

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**TEK MODELLİ DETERMİNİSTİK MONTAJ HATTI
DENGELEME PROBLEMLERİNE GENETİK
ALGORİTMA İLE ÇÖZÜM YAKLAŞIMI**

**YÜKSEK LİSANS TEZİ
End. Müh. Raşit ÖZKAN**

Anabilim Dalı : ENDÜSTRİ MÜHENDİSLİĞİ

Programı : ENDÜSTRİ MÜHENDİSLİĞİ

MAYIS 2003

**TEK MODELLİ DETERMİNİSTİK MONTAJ HATTI
DENGELEME PROBLEMLERİNE GENETİK
ALGORİTMA İLE ÇÖZÜM YAKLAŞIMI**

**YÜKSEK LİSANS TEZİ
End.Müh. Raşit ÖZKAN
(507001215)**

**Tezin Enstitüye Verildiği Tarih : 05 Mayıs 2003
Tezin Savunulduğu Tarih : 28 Mayıs 2003**

**Tez Danışmanı : Yrd.Doç.Dr. Murat BASKAK
Diğer Jüri Üyeleri Doç.Dr. Cengiz GÜNGÖR (İ.T.Ü)
Doç.Dr. Demet BAYRAKTAR (İ.T.Ü)**

MAYIS 2003

ÖNSÖZ

Dünya nüfusunun giderek arttığı ve buna bağlı olarak eldeki varolan kaynakların giderek artan bir hızda tüketildiği bir gerçektir. Dünya nüfusundaki bu artış doğal olarak her alanda tüketimin de artması demek olduğundan, buna bağlı olarak üretimin de artması kaçınılmazdır.

Artan üretim ve azalan kaynaklar içinde bulunduğumuz yüzyılda insanoğlunu kaynakların daha verimli kullanılması için araştırmaya ve geleneksel üretim yöntemlerini günümüzde daha verimli duruma getirecek çözümler geliştirmeye zorlamaktadır.

Bu yüzyılın ortalarında ortaya atılan “montaj hatlarının dengelenmesi” düşüncesi, yukarıda sözedilen nedenlerden dolayı kaynakları daha verimli kullanabilme çabasının bir sonucudur. Montaj Hattı Dengeleme problemleriyle hem ürün tasarımı, hem de ürün ömrü aşamasında sıklıkla karşılaşılmaktadır.

Montaj Hattı Dengeleme, kontrol edilemeyen etmenlerin oluşturduğu ve ürün model değişimine bağlı olarak ortaya çıkan üretimin değişik evrelerindeki dengesizliklerin çözümünde kullanılır. Zayıf bir tasarımın ortaya çıkardığı azımsanmayacak maddî kayıp sonuçlarını düşünürsek, montaj hatlarının etkin bir tasarıma sahip olması ve kontrol edilmesi bu konudaki sorumluluğu üstlenmiş yöneticiler için son derece önemlidir.

Bundan hareketle, bu tez çalışmasında montaj hattı dengeleme çalışmalarının geçtiği aşamaların gözlemlenmesini ve yapay zekâ kategorisinde yer alan genetik algoritma ile çözüm yaklaşımını tanıtmayı amaçladım. Çalışmamda beni yönlendirerek değerli yardımlarını esirgemeyen hocam Sn. Yrd. Doç. Dr. Murat BASKAK’a, Arçelik-LG Klima İşletmesi’ndeki uygulama çalışması için bana yardımcı olan montaj takım lideri Sn. Aylin ATAOL ve takım mühendisi okul arkadaşım Sn. Tansu YEŞİLSOY’a teşekkür eder, tezimin bu alanda çalışmalarını sürdürecektir araştırmacıların yararlanabileceği bir kaynak olmasını temenni ederim.

Mayıs 2003

Raşit ÖZKAN

İÇİNDEKİLER

KISALTMALAR	vi
TABLO LİSTESİ	vii
ŞEKİL LİSTESİ	viii
SEMBOL LİSTESİ	xi
ÖZET	x
SUMMARY	xi
1. GİRİŞ	1
1.1. Giriş ve Çalışmanın Amacı	1
2. ÜRETİM VE MONTAJ HATLARI	3
2.1. Üretim	3
2.2. Üretim Tipleri	5
2.2.1. Genel Bilgi	5
2.2.2. Sürekli(Seri) Üretim	5
2.2.3. Kesikli(Parti) Üretim	6
2.2.4. Proje(Sipariş) Üretim	7
2.3. Montaj Hatlarının Üretimdeki Yeri	7
2.4. Montaj Hattı ve Tipleri	9
2.4.1. Genel Bilgi	9
2.4.2. Mekanik Olmayan Hatlar	12
2.4.3. Hareketli Bant Hatları	12
3. MONTAJ HATLARI VE HAT Dengeleme	13
3.1. Montaj Hatları Ve Hat Dengeleme Kavramı	13
3.2. Montaj Hatlarının Dengelenmesinin Amaçları	15
3.3. Montaj Hatlarının Dengelenmesini Etkileyen Temel Etmen ve Kısıtlar	16
3.4. Montaj Hatlarının Dengelenmesinde Kullanılan Temel Kavramlar	18
3.4.1. İş Ögesi	18
3.4.2. İş İstasyonu	18
3.4.3. Toplam İş Süresi	18
3.4.4. İş İstasyonu Süresi	19
3.4.5. Çevrim Süresi	19
3.4.6. Gerekli En Az İş İstasyonu Sayısı	20
3.4.7. Ortalama İş İstasyonu Süresi	21

3.4.8. Teknolojik Öncelik Diagramı	22
3.4.9. Öncelik Matrisi	22
3.4.10. Esneklik Oranı	23
3.4.11. Denge Kaybı	24
3.4.12. Hat Etkinliği	25
3.5. Montaj Hatlarında Model Sayısının Etkisi	26
3.6. Basit ve Genel Montaj Hattı Problemlerinin Sınıflandırılması	28
3.7. Montaj Hattı Dengeleme Yöntemlerinin Sınıflandırılması	30
3.7.1. Probleme Göre Sınıflandırma	30
3.7.2. Çözüm Yaklaşımına Göre Sınıflandırma	30
3.7.2.1. Sezgisel(Bulgusal) Yöntemler	30
3.7.2.2. Analitik Yöntemler	31
3.7.2.3. Benzetim(Simulasyon) Teknikleri	32
3.7.3. İşlem sürelerinin Deterministik Olmasına Göre Sınıflandırma	33
3.8. Konum Ağırlıklı Dengeleme Tekniği	34
3.9. Aşamalı Sıralamayla Çözüm	36
3.10. Öncelik Diyagramı Çözümü	36
4. GENETİK ALGORİTMALAR	37
4.1. Genetik Algoritmanın Tarihçesi	37
4.2. Kuramsal Temeller	41
4.2.1. Genetik Algoritmanın Tanımı	41
4.2.2. Genetik Algoritmalarla İlgili Temel Kavramlar	42
4.2.3. Genetik Algoritmaların Çalışma İlkesi	44
4.2.4. Genetik Algoritmada Kullanılan Operatörler	48
4.2.4.1. Giriş	48
4.2.4.2. Parametre Kodlama Operatörü	48
4.2.4.3. Üreme Operatörü	51
4.2.4.4. Crossover (Çaprazlama) Operatörü	53
4.2.4.5. Mutasyon Operatörü	55
4.3. Genetik Algoritma Parametreleri	57
4.3.1. Genel Bilgi	57
4.3.2. Populasyon Büyüklüğü	57
4.3.3. Crossover(Çaprazlama) Olasılığı	57
4.3.4. Mutasyon Olasılığı	58
4.4. Genetik Algoritmalarda Parametre Optimizasyonu	58
4.4.1. Genel Bilgi	58
4.4.2. Başlangıç Populasyonunun Optimizasyonu	59

4.4.3. Seçim Operatörü Optimizasyonu	59
4.4.4. Çaprazlama Operatörü Optimizasyonu	59
4.4.5. Mutasyon Operatörü Optimizasyonu	60
4.4.6. Populasyon Büyüklüğünün Optimizasyonu	60
4.5. Genetik Algoritmaların Kullanılma Nedenleri	60
4.5.1. Genel Bilgi	60
4.5.2. Genetik Algoritmaların Klasik Yöntemlerden Farkları	62
4.6. Uygulama Alanları	63
5. MONTAJ HATTI DENGELEME-GENETİK ALGORİTMA İLİŞKİSİ	64
5.1. Giriş	64
5.2. Genetik Algoritmanın MHD Uygulama Karakteristikleri	67
5.2.1. Kromozom Temsili	67
5.2.1.1. Genel Bilgi	67
5.2.1.2. İş Ögesi Odaklı Kodlama	67
5.2.1.3. Öncelik Sıralı Kodlama	68
5.2.1.4. Bölme Esaslı Kodlama	69
5.2.1.5. Önerilen Kodlama Yapısı ve Avantajları	69
5.2.2. Başlangıç Populasyonu ve Büyüklüğü	69
5.2.3. Uygunluk Fonksiyonu	70
5.2.4. Çaprazlama ve Mutasyon	71
5.2.4.1. Genel Bilgi	71
5.2.4.2. İki Kesim Noktalı Sıralama Çaprazlama	72
5.2.4.3. Scramble (karıştırılmış) Mutasyon	74
5.2.5. Ölçeklendirme(Scaling)	75
5.2.6. Seçim Prosedürü	75
5.2.7. Elitizm	75
5.2.8. Dinamik Bölümleme	76
5.2.8.1. Giriş	76
5.2.8.2. Yürütme	77
5.2.9. Durma Şartı	79
6. LİTERATÜR ARAŞTIRMASI	80
6.1. Giriş	80
6.2. Genel Olarak Genetik Algoritma Yaklaşımlı Çözümler	83
6.3. Çok Amaçlı MHD Genetik Algoritma Çalışmaları	88
6.4. Simülasyon Esaslı Genetik Algoritma Çalışmaları	90
6.5. Bulanık Mantık Tabanlı Genetik Algoritma Çalışmaları	91

7. UYGULAMA	92
7.1. Giriş	92
7.2. Firma Tanıtımı	92
7.3. Genetik Algoritmali Optiline Yazılım Tanıtımı	93
7.4. Uygulama Problemi	96
7.4.1. Problemin Tanımı	96
7.4.2. Varolan Durum Analizi	98
7.4.3. Problemin Öncelik Diyagramı	99
7.5. Problemin Optiline Kullanılarak Çözümü	100
7.6. Sonuçların Değerlendirilmesi	103
8. SONUÇLAR ve TARTIŞMA	106
KAYNAKLAR	107
ÖZGEÇMİŞ	113

KISALTMALAR

GA	: Genetik Algoritma
MHD	: Montaj Hattı Dengeleme
BMHD	: Basit Montaj Hattı Dengeleme
GMHD	: Genel Montaj Hattı Dengeleme
TMD	: Tek Modelli Deterministik
TMDO	: Tek Modelli Deterministik Olmayan
ÇMD	: Çok Modelli Deterministik
ÇMDO	: Çok Modelli Deterministik Olmayan
DB	: Dinamik Bölümleme
DBS	: Dinamik Bölümleme Sabiti

TABLO LİSTESİ

	<u>Sayfa No</u>
Tablo 3.1. Teknolojik öncelik diyagramına ilişkin öncelik matrisi.....	23
Tablo 4.1. Genetik algoritmanın işleyiş adımları.....	45
Tablo 6.1. 1994-2002 yılları arası MHD yöntemleri için literatür araştırması sonuçları.....	81
Tablo 7.1. Dış ünite bandı iş öğeleri, öncelikler, süreler ve varolan durumdaki atama.....	96
Tablo 7.2. Varolan durum istasyon ataması kayıp zamanlar tablosu.....	98
Tablo 7.3. GA farklı parametrelerine göre iş öğelerinin istasyonlara atama sonuçları.....	101
Tablo 7.4. Elde edilen çözümlerin performans kıyaslaması.....	103

ŞEKİL LİSTESİ

	<u>Sayfa No</u>
Şekil 2.1 : Genel bir üretim sistemi.....	4
Şekil 2.2 : Kitle üretim sistemleri.....	8
Şekil 2.3 : Montaj hattı tasarımları.....	11
Şekil 3.1 : n istasyonlu bir montaj hattında çevrim süresinin alt süreleri..	19
Şekil 3.2 : 11 iş öğeli bir teknolojik öncelik diyagramı.....	22
Şekil 3.3 : n istasyonlu bir montaj hattında tam ve tam kurulamamış denge durumları.....	25
Şekil 3.4 : 1. ürün (A) ile 2. ürün (B) için ve birleştirilmiş (C) teknolojik öncelik diyagramları.....	27
Şekil 3.5 : Montaj hattı dengeleme problemlerinin sınıflandırılması.....	34
Şekil 4.1 : Genetik algoritma temel işleyişinin şematik gösterimi.....	45
Şekil 4.2 : Gen takası.....	47
Şekil 4.3 : Kodlamada uygun olma ve yasaklı olma durumları.....	49
Şekil 4.4 : 24 bitlik kromozom yapılarının iki tabanlı kodlanması.....	49
Şekil 4.5 : Permütasyon kodlama örneği.....	50
Şekil 4.6 : Değer tabanlı kodlama örneği.....	50
Şekil 4.7 : Ağaç yapısı kodlama örneği.....	51
Şekil 4.8 : Rulet tekerleği seçimi.....	53
Şekil 4.9 : Tek noktalı çaprazlama.....	55
Şekil 4.10 : İki noktalı çaprazlama.....	55
Şekil 4.11 : Düzgün (uniform) çaprazlama.....	55
Şekil 4.12 : İkili değerlerin (binary) mutasyonu.....	56
Şekil 4.13 : Genetik algoritmanın klasik optimizasyon arama yöntemleri arasındaki yeri.....	61
Şekil 5.1 : Öncelik ilişkileri diyagramı.....	65
Şekil 5.2 : İş öğelerinin gruplanması ve kromozom gösterimi.....	67
Şekil 5.3 : Kromozom temsili (iş öğesi-istasyon odaklı).....	68
Şekil 5.4 : Kromozom kodlama (öncelik sıralı).....	68
Şekil 5.5 : İki noktalı çaprazlama.....	73
Şekil 5.6 : Scramble mutasyon.....	74
Şekil 5.7 : Dinamik bölümlleme mekanizması.....	78
Şekil 7.1 : Arcelik-LG fabrika ve montaj hattı görüntüleri.....	93
Şekil 7.2 : Öncelik diyagramı oluşturulmuş bir montaj hattı şeması.....	94
Şekil 7.3 : Optiline hat optimizasyonu diyalog kutusu.....	95
Şekil 7.4 : Optiline istasyonlara atanmış iş öğeleri gösterimi.....	95
Şekil 7.5 : Arçelik-LG dış ünite montaj hattı öncelik diagramı.....	99
Şekil 7.6 : Probleme ait GA parametreleri veri giriş penceresi şeması.....	100
Şekil 7.7 : Senaryo 2'ye ait optimize edilmiş optiline ataması.....	102
Şekil 7.8 : Senaryo 14'e ait 19 iş istasyonlu optiline dengeleme sonucu.....	105

SEMBOL LİSTESİ

C	: Çevrim süresi
CT	: Önceden kararlaştırılan tahminî çevrim süresi
D	: Denge kaybı
E	: Esneklik oranı
HE	: Hat etkinliği
HE_{enb}	: En büyük hat etkinliği
HE_{varolan}	: Varolan durumdaki hat etkinliği
m	: Varolan iş istasyonu sayısı
N	: Montaj hattındaki iş ögesi sayısı
n	: İş istasyonu sayısı
n_{enk}	: En az iş istasyonu sayısı
n*	: Olası en az iş istasyonu sayısı
P_c	: Crossover oranı (olasılığı)
P_m	: Mutasyon oranı (olasılığı)
Senb	: En büyük iş istasyonu süresi
S_k	: k. iş istasyonunun iş süreleri toplamı
S*	: Ortalama iş istasyonu süresi
T_j	: j. istasyonunun iş yükü
T_{ort}	: n adet iş istasyonu sürelerinin ortalaması
t_i	: i. no.'lu iş ögesinin işlem süresi
W	: Toplam iş ögeleri süresi

ÖZET

Günümüzde hâlen NP-zor kombinatoriyal optimizasyon sınıfına giren büyük ölçekli montaj hattı dengeleme problemlerinin etkin çözümü için sezgisel yöntemlere gereksinim duyulmaktadır. Stokastik bir arama tekniği olan GA'lar, gezgin satıcı problemi vb. birçok zor kombinatoriyal problemin etkin çözümünde kullanılan bir teknik olarak öne çıkmıştır. Bu açıdan GA yaklaşımının, montaj hattı dengeleme problemlerine uygulanması doğaldır. Genetik algoritmalar doğada oluşan, genetik mirasın aktarıldığı biyolojik evrim sürecini taklit ederek ve en iyinin yaşamını sürdürmesi ilkesi doğrultusunda çalışan stokastik arama yordamlarıdır. GA iteratif bir yordamdır. Her iterasyonda sınırlı bir küme olan bireylerden oluşan bir popülasyon oluşturularak çalışma sürdürülür. Her birey, probleme ait olası potansiyel bir çözümü temsil eder.

Her bireyin ne kadar uygun ve kaliteli olduğu, belirli bir değerlendirme fonksiyonuna göre ölçülerek saptanır. Daha sonra daha iyi bireyler seçilerek yeni bir popülasyon oluşturulur. Yeni popülasyonun bazı üyeleri, birtakım genetik operatörler kullanılarak değişime uğratılır. Bu işlemler, elde edilmesi beklenen ölçütler sağlanıncaya kadar sürdürülür. Bu tez çalışmasında, tek modellenli deterministik montaj hattı dengeleme problemlerinin genel tanımları yapıldıktan sonra geleneksel yöntemlerin etkin bir şekilde çözemediği problemlere uygulanan GA yaklaşımı araştırılmıştır. Genetik algoritmanın ana ilkeleri ve montaj hattı dengeleme problemlerine uygulanması süreci açıklanmıştır.

Sonraki aşamada, konuyla ilgili olarak 1994-2002 yılları arasındaki literatür çalışmaları incelenmiş olup, tez çalışması, iyi bilinen bir klima markası olan Arçelik-LG'nın Gebze İşletmesi'ndeki dış ünite montaj hattı üzerinde yapılan bir uygulama çalışmasıyla pekiştirilmiştir.

Elde edilen sonuçlar, bize GA'nın etkin bir yöntem olduğunu, özellikle büyük ölçekli montaj hattı dengeleme problemlerinin çözümünde gelecek vaat ettiğini göstermiştir.

SUMMARY

Since assembly line balancing is an NP-hard combinatorial optimization problem, some heuristic method is needed to solve the large-scale problem. A stochastic search technique known as the genetic algorithm has been proven effective in many combinatorial problems such as well known travelling salesman problem, so it seems natural to apply the approach to line balancing problems.

Genetic algorithms are stochastic procedures which imitate the biological evolutionary process of genetic inheritance and the survival of fittest. A genetic algorithm is an iterative procedure. During each iteration, a finite set, called a population of individuals are maintained. Each individual represents a potential solution to the problem. The fitness of each individual is measured according to an evaluation function. Then, a new population is formed by selecting the more fit individuals. Some members of the new population are altered by applying genetic operators. The above process is repeated until some termination criteria are met.

In this study, after giving general definitions of single model deterministic assembly line balancing problems, Genetic Algorithm approach to solve hard problems that can not be solved efficiently in a traditional way is researched. The main principles of GA and its application to line balancing area has been outlined.

Then, related to the subject, a literature research between 1994-2002 was introduced and the study was consolidated by a case study in a real world size problem on an assembly line of Arcelik-LG, a well-known air conditioner brand.

Consequently, the results proved the efficiency of GA which could be a promising method especially in large scale line balancing problems.

1. GİRİŞ

1.1. Giriş ve Çalışmanın Amaçları

Günümüzde çözülmesi zor ya da olanaksız olan çok sayıda problem vardır. Bu problemlerin çözülmesinde geliştirilmiş bir matematiksel fonksiyon olmadığı gibi, olası çözümlerin hesaplanması ve en iyi çözümün seçilmesi de çok zaman almaktadır. Bu problemlerin hemen hemen tümü çok geniş bir çözüm havzasının taranmasını gerektirmektedir. Bu çözüm havzasının geleneksel yöntemlerle taranması çok uzun sürmekte, olası tüm çözümlerin değil de salt bazı seçilmiş çözümlerin denenmesi yöntemiyle beklenen optimum sonucun elde edilmesini sağlayacak olan genetik algoritmayla ise kısa bir sürede kabul edilebilir bir sonuç alınabilmektedir.

Genetik algoritmalar, yapay zekânın gittikçe genişleyen bir kolu olan evrimsel hesaplama tekniğinin bir parçasını oluşturmaktadır. Adından da anlaşıldığı üzere, evrimsel hesaplama tekniğinin bir parçası olan genetik algoritma, Darwin'in evrim kuramından esinlenerek oluşturulmuştur. Herhangi bir problemin genetik algoritma ile çözümü, problemi sanal olarak evrimden geçirmek sûretiyle yapılmaktadır.

GA, bir dizi prosedürler kümesi olup yinelendiğinde ilgili problemin çözümünü olanaklı kılar. İstenen amacın gerçekleştirilmesi için GA'lar, istenen sonuçlar sağlanıncaya kadar eldeki çözümlerden başarılı popülasyonlar üretir. Üretilen her yeni kuşak ile bireysel çözümlerin kalitesinde iyileşme kazanılır. Bu yolla GA, istenen başarılı çıktıya, problemdeki olası tüm sonuçlar incelenmeden hızlı bir şekilde ulaşabilir.

Geniş bir uygulama alanı olan GA'lar mühendislik problemlerinde optimizasyon amaçlı olarak kullanılmaya başlanmıştır. Özellikle mekanizma tasarımında çok iyi sonuçlar verdiği bilinmektedir. Bunun yanı sıra, iş sıralama ve çizelgeleme, üretim

hattı yerleşimi ve montaj hattı dengelenmesi gibi NP-zor kombinatoriyal problemlerin çözümünde de etkili sonuçlar sağlamaktadır.

Montaj hatları ise insanoğlunun giderek artan gereksinimlerinin karşılanması için geliştirilmiş üretim sistemleridir. Bu sistemlerin kuruluşundaki amaç, istem miktarı çok yüksek olan ürünlerin; en kısa sürede, verimli bir şekilde, ucuz ve istenen kalitede üretilmesidir. Montaj hatlarının verimliliğinin artmasına yönelik birçok dengeleme yöntemi geliştirilmiştir.

İ.T.Ü. Fen Bilimleri Enstitüsü, Endüstri Mühendisliği Programı'nda yüksek lisans tezi olarak hazırlanmış bu çalışmada montaj hattı dengeleme problemlerinin etkin çözümüne yönelik genetik algoritma yaklaşımı incelenmiştir.

Çalışmanın ikinci bölümünde konuya genel olarak üretim ve montaj hatları kavramı ile giriş yapılarak, montaj hatlarının üretimdeki yeri ve çeşitleri anlatılmıştır.

Üçüncü bölümde montaj hattı dengeleme kavramı ve kullanılan terimler açıklanmıştır.

Dördüncü bölümde ise genetik algoritmanın tanımı, tarihçesi, işleyiş adımları ve uygulama alanları gibi yöntem hakkında genel bilgiler verilmiştir.

Beşinci bölümde de genetik algoritmanın, montaj hattı dengeleme problemleriyle olan ilişkisine değinilmiş olup, uygulanma süreci ayrıntılı olarak anlatılmıştır.

Altıncı bölümde 1994-2003 yılları arasında literatürde konu ile ilgili yapılan çalışmalar incelenmiş olup çalışmalar hakkında özet bilgiler verilerek sınıflandırılmıştır.

Yedinci bölümde, Arçelik-LG Gebze Klima İşletmesi dış ünite montaj bandındaki hat dengeleme problemi, GA yöntemiyle çözülmüştür. Elde edilen sonuçlar ile şu anda kullanılmakta olan çözüm kıyaslanmış olup, GA'nın etkin bir yöntem olduğu ortaya konmuştur.

Sekizinci ve son bölümde ise tez üzerine sonuç niteliğinde değerlendirmeler ve öneriler yer almaktadır.

2. ÜRETİM VE MONTAJ HATLARI

2.1 Üretim

Tanım olarak üretim, ekonomistler ve mühendisler tarafından farklı biçimlerde ortaya atılır. Mühendisler, bir fiziksel varlık üzerinde onun değerini artıracak bir değişiklik yapmayı veya hammadde ya da yarı ürünleri kullanılabilir bir ürüne dönüştürmeyi üretim sayarlar (Ellwoods, 1992). Ekonomistler ise üretimi yarar yaratmak şeklinde tanımlarlar. Sonuç olarak üretim, içeriği çok geniş olmasına karşın, temel amacı topluma değer yaratmak olan bir fonksiyon olup, insan gereksinimlerinin doğa tarafından tam olarak karşılanamaması sonucunda ortaya çıkan ve insanlar tarafından geliştirilen bir etkinliktir.

Diğer bir deyişle, üretim sistemleri salt fiziksel üretimi içeren sistemlerce sınırlandırılmamıştır. Hizmet üreten sistemlerde (eğitim, ulaşım, dağıtım vb.), girdilerin fiziksel veya kimyasal durumunda bir değişiklik sözkonusu olmamakla birlikte, yine topluma bir değer yaratılmaktadır; dolayısıyla bu sistemler de üretim sistemi olarak tanımlanırlar.

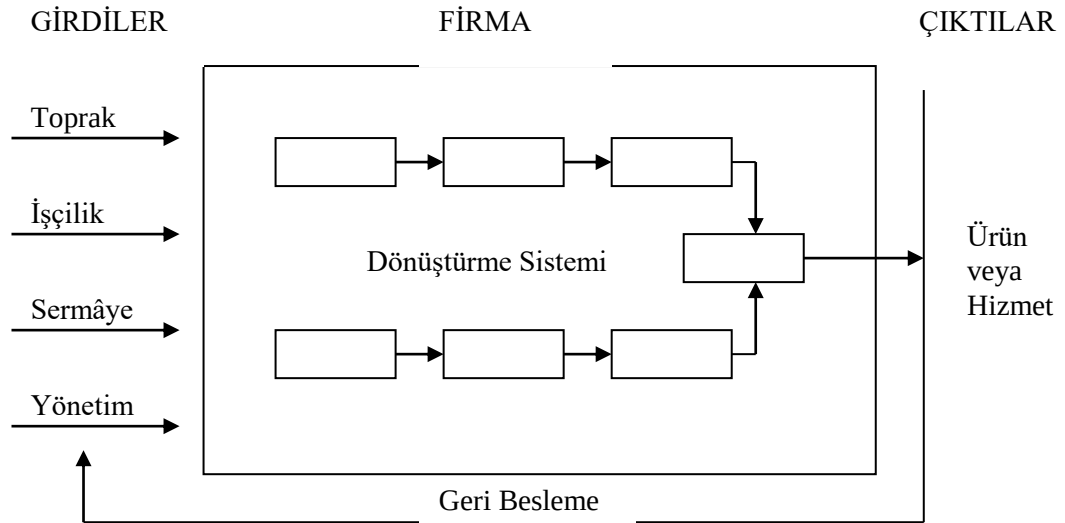
Üretim sonucu ortaya bir ürün veya bir hizmet çıkarılır. Üretimin amacına ulaşabilmesi için üretim etmenlerinin ve kaynaklarının, istenen özellikte birleştirilmesi ve kullanılması gerekir.

Ancak insan gereksinimlerinin sonsuz ve kaynakların sınırlı olması nedeniyle, gelişen teknoloji, üretim sistemlerinin karmaşıklaşması ve üretim kaynaklarının verimli kullanılması gereksiniminin bir sonucu olarak yönetim ögesi de üretim kaynaklarına katılmıştır.

Tüm üretim sistemlerinin ortak amaçlarına ulaşmak için uymak zorunda oldukları ortak nokta ise kaynakların verimli kullanılmasıdır. Üretim kaynakları genelde dört ana başlık altında toplanır:

- a) Toprak (Hammadde kaynakları)
- b) İşçilik (İnsangücü kaynakları)
- c) Sermâye
- d) Yönetim

Gerek hizmet gerekse ürün üreten üretim sistemlerinde, üretim süreci Şekil 2.1.'deki gibidir.



Şekil 2.1. Genel bir üretim sistemi (Acar, 2000)

Temelde tüm üretim sistemleri, kaynakların, ürün veya hizmete dönüştürülmesi için gerekli tüm dönüşüm süreçlerini içerir. Bu sistemlerin tasarlanması, plânlanması, yönetimi ve kontrolü aşağıdaki üretim süreçleri içinde gerçekleşir.

Bir üretim işletmesine girdi olarak katılan başlıca üretim kaynakları; hammadde, işçilik, sermâye ve yönetimdir. Üretim ile bu girdiler ürün veya hizmet hâline dönüştürülür. Yönetim etmeni ise bu faaliyetlerin düzenlenmesi ve yürütülmesini sağlar. Amacı, bir işletmenin elinde bulunan malzeme, makina, insangücü

kaynaklarının, istenen kalite ve sürede, en düşük mâliyetle biraraya getirilmesi, yâni miktar, kalite, zaman, mâliyet parametrelerinin optimize edilmesidir.

2.2. Üretim Tipleri

2.2.1. Genel Bilgi

Varolan üretim sistemlerini günümüzde kesin hatlarla birbirinden ayırmak ve sınıflandırmak tam anlamıyla olanaklı değildir. En önemli olarak; üretim tipleri arasında yapılacak bir ayırımdan sonra, oluşturulan bu sınıflar esas alınarak, bunlara özgü genel kurallar ve yöntemleri geliştirme olanağının pek olmaması göze çarpar.

Üretim sistemlerini:

1. Üretim yöntemlerine göre,
2. Ürün tipine göre,
3. Üretim miktarı ve akışına göre,
4. Üretim iş rotasına göre

çeşitli bakış açılarında sınıflandırmak olanaklıdır.

Üretimin miktarına ve akışına göre sınıflandırılması gözönüne alınarak üç ana üretim tipi ile karşılaşmak olanaklıdır: Sürekli üretim, kesikli üretim, proje tipi üretim.

2.2.2. Sürekli (Serî) Üretim

Sürekli (serî) üretim tipinde, üretim akışı sürekli olarak seyreder ve üretilen ürün çeşidi oldukça azdır. Tüm sistemlerde üretilen ürüne yönelik istemin en az üretim hızı kadar olması gereklidir. Yâni bu sistemle üretilen ürünlerin tümü pazar buluyorsa böyle bir sistem kurmanın anlamı vardır. Çünkü bu türde bir üretim, pahalı ve özel amaçlı tezgâhlarla gerçekleştirilir. Bu yüzden ilk yatırım mâliyeti yüksektir ve üretim yalnızca ürünün özellikleri üzerine kurulduğundan, başka ürünlere geçişi sağlayacak üretim esnekliği yoktur. Bu nedenle istemde oluşabilecek düşmelerin mâliyeti çok fazladır.

Bu türde üretim sistemlerinde ürün çeşidinin az olması ve üretim hacminin büyük olması nedeniyle uzmanlaşma yüksek düzeydedir ve üretilen ürünler standartlaştırılmıştır. Ürün birim mâliyeti, sipâriş tipine oranla düşüktür ve plânlama etkinlikleri daha kolay yürütülür.

Sürekli üretilen hammadde, yarı ürün ve parça gibi üretim girdileri, üretim sisteminin başlangıç noktasından girer, sistemdeki işlem birimlerinden geçer ve son ürün hâline gelerek sistemden çıkarlar.

Sistemdeki işlem birimlerinin işletme içindeki yerleşimi ürünün göreceği süreç akışına göre yapılır. Ürünün göreceği işlemlerin sırasına göre dizilen işlem birimleri bir üretim hattı oluştururlar. Bu nedenle ve üretimin özelliklerine göre transfer ve montaj hatları bu grubun içinde yer alırlar.

2.2.3. Kesikli (Parti) Üretim

Bu tür üretim sistemleri genellikle sipâriş esasına göre çalışırlar ve üretilen ürün çeşidi, seri üretimdeki ürün çeşidine göre daha fazladır. Bu nedenle genel amaçlı tezgâh ve iş makinalarının kullanımı tercih edilir. Ancak kullanılan makinaların kapasitelerinin birbirinden farklı olması üretimde boşluklara neden olur ve makinalar arasındaki bekleme ve hazırlık süreleri, üretim hızını önemli ölçüde düşürür.

Bu tür üretimde belirli partiler belirli aralıklarla üretildiği için üretim plânlama ve kontrol çalışmaları sipâriş tipi üretime göre daha kolaydır. Ancak bu sistemde ortaya çıkan iki ana sorun, üretilecek ürün partilerinin büyüklükleri ve parti hacminin yâni o partide üretilecek ürün sayısının saptanması ve bu partilerin çizelgelenmesi, işlerin sınırlandırılmasıdır.

Gerçekte ürün çeşitliliği azaldıkça, belirli ürünlere ilişkin üretim miktarları ve belirli iş makinalarındaki işlem yinelenmesi arttıkça, yâni kesikli üretim sistemi sürekli üretim sistemine dönüştükçe üretim plânlama ve kontrol etkinliklere daha kolaylaşır. Parti üretiminde sık sık yinelemelerle büyük partilerde ürün üretimine geçildiğinde, seri üretime yaklaşılr. Diğer yanda partiler küçülür ve üretime giriş aralıkları açılırsa sipâriş tipi üretim söz konusu olur. Özetle kesikli üretim sistemleri arasında sınıflandırma açısından kesin bir ayırım çizgisi yoktur.

Kesikli üretim sistemlerinde üretimin hızlandırılması amacıyla 1970'li yıllarda öncelikle S.S.C.B.'de, daha sonra ise İngiltere ve İtalya'da grup teknolojisi yaklaşımı geliştirilmiştir. Bu yaklaşıma göre üretilecek ürünlere partilere göre değil, ürünlere ilişkin parça ailesine göre oluşturulur. Aralarında geometrik benzerlik bulunan ve aynı işlevi gören parçalar, bir parça ailesini oluşturur. Değişik ürünlerde mamullerde yer alan aynı parça ailesine ilişkin operasyonlar birlikte yapılırken hazırlık süreleri en aza indirilir, üretim hızı ve üretim sürekliliği yükseltilir. Ortaya çıkarıldığı dönemlerde pek ilgi görmeyen grup teknolojisi yaklaşımı, bilgisayar destekli imalat sistemlerinin geliştirilmesiyle uygulama alanı genişlemiş ve kesikli üretim sistemleri için oldukça önemli bir yaklaşım hâline gelmiştir.

2.2.4. Proje (Sipariş) Tipi Üretim

Tek veya çok az sayıda üretilecek, genellikle büyük boyutlu ürünlerin üretimde sözkonusu bir üretim tipidir. Sipariş üzerine üretime geçiş ve özel ürünlerin üretimi gündemdedir. Doğal olarak işlemlerde yineleme en az düzeydedir. Kesikli üretim tiplerinde olduğu gibi genel amaçlı (üniversal) iş makinalarına gereksinim vardır. Ancak bunun yanında, değişik tezgâhlarda çalışabilen esnek işçi kullanımı sözkonusudur.

Gerçekleştirilecek projelerin plânlanması, optimum sürede ve en düşük mâliyetle gerçekleştirilmesi elbette ana sorundur. Ayrıca eldeki işgücü, donanım, finansman, malzeme vb. kaynakların dengeli olarak ve optimum şekilde kullanılması önemli bir sorundur. Bu nedenle projenin plânlanması ve kontrolü aşamasında CPM ve PERT teknikleri en sık kullanılan yöntemlerdir.

2.3. Montaj Hatlarının Üretimdeki Yeri

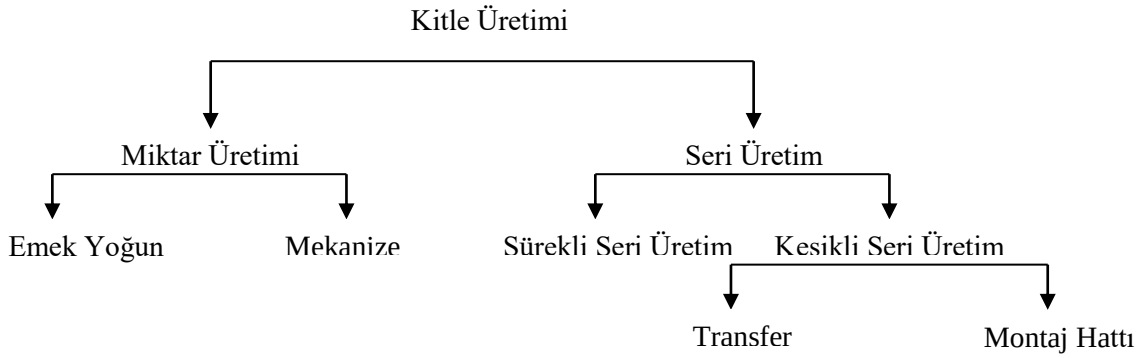
Uygulamada seri ve kesikli üretim sistemleri arasında yer alan parça üretimi, parti (kâfile) üretimi ve kitle üretimi ile bunların karışımı olan üretim sistemleriyle karşılaşır. 18. yüzyıl sonlarına doğru mekanik üretim yöntemlerinin gelişmesine de neden olan kitle üretim sistemlerinin doğuşu, daha sonra kitle üretiminin bir teknoloji gelişimi hâline gelmesiyle sonuçlanmıştır.

