

**Alternatif Rotalı Hücre Oluşturma Problemlerinin  
Çözümü için Yeni bir Melez Yaklaşım**

Mümin Sönmez  
Yüksek Lisans Tezi

Endüstri Mühendisliği Ana Bilim Dalı  
Haziran-2011

## JÜRİ VE ENSTİTÜ ONAYI

Mümin Sönmez'in "Alternatif Rotalı Hücre Oluşturma Problemlerinin Çözümü için Yeni bir Melez Yaklaşım" başlıklı Endüstri Mühendisliği Anabilim Dalındaki, Yüksek Lisans Tezi 20.05.2011 tarihinde, aşağıdaki jüri tarafından Anadolu Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin ilgili maddeleri uyarınca değerlendirilerek kabul edilmiştir.

	<u>Adı Soyadı</u>	<u>İmza</u>
Üye (Tez Danışmanı) :	Yard. Doç Dr. Gürkan ÖZTÜRK	.....
Üye :	Doç. Dr. Nil ARAS	.....
Üye :	Yard. Doç. Dr. Tuğba SARAÇ	.....

Anadolu Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun  
..... tarih ve ..... sayılı kararıyla onaylanmıştır.

Enstitü Müdürü



## ÖZET

**Yüksek Lisans Tezi**

### **ALTERNATİF ROTALI HÜCRE OLUŞTURMA PROBLEMLERİNİN ÇÖZÜMÜ İÇİN YENİ BİR MELEZ YAKLAŞIM**

**Mümin SÖNMEZ**

**Anadolu Üniversitesi  
Fen Bilimleri Enstitüsü  
Endüstri Mühendisliği Anabilim Dalı**

**Danışman: Yard. Doç. Dr. Gürkan ÖZTÜRK  
2011, 102 Sayfa**

Bu çalışmada alternatif rotalı hücre oluşturma problemlerinin çözümü için parçacık sürü optimizasyonu (PSO) ve sinir ağları temelinde melez bir yaklaşım önerilmiştir. Hücre oluşturma problemi için önerilen bu yaklaşım iki aşamada ele alınmaktadır. Birinci aşamada Parçacık Sürü Optimizasyon yöntemi ile alternatif rotalar belirlenir, daha sonra Rekabetçi Sinir Ağı birinci aşamada belirlenen rotalara göre hücreleri oluşturur. Çalışmanın özgün yanı PSO yönteminde yer alan parçacıkların birim hiperküp çözüm uzayında arama yapmasıdır. Buradaki sürekli değerler, kesikli değerlere dönüştürülerek alternatif rotalar belirlenmektedir. Geliştirilen yaklaşım, literatürde yer alan birçok test problemi için denenerek diğer yöntemlerle karşılaştırma yapılmaktadır. Bu karşılaştırmalar doğrultusunda, problemlerin çoğu için bilinen eniyi değerlere ulaşılırken, bazılarında ise daha iyi sonuçlara elde edilmektedir.

**Anahtar Kelimeler:** Hücresel üretim, grup teknolojileri, yapay zeka, parçacık sürü optimizasyonu (PSO), rekabetçi sinir ağı.

## ABSTRACT

Master of Science Thesis

### A NEW HYBRID ALGORITHM TO SOLVE THE CELL FORMATION PROBLEM CONSIDERING MULTIPLE PROCESS ROUTINGS

Mümin SÖNMEZ

Anadolu University  
Graduate School of Sciences  
Industrial Engineering Program

Supervisor: Assoc. Prof. Dr. Gürkan ÖZTÜRK  
2011, 102 pages

In this study, a hybrid algorithm with Particle Swarm Optimization and Competitive Neural Network is proposed to solve the manufacturing cell formation problem considering multiple process routings for parts. Alternate process routings are selected by Particle Swarm Optimization, after that Competitive Neural Network is working on composing the cells using these selected routes. The main objective of this algorithm is maximizing the Grouping Efficacy. Specific side of this study is particles that inside the PSO method searching in unit hypercube solution space with continuous values. Continuous values are transformed to discrete values for obtaining alternate routes. This approach is tested on the most known problems in literature and compared with other methods. By these comparisons, it is comprehended that this new approach reaches better solution for some of these problems.

**Keywords:** Cell formation problem, grouping technology, artificial intelligence, particle swarm optimization, competitive neural network

## TEŞEKKÜR

Öğrencisi olduğum günden beri desteğini esirgemeyen, her açıdan kendime örnek almaya çalıştığım değerli hocam sayın Yard. Doç. Dr. Gürkan ÖZTÜRK'e teşekkürlerimi sunuyorum.

Eğitim-öğretim hayatım boyunca üzerimde emeği olan hocalarıma, özellikle tez jürimde yer alan hocalarım sayın Doç. Dr. Nil ARAS ve Yard. Doç. Dr. Tuğba SARAÇ'a teşekkür ederim. Destekleri ile yanımda olan arkadaşlarım Araş. Gör. Gülçin DİNÇ YALÇIN, Araş. Gör. Erdener ÖZÇETİN, Emre ÇİMEN'e ve beni yetiştirip bu günlere ulaşmamı sağlayan aileme teşekkürlerimi sunuyorum.

## İÇİNDEKİLER

ÖZET.....	i
ABSTRACT.....	ii
TEŞEKKÜR.....	iii
İÇİNDEKİLER.....	iv
ŞEKİLLER DİZİNİ.....	vi
ÇİZELGELER DİZİNİ.....	ix
1. GİRİŞ.....	1
2. GRUP TEKNOLOJİSİ VE HÜCRESEL ÜRETİM SİSTEMİ.....	3
2.1. Grup Teknolojisi.....	3
2.2. Hücresel Üretim.....	6
2.2.1. Hücresel üretimin avantajları.....	8
2.2.2. Hücresel üretimin dezavantajları.....	10
2.2.3. Hücresel üretim sistemi uygulamaları.....	11
3. HÜCRE OLUŞTURMA PROBLEMİ VE FARKLI TÜRLERİ.....	14
3.1. Hücre Oluşturma Problemi.....	14
3.2. Alternatif Rotalı Hücre Oluşturma Problemi.....	17
3.3. Literatür Taraması.....	19
3.4. Hücre Oluşturma Problemi Çözüm Yöntemleri.....	22
3.4.1. Sezgisel yöntemler.....	23
3.4.2. Matematiksel programlama yöntemi.....	23
3.4.3. Kümeleme analizini temel alan yöntemler.....	25
3.4.4. Grafik bölümlenme yöntemi.....	26
3.4.5. Yapay zeka yöntemleri.....	26
3.4.6. Çözüm başarısının değerlendirilmesi.....	36
4. ALTERNATİF ROTALI HÜCRE OLUŞTURMA PROBLEMLERİNİN ÇÖZÜMÜ İÇİN YENİ BİR MELEZ YAKLAŞIM.....	38
4.1. Rekabetçi Sınır Ağı.....	38
4.2. Parçacık Sürü Optimizasyon Yöntemi.....	42
4.3. Kesikli Parçacık Sürü Optimizasyon Yöntemi için Yeni bir Yaklaşım.....	44
4.4. Alternatif Rotalı Hücre Oluşturma Problemlerinin Çözümü için Melez Bir Yaklaşım.....	49

4.5. Problem Çözümü için Tasarlanan Program.....	56
5. HESAPSAL SONUÇLAR	59
6. SONUÇ ve ÖNERİLER	63
KAYNAKLAR	65
EK-1 Hücre Oluşturma Problemlerinin Karşılaştırılması .....	71
EK-2 Peaks Fonksiyonunda Bütünsel Enküçük için Yeni Kesikli PSO Adımları	74
EK-3 Alternatif Rotalı Hücre Oluşturma Problemleri ve Çözümleri.....	78

## ŞEKİLLER DİZİNİ

Şekil 2.1 Atölye tipi üretim sistemi yerleşim planı .....	5
Şekil 2.2 Akış tipi üretim sistemi yerleşim planı .....	5
Şekil 2.3 Hücresel üretim yerleşim planı .....	7
Şekil 2.4 Farklı Düzeyde Üretim Miktarı ve Ürün çeşitliliği için Yerleşim Tipleri ve Üretim Sistemleri .....	8
Şekil 3.1 Temas matrisi (a), köşegenleştirilmiş temas matrisi (b) .....	17
Şekil 3.2 Alternatif rota olmadığı durumdaki matriste blok köşegenleşme.....	18
Şekil 3.3 Alternatif rotaların olduğu durumdaki matriste blok köşegenleşme.....	18
Şekil 3.4 Kopya makinelerin olması durumdaki matriste blok köşegenleşme .....	19
Şekil 3.5 Hücre oluşturma çözüm yöntemleri.....	23
Şekil 3.6 GA sözde kodlar .....	28
Şekil 3.7 Tavlama benzetimi sözde kodlar .....	30
Şekil 3.8 Yasaklı arama sözde kodlar .....	31
Şekil 3.9 Karınca kolonileri sözde kodlar .....	33
Şekil 3.10 Yapay sinir ağı.....	33
Şekil 4.1 Rekabetçi sinir ağının yapısı .....	39
Şekil 4.2 Örnek bir parça-makine (temas) matrisi .....	40
Şekil 4.3 Rekabetçi sinir ağının açıklayıcı adımları .....	41
Şekil 4.4 RSA sözde kodlar .....	42
Şekil 4.5 PSO ile parçacığın pbest <sub>i</sub> ve gbest'e doğru hareketi .....	44
Şekil 4.6 Parçacık sürü optimizasyon sözde kodlar .....	45
Şekil 4.7 Parçacıkların eski ve yeni koordinatları.....	48
Şekil 4.8 "Peaks" fonksiyonunun grafiği .....	50
Şekil 4.9 Melez yaklaşıma ait sözde kodlar .....	51
Şekil 4.10 Alternatif rotalara sahip bir parça-makine (temas) matrisi .....	52
Şekil 4.11 Örnek bir parçacık ve boyutları .....	52
Şekil 4.12 Alternatif rotalı hücre oluşturma probleminin çözümü .....	54
Şekil 4.13 Sadece PSO yöntemi için parçacık yapısı.....	54
Şekil 4.14 Excel ortamında tasarlanan programın arayüzü.....	56
Şekil 4.15 Google Spreadsheets ortamında tasarlanan program arayüzü .....	57



## ŞEKİLLER DİZİNİ (Devam)

Şekil P.1 Problem P1 .....	78
Şekil P.2 Çözüm P1 .....	78
Şekil P.3 Problem P2 .....	78
Şekil P.4 Çözüm P2 .....	78
Şekil P.5 Problem P3 .....	79
Şekil P.6 Çözüm P3 .....	79
Şekil P.7 Problem P4 .....	79
Şekil P.8 Çözüm P4 .....	79
Şekil P.9 Problem P5 .....	80
Şekil P.10 Çözüm P5 .....	80
Şekil P.11 Problem P6 .....	80
Şekil P.12 Çözüm P6 .....	81
Şekil P.13 Problem P7 .....	81
Şekil P.14 Çözüm P7 .....	81
Şekil P.15 Problem P8 .....	82
Şekil P.16 Çözüm P8 .....	82
Şekil P.17 Problem P9 .....	83
Şekil P.18 Çözüm P9 .....	83
Şekil P.19 Problem P10 .....	84
Şekil P.20 Çözüm P10 .....	84
Şekil P.21 Problem P11 .....	85
Şekil P.22 Çözüm P11 .....	85
Şekil P.23 Problem P12 .....	86
Şekil P.24 Çözüm P12 .....	86
Şekil P.25 Problem P13 .....	87
Şekil P.26 Çözüm P13 .....	87
Şekil P.27 Problem P14 .....	88
Şekil P.28 Çözüm P14 .....	88
Şekil P.29 Problem P15 .....	90
Şekil P.30 Çözüm P15 .....	91
Şekil P.31 Problem P16 .....	94

## ŞEKİLLER DİZİNİ (Devam)

Şekil P.32 Çözüm P16 .....	96
Şekil P.33 Problem P17 .....	97
Şekil P.34 Çözüm P17 .....	97
Şekil P.35 Problem P18 .....	98
Şekil P.36 Çözüm P18 .....	98
Şekil P.37 Problem P19 .....	98
Şekil P.38 Çözüm P19 .....	99
Şekil P.39 Problem P20 .....	100
Şekil P.40 Çözüm P20 .....	101
Şekil P.41 Problem P21 .....	102
Şekil P.42 Çözüm P21 .....	102

## ÇİZELGELER DİZİNİ

Çizelge 2.1 A.B.D.’deki işletmelerin HÜS kurma nedenleri ve beklenen gelişim oranları .....	12
Çizelge 2.2 Türkiye’deki işletmelerin HÜS kurma nedenleri ve beklenen gelişim oranları .....	13
Çizelge 4.1 Parçacıkların başlangıç konum ve hız değerleri .....	47
Çizelge 4.2 Parçacık birim hız vektörleri.....	47
Çizelge 4.3 Yeni $X_i$ değerlerine geçiş .....	48
Çizelge 4.4 Kesikli değerler ve dönüşüm aralığı .....	49
Çizelge 5.1 Test problemleri listesi.....	59
Çizelge 5.2 Alternatif rotalı hücre oluşturma problemleri için çözüm yöntemlerinin karşılaştırılması .....	61

## 1. GİRİŞ

Üretim sektöründe hızla artan rekabet koşulları, işletmeleri müşteri odaklı üretim sistemlerine yönlendirmiştir. Tek tip ürün tasarımları müşteri istek ve beklentilerini karşılayamaz hale gelmiştir. Günümüzde müşteriler satın almak istedikleri ürünlere ait özellikleri belirterek işletmelerin bu isteklerini hızlı, kaliteli ve düşük fiyat üzerinden karşılamasını beklemektedir. Bu sebeple üretim sistemlerinin talebe hızlı ve etkili cevap verebilecek bir şekilde tasarlanması kaçınılmaz olmuştur.

Geleneksel üretim sistemleri Atölye Tipi Üretim Sistemi ve Akış Tipi Üretim Sistemi olarak ikiye ayrılabilir [1]. Atölye Tipi Üretim Sisteminde parça çeşitliliği fazla üretim hacmi düşükken, Akış Tipi Üretim Sisteminde ise parça çeşitliliği az üretim hacmi yüksektir.

Atölye Tipi Üretim Sisteminde makineler benzer fonksiyonlarına göre bir araya getirildiğinden işlem için gelen parçalar en az bir bölümde işlem görmektedir ve bu nedenle parça dolaşimleri fazla olmaktadır. Akış Tipi Üretim Sisteminde parçalar hat boyunca hareket ettiğinden yerleşim planı esnek olmamakta, bir makineden meydana gelen arıza tüm hattı etkilemektedir.

Bu iki üretim sisteminin dezavantajlarını gidermek için alternatif üretim sistemleri geliştirilmiştir. Parça taşıma mesafesi, hazırlık süresi, stok miktarı, parti büyüklüğünü azaltmayı hedefleyen Grup Teknolojisi (GT), üretim verimliliğini enbüyüklemek için benzer ve tekrarlayan işlem veya görevleri bir araya getirmeyi amaçlayan bir yaklaşımdır [3]. GT'nin, bahsedilen üretim sistemlerine göre avantajlı olma nedeni parça çeşitliliğinin ve üretim hacminin orta büyüklükte olmasıdır. GT'deki üretim uygulaması Hücreli Üretim Sistemi (HÜS) olarak adlandırılmaktadır [4]. Hücreli üretim, benzer işlemleri gören parça aileleri için benzer özelliklere sahip makinelerin gruplandırıldığı ve bu gruplara hücre adı verildiği üretim şeklidir. Hücrelerde yer alan makine türleri her hücre için farklıdır ve hücre yerleşimleri birbirlerine yakın olarak konumlanmaktadır. Her hücrenin belirli parça ya da parça ailelerini işleyebilecek kapasiteye sahip olması gerekir. Hücreli üretim hazırlık zamanını kısaltır böylece parti büyüklüğü azalır, süreç içi stoklarda azalma ve üretim süresinde de bir azalma sağlanmaktadır [5].

Hücreyel üretim sisteminin uygulanmasında karşılaşılan enbüyük problem oluşturulacak parça/makine gruplarının belirlenmesidir. Bu problem hücre oluşturma problemi olarak ele alınır. Parçaların hangi makinelerde işlendiğini gösteren temas matrisi, blok diyagonal matris formuna dönüştürülmeye çalışılır. Böylece hücreler arası taşıma mesafesini en aza indirilirken, hücre içindeki makine kullanımından da maksimum ölçüde faydalanılmaktadır [6].

Hücre oluşturma problemleri NP-Zor problemler sınıfında yer almaktadır [7]. Literatürde hücreyel üretim problemlerinin çözümü için sezgisel yaklaşımlardan matematiksel programlama yaklaşımlarına kadar birçok algoritma geliştirilmiştir [8]. Hücre oluşturma problemlerinin çözümü için Yapay Zeka (YZ) teknikleri de kullanılmaktadır.

Çalışmada hücre oluşturma problemlerinin farklı bir türü olan alternatif rotalı hücre oluşturma problemi ele alınmaktadır. Problemin bu şekli ile ele alınma nedeni, gerçek hayat problemlerinde parçaların izleyebilecekleri rotaların birden fazla olmasıdır. Problem çözümü için Parçacık Sürü Optimizasyon (PSO) ve Rekabetçi Sinir Ağını (RSA) temel alan iki aşamalı melez bir yöntem önerilmektedir. Birinci aşamada, parçacıklar birim hiperküp çözüm uzayında arama yapmakta ve buradaki sürekli değerler kesikli değerlere dönüştürülerek parçaların izleyecekleri rotalar belirlenmektedir. Sonraki aşamada ise RSA, belirlenen rotalara göre indirgenmiş temas matrisini girdi alarak blok köşegen bir matris formuna dönüştürmektedir. Böylece parça ve makinelerin atandığı hücreler belirlenmektedir. Yöntemin amacı blok köşegen matris formunu temel alan gruplama etkinlik değerini enbüyüklemeektir. Ayrıca Mayıs 2011 tarihine kadar literatürde bu problemin çözümü için bahsedilen melez yaklaşımı temel alan bir çalışma ile karşılaşılmamıştır.

