1), 185-198.

Heragu, S. S. (1994). Group technology and cellular manufacturing. *IEEE Transactions on Systems, man and cybernetics* , 24 (2), 203-215.

Heragu, S. S., & Gupta, Y. P. (1994). A heuristic for designing cellular manufacturing facilities. *International Journal of Production Research* , 32 (1), 125-140.

Heragu, S. S., & Kakuturi, S. R. (1997). Grouping and placement of machine cells. *IIE Transactions* (29), 561-571.

Houshyar, A., & Ginnis, L. M. (1990). A heuristic for assigning facilities to locations to minimize WIP travel distance in a linear facility. *International Journal of Production Research* , 28 (8), 1485-1498.

Kamrahi, A., & Parsaei, H. (1994). A methodology for the design of manufacturing systems using group technology. *Production Planning & Control* , 5 (5), 450-464.

Keskintürk, T., & Küçük, B. (2006). Karışık modellenli montaj hatlarının genetik algoritma kullanılarak dengelenmesi. *Yönetim (İ.Ü. İşletme Fakültesi)* (53), 52-63.

King, J. R. (1980). Machine-component grouping in production flow analysis: an approach using a rank order clustering algorithm. *International Journal of Production Research* , 18 (2), 213-232.

Kusiak, A. (1987). The generalized group technology concept. *International Journal of Production Research* , 25 (4), 561-569.

Kusiak, A., & Chow, W. S. (1988). Decomposition of manufacturing systems. *IEEE Journal of Robotics and Automation* , 4 (5), 457-471.

Lee, S.-D., & Chiang, C.-P. (2001). A cut tree based approach for clustering machine cells in the bidirectional linear flow layout. *International Journal of Production Research* , 39 (15), 3491-3512.

Mahdavi, I., Shirazi, B., & Paydar, M. M. (2008). A flow matrix-based heuristic algorithm for cell formation and layout design in cellular manufacturing

system. *International Journal of Advanced Manufacturing Technologies* (39), 943-953.

Mansouri, S. A., Husseini, S. M., & Newman, S. (2000). A review of the modern approaches to multi-criteria cell design. *International Journal of Production Research* , 38 (5), 1201-1218.

Nair, G. J., & Narendran, T. T. (1998). CASE: a clustering algorithm for cell formation with sequence data. *International Journal of Production Research* , 36 (1), 157-179.

Offodile, Mehrez, & Grznar. (1994). Cellular Manufacturing: a taxonomic review framework. *Journal of Manufacturing Systems* , 13 (3), 196-220.

Parsaei, H. R., Kolli, S., & Hanley, T. R. (1997). *Manufacturing decision support systems*. Londra: Chapman & Hall.

Rajagopalan, R., & Batra, J. L. (1975). Design of cellular production systems A graph-theoretic approach. *International Journal of Production Systems* , 13 (6), 567-579.

Reisman, A., Kumar, A., Motwani, J., & Cheng, C. H. (1997). Cellular manufacturing: a statistical review of the literature. *Operations Research* , 45 (4).

Saraç, T., & Özçelik, F. (2007). Alternatif rotaların varlığında üretim hücrelerinin genetik algoritma kullanılarak oluşturulması. *Endüstri Mühendisliği Dergisi* , 17 (4), 22-36.

Sarker, B. R., & Xu, Y. (1998). Operation sequences-based cell formation methods: A critical survey. *Production Planning & Control* , 9 (8), 771-783.

Selim, H. M., Askin, R. G., & Vakharia, A. J. (1998). Cell formation in group Technology: Review, evaluation and directions for future research. *Computers & Industrial Engineering* , 34 (1), 3.

Sezen, P. D. (2007). *Yöneylem Araştırması*. Bursa: Ekin.

Srinivasan, G., & Narendran, T. T. (1991). GRAFICS-a nonhierarchical clustering algorithm for group technology. *International Journal of Production Research* , 29 (3), 463-478.

Srinivasan, G., Naredran, T. T., & Mahadevan, B. (1990). An assignment model for the part families in group technology. *International Journal of Production Research* , 28 (1), 145-152.

Waghodekar, P. H., & Sahu, S. (1984). Machine component cell formation in group technology: MACE. *International Journal of Production Research* , 22 (6), 937-948.

Wang, T. Y., & K. B. Wu, Y. W. (2001). A simulated annealing algorithm for facility layout problems under variable demand in Cellular Manufacturing Systems. *Computers in Industry* , 46, 181-188.

Wemmerlov, U., & Hyer, N. (1989). Cellular manufacturing in the U.S. industry: a survey of users. *Interntional Journal of Production Research* , 27 (9), 1512.

Won, Y., & Kim, S. (1997). Multiple criteria clustering algorithm for solving the group technology problem with multiple process routings. *Computers & Industrial Engineering* , 32 (1), 207-220.

Xambre, A. R., & Vilarinho, P. M. (2003). A simulated annealing approach for manufacturing cell formation with multiple identical machines. *European Journal of Operational Research* (151), 434-446.

Yamak, Y. D. (1993). *Üretim Yönetimi*. İstanbul: Alfa Basım.

Yin, Y., & Yasuda, K. (2005). Similarity coefficient methods applied to the cell formation problem: a comparative investigation. *Computers & Industrial Engineering* (48), 471-489.

Yin, Y., & Yasuda, K. (2006). Similarity coeffiient methods applied to the cell formation problem: A taxonomy and review. *International Journal of Production Economics* (101), 329-352.