Kitle üretimi terimi iki değişik üretim teknolojisini içerir. Bunlar:

a) Miktar üretimi

b) Akış tipi üretim

olarak karşımıza çıkarlar (Şekil 2.2.). Kitle üretiminde ürün yapısının karmaşıklığı arttıkça, akış tipi üretim sistemlerine geçilir. Fazla karmaşık olmayan, yapısal olarak basit ürünlerin büyük miktarlardaki üretimi için ise miktar üretimi uygundur. Karmaşık ürünlerin ana özelliği, ürün akışı olan değişik bir kitle üretimi teknolojisiyle üretimleri gerçekleştirilebilmekte olmasıdır (Acar ve diğ., 1986).



Şekil 2.2. Kitle üretim sistemleri (Wild, 1973)

Kesikli seri üretim sistemleri, diğer bir tanımlama ile akış hatları iki gruba ayrılır:

Transfer Hatları: Transfer âletleri ile birbirine bağlanmış otomatik imalat makina serilerinin oluşturduğu üretim hatları transfer hatları olarak adlandırılır. Büyük ve karmaşık makinalardan oluşan sistemlerdir. Bu makinaların oluşturduğu transfer hatlarında malzeme, bu hat boyunca otomatik olarak transfer edilir ve parçalar otomatik olarak işlenir. Transfer hatları düz ya da devirli tiplerde olabilir. Bu hatların en temel özellikleri; malzemenin bir hat boyunca otomatik olarak transferi, hat boyunca malzemenin çoğunlukla işgücüne dayalı bir şekilde işlenmesi ve transferi, parçaların otomatik olarak işlenmesi şeklinde özetlenebilir.

2) **Montaj Hatları:** Bu tip hatlarda malzeme çoğunlukla hat boyunca işgücüne dayalı olarak işlenir ve transfer edilir. Parça üzerindeki işlemler de yine bir hat boyunca sıralanır. Hat boyunca iş akışı düzenli hâle geldikçe, normal olarak birimlerin hat üzerinde gideceği süre azalır ve işgücü gereksinimi en aza indirilir. Montaj hatları büyük ölçüde ikâme edilebilen parçaların kullanımına

bağlıdır. Yani üretimde kullanılan yarı ürün ve parçaların yeterli doğruluk standartlarında olmaları gerekir.

Bir diğer yaklaşım ile; seri durumda bulunan iş istasyonlarından oluşan akış hatlarında, bu istasyonlardan daha fazla makina bulunduran hatta transfer hattı, daha çok çeşitli âletlerde donatılmış işçi bulunduran hatta ise montaj hattı adı verilir.

2.4. Montaj Hattı ve Tipleri

2.4.1. Genel Bilgi

Malzemelerin akış hattı boyunca işgücünden yararlanılarak transfer edildiği ve parça üzerindeki işlemlerin aralarındaki öncelik ilişkileri gibi kısıtlar gözönüne alınarak birleştirilmesiyle oluşturdukları istasyonların da, yine bir hat boyunca sıralanmalarıyla oluşan sisteme montaj hattı adı verilir. Toplam işgücü iş istasyonlarına öyle bağlanmıştır ki, son işleme gelinceye kadar yavaş yavaş işlemler tamamlanır ve son istasyonda hatta giren yarı ürün hâlini alır.

Kesikli seri üretim akış hatlarının bir alt sistemi olan montaj hatlarının endüstride yeri oldukça önemlidir. İşlemlerin küçük parçalara bölünebilmeleri ve istasyonlar arasında dağıtılabilme esnekliği, montaj hatlarının olası çözümler sağlaması sonucunu doğurur.

İlk kez 20. yy. başlarında Ford motor fabrikalarında kullanımına başlanan montaj hatları, tek tek büyük birimlerin, örneğin motor, motorlu taşıt araçları, beyaz eşya, ev âletleri gibi ürünlerin büyük miktarlarda üretimi için tasarlanırlar. Bir montaj hattı kurulurken genellikle ekonomik görüşler ön planda tutulmalıdır. Böyle bir aşamada incelenmesi gereken birçok etmen vardır. Bunların içinde kullanım düzeyleri konusu ayrı bir önem taşır. Örneğin mekanik tesislerin ağırlıkta olduğu bir sistem söz konusu ise akış hattı, sistem veya süreç tipi yerleşim düzeni arasında bir karar vermek gerekir. Bu durumda ekipman kullanım oranlarının analize dayandırmak akılcı bir yaklaşımdır (Uzmen, 1990).

Üretimde kullanılan ekipman kullanım oranı belirlenirken, çıktının standart saat değeri eldeki zamanla karşılaştırılır. Ancak böyle bir çözümle problemlerin yalnızca

bir bölümü ele alınmış olur. İncelenmesi gereken diğer etmenler ise, istemin sürekliliği, ekipman güvenilirliği, yatırım sermayesinin kullanımıdır.

Montaj hatlarında kullanılacak işgücünün mâliyeti, özel amaçlı tezgâhlarda çalışacak işgücünün mâliyetinden daha düşük olacaktır. Bu işgücünün eğitim gereksinimi de fazla olmayacağından, gerek işe alma ve gerek işte yer değiştirme konusunda fazlaca sorunla karşılaşılmaz (Acar ve diğ., 1986).

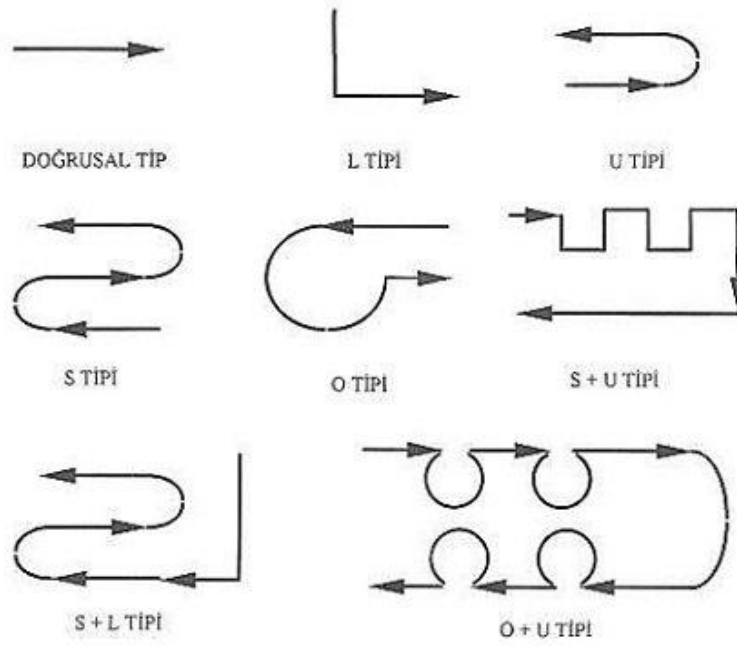
Montaj hatları, iş istasyonundaki işlemlerin ve/veya bu istasyonlar arasındaki bağlantıların otomatik ve yarı otomatik olmasına bağlı olarak sınıflandırılırlar.

Montaj hatları, üretilen model çeşitliliği açısından da sınıflandırılırlar. Buna göre;

- 1) **Tek Modelli Hatlar:** Tek tip ürün veya modelin üretiminde kullanılır. Bu tip hatların tasarımı oldukça basittir. Ancak başka ürünlerin üretimine uygun bir esnekliğe sahip değildir.
- 2) **Çok Modelli Hatlar:** Bu tür hatlarda, farklı ürünler veya bir ürünün iki veya daha fazla benzer tipi ayrı yığınlar hâlinde üretilir. Her model, bu hat üzerinde ayrı bir yığın oluşturur. Çok modellenli montaj hattı eğer yığınlar büyük ise tek modellenli montaj hatlarına, küçük ise karışık modellenli montaj hattına benzerlik gösterir.
- 3) **Karışık Modelli Hatlar:** İki veya daha fazla benzer ürün veya bir ürünün değişik modellerinin aynı anda ve karışık olarak üretildiği montaj hatlarıdır. Bu tür üretimde, kuramsal olarak büyük miktarlarda bitmiş ürün stoklarına gereksinim olmayıp, çok modellenli hatların tersine, tüketicinin istekleri sürekli bir şekilde yapılan üretimle karşılanır. Karışık modellenli hatlarda gözlenen temel olumsuzluk, modellerin özelliğinden kaynaklanan farklı iş parçalarının, eşit olmayan iş akışları, boş istasyon süreleri ve yarı bitmiş ürün stoklarına neden olmasıdır. Bu tip montaj hatlarında oldukça karmaşık tasarım ve işlem sorunları söz konusudur.

Donanımın ve iş istasyonlarının yerleşim biçimi, hat tipindeki üretimleri etkileyen önemli bir etmendir. Hattın bulunduğu yer ve üretilecek ürünün özellikleri, hattın alacağı şekli belirler. Montaj hatları, yatay ve dikey olarak yerleştirilebildikleri gibi

düz, dairesel, oval, yılankavi, zig-zag, “U” şeklinde, rassal veya değişik açılı olarak da tasarlanabilirler (Maynard, 1969).



Şekil 2.3. Montaj Hattı Tasarımları (Aksoy, 1997)

Genel olarak üretim hatlarının yerleşiminde doğrusal hatlar tercih edilir. Bunun nedeni, doğrusal hatların basit ve sistematik olması, kolayca yerleşim yapılabilmesi, konveyör sistemlerinin uygulanabilirliğinin artması ve mâliyetlerinin düşmesi, köşelerde meydana gelebilecek transfer zorluklarının ortadan kalkmasıdır. Ancak aşağıda verilen özel durumlarda değişik şekillerde montaj hatlarının kullanımı sözkonusudur:

1. Hattın boyu çok uzunsa, varolan alana sığmıyorsa veya düz hat olarak yerleştirme yapıldığında boşta kalacak alanın boyutları büyükse, U şeklinde veya dairesel hat kullanılır.
2. Bir operatör, değişik işlem sıralarındaki ve aynı hat üzerindeki makinalardan birden fazlası ile çalışıyorsa, yılankavi veya U şeklinde hat kullanılır.
3. Mâliyeti yüksek bir makinaya birbirinden ayrık iki operasyon yaptırılması gerekiyorsa, U şeklindeki hatlar yeğlenir.

4. Varolan alan uygun değilse ve çok bitişik düzenlemeler gerekiyorsa düz hat dışında kalan uygun bir hat yeğlenir.

5. Elektrik, hava gibi tesisat bağlantıları birden fazla istasyona aynı kaynaktan yapılıyorsa, U şeklinde veya yılankavi hatlar kullanılır.

Şekil imajının yanısıra, üretilecek ürünün özelliklerine göre seri, bileşik, paralel ve besleyici montaj hattı tipleri olmak üzere dört tip fonksiyonel montaj hattı sınıflandırması yapılabilir.

Bunun yanısıra, basit montaj hatlarında hattın işleyişine göre bir sınıflandırma yapmaya gerek yoktur.

Ancak karmaşık üretim hatları, hattın işleyiş özelliklerine göre iki ana sınıfta gruplandırılır: Mekanik olmayan bantlar ve hareketli bant hatları.

2.4.2. Mekanik Olmayan Hatlar

Bu tür hatlarda, hat boyunca malzeme transferi için gerekli bant veya konveyörler kullanılmaz. Bu gruptaki üretim hatlarında yapılan işlemler için genellikle elle işletim sözkonusudur. Uygulamada bu yöntemle işletilen montaj hatlarında, istasyonlar arası güvenlik stokları oluşturulması, istasyonların boş kalma ve bloke edilmesi olasılığı azaltılır ve boş süreler en aza indirilir.

2.4.3. Hareketli Bant Hatları

Hat boyunca malzeme ve ürün transferlerinin, hareketli bant veya konveyörlerle gerçekleştirildiği bu hatlar, kendi aralarında iki gruba ayrılırlar:

a) Birimler banttan alınabilir: Bu tip hatlarda, tolerans zaman çok büyük önem taşır. İş parçasının hattan alınması için gerekli zaman, tolerans zamanı olarak adlandırılıp, iş parçasının geçtiği istasyonun ve hattın uzunluğunu belirler.

b) Birimler bant üzerinde sabittir: Bu tip hatlarda, iş akışı düzenli ve çevrim süresi değişkendir. Hattın özelliğinden dolayı işçiler, işlerini görebilmek için yukarı ya da aşağı uzanmak veya parçalar ile birlikte hat üzerinde ilerlemek zorunda kalırlar. Bu hatlarda birimlerin istasyonlar arasında akışını sağlamak amacıyla, görsel, işitsel sinyallerden yararlanılır.

3. MONTAJ HATLARININ DENGELENMESİ

3.1. Montaj Hatları ve Hat Dengeleme Kavramı

Ürünler genelde birkaç parçadan oluşur. Bu parçaların da birçok alt parçanın birleştirilmesinden oluştuğu durumlar vardır. Örneğin, bir radyonun veya televizyon setinin elektronik tüpü, çeşitli küçük bileşenler monte edilerek yapılmaktadır (MESS, 1987, s.149).

Endüstrileşme sürecinde, toplam işin ögelerine (parçalarına, modüllerine) ayrılarak, bu parçaların ayrı ayrı işçiler tarafından yapılmasıyla daha hızlı, kitlesel (serî) ve daha ucuz üretim yapılabilceği görüşü ortaya çıkmıştır. Bunun sonucu olarak üretim, üzerinde değişik iş istasyonlarının bulunduğu belirli bir hat üzerinden malzemelerin geçirilmesi yoluyla yapılır. Malzemelerin, akış hattı boyunca işgücü veya donanımdan yararlanılarak transfer edildiği ve parça üzerindeki işlemlerin; aralarındaki öncelik ilişkileri ve çevrim süresi gibi kısıtlar gözönüne alınarak birleştirilmesiyle oluşturulan istasyonların, yine bir hat boyunca sıralanmalarıyla oluşan sisteme, **montaj hattı** denir. Hat üzerindeki iş istasyonlarında bulunan işçiler, ürün durumuna getirilecek yarı ürün önlerinden geçerken, kendilerine ait iş öğeleriyle ilgili bir veya birkaç işlemi yaparlar. Bu işlem sonucunda, hatta giren parça ve yarı ürünler, gereken tüm işler yapılmış şekilde, hattın sonundan ürün olarak çıkarlar.

Bir veya birkaç ürün için yapılacak montaj hattı üretimi tasarlandığında; üretim hattındaki iş istasyonlarına ilişkin işlem sürelerinin dengelenmesi sorunu ortaya çıkar. Bundaki amaç; kurulan montaj hattının verimli olarak çalışabilmesi için; üretim süresi içinde her bir montajcıya, çok az boş süre bırakılacak veya hiç boş süre bırakılmayacak şekilde işlemlerin istasyonlara dağıtılması, yâni varolan kısıtlar altında -işlem sayısının çok ve üretim hızının yüksek olmasından dolayı- iş istasyonları arasındaki işlem süresi farkları toplamının enküçüklenmesidir. Sorunun

bu noktada, sürekli üretim yapan sistemlerin yerleşim düzeninin kurulmasında, hat dengeleme problemi ortaya çıkar.

Ürün oluşumu sırasında yapılması gereken işlerin, montaj istasyonlarına, kayıp süreleri en aza indirecek şekilde atanması olayına, bir başka tanımla iş öğelerinin iş duraklarına özgülmesine, **montaj hattı dengeleme (assembly line balancing)** veya kısaca **hat dengeleme (line balancing)** denir.

Montaj hattı dengeleme konusu; üretim hızının artırılması, sağlıklı bir planlama yapılması ve işletmenin ekonomik sorunlarına çözüm getirmeye yönelik olmasından dolayı, endüstri dünyasında büyük önem taşır.

Görüldüğü gibi hat dengeleme problemleri, üretim hatlarının tasarımında ve işletilmesinde her an ortaya çıkabilen bir konudur. Gerçekte hat dengeleme problemleri, çeşitli kapasitelerde ve üretim hızlarında çalışıldığında, hatta oluşabilecek ve iş istasyonlarının verimliliğini etkileyecek başlıca etkenlerden olan âtil kaynak kullanımını en alt düzeyde tutmak için en uygun çözümün belirlenmesinde ortaya çıkmıştır.

Yukarıda genel olarak açıklanan hat dengeleme konusu, gerçekte, üretimi yapan işletmeciler için çok önemli bir konudur. Kaliteli ve yüksek hızlı bir üretim düzeyi tutturmak, gerçekte hat dengeleme çalışmalarının ayrıntılı ve iyi bir şekilde yapılması ile sağlanır.

Günümüzde ise üretim miktarları, rekâbet arttığından ve piyasa koşulları zorlaşmaya başladığından, artık üretim hatlarını tasarlamak ve bu hatlara, oluşan çeşitli dalgalanmalar karşısında yeterli esnekliği verebilmek için girilen çalışmalar, bilgisayar desteği ile geliştirilen simülasyon programları ile oldukça kolay olmaktadır.

Üretimde, tezgâhlar ve çağdaş montaj âletleri kullanıldığı takdirde, görece daha kısa bir sürede daha çok üretim yapılabilir. Ürün şeması veya montaj hatları kullanılarak bir kez üretim yapılmasıyla, büyük hacimdeki mallar verimli bir şekilde üretilir.

3.2. Montaj Hatlarının Dengelenmesinin Amaçları

Montaj hatları, kitle üretiminin önemli bir alt sistemidir. Bu tip sistemler ayrıntıda farklı olmakla birlikte, temelde birbiri ardına dizilmiş iş istasyonlarından oluşur. Hammadde ve yarı ürün parçalar, hat içine, hattın başlangıcından veya ara istasyonlardan girerler. Giren parçalar, bir iş istasyonundan diğerine geçerek, en son istasyondan hattı ürün olarak terkederler.

Transfer donanımı ile birbirine bağlanmış otomatik üretim makina serîlerinin oluşturduğu üretim hatları olan ve transfer makinaları olarak da bilinen transfer hatları, montaj hatlarından farklı olarak insangücünü çok az kullanır. Transfer hatlarının belirgin özellikleri; malzemenin bir hat boyunca otomatik olarak transferi ve parçaların otomatik olarak işlenmesidir. Oysa montaj hatlarında; hat boyunca malzeme çoğunlukla işgücüne dayalı bir şekilde işlenir ve transfer edilir.

Bir montaj hattının kurulmasında ulaşılması gereken amaçlar şunlar olabilir:

1. Düzenli bir malzeme akışını sağlamak.
2. İnsangücü ve tezgâh kapasitelerini en üst düzeyde kullanmak.
3. İşlemleri en kısa sürede tamamlamak.
4. Boş (âtıl) süreleri enazlamak.
5. Montaj hattı üzerindeki iş istasyonu sayısını enazlamak.
6. Âtıl süreleri, iş istasyonları arasında düzgün şekilde dağıtmak.
7. Üretim mâliyetini enküçükmek.

Montaj hattı dengelemenin amaçları birbirleriyle çeliştiklerinden, hepsini birden en üst düzeye ulaştırmak olanaklı olmayabilir. Dengelemede ana amaç, bu çelişkilerin gözönüne alınarak en uygun çözüme ulaşılmasıdır. Bu yapılırken, montaj mâliyeti de enküçüklenmelidir. Dengeleme işleminde düşünülen etmenler içinde, mâliyet etkisi olan iki değişken, işgücü büyüklüğü (insangücü mâliyeti) ve hattın uzunluğudur (kullanılan alan, sermâye mâliyeti).

İşgücü yükünün dengelenmesinde başvurulabilecek bâzı çâreler şunlardır:

1. Otomatik işlem süresi uzun olan iki veya daha fazla tezgâhta tek bir işçi çalıştırılabilir.
2. İki kısa işlem süresi, diğerleri kadar veya daha az ise, bunlar, yapılmak üzere bir işçiye verilebilir.
3. İşçinin yükü artırılabilir.
4. İşçiler, çalışma hızlarına göre dizilebilir.

3.3. Montaj Hatlarının Dengelenmesini Etkileyen Temel Etmenler ve Kısıtlar

Montaj hattı dengelemeyi etkileyen temel etmenler, aşağıdaki şekilde sınıflandırılabilir:

Mühendislik spesifikasyonları, işlemler arası öncelikler ve gerekli kaynak (girdi) gereksinimleri.

İşin yapılmasında izlenen yöntem (bâzı montaj işlerinde verilen belirli bir teknolojik sıra, işin kolay ve daha hızlı yapılmasını sağlayabilir).

Kullanılan aygıtlar ve tezgâhlar (bâzı âletlerin, montaj hattının birden çok yerinde kullanılmasına gerek duyulabilir; böyle durumlarda aynı âlet grubunu birden fazla işlemcinin kullanabileceği bir şekilde, bâzı istasyonları ardarda yerleştirmek gerekebilir).

Montaj hatlarında iki temel kısıt vardır. Bunlar çevrim süresi ve öncelik ilişkileridir.

Çevrim Süresi: Çevrim süresini; verilmiş net üretim hedefi, brüt çalışma süresi ve tolerans yüzdesi (yâni kontrol edilemeyen nedenlerle yitirilen süreler ve önceden tasarlanmış duruş süreleri toplamının, brüt çalışma süresinin yüzdesi olarak ifâdesi) birlikte belirlerler. Çevrim süresi; bir adet ürünün, üretimi sırasında herhangi bir istasyonda işlem görebileceği en büyük süre değeridir. Bir istasyona atanan iş öğelerinin süreleri toplamı, çevrim süresini aşamaz.

Öncelik İlişkileri: Tüm montajın içerdği iş öğelerinin kendi aralarında sözkonusudur. Yâni bir iş öğesinin yapılmaya başlanabilmesi için, diğer iş öğesi veya öğelerinin kesinlikle bitirilmiş olması gerekebilir. İstasyonlara yapılan iş öğesi atamalarının, bu öncelik ilişkilerine aykırı olmamaları zorunludur.

Temel kısıtların dışında beş tane de yan kısıt vardır. Bunlar aşağıda açıklanmıştır.

Konum Kısıtı: Konumsal kısıtlamalar, montajı yapılan nesnenin konumu ile, işlemcilerin banttaki konumu arasındaki ilişkiyi ifade etmekte kullanılan bir kavramdır.

Sabit Donanım Kısıtı: Tezgâhlar, test araçları gibi sabit donanımlar, montaj hatlarının bütünleşik parçalarıdır ve değiştirilemez istasyonları oluştururlar. Sabit donanım kısıtı, iş öğelerinin değiştirilebilirliğini azaltır.

İstasyon Yüğü: Montaj hattında bazı istasyonların yüklerinin, çevrim süresinin %100'ünden az olması yeğlenebilir. Bu özellikle, ilk istasyonda veya istasyonlarda, hattın başında olabilecek aksamaların, tüm hatta etkisini azaltabilmek için yapılabilir.

Aynı İstasyona Atanması İstenen İş Öğeleri: Bu özelliğe sahip işlerin, aynı veya birbirlerini izleyen istasyonlara atanması gereklidir. Böyle durumlarda bir iş öğesi alt grubu, tek bir iş öğesi gibi düşünülebilir. Örneğin, özel aygıt kullanımını gerektiren iki iş öğesinin aynı işçi tarafından yapılması, ikinci bir aygıt gereksinimini ortadan kaldıracağı için, istenilen bir durumdur.

Aynı İstasyona Atanmaması İstenen İş Öğeleri: Bu özelliğe sahip bir iş öğesi, diğer bazı iş öğeleriyle, aynı istasyona atanamaz. Örneğin, aşırı fiziksel güç uygulamasını gerektiren iki iş öğesinin, iş yükü açısından ayrı istasyonlara atanmaları istenebilir. Benzer şekilde, birden fazla iş öğesi, teknolojik olarak istenmediği için aynı işlemciye atanmazlar veya en azından bir aracı (tampon) istasyon ile birbirlerinden ayrılırlar. Titreşimli bir çalışma ile, hassas ölçme gerektiren bir çalışmanın ayrılması buna bir örnek olarak verilebilir.

3.4. Montaj Hatlarının Dengelemesinde Kullanılan Temel Kavramlar

3.4.1 İş Ögesi

İşler, temel hareketlerden veya iş parçacıklarından oluşur. İş ögeleri; toplam işin, uygun ve pratik en küçük alt parçalarıdır ve bu iş parçacıklarının bir veya birkaçı tarafından oluşturulurlar. İş ögesi (work element); üretim süreci içinde, toplam iş içeriğinin, mantıksal olarak bölünmüş bir parçasıdır.

Örneğin bir parçaya beş tane delik açmak için beş farklı iş ögesi tanımı yapılabilir. Ama montaj hattının dengelenmesi sözkonusu olunca, mantıksal iş ögesi, beş deliği birden açmayı içeren iş grubu olarak tanımlanmalıdır. Ayrıca bu işlem otomatik bir tezgâhta tek bir seferde yapılıyor ise o zaman beş tane delik açma işi tek bir iş ögesidir.

3.4.2 İş İstasyonu

Montaj hattı üzerinde verilen bir işin, işçi/işçiler tarafından yapıldığı alandır. Her istasyonda, bir işçinin, bir işlem için gerekli araçlarla çalıştığı varsayılır. Genellikle iş istasyonu (workstation), bir montajcı tarafından doldurulan yer olarak düşünülür (WEE ve MAGAZINE, 1982). Bir montaj hattı için; en az istasyon sayısının 1 olduğu ve istasyon sayısı dengeleme çalışması sırasında saptanan gerekli en az istasyon sayısının altına düşmemek gerektiği kısıtları vardır.

3.4.3 Toplam İş Süresi

Montaj hattı üzerinde üretilecek bir ürünün montajı için gerekli olan süre veya işi oluşturan tüm iş ögelerinin standart süreleri toplamıdır.

Toplam iş süresi;

N : Montaj hattındaki iş ögesi sayısı

t_i : i no.lu iş ögesinin işlem süresi

olmak üzere şu şekilde hesaplanır:

$$\sum_{i=1}^N t_i \quad (3.1)$$

3.4.4 İş İstasyonu Süresi

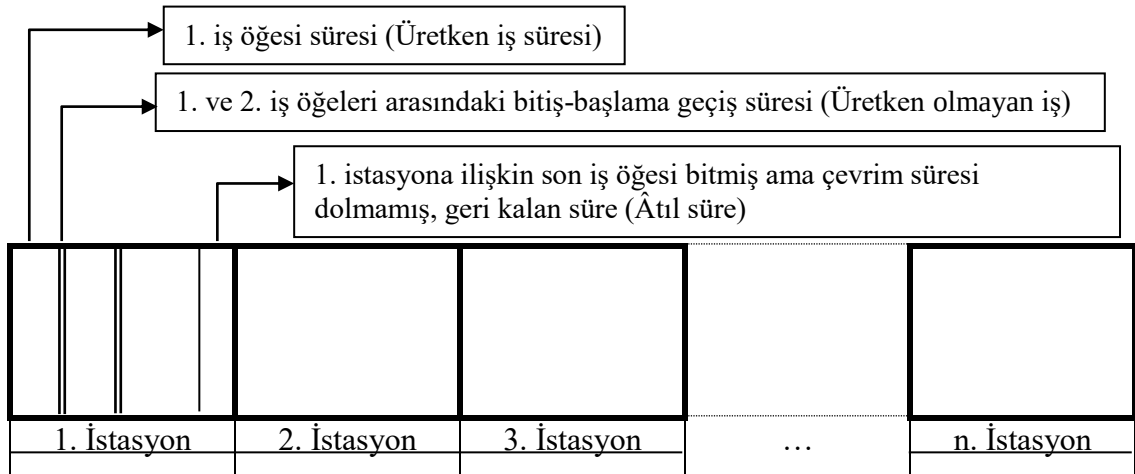
Bir iş istasyonunda yapılması gerekli olan iş ögelerinin standart süreleri toplamıdır. Yani, istasyona gelen bir parça üzerinde o istasyonda yapılması gereken ilk iş ögesinin başlangıç anı ile son iş ögesinin bitiş anı arasındaki süre farkıdır (DE JONG, 1957). Bir iş istasyonu süresi (workstation time), o hattaki iş ögesi sürelerinin en büyüğünden küçük, çevrim süresinden büyük olamaz (paralel istasyon kullanılmaması durumunda).

3.4.5 Çevrim Süresi

Çevrim süresi (cycle time), montaj hattında, ürünün bir istasyonda kalabileceği en büyük süre veya bir iş istasyonundaki işçinin o istasyonda yapılması gerekli işleri tamamlaması için gerekli süre olarak tanımlanabilir.

Çevrim süresi, iş istasyonu süresine eşit veya daha büyük olabilen, iş istasyonundaki işçinin, işini tamamlayabilmesi için kullanabileceği süredir. Çevrim süresini seçmekteki ana düşünce, gerek duyulan üretim hızıdır.

Bir istasyonda, ardışık ögeler için iş tamamlama ve başlatma arasında bir süre geçer. Ayrıca o istasyondaki tüm işler bitmesine rağmen çevrim süresi dolmamış olabilir. Bu nedenle çevrim süresi üç alt süreye ayrılabilir: Üretken iş süresi, üretken olmayan iş süresi, atıl süre. Bunlar Şekil 3.1’de gösterilmiştir (WILD, 1979, s.47).



Şekil 3.1. n İstasyonlu Bir Montaj Hattında Çevrim Süresinin Alt Süreleri

Kuramsal olarak çevrim süresi, gerçekleşmesi istenen ürün çıktısından hesaplanabilir:

C : Çevrim süresi

T : Kullanılabilir üretim süresi

ÜS : Üretilmek istenen ürün sayısı

olmak üzere şu ifade yazılabilir:

$$C = T / \text{ÜS} \quad (3.2)$$

Örneğin bir üründen 8 saatlik bir çalışma süresi içinde 60 adet üretilmek istenirse, kurulmak istenen montaj hattının çevrim süresi;

$$C = 8 * 60 / 60 = 8 \text{ dk/adet}$$

3.4.6 Gerekli En Az İş İstasyonu Sayısı

Montaj hattındaki işlemleri, her istasyona, çevrim süresini tümüyle veya en az bir tanesi dışında tümüyle dolduracak şekilde atadığımızı düşünecek olursak gerekli en az iş istasyonu sayısı (n_{enk}) şu şekilde bulunacaktır:

$$n_{enk} = \left\lceil \sum_{i=1}^N t_i / C \right\rceil^+ \quad (3.3)$$

Burada $[x]^+$; x 'e eşit ya da x 'ten büyük en küçük tamsayı değeri ifade etmektedir.

Çevrim süresinin yarısından daha büyük süreye sahip iki iş ögesinin süreleri toplamı çevrim süresini aşacağı için bu ögeler, aynı istasyona atanamazlar. Dolayısıyla bu durumdaki iş ögeleri ayrı istasyonlarda bulunmalıdırlar.

Bu durumda gerekli iş istasyonu sayısı ($n_{olası}$) şu şekilde bulunacaktır:

$$n_{olası}: \quad \text{Çevrim süresinin yarısından daha büyük süreye sahip olan iş ögesi sayısı} \quad (3.4)$$

Bu durumda montaj hattını dengelemek için gerekli en az iş istasyonu sayısı (n_{enaz}), bu iki değerin en büyüğü olarak tanımlanır (IGNALL, 1965):

$$n_{enaz} = \text{Enb} (n_{enk}; n_{olası}) \quad (3.5)$$

İş ögesi süreleri toplamı 54 dk. olan bir montaj hattının çevrim süresi 8 dk/adet olarak belirlenmişse, bu hattı dengelemek için gerekli en az iş istasyonu sayısı;

$$n_{enk} = 54 / 8 \cong 7$$

olacaktır. Bu montaj hattındaki iş öğelerinden 6 tanesinin süresi çevrim süresi olan 8'in yarısından yani 4'ten büyükse bu durumda;

$$n_{olası} = 6$$

ve

$$n_{enaz} = \text{enb} (7, 6) = 7$$

olacaktır.

3.4.7. Ortalama İş İstasyonu Süresi

C^* : Ortalama iş istasyonu süresi

n : Dengeleme sonucunda elde edilen iş istasyonu sayısı

olmak üzere şu eşitlik yazılabilir (LEHMAN, 1969):

$$C^* = \sum_{i=1}^N t_i / n \quad (n \geq n_{enaz}, C \geq C^*) \quad (3.6)$$

Örneğin yukarıda verdiğimiz örnekteki montaj hattını herhangi bir dengeleme yöntemi ile dengelediğimizi ve dengeleme sonucunda elde edilen iş istasyonu sayısının 9 olduğunu düşünelim. Bu durumda;

$$C^* = 54 / 9 = 6 \text{ dk/adet}$$

olacaktır. Ayrıca

$$n \geq n_{\text{enaz}} (9 \geq 7)$$

ve

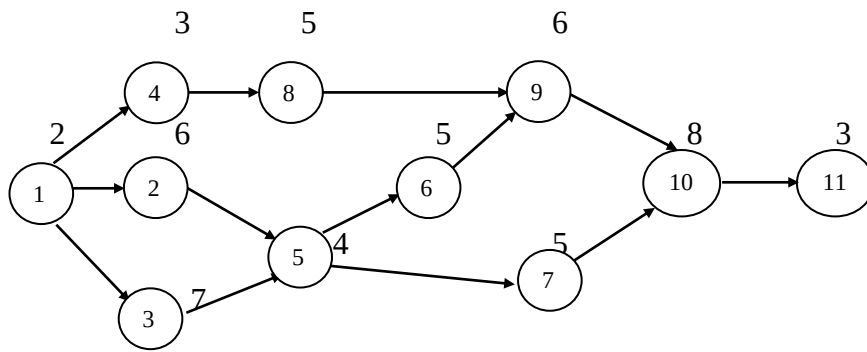
$$C \geq C^* (8 \geq 6)$$

koşulları da sağlanmıştır.

3.4.8. Teknolojik Öncelik Diyagramı

Montajın teknik özelliklerinden dolayı, bazı iş öğelerinin zorunlu olarak birbirini izlemesi gerekir. Bu özelliklerin tümü, öncelik ilişkileri adı altında toplanır (SNIEDOVICH, 1981). Bu ilişkiler genellikle bir grafik ile gösterilir. Bu grafik gösterim, hat dengeleme sistemlerinde öncelik ilişkilerinin belirtilmesinde çok yaygın olarak kullanılan teknolojik öncelik diyagramıdır (technological precedence network).

Bu diyagram; bir okla birbirine bağlanmış iki iş ögesinden okun çıktığı yönde bulunanın, okun ucunda bulunan iş ögesinden daha önce işleme alınacağını gösterir. Çemberlerin içindeki numaralar iş ögesi numaralarını, çemberlerin sağ üstündeki sayılar ise iş ögesi sürelerini gösterir. Şekil 3.2’de bir montaj hattına ilişkin teknolojik öncelik diyagramı görülmektedir.



Şekil 3.2. 11 İş Öğeli Bir Teknolojik Öncelik Diyagramı

3.4.9. Öncelik Matrisi

Teknolojik öncelik diyagramının üst üçgensel matris durumuna dönüştürülmüş şeklidir (DAR-EL ve RUBINOVITCH, 1979). Bu matriste, aralarında doğrudan veya dolaylı öncelik ilişkisi bulunan iş ögeleri için, matriste önde gelen iş ögesi numaralı satırla; izleyen iş ögesi numaralı sütunun kesiştiği göze

"1", diğer gözlere ise "0" konur. Tablo 3.1’de, Şekil 3.2’de teknolojik öncelik diyagramı verilen montaj hattı için hazırlanmış öncelik matrisi (precedence matrix) görülmektedir.