Çalışmanın ikinci bölümünde hücreyel üretim sistemi hakkında bilgi verilmektedir. Bir sonraki bölümde hücre oluşturma problemi ele alınarak literatürde karşılaşılan çözüm yaklaşımlarından bahsedilmektedir. Dördüncü bölümde ise problem çözümü için önerilen yöntem açıklanmaktadır. Beşinci bölümde önerilen yaklaşım literatürde yer alan test problemleri üzerinde denenerek karşılaştırma yapılmaktadır.

## 2. GRUP TEKNOLOJİSİ VE HÜCRESEL ÜRETİM SİSTEMİ

GT, parça üretim ve tasarımında ekonomik fayda sağlamak amacıyla, benzer parça ailelerini ve makine gruplarını bir araya getirmeyi amaçlayan bir üretim felsefesidir.

Atölye ve Akış Tipi Üretim Sistemine alternatif olan Parti Tipi Üretim, üretim hacminin orta, ürün çeşitliliğinin fazla olduğu bir sistemdir. Bu sistem endüstriye büyük katkılar sağladığından dolayı dünyada büyük ölçüde kullanılmaktadır. Tüm üretim faaliyetleri içindeki payı %60-%80 arasındadır. Fakat Parti Tipi Üretimde parça çeşitliliğinin fazla olması nedeniyle, parça tasarımları karmaşık olmaktadır. Tasarım sürecinde alınan kararlar ürünlerin fiyatını, kalitesini ve teslimat süresini doğrudan etkilemektedir, bu nedenle tasarım süreci, üretimde kullanılan makine ve ekipmanları göz önüne alınarak gerçekleştirilmektedir. Yine parça çeşitliliğinin fazla olmasından dolayı üretim ekipmanı, çizelgeleme, hazırlık zamanı, süreç içi stok vb. gibi üretim faktörlerindeki olumsuzluklar nedeniyle maliyet artmaktadır. Ürünlerin küresel pazarda rekabet edebilmesi için parti tipi üretim sisteminin değişikliğe uğraması gerekir. Bu nedenle fiyat, kalite, zamanında dağıtım gibi konularda ilerleme sağlayan, tasarım ve üretim faaliyetlerinin bütünleşik bir yapı içinde sunulduğu bir üretim sistemine olan ihtiyaç doğmuştur. Bu gerekliliği karşılamak için GT, tasarım ve üretim faaliyetleri arasında bağlantıyı kurarak, seri üretim ve atölye tipi üretimin faydalarını ortak bir sistemde toplamaktadır [9].

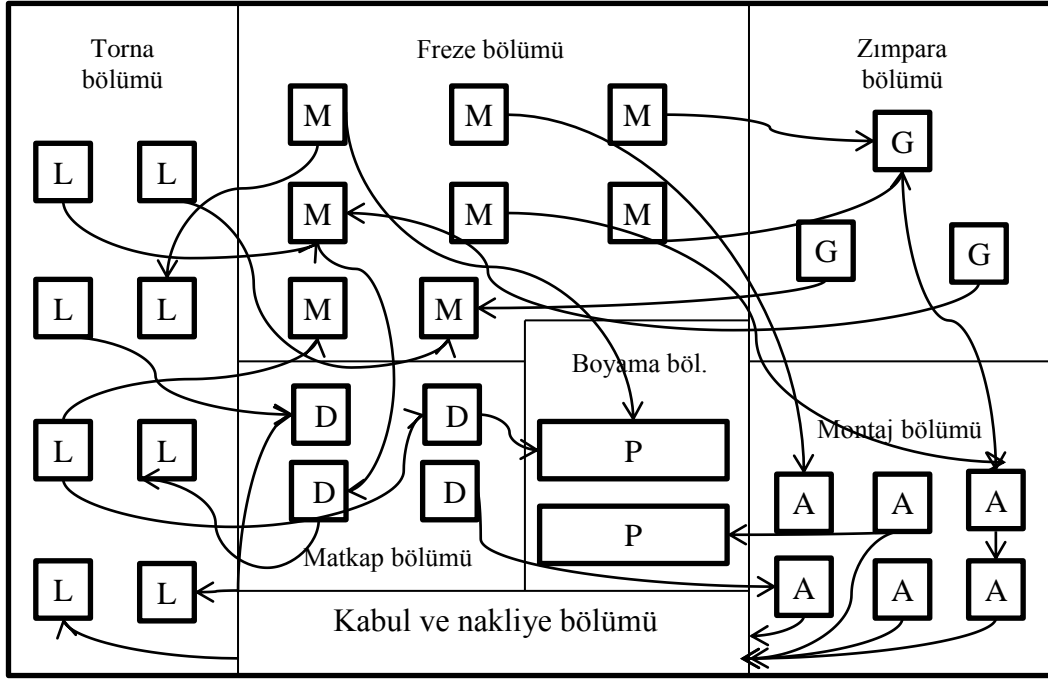
### 2.1. Grup Teknolojisi

1925 yılında R.E. Flanders bir makine işletmesinde üretilen standart parçaların taşıma mesafesini küçükmek için ürünlere yönelik bölümlerin oluşturulması gerektiğini önermiştir, bu yaklaşım GT başlangıcı olarak alınmaktadır. 1937'de A. P. Sokolovski benzer özelliklere sahip parçaların standartlaştırılmış işlemlerle üretilebileceğini önermiştir. Bu düşünce 1960'da İngiltere'de çalışan Mitrofanov tarafından geliştirilmiştir, yine J. L.Burbidge bu kavrama bağlı kalarak sistematik planlama yaklaşımını geliştirmiştir. Sonraki 20 yıl içinde bu felsefe tüm dünyaya yayılmıştır. Günümüzde geniş tesislerin içinde

birbirinden bağımsız alt birimler oluşturularak üretim yapılmaktadır. Tüm bu çabalar “benzer şeyler benzer şekilde yapılmalıdır” yönetim ilkesini temel alan GT’yi oluşturmaktadır. Burada benzer şeyler sözü ile; ürün tasarımı, süreç tasarımı, üretim, montaj ve üretim kontrol gibi faaliyetler ifade edilmektedir [2].

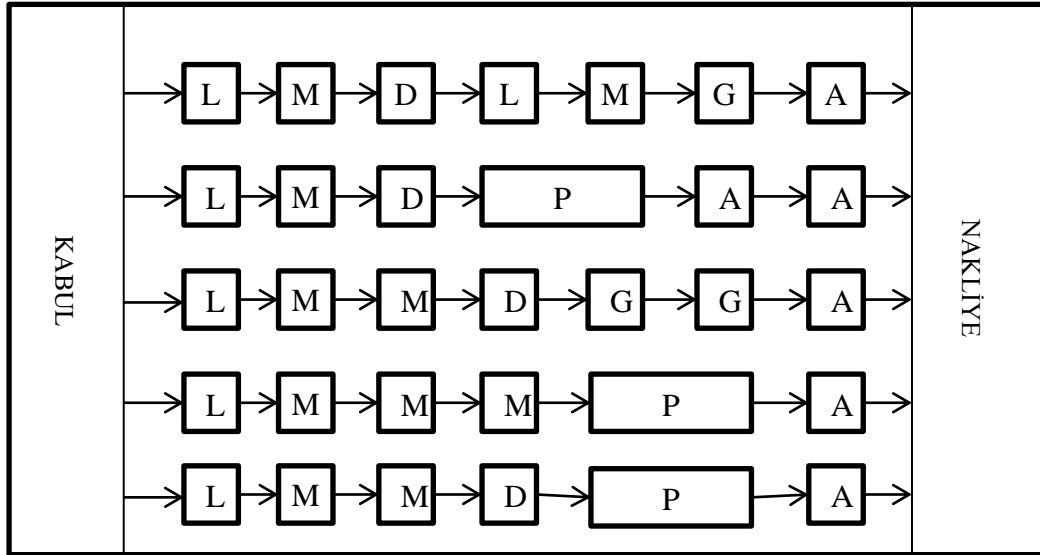
Üretim sistemleri, üretim yönetimi, üretim miktarı ve ürün çeşitliliğine göre alt dallara ayrılmaktadır [10]. Ayrıca; üretim miktarlarına göre, atölye tipi üretim sistemi, seri üretim sistemi olarak ikiye ayrılabilir. Değişen koşullar sonucunda işletmeler farklı üretim sistemlerine ihtiyaç duymuştur: yalın üretim sistemi, parti tipi üretim sistemi, esnek üretim sistemi, hücreli üretim sistemi bunlardan bazılarıdır. Atölye tipi üretim sisteminde ürün çeşitliliği fazla üretim miktarı düşük bir üretim yapısı vardır. Genellikle fiziksel büyüklüğü fazla olan ürünler üretilmektedir. Atölye yerleşimi esnek üretime uygun şekilde tasarlanmış olmalıdır. Bu yöntemin dezavantajı ürünlerin bölümler arasında taşınarak işlenmesidir, bu da süreç içi stok düzeyinde artışa neden olmaktadır. Şekil 2.1 atölye tipi yerleşim planını ve parçaların izlediği yolları göstermektedir. Seri üretim sistemi ürüne olan talebin yüksek olduğu durumlarda uygulanır. Ayrıca seri üretim sistemi, miktar ve akış tipi üretim diye ikiye ayrılmaktadır. Miktar tipi üretim sistemi, bir tip üründen çok fazla üretilmesi durumuyken, Akış tipi üretim ise uzun bir hat boyunca parçaların işlem gördüğü iş istasyonlarının, sıralanmış durumudur [11] Şekil 2.2’de akış tipi üretim sistemi gösterilmektedir.

Parti tipi üretim sisteminin geliştirilmiş bir hali olan GT’de, benzer parçalar bir araya getirilerek birbirinden bağımsız makine gruplarında bu parçaların işlenmesi sağlanır. Buradaki kriter her parça ailesinin ait olduğu makine grubunda işlenerek süreci tamamlamasıdır. Parçaların işlem gördüğü bağımsız alt makine gruplarına hücre adı verilir. Bağımsız hücreler arası taşıma mesafesinin azalması ve parçalara göre gerekli işlemlerin sadece ait oldukları hücrede yapılması sayesinde işletme içi stoklarda azalma meydana gelir, sistemde sırada bekleyen parçaların azalması sonucunda da ürün tedarik süresinde azalma durumu ortaya çıkar. Sonuçta işletmenin verimliliği artarak daha düşük maliyetlerde üretim yapma olanağı sağlanmış olur. GT felsefesinin aksine işletmenin atölye tipi üretim sistemini uygulaması durumunda, benzer fonksiyonlu makineler bir araya



Şekil 2.1 Atölye tipi üretim sistemi yerleşim planı [1]

getirilerek alt gruplar oluşturulur. Bu sistemde gelen her parça en az bir alt grupta işlem görmek durumunda kalır.



Şekil 2.2 Akış tipi üretim sistemi yerleşim planı [1]



Günümüzde GT kavramının önemi, süreç içi stok, üretim öncesi kurulum süresi, parça taşıma mesafesi ve parti büyüklüklerinde azalma sağlamasından dolayı artmıştır [12].

GT yerleşim planının avantajları ve dezavantajları aşağıda belirtilmiştir [13]:

Avantajlar:

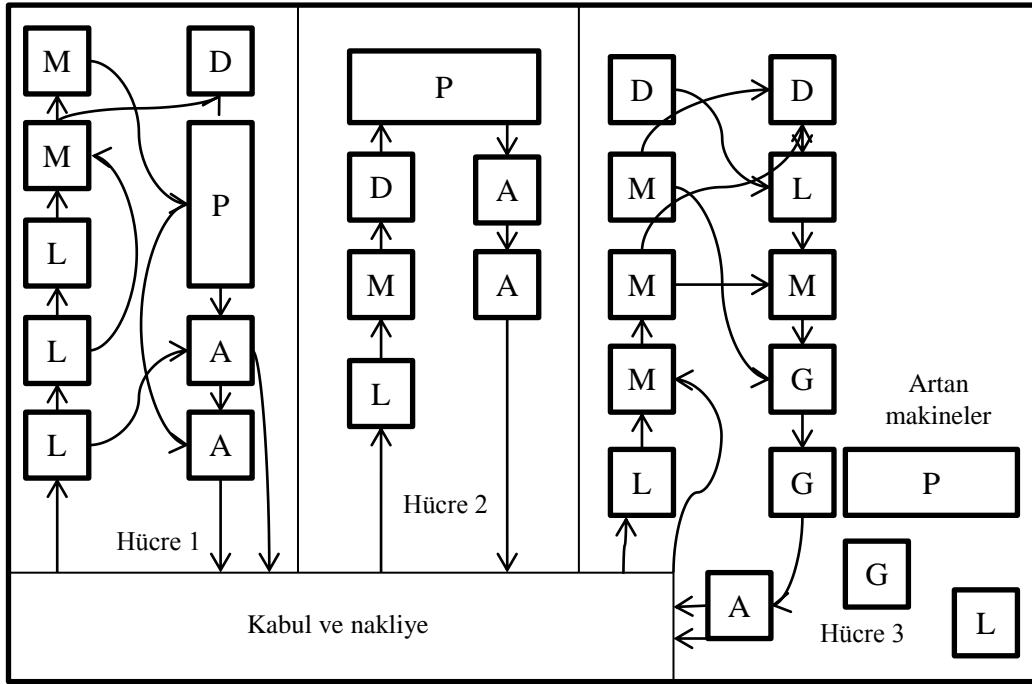
- Parçaların gruplanması sonucunda, makine kullanım oranı artar.
- Süreç odaklı yerleşim planları, ürünler için düzgün ve kısa mesafeli akış sağlar.
- Takım ruhunun gelişmesi ve üretim miktarındaki artış sık rastlanan faydalardandır.
- Atölye tipi ve akış tipi yerleşim planlarının avantajlarına sahiptir.
- Genel amaçlı makinelerin önemi artar.

Dezavantajlar:

- Genel denetim gereklidir.
- Takım elemanlarının tüm operasyonları gerçekleştirebilecek yeteneğe sahip olması gerekir.
- Hücre içi üretim kontrolünün dengeli olması gerekir.
- Eğer her hücrenin akışı dengeli değilse, hücre içi yarı mamul ve yedek ürün stoklarında artış meydana gelir.
- Atölye tipi ve akış tipi yerleşim planlarının dezavantajlarına sahiptir.
- Özel amaçlar için kullanılan makinelerin kullanım oranı düşer.

## 2.2. Hücresel Üretim

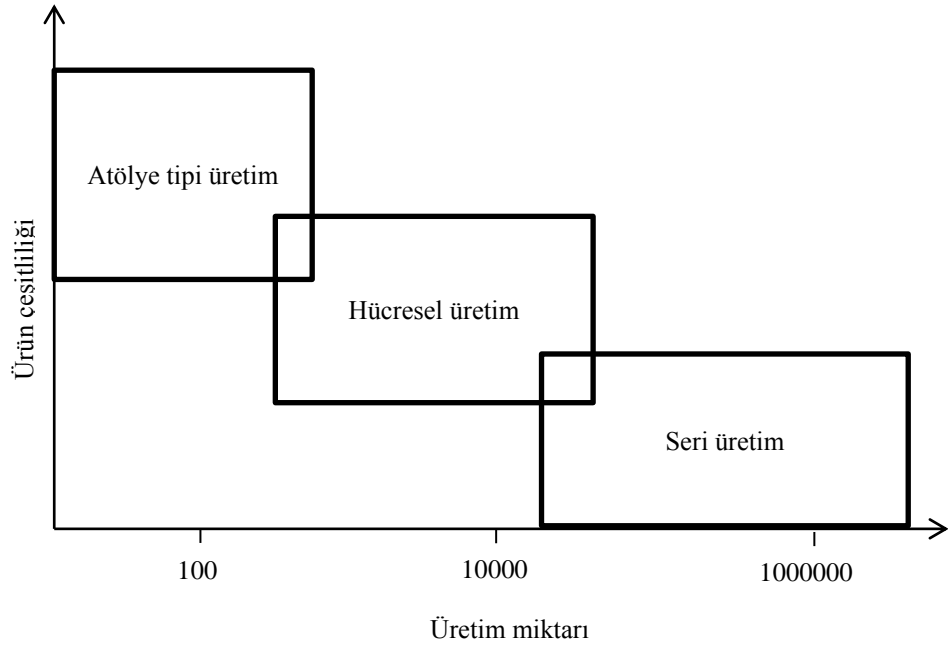
Hücresel üretim, GT'nin üretim alanındaki uygulamasıdır. Hücresel üretim benzer parçalar bir araya getirilerek oluşturulan parça ailelerinin, hücrelere atanan makinelerde işlenmesidir. Hücre oluşturma problemi ise üretim sistemini hangi kriterlere göre alt gruplara ayrılacağına belirlenmesidir. Parça ailelerinin görmesi gereken tüm işlemler makine grupları tarafından karşılanır. Hazırlık zamanında azalma, daha az makine gereksinimi, verimlilikte artış, işlemlerin daha iyi kontrol edilmesi vb. gibi GT'de yer alan avantajlar hücresel üretim için de geçerlidir.



Şekil 2.3 Hücresel üretim yerleşim planı

Genel dezavantajı; özel makine kullanım oranı düşer ve her zaman vasıflı bir iş gücü gerekir. Ayrıca hücrelerin bağımsız olması için üretimde kullanılan mevcut makine ve ekipmanlardan daha fazlası da gerekebilir, bu ise yatırımların artması anlamına gelmektedir [14]. Hücresel üretim sisteminin yerleşim planı Şekil 2.3 gösterilmektedir. Atölye tipi yerleşim planı ile karşılaştırıldığında parça hareketleri karmaşık bir yapıda değildir. Hücreler arası taşınım az veya hiç yoktur. Parçalar tüm işlemlerini hücre içinde tamamlayabilir. Üretimin kontrol altında tutulması çok daha kolay olmaktadır.

Hücreler oluşturulurken, parça ve makine gruplarını etkileyen çok fazla kriter vardır, örneğin hücreler arası taşımayı en küçükleme bir kriter olarak alınabilir. Üretim endüstrisinde yer alan işletmeler rekabet düzeylerini arttırmak için fiyat, dağıtım zamanı, kurulum zamanının azaltılması gerekirken ürün çeşitliliğini de arttırmak zorundadır. İşletmeler hangi hücre oluşturma kriterine göre üretim yapacağını iyi analiz etmelidir; çünkü bu kriter etkili bir iş performansını doğrudan etkilemektedir [15].



**Şekil 2.4** Farklı Düzeyde Üretim Miktarı ve Ürün çeşitliliği için Yerleşim Tipleri ve Üretim Sistemleri [16]

Şekil 2.4’de HÜS’nin diğer üretim sistemlerine göre nerede yer aldığı ürün çeşitliliği ve üretim miktarı göz önüne alınarak gösterilmektedir.

### 2.2.1. Hücresel üretimin avantajları

Hücre sel üretim sisteminin başlıca faydaları, tedarik, kurulum ve taşıma sürelerinin, süreç içi stokların, parti büyüklüğünün azalmasıdır. HÜS avantajları aşağıda ayrıntılı olarak açıklanmaktadır [1, 11, 16-18]:

- Üretim süresi kısalmır: Hazırlık, taşıma süresinde ve parti büyüklüğünde azalma olmasından dolayı toplam üretim süresi azalmaktadır. Bu yönüyle akış tipi üretimin avantajından yararlanılmaktadır.
- Süreç içi yarı mamul ve bitmiş ürün stoğu azalır: Parti büyüklüğünde azalma ve üretim süresindeki azalmadan dolayı stok büyüklüğünde de azalma meydana gelir.
- Üretim çizelgelemede kolaylık sağlar: Parça ailelerindeki benzerlikten dolayı üretim çizelgelemedeki karmaşıklık azalır. Atölye tipi üretim

sisteminde üretim çizelgeleme, tüm süreci kapsayacak şekilde yapılırken HÜS çizelgeleme sadece hücre içi süreci kapsayacak şekilde yapılmaktadır.

- Hazırlık süresinde azalma olur: Parçalar benzerliklerine göre gruplandırıldığından, benzer makine ve ekipmanlar kullanılır. Süreçte benzer biçimlendirmeler yapılacağından hazırlık süresi azalır. Böylelikle süreç sadece her bir parça için hazırlanmamakta, parça ailesi için hazırlanmaktadır. Makinelerde gerekli değişiklikler parça ailesi için olacağından, hazırlık zamanından kazanç sağlanır.
- Süreç kontrolü artar: HÜS'te parçalar Akış Tipi Üretim sisteminde olduğu gibi bir hat boyunca hareket eder. Her bir hücre içi akış, bir hattı temsil ettiğinden kontrol daha rahat sağlanır.
- İşçilik maliyetleri azalır: Hücrelere atanan elemanlar, birden fazla işi hücre içinde yapabildiğinden işçilik maliyeti azalır.
- Malzeme taşıma azalır: Parçaların izleyecekleri işlem rotaları hücre içindeki makineler tarafından karşılandığından, kat ettikleri mesafe ve harcadıkları süre azalmaktadır.
- Parti büyüklükleri azalır: Hazırlık zamanının azalması sonucunda küçük parti büyüklükleri ile çalışma sağlanabilir böylece üretim akışı da daha düzgün bir hale gelmektedir.
- Üretimde kullanılan ekipmanlarda azalma olur: Hücre içinde üretilen parçalar şekil, nitelik, büyüklük yönünden birbirlerine benzediklerinden aynı ekipmanlar kullanılır.
- Gerekli alan azalır: Üretimdeki yarı mamul, bitmiş ürün stoklarında ve parti büyüklüğündeki azalmadan dolayı işletme içi yerleşimde alan tasarrufu sağlanmış olur. Böylece önemli ölçüde kullanılabilir alan kazanılır.
- Ürün kalitesi artar: Ürünlerin dar bir alanda dolaşmasından dolayı hatalı ürüne ilişkin geri dönüşler çok hızlı olur bu da hataların hızlı giderilmesinde kolaylık sağlar.

- Yatırımın geri dönüşü hızlıdır: İşletme içi yerleşimin HÜS'e göre yapılandırılması zaman alıcıdır; fakat sistemde stok, taşıma, kalite, makine kullanımındaki verimliliğin artması sonucu bu maliyet karşılanabilir.

### 2.2.2. Hücresel üretimin dezavantajları

Hücresel üretimin dezavantajları aşağıda verilmektedir [1]:

- Uygulama maliyeti yüksektir: Mevcut yerleşim planının tekrar düzenleme maliyeti yüksektir. Ayrıca fiziksel düzenleme, işçilerin eğitim ve özel vasıf kazandırma maliyeti de ortaya çıkmaktadır.
- Makine arızalarını ve kullanım oranlarını yönetmek zordur: Hücre içindeki her bir makine üretim faaliyetlerini yerine getirmek için kritik düzeyde öneme sahiptir. Bir makinede meydana gelen arıza diğer makinelerin de durmasına neden olur. Üretimin durmaması için bu faaliyet başka bir hücredeki makineye kaydırılabilir; fakat bu durumda da hücreler arası taşıma mesafesi artmaktadır. Ayrıca aynı işi yapan alternatif bir makinenin seçim işlemi hücre oluşturmayı daha zor bir hale getirmektedir.
- İşgücünün özelleştirilmesi: Hücrelerde yer alan ve farklı faaliyetleri yerine getiren makinelerin amacı hücreler arası taşıma mesafesini en küçükmektir. Aynı zamanda işçilerin dolaşım mesafesinin artmasından da kaçınmak gerekmektedir. Bu nedenle işçiler hücrede yer alan tüm makineleri kullanabilecek şekilde eğitilir; bu da işgücü maliyetini artırır.
- Parçalar montaj aşamasına eş zamanlı gelmeyebilir: Parçalar farklı hücrelerde üretiliyor olabilir. Böylece montaj işlemine alınacak parçalar aynı anda işleme alınamamaktadır.
- Hücre dışı faaliyetlerin kontrolü zordur: Çoğu durumda parçaların sadece bir tek hücrede işlemini tamamlaması zor olduğundan, başka bir hücrede de işlem görmesi gerekebilir. Bu durumda ise parça taşınımına ilişkin problemler çıkabilir.

- Makine kullanım oranı azalır: Bağımsız hücreler oluşturmak amacıyla ekstra makinelere gereksinim duyulabilir. Makine sayısındaki artış nedeniyle ortalama makine kullanım oranı da azalır.
- Esnek olmamak: Hücreler, farklı tipteki parçaları işleyebilmesi için, enbüyük esnekliği sağlayacak şekilde tasarlanmalıdır; fakat bu gerçekte yapılamamakta ve hücre esnekliği limitli kalmaktadır.
- Parça değişikliklerini yerleşim planına yansıtma zordur: Parçalarda meydana gelen değişiklikler sonrasında üretim hücrelerinin yerleşim planlarını düzenlemek kolay değildir. Böyle bir durumda bazı hücrelerde darboğazlar oluşacak, bazıları ise kullanım oranlarını çok altında kalacaktır.

Tüm bu dezavantajlar hücresel üretimin uygulanmasını zorlaştırır da avantajlar üretim verimliliğine önemli derecede olumlu yönde etki etmektedir.

### 2.2.3. Hücresel üretim sistemi uygulamaları

HÜS, atölye tipi üretim sistemi ile akış tipi üretim sisteminin özelleştirilmiş bir halidir. HÜS, üretim zamanının ve süreç içi stokların azaltılmasını sağlayarak maliyetlerin düşmesinde büyük payı olmaktadır. Fakat bu sistemin kurulabilmesi için: yerleşim planı tekrar düzenlenir, çalışanlara eğitim verilir bununla birlikte bağımsız hücrelerin oluşturulabilmesi için gerekli ekipman alımlarına gidilir. Tüm bunlar HÜS uygulamak isteyen endüstrilere ciddi yükler getirmektedir.

HÜS'nin akademik ve endüstriyel dünyada büyük ilgi görmesinden dolayı, bu üretim sisteminin uygulamadaki yerini tespit etmek için Wemmerlov ve Johnson [19] tarafından bir anket çalışması yapılmıştır. Bu çalışma A.B.D.'de faaliyet gösteren 46 işletmeye uygulanmış, işletmelerin HÜS'ü kullanma nedenleri, sistemin getirileri ve olumsuz yönleri ortaya konulmuştur. İşletmelerin HÜS kurma nedenleri, ortalama önem dereceleri ve sayısal tahminlere göre beklenen ortalama gelişim oranları Çizelge 2.1'de verilmektedir.

Ortalama önem derecesi enbüyük olan ilk 5 nedene bakıldığında, birbirleriyle ilişkili olduğu görülmektedir. Ayrıca bu nedenler üretimdeki verim-

**Çizelge 2.1** A.B.D.’deki işletmelerin HÜS kurma nedenleri ve beklenen gelişim oranları

Sıra No	Neden	Ortalama önem derecesi	Beklenen ortalama gelişim oranı
1	Akış zamanını azaltmak	4.51	%61.2
2	Süreç içi stokları azaltmak	4.33	%48.2
3	Parça/ürün kalitesini arttırmak	4.22	%28.4
4	Müşteri için tedarik süresini azaltmak	4.22	%50.1
5	Taşıma mesafesini ve süresini azaltmak	4.24	%61.3
6	Üretimde esnekliği arttırmak	3.82	-
7	Birim maliyetleri azaltmak	3.80	%16.0
8	Üretim planlama ve kontrolünü basitleştirmek	3.62	-
9	Sistemdeki çalışan karmaşıklığını basitleştirmek	3.57	-
10	Hazırlık zamanını azaltmak	3.43	%44.2
11	Bitmiş ürün stokunu azaltmak	3.41	%39.3

Not: Her bir neden 1-5 aralığında derecelendirilmektedir. 1=Önemsiz, 5=çok önemli anlamına gelmektedir.

liliği zaman yönünden arttırmaya yöneliktir. İşletmelerin HÜS’ü kurmaları doğrultusunda beklenen gelişmeler, taşıma mesafesi ve üretim zamanı ile ilgilidir.

Durmuşoğlu ve ark. [20] HÜS’ün Türkiye’de değerlendirilmesi için bir saha çalışması yapmıştır. Bu çalışmaya konu olan sanayi türleri; otomotiv yan sanayi, makine sanayi, elektrik-elektronik sanayi, savunma sanayi ve tarım makineleri sanayidir. Türkiye’deki işletmelerin HÜS’e geçme nedenleri, ortalama önem dereceleri ve bu sisteme geçme sonrası beklenen gelişim oranları Çizelge 2.2’ de gösterilmektedir.

A.B.D. ve Türkiye’deki işletmelerin hücresele üretime geçişte en fazla önem verdikleri neden akış zamanını azaltmaktır. A.B.D.’deki işletmeler akış zamanını %61.2 oranında azaltmayı hedeflerken Türkiye’deki işletmeler bu oranı %31 olarak belirtmektedir. A.B.D.’de beklenen eniyi gelişim oranı %61.3 ile taşıma mesafesini ve süresini azaltmak iken, Türkiye’deki eniyi beklenen gelişim oranı %53 ile süreç içi stokları azaltmaktır.

**Çizelge 2.2** Türkiye’deki işletmelerin HÜS kurma nedenleri ve beklenen gelişim oranları

Sıra No	Neden	Ortalama önem derecesi	Beklenen ortalama gelişim oranı
1	Akış zamanını azaltmak	4.51	%31
2	Müşteriye tepki süresini kısaltmak	4.47	%50
3	Taşıma mesafelerini / sürelerini azaltmak	4.29	-
4	Parça üretim maliyetini düşürmek	4.24	%10
5	Hazırlık süresini düşürmek	4.24	%29
6	Süreç içi stokları azaltmak	3.93	%53
7	Üretim miktarını arttırmak	3.76	%48
8	Hatalı parça / mamul oranını azaltmak	3.56	-
9	Üretim alanından tasarruf yapmak	3.34	%40
10	Talebi karşılama oranını arttırmak	-	%10

Not: Her bir neden 1-5 aralığında derecelendirilmektedir. 1=dikkate alınmadı, 2=düşük düzeyde önemli, 3=orta düzeyde önemli, 4=önemli, 5=çok önemli anlamına gelmektedir.



### 3. HÜCRE OLUŞTURMA PROBLEMİ VE FARKLI TÜRLERİ

HÜS'ün en önemli adımı Hücre Oluşturma Problemi'dir (HOP). Hücrelerde yer alacak makine gruplarının ve parça ailelerini belirlenmesi NP-Zor problemidir [7]. Bununla beraber problem boyutu büyüdükçe çözüm işlemi karmaşık bir hal alırken, makul zamanlarda çözüm bulmak da gittikçe zorlaşmaktadır.

Literatürde HOP farklı şekillerde ele alınmıştır: alternatif rota, talep değişimi, iş gücü dengeleme, makine sayısı, işlem sıralama, makine kapasite. durumları kısıt olarak alınırken, bunun yanında kurulum maliyeti, parça taşıma maliyeti, makine yatırım maliyeti, birim üretim maliyeti gibi kriterlerde problemin amaçları olarak ele alınmaktadır.

#### 3.1. Hücre Oluşturma Problemi

Hücreyel üretim sistemi, tüm sistemin alt bölümlere ayrılarak yönetim ve kontrolün daha verimli yapıldığı bir GT uygulamasıdır. Parça ailelerinin sınıflandırılması, sahip oldukları benzer özelliklere göre yapılmaktadır. Bu benzer özelliklerden kasıt işlem planları ve makine tiplerindeki benzerliktir. Bu adımlar gerçekleştirildiğinde üretim sistemi basit ve bütünleşik bir yapıya sahip olmaktadır. HÜS'e geçişteki en önemli adım benzer parça ve makine gruplarının belirlenmesidir, bu işlem Hücre Oluşturma Problemi olarak ele alınmaktadır. HOP'da benzer parçaların bir araya getirilmesinde göz önüne alınan özellikler şu şekildedir: geometrik şekil, üretim gereksinimleri ve üretim akışıdır. Her bir üretim hücresi bağlı bulunduğu parça ailesini işleyebilecek makine, ekipman ve işgücüne sahip olmaktadır [21].

Tasarım özelliklerini ve parça özelliklerini temel alarak, büyük üretim sistemlerini kolay yönetilebilir üretim sistemlerine ayıran yaklaşımlar geliştirilmiştir. Tüm bu yaklaşımları geniş anlamda iki kategoriye ayrılması mümkündür:

- Sınıflandırma yaklaşımları; kodlama sistemini kullanmaktadır.
- Hücre oluşturma yaklaşımları; üretim akış bilgilerini kullanmaktadır.

Sınıflandırma yaklaşımları görsel inceleme metodu ve kodlama metodu olmak üzere ikiye ayrılır [9].

Görsel inceleme metodu parçaları geometrik benzerliklerine göre gruplara ayırır. Parçaların fiziksel özellikleri, teknik resimleri benzer parça ailelerini oluşturmak için kullanılmaktadır. Kodlama metodunda parça özelliklerine göre kodlar verilmektedir. Her bir parça için tasarım, üretim ya da her ikisinin birlikte kullanıldığı kodlar vardır. Bunun yanında üretim akış metodunda ise parçaların hangi makinelerde işlendiği bilgisi kullanılmaktadır.

HÜS'ün tasarımında en önemli problem hücre oluşturma problemidir. Hücre oluşturma sırasında üç genel amaç vardır: parça ailelerinin oluşturulması, makine hücrelerinin oluşturulması ve parça ailelerinin makine hücrelerine atanması. Hücre oluşturma sırasında dikkate alınan çok fazla faktör vardır: makine gereksinimleri, makine hazırlık zamanı, makine kullanım oranı, iş gücü, alternatif rotalar, makine kapasiteleri, işlem sırası, hücre yerleşimi bunlardan bazılarıdır. Bahsedilen bu faktörlerin temel alındığı hücre oluşturma problemlerinin çözümü için birçok yöntem geliştirilmiş olmasına rağmen, üretim akış bilgilerini kullanan çözüm yöntemleri genel olarak: ikili makine-parça matrisi yöntemi ve kümeleme yöntemidir. İkili makine-parça yöntemi ilk defa 1980 yılında King tarafından önerilmiştir. Bu yöntem 0-1'lerden oluşan temas matrisini biçimlendirmektedir. Temas matrisinin  $i$ . satır ve  $j$ . sütununda yer alan 1 değeri,  $i$ . parçanın  $j$ . makinede işlendiğini gösterirken, 0 değeri,  $i$ . makinenin  $j$ . parçayı işleyemeyeceğini belirtmektedir. Yöntemin amacı temas matrisinde yer alan 1'leri düzenleyerek blok köşegen bir matris formuna dönüştürmektir. Böylece makinelerin yer aldığı hücreler ve buralarda işlem görecekt parçalar belirlenmiş olur. Diğer bir yöntem olan kümeleme yöntemi ise 1972'de McAuley tarafından önerilmiştir. Kümeleme yönteminin temeli parçaların makine gereksinim katsayılarındaki benzerlikleri kullanmasıdır. Bu yöntemde makine gereksinim benzerlik katsayıları yüksek olan parçalar bir araya getirilir [22].

HÜS tasarımındaki genel yaklaşımlar:

- Parça aileleri işlem gereksinimlerine göre bir araya getirilir.
- Makineler üretim hücrelerinde gruplanır.
- Parça aileleri makinelere atanır.