Tablo 3.1 Şekil 3.2’deki Teknolojik Öncelik Diyagramına İlişkin Öncelik Matrisi

		ARTÇIL ÖĞELER										
		1	2	3	4	5	6	7	8	9	10	11
Ö N Ç Ü L Ö Ğ E L E R	1	-	1	1	1	1	1	1	1	1	1	1
	2		-	0	0	1	1	1	0	1	1	1
	3			-	0	1	1	1	0	1	1	1
	4				-	0	0	0	1	1	1	1
	5					-	1	1	0	1	1	1
	6						-	0	0	1	1	1
	7							-	0	0	1	1
	8								-	1	1	1
	9									-	1	1
	10										-	1
	11											-

3.4.10. Esneklik Oranı

Bir montaj sürecinin öncelik yapısının belirlenmesinde kullanılan bir ölçüttür (BROSCH ve WIENDAHL, 1990). Esneklik oranı (flexibility ratio), bir montaj sürecinde elde edilebilen uygun sıraların sayısının bir ölçüsüdür.

E : Esneklik oranı

Y : Öncelik matrisinde sıfır (0) değerine sahip göz sayısı

olmak üzere şu ifâde yazılabilir:

$$E = (2*Y) / [N*(N-1)] \quad (3.7)$$

Bu oran, teknolojik öncelik diyagramı seri şekilde sıralıysa sıfıra (0), öncelik ilişkileri olmayan diyagramlar için ise 1’e eşittir. E oranının sıfıra yakın olması, istasyonlara iş ögesi atamada esnekliğin az olduğunu göstermektedir.

Tablo 3.1’de verilen öncelik matrisinden elde edilen esneklik oranı (flexibility ratio),

$$E = 2*13 / (11*10) = 0,236$$

'dır.

3.4.11. Denge Kaybı

Denge kaybı (balance loss); işlerin, işlemciler veya istasyonlara dağıtımının ne ölçüde dengeli olduğunu gösteren bir ölçüttür (KAO, 1979).

D : Denge Kaybı

olmak üzere şu ifâde yazılabilir:

$$D(\%) = [(C - C^*) / C] * 100 = \left[(n * C - \sum_{i=1}^n t_i) / (n * C) \right] * 100 \quad (3.8)$$

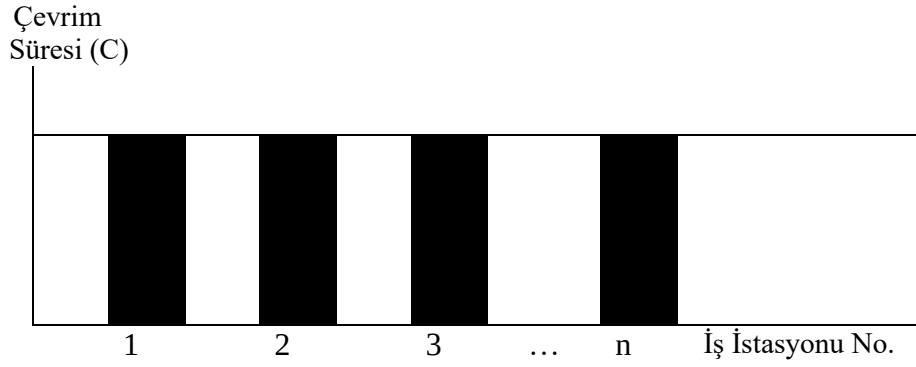
Daha önce verilen örnek için denge kaybı,

$$D (\%) = [(9*8 - 54) / (9*8)] * 100 = \% 25$$

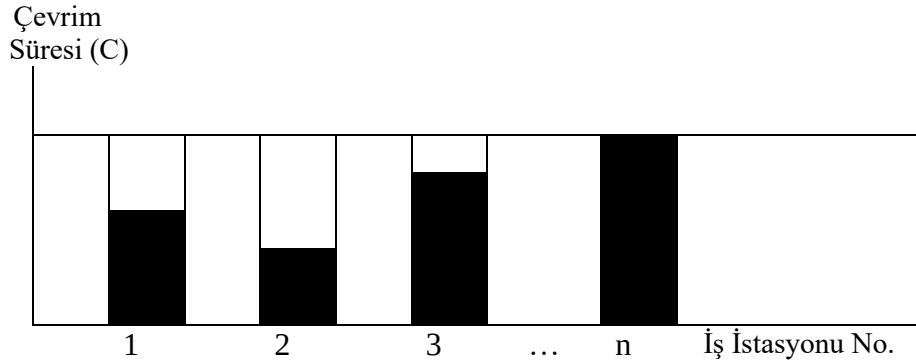
olacaktır.

Denge kaybı, her istasyonda, birim üretim için ayrılan toplam süreyle gerekli süre arasındaki farkın, ayrılan süreye oranıdır ve çoğunlukla sıfırdan büyük bir değerdir. Ancak bu değer 0 olması ideal durumdur.

Denge kaybının bulunmasında kullanılan çevrim süresi (C), eğer tüm istasyon süreleri çevrim süresinden küçükse, en büyük istasyon süresi olarak alınabilir. Çünkü en büyük istasyon süresi ve C arasındaki fark süre, hiç bir istasyonda kullanılmayacaktır. Dolayısıyla bu fark süreyi, kullanılabilecek süre içinde düşünmek gereksizdir. Şekil 3.3.'de tam dengelenmiş ve tam dengelenmemiş hat durumları gösterilmiştir.



a) Tam Denge



b) Tam Kurulamamış Denge

Şekil 3.3. n İstasyonlu Bir Montaj Hattında Tam ve Tam Kurulamamış Denge Durumları

3.4.12. Hat Etkinliği

Toplam iş süresi, iş istasyonlarına atanan iş öğelerinin süreleri toplamı olan iş istasyonu süresi değerlerinin genel toplamıdır. Bir anlamda bu süre, hat nasıl dengelenirse dengelensin, etkin olarak gerçekleştirilen işlerin toplam süresidir. İşte bu toplam etkin sürenin (toplam iş süresinin) dengeleme sonucunda montaj için ayrılması gerektiği saptanan süreye oranı, **hat etkinliği** olarak adlandırılan değeri verir.

HE : Hat etkinliği

olmak üzere şu ifade yazılabilir:

$$HE(\%) = \left[\sum_{i=1}^N t_i / (n * C) \right] * 100 \quad (3.9)$$

Hat etkinliği; montaj hattındaki toplam işgücünün ne kadarlık bir yüzdesinin kullanıldığını gösteren önemli bir ölçektir.

3.5. Montaj Hatlarında Model Sayısının Etkisi

Hem mekanik hem de mekanik olmayan hatlarda, işlerin, yapılacağı istasyonlara atanması istenir. Amaç, iş istasyonlarındaki işlem veya montaj sürelerini eşitlemektir. Sorun bâzen, aynı üretim hattında birden fazla benzer ürün yapılabileceği gerçeğiyle karmaşıklaşır. Bu karmaşıklık; üç farklı akış durumunun -ve bu nedenle üç farklı tip hat dengeleme probleminin- tanımlanmasını gerektirir.

Montaj hatları üzerindeki bu üç farklı üretim durumu, hat üzerinde yapılması gereken ürün veya ürünlere göre tanımlanır. Montaj hattı, tek bir ürünün üretilmesi için mi kullanılacaktır ? Yoksa birçok farklı modeli üretmek için mi kullanılacaktır, eğer öyleyse bunlar hat üzerinde nasıl tasarlanacak ve sıralanacaktır ? Bu soruları yanıtlamak için tanımlanabilecek üç durum vardır:

1. Tek Modelli Hatlar.
2. Çok Modelli Hatlar.
3. Karışık Modelli Hatlar.

Tek modelli hatlar; tek bir model veya ürünün üretimine ayrılmış özel hatlardır. Bu tip hatların tasarımı daha basittir.

Çok modelli hatlar; farklı ürünler veya aynı ürünün iki veya daha çok benzer tipinin, ayrı yığınlar hâlinde üretildiği hatlardır. Her model, bu hat üzerinde ayrı bir yığın oluşturur. Ürünler veya modeller, genellikle iş öğelerinin benzer bir sırasını gerektirir. Bu nedenden dolayı, değişik modellerin üretimi için aynı hat kullanılabilir.

Çok modelli montaj hatları, eğer yığınlar büyük ise tek modelli montaj hatlarına, küçük ise karışık modelli montaj hatlarına benzerlik gösterirler.

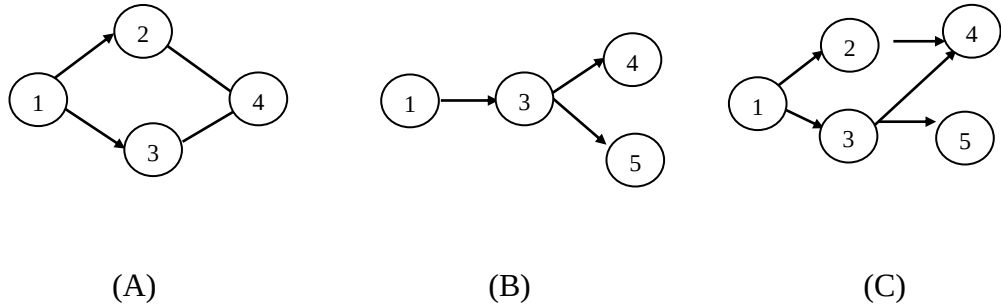
Çok modelli bir montaj hattının tasarımı için yordam, başlıca şu adımlardan oluşur (MONDEN, 1983, s.181):

1. Bir çevrim süresinin, istenen üretim hızlarına bağlı olarak belirlenmesi.
2. Gerekli en az iş öğesi sayısının hesaplanması.

3. Elemanter iş öğeleri arasındaki birleştirilmiş (bütünleşik) teknolojik öncelik diyagramının hazırlanması.
4. Hattın dengelenmesi.
5. Farklı ürünlerin/modellerin hatta verilmeleri arasındaki sürenin saptanması.
6. Hatta giren çeşitli ürünler/modeller için bir sıranın belirlenmesi.

Öncelikle farklı ürünlerin istenen üretim miktarlarına bağlı olarak bir ortak çevrim süresi saptanır. Her bir ürünün üretimi için gerekli iş öğeleri saptanıp teknolojik öncelik diyagramları oluşturulduktan sonra bunlar tek bir diyagramda birleştirilir.

Birleştirilmiş teknolojik öncelik diyagramı kavramı, THOMOPOULOS (1970) tarafından geliştirilmiş, biçimsel şekilde MACASKİLL (1972) tarafından formüle edilmiştir. Şekil 3.4’de, iki ayrı modelin teknolojik öncelik diyagramları ve bunların tek bir diyagram halinde birleştirilmiş durumu görülmektedir. Birleştirilmiş diyagram, farklı modeller arasındaki öncelik ilişkilerinde bir çelişkinin olmadığı durumlar için uygundur. Örneğin, eğer A modelinde 1 no.lu iş öğesi 2 no.lu iş öğesinden önce geliyorsa, diğer hiç bir modelde 2 no.lu iş öğesi, 1 no.lu iş öğesinden önce gelemez (MACASKILL, 1972).



Şekil 3.4. 1. Ürün (A) ile 2. Ürün (B) için ve Birleştirilmiş (C) Teknolojik Öncelik Diyagramları

Dengeleme sırasında paralel istasyon (aynı iş istasyonları için ek operatör) kullanılabilir. Paralel istasyon kullanmanın aşağıdaki yararları vardır (BARD, 1989):

1. Gereken toplam istasyon sayısının azaltılmasıyla denge kaybı azalır.
2. Yinelenen her istasyon, özgün olanın iki katı etkin çevrim süresine sahiptir. Bu nedenle daha iyi bir uyum sözkonusudur. Bu özellikle, iş

öğelerinin bir alt kümesi, çevrim süresine yakın iş ögesi süreleri toplamı kadar süreye sahip olduğunda doğrudur.

3. İş ögesi sürelerinin herhangi biri çevrim süresinden büyük olduğunda, istenen çevrim süresinin karşılanması olanaklı olur.
4. Bir serî hatta herhangi bir istasyonun başarısızlığı tüm hattın durmasına yol açar. Oysa paralel duruma getirilmiş bir istasyonun başarısızlığı, etkinliklerin sürmesini engellemez.

Karışık modelli hatlar; iki veya daha çok benzer ürünün veya bir ürünün değişik modellerinin, aynı anda ve karışık olarak üretildiği montaj hatlarıdır. Bu tür üretimde, kuramsal olarak büyük miktarlarda bitmiş ürün stoklarına gereksinim olmayıp, çok modelli hatların tersine, tüketicinin istekleri, sürekli bir şekilde yapılan üretimle karşılanır. Otomobil ve kamyon montaj hatları, bu duruma örnek olarak verilebilir.

Karışık modelli hatlarda gözlenen temel olumsuzluk, modellerin özelliğinden kaynaklanan, ayrı iş parçalarının; eşit olmayan iş akışlarına, boş istasyon sürelerine ve daha fazla istasyon sayısına neden olmasıdır. Bu tip montaj hatlarında oldukça karmaşık tasarım sorunları söz konusudur..

3.6. Basit ve Genel Montaj Hattı Dengeleme Problemlerinin Tanımlanması

Basit montaj hattı dengeleme problemi (simple assembly line balancing problem - BMHDP) iki türdür. Bu problemin tanımı, aşağıda verilen ilk dokuz madde ve ona ek olarak verilen diğer iki maddenin herhangi biridir.

Yâni birinci tür BMHDP ilk on maddeyi, ikinci tür BMHDP ise ilk dokuz madde ve onbirinci maddeyi içerir (BAYBARS, 1986b).

1. Tüm girdi parametreleri belirlidir.
2. Bir iş ögesi, iki veya daha çok istasyon arasında bölünemez.
3. İş öğeleri, teknolojik öncelik gereksinimlerinden dolayı, keyfi sıralarda işlem göremez.

4. Tüm iş öğeleri yapılmalıdır.
5. İstasyonlar tüm iş öğelerini yapmak için gerekli donanım ve işgücüne sahiptir.
6. İş ögesi süreleri, yapıldıkları istasyonlardan ve önceki/sonraki iş öğelerinden bağımsızdır.
7. Her işlem her istasyonda yapılabilir.
8. Tüm hat, besleyici veya paralel alt montaj hatlı olmayacak şekilde seri olarak düzenlenmelidir.
9. Montaj sisteminin, tek bir ürünün tek bir modeli için tasarımılandığı varsayılır.
10. Çevrim süresi verilmiştir ve sabittir.
11. İstasyon sayısı verilmiştir ve sabittir.

İlk dokuz maddeden herhangi birinde değişiklik olursa bu durumdaki problem, bir genel montaj hattı dengeleme problemi (general assembly line balancing problem) olur. Örneğin iş ögesi süresinde değişkenlik olabilir. Bu durumun bazı nedenleri şunlar olabilir:

1. Hatta hatalı malzeme veya parça gelebilir.
2. Kullanılan donanım sorun çıkarabilir.
3. İşçinin temposu (performansı) değişebilir.
4. Hatalı montaj yapılması durumunda hata, hatta düzeltilebilir.

İş istasyonları arasında emniyet stokları bulundurularak, bu tür aksaklıklar önlenmeye çalışılır.

3.7. Montaj Hattı Dengeleme Yöntemlerinin Sınıflandırılması

3.7.1. Probleme Göre Sınıflandırma

Montaj hattı dengeleme problemi, farklı ölçütler dikkate alınırca çeşitli tiplerde olabilir.

- 1) Amaç sayısı
- 2) İşlem süreleri
- 3) Ürün/Model sayısı
- 4) Paralel tezgâh durumu
- 5) İstasyondaki işçi sayısı
- 6) Hattın durumu
- 7) İstasyon durumu
- 8) Kaynak kısıtı
- 9) Malzeme ikâmesi
- 10) Kusur oranlarının verilmesi durumu
- 11) Özel problemler (Öğrenme, mâliyet, atanan iş öğelerinin istasyon içinde sıralanması vb.)

3.7.2. Çözüm Yaklaşımına Göre Sınıflandırma

3.7.2.1. Sezgisel (Bulgusal) yöntemler

Bu yöntemler, belirli bir yordamın (prosedürün) izlenmesi ve belirli varsayımların yapılması yoluyla, montaj hatlarının dengelenmesi konusunda yaklaşık çözüm verirler. Şimdiye kadar geliştirilen oldukça çok sayıda sezgisel yöntem vardır. Bu yöntemlerin çoğu, çevrim süresini sabit kabul ederek istasyon sayısını ve buna bağlı

olarak dengeleme kaybını enküçüklemeye çalışır. Birkaç tanesi de istasyon sayısını sabit kabul ederek en uygun çevrim süresini saptamaya çalışır.

Bu yöntemlerin en çok kullanılanları; J.R.JACKSON (1956) tarafından geliştirilen "**aşamalı sıralamayla çözüm**", W.HELGESON ve D.P.BIRNIE (1961) tarafından geliştirilen "**konum ağırlıklı dengeleme tekniği**", M.D.KILBRIDGE ve L.WESTER (1961) tarafından geliştirilen "**Kilbridge-Wester yöntemi**" ve A.L.ARCUS (1966) tarafından geliştirilen "**COMSOAL tekniği**" olup, bu yöntemlerin sayısı zaman içinde oldukça artmıştır.

Sezgisel yöntemler, birbirlerinden çeşitli açılardan ayrılırlar. Örneğin JACKSON (1956), KILBRIDGE-WESTER (1961) ve SALVESON (1955)'un geliştirdiği yöntemlerde sezgisel kurallar ön plana çıkarlar. Oysa diğer yöntemler, daha az esnek yapıların kullanımını gerektirir. Bâzı yöntemler işlem sürelerine ek olarak öncelik sıralarını gözönüne alıp ilerideki işlem sürelerini de kullanırlar.

Sezgisel yöntemler, birbirlerinden çeşitli açılardan ayrılırlar. Örneğin Jackson, Kilbridge-Wester ve Salveson'un geliştirdiği yöntemlerde sezgisel kurallar ön plana çıkarlar. Oysa diğer yöntemler, daha az esnek yapıların kullanımını gerektirir. Bâzı yöntemler işlem sürelerine ek olarak öncelik sıralarını gözönüne alıp ilerideki işlem sürelerini de kullanırlar.

3.7.2.2. Analitik yöntemler

Bu yöntemler, "**Matematiksel programlama yöntemleri**" olarak da anılırlar ve en uygun sonucu verirler. Bu tip yöntemlerin ilki, E.H.BOWMAN (1960) tarafından geliştirilen "**doğrusal tamsayılı programlamayla çözüm**" olup, daha sonra 0-1 tamsayılı programlamayı kullanan yöntemler de geliştirilmiştir (Örneğin TALBOT ve PATTERSON (1984)). Bu yöntemlerde kısıt ve amaç denklemleri bulunur. Özellikle işlem sayılarının arttığı durumlarda çözüm bulmak zorlaşmaktadır. Bu nedenle montaj hattı dengeleme problemlerine bulgusal yöntemlerle yaklaşılması daha yaygın bir durumdur.

3.7.2.3. Benzetim (Simülasyon) teknikleri

Benzetim tekniği, yöneticinin denemeler yoluyla karar vermesini kolaylaştırır. Herhangi bir sistemin işleyişini anlamak veya bu sistemin işleyişi ile ilgili değişik stratejileri değerlendirmek için sistemin zaman içindeki çalışmasını taklit eden bir bilgisayar modelinin kurulması ve bu model ile deneyler yapılmasına "**benzetim**" denir.

Benzetim tekniğinin uygulanmasındaki en önemli gerekçeler şöyle sıralanabilir: Gerçek yaşamda, herhangi bir sistemi veya işlem dizisini gözlemek olanaksız veya çok masraflı olabilir. Gözlemlenen sistem o kadar karmaşık olabilir ki, bu sistemi matematiksel denklemlerle tanımlamak ve sistem işleyişi ile ilgili kestirime yardımcı analitik çözümleri elde etmek olanaksız olabilir. İrdelenen sistemin matematiksel modeli kurulabilse bile, modele çözüm getirmek için gereken analitik teknikler yetersiz kalabilir. Sistemi tanımlayan matematiksel modellerin doğrulanmasına yönelik deneylerin yapılması, olanaksız veya çok masraflı olabilir.

Bu üstünlüklerin yanısıra bâzı dezavantajları da vardır. Bunlardan en önemlileri şunlardır: Modelin tasarımı ve kullanımı, zaman alıcı ve masraflıdır. Her benzetim modeli diğerinden farklıdır; bunun sonucu olarak model doğrulaması, çok dikkat isteyen ve zor bir iş olarak belirir. Benzetim modelleriyle en iyi sonuçları elde etmek çok zordur. Bu ise benzetim modellerinin, çözülen modellerden çok, deneysel modeller olmalarından kaynaklanır. Diğer bir deyişle benzetim tekniği, seçeneklerin değerlendirilmesinde daha yaygın olarak kullanılır (ACAR ve EŞTAŞ, 1986, s.110).

Montaj hatlarında istasyonlar arasındaki stok düzeylerinin belirlenmesi için çeşitli matematiksel modeller geliştirilmiştir. Genelde bu modellerin uygulamaya konulması oldukça zordur. Çünkü, akış hatları dinamik sistemler olup, bunların gözlemlenmesi, karmaşık yapılarına uygun bir modelin kurulabilmesi ve bu modelin denenmesi çok masraflı veya olanaksız olabilmektedir.

Son yıllarda benzetim tekniği, bu tür hatlarda ara stok düzeylerinin belirlenmesinde oldukça yaygın olarak kullanılmaktadır. Matematiksel yöntemlerle en iyi çözüm bulunmasına karşın, bu tür modelleri gerçek sisteme uyarlayabilmek için o kadar çok varsayım ve olasılık hesapları yapılmaktadır ki, bu nedenle benzetim tekniğinin

çeşitli seçenekleri deneyerek bulduğu yaklaşık çözüm, genellikle bu en iyi çözümden çok daha kullanılabilir olmaktadır. Benzetim tekniği, paralel hatların kullanılmasında da yararlanılan bir tekniktir.

3.7.3. İşlem Sürelerinin Deterministik Olup Olmaması Durumuna Göre Sınıflandırma

Montaj hattı dengeleme problemlerinde, çok genel iki farklı durum söz konusudur:

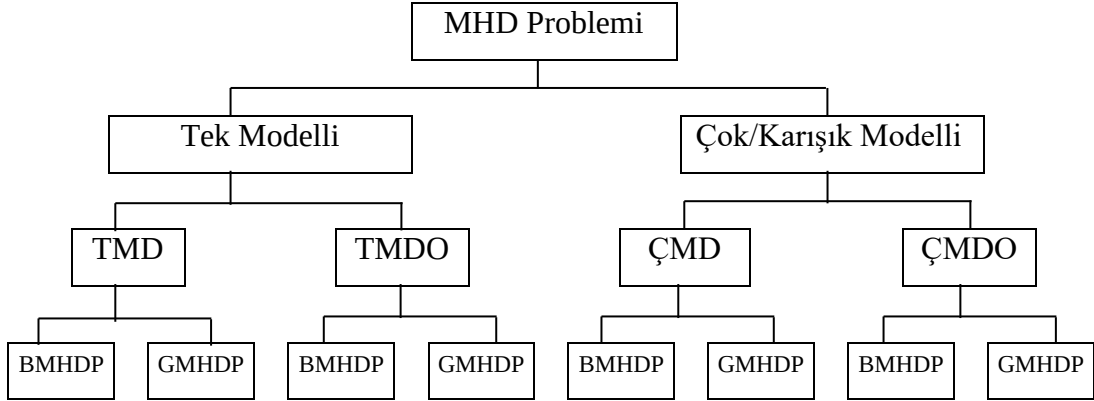
1. İşlem sürelerinin belirli olması (Deterministik montaj hattı dengeleme)
2. İşlem sürelerinin değişken olması (Deterministik olmayan montaj hattı dengeleme)

Deterministik montaj hattı dengeleme problemlerinde, işlem sürelerinin verilmiş (belirli) olduğu ve bu sürelerin bir birimden diğerine herhangi bir değişim göstermediği varsayılmaktadır. Bu varsayım, özellikle robot teknolojisinin uygulama alanı bulabildiği, ileri teknoloji endüstrilerinde geçerlidir.

Deterministik olmayan montaj hattı dengeleme problemlerinde ise işlem süreleri, belirli bir dağılımla ifade edilir. İnsan unsuru, işlem sürelerinin değişken olmasına yol açmaktadır. Bu değişkenliğin nedenleri arasında şunlar sayılabilir: Yorulma, dikkatin dağılması, yetersiz nitelikli işgücü, iş doyumsuzluğu, hatalı girdiler, araç/gereç bozulmaları. Bu durum, istasyonlara atanan iş öğelerinin toplam süresinin, çevrim süresini aşmasına ve dolayısıyla bazı işlemlerin zamanında bitirilememesine neden olabilmektedir. Özellikle işler arasındaki öncelik ilişkileri gözönüne alındığında, bazı işlemlere geç başlanmaktadır. Bu tür dengeleme problemlerinde araştırmacıların çoğu, işlem sürelerinin normal dağılıma göre değer aldıklarını, bazıları ise işlem sürelerinin değişkenlik katsayılarının tüm iş öğeleri için sabit olduğunu varsayımlardır. İşlem sürelerinin herhangi bir dağılıma uygun olmaması durumunda ise stokastik bir yapı karşımıza çıkar.

Montaj hattı dengeleme problemleri, dört grupta toplanabilir: Tek modellenli deterministik (TMD), tek modellenli deterministik olmayan (TMDO), çok/karışık modellenli deterministik (ÇMD) ve çok/karışık modellenli deterministik olmayan (ÇMDO)

problemler. Şekil 3.5.'de montaj hattı dengeleme (MHD) problemlerinin sınıflandırılması verilmiştir.



Şekil 3.5. Montaj Hattı Dengeleme Problemlerinin Sınıflandırılması

3.8. Konum Ağırlıklı Dengeleme Tekniği

Bu yöntem, 1960'lı yılların başlarında Helgeson ve Birnie tarafından General Electric Company'de geliştirilmiş olan sezgisel bir yöntemdir. Diğer yöntemlere göre daha hızlı ve iyi çözümler veren ancak optimal çözümü vermeyi garanti etmeyen bulgusal bir yöntemdir. Her iş ögesine, kendisinden sonra gelen iş ögelerinin toplam süresinin büyüklüğüne göre bir ağırlık verilir. Bu ağırlığa **konum ağırlığı** denir. Burada amaç, konum ağırlığı büyük olan iş ögesine, atamada öncelik vermektir (Helgeson ve Birnie, 1961).

Bu yöntemde şu adımlar izlenir:

1. Her iş ögesi için konum ağırlıkları belirlenir (Bir iş ögesinin konum ağırlığı; o işlemi yapmak için gerekli süre ve seri olarak ondan sonra gelen işleri yapmak için gerekli sürelerin toplamıdır).
2. İş ögeleri, azalan konum ağırlıklarına göre sıralanır.
3. İş ögeleri iş istasyonlarına atanır (Bu atama işlemi sırasında en yüksek konum ağırlıklı iş ögelerine öncelik verilir).
 - a. En büyük konum ağırlıklı iş ögesi ilk istasyona atanır.
 - b. İş istasyonunun kullanılmamış süresi; çevrim süresinden, atanmış süreler toplamı çıkarılarak bulunur.

- c. Kalan iş öğeleri içinden en büyük konum ağırlıklısı seçilir ve aşağıdaki kontrollardan sonra iş öğesi, istasyona atanmaya çalışılır:
1. Ayrılmış işler listesi kontrol edilir. Eğer öncülü olmayan iş öğeleri atanmışsa öncelik koşulu bozulmayacaktır. Bu durumda adım c-2'ye, aksi durumda adım-d'ye gidilir.
 2. İş öğelerinin süresi, istasyonun kullanılmamış süresi ile karşılaştırılır. Eğer işlem süresi, atanmamış süreden küçükse, iş öğesi istasyona atanır ve kullanılmamış istasyon süresi yeniden hesaplanıp adım-c'ye dönlür. Eğer işlem süresi, kullanılmamış süreden büyük ise adım-d'ye geçilir.
- d. Seçme, kontrol etme ve olanaklı ise istasyona atama, aşağıdaki iki koşuldan biri sağlanıncaya kadar sürdürülür:
1. Tüm iş öğelerinin atanması bitmiştir.
 2. Hem öncelik koşullarını, hem de atanmamış süre koşullarını sağlayan iş öğesi kalmamıştır.
- e. Atanmamış en yüksek konum ağırlıklı iş öğesi bir sonraki istasyona atanıp, ilk dört adım (a, b, c ve d) aynen uygulanır.
- f. İş öğelerini istasyonlara atama işlemi; atanmamış iş öğesi kalmayıncaya dek sürdürülür.

Bu anda, montaj hattını dengeleme problemi çözümlenmiştir. Elde edilen çözümün kontrolü için bir seçenek çözüm tekniği uygulanabilir. Bu teknikte; hattın sonundan başlanarak ve kurallar tersine uygulanarak **ters konum ağırlıkları** ölçülür. Benzer adımların yinelenmesiyle elde edilen seçenek çözüm, ilk çözümle karşılaştırılarak, daha iyi olan çözüm kullanılır.

Uygulamada bu yöntem sıkça kullanılmaktadır. Bunun nedeni, iş öğelerinin, kendisi ve kendisine bağlı (o iş öğesi bitirilmeden başlanamayan) iş öğelerinin süreleri toplamı yüksek olan iş öğelerinin daha önce yapılmasını sağlamaktır. Bu yöntem, hat

dengeleme konusunda çalışan arařtırmacılara da, yeni yöntemler geliştirme konusunda ışık tutmuřtur.

3.9. Ařamalı Sıralamayla Çözüm

1956 yılında Jackson tarafından geliştirilen bu yöntemin, hat dengelemesi sorununa getirdiđi yaklaşım, çözüm ilerledikçe, küçük değerli seçeneklerin sistemli bir şekilde yok edilmesi temeline dayanır (Jackson, 1956).

Yöntemin temel ilkesi; çevrim süresini aşmamak, öncelik koşullarına uymak ve tüm işlemlerin olası tüm karışımalarını aşamalı olarak belirlemektir. Daha sonra, çözümü kolaylařtırmak için bâzı sâdeleřtirmeler yapılabilir.

3.10. Öncelik Diyagramı İle Çözüm

1963 yılında Hoffman tarafından geliştirilen bu yöntem; ana çözüm aracı olarak öncelik diyagramının kullanıldığı ve genellikle en uygun çözümü veren bir hat dengeleme yöntemidir (Hoffman, 1963).

4. GENETİK ALGORİTMALAR

4.1. Genetik Algoritmanın Tarihçesi

Genetik Algoritma”lara adını veren genetik bilimi, biyolojinin kalıtımla ilgilenen dalı olup, “kalıtım” ise, bâzı genetik özelliklerin bir kuşaktan diğer kuşağa aktarılma sürecidir. 1859 yılında Charles Darwin [1809-1882] tartışmalara yol açan *Türlerin Orijini* (The Origin of Species) adlı kitabını yayımladı. Bu çalışmada türlerin varolduğundan bu yana sürekli olarak gelişmekte olduğunu ve yoğun tartışmalara neden olan bunun bir diğer anlamının da, insan ırkının kuyruksuz maymun ırkından gelebileceği olduğunu öneren tezini ortaya attı.

Araştırmaları süresince türler arasındaki çeşitlilikten oldukça etkilenen Darwin, neredeyse yaşayan tüm organizmaların oldukça yüksek bir üreme potansiyeline, yumurta veya sporlarla sahip olduklarına ama bunlardan salt küçük bir yüzdenin hayatta kalabildiklerine dikkat çekti. Aynı zamanda bir popülasyon içindeki varyasyonların çok sayıda olduğunu gözlemledi.

Bu varyasyonlardan salt muhtemelen daha güçlü olan bireyler yaşam mücadelesinde başarılı olup hayatta kalabiliyorlardı. Bireyler arasındaki çeşitlilik kendilerine ailelerinden gelen kalıtsal mirasla oğul döllere aktarılıyor ve evrim bu miras varyasyonlardan doğal seleksiyon yoluyla gerçekleşiyor olmalıydı.

Aynı tarihlerde çağdaş genetik biliminin kurucusu sayılan Gregor Mendel (1822-1884), deneylerinde organizmalar arasındaki karakter ve özelliklerin oğul döllere aktarımı konusunda araştırmalarını bezelyeler üzerinde sürdürüyordu. Farklı türlerden alınmış bitkilerin birbirleriyle melezleştirilmesi sonucu ortaya çıkan karakter etkileşimi üzerine incelemelerde bulunmuş ve bir sonraki kuşağa aktarılacak özelliklerin birtakım kurallara uyduğunu ortaya çıkarmıştı.

Örneğin kısa ve uzun boylu bitkilerin birbirleriyle çaprazlaması sonucu elde edilen bitkilerin hepsinin boy karakteri uzun olarak ortaya çıkmıştı. Buradan uzun boy karakter geninin baskın (dominant), kısa boy özelliğinin ise çekinik (resesif) olduğu, dolayısıyla fenotipte ortaya çıkamadığı kuramını ilân etti. Mendel'in bu çalışmaları, genetik bilimine ilişkin ilk çalışmalar olarak kabul edilmektedir. Bu çalışmalar, Mendel'den 30 yıl sonra Walter Sulton (1877-1916)'un genlerin hücre çekirdeği içinde yer alan kromozomların bir parçası olduğunu keşfetmesiyle sürecekti.

Her ne kadar Darwin'in kuramı, türlerde oluşan varyasyon değişiminin sürekli olduğu konusu üzerinde vurgulamalar yapsa da, buna zıt olarak bazı türlerde oluşan âni kesikli değişimleri açıklamada yetersiz kalıyordu. Hugo De Varis (1848-1935), bitkilerde zaman zaman oluşan bu değişimleri gözlemlemişti.

Bu âni değişimleri açıklamak üzere mutasyon kuramını geliştirdi. Kromozomlar üzerindeki genlerde farklı dış etkenlerden dolayı oluşabilen bu değişim, belli bir karakter özelliğini kontrol eden bir gene isâbet ettiğinde çok dramatik sonuçlara yol açabilmektedir. Ama bir dizi genin kontrol ettiği bir karakteristikte ihmâl edilebilir bir etki de oluşturabilmektedir. Bu bağlamda mutasyon kuramı, günümüz çağdaş genetik biliminin kabul ettiği ve Darwinizmle keskin ayrımları olmayıp çeşitliliği açıklamakta farklı bir bakış açısı olarak örtüşmektedir (Fleming, P.J., Zalzal, A.M.S., 1997).

Sonuç olarak yukarıda genel hatlarıyla sözedilen Darwin'in evrim kuramından ve genetik biliminden etkilenecek canlılarda yaşanan genetik süreci bilgisayar ortamında gerçekleştirme düşüncesi; çağdaş genetik algoritmalarının temeli olarak adlandırılacak ilk örnek çalışmalar, 1960 yılında Rechenberg [1] tarafından önerilmiştir. Bu önerilen tekniğin adı "Evrin Stratejileri", Almanca orijinal adıyla "Evolutionstrategie" idi.

Ana amacı uçak kanatlarındaki kapakçıkların belirli bir amaç fonksiyonu altında tasarlanması olan teknik, daha sonraki yıllarda Schwefel [9, 10]'ın çalışmalarının diğer araştırmacıları bu alan üzerinde odaklamasıyla sürdü. Temel olarak evrim stratejileri tekniği, olası çözüm adayları olarak, ana birey ve ana bireyden mutasyon yoluyla türetilmiş olan oğul bireyleri içermekteydi [6] .

Bu tekniğin bir sonraki evresi olan evrimsel programlama (Evolutionary programming) günümüzdeki çağdaş Genetik Algoritmaya benzerliği ile en yakın olanı olup, yöntem Fogel, Owens ve Walsh tarafından geliştirilmiştir. Burada bir önceki aşamada tanıtılan özelliklere ek olarak yeni üretilen çözüm adaylarının belirli bir uygunluk fonksiyonu altında değerlendirilmesi eklenmiştir.

“Genetik Algoritma” kavramından ilk sözeden ve konu ile ilgili ilk yayını yapan ise Bagley (1967)’dir. Bagley ile aynı tarihte Rossenberg, biyolojik ve benzetim esaslı bir çalışma yapmıştır (Goldberg, 1989).

De Jongs, 1975 yılında matematiksel fonsiyonları genetik algoritma ile çözmeye çalışmıştır. De Jong, fonsiyonların enküçüklenmesi (minimizasyonu) için altı ayrı problem incelemiştir (Goldberg, 1989):

1. Sürekli ve sürekli olmayan fonsiyonlar,
2. Dışbükey (konveks) ve dışbükey olmayan fonsiyonlar,
3. Tek ve çok değişkenli modeller,
4. Kuadratik ve kuadratik olmayan fonsiyonlar,
5. Düşük ve yüksek dizili fonsiyonlar,
6. Deterministik ve stokastik problemler.