Yukarıda bahsedilen adımların sırasıyla gerçekleşmesi beklenmemektedir. Bahsedilen yaklaşımlara bağlı kalarak oluşturulacak hücre ve makine grupları için 3 tip çözüm tanımlanmıştır:

- Öncelikle parça aileleri oluşturulur ve bunlara bağlı kalarak makineler hücrelerde gruplanır. Bu strateji parça ailesi temelli yaklaşım olarak tanımlanmaktadır.
- Öncelikle parçaların işlem rotalarına bakılarak makine grupları oluşturulur ve parçalar bu gruplara sonradan atanır. Bu strateji de makine grup temelli yaklaşım olarak tanımlanmaktadır.
- Parça aileleri ve üretim hücreleri eş zamanlı olarak oluşturulmaktadır. Diğerleri gibi bu strateji de parça aile/makine grup temelli yaklaşım olarak tanımlanmaktadır [23].

Daha öncede belirtildiği gibi parçaların hangi makinelerde işlem gördüğünü gösteren matrise temas matrisi denmektedir. Şekil 3.1(a) ilk satırda makine isimleri, ilk sütunda da parça isimlerinin olduğu bir temas matrisi verilmektedir. Matris incelendiğinde 3. parça (P3) 2. (M2), 4. (M4) ve 5. (M5) makinelerde işlem görmektedir. Şekil 3.1(b) matrisi temas matrisinin blok köşegen halidir. Parça ailelerinin ve makine gruplarının eş zamanlı oluşturulması sonucunda iki adet hücre ortaya çıkmaktadır. M1 ve M3 makineleri birinci hücrede yer almakta ve P1, P2, P4, P7 parçaları da bu hücrede işlenmektedir. Diğer hücrede M2, M4, M5 makineleri yer alırken, hücreye atanan parçaların P3, P5, P6 olduğu görülmektedir.

	M1	M2	M3	M4	M5
P1	1		1		
P2	1		1		
P3		1		1	1
P4	1		1		
P5		1			1
P6				1	1
P7	1		1		

(a)

	M1	M3	M2	M4	M5
P1	1	1			
P2	1	1			
P4	1	1			
P7	1	1			
P3			1	1	1
P5			1		1
P6				1	1

(b)

Şekil 3.1 Temas matrisi (a), köşegenleştirilmiş temas matrisi (b)

Blok köşegen matriste, tüm 1 değerleri hücrelerin içinde yer almaktadır. Dışarıda 1 değeri olmadığından, gelen parçalar tüm işlemlerini hücre içinde tamamlamakta ve böylece hücreler arası taşıma olmamaktadır.

### 3.2. Alternatif Rotalı Hücre Oluşturma Problemi

Grup teknolojilerinin asıl amacı, üretim sistemini özelliklerine göre parçalara ayırarak, benzer parçaların işlendiği makinelerin bir arada bulunduğu hücreler oluşturmaktır. Böylece üretim, stok tutma ve taşıma maliyetlerinde azalma sağlanırken üretim yönetim sistemi de basitleştirilmektedir. Bazı üretim sistemlerinde, üretim planı oluştururken karşılaştırılabilir farklı planlar mevcut olabilir. Alternatif rotaların varlığı durumunda pratik üretim planlarının seçimi yapılabilir. Alternatif rotalı üretimde, parçaların izlediği rotalardan biri seçilmektedir. Bu planların seçilme nedenlerinden biri de benzer fonksiyonlara sahip işlem merkezleri kurmaktır. Bir parçanın iki tezgahda da işlenebilir olduğu bir durumda, parça işlem merkezleri dışında kalmayacak şekilde herhangi bir alternatif yolu izleyebilir [24].

Bir parçanın alternatif rotalara sahip olması için ya bir makineden birden fazla olması (kopya makine) ya da parçanın farklı makinelerde işlenerek üretilebilir olması gerekir [25].

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

Pa çalar	Makineler					
	1	3	5	2	4	6
1	1	1	1	1		
	1	1	1			
5		1	1	1		
2			1	1	1	1
3				1	1	1
6				1	1	1

Şekil 3.2 Alternatif rota olmadığı durumdaki matriste blok köşegenleşme

Temas matrisinin blok diyagonal bir forma getirilmesi durumu alternatif rotaların olması durumunda daha iyi sağlanmaktadır [26].

Alternatif rotaların bulunmadığı altı parça ve altı makine için oluşturulan temas matrisi Şekil 3.2’de gösterilmektedir. Örneğin 5. parça sadece bir rota izleyebilmekte, bu durumda 2, 3, 5 numaralı makinelerde işlem görmektedir. Alternatif rotaların bulunmadığı durumda hücreler oluşturulduğunda birinci hücrede 1, 3, 5 makineleri yer almakta, diğer hücrede ise 2, 4, 6 makineleri yer almaktadır. Oluşturulan hücrelerin dışında kalan istisnai eleman sayısı üç iken, hücre içi boşluk sayısı da birdir.

Şekil 3.3’de beşinci parçanın iki alternatif rotası olduğu varsayılmıştır. Alternatif rotaların ele alındığı bu durumda yine iki hücre oluşturulmakta ve alternatif rota olmayan duruma göre hücre dışında kalan istisnai parçaların ikiye düştüğü görülmektedir. Parçaların farklı makinelerde işlem görebildiği durumu temsi eden bu örnek, alternatif rotaların olması durumunda daha etkin hücrelerin oluşturulacağını göstermektedir.

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

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

Şekil 3.3 Alternatif rotaların olduğu durumdaki matriste blok köşegenleşme

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

Şekil 3.4 Kopya makinelerin olması durumdaki matriste blok köşegenleşme

Şekil 3.4’de ikinci makineden iki kopya olduğu varsayılmıştır. Bu durumda hücre dışında kalan iki makinesi her iki hücrede de yer alabilecektir. Kopya makine olması durumunda oluşturulan hücrelerde iki boşluk yer alırken, hücre dışı istisnai eleman sayısının da bire düştüğü görülmektedir. Daha öncede bahsedildiği gibi parçaların farklı makinelerde işlem görmesi veya kopya makine olması durumunda oluşacak alternatif rotanın tasarlanan hücrelerde etkileri açıkça gösterilmektedir.

### 3.3. Literatür Taraması

Son 30 yıla bakıldığında hücre oluşturma problemine olan ilginin arttığı görülmektedir. Araştırmacılar talep değişimi, makine kapasitesi, alternatif rota, işlem sıralama gibi kısıtları dikkate alarak hücreler arası taşıma mesafesi, hücre içi taşıma mesafesi, makine hazırlık maliyeti, yerleşim maliyeti gibi amaçları en küçüklemek için bu problemi çözmeye çalışmaktadırlar. Bu problemin ilk çözüm yaklaşımları parçaların görsel özelliklerini dikkate alarak ve tasarım özelliklerine göre verilen kodların birbirine benzer olanlarının gruplandırılması şeklindedir. Bu yaklaşımlar parça sayısının çok fazla olmadığı durumlarda kullanılması mümkündür. Üretim akış bilgilerini kullanarak çözüm arayan kümeleme, matematiksel programlama, sezgisel, metasezgisel ve yapay zeka yaklaşımları da sıklıkla kullanılmaktadır.

Hücre oluşturma problemleri NP-zor problemler sınıfında yer alır [7], orta ve büyük boyuttaki problemlerde eniyi çözüme makul zamanlarda ulaşmak kolay olmayabilir. Bu yüzden literatürde sezgisel ve metasezgisel yaklaşımlara olan ilgi artmaktadır.

Papaiouannou ve Wilson [23], 1997-2008 yıllarında hücre oluşturma problemlerini ele alan çalışmaların geniş bir araştırmasını yapmıştır. İlgili çalışmada matematiksel programlama, sezgisel, metasezgisel ve yapay zeka yöntemleri ile probleme ilişkin çözüm yaklaşımları değerlendirilmiştir. Ek-1’de makalelerde ele alınan kısıtlar, amaçlar ve çözüm yaklaşımları ayrıntılı olarak karşılaştırılmaktadır.

Bu çalışma kapsamında, alternatif rotaların varlığında hücre oluşturma problemi, Parçacık Sürü Optimizasyonu ve Sınır Ağları yöntemlerinin birlikte kullanıldığı bir yaklaşımla çözülmektedir. Bu nedenle bahsedilen yöntemlerin kullanıldığı çalışmalar incelenmiştir.

Lee ve ark. [27], üretim hacmini, alternatif rotaları ve işlem sırasını göz önüne alarak hücre oluşturma problemini çözmüştür. Çalışmada eniyi alternatif rotalar seçilerek, maliyeti en küçükmeye yönelik bir model kurulmuştur. Malzeme taşıma mesafesini en küçükmeyi amaçlayan bu model için genetik algoritma ile çözüm aranmıştır. GT alanında bu kriterleri ele alarak çözüm arayan ilk çalışmadır.

Lozano ve ark. [28], hücre oluşturma problemini tasarım ve yükleme problemi olarak ayırmıştır. Hücre tasarım problemi için 2 alternatif önermişlerdir: ilki her bir parça tipi için alternatif planın parça tiplerine göre ayrıldığı, ikincisi ise parça tiplerine göre birleştirildiği yaklaşımlardır. Hücre yükleme problemi çok dönemli ve taleplerin de dikkate alındığı doğrusal programlamayla modellenmiştir. Modelin amacı her bir dönemde, hangi alternatif rotayı kullanarak, her bir parçadan ne kadar üretilerek stok ve taşıma maliyetlerini en küçükmek şeklindedir.

Caux ve ark. [29], alternatif süreç planlarının ve makine kapasitelerinin dikkate alındığı bir çalışma yapmıştır. Üretilecek parça sayısını ve makinelerin üretim kapasitesini dikkate alarak, alternatif işlem planlarının içinden en uygununun seçimi ve hücreler arası taşıma mesafesinin en küçükmemesi amaçlanmıştır. Hücre oluşturma problemi çözümü için tavlama benzetimi yöntemi ve alternatif rota seçimi için de dal-sınır yöntemi birleştirilmiştir.

Spiliopoulos ve Sofianopoulou [30], kopya makinelerle alternatif rota oluşturan ve bunun yanında parçaların alternatif süreç planlarının olduğu

problemler için çözüm aramıştır. Önerdikleri yaklaşımda sınır şeması oluşturularak tüm alternatif rotalar incelenmekte ve sadece birkaçı için çözüm aranmaktadır. Böylece çözüm uzayı sezgisel olarak daraltılmaktadır. Bu çalışmada da hücreler arası taşıma mesafesi en küçüklenmek istenmiştir.

Sormaz ve Rajaraman [31], eş zamanlı alternatif rota seçimi ve hücrelerin oluşturulduğu bir uzay arama algoritması önermiştir. Hücreler arası taşıma mesafesi en küçüklenmek istenirken, hücrelerde yer alacak makine sayıları kısıt olarak modele eklenmiştir. Literatürde yer alan benzer problemlerle karşılaştırma yapılarak, diğer yöntemlere göre hızlı çözüm bulunduğu iddia edilmiştir.

Susanto ve ark. [32], işlem sırasının, talep miktarının ve alternatif rotaların dikkate alınarak c-ortalama kümeleme algoritması ile parça ailelerini ve makineleri gruplamıştır, Hungarian metodu ile hücreler oluşturulmuştur. Bahsedilen kısıtlar göz önüne alınarak hücreler arası taşıma mesafesini en küçüklemek için doğrusal programlama modeli kurulmuştur.

Wu ve ark. [33], tavlama benzetimi ve genetik algoritma mutasyon operatörünü birlikte kullanarak alternatif rotalı hücre oluşturma problemleri için melez bir yaklaşım önermiştir. Kurulan modelin amaç fonksiyonu hücreler arası taşıma mesafesini en küçüklemektir, dolayısıyla hücre değerlendirmesi için kullanılan gruplama etkinlik değeri enbüyüklenmek istenmiştir.

Chung ve ark. [34], alternatif rotalı hücre oluşturma problemlerinin çözümü için benzerlik katsayılarını temel alan yasaklı arama yöntemini kullanmıştır. Önerilen yaklaşım, iyi çözümler üretilerek başlar, sonra da yasaklı arama ve mutasyon operatörü birlikte kullanılarak arama yapılmaktadır. Literatürde yer alan matematiksel programlama yaklaşımları ile karşılaştırma yapılmıştır.

Saraç ve Özçelik [25], hücre oluşturma probleminin alternatif rotalarla ele alındığı bir çalışma yapmıştır. Yaklaşımda yeni geliştirilen bir Genetik Algoritma yöntemi önerilmiştir. Diğer yöntemlerden farklı olarak algoritma, gruplama etkinliğini enbüyükleyen hücre sayısını da kendi belirlemektedir. Literatürde yer alan farklı problemlerde test edilerek başarılı sonuçlar elde edilmiştir.

Diaz ve ark. [46], alternatif rotalı hücre oluşturma problemleri için yaptıkları çalışmada, taşıma maliyetini en küçüklenmek istemişlerdir. Hücrelere atanacak makine sayısı, bazı makinelerin hücrelere birlikte atanması/atanmaması



kısıtları modele dahil edilmiştir. Problem çözümü için yasaklı arama yöntemi önerilerek literatürde yer alan büyük problemler için test edilmiştir. Yasaklı arama yönteminin başarısı tavlama benzetimi ile karşılaştırılmıştır.

### 3.4. Hücre Oluşturma Problemi Çözüm Yöntemleri

Yapılan çalışmalarda hücre oluşturma problemleri için etkili yaklaşımlar önerilmiştir. Bu yaklaşımları 3 kategoride toplamak mümkündür [35]:

- Görsel inceleme metodu
- Parça kodlama metodu
- Üretim akış metodu

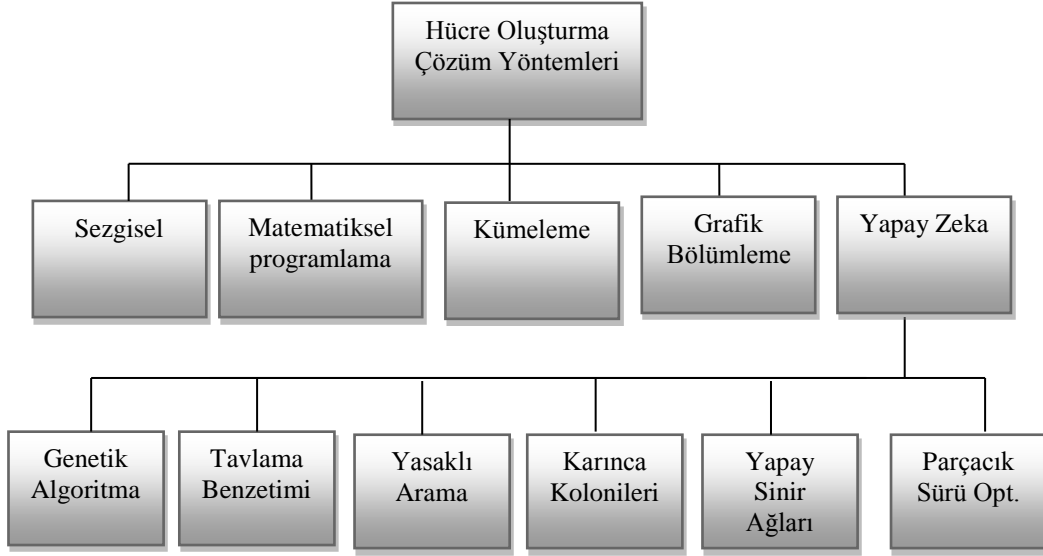
Görsel inceleme metodu, parça çeşitliliğinin az olduğu durumlarda kullanılmaktadır. Parçaların geometrik benzerliklerine göre parça aileleri oluşturulmaktadır. Bu işlem tecrübeli bir kişi tarafından geçmiş deneyimlere bağlı kalarak yapılmaktadır, bu yüzden pahalı bir yöntemdir.

Parça kodlama metodunda, parçaların tasarım özellikleri kullanılır. Parça karakteristiklerine sayısal ağırlıklar atanarak bazı kümeleme teknikleri kullanılması sonucu parça aileleri bir araya getirilmektedir.

Üretim akış metodu, en fazla çalışma yapılan araç olarak literatürde yer almaktadır. Parçaların işleneceği makineleri temel alır, bir başka deyişle parçaların işlem rotaları göz önüne alınarak analiz yapılmaktadır. Bu yaklaşım, parça ve makine gruplamasını daha hızlı ve etkili yapmaya olanak sağlamaktadır. Üretim akış metodunun kendi içindeki yöntemleri aşağıdaki gibidir. Hücre oluşturma çözüm yöntemleri Şekil 3.5’de gösterilmiştir [23]:

- Sezgisel yöntemler
- Matematiksel programlama
- Kümeleme analizi
- Grafik bölümlenme yaklaşımı
- Yapay Zeka yöntemleri

Çalışmanın bu kısmında literatürde çalışma konusu olan hücre oluşturma çözüm yöntemleri hakkında bilgi verilmektedir.



Şekil 3.5 Hücre oluşturma çözüm yöntemleri

### 3.4.1. Sezgisel yöntemler

Sezgisel yöntemler kısa sürede iyi çözümler vermeyi garanti ederken eniyi çözüme ulaşmayı garanti etmeyen yaklaşımlardır. Literatürde hücre oluşturma problemleri için birçok sezgisel algoritma vardır.

Liu ve ark. [36], gerçek hayat problemlerini daha iyi temsil etmesi için alternatif rotaları, hücre büyüklüğünü, üretim hacmi kriterleri ile hücreler arası/içi taşıma maliyetlerini temel alan bir matematiksel model kurmuştur. Önerilen sezgisel yaklaşımda: (1) her parçanın izleyebileceği alternatif rotaları da göz önüne alarak geçici bir makine grubu oluşturulur, (2) her parçanın alternatif rotaları için taşıma maliyetleri sıralanır, (3) uygun rotalara göre hücreleri düzenlenir.

### 3.4.2. Matematiksel programlama yöntemi

Yapılan çalışmalarda, hücre tasarımı için kapasite, talep, hücrelerdeki makine sayısı gibi, önemli olan birçok kısıt göz ardı edilmiştir. Bu durumun başlıca nedeni, problem çözümü için düşünülen tekniğe bu kısıtların kolayca eklenememesidir. Matematiksel programlama yöntemi farklı kısıt ve amaçları ele alabildiğinden hücre oluşturma problemlerinin çözümünde iyi bir alternatiftir.

Literatürde işlem sıralama, alternatif rota, yeni ekipman, parti büyüklüğü, darboğaz maliyeti ve hücre dışı makine faktörleri modele dahil edilerek matematiksel programlama ile çözüm aranmıştır. Matematiksel programlama yöntemi 4 ana sınıfa ayrılmaktadır [1]:

- Doğrusal programlama
- Doğrusal ve karesel tam sayılı programlama
- Dinamik programlama
- Hedef programlama

Yöntemin iyi bir alternatif olmasının yanında, kullanımı sırasında bazı kısıtları da göz ardı etmemek gerekir. Öncelikle, amaç fonksiyonu doğrusal olmayan modellerde genellikle makine ve parça atamaları eş zamanlı yapılamamaktadır. Bir diğer kısıt ise karar değişkenlerinin tam sayılı olması nedeniyle büyük boyutlu problemlerin çözümüne makul zamanlarda ulaşılamamaktadır [1].

Hücre oluşturma problemlerinin çözümü için farklı sezgisel yaklaşımlar önerilmiştir. Optimum değeri vermeyen bu yaklaşımlara karşı Adil ve ark. [26], doğrusal olmayan bir model kurmuştur. Çalışmada parça ve makineler hücrelere eş zamanlı atanmaktadır. Adil ve ark. geliştirdiği doğrusal olmayan matematiksel modeli temel alarak oluşturulan model aşağıdaki gibidir [25]:

**İndisler:**

$p$ :parça ( $p=1,2, \dots,P$ )

$t$ :tezgah tipi ( $t=1,2,\dots,T$ )

$h$ :hücre ( $h = 1,2,\dots,H$ ),  $H \in [1,T]$  (hücre sayısı en fazla, tezgah sayısı kadar olabilir)

$r$ :rota ( $r=1,2,..Rp$ )

**parametreler:**

$a_{ptr} = \begin{cases} 1 & \text{p parçasının r alternatif rotasında t tezgahı kullanılıyor} \\ 0 & \text{dd} \end{cases}$

$N_t = t$  tipi tezgah sayısı

$w_1 =$  amaç fonksiyonunda hücre dışı eleman sayısı teriminin ağırlığı

$w_2 =$  amaç fonksiyonunda kullanılmayan eleman sayısı teriminin ağırlığı

**karar değişkenleri:**

$$x_{p hr} = \begin{cases} 1 & \text{p parçası r rotası ile h hücresine atandı} \\ 0 & \text{diğer} \end{cases}$$

$$y_{th} = \begin{cases} 1 & \text{t tezgahı h hücresine atandı} \\ 0 & \text{diğer} \end{cases}$$

$$\sum_r \sum_h x_{p hr} = 1 \quad \forall p \quad (3.1)$$

$$\sum_h y_{th} \leq N_t \quad \forall t \quad (3.2)$$

$$x_{p hr}, y_{th} \in \{0,1\} \quad \forall p, t, h, r \quad (3.3)$$

$$\text{enkz} = w_1 \sum_p \sum_t \sum_h \sum_r a_{ptr} x_{p hr} (1 - y_{th}) + w_2 \sum_p \sum_t \sum_h \sum_r (1 - a_{ptr}) x_{p hr} y_{th} \quad (3.4)$$

Denklem (3.1) kısıtı, her parçanın sadece bir alternatif rotayı takip ederek bir hücreye atanmasını garanti etmektedir. İkinci kısıt olan (3.2) ise her tezgahın kopya sayısı kadar hücrelere atanabileceğini ifade etmektedir. Üçüncü kısıt (3.3), karar değişkenlerinin tümünün 0-1 tam sayılı değişken olduğunu gösterir. Modelin amaç fonksiyonu (3.4)'deki eşitlikte verilmektedir. Denklemdeki ilk terim hücre dışı elemanları gösterirken, ikinci terim de hücre içi kullanılmayan elemanları göstermektedir. Her iki terimde belirlenen ağırlıklarla çarpılmaktadır. Amaç, bu ağırlıklandırılmış toplamı enküçükmektir. Dolayısıyla hücreler arası taşıma mesafesi en küçüklenecektir.

### 3.4.3. Kümeleme analizini temel alan yöntemler

Kümeleme analizi birçok karmaşık veri setini tanımlamak için kullanılan bir yöntemdir. Bu istatistiksel aracın amacı, kümelerin içinde yer alan nesnelerin ilişki seviyesini maksimum yaparken, kümeler arasındaki ilişki derecesini de en aza indirmektir. Kümeleme yöntemleri sıralama temelli kümeleme yöntemleri, hiyerarşik kümeleme yöntemleri ve hiyerarşik olmayan kümeleme yöntemleri olmak üzere 3'e ayrılmaktadır [4]:

**Sıralama temelli kümeleme yöntemleri:** bu yöntemde parçaların işlem görecekları makineleri gösteren temas matrisi temel alınır. Matris üzerinde yer alan  $a_{ij}$  elemanı 1'e eşit olduğunda i. parça j makinesinde işlem görecektir, eğer 0 değerini aldıysa da i. parça j. makineden işlem görmemektedir. Yöntemin amacı temas matrisinde yer alan 1'leri satır ve sütun işlemleri yaparak diyagonal bir matris formuna dönüştürmektir. Literatürde yer alan sıralama temelli yöntemler;

Bond Enerji Analizi, Derece Sırası Kümeleme (ROC), Düzenlenmiş Derece Sırası Kümeleme (MODROC), Direkt Kümeleme Analizi (DCA) ve Küme Belirleme Yöntemi (CIA)'dir.

**Hiyerarşik kümeleme teknikleri:** bu yöntemde temas matrisindeki veriler tek bir adımda hücrelere ayrılmamaktadır. Öncelikle geniş hücreler oluşturulur, sonra bu hücreler bölünmeye başlar ve bu işlem daha alt bir hücre oluşturulamayınca kadar devam eder. Genellikle hiyerarşik teknikler birleştirici ve bölücü teknikler olmak üzere 2'ye ayrılır. Birleştirici teknik makine ve parçaları bir hücrede birleştirirken, bölücü teknik makine ve parçaları hücrelere böler. Literatürde yer alan yaklaşımlar Tekli Bağlantı Kümelemesi (SLC), Tam Bağlantı Kümelemesi (CLC), Ortalama Bağlantı Kümelemesi (ALC) ve Lineer Hücre Kümelemesi (LCC) 'dir.

**Hiyerarşik olmayan kümeleme teknikleri:** veri setinin kümeler ayrılması veya birkaç tane çekirdek nokta seçimi ile başlayan bir tekniktir, bu teknikte küme sayısı önceden bilinmelidir. Tekrarlayan bir yapıya sahip olan bu tekniği Chandrasekharan ve Rajagopalan, Lemoine ve Mutel, Srinivasan ve Narendran geliştirmiştir.

#### 3.4.4. Grafik bölümlenme yöntemi

Bu yaklaşımda parçalar ve makineler düğümler gibi düşünülürken, parça işlemleri de bu düğümlere yapılan bağlantılar olarak düşünülmektedir. Bu modelin amacı makine-makine alt grafikleri veya makine-parça grafiklerini birbirinden bağımsız hale getirmektir [1]. Kümeleme yönteminde olduğu gibi bu yöntemde de taşıma mesafesini azaltmak amaçlanmaktadır. Grafik bölümlenme yöntemine bir çok veri dahil edilememektedir [23].

#### 3.4.5. Yapay zeka yöntemleri

NP-Zor sınıfta yer alan hücre oluşturma problemlerinin çözümü matematiksel programlama ile yapılabiliyor olsa dahi, büyük bir örnekle karşılaşıldığında bunu makul bir zamanda çözmek kolay olmamaktadır. Ayrıca problem karmaşıklıklaştıkça modelin kurulumu da zorlaşmaktadır. Diğer yandan

problemin çözümü için sezgisel yöntemler makul zaman aralığında iyi çözümler verebilse de probleme sınırlı ölçüde kısıt ve amaçlar dahil edilebilmektedir. Bunlara alternatif olarak geliştirilen yapay zeka yöntemleri uygun bir zaman aralığında eniyi değeri bulan ya da bu değere çok yaklaşan yöntemlerdir.

Bu kısımda NP-Zor problemlerin çözümünde kullanılan Genetik Algoritmalar, Tavlama Benzetimi, Yasaklı Arama, Yapay Sinir Ağları, Parçacık Sürü Optimizasyon yöntemlerinden bahsedilecektir.

Sezgisel yöntemler, eldeki başlangıç çözümü ile daha iyi çözümlere ulaşabilmektedir. Daha açık bir ifadeyle sıralama ve gruplama gibi problemlerde nesnelerin değiştirilmesi ile farklı çözümler elde edilebilmektedir. Bilinen sezgisel yöntemler genelde başlangıç çözümünde uzaklaşmamakta, böylece arama uzayının kısıtlı bir bölgesinde arama yapmaktadır. Buna karşın Yapay Zeka yöntemleri yerel eniyi çözümlerden kurtulacak mekanizmalara sahiptir. Bu nedenle Yapay Zeka yöntemleri etkili ve hızlı çözümler üretmektedir [37].

**Genetik algoritmalar:** Genetik algoritmalar evrim sürecini dikkate alarak, eniyi değeri bulmaya çalışan bir yöntemdir. Holland tarafından geliştirilen algoritma farklı eniyileme problemlerine uygulanabildiğinden, sıklıkla kullanılan yapay zeka yöntemleri arasında gösterilmektedir.

Kromozom adı verilen çözüm popülasyonu, seçim, çaprazlama ve mutasyon operatörlerine göre gelişim göstererek iyiye yakınsamaktadır. Seçim işlemi var olan çözümler üzerinden seçim yapılarak gelecek nesillere aktarım işlemidir. Seçim eniyi çözümler arasından yapılabilmektedir, ama kötü çözümler arasından da yapılarak tek bir noktaya olan yakınsama işleminin önüne geçilmiş olur. Seçim işlemi matematiksel modelin amaç fonksiyonu göz önüne alınarak yapılır ve Rulet Çemberi Seçim Yöntemi, Stokastik Artan Seçim Yöntemi, Turnuva Seçim Yöntemi bunlardan bazılarıdır. Çaprazlama operatörü yeni nesillerin türetilmesi sırasında var olan bireylerin parçalarının yer değiştirme işlemidir. Bu Operatörlerden bazıları: Tek Noktalı Çaprazlama operatörü, İki Noktalı Çaprazlama operatörü, Çok Noktalı Çaprazlama operatörüdür. Mutasyon operatörü ise bireyde seçilen bir genin yerine farklı bir değer alınması işlemidir [25, 38].

---

**başla;**

**tekrar et**

*bireyleri değerlendir;*

**tekrar et**

*ebeveynleri seç;*

*yeni bireyleri türet;*

*eğer yeterince çözüm üretildiyse mutasyon uygula;*

popülasyon sayısına ulaşına **kadar**;

*eniye bireyleri kopyala;*

gerekli nesil sayısı üretilene **kadar**

---

### Şekil 3.6 GA sözde kodlar

Genetik Algoritmaları (GA) diğer yöntemlerden ayıran özellikler aşağıdaki şekildedir [25]:

- GA, parametrelerde değil onların kodları ile çalışır.

GA, tek bir noktada arama yapmak yerine bir topluluk üzerinde arama yapmaktadır.

- GA, yardımcı bilgileri değil amaç fonksiyonu (uyum fonksiyonu) değerini kullanmaktadır.
- GA aramayı deterministik kurallar yerine, olasılıklı kurallarla yapmaktadır.

Genetik algoritmalara ait sözde kodlar Şekil 3.6’da verilmektedir [8].

Bir GA adımları şu şekilde olabilir [37]:

0. Gerekli verileri topla ve parametreleri belirle.
1. İlk popülasyonu rassal olarak oluştur.
2. Bireylerin uyum fonksiyonlarının hesapla.
3. Bir seçim ilkesine göre bireyleri seç.
4. Bir olasılığa bağlı olarak bir çift birey seç, kopyalama işlemi.
5. Bunları rassal belirlenen noktalara göre çaprazla
6. Belirlenen oranda birey kopyalanmadıysa 4’e dön.
7. Mutasyon sırası gelmediyse 9’a atla.
8. Rassal bir noktadan mutasyon uygula.
9. Yeterli sayıda yeni birey yetiştiyse DUR, aksi durumda 2’ye dön.

Genetik algoritmalar doğrusal ve doğrusal olmayan eniyileme problemlerinde, arama uzayının geniş bir alanını taradığından, sıklıkla kullanılan

bir yöntemdir. Uyum değeri iyi bireyler, gelecek nesillere aktarıldığı için popülasyon iyi noktalara doğru yönlendirilmektedir. Ayrıca kötü bireylerdeki iyi genlerde var olan bilgi kullanılarak arama uzayının farklı yönlerinin de taranması sağlanmaktadır.

Hücrel üretim problemlerinin çözümü için GA kullanan çalışmalar vardır [1, 6, 25, 38].

**Tavlama Benzetimi:** Tavlama Benzetimi, eniyileme problemleri için rassal bir arama yöntemidir. Malzemelerdeki ısı işlem sürecini temel alır. Metalin soğuması ve donması sırasında en az enerji ile kristalleşme meydana gelerek, malzeme yapısındaki bozulmalar en aza indirilmiş olur. Tavlama işleminde kontrol edilmesi gereken parametreler sıcaklık değişimi ve soğutma prensibidir. Tavlama benzetimini eniyileme problemlerine ilk uygulayan 1983 yılında Kirkpatrick, Gelatt ve Vecchi olmuştur. Bazı yaklaşımların dezavantajı arama sırasında yerel eniyeye takılmalarıdır. Tavlama Benzetimi ise yerel eniyilerden kurtulabilecek mekanizmalara sahiptir. Yavaş bir soğutma işlemi ve yeterli rassallık sağlandığında, Tavlama Benzetimi ile aramalar bütünsel eniyi noktaya yaklaşmaktadır [39].

Rassal arama ile eniyi değere yaklaşmayı sağlayan TB, NP-Zor problemlerin çözümünde sıklıkla kullanılmaktadır. Yöntemde başlangıç çözüm ( $S_0$ ) alınarak amaç her tekrarda iyileşme sağlamaktadır. Kontrol parametresi olan sıcaklık da başlangıç parametresi ( $T_0$ ) olarak belirlenir. Böylece çözüm uzayında diğer bölgelerinde taranması sağlanmış olur.

Belirlenen başlangıç sıcaklığı ( $T_0$ ), bir soğutma fonksiyonuna bağlı olarak azalmaktadır, bu işlem belirlenen bir durdurma kriteri sağlanıncaya kadar devam eder. İşlem sırasında var olan çözümün ( $S$ ) komşularına geçiş yapılarak yeni bir çözüm ( $S_n$ ) türetilir. Eğer yeni amaç fonksiyonu ( $f(S_n)$ ) değeri bir önceki değerden daha iyiyse yeni çözüm ( $S_n$ ) kabul edilir, aksi durumda yerel eniyiden kaçmak için yeni çözüm  $P = e^{-\left(\frac{\Delta}{T}\right)}$ ,  $\Delta = \frac{f(S_n) - f(S)}{f(S_n)} * 100$  olasılığı ile kabul edilir. Algoritma adımları istenilen durdurma koşulu sağlanıncaya kadar devam etmektedir [40]. Tavlama Benzetimi için sözde kodlar Şekil 3.7’de verilmektedir [8]. Öztürk [11] alternatif rotalı hücrel üretim problemlerini çözen bir TB algoritması önermiştir.



---

**başla;**

**tekrar et**

*bir asay çözüm türet;*

*adayı değerlendir;*

*mevcut çözüme karar ver;*

*sıcaklığı düşür;*

**sonlanma kriteri sağlanıncaya kadar devam et**

---

Şekil 3.7 Tavlama benzetimi sözde kodlar

**Yasaklı Arama:** Yasaklı Arama 1970’de Glover tarafından geliştirilen bir yöntemdir. Hafıza ve arama geçmişini yöntemin bileşenleri arasında yer alır. Arama temeline dayanan birçok başarılı algoritma, hafıza yöntemini kullanmamaktadır, bunun nedeni, öncelikle hafızanın arama etkinliği üzerindeki başarısının nasıl olacağını bilmemesidir. Bu yüzden hafıza kullanılmış olsa dahi çözümde gelişim görülemeyebilir; çünkü hafızanın ortaya koyduğu serbestlik derecesinin çok fazla olmasıdır, özellikle uyarlamalı hafıza kullanımında kararlı ve etkili çözümler bulan algoritmaların geliştirilmesi neredeyse imkansızdır.

Yasaklı arama bazı problemlerin çözümlerinde etkili sonuçlar bulsa da, buna neden olan matematiksel yaklaşım açıklanamamaktadır. Üzerinde çalışılan en önemli iki konu; daha etkili bir hafıza yönetimi nasıl yapılacağı, bir diğeri de diğer algoritmalar ile birleştirilerek daha etkili çalışan hibrid yöntemler nasıl geliştirileceği şeklindedir.

Yasaklı arama, yoğun olarak yerel arama yapan bir algoritmadır ve döngüden kurtulmak için de hafıza yöntemini kullanır. Eldeki sonuçlar ve önceki aramalardan elde edilen sonuçlar yasak listesinde tutulur, yapılacak olan yeni arama bu listedekiler dışından olmalıdır. Yasaklı Arama’da yer alan liste, yöntemin en önemli bileşenidir, geçmiş ve şimdiki arama hareketleri bu listede tutulduğundan yeni arama hareketi listeden olmamalıdır, böylece algoritma aynı işlemleri yapmaktan kaçınır. Algoritma çok sayıda tekrar yaparsa, liste sayesinde arama yapmak için geçen hesaplama süresinden önemli ölçüde kar edilecektir [39].