Bu çalışmayı aynı tarihlerde Michigan Üniversitesi’nde psikoloji ve bilgisayar bilimi uzmanı olan John Holland’ın önderliğindeki “Cellular Automata” çalışmaları izlemiştir [Hasşerbetçi, 1997]. Mekanik öğrenme (machine learning) konusunda çalışan Holland, Darwin’in evrim kuramından etkilenecek canlılarda yaşanan genetik süreci bilgisayar ortamında gerçekleştirmeyi düşünmüştür.

Tek bir mekanik yapının öğrenme yeteneğini geliştirmek yerine böyle yapılarda oluşan bir topluluğun çoğalma, çiftleşme, mutasyon vb. genetik süreçlerden geçerek başarılı (öğrenebilen) yeni bireyler oluşturabildiğini gördü. Çalışmalarının sonucunu açıkladığı "Doğal ve Yapay Sistemlerde Uyarlama” adlı kitabının 1975’te yayımlanmasından sonra geliştirdiği yöntemin adı *Genetik Algoritmalar* (kısaca GA) olarak yerleşti.

Holland'ın bu çalışmada 2 hedefi vardı:

- doğal uyarlama sürecinin anlaşılmasını geliştirmek,
- doğal sistemdeki benzer özelliklere sahip yapay sistemlerin tasarlanması.

Buradaki temel düşünce şu şekilde ifade edilebilir: Verilen bir popülasyonun genetik havuzu potansiyel olarak istenen çözümü içerir veya uyarlanan probleme ilişkin iyi bir çözüm vardır (Holland J.H., 1975).

Bu çözüm aktif değildir, dikkate alınan genetik kombinasyon, birkaç nesne arasında bölünmüştür. Yalnızca farklı genomların birleşimi sonucunda ortaya çıkabilen bir sonuç bizi çözüme götürebilmektedir. Bunu da Holland'ın yönteminde belirttiği gibi yeniden üreme (reproduction) ve crossover (genomlarının çaprazlanması) operatörleri sağlamaktadır.

Kullanılan bu yöntem özellikle daha etkilidir, çünkü salt mutasyonu değil, yanısıra crossover kullanarak yeniden üremeyi sağlar, böylelikle yeni bireylere aktarılan genlerin aranılan çözüme daha da yakınsattığı gerçeğinden hareketle sonuçta en iyi (optimum) çözüme ulaşılır (Emmeche C., 1994).

Temel ilkeleri ve ana hatları Holland'ın kitabında belirtildiği gibi olan Genetik Algoritmanın gelişimini, yine Holland'ın öğrencisi olarak doktorasını veren inşaat mühendisi David E. Goldberg'in, 1985 yılındaki çalışmaları sağlamıştır. Bu çalışmadaki amaç, doğal gaz borularındaki kayıpları, basınç oranlarını değiştirmek sûretiyle, kompresörlerin enerjilerini enküçüklemezdır. Dinamik programlama yardımı ile Wong ve Larson tarafından basınç oranları hesaplanan bu problem, Goldberg tarafından GA ile çözülmeye çalışılmıştır (Goldberg, 1989).

Goldberg'in gaz boru hatlarının denetimi üzerine yaptığı doktora tezi ona salt 1985 National Science Foundation Genç Araştırmacı ödölünü kazandırmakla kalmamış, GA'ların uygulamada da kullanımının olabirirliğini kanıtladı.

Ayrıca 1989'da konusunda bir klasik sayılan kitabını yayımlayana dek GS'ların pek pratik yararı olmayan bir araştırma konusu olduđu düşünölüyordu. Halbuki yayımladığı kitabında GA'lara dayalı tam 83 uygulamaya yer vererek GA'nın dünyanın her yerinde çeşitli konularda kullanılmakta olduğunu göstermiştir.

Goldberg”in dünya bilim çevrelerine duyurduğu çalışmasını tâkiben 1992”de John Koza tarafından belirli işlerin bilgisayara yaptırılmasında GS”lar kullanıldı. Böylece kendisi tarafından “Genetik Programlama” olarak ifâde edilen yöntem ortaya çıkmıştır. Bu çalışmalarda LISP dili kullanılmıştır, çünkü LISP programları, özellikle yapay zekada kullanılan ve GA’ların üzerinde işlem yaptığı “karar ağaçları” “nı oluşturabilmekteydi.

4.2.Kuramsal Temeller

4.2.1. Genetik Algoritmanın Tanımı

GA, doğadaki evrim mekanizmasını örnek alan bir arama yöntemidir (Holland, 1975). GA, rassal arama tekniklerini kullanarak çözüm bulmaya çalışan, parametre kodlama esasına dayanan bir arama tekniğidir (Goldberg, 1989) ve bir veri grubu içinde özel bir veriyi bulmak için kullanılır.

GA’lar, “Yapay Zeka” “nın gittikçe genişleyen bir kolu olan evrimsel hesaplama tekniğinin bir parçasını oluşturmaktadır. Adından da anlaşıldığı üzere, evrimsel hesaplama tekniğinin bir parçası olan GA, Darwin”in evrim kuramından esinlenerek oluşturulmuştur. Herhangi bir problemin GA ile çözümü, problemi sanal olarak evrimden geçirmek sûretiyle yapılmaktadır (Hasşerbetçi, 1997).

Genel anlamıyla GA bir araştırma konusu olup model hâline getirilmiş neden-sonuç işleminin tersine rastgele örnekleme olgusu altında modellenmiştir. Kontrol edilerek onaylanan bilgi organizma olarak adlandırılan aday çözümler içerisinde saklanmıştır. Organizmalar popülasyon olarak adlandırılıp grup hâlinde yer almışlardır.

GA, çözüm grupları ile çalışarak, daha optimal, yapılabilir sonuçlara ulaşmada gerekli olan araştırmalara liderlik eder.

GA’lar doğada geçerli olan en iyinin yaşaması kuralına dayanarak sürekli iyileşen çözümler üretir. Bunun için “iyi” “nin ne olduğunu belirleyen bir *uygunluk* (fitness) fonksiyonu ve yeni çözümler üretmek için *yeniden kopyalama* (recombination), *değiştirme* (mutation) gibi operatörleri kullanır. GA’ların bir diğer önemli özelliği de bir grup çözümle uğraşmasıdır. Bu sâyede çok sayıda çözümün içinden iyileri seçilip kötülerini elenebilir (Kurt, 1998).

GA geleneksel yöntemlerle çözümü zor veya olanaksız olan problemlerin çözümünde kullanılmaktadır.

GA'lar klasik yöntemlerin çok uzun zamanda yapacakları işlemleri kısa bir zamanda çok net olmasa da yeterli bir doğrulukla yapabilirler.

4.2.2. Genetik Algoritmalarla İlgili Temel Kavramlar

Gen: Kalıtsal molekülde bulunan ve organizmanın karakterlerinin tayininde rol oynayan kalıtsal birimlere denir (Oraler, 1990). Yapay sistemlerde gen, kendi başına anlamlı bilgi taşıyan en küçük birim olarak alınır. Örneğin 101 bit dizisi bir noktanın x-koordinatının ikilik düzende kodlandığı bir gen olabilir.

Allel (Allele): Bir özelliği temsil eden bir genin alabileceği değişik değerlerdir.

Kromozom: Birden fazla genin bir araya gelerek oluşturduğu diziye denir. Kromozom temsil ettiği çözümle ilgili bilgi içermelidir. Her kromozom belli bir parametre kodlama sistemiyle oluşturulmuş bir diziden oluşur. Bu dizi içindeki bit adı verilen her bir sayı çözümün bir karakteristiğini temsil edebilir veya bir dizi bütünüyle bir sayıya işâret edebilir. Kromozomlar, alternatif aday çözümleri gösterirler (Engin, 2001).

Genom : Genetik materyal diyebileceğimiz kromozomların bütününe birden *genom* adı verilir. Genom içindeki genlerin belirli bir grubuna ise *genotip* denir. Genotipin doğumdan sonraki gelişmeler sonrasında çıkan somut fiziksel hâline ise *fenotip* adı verilir. Örneğin zihnî özellikler, zeka vb. (Özkaya, 2002).

Locus : Kromozom üzerindeki her bitin yerine verilen addır.

Popülasyon: Kromozomlardan oluşan topluluğa denir. Popülasyon, geçerli alternatif çözüm kümesidir. Popülasyondaki birey sayısı (kromozom) genelde sabit tutulur. GA'da popülasyondaki birey sayısı ile ilgili genel bir kural yoktur. Popülasyondaki kromozom sayısı arttıkça çözüme ulaşma süresi (iterasyon sayısı) azalır. Problemin büyüklüğüne bağlı olarak popülasyon büyüklüğü, (10 ile 100) arasında seçilebilir (Ghedjati, 1990).

Seçim (Selection): Temel olarak crossover ve mutasyona uğruyacak bireylerin popülasyon içinden seçilmesi farklı yöntemlerle yapılabilir. Bu yöntemlerden en temeli rulet tekerleği seçimidir. Seçim, algoritmanın çalışma hızını dolaylı olarak etkiler. Seçim yönteminin uygun olması, algoritmanın iyiye gidişini hızlandırır.

Elitizm: Üreme, çaprazlama ve mutasyon işlemleri sonrasında kuşakta bulunan en iyi uyumluluğa sahip birey, sonraki kuşağa aktarılamayabilir. Bunu önlemek için bu işlemlerden sonra oluşan yeni kuşağa, bir önceki kuşağın en iyi (elit) bireyi, yeni kuşaktaki herhangi bir birey ile değiştirilir. Buna *elitizm* adı verilir [1, 9, 10].

Evrimsel Algoritma (Evolutionary Algorithm): GA'ları da içine alan bir algoritmik yöntem.

Evrim (Evolution): Genetik bilgi taşıyan bir topluluk üzerinde genetik işlemlerin uygulanması süreci.

Evrilmek (Evolve): Bir evrim sürecinden geçmek.

1. Tersinme (Inversion) : Bir kromozomu oluşturan genlerden ardışık bir grubun kendi içerisinde birbirleri ile yer değiştirerek ters dizilmeleri.

Örneğin; 011110101 kromozomu (her genin bir konum olduğu varsayımı ile) 5. ve 8. gen konumları arasında tersindiğinde ortaya 011101011 kromozomu çıkıyor.

2. Eşleme (Mating): İki kromozomun çaprazlanma amacı ile seçilmesi.

3. Uygunluk Fonksiyonu (Fitness Function): Popülasyon içindeki bireylerin, diğer bir ifadeyle her biri bir çözüm grubunu ifade eden kromozomların performanslarını değerlendirmekte kullanılan, temel değerlendirme fonksiyonudur.

4. Çaprazlama (Crossover): İki kromozomun bir araya gelerek genetik bilgi değişimi yapması.

5. Örneğin; 100011101 ve 010110001 kromozomları üzerinde 4. locusdan başlayarak tek noktalı çaprazlama yapıldığında 100110001 ve 010011101 kromozomları elde edilir.

6. Mutasyon (Mutation) : Bir kromozomum taşıdığı genetik bilgide bir nedene bağlı olmaksızın (rastgele) değişme olması.

Örneğin; 100110000 kromozomunun 3. konumunun değişmeye uğraması sonucunda 101110000 kromozomu oluşur.

4.2.3. Genetik Algoritmanın Çalışma İlkesi

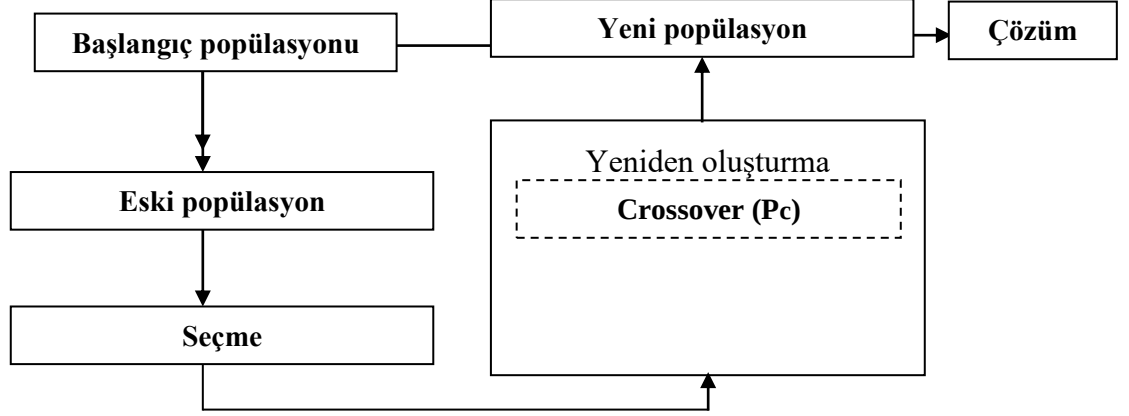
GA'nın işleyişi çok basittir ama bu kadar basit olan yöntemin, en zor problemleri nasıl çözdüğünün anlaşılması da o kadar zordur. Bu da GA'nın en karmaşık ve bilim adamlarının yıllardır çözmeye çalıştıkları en önemli sorulardan biridir.

GA'nın bu yönünü biraz açıklayalım; GA, çözüm(ler) bulmak için taranması gereken parametre uzayının çok büyük olduğu durumlarda bu arama işlemi için en akılcı yöntemdir. Evrimin her sürecinde edinilen bilgi, sonraki kuşaklara aktarılarak taramanın daha uygun bölgelerde gezmesi sağlandığı gibi değişim işlemi yardımıyla yerel çözüm noktalarına sıkışıp kalma olasılığı da azaltılmaktadır. Ayrıca GA'nın paralel işlem yapılan bilgisayarlarda kullanılmaya elverişli yapısı da, zaman alıcı problemlerin çözümü için çekici bir seçenek olmasını sağlamaktadır.

GA'nın çalışmasını, Tablo 4.1."deki gibi özetleyebilir ve bunun şematik gösterimini Şekil 4.2."deki gibi çizebiliriz.

Tablo 4.1. Genetik algoritmanın işleyiş adımları

Adım 1	Olası çözümlerin kodlandığı bir çözüm grubu oluşturulur (çözüm grubu, biyolojideki benzerliği nedeniyle toplum (population), çözümlerin kodları (string) da kromozom olarak adlandırılır).
Adım 2	Her kromozomun ne kadar iyi olduğu bulunur (fitness function).
Adım 3	Bu kromozomlar eşlenerek (mating), yeniden kopyalama (recombination) ve değiştirme (crossover) operatörleri uygulanır. Bu sayede yeni bir toplum oluşturulur.
Adım 4	Yeni kromozomlara yer açmak için eski kromozomlar ortadan kaldırılır.
Adım 5	Tüm kromozomların uygunlukları yeniden hesaplanır.
Adım 6	Eğer jenerasyon süresi dolmamışsa 3. adıma gidilir.
Adım 7	O ana kadar bulunmuş en iyi kromozom, sonuçtur.



Şekil 4.1. Genetik algoritma temel işleyişinin şematik gösterimi

İşlemleri adım adım açıklayalım:

Adım-1: Bu adıma toplumda bulunacak birey sayısını belirleyerek başlanmaktadır. Kullanılacak sayı için bir standart yoktur. Genel olarak önerilen 100-300 aralığında bir büyüklüktür. Büyüklük seçiminde yapılan işlemlerin karmaşıklığı ve aramanın derinliği önemlidir. İlk popülasyonun uygun oluşturulması, ortalama uygunluk değerini yükseltir ve hesaplama zamanını (kuşak sayısını) düşürür. Bu nedenlerden

dolayı farklı prosedürlerle ilk popülasyonu oluşturmak, popülasyon ortalama uygunluk değerini yükseltecek ve çözüme ulaşma süresini kısaltacaktır (Chen ve diğ., 1996).

Başlangıç popülasyonu, sonuçlardaki düzgün (uniform) yapının korunması için genellikle rastgele oluşturulur. Arama uzayındaki olası çözümler, dizi olarak kodlanır. Bu diziyi (kromozomu) oluşturan her bir elemana *gen* denir. Her bir dizi, arama uzayında belirli bir bölgeye karşılık gelir.

Adım-2: Kromozomların ne kadar iyi olduğunu bulan fonksiyona *uygunluk fonksiyonu* denir. Bu fonksiyon işletilerek, kromozomların uygunluklarının bulunmasına ise *hesaplama* (evaluation) adı verilir. Bu fonksiyon GA'nın beynini oluşturmaktadır. GA'da probleme özel çalışan tek kısım bu fonksiyondur. Uygunluk fonksiyonu, kromozomları problemin parametreleri hâline getirerek onların bir bakıma şifresini *çözmektedir* (decoding). Daha sonra bu parametrelere göre hesaplamayı yaparak kromozomların uygunluğunu bulur. Çoğu zaman GA'nın başarısı, bu fonksiyonun verimli ve hassas olmasına bağlı olmaktadır. Bulunan uygunluk değerleri, dizilerin çözüm kalitesini gösterir.

Adım-3: Kromozomların eşlenmesi kromozomların uygunluk değerlerine göre yapılır. Bu seçimi yapmak için *rulet tekerleği seçimi* (roulette wheel selection), *turnuva seçimi* (Tournament Selection) gibi seçme yöntemleri vardır. Örnek olarak bu çalışmada kullanılan rulet tekerleği seçimi aşağıda açıklanmıştır.

- 1- Tüm bireylerin uygunluk değerleri bir tabloya yazılır.
- 2- Bu değerler toplanır.
- 3- Tüm bireylerin uygunluk değerleri toplama bölünerek [0,1] aralığında sayılar elde edilir. Bu sayılar bireylerin seçilme olasılıklarıdır. Sayıların hepsi bir tabloda tutulur.
- 4- Seçilme olasılıklarını tuttuğumuz tablodaki sayılar birbirine eklenerek rastgele bir sayıya kadar ilerlenir. Bu sayıya ulaşıldığında ya da geçildiğinde, son eklenen sayının ait olduğu çözüm seçilmiş olur.

Bu yönteme “rulet tekerleği seçimi” adı, bir daireyi, çözümlerin uygunluklarına göre dilimleyip çevirdiğimizde olacakların benzeşimi olduğu için verilmiştir.

Rulet tekerleği seçimi, çözümlerin uygunluk değerlerinin negatif olmamasını gerektirir. Çünkü olasılıklar negatif olursa bu çözümlerin seçilme şansı yoktur. Çoğunluğunun uygunluk değeri negatif olan bir toplumda yeni kuşaklar belli noktalara takılıp kalabilir.

Gen takası (crossover), GA'nın motoru kabul edilir. Basitçe olay iki ebeveyn kromozomun arasında belirlenen parçaların takasıdır. GA'lar, bu olayın benzeşimini temelde en çok kullanılan iki yöntemle gerçekleştirirler:

- Tek noktalı (Single (one) point crossover) gen takası
- Çok noktalı (Multi point crossover) gen takası

GA'larda *ikilik dizi* (binary string) çok kullanılır. Doğadaki genlerin benzeşimi *bit*'lerdir. İkilik dizilerde bir gen takası Şekil 4.2.'deki gibi gerçekleşebilir.

	Önce		Sonra
Birey A	010	111	010001
Birey B	101	001	101111

↓ Crossover noktası

Şekil 4.2. Gen takası (takas için seçilen nokta kesikli çizgilerle gösterilmiştir.)

Gen takası toplumda çeşitliliği sağlar. İyi özelliklerin bir araya gelmesini kolaylaştırarak en iyiye yaklaşmayı sağlar. Mutasyon, kromozomun bir parçasının dışarıdan değiştirilmesi şeklinde tanımlanır. Bu değiştirme, görünüşte GA'nın dayanak noktasıdır, ancak etkisi bir çözüm üzerindedir. Bu da yalnız başına başarılı olmasını zorlaştırır.

İkilik dizilerde değiştirme, rastgele bir bit'in değiştirilmesiyle sağlanabilir. Çok düşük bir değiştirme olasılığı toplumda bazı özelliklerin kaybolmasına neden olabilir. Bu da en iyi sonuçların bulunmasına engeldir. Ancak yüksek bir değiştirme olasılığı da, eldeki çözümleri bozarak sonuca ulaşmayı zorlaştırır. Gen takası ve değiştirmenin olasılıkları için kesin bir sayı yoktur. Değiştirme (mutasyon) olasılığı 0,01-0,001, gen takası (cross-over) olasılığı ise 0,5-1,0 aralığında tavsiye edilir ama bu oranlar, uygulama alanına göre değişken olabilir.

Adım-4. Eski kromozomlar çıkartılarak sabit büyüklükte bir toplum sağlanır.

Adım-5. Tüm kromozomlar yeniden hesaplanarak yeni toplumun başarısı bulunur.

Adım-6. GA defalarca çalıştırılarak çok sayıda toplum oluşturulup hesaplanır.

Adım-7. Toplumların hesaplanması sırasında en iyi bireyler saklandığı için, o ana kadar bulunmuş en iyi çözüm çözümdür.

4.2.4. Genetik Algoritmada Kullanılan Operatörler

4.2.4.1. Giriş

GA’larda dört ayrı operatör kullanılmaktadır:

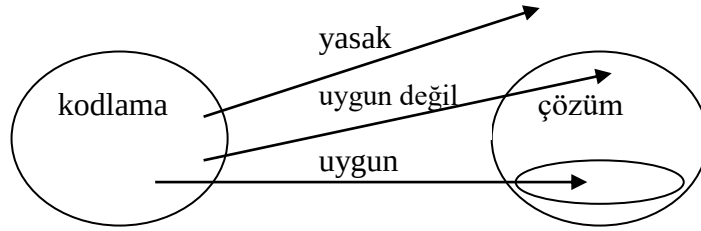
- 1) Parametre kodlama operatörü
- 2) Üreme operatörü
- 3) Çaprazlama operatörü
- 4) Mutasyon operatörü

4.2.4.2. Parametre kodlama operatörü

GA’ların temel özelliği, çözüm uzayı ve kod uzayında çalışmasıdır. Çözüm uzayından kod uzayına geçişi, kromozom yapıları sağlar. Kromozomlar, alternatif aday çözümleri gösterirler. Kromozom temsil ettiği çözümle ilgili bilgi içermelidir. Her kromozom belli bir parametre kodlama sistemiyle oluşturulmuş bir diziden oluşur. Bu dizi içindeki bit adı verilen her bir sayı, çözümün bir karakteristiğini temsil edebilir veya bir dizi bütünüyle bir sayıya işaret edebilir. Problemlerin başarılı bir şekilde çözümü için en kısa kodlama şekillerinin kullanılması gerekir (Goldberg, 1989).

Son on yıl boyunca değişik kodlama teknikleri geliştirilmiştir. Çözümün kromozomlara nasıl kodlanacağı GA’da anahtar niteliğinde bir çalışmadır. Kromozomların kodlanması ve şifre çözümü aşamasında, üç kritik noktaya dikkat edilmesi gerekir (Cheng ve diğ., 1996).

1. Kromozomun uygunluğu (feasibility) kontrol edilir.
2. Kromozomun belirlenen koşulları (kısıtları) sağlayıp sağlamadığı kontrol edilir.
3. Kodlama haritasında, kromozomun tek olduğu kontrol edilir. Kromozomun uygunluğu ile sağlayıp sağlamadığı işlemleri Şekil 4.3.”de sunulmuştur (Cheng ve diğ., 1996).



Şekil 4.3. Kodlamada uygun olma ve yasaklı olma durumları

Koşulları sağlamayan (illegal) bir kromozom, çözüm uzayındaki değerleri şifreleyemez. Böyle bir kromozom, evrime uğrayamaz.

İki Tabanlı (Binary) Kodlama Sistemi: En yaygın kullanımı olan kodlama sistemidir. John Holland’ın çalışmalarında genelde iki tabanlı kodlama sistemi kullanılmıştır. Kromozomu ikili düzendeki sayılar dizisiyle ifade etmek, çok tercih edilen bir temsil şeklidir.

Bu kodlama yapısında, dizi içindeki her gen 0 ve 1’li sayı değerlerinden birini almaktadır (Şekil 4.4.).

Kromozom A	101100101100101011100101
Kromozom B	111111100000110000011111

Şekil 4.4. 24 bitlik kromozom yapılarının iki tabanlı kodlanması

Bu kodlama sistemi yalnızca 0 ve 1’in kullanılmasına rağmen, birçok olası farklı kromozom yapısı üretebilir. Ama birçok problem için kullanılabilecek yeter düzeyde değildir, bazen crossover veya mutasyon aşamalarından sonra düzeltme yapmak gerekebilir. İkili düzenin yeğlenmesinin nedeni, basit olması ve bilgisayar tarafından daha kolay ve hızlı bir biçimde işlenebilmesidir

Permütasyon Kodlama Sistemi: Bu kodlama sistemi, iş sıralama ve gezgin satıcı problemi benzeri sıralama problemlerinin kodlanmasında kullanılır. Her kromozom belli bir sıradaki değeri tanımlayan sayı dizilerinden oluşur (Şekil 4.5.).

Kromozom A	1 5 3 2 6 4 7 9 8
Kromozom B	8 5 6 7 2 3 1 4 9

Şekil 4.5. Permütasyon kodlama örneği

Örneğin; gezgin satıcı probleminde kromozomdaki sayısal değerler, sırayla gidilmesi gereken kent numaralarını belirtmektedir.

Değer Tabanlı Kodlama Sistemi: Doğrudan değer kodlaması, karmaşık değer ve rakamların yer aldığı problemlerde kullanılabilir. Bâzı problemler, yapıları gereği iki tabanlı kodlama sisteminde kullanılamazlar. Bu durumda, ilgili sayısal ya da alfasayısal değerlerin kromozomlarda kullanılması uygun olmaktadır.

Her kromozom, belli bir dizi değerleri şeklinde ifâde edilir. Bu değerler probleme bağlı olarak sayılar, harfler, karmaşık başka şekiller olarak Şekil 4.6.”daki örnekte görüldüğü gibi değişkenlik göstermektedir.

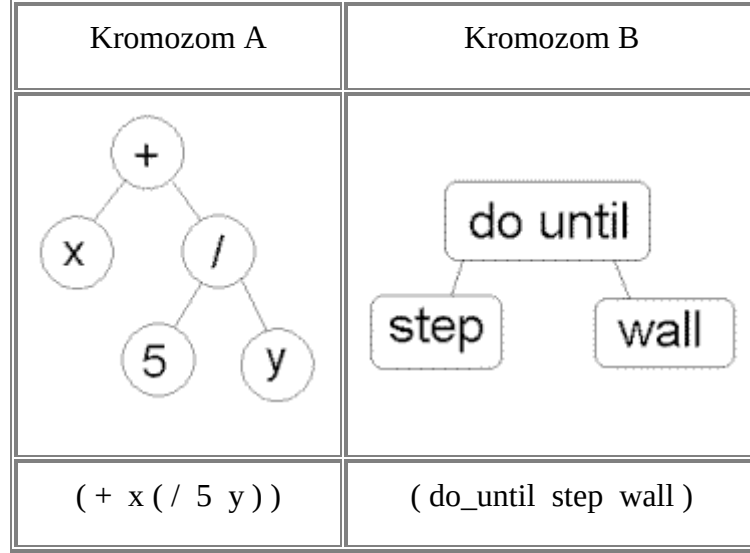
Kromozom A	1.2324 5.3243 0.4556 2.3293 2.4545
Kromozom B	ABDJEIFJDHDIERJFDLDFLFEGT
Kromozom C	(arka), (arka), (sağ), (ön), (sol), (sağ)

Şekil 4.6. Değer tabanlı kodlama örneği

Değer kodlama sistemi, bâzı özel problemler için oldukça iyi sonuçlar vermektedir. Yalnız bu parametre kodlamaya uygun yeni crossover ve mutasyon yöntemleri geliştirilmelidir. Örnek olarak yapay ağlarda ağırlıkların bulunmasında bu sistem kullanılabilir. Mimarîsi verilen bir yapay ağın istenilen doğrultuda eğitilmesi için gerekli nöronlara ilişkin ağırlık değerlerinin bulunmasına yönelik her gen bir ağırlık değerini temsil edecek şekilde kodlanır.

Ağaç Yapısı Kodlama Sistemi: Bu sistem esas olarak genetik programlamada programların ve ifâdelerin değerlendirilmesinde kullanılır.

Ağaç yapısı kodlama sisteminde her kromozom bâzı nesnelerin ağacıdır. Bunu, programlama dilindeki komutlar ve fonksiyonlar gibi düşünebiliriz (Şekil 4.7.).



Şekil 4.7. Ağaç yapısı kodlama örneği

Programların değerlendirilmesinde kullanılır, örneğin LISP programlama dili bu yapıya sıklıkla başvurur. Input ve output değerlerinden ilgili fonksiyonun bulunması probleminde bu kodlama sistemi kullanılır.

4.2.4.3. Üreme (Seçim) operatörü

Bu prosedürde, bireysel diziler, amaç fonksiyonuna göre kopyalanır, gelecek kuşakta daha iyi döl verebilecek bireyler seçilir. Üreme operatörü, yapay bir seleksiyondur (Goldberg, 1989).

Burada bir kromozom havuzundaki kromozomlardan hangilerinin yeni yaratılacak havuza aktarılacağını, kromozomların başarı değerlerine bağlı olarak belirlenmesi esastır.

Yeni popülasyondaki ebeveynleri oluşturmak üzere bâzı bireylerin seçilmesi gerekir. Kurama göre iyi olan bireyler yaşamını sürdürmeli ve bu bireylerden yeni bireyler oluşmalıdır. Bu seçim çeşitli ölçütlere göre yapılabilir. Bu yöntemlerden bâzıları şunlardır:

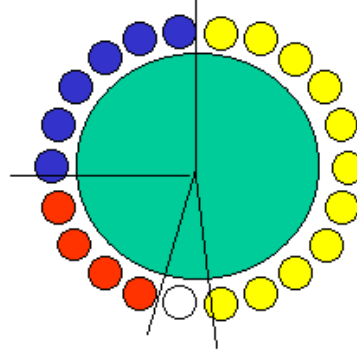
- Rulet Seçimi
- Boltzman Seçimi
- Turnuva Seçimi
- Sıralı Seçim

Bunların yanısıra problem içeriği ve yapısına göre daha karmaşık, algoritmanın işleyişi sırasında seçim parametrelerini değiştiren seçim yöntemleri de vardır. Temel olarak bu yöntemler benzetim-tavlama yöntemi gibi çalışır. Elitizm denilen seçicilik yöntemi de her kuşaktaki belli sayıdaki en iyi sonucun diğer popülasyonlarda varlığını sürdürmesi demek olup isteğe göre kullanılabilir. Örnek olarak bu çalışmada kullanılan rulet tekerleği seçimi aşağıda açıklanmıştır:

1. Tüm bireylerin uygunluk değerleri bir tabloya yazılır.
2. Bu değerler toplanır.
3. Tüm bireylerin uygunluk değerleri toplama bölünerek $[0,1]$ aralığında sayılar elde edilir. Bu sayılar bireylerin seçilme olasılıklarıdır. Sayıların hepsi bir tabloda tutulur.
4. Seçilme olasılıklarını tuttuğumuz tablodaki sayılar birbirine eklenerek rastgele bir sayıya kadar ilerlenir. Bu sayıya ulaşıldığında ya da geçildiğinde, son eklenen sayının ait olduğu çözüm seçilmiş olur.

Rulet seçiminde kromozomlar uyumluluk fonksiyonuna göre bir rulet çevresine gruplanır. Uyumluluk fonksiyonu, herhangi bir ölçüte uyan bireylerin seçilmesi için kullanılır. Bu rulet üzerinden rastgele bir birey seçilir.

Bu yönteme rulet tekerleği seçimi adı, bir daireyi, çözümlerin uygunluklarına göre dilimleyip çevirdiğimizde olacakların benzeşimi olduğu için verilmiştir (Şekil 4.8.).



Şekil 4.8. Rulet tekerleği seçimi

Görüldüğü üzere daha büyük alana sahip bireyin seçilme şansı daha fazla olacaktır.

Rulet tekerleği seçimi, eğer uyumluluk çok fazla değişiyorsa sorun çıkartabilir. Örneğin en iyi kromozomun uyumluluğu % 90 ise, diğer kromozomların seçilme şansı azalacaktır.

Bunu önlemek için sıralı seçim kullanılabilir. Sıralı seçimde en kötü uyumlulukta olan kromozoma 1 değeri, sonrakine 2 değeri verilir ve böylelikle seçilmede bunlara öncelik tanınmış olur. Bu şekilde onların da seçilme şansı artar ama bu, çözümün daha geç yakınsamasına neden olabilir.

Üreme yönteminde genellikle rulet çemberi yöntemi kullanılır. Varolan popülasyondaki en uygun kromozomlar rulet çemberi üzerine yerleştirilir, çember popülasyondaki dizi sayısı kadar döndürülerek yeni kuşak elde edilir.

4.2.4.4. Crossover (Çaprazlama) operatörü

GA'nın performansını etkileyen parametrelerden biri de çaprazlama işlemidir. Doğal popülasyonlarda çaprazlama, organizmalar arasında yapılan döllenme (hayvanlarda çiftleşme, bitkilerde tozlaşma) çalışmaları olarak bilinir (Oraler, 1990).

“Crossover” olarak adlandırılan bu operatör iki çözümün yapı taşları kullanılarak yeni bir çözüm oluşturulması temeline dayanır. Bu işlem doğada görülen “crossing over” olayının analogudur. “Crossover” işlemi genel olarak ikili dizilerin parçalarının değiş-tokuşu şeklinde gerçekleştirilir. Çaprazlama, ebeveynlerden bazı genleri alarak yeni bireyler oluşturma işlemidir. Farklı uygulamalarda, farklı kodlama yöntemlerinden yararlanıldığı için farklı “crossover” yöntemleri kullanılır.

Tek Noktalı Çaprazlama: Rulet çemberi yöntemi ile yapay seçim sonucunda elde edilen yeni popülasyon dizisinde rassal olarak iki kromozom seçilir ve karşılıklı çaprazlama işlemine tâbi tutulur. Çaprazlama işleminde, L dizi uzunluğu olmak üzere,

$\{1 \leq k \leq L-1\}$ aralığında düzgün (uniform) olarak k tamsayısı seçilir. Bu tamsayı değerine göre dizi, çaprazlamaya tâbi tutulur.

Örneğin; başlangıç popülasyonunda A_1 ve A_2 gibi iki dizi rassal olarak seçilmiş olsun.

$$A_1 = 0 \ 1 \ 1 \ 0 \ 1$$

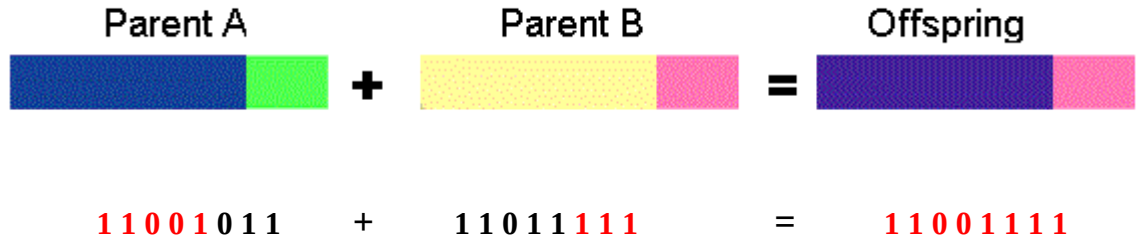
$$A_2 = 1 \ 1 \ 0 \ 0 \ 0$$

Bu örnekte dizi uzunluğu, (L=) 5'dir. 1 ile 5 arasında (k=) 4 seçildiğini varsayarsak, her iki dizinin de kesim noktası olarak kabul edilen 4. genden sonra karşılıklı çaprazlama işlemine tâbi tutulacağı kabul edilir. Bu durumda yeni kromozomlar aşağıdaki gibi olur:

$$A_1^1 = 0 \ 1 \ 1 \ 0 \ 0$$

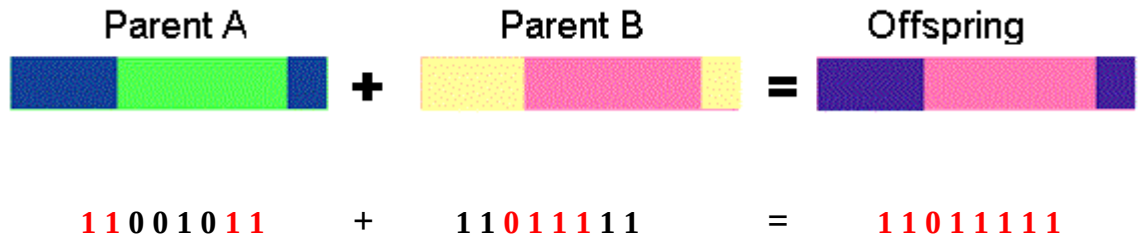
$$A_2^1 = 1 \ 1 \ 0 \ 0 \ 1$$

En basit çaprazlama yöntemi, Şekil 4.9.'da verildiği gibi bir noktalı çaprazlama yöntemidir. Bir noktalı çaprazlama yapılabilmesi için her iki kromozomun da aynı gen uzunluğunda olması gerekir.



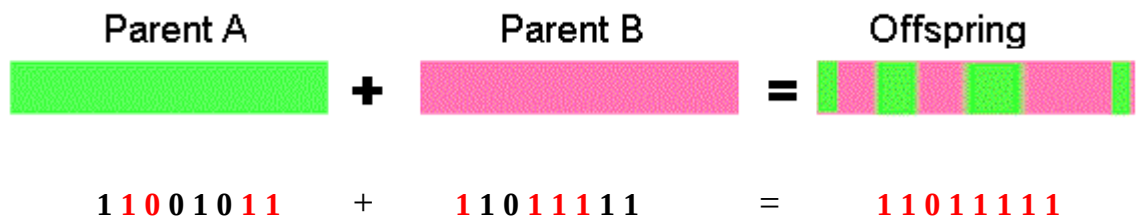
Şekil 3.2. Tek noktalı çaprazlama

İki Noktalı Çaprazlama: İki noktalı çaprazlamada ise kromozom iki noktadan kesilir ve karşılıklı olarak pozisyonlar yer değiştirilir (Şekil 4.10.) (Croce ve diğ., 1995).



Şekil 4.10. İki noktalı çaprazlama

Düzgün (Uniform) Çaprazlama: Kromozom üzerindeki bit değerleri rassal olarak belirli noktalardan bir diğer ebeveyn üzerine kopyalanır (Şekil 4.11.).



Şekil 4.11. Düzgün (Uniform) çaprazlama

4.2.4.5. Mutasyon Operatörü

GA’da önemli rol oynayan proseslerden biri de mutasyon operatörüdür. Canlılarda gen rekombinasyonlarının dışındaki diğer nedenlerle ve âni olarak oluşan kalıtsal değişimlere *mutasyon* denir. Doğal popülasyonlarda mutasyon işlemi ile aşağıdaki değişimler gerçekleşir (Kuru, 1987):

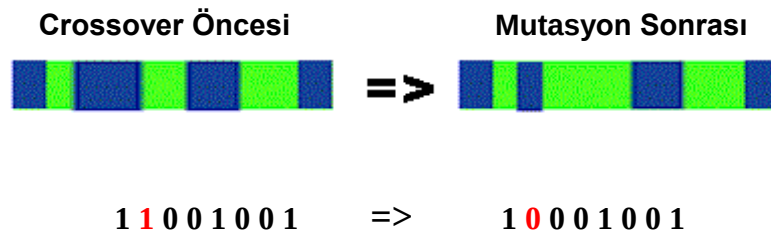
1. Kromozom yapısındaki deęişmeler
2. Kromozom sayısındaki deęişmeler
3. Gen yapısındaki fiziksel ve kimyasal deęişmeler

Mutasyon, aprazlama operatörü gerekleştikten sonra gerekleştirilir. Mutasyon, oluşan yeni özümün önceki özümü kopyalamasını önlemek ve sonuca daha hızlı ulaşmak amacıyla yapılır. Mutasyon, oluşan yeni bireyin bir bitini (eđer ikili düzende ifade edilmiş ise) rastgele deęiştirir.

Yapay sistemlerde mutasyon işlemi sırasında, kromozomdaki gen sayısı deęişmez, sâbit kalır. Doğal popölasyonlarda mutasyon oranı oldukça düşüktür. Mutasyon frekansının büyüklüğü GA'nın performansını etkilemektedir.

Örneğin, kromozom uzunluğu 20 gen olan bir dizide, mutasyon oranı yüzde 0,1 seçildiğinde, kromozom üzerinde mutasyondan dolayı beklenen deęişim olasılığı, $(20 \cdot 0,001 =) 0,02$ olarak bulunur. Bu durumda yenisi oluşuncaya kadar mutasyon işlemi gerekleştirilemez (Goldberg, 1989).

Mutasyon işlemi bir tek kromozom üzerinde yapılır. Mutasyon frekansına göre, mutasyona uğratılacak sayıdaki diziler, popölasyondan rassal olarak seçilir ve belirlenen mutasyon yöntemine göre deęişime uğratılır. Bu deęişim işlemi ok deęişik şekillerde yapılabilir. Problemin yapısı bu aşamada ok önemlidir. Örneğin sıralama problemlerinde sıralamayı deęiştirmek, ikilik dizi gösteriminde bitleri deęiştirmek, özüm ağalarında paraları deęiştirmek, işlem olarak tanımlanır (Şekil 4.12.).



Şekil 4.12. İkili deęerlerin (binary) mutasyonu

4.3. Genetik Algoritma Parametreleri

4.3.1. Genel Bilgi

GA tekniğinin popülasyon büyüklüğü, çaprazlama (crossover) olasılığı ve mutasyon olasılığı olmak üzere üç temel parametresi vardır. Üreme, çaprazlama ve mutasyon yöntemleri de bazı yazarlarca parametre bağlamında ele alınmakla birlikte, algoritmanın işleyişine ve performansına etkisi, diğerleri kadar baskın değildir.

4.3.2. Popülasyon Büyüklüğü

Bu parametre, popülasyon içinde (yalnızca bir kuşakta) kaç adet kromozom yâni birey olduğunu söyler. Eğer kromozom sayısı az olursa, GA, çözüm aranan uzayın ancak bir kısmını gezebilir ve çaprazlama için fazla bir seçeneği yoktur. Kromozom sayısı çok fazla olursa GA çok yavaş çalışır. Araştırmalar belli bir noktadan sonra popülasyon sayısını artırmanın bir yararı olmadığını göstermiştir.

Popülasyon büyüklüğünün çok yüksek bir değer seçilmesi, algoritmanın üzerinde olumsuz bir etki yaratmakta olup performansı, sonucun kısa zaman aralığında bulunması bağlamında düşürmektedir. Olması gereken popülasyon büyüklüğü değeri genel olarak 20-100 arası verilse de bu değer daha sonraları 100-300 arasına kadar genişletilmiştir. Çünkü araştırmalar, bu değer, kromozomların parametre kodlamasından etkilendiğini göstermektedir (Engin, 2002).

Yeni bireyler uyumluluğa göre veya rastgele olarak seçilebilir. Yeni bireylerin tümüyle rastgele seçilme durumunda yakınsama zorlaşabilir. Tüm bireyler uyumluluğa göre seçildiğinde ise yeni kuşak içinde bölgesel yakınsamalar olabilir. Bu sorunların üstesinden gelmek için belli bir oranda uyumluluk seçimi, belli bir oranda da rastgele seçim yapılabilir. Bu oran, Kuşak Farkı (Generation Gap) ile ifade edilir. Kuşak farkı % 100 olduğunda, yeni bireylerin tümü uyumluluğa göre seçilir.

4.3.3. Crossover (Çaprazlama) Olasılığı

Crossover olasılığı (Pc), olabildiğince yeni popülasyonların, oluşması olası daha iyi çözümlerin yaşam bulması için yüksek olmalıdır. Crossover oranı, çaprazlama operatörünün hangi olasılıkla kullanılacağını kontrol eder. Eğer crossover olmazsa, oğul bireyler ana bireylerin benzer kopyası olur. Eğer crossover gerçekleşirse, oğul

bireyler ana bireylerin kromozom yapısından belirli bir parçasını taşır, crossover oranı % 100 olduğunda, tüm oğul bireyler ana bireylerden üretilir, bu oranın % 0 olması durumunda ise yeni kuşak, eski popülasyonun bire-bir kopyası olur.

Crossover olasılığı % 80-95 arasında olmalıdır. Problemin yapısına göre farklılık göstermekle birlikte bazı araştırmacılar tarafından % 60 olasılığı optimum olarak düşünülmektedir. Çaprazlama, eski bireylerden iyi yanlar alınarak elde edilen yeni bireylerin daha iyi olması umuduyla yapılır. Bununla birlikte eski kuşaktan yeniye geçişte, bazı uygunluk değeri yüksek seçkin bireylerin, oluşacak yeni popülasyonda da yer alması da istenebilir.

4.3.4. Mutasyon Olasılığı

Mutasyon oranı (P_m), kromozomun ilgili hangi parçalarının popülasyon içinde % kaç sıklıkla mutasyona uğruyacağını ifade eder. Eğer mutasyon olmazsa crossover sonucu elde edilen bireyin tümü, hiç değişmeden yeni kuşağı oluşturur. Mutasyon gerçekleşirse kromozomun bir bölümü yapısal değişikliğe uğrar.

Eğer mutasyon oranı % 100 olursa, kromozomun tümü değişir, % 0 olması durumunda ise hiç bir değişiklik olmaz. Mutasyon oranı crossover oranının tersine, olabildiğince düşük olmalıdır. Bu oranın % 0,5-1 aralığında olması önerilmektedir.

Mutasyon, GA'nın yerel ekstrem noktalara takılmasını önler, eğer mutasyon oranı çok yüksek olursa algoritma rassal arama işlemine dönüşür.

4.4. Genetik Algoritmalarla Parametre Optimizasyonu

4.4.1. Genel Bilgi

GA'lar, çözüm uzayında arama yaparken birtakım parametreleri kullanırlar. Bunlar; üreme, çaprazlama ve mutasyon yöntemleri ile çaprazlama-mutasyon oranları ve başlangıç popülasyonundan oluşur. Kısa sürede çözüme ulaşmak için bu parametrelerin seçimi ile ilgili herhangi bir kural yoktur (Cicirello ve Smith, 2000).

Herhangi bir problemin çözümünde kullanılan GA için optimum veya optimuma yakın çözüm veren kontrol parametresi kümesi, başka bir GA uygulaması için

genelleştirilemez. GA'nın uygulandığı problemlerde bu parametreler, deneme-yanılma yöntemi ile seçilir (Cicirello ve Smith, 2000).

Minumum beş farklı başlangıç popülasyonunun, beş üreme yönteminin, altı çaprazlama ve beş mutasyon yönteminin kullanıldığı; (0-100) aralığında çaprazlama ve mutasyon oranının seçildiği basit bir GA probleminde en az 75.000 seçenek arasında deneme-yanılma yöntemi ile uygun parametrelerin seçiminin oldukça zor olduğu görülmektedir.

4.4.2. Başlangıç Popülasyonunun Optimizasyonu

Başlangıç popülasyonu, başlangıçta istenen optimum çözümden uzaklığı gösterir. Dolayısıyla başlangıçta optimuma yakın olan çözüm kümeleri kullanılmalıdır. İlk popülasyonun uygun oluşturulması, popülasyonun ortalama uygunluk değerini yükseltecek ve çözüme ulaşma süresini kısaltacaktır. Bunun için geliştirilen belirli yöntemler olmakla birlikte, GA uygulayıcısı problemin yapısına göre sezgileriyle başlangıç popülasyonunu en iyi değerlerden seçmelidir.

4.4.3. Seçim Operatörü Optimizasyonu

Seçim operatörü algoritmanın çalışma hızını etkiler. Ancak bu doğrudan olmaz. Seçim sırasında en iyilerin seçimi sağlanarak optimuma gidiş süreci hızlandırılmalıdır. Eğer yeniden yerleştirmede (reinsertion) kullanılan teknik, her popülasyonun yenilenmesi (pure insertion) şeklinde ise, seçimdeki performans eldeki sonuçların yitirilmemesi açısından önemlidir. Problemin yapısı gereği seçim yöntemi en iyi şekilde belirlenmelidir. Özellikle seçim aşamasında ortaya çıkan iyi bilgi kaybının önüne geçilmeye çalışılmalıdır.

4.4.4. Çaprazlama Operatörü Optimizasyonu

Çaprazlama operatörü neredeyse tümüyle kodlama operatörüne bağlıdır. Mutasyon için seçilen yöntem de gözönünde tutularak, çaprazlama oranının ve çaprazlama yönteminin ne olacağı kararlaştırılmalıdır. Çaprazlama yöntemlerinin anlatıldığı bölümde de değinildiği gibi bu oranın % 85-100 arası değerlerde olması tavsiye edilir. Birkaç uygulamada bu değerden daha düşük olan %60 oranının da iyi sonuç verdiği gözlenmiştir. Bu parametrenin optimizasyonu, geliştiricinin gözlem yeteneğine ve sezgilerine bağlıdır.

4.4.5. Mutasyon Operatörü Optimizasyonu

Mutasyon operatörü, GA'ların çeşitli nedenlerle çözüm uzayının belli bölgelerinde gerekli-gereksiz döngülere girmemesini, eğer girmişse de bu döngülerden kurtulmasını sağlar. Mutasyon operatörü, salt sorun çıkmasına karşı bir önlem değil, aynı zamanda algoritmanın daha iyi çözüm noktalarına daha hızlı ulaşmasını sağlar niteliktedir. Kodlama tekniği ve çaprazlamada tercih edilen yöntem gözönüne alınarak, mutasyonun hangi yöntemle ve ne oranda yapılacağı kararlaştırılır.

Mutasyon oranı için, test edilmiş ve çoğu çalışma için tavsiye edilen iki farklı oran açıklanmıştır. Eğer bit düzeyinde çalışma yapılıyorsa, bireylere ilişkin dizilerin her bir değeri için ayrı mutasyon olasılığı gözetilir. Bu durumda tavsiye edilen oran 0,001'dir. Bireylerin her bir biti, 0,001'lik olasılık gözetilerek mutasyona uğratılır. Eğer bireylerin her biri için mutasyona uğrayıp uğramayacağı hesaplanacaksa bu durumda, n birey sayısı olmak üzere $1/n$ kabul edilebilir bir orandır.

4.4.6. Popülasyon Büyüklüğünün Optimizasyonu

Popülasyon nüfusu, genelde 20, 50 veya 100 olarak alınır. Algoritma bilgisayarda yapıldığı için, bu sayıların genelde çok küçük olduğu düşünülebilir. Ancak çoğu kez ideal değerlerdir. Gereğinden fazla popülasyon nüfusu iyi sonuç vermemektedir. Bu nedenle ideal bir değerin, geliştiricinin sezgileri ve gözlemleriyle bulunması gerekir.

4.5. Genetik Algoritmaların Kullanılma Nedenleri

4.5.1. Genel Bilgi

GA, açıkça tanımlanamayan problemlerin çözümünde tercih edilen bir yöntemdir. Geleneksel yöntemlerle çözülemeyen zor problemlere örnek, NP-Hard adı verilen problemlerdir. Bu ve bunun gibi problemlerde polinom benzeri bir fonksiyonla problem çözülmek istenirse aşırı işlem yükü ile karşılaşılır. Hattâ bâzı problemlerde bu biçimde fonksiyon oluşturularak çözüme gitmek olanaklı olmaz.

Örnek olarak Gezgin Satıcı Problemi (Travelling Salesman) NP problemleri kategorisine girmektedir. Burada N adet kente gitmek zorunda olan satıcının kentler arası uzaklıkları dikkate alarak, toplam alınan yolu enküçükleyecek şekilde kentlere hangi sırayla gitmesi gerektiği araştırılmaktadır. (N=) 30 farklı kent olduğu

düşünüldüğünde, bu problem için ($N!=30!=$) $2,65*10^{32}$ farklı seçenek sözkonusudur. Bilgisayarın, her saniyede, olası 1 milyar sonucu analiz ettiğini düşündüğümüzde bile, toplam seçeneklerin değerlendirilmesi 8 trilyar yıl sürmektedir. Görüldüğü üzere problemin optimum sonucunun bulunması bu koşullar altında olanaksızdır.

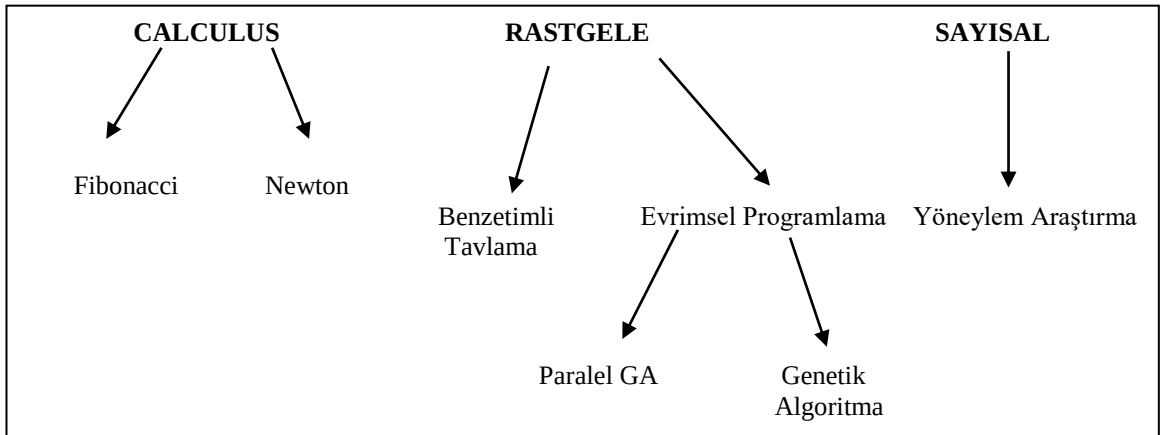
Bu ve benzeri problemlerin açıkça tanımlanarak modellenmesi olanaklı değildir ya da olası tüm çözümlerin bilgisayara çözdürülmesi çok uzun bir süre gerektirmektedir. İşte böyle durumlarda seçenek yöntemlere bakılır. GA'lar da bu seçeneklerden biri olup, gerçekten mükemmel denilebilecek pek çok çözümün çok hızlı bir şekilde elde edilmesini sağlamaktadırlar.

GA'lar, pek çok sayıda rastgele çözümün kontrollu ve etkin bir şekilde üretilmesi temeline dayanarak çalışmaktadır.

Klasik denklem *eniyilenmesinde* üç tip çözümünden sözedilir:

1. *Türev-İntegral hesabına* (calculus) dayananlar
2. *Numaralamaya* (enumeration) dayananlar
3. *Rastgele aramalar* (random searches)

GA'ların optimizasyon yöntemleri arasındaki yeri, Şekil 4.13.'de görülebilir. Türev-İntegral hesaplamalarına dayanan hesaplama yöntemleri ile çok derinlemesine çalışılmıştır. Bu yöntemler, fonksiyonun türevinin köklerinin, fonksiyonun en küçük ve en büyük değer veren noktaları olmasından yararlanır. Gerçek problemler için sıfır veren noktaları bulmak da ayrı bir problemidir.



Şekil 4. Genetik Algoritmanın klasik optimizasyon arama yöntemleri arasındaki yeri

Diğer bir yöntem ise, alınan bir noktadan sadece yukarı ilerleyerek en iyi sonucu bulmayı hedefler. *Tepe tırmanma* (hill climbing) denen bu yöntem, fonksiyon grafiğinin tepelerini tırmanır. Ancak çok sayıda dönme noktası içeren bir fonksiyonda çok sayıda tepe oluşur. Hangi tepenin en iyi çözüm olduğu bilenemez. Numaralama yöntemleri ise oldukça alışıl gelmiştir. Sürekli olan gerçel sayı aralıkları, belli sayıda parçaya ayrılarak parçalar denenir. Ancak problemler böyle çözülmek için büyük olabilir.

Bu yöntemin biraz daha geliştirilmiş şekli *Dinamik Programlama* (dynamic programming) ile oluşturulur. Parçalar arasından iyi görünenler seçilir. Bu parçalar alt parçalara ayrılarak işlem yinelenir. Bu yöntem de tepe tırmanma yöntemi gibi yanlış tepeleri araştırabilir. Dinamik Programlama tepelerin olmadığı aralıklarda başarılı ve hızlıdır.

En iyilemenin (optimizasyon); “bir işin daha iyi yapılması” ve “en doğru şekilde yapılması” olmak üzere iki amacı vardır.

Günümüzde rastgele aramaların kullanımı artmaktadır. Bu tip aramalar, eniyilemenin daha iyi yapma amacını sağlamakta daha başarılıdırlar. İnsanların bilgisayarlardan genel beklentisi mükemmellik olduğu için bu tip aramalar başarısız görünebilir. GA’lar klasik yöntemlerin çok uzun zamanda yapacakları işlemleri, kısa bir zamanda çok net olmasa da yeterli bir doğrulukla yapabilir.

4.5.2. Genetik Algoritmaların Klâsik Yöntemlerden Farkları

1. GA, problemin boyutundan bağımsızdır. Değişken sayısı ve bu değişkenlerin değerlerinin sonuçlara etkisi araştırma yöntemini değiştirmez.
2. GA araştırma uzayını dinamik olarak parçalar ve bir tek yerden değil, bir grup çözüm içinden arama yapar.
3. GA ne yaptığı konusunda bilgi içermez, nasıl yaptığını bilir.
4. Problemin nasıl çözüleceğine ilişkin hiç bilgi gerektirmez. Bu nedenle bir *kör arama* (blind search) yöntemidir.

5. GA'lar olasılık kurallarına göre çalışır. Programın ne kadar iyi çalışacağı önceden kesin olarak belirlenemez. Ama olasılıkla hesaplanabilir.

4.6. Uygulama Alanları

GA'ların en uygun olduğu problemler, geleneksel yöntemler ile çözümü olanaklı olmayan ya da çözüm süresi problemin büyüklüğü ile üstel orantılı olarak artanlardır. Bugüne kadar GA ile çözümüne çalışılan konulardan bazıları şunlardır:

1. **Optimizasyon (Optimization):** GA, sayısal optimizasyon ve kombinatoriyal optimizasyon problemleri olan devre tasarımı, doğrusal olmayan denklem sistemlerinin çözümü, tahmin sistemleri, veri analizi ve fabrika-üretim plânlaması için kullanılır.
2. **Otomatik Programlama (Automatic Programming):** GA, bilgisayar programları yardımıyla network sıralamasında (sorting), ders programı hazırlanmasında kullanılır.
3. **Makina Öğrenmesi (Machine Learning):** GA, robot sensorlarında, yerel ağlarda, mimarî tasarımı ve ağırlıkların bulunmasında, VLSI yonga tasarımı ve protein yapısal analizinde kullanılır.
4. **Ekonomi (Economics):** GA, ekonomik modellerin geliştirilmesinde ve işletilmesinde kullanılır.
5. **Bağışık Sistemler (Immune Systems):** GA, çok-gen'li ailelerin evrimi sırasında ve doğal bağışık sistem modellerinde kullanılır.
6. **Popülasyon Genetiği (Population Genetics):** GA, evrim ile ilgili sorulara yanıt bulmada kullanılır.
7. **Evrime ve Öğrenme (Evolution and Learning):** GA, bireylerin öğrenmesinde ve türlerin evrilmesinde kullanılır.
8. **Sosyal Sistemler (Social Systems):** GA, sosyal sistemlerin analizinde kullanılır.

5. MONTAJ HATTI DENGELEME – GENETİK ALGORİTMA İLİŞKİSİ

5.1 Giriş

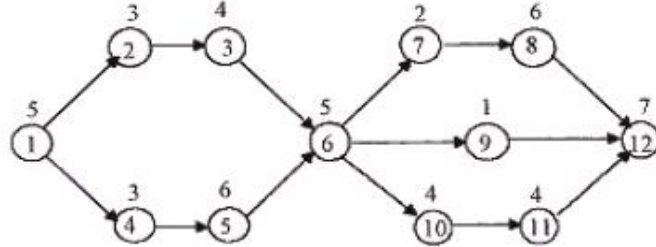
Çok istasyonlu hatların dengelenmesinde karşılaşılan sorunlardan biri de farklı iş öğelerinin, çeşitli iş istasyonlarına atanması problemidir. Önceki bölümlerde de sözedildiği gibi, literatürde bu problem, Montaj Hattı Dengeleme (MHD) problemi olarak geçer .

MHD, çok geniş bir araştırma alanına sahip olup, birçok endüstri dalında uygulanmaktadır. İlk çağdaş montaj hattı, 1913 yılında Ford Highland fabrikasında kurulmuştur. O zamandan beri, birçok araştırmacı, değişik MHD problemlerini çözmek için çeşitli çözüm yöntemleri geliştirmişlerdir (Aksoy, E., s.30).

MHD problemleri, NP-hard kombinatoriyal optimizasyon problemleri sınıfına dahil olduğundan bu yana, konu ile ilgili bilgisayar etkinlikli algoritma ve sezgisel yöntem geliştirmek amacıyla sayısız araştırmalar yapılmıştır (Ghosh ve Gagnon, 1989). Sezgisel yöntemlerin en temel ortak yönü problem odaklı olmaları ve problemin yapısına göre bilgiyi zekîce kullanarak arama çabasını olabildiğince en aza indirmekleridir. GA’lar da, zekî rastgele arama yöntemleri olarak, gezgin satıcı problemi, çizelgeleme ve MHD gibi çok çeşitli optimizasyon problemlerine uygulanmaktadır. Literatürde bu konuyla ilgili olarak şu ana kadar yapılan çalışmalar şunu göstermiştir ki, GA’lar zor problemlerin çözümünde oldukça verimli arama teknikleri olup bir çözüm kümesinden diğerine çok esnek hareket edebilme yetenekleriyle, probleme ilişkin spesifik karakteristikleri birleştirerek MHD problemlerinin çözümünde oldukça yararlı olmuştur.

Bu yararları sağlaması için yalnız GA operatörleri, MHD problemlerine uygun bir biçime getirilmek sûretiyle problemin yapısına uyumu sağlanmalıdır.

Tez kapsamındaki çalışmamız tek model ve işlem süreleri deterministik olan MHD problemlerini içermektedir. Gösterim aracı olarak problemde, çoğunlukla bir teknolojik öncelik diyagramı kullanılır. Şekil 5.1.'de örnek bir MHD problemi gösterilmiştir (Kim, Y.K., Kim, Y.J, 1996).



Şekil 5.1. Teknolojik öncelik diyagramı

- n : İstasyon sayısı (karar değişkeni)
N : Önceden kararlaştırılan tahminî istasyon sayısı
ct : Çevrim süresi (karar değişkeni)
CT : Önceden kararlaştırılan tahminî çevrim süresi
W : Toplam iş süresi
 T_j : = j. istasyonun istasyon süresi

MHD problemleri farklı amaçlar gözönünde bulundurularak çözülebilir. Buna göre üç tip MHD problemi vardır:

1. tip problem : Enküçükleme $f_1 = n$ (5.1)

Verilen bir CT çevrim süresi kısıtına bağlı olarak

2. tip problem : Enküçükleme $f_2 = ct$ (5.2)

Verilen bir N istasyon sayısı kısıtına bağlı olarak

3. tip problem : Enküçükleme $f_3 = \frac{1}{N} \sum_{j=1}^N \left(T_j - \frac{W}{N} \right)^2$ (5.3)

Verilen bir N istasyon sayısı kısıtına bağlı olarak

1. tip problemlerde, istenen üretim hızı (çevrim süresi), montaj öğeleri, iş öğesi süreleri, öncelik ilişkileri verilir. Burada amaç, istasyon sayısını enküçüklemeektir.

Daha az iş istasyonuna sahip bir hat daha düşük işçilik mâliyeti ve alan gereksiniminin indirgenmesi gibi sonuçlar doğuracaktır. Bu tür problemler, genellikle yeni bir montaj hattı tasarımında ortaya çıkar. Bu amaçla, öngörülen istemi elde edebilmek için istasyon sayısı azaltılmalıdır (Ponnambalam ve diğ., 2000).

Bunu daha da genişletirsek istemin arttığı durumlarda bu tip problem, yüklenecek ekstra istasyon sayısının enküçüklenmesinde de kullanılır.

2. tip problemlerde iş istasyonu sayısı ve üretim işçileri sayısı sabit olup amaç, çevrim süresini enküçükmektir. Bu, üretim hızını enbüyüklemeyle eşdeğer anlama sahiptir. 2. tip problemler, genelde organizasyonun elindeki sabit makina sayısını kullanarak yeni bir yatırım yapmadan, üretim miktarının artırılmasının istenmesi durumunda ortaya çıkar.

Burada öncelik ilişkilerinin yanısıra yerel kısıtları da tanımlayabiliriz. Ana montaj hattını dengelerken alt montaj hatlarını da hesaba katmalıyız.

1. tip problemler, 2. tip problemlere göre daha geneldir. Varolan kullanımdaki problem boyutu arttığında kullanımı zorlaşabilmektedir.

3. tip problemlerde amaç, iş yükünün olabildiğince aynı düzeyde tutulması ve dağıtılmasıdır. Bu dengeleme, ürün çıkış hızını artırmasının yanında daha önemli olarak, işçiler arasındaki iş yükünün istasyonlara eşit dağıtılmamasının doğuracağı endişeleri de ortadan kaldırır. Farklı seçenek ölçütlerin, bu amacın ölçütleri olarak nasıl kullanılabileceği, Rachamadugu ve Talbot (1991) tarafından anlatılmıştır. İş yükünün düzgünleştirilmesinin enbüyüklenmesi, 3. tip problem formülasyonunda gösterilen iş yükü varyansının enküçüklenmesiyle eşdeğerdir.

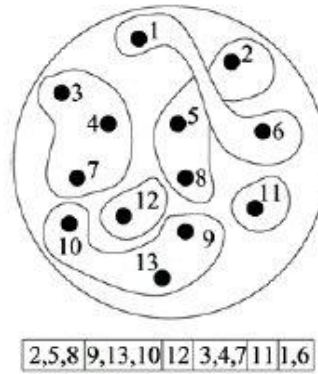
5.2. Genetik Algoritmanın MHD Uygulama Karakteristikleri

5.2.1. Kromozom Temsili

5.2.1.1. Genel bilgi

Genetik algoritmanın oluşturulmasındaki ilk adım, problemin kromozom temsiliinin oluşturulmasıdır (Kim, Y.K., Kim, J.J, 1996). Bu temsiliyetin MHD problemi karakteristiklerine göre çok iyi tanımlanarak yapılması, GA'nın diğer izleyen adımlarını da önemli derecede etkilemektedir. Burada temel olarak iki tür yaklaşım vardır. Birincisi, iş ögelerinin çözüm sonrası atanacağı belirli olan iş istasyonuna göre sıralamayı, daha çok rağbet edilen diğeri ise iş ögelerinin öncelik sıralarını koruyarak kromozom üzerinde dizilmesi ve istenen amaç fonksiyonu doğrultusunda gruplanmasını içermektedir.

Kromozom üzerindeki iş ögelerini temsil eden genlerin gruplanarak iş istasyonuna atanması düşüncesi, Şekil 5.2.'de gösterilmektedir.



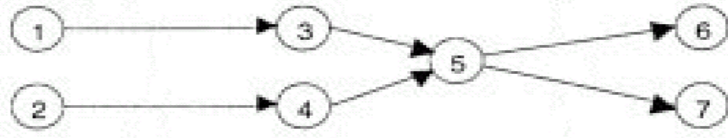
Şekil 5.2. İş ögelerinin gruplanması ve kromozom gösterimi (Falkenaur, 2001)

Bu açıdan MHD problemlerine uyarlanabilir üç kodlama yöntemi vardır.

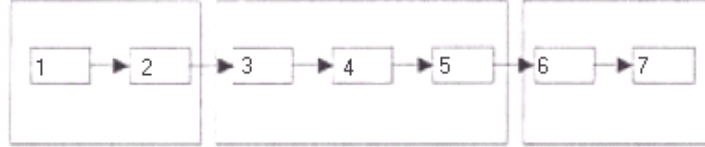
5.2.1.2. İş ögesi odaklı kodlama

Eğer i iş ögesi, j iş istasyonuna atanıyorsa, j istasyon numarası kromozomdaki i . konumda bulunan iş ögesinin dizi numarası olur [Anderson, E.J., Ferris, M.C., 1994]. Bu kodlamanın öncelik diyagramı ve montaj hattıyla olan bağlantısı, Şekil 5.3.'de gösterilmiştir.

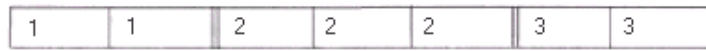
Öncelik Diyagramı



Montaj Hattı



Kromozom Temsili

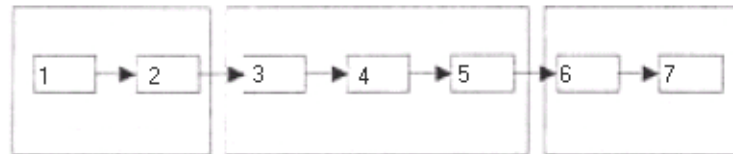


Şekil 5.3. Kromozom temsili (iş ögesi-istasyon odaklı)

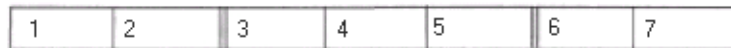
5.2.1.3. Öncelik sıralı kodlama

Tüm iş ögeleri, iş istasyonlarına atandıkları sırada önceliklerini bozmadan kromozom üzerinde birbiri ardınca dizilirler (Leu ve diğ., 1994).

Her iş ögesi, kromozom üzerindeki dizide belli bir rakamla temsil edilir. Dizi uzunluğu, montaj hattındaki iş ögesi uzunluğu kadardır. İş ögeleri, kromozom üzerinde işlem sıralarına göre göreceli olarak sıralanır. Daha sonra iş ögeleri, istasyonlara, toplam süreleri çevrim süresimi geçmeyecek şekilde atanır. Bu kodlama, Şekil 5.4.’de gösterilmiştir (Erel ve diğ., 2000).



Kromozom Temsili



Şekil 5.4. Kromozom kodlama (öncelik sıralı)

5.2.1.4. Bölme esaslı tanımlama

Öncelik sıralı tanımlamaya ek olarak kromozomu iş istasyonlarına bölecek bölmeler tanımlanır.

5.2.1.5. Önerilen kodlama yapısı ve avantajları

Bu kodlama şekillerinden hangisinin kullanılacağı BMHD problemlerinin tipine göre değişmektedir. Örneğin, öncelik sıra tanımlamalı kodlama şekli biraz önce sözügeçen 3. tip problem için uygulanabilirken diğer iki kodlama şekli, iş istasyonu sayısı önceden belirtilmesi gereken 2. ve 3. tip problemler için uygundur. İkinci avantajı olarak MHD için geliştirilmiş, birçok farklı genetik operatör, öncelik sıralı kodlama yöntemine uygulanabilir. Diğer iki kodlama yöntemi ise, ceza fonsksiyonu kullanımı gibi daha karmaşık ek işlemcilere gereksinim duyar.

Öncelik sıralı tanımlamada, iş öğelerinin birbirlerinin öncelik sıralarını bozmadan dizilmesiyle oluşturdukları kromozom temsili olurlu sıralama olarak adlandırılır. Bu olurlu sıralama gruplanmaya geçmeden önceden tek bir atamadan avantajlı olarak birçok olası gruplama kombinasyonlarını içerir. İş öğelerinin en iyi şekilde tahsis edilmesi ve çözüme ulaşmak üzere, bu dizinin uygun bir şekilde düzenlenerek (decoding) bir çözüm kümesi hâline getirilmesi gerekir.

Çevrim süresi verilmiş ise Leu (1994)'nın uyguladığı atama yöntemi kullanılır. Burada çevrim süresi kısıtını çiğnemedi, iş öğeleri sırayla istasyona atanır. İş öğeleri bitene kadar yeni bir istasyon oluşturularak işlem sürdürülür. 1. tip problemler için uygulanabilen bu yöntem, iş istasyonu sayısının verildiği 2. ve 3. tip problemlerde elverişli değildir. Burada ise benzer şekilde, verilen istasyon sayısı toplam işlem süresine bölünerek kuramsal en küçük çevrim süresi elde edilerek atamalar yapılır.

5.2.2. Başlangıç popülasyonu ve büyüklüğü

Başlangıç popülasyonu, öncelik ilişkilerini sağlayacak şekilde seçilir. Bu işlem sezgisel yöntemlerle ya da rassal olarak yapılabilir. Rassal olarak üretildiği

varsayılan bir başlangıç kromozomu, aşağıdaki adımlar izlenerek olurlu (feasible) duruma getirilir.

Adım 1: Öncelik ilişkileri sıralanmamış uygun iş ögesi numaralarını içeren random bir küme oluştur ve boş bir dizi yarat.

Adım 2: Eğer başlangıçta oluşturduğumuz küme boş ise işlemi sona erdir, aksi durumda adım-3'e git.

Adım 3: Kümeden rassal olarak bir öge seç ve onu olurlu çözümü oluşturacağımız diziye ekle.

Adım 4: Seçilen ögeyi çıkartarak rassal kümeyi güncelle ve aynı zamanda diziye yeni gelen her ögeyi, varolan iş öğelerini dikkate alarak öncül ya da ardıl olarak sıralayıp diziyi güncelleştir.