---

**başla;**

**tekrar et**

*kabul edilebilir bütün komşu çözümleri türet;*

*türetilen çözümleri değerlendir;*

*eniyesini aday çözüm olarak belirle*

*eğer aday çözüm bulunamazsa yasak çözümlerden eniyesini aday olarak belirle;*

*yasak listesini güncelle; aday çözüme hareket et;*

*eğer yeterli sayıda çözüm türetiliyse, değiştir;*

**sonlanma kriteri sağlanıncaya kadar devam et;**

---

**Şekil 3.8** Yasaklı arama sözde kodlar

Yasak listesinde daha önce yapılan arama geçmişine ait çözümlerin nitelikleri tutulmaktadır. Yeni bir aramadan sonraki çözüm, eniyi gelişim stratejisine göre kabul edilmektedir, daha açık olarak klasik yerel arama yöntemlerinde olduğu gibi biri diğerinden daha iyiyse kabul et şeklinde değildir. Yasak listesinde bilgileri depolamanın 2 yolu vardır [1]:

- Bütün çözümleri depola: bu yöntemin avantajı önceki çözümlerin tekrar ziyaret edilmemesidir. Dezavantajı ise yasaklı listesinin büyümesi dolayısıyla yeni çözümün listede var olup olmadığını kontrol etmek için geçen sürenin büyümesidir. Algoritma sonlandırma süresinde de kaçınılmaz olarak bir artışa neden olacaktır.
- Çözümlerin niteliklerini depola: genel olarak kullanılan teknik çözümlerin niteliklerini yasaklı listeye almaktır. Böylece komşu çözümlerin hepsi niteliklerine göre yasaklanmış olur. Öz olarak bir noktadan komşu bir noktaya geçiş niteliği listeye alınarak aynı hareketi tekrar yapmanın önüne geçilmiş olur. Önceki tekniğin aksine hafıza kullanımı ve hesaplama süresin çok daha azdır. Ancak bu nitelikler aynı anda birçok hareketi yasaklayabilir, böylece iyileşme sağlanacak noktaların da önüne geçilmiş olur. Bu nedenle listeye alınacak çözüm niteliklerini iyi belirlemek gerekir.

Yasaklı Arama yöntemine ait sözde kodlar Şekil 3.8’de verildiği gibidir [8]. Çeşitli hücreli imalat problemlerinin çözümü için Chung ve ark. [34], Wu ve ark. [41] Yasaklı Arama yöntemini kullanmıştır.

**Karınca Kolonileri:** Adından da anlaşılacağı üzere bu yöntem birlikte yaşayan karınca kolonilerinden esinlenilerek 1992 yılında Dorigo tarafından

ortaya atılmıştır. Karıncalar yiyecek arama sırasında bir-birleri ile etkileşim içindedir. İletişim, her karıncanın bıraktığı bir koku veya feromon adı verilen kimyasal maddeye göre yapılmaktadır. Kendi bölgesinde arama yapan karınca etrafa feromon yaydığına, ondan sonra gelen karıncalar bu feromonu dikkate alarak bir öncekinin izlediği rotayı izler. Yiyecek arama sıklıkla etraftaki feromon miktarı da yoğunlaşır ve karıncalar yoğun olan feromon miktarına göre aramalarını sürdürür. Karınca sayısında artış olduğunda aynı yoldaki feromon miktarı artacağından, bu yol en etkili ve kısa yol olacaktır. Bu yöntem geriye bilgi akışı sayesinde avantajlı bir strateji haline gelmektedir. Karınca özelliklerine bazı eklemeler yapılarak yöntem çok daha etkili hale getirilebilir. Bu yöntemdeki en önemli konu, rotanın seçilme olasılığı ve feromon miktarındaki buharlaşma oranıdır [39]. Karınca kolonilerine ait sözde kodlar Şekil 3.9'da verilmektedir. Solimanpur ve ark. [21], İşlier [42], hücresel üretim problemlerinin çözümü için Karınca Kolonilerini kullanmıştır.

**Yapay Sinir Ağları:** Yapay sinir ağları, biyolojik sinir ağlarından esinlenilerek geliştirilmiş bir bilgi işleme sistemidir ve sinir hücrelerinin birbirleri ile bağlanma şekillerini temel alan bir yapıya sahiptir. Katmanlar şeklinde olan bu yapay sinir hücrelerinin en belirgin özellikleri, birbirine bağlı nöronlar, bağlantılar arasındaki ağırlıklar ve aktivasyon fonksiyonudur.

Yöntemin geçmişi 1942 yılına kadar gitmektedir. Yapay sinir ağlarını oluşturan her nöronun bir iç hali vardır. Bu iç hale aktivasyon veya aktivasyon seviyesi denilir. Bu seviye, gelen girdileri tanımlayan bir fonksiyondur. Ağ içerisinde bir nöron aynı doğal nöronların yaptığı gibi diğer nöronlara bir işaret (sinyal gönderir). Nöronlar bir seferde tek bir işaret gönderebilirler. Gönderilen sinyaller gönderilen nöronlar için giriş fonksiyonlarını oluşturur. Nöronlar aynı anda birden fazla nörona sinyal gönderebilmektedir.

Sinir ağları 2 veya 3 katmandan oluşur, bu katmanlar girdi, gizli ve çıktı katmanı olarak adlandırılır. İki katmandan oluşan sinir ağlarında gizli katman yer almamaktadır. Şekil 3.10'da yapay sinir ağının yapısı görülmektedir. Y nöronuna bakıldığında  $X_1$ ,  $X_2$ ,  $X_3$  nöronlarından sinyal alır ( $x_1$ ,  $x_2$ ,  $x_3$  işaretler).  $X_1$ ,  $X_2$ ,  $X_3$  nöronlarını, Y nöronuna bağlayan ağırlıklar ise  $w_1$ ,  $w_2$ ,  $w_3$  olur. Gizli nöron olan

---

**başla;**

$i \leftarrow 0;$

**tekrar et**

*bir uygun çözüm türet;*

*$\eta$ 'yi değerlendir;*

*eğer  $\eta > \eta_{max}$  ise listeyi güncellemete başla; sınırların sonunu değiştir;*

*if  $i \bmod \alpha = 0$  ise çözümü değiştir;*

*if  $i \bmod \beta = 0$  ise feromon izlerini arttır;*

*feromon izlerini güncelle;*

$i \leftarrow i + 1;$

$i = \sigma$  oluncaya **kadar devam et;**

---

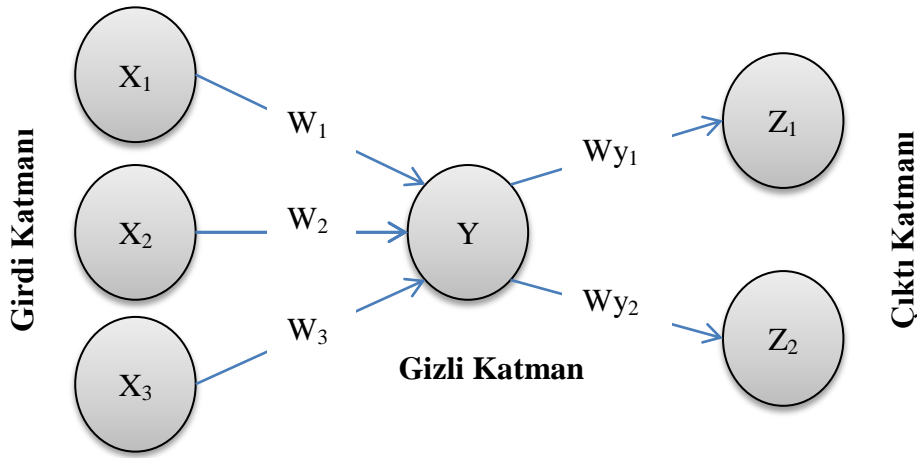
**Şekil 3.9** Karınca kolonileri sözde kodlar

Y nöronunun girdisi ( $y_{girdi}$ ), gelen işaretlerin ağırlıklarla çarpımının toplamıdır ve matematiksel ifadesi (3.5)'de verilmektedir.

$$y_{girdi} = w_1 * x_1 + w_2 * x_2 + w_3 * x_3 \quad (3.5)$$

Y nöronunun etkin hale gelebilmesi için bir fonksiyon yardımıyla eşik değere ulaşması gerekmektedir. Bu fonksiyonlara aktivasyon fonksiyonları denir ve en çok kullanılanı, görüntüsü "S" şeklinde olan lojistik sigmoid fonksiyondur [43].

Ağırlıkların büyüklüğü nöron çıktısının diğer nöronu hangi derecede etkilediğini belirler. İki nöron arasında ağırlığın büyük olması, onların güçlü ve etkili bir iletişim içinde olduğunu gösterir. Ağırlıkların ve işaretlerin çarpılarak toplanmasıyla elde edilen girdi değeri çıktı değeriyle aynı değildir. Nöronun akti-



**Şekil 3.10** Yapay sinir ağı

ve edilmesi için bir eşik değer söz konusudur. Girdi değerini çıktı değerine dönüştürürken doğrusal olmayan çıktı fonksiyonları kullanılır. Dönüşüm yapıldıktan sonraki yeni ağırlık değeri diğer nöron ile iletişimi sağlamaktadır. Bir nöronun temel işlemlerini şu şekilde özetleyebiliriz [44]:

- Diğer nöronlardan gelen ağırlık ve işaretlere göre girdi değeri hesaplanır
- Nöron aktivasyonuna göre çıktı değeri hesaplanır.
- Çıktı değeri tüm komşu nöronlar boyunca devam eder, bu işlemler tüm yapıyı kapsar.

Sinir ağlarının yapısı adından da anlaşılacağı gibi düğümlerin birbirleriyle etkileşim içinde olduğu ağdır. Her bir düğümün bir işlem birimi vardır ve diğerleri

ile bu işlem birimi bağlantı kurmaktadır. Düğümlerin çıktıları değişebilir parametrelerdir. Yapay sinir ağları kavramsal olarak birim işlemi gerçekleştirebilmek için geniş bir alana yayılmış paralel işlemlerdir. Birimler arasındaki iletişim gücüne bağlı olarak, bilgi çıkarma kabiliyetine sahiptir. Yapay sinir ağlarının avantajı paralel işlem yapabilme sayesindeki hızlı hesaplama yapabilmesi ve öğrenme kabiliyetinin olmasıdır. Yapay sinir ağlarının özellikleri [45]:

- Doğrusal olmayan yapı: bir nöron kendi başına doğrusal veya doğrusal olmayan yapıda olsa da tüm ağ kesinlikle doğrusal olmayan bir yapıdadır. Bu özellik oldukça önemlidir çünkü gerçek bir mekanizmadan öğrenme için veri türetme işi doğrusal olmayan bir model olacaktır.
- Örneklerden öğrenebilme: bağlantı ağırlıkları eğitim kümelerinden çıkarılabilir. Böylece eldeki probleme uyarlanarak tüm bir yapı boyunca kullanılacak parametrelere dönüştürülebilir.
- Uyarılma: ağ durağan olmayan bir yapıya sahiptir, yani bağlantılar arasında ki ağırlıklarda bir düzenleme olduğunda tüm çevredeki ağırlıklar yeni uyarlamaya göre düzenlenir.
- Kanıtsal tepki: örneğin veri sınıflandırmada, yapay sinir ağı verilen örnek için sadece bilgi vermemekte aynı zamanda karar verebilmek için bir teknik de sunmaktadır. Bu bilgi sayesinde belirsiz verilere değerler atanabilir.

- Hata payı: nöronlar arasında kopma olması ya da anlamsız verilerin olması durumunda bile yapay sinir ağlarının performansı azalmayarak güçlü bir yapı sergiler.
- Analiz ve tasarımda değişmezlik: yapay sinir ağları değişmez bilgi işlemcileridir. Uygulamalarda aynı prensip, notasyon ve yöntem adımları kullanılır.

Öztürk ve ark. [8] alternatif rotası olmayan hücre oluşturma problemlerini, hücreler arası taşıma mesafesini en aza indirmeyi amaçlayan modeli rekabetçi sinir ağları ile çözmüşlerdir.

**Parçacık sürü optimizasyonu:** Parçacık sürü optimizasyonu (PSO)1995 yılında, Eberhat ve Kennedy tarafından geliştirilen, balık ve kuş sürülerinin doğadaki davranışlarını temel alan bir yöntemdir. PSO'nun genetik algoritmalar ve karınca kolonileri ile birçok ortak yönü olmasına rağmen mutasyon, çaprazlama, feromon gibi operatörleri içinde barındırmadığından uygulaması çok daha kolay olmaktadır. Reel sayıları kullanabilme yeteneği ve sürüdeki parçacıkların bütünsel eniyi ile etkileşim içinde olmalarından dolayı herhangi bir kodlama yöntemine başvurulmamaktadır. Bu algoritma arama uzayındaki hareketini amaç fonksiyonuna ve parçacık adı verilen sürü içindeki bireylerin yörüngelerine bağlı olarak yapar. Algoritmada parçacıkların hareketleri yarı stokastik bir davranış sergileyen vektörler doğrultusunda olmaktadır. Parçacık hareketi iki önemli bileşene sahiptir: stokastik ve deterministik bileşen. Parçacık hareketini o andaki bütünsel eniyi konumuna doğru rassal olarak yapar. Eğer parçacık daha önce bulduğu değerlerdeniyi bir değer bulursa, konumunu o andaki yere göre günceller. Bütünsel eniyi değere ulaşmak için parçacıkların kendi eniyi konumu ve tüm parçacıklar içindeki eniyi konum birlikte kullanılır. Algoritmada tanımlanan sonlandırma adımına kadar tekrar eder [39]. PSO algoritması ayrıntılı bir şekilde sonraki bölümde anlatılmaktadır.

Andres ve Lozano [47], hücreler arası taşıma mesafesini en aza indirmeyi amaçlayan bir çalışma yapmıştır. Çalışmanın özgün yanı literatürde hücre oluşturma problemlerinin çözümü için ilk defa PSO yaklaşımının kullanılmasıdır.

Rezazadeh ve ark. [48], sanal hücre üretim sistemleri için PSO yaklaşımını ve doğrusal programlamayı birlikte kullanmışlardır. Çalışmanın avantajı işlem

adımlarını, alternatif rotaları, makine kapasitelerini, işçi çalışma kapasitelerini, maksimum hücre büyüklüğünü, makine işçi dengesi kısıtlarını modele dahil etmesidir.

Anvari ve ark. [49], hücre dışı ve hücre içi kullanılmayan elemanları en aza indirmek için yeni bir PSO yaklaşımı önermiştir. Modelde hücrede bulunabilecek maksimum makine sayısı ve maksimum hücre sayısı kısıtları yer almaktadır. Bu alanda yapılan ikinci PSO yaklaşımıdır.

Duran ve ark. [50], hücreler arası taşıma mesafesini en aza indirmek için yeni bir PSO yaklaşımı önermiştir. Bu çalışmada klasik PSO yaklaşımında var olan hız güncelleme adımı kullanılmamaktadır. Veri madenciliği uygulamalarında kullanılan orantılı benzerlik tekniği yaklaşıma dahil edilmiştir.

#### 3.4.6. Çözüm başarısının değerlendirilmesi

Literatürde hücre oluşturma problemlerinin değerlendirilmesine ilişkin farklı ölçüm teknikleri ileri sürülmüştür. Temas matrisinde yer alan 1'lerin köşegen bloklar üzerinde olması istenir. Blok köşegen matris oluşturulduğunda, her bir blok hücreleri göstermektedir. Bu grupların dışında kalan 1'ler istisnai elemanlar olurken içeride kalan 0'larda boşlukları göstermektedir. Hücre oluşturma problemlerinin çözümünü değerlendiren performans ölçüleri, genellikle bahsedilen bu boş ve harici elemanları kullanmaktadır. Literatürde sıklıkla kullanılan iki ölçüm tekniği vardır: Chandrasekharan ve Rajagopalan [51] tarafından gruplama verimliliği, Kumar ve Chandrasekharan [52] gruplama etkinliği önerilmiştir. Bu değerlendirme yöntemlerine ilişkin gösterim aşağıdaki gibidir:

$n_1^i$  diyagonal blok içindeki 1'lerin toplam sayısı.

$n_1^d$ =diyagonal blok dışındaki harici elemanların toplam sayısı

$n_0^i$ =diyagonal blok içindeki boşlukların toplam sayısı

$n_0^d$ =diyagonal blok dışındaki 0'ların toplam sayısı

$n_1$ =parça-makine matrisindeki 1'lerin toplam sayısı

$n_0$  =parça-makine matrisindeki 0'ların toplam sayısı

**Gruplama verimliliği:** Gruplama verimliliğini ifade eden  $\eta$  (3.6)'da verilmiştir:

$$\eta = q\eta_1 + (1 - q)\eta_2 \quad (3.6)$$

$\eta_1$  diyagonal matris içindeki yer alan 1'lerin toplam sayısının, diyagonal matris içindeki 1 ve 0'ların toplam sayısına oranıdır.

$$\eta_1 = \frac{n_1^i}{n_1^i + n_0^i} \quad (3.7)$$

$\eta_2$  diyagonal matris dışında yer alan 1'lerin toplam sayısının, diyagonal matris dışındaki 1 ve 0'ların toplam sayısına oranıdır.

$$\eta_2 = \frac{n_1^d}{n_1^d + n_0^d} \quad (3.8)$$

$\eta_1$  ve  $\eta_2$  değerlerine verilen ağırlıkları ifade eden  $q$ ,  $0 \leq q \leq 1$  aralığında yer alır.

**Gruplama Etkinliği:** Gruplama verimliliği temas matrisi büyük olan problemlerde iyi ve kötü yapıyı ayırt edemediği için Kumar ve Chandrasekharan [52], blok dışındaki harici elemanlara ve blok içindeki boşluklara eşit ağırlıklar atayan Gruplama Etkinliğini önermiştir. Hücre dışında yer alan harici elemanların az olması hücreler arası taşıma mesafesini azaltırken, hücre içindeki boşlukların az olması da makine kullanım oranının artmasına neden olmaktadır. Gruplama Etkinliği  $\Gamma$  ile tanımlanır:

$$\Gamma = \frac{n_1 - n_1^d}{n_1 + n_0^i} \quad (3.9)$$

Köşegen bloklarda yer alan 1'lerin toplam sayısının, temas matrisinde yer alan 1 ve köşegen blokta yer alan boşlukların toplamına oranı Gruplama Etkinliğini verir. Çalışmada performans ölçütü olarak Gruplama Etkinliği temel alınmıştır.