Popülasyon büyüklüğü, MHD problemlerinin tayininde kesin bir değer olmamakla birlikte genel kabule uygun olarak 50-300 arasında seçilebilir.

5.2.3. Uygunluk Fonksiyonu

Uygunluk fonksiyonu, potansiyel çözümlerin istenilen amaca ne kadar yakın veya uzak olduğunun hesaplanmasını sağlar. Bir sonraki kuşağı oluşturacak bireylerin, yeniden üreme (reproduction) operatörüne girmesi için varolan popülasyondan hangi bireylerin seçileceği, uygunluk fonksiyonu değeriyle belirlenir. GA çözümünün kalitesini etkileyen çok önemli bir etmendir.

Uygunluk fonksiyonu, iş istasyonu sayısının enküçüklenmesi gözönünde bulundurularak oluşturduğumuzu düşünelim. Ama aynı iş istasyonu sayısına sahip iki montaj hattından biri diğerine göre daha iyi dengelenmiş de olabilir. Örneğin üç iş istasyonuna sahip bir montaj hattındaki iş sürelerinin (dk) 30-50-40 ve 50-50-20 olduğunu varsayarsak, 30-50-40 çözümünün diğerine göre daha iyi dengelenmiş bir çözüm olarak önceliği vardır. Bu amacın gözetildiği bir genetik algorithmada, aşağıdaki gibi bir uygunluk fonksiyonu tanımlamak yerinde olacaktır:

$$\text{Uygunluk Fonksiyonu} = 2\sqrt{\frac{\sum_{k=1}^n (S_{enb} - S_k)^2}{n}} + \frac{\sum_{k=1}^n (S_{enb} - S_k)}{n} \quad (5.4)$$

N : İstasyon sayısı

S_{enb} : En büyük istasyon zamanı

S_k : k. istasyonun iş süresi toplamı

Formülün birinci kısmı, aynı iş istasyonuna sahip çözümlerdeki en iyi dengeyi bulmayı amaçlarken, ikinci kısımda ise çözümdeki istasyon sayısı enküçülenmeye çalışılır.

Burada keyfi olarak birinci amacın ikincisine görece daha baskın olduğu düşüncesiyle, birinci amaç fonksiyonu 2 katsayısıyla çarpılmıştır. GA'da uygunluk fonksiyonu, farklı amaçları tek bir paydada toplamak sûretiyle çözüme esnek bir yapı kazandırır (Erel ve diğ., 2000).

3. tip problemin çözümünde, iş yükünün dengelenmesi amacını ele aldığımızda uygunluk fonksiyonu aşağıdaki gibi olur (Kim, Y.J., 1998).

$$\text{Uygunluk fonksiyonu} = \frac{1}{n} \sum_{j=1}^n \sqrt{|T_j - T_{ort}|} \quad (5.5)$$

T_j : j. istasyon için iş istasyonu süresi

T_{ort} : n adet iş istasyonu süresinin ortalaması

Bu açıdan bakıldığında GA, tek amaçlı problemlerin yanısıra çok amaçlı problemlerin çözümü için de oldukça uygundur. Zâten çok amaçlı GA ve klâsik tek amaçlı GA çözümü arasındaki tek fark, uygunluk fonksiyonu ile seçim prosedürlerinde izledikleri yöntemlerdeki farklılıktır.

5.2.4. Çaprazlama ve Mutasyon

5.2.4.1. Genel Bilgi

Çaprazlama ve mutasyon operatörleri belirli olasılıklar altında uygulandığından, probleme göre bu parametrelerin ne olması gerektiği net değildir. Her iki operatörde

kullanılacaksa, bu iki operatör olasılık toplamı genelde 1 olacak şekilde ayarlanabilir. Örneğin çaprazlama olasılığı % 98 iken mutasyon olasılığı % 2 olur.

Yalnızca mutasyon operatörünün kullanılması durumunda, mutasyon olasılığının % 40-60 ve salt çaprazlama operatörünün kullanıldığı durumlarda ise bu oranın % 85-95 aralığında olmasının, genelde kalitesi yüksek çözümler ürettiği kaydedilmiştir (Kim, Y.K., Kim, Y.J., 1996).

Çaprazlama operatörü bilindiği üzere iki ebeveyn diziden daha etkili oğul çözüm dizileri oluşturmak umuduyla yapılan bir süreçtir. MHD problemleri gibi klâsik sıralama problemlerinde uygulanabilecek standart çaprazlama operatörleri şunlardır:

- Kısmî bölgesel çaprazlama (Goldberg, 1985)
- Güçlendirilmiş kenar yeniden birleştirmesi (recombination) (Starkweather, 1991)
- Sıralama çaprazlama (Davis, 1985, pp. 162-164)
- Düzgün (Uniform) tabanlı sıralama çaprazlama (Davis.L., 1991)
- Dönüşlü (cycle) çaprazlama (Oliver, 1987)

Bu operatörler, iş öğelerinin sıralı temsilleri ya da permütasyonlarına göre işlem yaparlar. Bu yolla iş öğelerinin ebeveyn bireyden gelen sıralama ve pozisyon bilgisi kullanılarak bir sonraki kuşağa aktarılıp, çözüm kalitesinin sürekli daha iyiye gitmesi sağlanır.

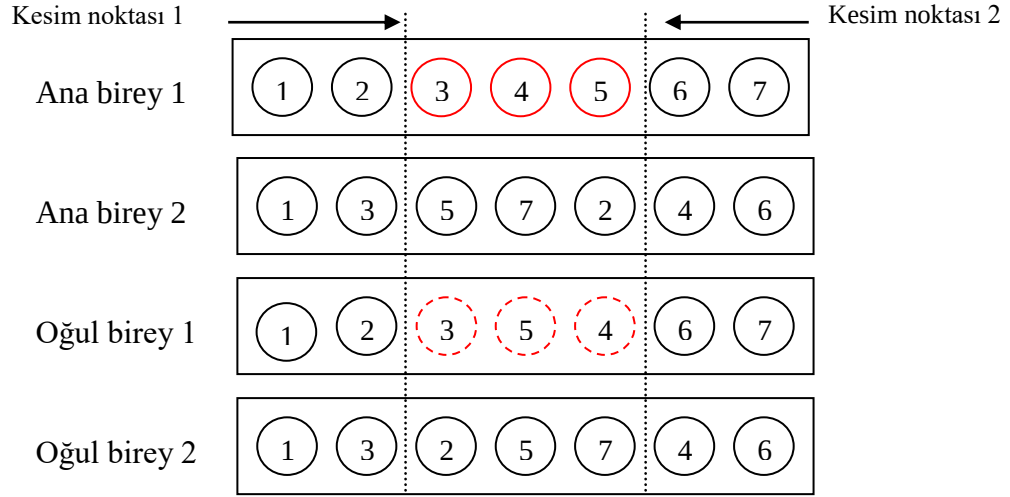
5.2.4.2. İki kesim noktalı sıralama çaprazlama

Sıralama ilkesi üzerine kurulmuş çaprazlama operatörü Davis (1985)'in önerdiği sıralı çaprazlamanın farklı bir varyantı olarak GA'ların MHD problemlerine uygulandığı, Erel ve diğ. (2000) çalışmasında kullanılmıştır.

Buna göre sürecin temel işleyişi şöyledir:

Seçilen iki ebeveyn kromozom, rasgele seçilen iki noktadan kesilir. Oğul birey (offspring) kesim noktası dışındaki ana bireylerin sahip olduğu lokasyonlardaki aynı

genleri alır. Kesim noktaları aralarındaki genler ise diğer bireydeki sıralamaya göre karıştırılır. Bu prosedür, Şekil 5.5.'de görülmektedir.



Şekil 5.5. İki noktalı çaprazlama

Bu işlemin ALB problemleri için uygun olmasının ana nedeni, oluşan oğul bireylerde olurluğu (feasible) korumasıdır. Ana bireyler olurlu çözüm gruplarından oluşmuş iken, bunlardan oluşan oğul bireyler de olurludur. MHD problemlerinde, popülasyonlardaki olurluluğun korunması anahtar bir rol oynar. Bu durum, hesaplamadaki işlem yükünü de oldukça azaltır.

Eğer bir problem içindeki öncelik ilişkileri ne kadar artarsa, ana bireylerden daha iyi oğul bireyler oluşması olasılığı da o derecede azalmaktadır. Böyle bir durumda genel crossover operatörlerini kullanmak yerine, problemin özüne uygun ana birey genleri üzerinde daha güçlü değişiklik yapabilecek operatörler geliştirilebilir.

Bizim kullandığımız iki kesim noktalı crossover operatöründeki temel amaç, ana bireye baş ve kuyruk kısmında benzeyen oğul bireyler yaratarak daha yakın komşu noktalarda aramayı yönetmektir. Oğul bireyler, bu çeşit bir operatörle ana bireye uygunluk fonksiyonu açısından daha yakın olurlar.

Salt kesim noktalarının ortasında kalan genler değişime uğrar. Bunun tersi olarak tek noktalı bir crossover ise kromozomun yarısı üzerinde tümüyle bir değişiklik yapabilecek olup böyle bir değişim çok güçlü olabilir ve oğul bireyi yerel arama çevresinden uzaklaştırabilir. Benzer şekilde ikiden fazla kesim noktası baz

alındığında da, uygunluk fonksiyonundaki değişim, kesim noktaları arasında kalan genlerin değiş-tokuş yönetiminin nasıl yapılacağına bağlı olarak, çok küçük ya da aşırı derecede olabilir.

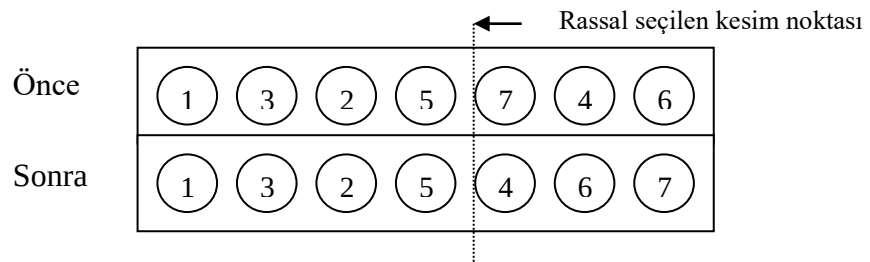
Bu noktada, tek noktalı crossover operatörünün kullanılan iki nokta kesimli crossover operatörüne kıyasla, yüksek esneklik oranlı (F-ratio) problemlerde daha iyi sonuçlar verebileceği söylenebilir.

5.2.4.3. Scramble (karıştırılmış) mutasyon

Mutasyon operatörü, problem için çözüm arama uzayının farklı noktalara anı sıçramalarla keşfedilmesi sūretiyle GA'yı güçlendirici bir işleve sahiptir. GA'nın geleneksel 0-1'lerin kullanıldığı ikili kromozom temsiliyetinde; mutasyon, bu bit değerlerinin gen üzerinde seçilen rassal bir noktada seçenek karşıt değerini alarak kromozom yapısının değişmesiyle sonuçlanmakta idi.

Sıralama problemlerindeki işleyiş de genelde aynı olup, mutasyona uğrayacak genin seçimi, sayısı ve değer değişimi sıralama öncelikleri temel alınarak farklılaşmaktadır. Örnek olarak MHD problemlerinin GA ile çözümündeki ilk çalışma olarak kabul edilen Leu (1994), “scramble (karıştırılmış) mutasyon” operatörünü kullanmıştır.

Bu operatörde, kromozom üzerinde belirli bir gen noktası düzgün olarak seçilir ve bu noktadan sonraki genlerin, öncelik ilişkileri bağlamında olurluluğu bozmayacak şekilde sıraları değiştirilir (Şekil 5.6.).



Şekil 5.6. Scramble mutasyon

Diğer bir mutasyon yönteminde ise yine rassal olarak seçilen bir iş ögesi, dizi içindeki diğer iş öğelerinden herhangi birinin arasına sıralama problemlerindeki genel ilke olan olurluluğu bozmayacak şekilde konur.

Geleneksel mutasyon operatörlerinin, MHD problemlerine uygulanması durumunda ise, mutasyon işleminden sonra kromozomu olurlu çözüm durumuna getirmek için

bâzı iterasyonlar uygulanır. Ama bu, işlem yükü getirdiği için pek yeğlenmez. Scramble mutasyon ve türevi olan, sıralama problemlerine uygun operatörler kullanılması tavsiye edilir.

5.2.5. Ölçeklendirme

Elde edilen uygunluk değerleri, seçim prosedüründe kullanılmak üzere, uygunluk değerleri toplamı 1 olacak şekilde normalize edilmelidir. Elimizdeki amaç fonksiyonu, uygunluk değerini enküçüklemek olduğu için ve kromozom uygunluğuyla orantılı seçim olasılığı belirlemek üzere yüksek ölçeklendirilmiş olan uygunluk değerini, elde edilen en düşük uygunluk değerine denk gelecek şekilde atama yapılmalıdır.

Bunu sağlamak için en yüksek (kötü) uygunluk değerinin iki katı alınarak her farklı uygunluk değerinden çıkartılır. Çıkan sayı ilgili uygunluk değerinin normalize edilmiş yeni değeridir. Elde edilen her yeni değer, toplam uygunluk değerine bölünerek, toplam da 1 olacak şekilde revize edilir. Eğer amaç fonksiyonumuz enbüyükleme olsaydı, elde edilen uygunluk değerlerini, toplam uygunluk değerine oranlamak sûretiyle ölçeklendirmiş olacaktık.

5.2.6. Seçim Prosedürü

Seçim aşamasında varolan yöntemler içinde en basit ve en kolay uygulanabilir niteliğiyle bilinen yöntem olan rulet tekerleği seçim prosedürü uygulanmıştır. Her kromozom uygunluk değerleriyle orantılı olarak $[0,1]$ aralığında bir çizgi üzerinde birbirine bitişik bölümler şeklinde dizilirler. Bununla birlikte, her bir bireye ilişkin bölümün uzunluğu, onun uygunluk değeri kadar olur. $[0,1]$ aralığında düzgün bir rastgele sayı üretilir ve rastgele sayı hangi bölüm içine denk gelirse, o bölümün ilişkin olduğu birey seçilir. İşlem, eşleşecek popülasyonun gerekli adedine ulaşıncaya dek sürdürülür (Erel ve diğ., 2000).

5.2.7. Elitizm

Elitizm kuralı ise ana bireyin oluşan oğul bireylerden daha uygun olması durumunda uygulanır. Sözedilen bu operatörler Leu (1994)'nın genetik algoritmaların montaj hatlarına uygulanması konusunda yaptığı ilk çalışmadakiler ile aynıdır. Elitizm

uygulamasının GA'nın performans artışına olumlu bir etkisi vardır. Bu nedenle MHD problemleri çözümünde uygulanması tavsiye edilir.

5.2.8. Dinamik Bölümleme

5.2.8.1. Giriş

MHD kromozom yapısına uygun olarak işlem süresini korumak üzere kromozom yapısı üzerinde değişiklik yapan Dinamik Bölümleme (Dynamic Partitioning) tekniği, MHD problemlerinde performans artışı sağladığından, Erel ve diğ. (2000) tarafından önerilmiştir. Standart GA operatörleri arasında yer almayan bu yöntem, kromozom yapısını belli ölçütleri tatmin eden iş öğelerini istasyonlara atayarak yeniden düzenler ve sonraki iterasyonlarda bu öğeleri dondurularak geriye kalan öğeler üzerinde işlemlerini sürdürür. Bu tekniğin geliştirilmesinin arkasında yatan temel motivasyon, performanstaki düşüşü engellemek ve hesap işlem süresinin azaltılmasıdır. Dinamik Bölümleme (DB), GA'nın bazı iş öğelerinin dondurularak sabit kılınması veya bölümlendirilmesi yoluyla geriye kalan daha az miktardaki iş öğelerinin optimizasyonuna odaklanması yoluyla daha etkili çalışmasını sağlar.

DB prosedürü sonucunda dondurularak sabitlenen iş öğeleriyle, oluşan istasyon süresi, uygunluk fonksiyonunu enküçükler. Bununla birlikte bazı iş öğelerinin optimal dengeyi çok fazla bozmadan sabitlenmesi, GA'nın geri kalan diğer iş öğeleri üzerine odaklanıp konsantre olmasını sağlar.

Mutasyon ve çaprazlama operatörleri, eğer bu prosedürle DB ölçütlerini sağlayan istasyonları sabitlemeseydi, mekanizması zayıf çözümlerle uğraşmak yerine, zaten varolan ve istenilen ölçütleri sağlayan istasyonları oluşturan iş öğeleri üzerinde zaman harcayıp, onları karıştırıp sıra yapılarını bozacaktı. Bununla birlikte aynı iterasyon sayısı ile DB'nin kullanıldığı süreç salt GA'nın kullanıldığı yöntemle göre daha fazla seçenek kombinasyonları denemektedir.

5.2.8.2. Yürütme

Geriye kalan iş öğeleri arasındaki öncelik ilişkilerini korumak üzere ilk ve son istasyonlardaki iş öğelerini, dondurulmuş olarak ele alıyoruz. Dondurulma için ikinci ölçüt, potansiyel olarak donduğu kabul edilen istasyonlarda optimum istasyon süresini elde etmektir.

Bu optimallik koşulu, uygunluk fonksiyonuna göre değişmektedir. İlgili sabitleme ölçütü, Erel ve diğ. (2000) makalesindeki uygunluk fonksiyonuna göre değerlendirildiğinde, aşağıdaki gibi olmaktadır.

$$\frac{|S^* - S^i|}{S^*} < DBS, \quad i = 1, \dots, n \quad (5.6)$$

$$DBS = 0,01; 0,02; 0,03; \dots$$

$$S^* = \frac{\sum_{i=1}^n S_i}{n^*} \quad n^* = \left\lceil \frac{\sum_{i=1}^n S_i}{CT} \right\rceil \quad (5.7)$$

n_{enaz} : Olası en az istasyon sayısı

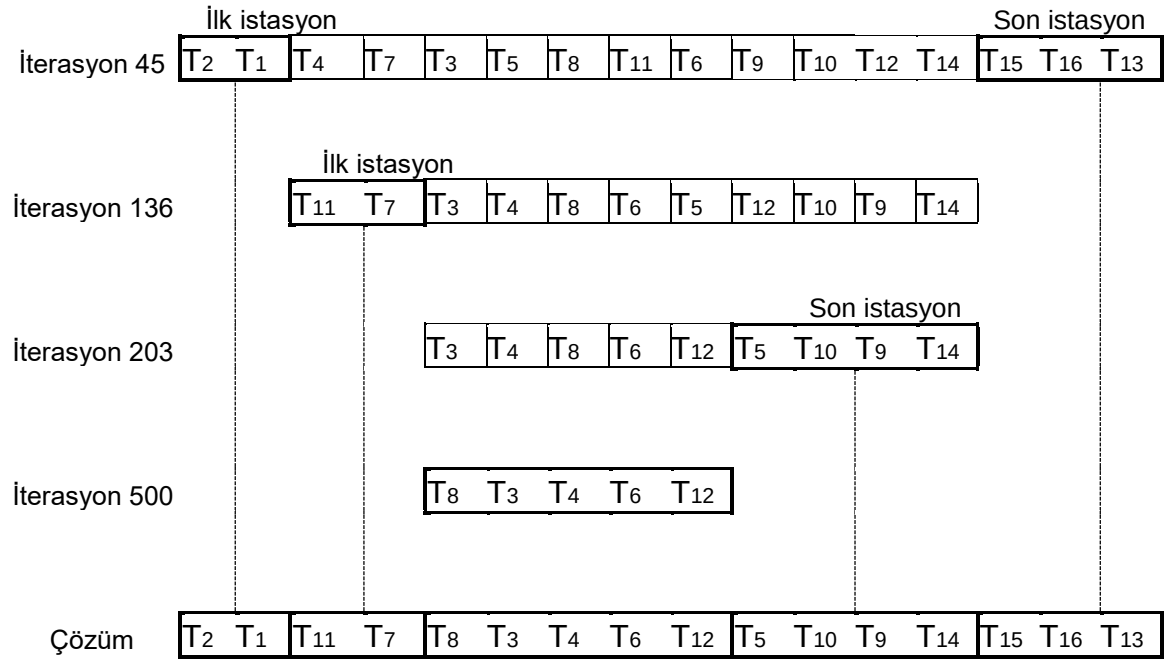
CT : Çevrim süresi

Dinamik Bölümleme Sabiti (DBS) parametresi, algoritmamızı istasyon dondurma ölçütümüzü düzenlemek sùretiyle akort etmemizi sağlar (Bu değerin artması, çalışma başına ortalama bölme sayısının artmasını, dolayısıyla işlem süresinden tasarruf edilmesini sağlar ama bu takdirde daha düşük bir çözüm değeriyle sonuçlanabilir.).

Her iterasyonda, Dinamik Bölümleme (DB) için yukarıda sözügeçen iki ölçütün kontrolü yapılır. Eğer ilk veya son istasyon, ölçütü sağlarsa, bu istasyon ya da istasyonlar dondurularak sabitlenir ve GA, bir sonraki iterasyona sabitlenmemiş diğer iş öğelerini kullanarak iterasyonu sürdürür. Bu şekilde kromozom uzunluğu, her bölme ve sabitleme işleminden sonra azalır, böylelikle GA diğer kalan iterasyonlarda iterasyon başına daha az süre harcar.

GA'daki popülasyon büyüklüğü ve kromozom sayısı değerleri, prosedürün başlangıcından bitişine kadar hep aynı kalır. En iyi çözümü sağlayan, en iyi uyumlu kromozom, her iterasyonda DB ölçütlerine göre kontrol edilir. Eğer ölçütleri sağlıyorsa DB en uygun kromozoma uygulanır ve sabit tutulan genler (iş öğeleri) popülasyondaki diğer tüm kromozomlar gözönünde bulundurularak çıkartılır. Sabitlenen bu iş öğeleri, kromozomun ister başlangıcında ister sonunda olsun, öncelik kısıtları açısından herhangi bir olumsuzluk yaratmazlar.

Dinamik Bölümleme mekanizması, Şekil 5.7.'de bir örnek üzerinde gösterilmiştir. Bu örnekte DB ölçütleri, hem ilk hem de son istasyonda 45. iterasyon sonunda sağlanmaktadır. Bu nedenle 1, 2, 13, 15 ve 16 no.'lu iş öğeleri dondurularak sabitlenir.



Şekil 5.7. Dinamik bölümleme mekanizması

Bundan sonra GA, kalan 11 iş öğesini, sabitlenen 5 öğeyi gözardı ederek dengeler. 136. iterasyonda salt ilk istasyon DB koşullarını sağlar. Bundan dolayı 7 ve 11 no.'lu iş öğeleri de sabitlenir. Bu dondurulan iş öğeleri, daha sonra bulunan en iyi kromozoma, son iterasyonda sabitlendikleri sırada eklenir. DB'nin başlangıçta ilk iterasyonla birlikte uygulanmasının bizi erken bir yerel optimal sonuca zorunlu bırakacağı kabul edilir. Bunu önlemek için başlangıç rassal popülasyonuna,

bölümleme için daha kabul edilebilir bir uygunluk değeri elde etmek üzere “ısınma dönemi” denilen belli bir alışma süresi verilir.

DB’nin performansa olan etkisi, arama uzayının daralması anlamına gelen esneklik oranının artmasıyla ters orantılı olarak düşmektedir. Bunun nedeni ise, öncelik ilişkilerinin artmasının sonucunda ortaya çıkan az miktardaki olurlu sonuca bağlı olarak aynı düzeydeki DBS bölümleme olasılığının azalmasıdır. Sonuç olarak, DB’nin GA yaklaşımlı esneklik oranının düşük olduğu MHD problemleri çözümünde, kullanılması tavsiye edilir.

5.2.9. Genetik Algoritma Durma Koşulu

Algoritma, belli bir kuşak sayısından sonra sona erer. Erdal Erel ve diğ. (2000)’nin yaptığı çalışmada bu değer, 500, 1.000 ve 2.000 olarak alınmakla birlikte, genel kabule uygun değerlerde alınabilir. Sonuç itibarıyla, en son kuşaktaki o ana kadar üretilmiş en iyi sonuç, problemin çözümü olarak alınır.

6. LİTERATÜR ARAŞTIRMASI

6.1. Giriş

MHD problemleri ilk olarak Helgeson (1954) tarafından formüle edildiğinden bu yana birçok çözüm yaklaşımı önerilmiştir. Klâsik MHD problemi ise, problemin matematik formülasyonunu ilk yayımlayan Salveson (1955) tarafından ortaya atılmıştır. O zamandan beri, birçok araştırmacı, değişik montaj hattı dengeleme problemlerini çözmek için çeşitli çözüm yöntemleri geliştirmişlerdir.

Ghosh ve Gagnon bu konuda yapılan çalışmaları dört ana kategori altında sınıflandırmıştır. Tek modellenli deterministik (TMD), tek modellenli stokastik (TMS), çok/karışık modellenli deterministik (KMD) ve çok/karışık modellenli stokastik (KMS). Bizim çalışmamızda da iş öğelerinin zamanlarının kesin olarak belirli olduğu tek model veya ürüne ilişkin belli bir etkinlik ölçütünün eniyilendiği TMD MHD problemi ele alınmıştır. TMD, üzerinde en çok çalışma yapılan kategorilerden biridir.

MHD problemlerinin TMD versiyonunun optimal çözümlerini bulmak üzere birçok çalışma yapılmıştır. Bunlar, genel olarak doğrusal programlama ile Salveson (1955); tamsayılı programlama ile Bowman (1960), Klein (1963), Patterson ve Albracht (1975), Talbot ve Patterson (1984); dinamik programlama ile Jackson (1956), Held (1963), Schrage ve Baker (1978); dal-sınır algoritması ile Jackson (1956), Johnson (1981), Wee ve Magazine (1981)'dir. 100 iş öğeli yalın bir problemi, sözü geçen benzeri yöntemlerle çözmek olanaksız iken, kayda değer bir araştırma çabası da sezgisel yaklaşımların geliştirilmesi için harcanmıştır. Bunlar arasında da en ileri gelenleri; Dar-El MALB (1973), Dar-El ve Rubinovitch MUST (1979), Baybars LBHA (1986), Tonge (1965), Moodie ve Young (1965), ve Nevin (1972) sezgisel yöntemleridir. Baybars (1986), kendi önerdiği sezgisel yöntemin performansını, Tonge (1965) problemi üzerinde Tonge, Moodie&Young (1965) ve Nevin sezgisel yöntemleriyle karşılaştırmıştır.

Bizim bu çalışmada yaptığımız literatür araştırması, MHD problemlerinin TMD ve TMS sınıflandırmalarını içermektedir. 1994-2002 dönemleri arasında yayımlanan bu makaleler; son 20 yılın en popüler çözüm yaklaşımları olan Genetik Algoritma, Bulanık Mantık ve diğer sezgisel tabanlı yöntemlerin karışımından oluşan çalışmaları içermektedir.

Tablo 6.1.'de literatür araştırması sonucunda bulunan yayınlara ilişkin bilgiler kronolojik olarak sıralanmıştır.

Tablo 6.1. 1994-2002 yılları arası MHD yöntemleri için literatür araştırması sonuçları

No	Makale Adı	Yayımlandığı Dergi	Yazar	Tarih	Sayfa
1	Assembly Line Balancing using Genetic Algorithms with Heuristic-Generated Initial Populations and Multiple Evaluation Criteria	Decision Sciences	Leu, Y.Y., Matheson, L.A, Rees, L.P.	Ağustos 1994	581-606
2	Genetic Algorithms for Combinatorial Optimization: The Assembly Line Balancing Problem	ORSA Journal on Computing	Anderson E.J., Ferris M.C.,”	Eylül 1994	161-173
3	Solving Fuzzy Assembly-Line Balancing Problem with Genetic Algorithms	Computers& Industrial Engineering	Tsujimura Y., Gen M, Kubota E.	Eylül 1995	543-547
4	Genetic Algorithm for Assembly Line Balancing	International Journal of Production Economics	Rubinovitz J., Levitin G.	Ekim 1995	343-354
5	A Genetic Algorithm for Assembly Line Balancing	Production Planning&Control	Suresh G., Vinod V.V, Sahu S	Şubat 1996	38-46
6	Genetic Algorithms for Assembly Line Balancing with Various Objectives	Computers& Industrial Engineering	Kim Y.K., Kim J.J., Kim Y.H	Haziran 1996	397-409
7	Balancing SMD Assembly Lines with Genetic Algorithms	Proceedings of the Japan/USA Symposium on Flexible Automation	Schaffer, J. David;(Philips Lab, USA); Eshelman, Larry J	Temmuz 1996	1547-1552

Tablo 6.1. (devamı)

No	Makale Adı	Yayımlandığı Dergi	Yazar	Tarih	Sayfa
8	Assembly Line Balancing using A Genetic Algorithm with Repair Routine	Intelligent Engineering Systems Through Artificial Neural Networks	Chan, K.C.:(The Univ of New South Wales, Sydney, Aust);Tantono, R	Kasım 1996	385-390
9	Fuzzy Assembly Line Balancing using Genetic Algorithms	Computers& Industrial Engineering	Gen M, Tsujimura Y, Li YX	Aralık 1996	631-634
10	Handling the Assembly Line Balancing Problem in the Clothing Industry using A Genetic Algorithm.	International Journal of Clothing Science and Technology	Kelth C.C. Chan, Patrick C.L. Hui, K.W. Yeung, Frency S.F.Ng	Ekim 1998	21-37
11	Method for Assembly Line Busing Genetic Algorithms	Intelligent Engineering Systems Through Artificial Neural Networks	Xu, G.; (Southampton Inst, Engl); Penson, R.; Littlefair, G.;Callan, R.	Kasım 1998	329-334
12	A Heuristic Based Genetic Algorithm for Workload Smoothing in Assembly Lines	Computers & Operation Research	Yong Yu Kim, Yeo Keun Kim, Yongkyun Cho	Şubat 1998	99-111
13	Optimizing An Assembly Line through Simulation Augmented by Genetic Algorithms	International Journal of Advanced Manufacturing Technology	Lee, S.G.; (Nanyang Technological Univ, Singapore); Khoo, L.P.; Yin, X.F.;	Mart 2000	220-228
14	Genetic Production Line-Balancing for the Hard Disk Drive Industry	International Journal Of Advanced Manufacturing Technology	Zhao, Z.Y., de Souza, R.	Nisan 2000	297-302
15	A Multi-Objective Genetic Algorithm for Solving Assembly Line Balancing Problem	International Journal Of Advanced Manufacturing Technology	Ponnambalam, S. G, Aravindan, P., Naidu, G. M.	Mayıs 2000	341-352
16	Assembly Line Balancing using Genetic Algorithms	Journal Of Intelligent Manufacturing	Sabuncuoglu, I., Erel, E, Tanyer, M.	Mayıs 2000	295-310

Tablo 6.1. (devamı)

No	Makale Adı	Yayımlandığı Dergi	Yazar	Tarih	Sayfa
17	Applying Genetic Algorithms to the U Shaped Assembly Line Balancing Problem	Mathmetical and Computer Sciences Department, The University of Tulsa	Debora, A. Ajenblit, Roger L. Wainwright	Eylül 2000	
18	Hybrid Assembly Line Design	Proceedings of the IEEE International Symposium on Assembly and Task Planning	Rekiek, B. (CAD/CAM Department, Univ. of Brussels), Delchambre, A.	Mayıs 2001	73-78
19	A Hybrid Genetic Algorithm for Assembly Line Balancing	Journal Of Heuristics	Goncalves, J.F., Almeida, J.R.	Kasım 2002	629-642

6.2. Genel Olarak Genetik Algoritma Yaklaşımlı Çözümler

Literatürde MHD probleminin stokastik ve deterministik işlem süreli versiyonlarının GA tabanlı çözümü konusunda 1994 yılından bu yana yayımlanmış 19 makale bulunmuş ve incelenmiştir. Bunlardan en bilinen ilk ikisi TMD problemi ile uğraşmışken, üçüncü çalışma TMS üzerinde yapılmıştır. Diğerleri ise genelde tek modellenli deterministik ve çok amaçlı hatların dengelenmesi üzerine olup farklı alanlarda çeşitli çalışmalar da vardır.

İlk çalışma, Leu ve diğ. (1994) tarafından yapılmış olup burada başlangıç popülasyonunun oluşumunda sezgisel yöntemlerden yararlanıldı. Aynı zamanda montaj hattında çoklu ölçütlerin ve tek bir iş ögesinin bir istasyona atanması zorunluluğu benzeri yan kısıtların da gözönünde bulundurularak dengelenebileceğini gösterdi. Yazara göre GA'nın iki avantajı vardı.

- (i) GA, popülasyonda tek bir noktada arama yapmaz. Bu da birçok çözümün yerel optimum noktalara takılarak düştüğü tuzığa düşmeme olasılığını artırır.

- (ii) GA uygunluk fonksiyonu gradyan tabanlı fonksiyonlardan farklı olarak birçok değişik şekilde olabilir ve bu farklı fonksiyonlar eşzamanlı olarak değerlendirilebilir.

Leu ve diğ. (1994), Killbridge-Wester (1961) problemini GA yardımıyla çözdü ve bu çözümü QS yazılım paketinde (Chang ve Sullivan, 1991) varolan beş farklı sezgisel yöntemle kıyasladı.

Orijinal problemin çevrim süresi 55 olmasına rağmen, Leu ve diğ. (1994), GA tabanlı olmayan bu sezgisel yöntemlerin problem kısıtlarındaki değişimlere duyarlılığını gözlemlemek üzere çevrim süresini 56 olarak aldı. Yazarlar aynı zamanda bu beş sezgisel yöntemi, dört farklı performans ölçütü kullanarak kıyasladı:

- i) Âtıl zamanların karelerinin ortalaması (varyans)
- ii) Âtıl zaman standart sapması
- iii) Hat etkinliği
- iv) En büyük istasyon süresi

Leu ve diğ. (1994), GA'nın daha iyi bir başlangıç popülasyonu ile çözüme başlamasının, elde edilecek sonucun kalitesini önemli ölçüde artırdığını göstermiştir.

İkinci çalışmada, Anderson ve Ferris (1994), GA'nın kombinatoriyal optimizasyon problemlerinde etkili bir şekilde kullanılabileceğini, MHD problemi üzerinde çalışarak gösterdi. Araştırmacı, ilk önce, algoritmanın kurallara uygun bir biçimde MHD problemleri için uygulanmasını açıkladı. Daha sonra algoritmaya paralel farklı bir versiyonu tanımlandı. Burada amaç, algoritmanın diğer geleneksel yöntemlerden üstün olduğunu kanıtlamak değildi, salt optimizasyon problemlerinde potansiyel kullanıma yönelik birtakım göstergeler vermektir. Bu yüzden GA'nın diğer sezgisel yöntemlerle karşılaştırılması yapılmamıştır.

Suresh (1996), GA'nın montaj hattı dengeleme problemlerinin stokastik yapıdaki tek modellenmiş durumunu incelemiştir. GA'nın en temel yeteneklerinden biri de, her çeşit amaç fonksiyonunu hesaba katabilmesidir.

Düzenli olmayan arama uzayının da hesaba katılabilmesi için, yeniden düzenlenmiş iki popülasyona yer verilen GA önerilmiştir. Bunlardan biri, olurlu olmayan (infeasible) popülasyon (öncelik ilişkilerine bağlı olarak) ve belirli aralıklarla bu türlerin değişimi işlemleridir. Araştırmacı, olurlu bir çözüm popülasyonunun

bölünmüş bir arama uzayına bizleri yönlendirdiğini iddia eder, böylece bu da yerel en küçük değerlere takılma olasılığını artırır. Olurlu olmayan çözümlere ancak genetik operatörlerin onları olurlu çözüme götürdüğü zaman izin verilebileceğini vurgular. Tümüyle olursuz bir çözümün, MHD probleminde olurlu çözüme gidecek şekilde yönlendirilemeyeceğini anlatır.

İki seçenek popülasyonun (biri tümüyle olurlu kromozomlardan oluşurken, diğeri belli bir yüzdede olurlu olmayan kromozomlara sahip) her biri, kendine göre avantajlarını kullanmak üzere kontrollü bir şekilde birleştirilir. Belli aralıklarla kendi popülasyon havuzlarında aynı uygunluk değerine sahip bazı kromozomlar yer değiştirilir. Bu çalışma, bu tür iki popülasyonlu çalışmanın, olurlu çözümlerin olduğu tek bir popülasyonla çalışmadan daha iyi sonuçlar verdiğini göstermiştir.