## 4. ALTERNATİF ROTALI HÜCRE OLUŞTURMA PROBLEMLERİNİN ÇÖZÜMÜ İÇİN YENİ BİR MELEZ YAKLAŞIM

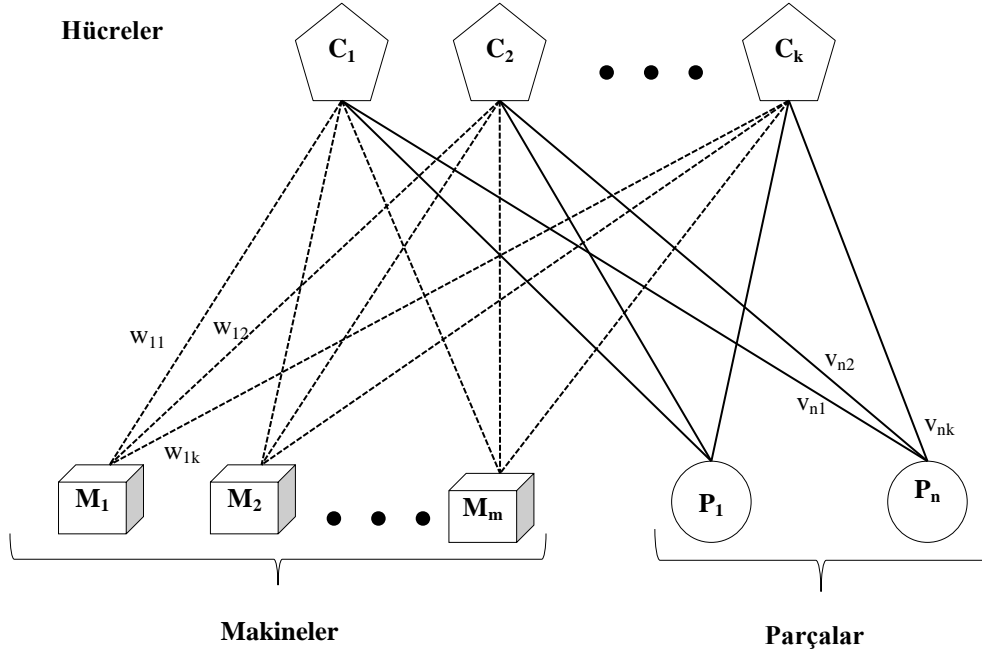
Bölümde sırasıyla Rekabetçi Sinir Ağları (RSA) ve PSO yöntemleri açıklanacaktır. RSA, Öztürk ve Öztürk [53] tarafından hücre oluşturma problemlerine uygulanmış ve başarılı sonuçlar elde edilmiştir. RSA aynı anda makine gruplarının parça ailelerine ve parça ailelerinin makine gruplarına atanmasını sağlayacak şekilde tasarlanmıştır. Geliştirilen bu yöntem parçaların tek bir rotası olduğu durumları ele almaktadır. Buna karşılık birçok üretim şeklinde parçaların izleyebileceği alternatif rotalar mevcuttur. Alternatif rotalar göz önüne alındığında problemin karmaşıklığı artmakla birlikte, oluşturulacak hücre yapılarının gruplama etkinlik değeri çok daha iyi olmaktadır. Alternatif rotalı hücre oluşturma problemlerini çözmek için RSA ve PSO'nun birlikte kullanıldığı melez bir yaklaşım önerilmiştir. Geliştirilen melez yaklaşımda PSO parçaların alternatif rotalarını seçerken, belirlenen rotalara göre parça ve makinelerin hücrelere atanması işlemi RSA tarafından yapılmaktadır.

### 4.1. Rekabetçi Sinir Ağı

Rekabetçi sinir ağının yapısı 3 farklı bileşenden oluşur. İlk iki bileşen, girdi desenini oluşturan parça ve makinelerden oluşurken, üçüncü bileşen de çıktı desenini oluşturan hücrelerdir. Girdi ve çıktı nöronları arasındaki bağlantıyı  $w_{ij}$  ve  $v_{pj}$  ile gösterilen ağırlıklar belirler. Ağırlık indisleri  $i=1,\dots,m$ ;  $p=1,\dots,n$  ve  $j=1,\dots,k$  şeklindedir ve sırasıyla makine, parça ve hücreleri simgelemektedir. Şekil 4.1'de bahsedilen ağ yapısı görülmektedir.

RSA algoritması, Chu tarafından 1989'da geliştirilen algoritmayı temel alır. Aralarındaki fark girdi deseninin belirlenmesindedir. Bu fark sayesinde RSA, parça ve makineleri eş zamanlı olarak hücrelerde gruplayabilmektedir.

RSA algoritmasında *öğrenme oranı* ve *tekrar sayısı* gibi parametreler başlangıçta verilir. Nöronlar arası ağırlıklar ilk başta rassal olarak oluşturulduktan sonra ağdaki bütün ağırlıklar normalleştirilir. Temas matrisinde yer alan her parça



Şekil 4.1 Rekabetçi sinir ağının yapısı [8]

ve makinenin ağ yapısı içinde kendine özgü bir deseni vardır. Sonuçta,  $m+n$  uzunluğunda ve  $m+n$  tane ayrı desen oluşmaktadır. Algoritma genel olarak iç ve dış olmak üzere iki döngüden oluşur. Dış döngü her tekrarda,  $m+n$  uzunluğunda rassal bir  $\pi$  dizilimi (permütasyonu) üretirken, iç döngü de her seferinde farklı desenler oluşturmak için bu dizilimi kullanır. Bu oluşturma işlemi  $\pi[q]$  ile uyumlu olarak yapılır.

$\pi$  dizilimindeki tüm elemanlar için sırayla girdi vektörü oluşturulur. Bunun için,  $m+n$  uzunluğunda bir vektör tanımlanır. Eğer  $\pi[q] \leq m$  ise desenin  $\pi[q]$ 'ci elemanı 1 değerini alır ve temas matrisindeki  $\pi[q]$ 'ci satır, desenin son "n" elemanına kopyalanır. Aksi bir durumda ise  $(\pi[q]-m)$ 'ci sütunun evriği girdi deseninin ilk "m" elemanına kopyalanır ve  $\pi[q]$ 'ci eleman 1 değerini alır. Bunun dışında desende yer alan diğer elemanlar 0 değerini almaktadır. Şekil 4.2'deki temas matrisi ele alındığında, 2. makine ve 4. parça için girdi desenleri (vektörleri) sırasıyla  $[01000|0010100]$  ve  $[10101|0001000]$  şeklinde oluşmaktadır.

Sonraki aşamada, çıktı katmanındaki nöronların net çıktıları (4.1) kullanılarak hesaplanır. Net girdi değeri en yüksek olan nöron, kazanan nöron olarak belirlenir ve bu nöronun ağırlıkları, (4.2) ve (4.3) kullanılarak güncellenir.

	P1	P2	P3	P4	P5	P6	P7
M1		1		1			1
M2			1		1		
M3	1	1		1			1
M4	1		1			1	
M5			1	1	1	1	

Şekil 4.2 Örnek bir parça-makine (temas) matrisi

İç döngünün son adımında, girdi ve çıktı nöronları arasındaki bağlantı ağırlıkları normalleştirilir.

Algoritmanın adım adım ilerleyişi, ön değer atamaları, döngüler ve sonuçlar aşağıda belirtilmektedir.

İlk olarak, durdurma parametresi olan  $n_{epoch}$  (tekrar sayısı),  $l$  (öğrenme oranı) ve başlangıç ağırlıkları belirlenmektedir. Dış döngü tarafından oluşturulan dizilim ve buna karşılık gelen desenler ilk adımı oluşturmaktadır. Şekil 4.3'teki iç döngüde, dizilimin ilk elemanı olan 6. desen gösterilmektedir. Denklem (4.1) ve bu desen kullanılarak hesaplanan  $net_j$  değerleri şeklin sağ üst köşesinde verilmektedir.

$$net_j = \sum_{t=1}^m w_{tj}x_t + \sum_{t=m+1}^{m+n} v_{tj}x_t \quad , \forall j \quad (4.1)$$

Girdi deseni olan  $x_t$ 'nin değeri, ( $t=1, \dots, m+n$ ), nöron aktif ise  $x_t=1$  olurken diğer durumda 0 olur. En yüksek  $net_j$  değerine sahip nöron, "kazanan" nöron olur ve aktif duruma gelir. Kazanan çıktı nöronu aşağıdaki gibi belirlenir.

$$max_j \{net_j\} = net_c \longrightarrow max\{0,29;0,30\}=0,30 \text{ olduğundan kazanan 2. nörondur.}$$

Son ağırlıklar (4.2) ve (4.3) kullanılarak, ağırlıklandırılmış girdiler toplamı ise döngülerden sonra hesaplanır. Eğer kazanan nöron  $c$ ,  $r$ 'ye eşitse (4.2), değilse (4.3) kullanılmaktadır.

$$w'_{ir} = \frac{lx_t}{\sum_t x_t} + (1-l)w_{ir} \quad \text{ve} \quad v'_{pr} = \frac{lx_t}{\sum_t x_t} + (1-l)v_{pr} \quad (4.2)$$

$$w'_{ir} = w_{ir} \quad \text{ve} \quad v'_{pr} = v_{pr} \quad (4.3)$$

Eğer ağırlıklandırılmış girdiler toplamı bir makine için hesaplanırsa, son ağırlıklar, temas matrisinin karşı gelen satırlar ile çarpılır. Diğer durumda, son ağırlıklar her hücrenin sütunları ile çarpılır.

parametreleri belirle  $n_{epoch}=10$  (tekrar sayısı), öğrenme oranı  $l=0.05$   
başlangıç ağırlıkları

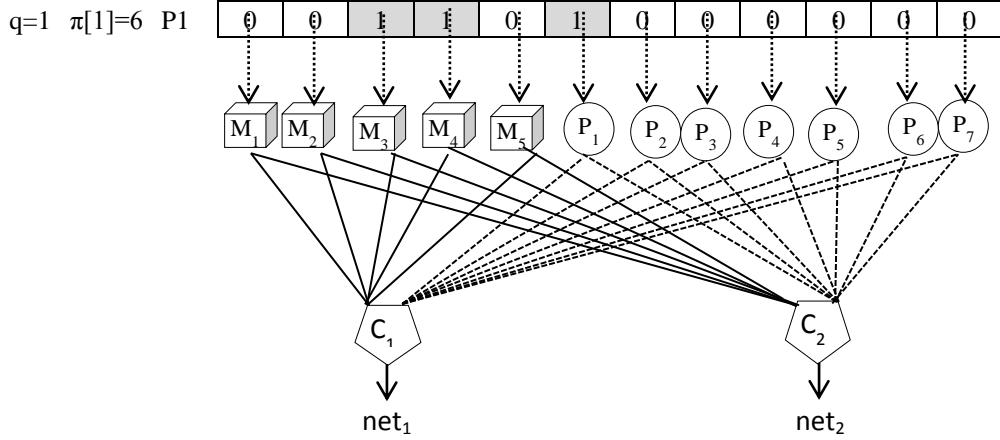
												net <sub>i</sub>	
1	0.06	0.12	0.12	0.10	0.07	0.07	0.06	0.10	0.01	0.09	0.11	0.06	0.29
2	0.03	0.08	0.12	0.06	0.10	0.12	0.12	0.09	0.06	0.12	0.01	0.09	0.30

$\pi_{n+m}$  permütasyon oluştur

q	$\pi[q]$													
1	6	P1	0	0	1	1	0	1	0	0	0	0	0	0
2	11	P6	0	0	0	1	1	0	0	0	0	1	0	
3	1	M1	1	0	0	0	0	0	1	0	1	0	0	
4	12	P7	1	0	1	0	0	0	0	0	0	0	1	
5	3	M3	0	0	1	0	0	1	1	0	1	0	0	
6	7	P2	1	0	1	0	0	0	1	0	0	0	0	
7	2	M2	0	1	0	0	0	0	0	1	0	1	0	
8	5	M5	0	0	0	0	1	0	0	1	1	1	0	
9	4	M4	0	0	0	1	0	1	0	0	0	1	0	
10	8	P3	0	1	0	1	1	0	0	0	0	0	0	
11	10	P5	0	1	0	0	1	0	0	1	0	1	0	
12	9	P4	1	0	1	0	1	0	0	0	1	0	0	
			x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12

DIŞ DÖNGÜ

sıradaki girdi desenini seç



İÇ DÖNGÜ

Son ağırlıklar

												q=n+m	S=n <sub>epoch</sub>
1	0.00	0.15	0.00	0.14	0.19	0.04	0.00	0.17	0.03	0.15	0.13	0.00	
2	0.20	0.00	0.24	0.05	0.04	0.09	0.13	0.00	0.12	0.00	0.00	0.13	

girdi ağırlıklarının toplamı

1	0.03	0.32	0.08	0.35	0.48	0.14	0.00	0.48	0.19	0.34	0.32	0.00
2	0.38	0.00	0.47	0.09	0.12	0.29	0.44	0.10	0.48	0.04	0.10	0.44

hücreler

	M1	M2	M3	M4	M5	P1	P2	P3	P4	P5	P6	P7
2	1	2	1	1	2	2	2	1	2	1	1	2

Şekil 4.3 Rekabetçi sinir ağının açıklayıcı adımları [8]

çözüm  
 #istisnai 2  
 #boşluk 3  
 #eleman 16  
 Etkinlik 0.74

	P3	P5	P6	P1	P2	P4	P7
M2	1	1					
M4	1		1	1			
M5	1	1	1			1	
M1					1	1	1
M3				1	1	1	1

Şekil 4.3 (Devam) Rekabetçi sinir ağıının açıklayıcı adımları [8]

Parça ve makinelerin hücelere atanması ağırlıklandırılmış toplamlara göre yapılır. Örneğin 1. makine için bu toplamlar  $\{0,03 ; 0,38\}$ 'dir. Bu durumda 1. makine 2. hücreye atanır. RSA algoritması için geliştirilen sözde kodlar şekil 4.4'de verilmektedir.

#### 4.2. Parçacık Sürü Optimizasyon Yöntemi

Parçacık Sürü Optimizasyonu (PSO) ilk defa Kennedy ve Eberhart [54] tarafından kuş ve balık sürülerinden esinlenerek ortaya atılmış bir yöntemdir. Kuş sürüleri, yiyeceğe ulaşmak için, yiyeceğe en yakın kuşun ve kendi eniyi konumlarından etkilenmektedir. Sosyal bilgi paylaşımını temel alan bu yöntem

---

**başla;**

$s \leftarrow 0$ ;

**tekrar et**

$s \leftarrow s + 1$ ;

$q \leftarrow 0$ ;

*bir rassal  $\pi_{m+n}$  permütasyonu üret*

**tekrar et**

$q \leftarrow q + 1$ ;

*desenlerdeki tüm elemanları sıfıra eşitle*

*$\pi[q]$ . girdi deseni oluştur*

*tüm çıktı nöronları (hücreler) için  $net_j$  değerlerini hesapla*

*kazanan çıktı nöronunu belirle*

*ağırlıkları güncelleştir*

$q=m+n$  olana **kadar devam et**;

**Until**  $s = n_{epoch}$  olana **kadar devam et**;

---

Şekil 4.4 RSA sözde kodlar

sürekli doğrusal olmayan fonksiyonların çözümü için önerilmiştir.

PSO, Genetik algoritmalar gibi popülasyon-temelli bir yöntemdir, popülasyon sürüyü oluştururken, sürüde yer alan bireyler ise parçacıkları oluşturmaktadır. Parçacık uzunluğu, probleme ilişkin çözüm uzayının boyutu kadardır. Parçacıklar, buldukları konumları uygunluk fonksiyonuna (amaç fonksiyonun) göndererek bir uygunluk değeri elde ederler. Aslında bu değer parçacıkların yani kuşların yiyeceğe olan uzaklığını ifade etmektedir.

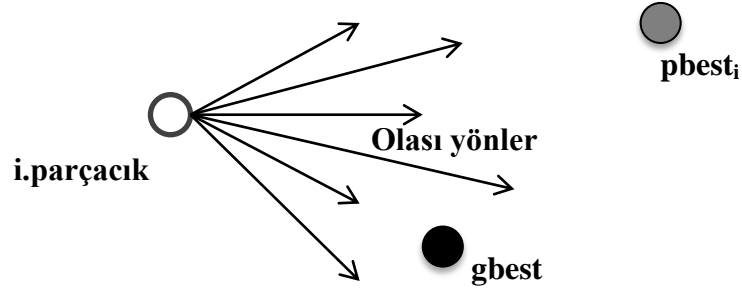
Parçacıklar eniyi değere ulaşmak için arama uzayını taramaktadır, eniyi değere doğru yönelme, her parçacığın o ana kadar elde ettiği kendi eniyi uygunluk değeri ile tüm parçacıklar içinden o ana kadar elde edilen eniyi uygunluk değerine bağlı olarak yapılır.

Yöntemde kullanılan parametreler; yöntemin sonlanma koşulu olan tekrar sayısı ve sürüde yer alacak parça sayısıdır. Parçacıkların konumları başlangıçta rasgele belirlenebileceği gibi problem doğrultusunda belirli değerler de verilebilir.