Yüksek hacimli problemlerin çözümüne yönelik tek modelli MHD çalışmaları birçok uygulanabilir algoritmanın üretilmesini sağlamıştır. Bu çözüm algoritmalarının birçoğu, ilgili probleme yönelik, teknolojik önceliklerden önce hesaba katılması gereken kısıtların ve iş istasyonlarına farklı atama tercihlerinin olduğu, gerçek bir tasarımdaki seçenek çözümlerin varlığına rağmen tek bir çözüm elde etmektedir.

Dar-El ve Rubinovitch tarafından önerilen MUST algoritması ise, çözümdeki bu farklılığı sağlayan varolan birkaç algoritmadan biridir. Sezgisellik, MUST algoritmasını göreceli olarak daha büyük ölçekli hat dengeleme problemlerinin çözümünde etkin kılmaktadır (Rubinovitz ve diğ., 1995). Rubinovitz'in yaptığı çalışmada montaj hattı dengeleme problemlerinde çoklu çözüm üretimine olanak veren bir genetik algoritmanın denenmesi ve gelişimi anlatılmıştır. Farklı sınıflardaki problemlerde elde edilen sonuçlar, MUST algoritmasındaki sonuçlarla kıyaslanmıştır.

Genetik algoritmanın basit yerel bir arama prosedürü ile birleştirilmesi, iyi sonuçlar elde edilmesini sağlamıştır. Bu prosedür, yüksek miktardaki istasyon sayılı ve yüksek esneklik oranının olduğu problemlerde MUST algoritmasından daha hızlı çalıştı. MHD problemlerinin çözümünde tavsiye edilmek üzere farklı mutasyon ve çaprazlama prosedürleri denenerek değerlendirildi.

GA'nın MHD problemlerine uygulanma aşamasında farklı bir yaklaşım da Türkiye'den Sabuncuoğlu ve Erel (2000)'den gelmiştir. Bu çalışmada, iş ögesi

süreleri deterministik olan tek modelli MHD problemlerinin çözümünde kullanılmak üzere problemin gerçek karakteristiklerini esas alan yeni bir GA önerilmiştir. MHD çözüm uzayını etkili bir şekilde aramaya yönelik kromozom yapısının dinamik olarak değiştirilebileceği gösterilmiştir. Dikkate değer bir şekilde arama prosedürü zarfınca hem çözüm kalitesinde hem de işlem süresinde gelişmeyi sağlayacak şekilde kromozom hacmini düşüren dinamik bölümlene ortaya atılmıştır. Bu dinamik kromozom yapısı GA gruplama problemlerine uygulanabilecek yeni bir kavram olarak öne çıkmıştır. Bunun yanı sıra SA (Simulated Annealing) düşüncesinden hareketle yeni bir elitizm yapısı benimsendi. Bu yeni elitizmin GA performansına önemli ölçüde katkıda bulunduğu görüldü. 45 iş ögesi Killbridge-Wester (1961) MHD problemi üzerinde, önerilen GA ile Leu (1994)'nın GA çözümü ve Baybars (1986)'ın sezgisel yöntemi kıyaslanmıştır. Aynı zamanda Tonge (1961)'un 70 iş ögesi problemi üzerinde diğer geleneksel sezgisel yöntemlerle karşılaştırması da yapılmıştır (Sabuncuoğlu ve diğ., 2000).

DB etkinliğini araştırmak üzere Leu (1994)'nın ele aldığı probleme benzer 30 farklı MHD problemi oluşturulmuştur. Buna ek olarak çözüm üzerindeki farklı DB ve GA parametrelerinin etkinlikleri de ölçülmüştür.

Her biri 50 iş ögesi içeren 30 problem ve bunlar arasındaki öncelik ilişkilerinin göstergesi olan üç farklı esneklik oranı (%10, %50 ve %90) gözönünde bulundurularak rassal olarak oluşturulmuştur.

Yapılan geniş kapsamlı deneysel hesaplamalar, GA'nın, literatürde iyi bilinen sezgisel algoritmaların performansını gösterdiğini kanıtladı.

1998 yılında Xu (1998) tarafından yapılan çalışmada ise MHD problemleri için farklı bir GA yöntemi önerilmiştir. Burada MHD problemlerinin klâsik sıralama kısıtı, yönlendirilmiş bir grafik ve basitçe 0-1 uygunluk matrisiyle tanımlanmıştır. Çözümler permütasyon matris formundadır. Arama uzayını indirgemek için olurlu çözüm ayırdetme kuramı ileri sürülmüştür. Bu şekliyle algoritmayı esnek iş ögesi süreleri gibi MHD problemlerinin diğer formlarına uydurmak kolaylaşmıştır (Xu ve diğ., 1998).

Bir diğer çalışmada ise kromozom temsilinde 10 tabanlı tamsayı kodlaması kullanılmıştır, tek noktalı çaprazlama ve düzgün çaprazlama operatörlerine yer

verilmiştir. Katı öncelik ilişkileri ve çevrim süresi kısıtlarına bağlı olarak başlangıç popülasyonunda olurlu bir çözüm elde etme olasılığı oldukça düşük olduğundan, başlangıçta rassal ve olurlu çözüm elde etmek üzere etkin bir yordam geliştirilmiştir. Sözedilen kısıtlardan dolayı çaprazlama ve mutasyon işlemleri sonucunda da olursuz çözüm elde edilmektedir. İşte bu olursuz çözümleri olurlu çözüme dönüştüren bir yöntem geliştirilmiştir. İşte bu olurlu başlangıç çözümü ve onarım yöntemi birbiriyle bağlı olarak önerilen GA, 21 iş öğeli bir problem üzerinde denenerek doyurucu sonuçlara ulaşılmıştır (Chan K. ve diğ., 1996).

MHD problemlerinin GA'larla çözümüne yönelik tanıtıcı ve tekstil sektöründeki uygulamasına yer verilen literatür araştırmasını yaptığımız “Handling the Assembly Line Balancing Problem in the Clothing Industry using A Genetic Algorithm” başlıklı makalede ise GA'nın ayrıntıları sunulmuş olup, montaj hatlarının tekstil endüstrisindeki ana karakteristikleri ele alınmıştır. Problemlerin genetik algoritmalarla çözülebilmesi için nasıl formüle edileceği açıklanmıştır. Elde edilen sonuçlar, GA yaklaşımının, birçok fabrika yöneticisinin MHD probleminin çözümünde kullandığı bilinen algoritmalarından daha iyi olduğunu ortaya koymuştur (Kelth C. ve diğ., 1998).

Çevrim süresinin belirli olduğu basit MHD problemleri için melez bir genetik algoritma yapısı önerilen çalışmada, problemin kromozom temsili rassal olarak belirlenmiştir. Operasyonların iş istasyonlarına atanması, operasyon önceliklerinin kromozomlar tarafından belirlendiği sezgisel bir öncelik kuralıyla yapılmıştır. Çözümü geliştirmek üzere yerel bir arama yöntemi kullanılmıştır. Önerilen çözüm yaklaşımı, test edilmek üzere literatürden alınan problemler kullanılarak diğer yöntemlerle kıyaslanmıştır. Elde edilen sonuçlar, önerilen melez genetik algoritmanın etkinliğini kanıtlamıştır (Goncalves ve diğ., 2002).

Schaffer J. (1996)'da yapılan Philips End. Elektronik Şirketinde yapılan çalışmada ise, firmanın pazarladığı FCM yüzey ağızlama aygıtı montaj robotları tanıtılarak, buradaki uygulamasına yer verilmiştir. Bu aygıtla, saatte 60 bileşen gibi yüksek bir çıktı yeteneğine sahip olmalarına rağmen, verilen herhangi bir ürün için bunların aynı performansı verecek şekilde konfigüre edilmesi gerekir. Bu kurulum problemi kombinatoryal zor bir problemdir. İşte bu problemin üstesinden GA ile gelinmiştir. Aynı zamanda keşfedilen plân ile genel anlamda bu tür kombinatoryal problemler

için GA kullanımıyla başarılı sonuçlara ulaşılabileceği gösterilmiştir (Schaffer ve diğ., 1996).

Bilindiği üzere, genel doğrusal hat şeklindeki montaj hatlarının yanısıra pratikte U tipi dediğimiz hatlar da vardır. İki yönlü operatör atamaya izin veren bu tür hatlarda dengeleme çalışması, konum kısıtları da gözönünde bulundurularak yapıldığından farklı olmaktadır. GA'ların bu yapıdaki hatlara uygulanmasına yönelik ilk çalışma, Debora A. (2000) tarafından yapılmıştır. Yapılan araştırma, toplam aylak sürenin enküçüklenmesi, istasyonlar arasındaki iş yükünün dengelenmesi ya da her ikisinin aynı anda sağlanması istendiğinde, problemin iki olası varyasyonuna yönelik genel bir çerçeve çizmektedir.

Kromozom yapısını yorumlayan ve öğeleri istasyonlara atayan altı farklı algoritma geliştirilmiştir. Sonuçlar, GA'nın mükemmel bir teknik olduğunu göstermiştir. Literatürden alınan 61 problem üzerinde yapılan testlerden 49'unda önceki yöntemlerde bulunan sonuçlara, 11'inde de daha üstün optimum sonuçlara ulaşılmıştır. Yalnızca bir durumda daha kötü bir sonuç bulunmuştur. Bununla birlikte GA, hesaplama açısından etkin olduğunu kanıtlamıştır (Debora, A., 2000).

6.3. Çok Amaçlı MHD Genetik Algoritma Çalışmaları

Genetik Algoritma yapısı itibariyle uygunluk fonksiyonunun çok amaçlı tanımlamaya elverişli olması çok önemli bir avantajdır.

Çok Amaçlı MHD problemlerine GA yaklaşımı çözüm çalışmalarının bilinen ilk örneği Kim Y.K. (1996) tarafından sunulmuştur. Bu çalışmada hedeflenen çeşitli amaçlar şöyledir:

- İş istasyonu sayısının enküçüklenmesi
- Çevrim süresinin enküçüklenmesi
- İş yükü düzgünlüğünün enbüyüklenmesi
- İş ilişkililiğinin enbüyüklenmesi
- Son iki amacın birleşik düşünülmesi

Özet olarak, önerilen GA'nın temel esasları tartışılmıştır. Kromozom kodlama, decoding ve genetik operatörlere vurgular yapılmıştır. Geleneksel GA'yı MHD problemlerinin birçok çeşidine esnek olarak uyumlandırılabilen bir düzenleme yöntemi geliştirilmiştir. Çok amaçlı MHD problemleri için farklı bir Pareto optimal çözüm kümesi araştırmasına önem verilerek yer ayrılmıştır. Yapılan geniş içerikli çalışmalar raporlanmıştır. Bilinen sezgisel algoritmalar ve önerilen GA'nın performansının kıyaslanması sonucunda GA yaklaşımının geleceğinin parlak olduğu görülmektedir (Kim Y. K. ve diğ., 1996).

Konuyla ilgili yapılan ikinci çalışmada çok amaçlı genetik algoritma ile MHD problemlerinin çözümüne yönelik bir yöntem önerilmiştir. Performans ölçütleri olarak istasyon sayısı, hat etkinliği, takastan önceki düzgünlük indeksi, takastan sonraki düzgünlük indeksi değerleri alınmıştır.

Geliştirilen genetik algoritma, literatürden bilinen altı popüler sezgisel yöntemle karşılaştırılmıştır:

- Konum ağırlıklı dengeleme tekniği
- Killbridge-Wester yöntemi
- Moodie-Young yöntemi
- Hoffman öncelik matrisi yöntemi
- “İlk uyanı anında güncelle” yöntemi
- Sıra ve atama yöntemi

Daha göreceli bir değerlendirme yapılması açısından, literatürden alınan 20 farklı problem ele alındı ve her birinde farklı beş çevrim süresi kullanıldı. Sezgisel yöntemler ve genetik algoritma C++ dilinde kodlandı. Belirlenen performans ölçütlerine göre genetik algoritmanın diğer sezgisel yöntemlere göre daha iyi sonuçlar verdiği görüldü. Her ne kadar genetik algoritmanın çalışma süresinin diğerlerine oranla fazla olması dezavantaj gibi görünse de, bunun nedeni GA'nın global optimum çözümü daha fazla iterasyon kullanarak aramasıdır (Ponnanbalam ve diğ., 2000).

Son çalışma ise, çok amaçlı melez MHD problemlerine hitâp etmektedir. Buradaki amaç, iş öğelerini iş istasyonlarına atamak ve uygun ekipman seçimlerini yerine

getirmektir. Hedef, hattın toplam mâliyetini; istasyon alanı, ekipman mâliyeti, tasarım kısıtı, çevrim süresi, öncelik ilişkileri gibi operasyonel kısıtlarla birlikte bütünleşik düşünerek enküçükmektir. Problemin üstesinden gelecek dal-sınır algoritması ve PROMETHE II adı verilen çok amaçlı karar destek yöntemi ile melezleştirilmiş bir genetik algoritma kullanılmıştır. Yönteme ilişkin kabul edilmiş gerekli kavramların tanımı yapılmış olup, algoritmanın bir uygulama çalışması üzerindeki sonuçları açıklanmıştır (Rekiek B., 2001).

6.4. Simülasyon Esaslı Genetik Algoritma Çalışmaları

Simülasyon her ne kadar montaj hatlarının modellenmesinde kullanılsa da, hat operasyonlarının optimize edilmesine uygun değildir. Optimizasyon tekniklerinden biri olan GA, hattın etkinliğinin ve verimliliğinin artırılmasında uygulanabilir bir yöntem olarak öne çıkmıştır. Bu çalışmada seri bir kompresör montaj hattının simülasyon yoluyla modellenmesi ve hat etkinliğinin GA kullanılarak optimize edilmesi anlatılmıştır. Popülasyon büyüklüğünün, çaprazlama ve mutasyon olasılıklarının ve kromozomların, algoritmanın performansına olan etkisi incelenmiştir. Yapılan uygulama analiz sonuçları, hattın iyi dengelendiğini ve bazı makinaların ortalama çıktıyla çakışmadan daha düşük hızlarda çalışabileceğini göstermiştir. Buna ek olarak makina hızının optimum çıktıda belirleyici olmadığı bulunmuştur. Makina kullanımı % 4,3 oranında iyileştirilmiş, öte yandan gecikme, %23 oranında kompresör bloklarının gelişi yavaşlatılarak iyileştirilmiştir (Lee S. G. ve diğ., 2000).

Sayısı sürekli artan küresel rekâbet firmaları, sürekli olarak üretim hatlarının performansını iyileştirmeye zorlamaktadırlar. Her ne kadar bu gelişme, karmaşık üretim sistemlerinde pek kolay olmamakta ise de, bu kategoride yapılan ikinci çalışmada, simülasyon tabanlı zekî bir üretim hattı dengeleme şeması sunulmuştur.

Önerilen yöntem, genetik algoritma bütünleşimli bilgisayar simülasyonuna optimal çözüm üretmek üzere yer vermektedir. Gerçek yaşamdan alınan bir üretim problemine önerilen çözüm, belirtilen performans indisleriyle optimal bir konfigürasyon sağlayabilmektedir. Gerçek bir matris kromozom kodlaması ve parabolik seçim yöntemi anlatılmıştır. Ayrıntılı bir Extend TM simülasyon

programına bütünleştirilmesi anlatılmış olup gerçek yaşamdan alınmış bir problem üzerinde de uygulama çalışması yapılmıştır (Zhao Z. Y. ve diğ., 2000).

6.5. Bulanık Mantık Tabanlı Genetik Algoritma Çalışmaları

Bilindiği üzere MHD problemleri, zor kombinatorial optimizasyon problemleri sınıfına girmektedir. Doğrusal ve dinamik programlama yaklaşımlarıyla çözülmeye çalışılan bu problemler, istenen etkinlikte değildir. Sözedilen GA'lar, sıralama, gezgin satıcı, ulaşım problemleri vb. gibi büyük ölçekli problemlerin çözümünde etkin olarak kullanılmaktadır. Bulanık küme kuramı ise çoğunlukla bilginin kesin olmadığı, belirsiz olduğu durumları temsil etmekte kullanılır.

Tsijumira (1995)'nın yaptığı çalışmada, gerçek dünyada yer alan problemlere ilişkin bilgiler bu bağlamda ele alınarak, işlem sürelerini temsil etmede bulanık sayılar kullanılmıştır ve bu yapıdaki bir problemin GA ile çözümünün performansı artırdığı gösterilmiştir (Tsijumira ve diğ., 1995).

Bu konuda 1996 yılında yapılan ikinci çalışmada ise NP-hard problemler sınıfına giren bulanık MHD problemleriyle sentezlemek üzere GA geliştirilmiştir. Genetik operatörlerin kromozom olurluluğu ile ilgisi konu edilmiş olup, ilgili çözümün performansı sayısal bir örnek ile gösterilmiştir (Gen M. ve diğ., 1996).

7. UYGULAMA

7.1. Giriş

MHD uygulama çalışmasına konu olan tek model dengeleme problemi ARÇELİK-LG Gebze Klima İşletmesi'nde dış ünite bandına ilişkin son hat üzerinden alınmıştır. Amacımız, genetik algoritmanın bulacağı çözüm ile varolan sistem arasında performans kıyaslaması yaparak genetik algoritma çözümünün etkinliğini ortaya koymaktır.

Problem, genetik algoritma çözüm mantığıyla çalışan C++ dilinde kodlanmış Optiline isimli Belçika orijinli bir yazılımda formüle edilerek çözülmüştür.

7.2 Firma Tanıtımı

Arçelik-LG, Arçelik A.Ş. ve Güney Kore kökenli LG Electronics ortaklığıyla Gebze Organize Sanayi Bölgesinde 50 milyon \$ yatırımla 2000 yılında kurularak faaliyete geçmiştir. Arcelik-LG tesisleri yıllık 300.000 adet klima üretim kapasitesine sahiptir. Kurulan ortaklığın sermâye yapısı, Türk ve Güney Koreli ortaklar arasında eşit ortaklık biçimindedir.

Türkiye klima pazarının % 50'sine sahip olan Arçelik LG klima, ülkemizin tek bütünsel klima fabrikasına sahiptir. 2002 yılında Arçelik LG toplam üretiminin % 30'unu 8 farklı ülkeye ihraç etmiştir.

Kuruluş tarihi 13 Mayıs 1999 olan Arçelik-LG Gebze Klima İşletmesi; elektrikli ev âletleri, özellikle klima aygıtları üretimi, montajı ve sözkonusu ürünlerin Türkiye ve yurtdışında pazarlanması, satışı ve dağıtımı ile ilgilenmektedir. Başlıca ürünü, ev tipi klima aygıtlarıdır. Genel Müdürlük binası Gebze/Kocaeli'nde olan işletme, LG Electronics (Güney Kore) Firması ile %50 ortak olarak kurulmuştur. Şirketin 2001 yılı cirosu 56.613.746.000.000 TL olup, 2002 yılındaki toplam personel sayısı

163'tür. Bu personelin 12'si yönetici, 51'i memur ve 100'ü de işçidir. Şirket, TSE ISO 9001 kalite belgesine sahiptir. Fabrikanın montaj hattı görüntüleri, Şekil 7.1.'de verilmiştir.



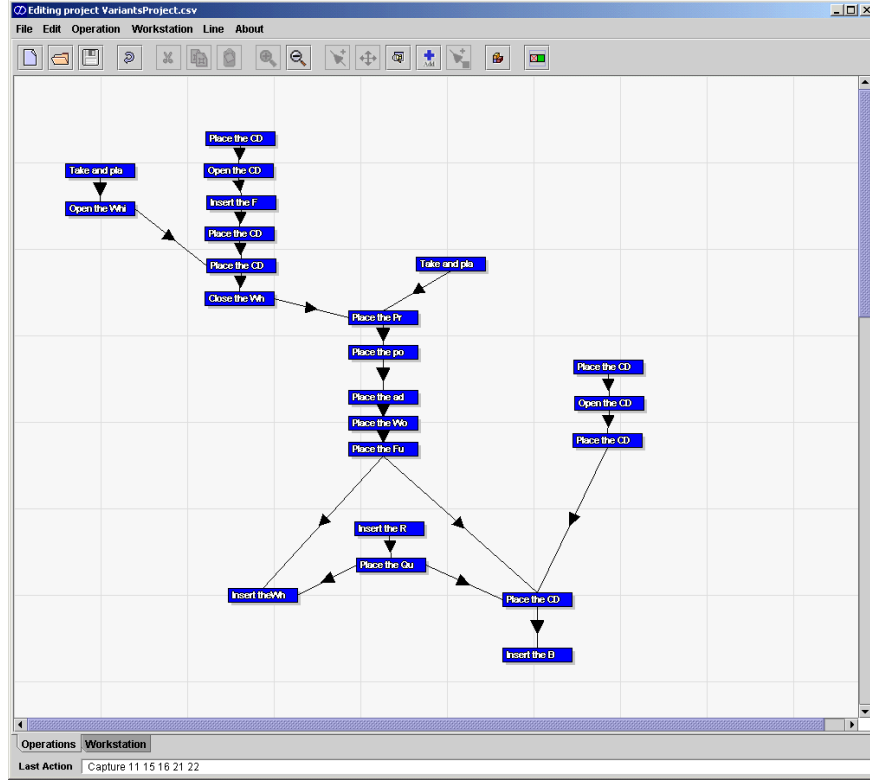
Şekil 7.1. Arcelik-LG fabrika ve montaj hattı görüntüleri

7.3. Genetik Algoritmali Optiline Yazılım Tanıtımı

Optiline, tek ve çok modelli montaj hatlarının optimizasyonu için geliştirilmiş grafik kullanıcı arabirimi ve gereksinim duyulan verileri yönetebilmek için gerekli tüm araçları içeren bütünleşik bir programdır. Belçika kökenli “OptimalDesign” adlı yazılım firmasının ürünü olan Optiline, genetik algoritmaları kullanarak çözüme gitmektedir. Program, C++ dilinde kodlanmış olup daha ayrıntılı bilgi almak için <http://www.optimaldesign.com> adresine başvurulabilir.

Program, tek modelli hat dengelemenin yanısıra, aynı ürünün birden çok varyantının üretilmesi durumu için de hat dengeleme yapabilmektedir. Bu durumda ilgili ürünlerin montaj hattındaki akışına ilişkin % dağılım parametresi girilmelidir.

İş öğeleri arasındaki öncelik kısıtlarının yanısıra, programın diğer bir üstünlüğü de “Zoning Constraints” denilen istasyonların bölgesel alan kısıtları da gözönünde bulundurularak çözüme gidebilmesidir. Aynı zamanda işin niteliğinden dolayı belirli bir istasyona atanma zorunluluğu olan iş öğeleri de program içinde sabitlenerek tanımlanabilmektedir. Bir örnek durum için öncelik diyagramı, Şekil 7.2.'de verilmiştir.



Şekil 7.2. Optiline üzerinde öncelik diyagramı oluşturulmuş bir montaj hattı şeması

Programda genel olarak aşağıdaki adımlar izlenerek çözüme gidilir:

- Ürün montaj süreçlerinin tanımlanması
- İş öğelerinin süreç türü (İnsanlı, Otomatik, Robotlu) ve süreleri
- Öncelik ilişkilerinin grafiksel olarak oluşturulması

Daha sonra ise aşağıda belirtilen parametrelerin bir kısmı Şekil 7.3.'de görülen diyalog kutusunda görüldüğü üzere programda girilir. Girilen parametre değerlerinin yüksek olmasıyla orantılı olarak çalışma süresi de uzamaktadır. Popülasyon büyüklüğünün 32, kuşak sayısının 50 olduğu bir problemde ortalama çözüme ulaşma süresi 35,62 sn'dir. Parametreler şunlardır:

- Çevrim süresi (cycle time)
- İş istasyonu sayısı (station number)
- Popülasyon büyüklüğü (population size)
- Kuşak sayısı (max age)

Optimize Line

MP / Balance Percentage: 100.0 %

100 % (Balance) 100 % (Max Peak)

Total WorkStation Time: 7500

Total Operation Time: 5217

Advanced Parameters

Population Size: 16

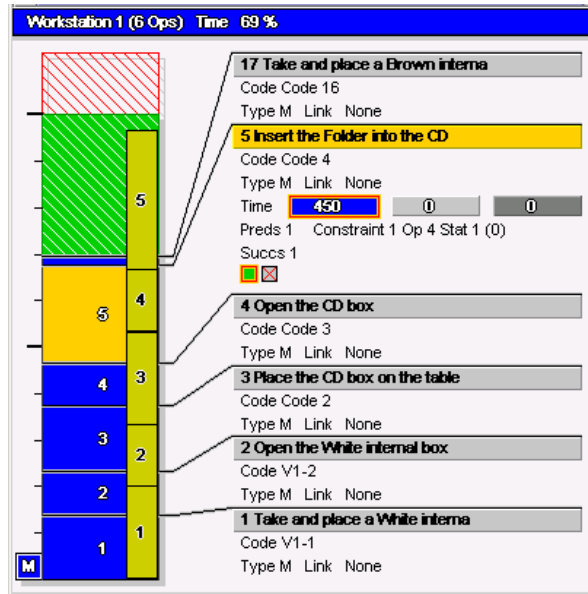
Max Age: 50

Include Current Solution: ☒

Cancel Optimize

Şekil 7.3. Optiline hat optimizasyonu diyalog kutusu

Dengeleme sonucu elde edilen çözümdeki istasyonlara atanan iş öğeleri Şekil 7.4.'de görüldüğü üzere grafiksel olarak ayrıntılı şekilde gösterilmektedir. İstasyonlara atanan iş öğelerinin adları, süre değerleri, tipi, istasyona ilişkin % denge kaybı bilgileri, buradan görsel olarak izlenebilmektedir.



Şekil 7.4. İstasyonlara atanmış iş öğelerinin grafiksel gösterimi

7.4. Uygulama Problemi

7.4.1. Problemin Tanımı

Genetik algoritma çözüm performansını kıyaslamak amacıyla Arcelik-LG Klima İşletmesi'nden dış ünite bandına ilişkin 41 iş öğeli bir tek modellenli deterministik iş ögesi süreli MHD problemi ele alınmıştır. Probleme ilişkin iş ögesi tanımları, süreleri, öncelik ilişkileri ve Arcelik-LG'de varolan durumdaki iş istasyonlarına atanmış değerleri, Tablo 7.1.'de görülmektedir.

Tablo 7.1. Dış ünite bandı iş öğeleri, öncelikler, süreler ve varolan durumdaki atama

Etkinlik No	Etkinlik Tanımı	Önceki Etkinlik Kodu	İşlem Süresi (dk)	Varolan Dengelemede İstasyon No
1	1 AD ARABÖLME SACI AL, YERİNE TAK	YOK	0,08760	1
2	ARABÖLME SACINA 2 VİDA AT	1	0,14400	1
3	FAN MOTOR KABLOSUNU SACDAN GEÇİR	1	0,07330	2
4	1 AD FAN AL, YERİNE TAK	1,3	0,15000	2
5	FANI 1 AD SOMUNLA SABİTLE	4	0,06660	2
6	1 AD İKAZ ETİKETİ AL, FANA YAPIŞTIR	4,5	0,08330	2
7	1 ADET BUTİL AL, KOMPRESOR BORUSUNA YAPIŞTIR	1,2	0,25110	4
8	BORULARIN GAZ KAÇAK KONTROLUNU YAP	7	0,27500	3
9	1 AD KONTROL KUTUSU AL, KABLOLARI DÜZENLE PALETE BIRAK	6,8	0,14860	5
10	1 AD BUTİL AL, EMİŞ BORUSUNA YAPIŞTIR	1	0,04100	3
11	KOMPRESÖR TERMİNALİ 3 ADET SOKET BAĞLANTISINI YAP	9	0,14970	5
12	TERMİNAL KAPAĞINI YERİNE TAK	11	0,07350	6
13	KAPAĞI SOMUNLA SABİTLE	12	0,13270	6
14	KAPILERİ BORU GRB NA BİTİL SAR	1	0,05600	4
15	VANA BOBİN GRUBUNU TAK	13	0,14270	7
16	KOMPRESÖR KECESİNİ SAR	10,14,15,18	0,03210	8
17	KONTROL KUTUSUNA 1 VİDA AT	15	0,11670	6
18	FAN MOTOR SOKETİNİ TAK	11	0,05500	7
19	KABLOLARI DÜZENLE	17	0,11200	7
20	VAKUM KAPLINLERİNİ SÖK	16,19	0,32560	8
21	ARKA GOVDE İZOLASYONLARINI YAPISTIR	19	0,35000	9
22	ARKA GOVDEYİ TAK	21	0,21220	10
23	ARKA GOVDE VİDALARINI AT	22	0,14760	10

Tablo 7.1. (Devamı)

Etkinlik No	Etkinlik Tanımı	Önceki Etkinlik Kodu	İşlem Süresi (dk)	Varolan Dengelemede İstasyon No
24	KONTROL KUTUSUNU OTURT	22	0,02100	10
25	ÖN GOVDE İZOLASYONLARINI YAPIŞTIR	23,24	0,31240	11
26	ORİFİS AL, YERİNE TAK	25	0,02880	12
27	İZGARAYI 6 VİDA İLE TUTTUR	26	0,32250	12
28	ÖN GOVDENİN VİDALARINI AT	25	0,31250	13
29	ÖN GOVDEYE 1 ADET MARKA ETİKETİ YAPIŞTIR	27,28	0,25120	14
30	2 ADET PİRİNÇ TAPA TAK	20	0,28000	15
31	1 AD TİP ETİKETİ AL YERİNE YAPIŞTIR	29	0,15860	16
32	1 AD KÜÇÜK BARKOD ETİKETİ AL, TİP ETİKETİNİN ALTINA YAPIŞTIR	31	0,12000	16
33	1 AD BÜYÜK BARKOD ETİKETİ AL, YERİNE YAPIŞTIR	31	0,09800	16
34	1 AD ALT STRAFOR VE KARTON AL, BANTLAYARAK GRUPLA	32,33	0,31010	17
35	ALT STRAFORA 1 AD LASTİK 4 ADET TAHLİYE PLASTİĞİ KOY	34	0,16220	18
36	ALT STRAFORU YERİNE YERLEŞTİR, PEDALA BAS	30,35	0,10060	18
37	1 AD KONTROL KUTUSU KAPAĞI AL, 1 AD VİDA İLE YERİNE YERLEŞTİR	36	0,22030	19
38	1 AD ÜST STRAFOR GRUBU AL, YERİNE YERLEŞTİR	37	0,08550	20
39	AMBALAJ KARTONUNU TAK, KATLA.	38	0,08550	20
40	AMBALAJ KARTONUNU 4 ADET ZİMBALA İLE ZİMBALA	39	0,07250	20
41	BARKOD ETİKETİNİ YAPIŞTIR	40	0,06720	20
İŞ ÖĞESİ SÜRELERİ TOPLAMI ($\sum t_i$)			6,2352 dk	374,112 sn

Probleme konu olan montaj hattı sistemi, şu anda varolan durum itibariyle 23 sn çevrim süresi, 20 iş istasyonu ve %18,67 denge kaybıyla çalışmaktadır.

Burada amacımız, iş öğelerinin öncelik kısıtları ve çevrim süresi kısıtı altında denge kaybını enküçükleyecek, en üst düzeyde etkin bir hat tasarımı için gerekli istasyon sayısı ve bu istasyonlara hangi iş öğelerinin atanması gerektiğinin GA ile bulunarak sistemde optimizasyon yapılması ve çıkan sonuçların varolan sistemle kıyaslanmasıdır.

7.4.2. Varolan Durum Analizi

Bu hatta ilişkin bazı değerler, aşağıda hesaplanmıştır:

$$\text{En büyük Hat Etkinliği: } HE_{enb} = \frac{\sum t_i}{n.C} = \frac{374,112}{17 * 23} = 0,95$$

$$\text{Gerekli En Az İş İstasyonu Sayısı: } n_{enk} = \frac{\sum t_i}{C} = \frac{374,112}{23} = 16,26 \cong 17$$

$$\text{Varolan Durumdaki Hat Etkinliği: } HE_{mevcut} = \frac{\sum t_i}{m.C} = \frac{374,112}{20 * 23} = 0,81 = \% 81$$

$$\text{Varolan Durumdaki Denge Kaybı: } D = \frac{m.C - \sum t_i}{m.C} = \frac{20 \times 23 - 374,112}{20 \times 23} = \%18,67$$

m = Varolan iş istasyonu sayısı = 20

C = Çevrim Süresi = 23 sn/adet

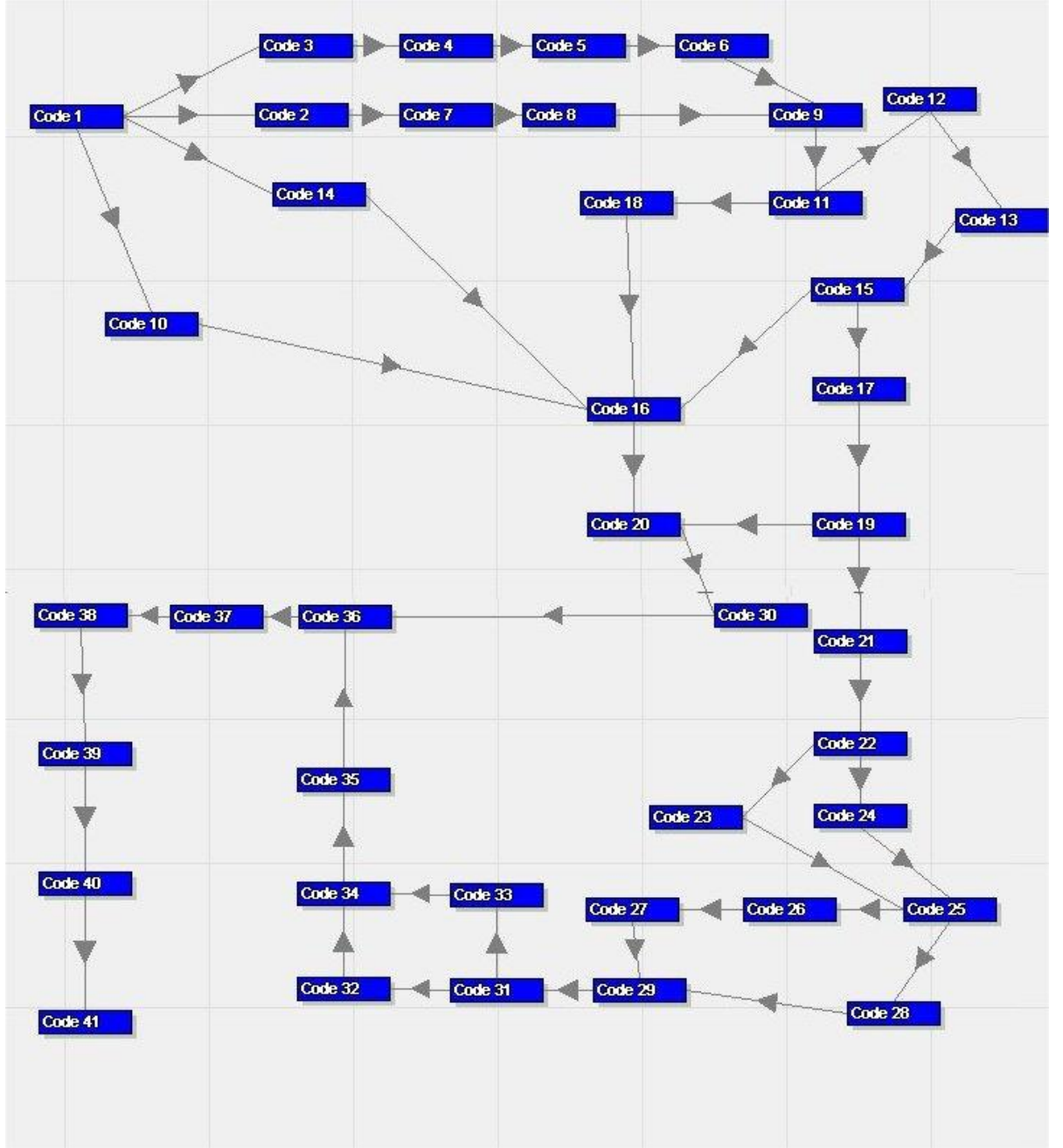
Varolan durumdaki istasyon atamasına ilişkin kayıp süreler, Tablo 7.2.'de görülmektedir.

Tablo 7.2. Varolan durumdaki istasyon atamasına ilişkin kayıp zamanlar

İstasyon No	İstasyon Süresi (sn)	Kayıp Süre (sn)
1	13,90	9,10 sn
2	22,39	0,61 sn
3	18,96	4,04 sn
4	18,43	4,57 sn
5	17,90	5,10 sn
6	19,37	3,63 sn
7	18,58	4,42 sn
8	21,46	1,54 sn
9	21,00	2,00 sn
10	22,85	0,15 sn
11	18,74	4,26 sn
12	21,08	1,92 sn
13	18,75	4,25 sn
14	15,07	7,93 sn
15	16,80	6,20 sn
16	22,60	0,40 sn
17	18,61	4,39 sn
18	15,77	7,23 sn
19	13,22	9,78 sn
20	18,64	4,36 sn
TOPLAM		85,88 sn

7.4.3. Problemin Öncelik Diyagramı

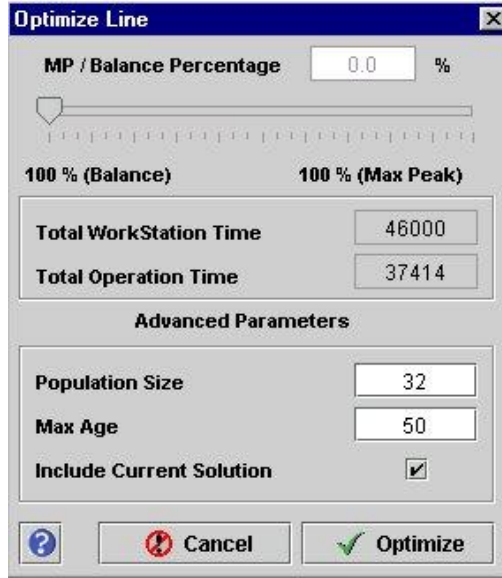
Probleme ilişkin 41 iş ögeli montaj hattının öncelik diyagramı, Optiline üzerinde Şekil 7.4.'de görüldüğü gibi tanımlanarak çizilmiştir.



Şekil 7. 4. Arcelik-LG dış ünite montaj hattı öncelik diyagramı

7.5. Problemin Optiline İle Çözümü

Probleme ilişkin öncelik diyagramı, çevrim süresi ve istasyon sayısı gibi parametreler girildikten sonra Optimize Line diyalog kutusunda genetik algoritmaya ilişkin popülasyon büyüklüğü ve kuşak sayısı gibi değerler de girilerek problemin çözümüne geçilir (Şekil 7.5.).



The image shows a software dialog box titled "Optimize Line". It contains several input fields and checkboxes. At the top, there is a field for "MP / Balance Percentage" set to "0.0 %". Below this is a horizontal slider bar. Further down, there are two columns of data: "100 % (Balance)" and "100 % (Max Peak)". Under "100 % (Balance)", there are two fields: "Total WorkStation Time" with the value "46000" and "Total Operation Time" with the value "37414". Below these is a section titled "Advanced Parameters" which contains three fields: "Population Size" set to "32", "Max Age" set to "50", and a checkbox for "Include Current Solution" which is checked. At the bottom of the dialog are three buttons: a help button (question mark icon), a "Cancel" button, and an "Optimize" button with a green checkmark icon.

Şekil 7.5. Probleme ilişkin GA parametreleri veri giriş penceresi

Program; varsayılan değer olarak genelde popülasyon büyüklüğünü 32, kuşak sayısını da 50 olarak aldığıında optimum sonuçlar vermektedir. Ama genetik algoritma, farklı parametrelerde değişik sonuçlar üretebildiği için değişken parametre değerlerindeki üretilen çözümleri değerlendirmek için Tablo 7.3.’deki senaryolar da çalıştırılarak çözümler elde edilmiştir. Tablo 7.3.’de, tasarlanan senaryolara ilişkin iş öğelerinin istasyonlara atama değerleri verilmiştir. Aynı renklerde olan sütunlar, ilgili senaryoların aynı sonuçları ürettiğine işaret etmektedir.

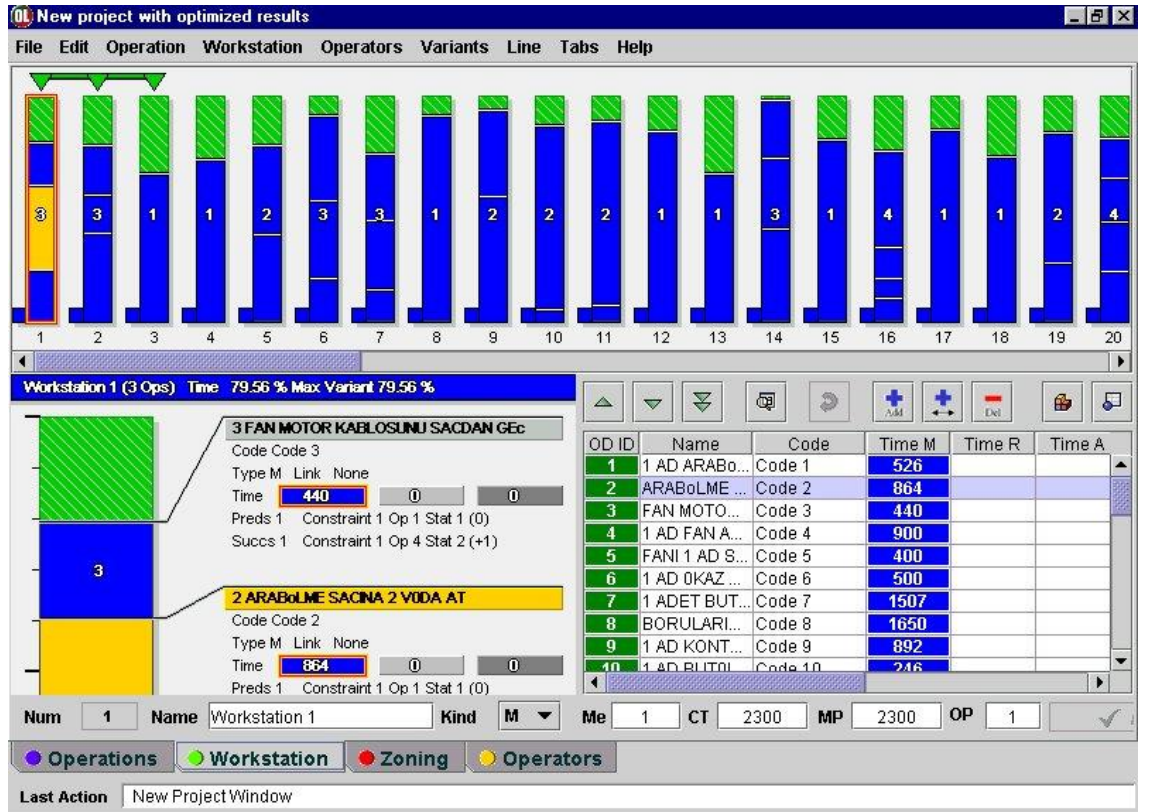
Tablo 7.3. GA farklı parametrelerine göre iş öğelerinin istasyonlara atama sonuçları

İş NO	Etkinlik Tanımı	Genetik Algoritma Parametreleri	Senaryo 1	Senaryo 2	Senaryo 3	Senaryo 4	Senaryo 5	Senaryo 6	Senaryo 7	Senaryo 8	Senaryo 9	Senaryo 10	Senaryo 11	Senaryo 12	Senaryo 13	Senaryo 14
		Popülasyon Büyüklüğü	32	50	100	75	25	60	32	50	100	32	500	20	32	32
		Kuşak Sayısı	50	100	200	100	75	120	80	50	25	150	250	15	50	50
1	1 AD ARABÖLME SACI AL, YERİNE TAK		1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	ARABÖLME SACINA 2 VİDA AT		1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	FAN MOTOR KABLOSUNU SACDAN GEÇİR		1	1	1	1	1	1	1	1	1	1	1	1	1	1
4	1 AD FAN AL, YERİNE TAK		4	3	4	3	4	3	2	3	4	2	4	3	2	2
5	FANI 1 AD SOMUNLA SABİTLE		4	3	4	3	4	3	2	3	4	2	4	3	2	2
6	1 AD İKAZ ETİKETİ AL, FANA YAPIŞTIR		4	3	4	3	4	3	2	3	4	2	4	3	2	2
7	1 ADET BUTİL AL, KOMPRESOR BORUSUNA YAPIŞTIR		2	2	2	2	2	2	3	2	2	3	2	2	3	3
8	BORULARIN GAZ KAÇAK KONTROLUNU YAP		3	4	3	4	3	4	4	4	3	4	3	4	4	4
9	1 AD KONTROL KUTUSU AL, KABLOLARI DÜZENLE PALETE BIRAK		5	5	5	5	5	5	5	5	5	5	5	5	5	5
10	1 AD BUTİL AL, EMİŞ BORUSUNA YAPIŞTIR		7	16	16	16	16	16	16	16	16	16	16	4	16	4
11	KOMPRESÖR TERMİNALİ 3 ADET SOKET BAĞLANTISINI YAP		5	5	5	5	5	5	5	5	5	5	5	5	5	5
12	TERMİNAL KAPAĞINI YERİNE TAK		6	6	6	6	6	6	6	6	6	6	6	6	6	6
13	KAPAĞI SOMUNLA SABİTLE		6	6	6	6	6	6	6	6	6	6	6	6	6	6
14	KAPİLERİ BORU GRB NA BİTİL SAR		10	7	7	7	7	7	7	7	7	7	7	2	7	3
15	VANA BOBİN GRUBUNU TAK		6	6	6	6	6	6	6	6	6	6	6	6	6	6
16	KOMPRESÖR KECESİNİ SAR		10	16	16	16	16	16	16	16	16	16	16	13	16	13
17	KONTROL KUTUSUNA 1 VİDA AT		7	7	7	7	7	7	7	7	7	7	7	7	7	7
18	FAN MOTOR SOKETİNİ TAK		9	16	16	16	16	16	16	16	16	16	16	7	16	7
19	KABLOLARI DÜZENLE		7	7	7	7	7	7	7	7	7	7	7	7	7	7
20	VAKUM KAPLİNLERİNİ SÖK		12	17	17	17	17	17	17	17	17	17	17	14	17	15
21	ARKA GOVDE İZOLASYONLARINI YAPIŞTIR		8	8	8	8	8	8	8	8	8	8	8	8	8	8
22	ARKA GOVDEYİ TAK		9	9	9	9	9	9	9	9	9	9	9	9	9	9
23	ARKA GOVDE VİDALARINI AT		10	9	9	9	9	9	9	9	9	9	9	9	9	9
24	KONTROL KUTUSUNU OTURT		10	10	10	10	10	10	10	10	10	10	10	10	10	10
25	ÖN GOVDE İZOLASYONLARINI YAPIŞTIR		11	10	10	10	10	10	10	10	10	10	10	10	10	10
26	ORİFİS AL, YERİNE TAK		14	11	11	11	11	11	11	11	11	11	11	11	11	11
27	İZGARAYI 6 VİDA İLE TUTTUR		15	12	12	12	12	12	12	12	12	12	12	12	12	12
28	ÖN GOVDENİN VİDALARINI AT		13	11	11	11	11	11	11	11	11	11	11	11	11	11
29	ÖN GOVDEYE 1 ADET MARKA ETİKETİ YAPIŞTIR		16	13	13	13	13	13	13	13	13	13	13	13	13	13
30	2 ADET PİRİNÇ TAPA TAK		14	18	18	18	18	18	18	18	18	18	18	15	18	17

Tablo 7.3. (devamı)

31	1 AD TİP ETİKETİ AL YERİNE YAPIŞTIR		17	14	14	14	14	14	14	14	14	14	14	16	14	14
32	1 AD KÜÇÜK BARKOD ETİKETİ AL, TİP ETİKETİNİN ALTINA YAPIŞTIR		17	14	14	14	14	14	14	14	14	14	14	16	14	14
33	1 AD BÜYÜK BARKOD ETİKETİ AL, YERİNE YAPIŞTIR		17	14	14	14	14	14	14	14	14	14	14	16	14	14
34	1 AD ALT STRAFOR VE KARTON AL, BANTLAYARAK GRUPLA		18	15	15	15	15	15	15	15	15	15	15	17	15	16
35	ALT STRAFORA 1 AD LASTİK 4 ADET TAHLİYE PLASTİĞİ KOY		19	16	16	16	16	16	16	16	16	16	16	18	16	17
36	ALT STRAFORU YERİNE YERLEŞTİR, PEDALA BAS		19	19	19	19	19	19	19	19	19	19	19	18	19	18
37	1 AD KONTROL KUTUSU KAPAĞI AL, 1 AD VİDA İLE YERİNE YERLEŞTİR		20	19	19	19	19	19	19	19	19	19	19	19	19	18
38	1 AD ÜST STRAFOR GRUBU AL, YERİNE YERLEŞTİR		20	20	20	20	20	20	20	20	20	20	20	19	20	19
39	AMBALAJ KARTONUNU TAK, KATLA.		21	20	20	20	20	20	20	20	20	20	20	20	20	19
40	AMBALAJ KARTONUNU 4 ADET ZİMBA İLE ZİMBALA		21	20	20	20	20	20	20	20	20	20	20	20	20	19
41	BARKOD ETİKETİNİ YAPIŞTIR		21	20	20	20	20	20	20	20	20	20	20	20	20	19

Şekil 7.6.'da ise senaryo 2'ye ilişkin atama sonuçları görülmektedir.



Şekil 7.6. Senaryo 2'ye ilişkin optimize edilmiş 20 iş istasyonlu dengeleme sonucu

Üst kısımda iş ögelerinin atandığı istasyonlar, bu istasyonlardaki iş ögesi sayıları, kayıp süreler görülmektedir. Aynı zamanda istasyonlar içindeki iş ögeleri de “sürükle ve bırak” yöntemiyle bulunan çözüm dışında öncelik ilişkilerinin izin verdiği istasyonlara elle de atanabilmektedir.

Örneğin bunlardan şekilde görülen 1 no.’lu istasyona atanmış olan 2 no.’lu iş ögesine ilişkin süreç ve süre bilgileri sağ taraftaki bölümde izlenebilmektedir. Tıklanan her iş ögesine ilişkin öncelik kısıtları dahilinde elle atanabilecek istasyonlar da yeşil oklarla işâret edilmektedir. Bu bakımdan 2 no.’lu iş ögesinin 1, 2 ve 3 no.’lu iş istasyonlarına elle atanma olasılığı vardır. Programdaki bu esneklik, GA’nın bulduğu çözümden sonra kullanıcının isterse varolan çözüm üzerinde sezgisel olarak değişim yapmasına olanak sağlar.

7.6. Sonuçların Değerlendirilmesi

Program, farklı parametre değerlerinin oluşturduğu senaryolar için Tablo 7.4.’de görülen sonuçları üretmiştir. Aynı renklerdeki sütunlar, üretilen sonuçların birbirleriyle benzer olduğunu ifâde etmektedir.

Tablo 7.4. Elde edilen çözümlerin performans kıyaslaması

Genetik Algoritma Parametreleri	Arçelik-LG Varolan Durum	Senaryo 1	Senaryo 2	Senaryo 3	Senaryo 4	Senaryo 5	Senaryo 6	Senaryo 7	Senaryo 8	Senaryo 9	Senaryo 10	Senaryo 11	Senaryo 12	Senaryo 13	Senaryo 14
Popülasyon Büyüklüğü		32	50	100	75	25	60	32	50	100	32	500	20	32	32
Kuşak Sayısı		50	100	200	100	75	120	80	50	25	150	250	15	50	50
Çevrim Süresi	23	23	23	23	23	23	23	23	23	23	23	23	23	23	23
İstasyon Sayısı	20	21	20	20	20	20	20	20	20	20	20	20	20	20	19
İstasyon Süreleri Standart Sapma Değeri	2,76	2,09	2,06	2,06	2,06	2,06	2,06	2,06	2,06	2,06	2,06	2,06	2,13	2,06	2,24
Kayıp Süre Kareleri Toplamı	25,76	31,8	22,5	22,5	22,5	22,5	22,5	22,5	22,5	22,5	22,5	22,5	22,8	22,5	15,69
% Denge Kaybı	18,67%	22,54	18,67	18,67	18,67	18,67	18,67	18,67	18,67	18,67	18,67	18,67	18,67	18,67	14,39

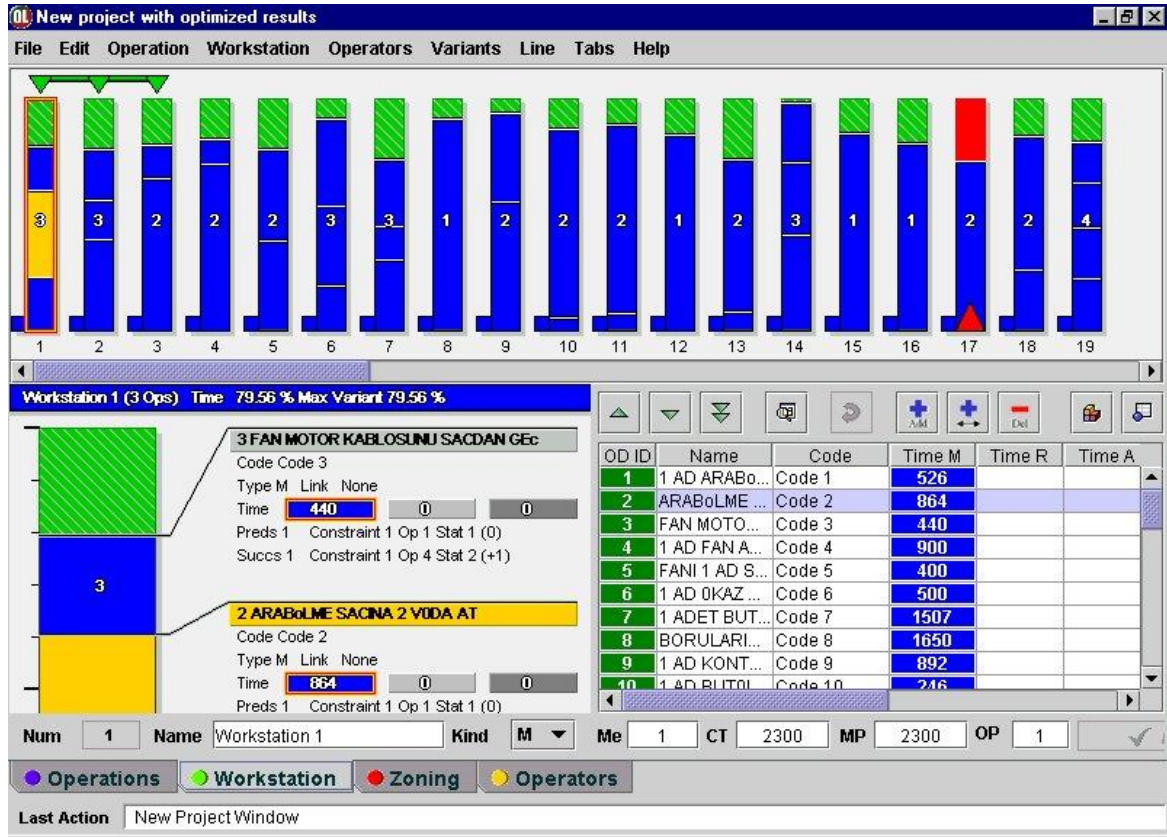
Programın ürettiği bu senaryolar içerisinde, istasyonlara atama bağlamında, senaryo 2-3-7-12’de farklı çözümler söz konusudur. GA ile 20 istasyon dahilinde üretilen tüm sonuçlar, Arçelik-LG’de şu anda kullanılan çözümdeki % denge kaybı olan 18,67’ye eşit olarak bulunmuştur.

Ama Arçelik-LG’de kullanılan çözüme seçenek dengelemeler sunan GA, istasyonlar arasındaki yükü daha dengeli olarak dağıtmıştır. Elde edilen çözümlerde 20 iş istasyonundaki istasyon sürelerinin standart sapması 2,76’dan 2,06’ya düşürülmüştür. Dolayısıyla kayıp sürelerin kareleri değeri de 25,76 sn’dan 22,5 sn’ye düşürülerek, iş yükünün daha dengeli dağıtılması sağlanmıştır.

Montaj hattımızın çevrim süresinin 23 sn olması durumuna göre atayabileceğimiz en az iş istasyonu sayısı daha önceki hesaplamalarımızda 17 sn olarak bulunmuş idi. Çözümler sırasındaki denemelerde bu sayı, eldeki öncelik ve süre kısıtları dolayısıyla genetik algoritma çözümünde 20’den daha aşağı çekilememektedir. Çünkü 19 iş istasyonuna göre yaptığımız tüm denemelerde en az 1 ya da 2 iş istasyonun süresinin, çevrim süresini aştığı çözümlerle karşılaştık.

Bu çözümlerden Şekil 7.7.’de sonuçları görülen senaryo 14, 19 iş istasyonuna indirgenerek denge kaybı % 14,69’a kadar düşürüldü ama 17. istasyonun çevrim süresi 3,5 sn aşıldı. Eğer 17. istasyona atanan 30 ve 35 no.’lu iş öğelerinin sürelerini azaltmaya yönelik küçük bir iyileştirme çalışması yapılabilirse MHD kaybı da % 18,67’den % 14,39’a ve istasyon sayısı da 20’den 19’a düşürülmüş olacaktır.

Sonuç olarak, bu problem için etkin çözümler sağlanmış olup, literatür çalışmalarındaki elde edilen sonuçlara da dayanarak iş öğeleri arasındaki önceliklerin esnek olduğu problemlerde daha da etkili ve hızlı çözümler sunabildiği görülmektedir.



Şekil 7.7. Senaryo 14'e ilişkin 19 iş istasyonlu optiline dengeleme sonucu

8. SONUÇLAR ve TARTIŞMA

Bu çalışmada evrimsel kuramın temellerine dayandırılarak geliştirilmiş olan genetik algoritmalar konusu ve montaj hattı dengeleme problemlerine uygulanma sistemiği incelenmeye çalışıldı. Çözümlenmesi, ancak olasılıkların tümünün denenmesiyle olası problemlerin daha kolay çözümünü sağlayan bu algoritmalar, özellikle yapay zekâda sıklıkla kullanılmaktadır.

Uygulama sonuçları da bize, elde edilen çözümlerin, kalitesi yönünden olumlu olduğunu göstermektedir. NP diye adlandırılan bu problem grubuna MHD problemlerinin yanı sıra günümüzün pek çok problemi de girmektedir. Bu tür problemlerin çözümünde kullanılmak üzere basit ve uygulanabilir bir fonksiyon geliştirilmediği sürece genetik algoritmalar hızlı ve güçlü bir çözüm yolu olarak kalmayı sürdürecektir.

KAYNAKLAR

- Acar, N.**, 1983. “Üretim Plânlama Yöntem ve Uygulamaları”, MPM Yayınları, No:203, Ankara.
- Acar, N.**, 2000 “Üretim Plânlama Yöntem ve Uygulamaları”, MPM Yayınları, No:280, Ankara.
- Acar, N., Eştaş, S.**, 1986. “Kesikli Seri Üretim Sistemlerinde Plânlama ve Kontrol Çalışmaları”, Milli Prodüktivite Merkezi Yayınları, No:309, Ankara, 2. Basım.
- Aksoy, E.**, 1997. “Montaj Hattı Dengelemede Çok Kriterli Karar Verme Yöntemlerinin Kullanılması”, *Yüksek Lisans Tezi*, İ.Ü. İşletme Fakültesi, İstanbul.
- Andersen E.J., Ferris, M.C.**, 1990. “A Genetic Algorithm for the Assembly Line Balancing Problem”, *Computer Science Technical Report, No:926*, University of Wisconsin, USA.
- Bard, J.F.**, 1989. “Assembly Line Balancing with Parallel Work Stations and Dead Time”, *International Journal of Production Research*, V.27, N.6, pp. 1005-1018.
- Baskak, Murat.**, 1991. “Montaj Hatlarının Dengelenmesinde Çok Amaçlı Bir Yaklaşım”, *Yüksek Lisans Tezi*, İ.T.Ü. Fen Bilimleri Enstitüsü, İstanbul.
- Baybars, İ.**, 1986. “A Survey of Exact Algorithms for the Simple Assembly Line Balancing Problem, *Management Science*, V.32, N.8, pp. 909-932.
- Brosch, A., Wiendahl, H. P.**, 1990. “Simultaneous Assembly: Shortening Production Time”, *Industrial Management and Data Systems*, V.90, N.4, pp. 6-12.
- Chan, K.C., Tanton, R.**, 1996. “Assembly Line Balancing Using a Genetic Algorithm With Repair Routine, Source: Intelligent Engineering Systems Through Artificial Neural Networks”, V6, *Proceedings of the 1996 Artificial Neural Networks in Engineering, ANNIE'96*, Nov 10-13 1996, St.Louis, MO, USA, pp. 385-390.

- Charles, L.K., Freeman, L.M.**, 1999. "Industrial Applications of Genetic Algorithms", Department of Aerospace Engineering and Mechanics, University of Alabama, pp. 1-13.
- Cheng, R., Gen, M., Tsujimura, Y.** 1996. "A Tutorial Survey of Job Scheduling Problems Using Genetic Algorithms-I", *Computers and Industrial Engineering*, **30**, pp. 983-997.
- Cicirello, V.A., Smith, S., F.**, 2000. "Modeling GA Performance for Control Parameter Optimization", *Genetic and Evolutionary Computation Conference (GECCO 2000)*, LasVegas, Nevada, USA, July 8-12.
- Croce, F.D., Tadei, R., Volta, G.**, 1995. "A Genetic Algorithm for the Job Problem", *Computers and Industrial Engineering*, **25**, 1-4.
- Dar-El, E. M., Rubinovich, M.**, 1979. "MUST-A Multiple Solutions Technique for Balancing Single Model Assembly Lines", *Management Science*, V.25, N.11, pp. 1105-1114.
- Davis, L.**, 1985. "Applying Adaptive Algorithms to Epistatic Domains", *Proc. of Int. Joint Conf. on Artificial Intelligence*, USA, pp. 162-164.
- Davis, L.**, 1991. "Handbook of Genetic Algorithms", Van Nostrand Reinhold, New York, pp. 72-90
- Debora, A., Roger, L.W.**, 2000. "Applying Genetic Algorithms to the U shaped Assembly Line Balancing Problem", Mathematical and Computer Sciences Department, The University of Tulsa, USA.
- De Jong, J. R.**, 1957. "The Effects of Increasing Skill On Cycle Time and Its Consequences for Time Standards", *Ergonomics*, V.1, N.1, pp. 51-60.
- Elwoods, B.**, 1965. "Modern Production Management", John Willey and Sons, Inc.
- Emmeche, C.**, 1994. "Garden in the Machine. The Emerging Science of Artificial Life", Princeton University Press, USA.
- Engin, O.**, 2001. "Akış Tipi Çizelgeleme Problemlerinin Genetik Algoritma ile Çözüm Performansının Artırılmasında Parametre Optimizasyonu", *Doktora Tezi*, İ.T.Ü. Fen Bilimleri Enstitüsü, İstanbul.

- Erel, E., Sabuncuoğlu, İ., Tanyer, M.**, 2000. "Assembly Line Balancing Using Genetic Algorithms", *Journal Of Intelligent Manufacturing*, **11**, pp. 295-310.
- Falkenauer E., Brahim R., Patrick F.**, 2001. "A Multiple Objective Grouping Genetic Algorithm for Assembly Line Design", *Journal of Intelligent Manufacturing*, **12**, pp. 467-485.
- Fleming, P.J., Zalzal, A.M.S.**, 1997. "Genetic Algorithms in Engineering Systems", IEE Control Engineering Series 55, Published by The Institution of Electrical Engineers.
- Gen M., Tsujimura, Y., Li, Y.X.**, 1996. "Fuzzy Assembly Line Balancing Using Genetic Algorithms", *Computers & Industrial Engineering*, **31**, pp. 631-634.
- Ghedjati, F.**, 1999. "Genetic Algorithms for the Job-Shop Scheduling Problem With Unrelated Parallel Constraints: Heuristic Mixing Method Machines and Precedence", *Computers and Industrial Engineering*, **37**, pp. 39-42.
- Goldberg, D.E.**, 1989. "Genetic Algorithms in Search Optimization and Machine Learning", Addison Wesley Publishing Company, USA.
- Goldberg, D.E., Lingle, R.**, 1985. "Alleles, Loci and TSP", *Proc. 1st Int. Conf. Genetic Algorithms. Lawrence Erlbaum Associates*, Hillsdale NJ, USA, pp.154-159.
- Goncalves J.F., Almeida J.R.**, 2002. "A Hybrid Genetic Algorithm for Assembly Line Balancing", *Journal of Heuristics*, **8**, pp. 629-642.
- Hasşerbetçi, K.H.**, 1997. "Genetik Algoritmaların Yöneylem Araştırmasında Kullanımı", *Yüksek Lisans Tezi*, İstanbul Üniversitesi İşletme Fakültesi, İstanbul.
- Helgeson, W.P., Birnie, D.P.**, 1961. "Assembly Line Balancing using the Ranked Positional Weight Technique", *Journal of Industrial Engineering*, V.12, N.6, pp.384-398.
- Hoffmann, T.R.**, 1963. "Assembly Line Balancing with a Precedence Matrix", *Management Science*, V.9, N.4, pp. 551-563.

- Holland, J.H.**, 1975. "Adaptation in Natural and Artificial System", Ann Arbor, The University of Michigan Press, USA.
- Ignall, E.J.**, 1965. "A Review of Assembly Line Balancing", *Journal of Industrial Engineering*, V.16, N.4, pp. 244-254.
- Jackson, J.R.**, 1956. "A Computing Procedure for a Line Balancing Problem", *Management Science*, V.2, N.3, pp. 261-271.
- Kao, E.P.C.**, 1979. "Computational Experience with a Stochastic Assembly Line Balancing Algorithm", *Computers & Operations Research*, V.6, N.2, pp. 79-86.
- Kelth C.C., Patrick C.L., Hui, K.W., Frency S.F.N.**, 1998. "Handling the Assembly Line Balancing Problem in the Clothing Industry using a Genetic Algorithm", *International Journal of Clothing Science and Technology*, **10**, pp. 21-37.
- Kim Y.K., Kim J.J., Kim Y.H.**, 1996. "Genetic Algorithms for Assembly Line Balancing with Various Objectives", *Computers & Industrial Engineering*, **30**, pp. 397-409.
- Kurt, M., Emetay, C.**, 1998. "Genetik Algoritma ve Uygulama Alanları", *Lisans Tezi*, M.Ü. Teknik Eğitim Fakültesi Makina Bölümü, İstanbul.
- Kuru, M.**, 1987. "*Biyoloji Eğitimi Ders Notları*", Moleküler Biyoloji, Atatürk Üniversitesi Eğitim Fakültesi, Erzurum.
- Lee, S.G., Khoo, L.P., Yin, X.F.**, 2000. "Optimizing an Assembly Line Through Simulation Augmented by Genetic Algorithms", *International Journal of Advanced Manufacturing Technology*, **16**, pp. 220-228.
- Lehman, M.**, 1969. "On Criteria for Assigning Models to Assembly Lines", *International Journal of Production Research*, V.7, N.4, pp. 269-285.
- Leu Y.Y., Matheson L.A., Rees L.P.**, 1994. "Assembly-Line Balancing using Genetic Algorithms with Heuristic-Generated Initial Populations and Multiple Evaluation Criteria", *Decision Sciences*, **25**, pp. 581-606.
- Macaskill, J.L.C.**, 1972. "Production Line Balances for Mixed-Model Lines", *Management Science*, V.19, N.4, pp. 423-434.

- Maynard, H.B.**, 1969. "Industrial Engineering Handbook", Mc GrawHill Company, Newyork.
- McKenzie, D.P., Yılmaz, Y.**, 1991. "Deformation and Volcanism in Western Turkey and the Aegean", *Bull. Tech. Univ. İstanbul*, **44**, pp. 345-373.
- MESS**, 1987, "Sanayide Bilgisayarların Etkin ve Verimli Kullanımı", MESS 1987 Yılı Seminerleri Dizisi-II, Eğitim Kitapları Dizisi-15
- Monden, Y.**, 1983. "Toyota Production System", *Industrial Engineering and Management Press*, Atlanta, GA.
- Oğuz, M., Akbaş, S.**, 1997. "Genetik Algoritma ve Bilgisayar Uygulaması", *Lisans Tezi*, Y.T.Ü. Endüstri Mühendisliği Bölümü, İstanbul.
- Oliver, I.M., Smith, D.J., Holland, J.R.C.**, 1987. "A Study of Permutation Crossover Operators on the Travelling Salesman Problem", *Proc. 2nd Int. Conf. Genetic Algorithms*, Lawrence Erlbaum Associates, Hillsdale NJ, pp. 224-230.
- Oraler, G.**, 1990. "Genetik I" , İ.Ü. Fen Fakültesi, *Genetik Ders Notları*, İstanbul.
- Özkaya, İsmail.**, 2002. "Genetik Algoritmalar", *Lisans Tezi*, İ.Ü. Mühendislik Fakültesi Endüstri Mühendisliği Bölümü, İstanbul.
- Özuysal M., Soysal O.**, 1997. "Genetik Algoritmalar Kullanan Polinom İşlemcisi", Tübitak Projesi, Ankara.
- Ponnambalam S.G., Aravindan P., Naidu G.M.**, 2000. "A Multi-Objective Genetic Algorithm for Solving Assembly Line Balancing Problem", *International Journal of Advanced Manufacturing Technology*, **16**, pp. 341-352.
- Rekiek, B., Delchambre, A.**, 2001. "Hybrid Assembly Line Design", Proceedings of the IEEE International Symposium on Assembly and Task Planning, (ISATP2001), May 28-29 2001, Fukuoka, pp. 73-78.
- Rubinovitz, J., Levitin, G.**, 1995. Genetic Algorithm for Assembly line Balancing, *International Journal of Production Economics*, **41**, 343-354.

- Schaffer, J.D., Eshelman, L.J.**, 1996. "Balancing SMD Assembly Lines with Genetic Algorithms", *Proceedings of the 1996 Japan-USA Symposium on Flexible Automation*, Jul 7-10 1996, Boston, MA, USA, pp. 1547-1552.
- Sniedovich, M.**, 1981. "Analysis of Preference Order Assembly Line Problem", *Management Science*, V.27, N.9, pp. 1067-1104.
- Starkweather, T., Mcdaniel, S., Mathias C., Withley, C., Whitley, D.**, 1991. "A Comparison of Genetic Sequencing Operators", *Proc. 4th Int. Conf. Genetic Algorithms*, Morgan Kaufman Publishers, Los Altos, CA, pp. 69-76.
- Suresh, G., Vinod, V.V., Sahu S.**, 1996. "A Genetic Algorithm for Assembly Line Balancing", *Production Planning & Control*, 7, pp. 38-46.
- Tsujimura Y., Gen M., Kubota E.**, 1995. "Solving Fuzzy Assembly-Line Balancing Problem with Genetic Algorithms", *Computers & Industrial Engineering*, 29, pp. 543-547.
- Uzmen, M.**, 1990. "Montaj Hattı Dengeleme", *Yüksek Lisans Tezi, İ.T.Ü. Fen Bilimleri Enstitüsü, İstanbul*.
- Yong, J.K., Yeo, K.K., Yong, K.C.**, 1998. "A Heuristic Based Genetic Algorithm for Workload Smoothing in Assembly Line Balancing", *Computers Operations Research*, 25, pp. 99-111.
- Wee, T. S., Magazine, M.J.**, 1982. "Assembly Line Balancing as Generalised Bin Packing", *Operations Research Letters*, V.1, N.2, pp. 56-58.
- Wild, R.**, 1979. "Production and Operations Management", Holt, Rinehart Winston, Great Britain.
- Xu, G., Penson, R., Callan, R.**, 1998. "Method for Assembly Line Balancing using Genetic Algorithms", Source: Intelligent Engineering Systems Through Artificial Neural Networks, V8, *Proceedings of the 1998 Artificial Networks in Engineering Conference, ANNIE*, Nov 1-4 1998, St.Louis, MO, USA, pp. 329-334.
- Zhao Z.Y., Souza R.**, 2000. "Genetic Production Line-Balancing for the Hard Disk Drive Industry", *International Journal of Advanced Manufacturing Technology*, 16, pp. 297-302.

ÖZGEÇMİŞ

Raşı̇t Özkan, 29 Mayıs 1978 İstanbul doğumludur. 1995’de Avcılar Anadolu Teknik Lisesi Makine Bölümü’nden, 2000’de İstanbul Üniversitesi Mühendislik Fakültesi Endüstri Mühendisliğı Bölümü’nden dereceyle mezun oldu. Ekim 2000 itibariyle İstanbul Teknik Üniversitesi Endüstri Mühendisliğı Yüksek Lisans Programı’na başlamıştır. İyi derecede İngilizce, temel düzeyde Fransızca ve Bulgarca bilmektedir. Temmuz 2001’den bu yana Armaform End. İzo. San. Tic. A.Ş’de Mâliyet ve Plânlama Koordinatörü olarak çalışmaktadır.