Fonksiyondaki değişken sayısı “ $n$ ” olarak gösterilirse her parçacık “ $n$ ” boyutlu uzayda ( $R^n$ ) arama yapmaktadır. Yani parçacık “ $n$ ” adet bileşene sahip olmaktadır.  $x_{in}^k$ ,  $i$ .parçacığın  $n$ . bileşeninin  $k$ . tekrarda aldığı değeri gösteren ifadedir. Her parçacığın  $k$ . tekrara kadar elde ettiği eniyi uygunluk değerine karşılık gelen bileşen değerleri  $pbest_i$ 'de tutulmaktadır. Aynı şekilde  $k$ . tekrara kadar tüm parçacıklar içinde elde edilen eniyi uygunluk değerine karşılık gelen bileşen değerleri de  $gbest$ 'de tutulur. Tüm bu terimler matris formunda gösterilirse:

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \dots & \dots & \dots & \dots \\ x_{i1} & x_{i2} & \dots & x_{in} \end{bmatrix}_{i*n}$$

$i$  adet parçacığın her biri “ $n$ ” boyutlu arama uzayında arama yapmaktadır.  $i$ . parçacığın o ana kadar elde ettiği eniyi uygunluk değerini veren koordinatlar  $pbest_i = [p_{i1} p_{i2} \dots p_{i3}]$  ile gösterilirken, tüm parçacıklar içinde elde edilen eniyi uygunluk değerine karşılık gelen koordinatlar  $gbest = [g_1 g_2 \dots g_3]$  (bütünsel) ile gösterilmektedir. PSO ile  $i$ . parçacığın  $pbest_i$  ve  $gbest$  değerlerine doğru nasıl hareket ettiği şekil 4.5’de verilmektedir. Parçacığın yön güncellemesi (4.4)’de verilirken, parçacıkların bir sonraki tekrarda alacağı değer (4.5)’de verilmektedir [54].



Şekil 4.5 PSO ile parçacığın pbest<sub>i</sub> ve gbest'e doğru hareketi [39]

$$V_i^{k+1} = V_i^k + c_1 \cdot rnd_1 \cdot (pbest_i^k - x_i^k) + c_2 \cdot rnd_2 \cdot (gbest^k - x_i^k) \quad v_i^k \in R \quad (4.4)$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \quad x_i^k \in R \quad (4.5)$$

Algoritma başlangıcında tekrar ve parça sayısı parametrelerinin yanında, “ $c_1$ ” ve “ $c_2$ ” parametrelerinin de belirlenmesi gerekir. Genelde  $0 < c_1 < 2$  ve  $0 < c_2 < 2$  olmak üzere farklı değerler alabilir,  $c_1$  kavramsal katsayı olarak tanımlanır ve parçacığın o ana kadar elde ettiği eniyi aday çözüme götüren adım büyüklüğünü ifade ederken  $c_2$  ise sosyal katsayı olarak tanımlanır ve bütünsel aday çözüme götüren adım büyüklüğü ifade eder. Arama uzayı belirli bir  $[-x_{max}, x_{max}]$  aralığında kalabilmesi için yönü ifade eden değerlerinde  $[-v_{max}, v_{max}]$  aralığında sabitlenmesi gerekmektedir. Eşitlikteki  $rnd_1$  ve  $rnd_2$  ise 0-1 aralığında düzgün dağılımdan üretilen rassal değerlerdir [55]. PSO algoritması için sözde kodlar Şekil 4.6’da verilmektedir [39].

### 4.3. Kesikli Parçacık Sürü Optimizasyon Yöntemi için Yeni bir Yaklaşım

İlk versiyonu sürekli değişkenlerin çözümü için geliştirilen PSO, Kennedy ve Eberhart [61] tarafından yapılan çalışmada ikili kesikli değişkenler için kullanılabileceği gösterilmiş ve kesikli değer alan bazı problemlere uygulanmıştır. Yöntemde hız vektörü bileşenleri olan  $v_{in}$ 'ler [0-1] aralığında kalmakta ve *sigmoid fonksiyon* yardımıyla yapılan karşılaştırma sonucunda  $x_{in}$ 'deler 0 veya 1 değerini almaktadır.

---

**başla**

uyum fonksiyonunun belirle  $f(x)$ ,  $x = (x_1, \dots, x_p)^T$

parçacık başlangıç konumlarını  $x_i$  ve hızlarını oluştur  $v_i$

başlangıç enküçük değeri belirle  $f_{min}^{t=0} = \min\{f(x_1), \dots, f(x_n)\}$  (at  $t=0$ )

**tekrar et**

$t = t + 1$

**tekrar et** (tüm parçacık ve boyutlar kadar)

yeni hız vektörünü türet  $v_i^{t+1}$

yeni parçacık konumlarını hesapla  $x_i^{t+1} = x_i^t + v_i^{t+1}$

yeni konumlara göre uyum değerini hesapla

güncel eniyi değeri bul  $f_{min}^{t+1}$

**sonlandır**

güncel bütünsel eniyi değeri ve konumu bul

sonlanma koşulu **boyunca devam et**

bütünsel eniyi değeri ve konumu çıktı olarak ver

**sonlandır**

---

**Şekil 4.6** Parçacık sürü optimizasyon sözde kodlar

$X_i^k = (x_{i1}^k, x_{i2}^k, \dots, x_{in}^k)$ ,  $x_{in} \in \{0,1\}$ , vektörü  $i$ . parçacığın  $k$ . tekrardaki bileşenlerinin “0” ya da “1” değerini aldığı göstermektedir. Parçacığın hızını gösteren vektör  $V_i^k = (v_{i1}^k, v_{i2}^k, \dots, v_{in}^k)$ ,  $v_{in} \in [0,1]$ , aralığında reel değer alır. Kesikli PSO yaklaşımında hız güncelleme, aynı sürekli yaklaşımda olduğu gibi (4.4) kullanılarak yapılır. Bununla beraber  $X_i^k$ 'nin alacağı değerler, eşitlik (4.6)'da verilen *sigmoid fonksiyon* yardımıyla bulunur. Fonksiyon değeri büyük olan bileşenin “1” değerini alma olasılığı yüksektir temeline dayanan bu yöntemde, üretilen rassal sayı  $s(v_{in}^k)$  değerinden daha küçük olması durumunda ( $rnd < s(v_{in}^k)$ ),  $x_{in}^k = 1$  olur.

$$s(v_{in}^k) = \frac{1}{1 + \exp(-v_{in}^k)} \quad (4.6)$$

Çalışma kapsamında önerilen yaklaşımda  $x_{in}$  değerlerinin [0-1] aralığında kalması sağlanarak, bu bileşenlere karşılık gelen değerler kesikli hale dönüştürülür. Böylelikle parçacıkların “n” boyutlu birim hiperküp içinde kalarak arama yapması sağlanır. Her parçacığın  $v_{in}$  değerleri (4.4)'e göre güncellendikten



sonra birim vektöre dönüştürülür. Algoritmada parçacıkların hız bileşenleri olarak birim  $v_{in}$ 'ler kullanılmaktadır.

PSO'nun temel mantığı, arama uzayını her parçacığın o ana kadar kendi eniyi koordinatlarına ve tüm parçacıklar içinden eniyi koordinatlara bağlı olarak taramasıdır. Bu mantığa bağlı kalınarak geliştirilen yaklaşımda, birim hiperküp içinde bulunan parçacığın  $x_{in}$  değerleri, hız bileşeni olan  $v_{in}$  değerlerine bağlı olarak hangi yönde hareket edecekleri belirlenmektedir. Eğer  $v_{in}$  değeri negatif ise  $x_{in}$  "0" yönünde hareket ederken, pozitif olması durumunda "1" yönünde hareket etmektedir. Böylelikle parçacığın belirlenen yönde hareket etmesi durumunda, birim hiperküpün hangi sınırına doğru yönleneceği belirlenmektedir.

Daha sonra parçacıkların hız vektörleri birim vektöre dönüştürülür, bununun nedeni parçacıkların her zaman birim hiperküp içinde kalmasını sağlamaktır. Parçacıkların yön bileşenlerinin ( $v_{in}$ ) işaretlerine göre hareket yönü belirlenmesinin ardından,  $x_{in}$  değerleri ile belirlen "0" ya da "1" değeri arasındaki mutlak fark bulunur, böylelikle parçacığın sınır değerlerine olan uzaklık bulunmuş olur.

Her parçacığın tüm  $x_{in}$  değerleri ile sınır değerleri olan 0 ya da 1 arasındaki uzaklıklar birim  $v_{in}$  değerine bölünerek o doğrultuda parçacığın birim  $v_{in}$ 'nin kaç katı mesafe gideceği belirlenir. Sınırlardan dışarı çıkmamak için her parçacığın  $v_{in}$  değerlerine bağlı kalarak belirlenen bu katlardan en küçüğü rassal bir sayı ile çarpılır ve mevcut  $x_{in}$  değerlerine eklenir. Böylelikle yeni  $x_{in}$  değerleri elde edilmektedir, yeni değerlerle birlikte arama uzayında komşu çözümlere geçiş sağlanmış olur. Kesikli değerlere geçiş ise  $x_{in}$ 'lerin aldıkları değerlerin hangi aralıkta olduğuna bakılarak karar verilir, örneğin parçacığın bileşeni [1,2] aralığında tam sayı alıyorsa  $x_{in}$  [0,0.5] aralığında "1" [0.5, 1.0] aralığında "2" değerini alır.  $v_{in}$  değerlerinin güncellenmesi PSO'da açıklanan şekilde yapılmaktadır. Tüm bu adımlar algoritma sonlanma koşuluna kadar yinelenmektedir.

#### **Açıklayıcı Bir Örnek:**

Örnekte kesikli PSO için önerilen yaklaşımda parçacıkların [0-1] arama uzayında nasıl hareket ettikleri gösterilmektedir. Çizelge 4.1'de parçacıkların başlangıç  $X_i$  değerleri [0,1] aralığında,  $V_i$  değerleri [-1,1] aralığında rasgele oluşturulmuştur.

**Çizelge 4.1** Parçacıkların başlangıç konum ve hız değerleri

Parçacık No	Konum		Hız		Hız vektörü uzunluğu
	$x_{i1}$	$x_{i2}$	$v_{i1}^{k+1}$	$v_{i2}^{k+1}$	
P1	0.14	0.64	0.47	0.43	0.64
P2	0.48	0.74	0.73	-0.24	0.77
P3	0.26	0.65	-0.18	0.38	0.42
P4	0.76	0.48	0.1	0.48	0.49
P5	0.38	0.4	-0.47	-0.24	0.53

Her parçacığın hız vektörü uzunluğu bulunduktan sonra Çizelge 4.2'deki gibi birim vektöre dönüştürülür.

Yukarıdaki çizelgede  $x_{i1}$  ve  $x_{i2}$  bileşenleri verilen parçacıkların, yönleri V vektörü ile belirlenmektedir. Önceden de bahsedildiği gibi her parçacık, karşı gelen yön vektörü doğrultusunda hareket eder, hareket yönü “-” ise “0” yönünde, “+” ise “1” yönünde ilerlemektedir. Parçacık koordinatları ile ilerleyecekleri yön arasındaki mutlak fark (sınıra uzaklık) bulunur ve birim hız vektörünün o yönde kaç kat gideceğini gösteren değerler Çizelge 4.3'de verilmektedir. Bu katların en ufağı birim hız vektörü ve rassal bir sayı ile çarpılarak, parçacığı o andaki koordinatına eklenir. Burada rassal değer olarak 0.7 olduğu varsayılmıştır.

Tüm bu işlem adımları gerçekleştirildiğinde parçacıkların yeni konumlarını ifade eden P1', P2', P3', P4' ve P5' simgeleri Şekil 4.7'de gösterilmektedir.

Parçacıklar yönleri doğrultusunda en fazla gidebileceği değer 0 ya da 1'dir, bu aralıktaki değerlere bakılarak kesikli değerlere dönüşüm yapılır. Örneğin P2 parçacığının [1-5] aralığında tam sayı değer aldığı varsayalım, yeni P2 parçacığının birinci bileşen değeri 0.84'dür, bu değer 0.8-1.0 aralığında yer aldığından karşılık gelen kesikli değer “5” olmaktadır.

**Çizelge 4.2** Parçacık birim hız vektörleri

Parçacık No	Birim			
	$x_{i1}$	$x_{i2}$	$v_{i1}^{k+1}$	$v_{i2}^{k+1}$
P1	0.14	0.64	0.74	0.68
P2	0.48	0.74	1.15	-0.38
P3	0.26	0.65	-0.28	0.60
P4	0.76	0.48	0.16	0.75
P5	0.38	0.4	-0.74	-0.38

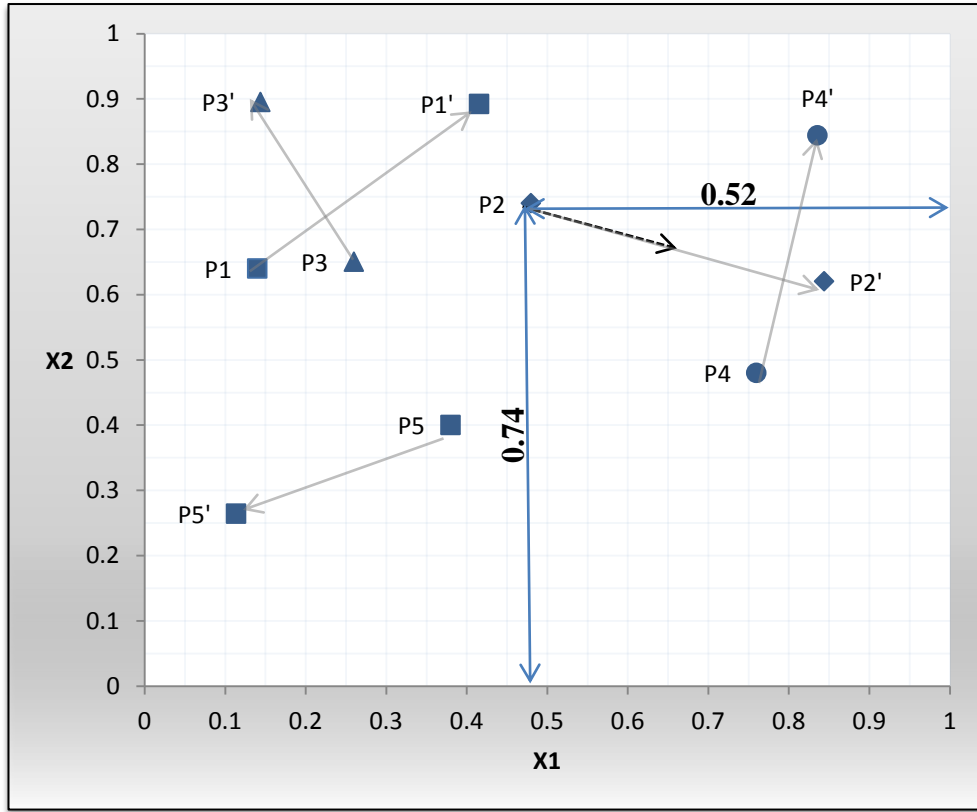
Çizelge 4.3 Yeni  $X_i$  değerlerine geçiş

Parçacık No	Yöne göre sınırlar		Sınır uzaklık		Kat edilebilecek mesafe (uzaklık/ birim V )		En küçük kat	Eklenecek hız değerleri		Yeni X değerleri	
	$v_{i1}^{k+1}$	$v_{i2}^{k+1}$	$x_{i1}^{k+1}$	$x_{i2}^{k+1}$							
P1	1	1	0.86	0.36	1.17	0.53	0.53	0.28	0.25	0.42	0.89
P2	1	0	0.52	0.74	0.45	1.96	0.45	0.36	-0.12	0.84	0.62
P3	0	1	0.26	0.35	0.92	0.59	0.59	-0.12	0.25	0.14	0.90
P4	1	1	0.24	0.52	1.53	0.69	0.69	0.08	0.36	0.84	0.84
P5	0	0	0.38	0.4	0.52	1.06	0.52	-0.27	-0.14	0.11	0.26

P2 parçacığının [0-1] aralığındaki eski ve yeni değerleri kullanılarak dönüştürülen kesikli değerler Çizelge 4.4’de verilmektedir.

#### Görsel Örnek:

Yeni geliştirilen PSO yaklaşımının başarısını test edebilmek için matematiksel ifadesi bilinen, içinde yerel maksimum ve yerel minimum noktaları



Şekil 4.7 Parçacıkların eski ve yeni koordinatları

**Çizelge 4.4** Kesikli değerler ve dönüşüm aralığı

	$x_{21}$	$x_{22}$	Değer Aralığı	
Sürekli P2	0.48	0.74	0.00 - 0.20	1
Kesikli P2	3	4	0.20 - 0.40	2
Sürekli P2'	0.84	0.62	0.40 - 0.60	3
Kesikli P2'	5	4	0.60 - 0.80	4
			0.80 - 1.00	5

barındıran “peaks” fonksiyonu örnek olarak kullanılmıştır. “Peaks” fonksiyonunun matematiksel ifadesi (4.7)’de, grafiği Şekil 4.8’de verilmektedir.

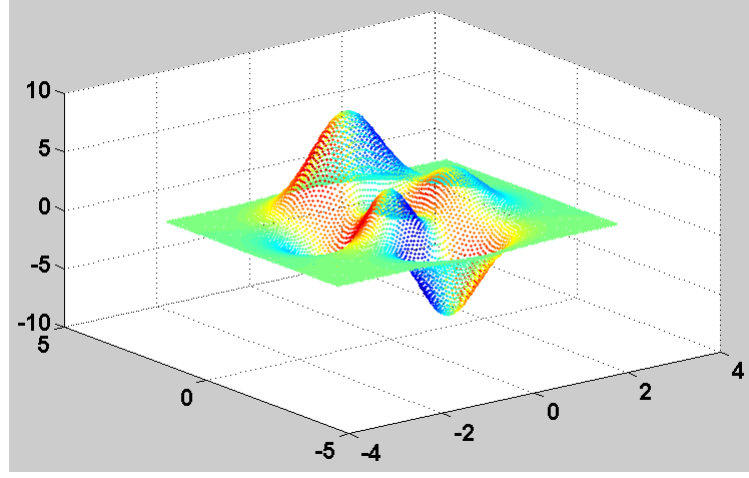
$$y = 3 \cdot (1 - x_1)^2 \cdot e^{[-x_1^2 - (x_2^2 + 1)^2]} - 10 \cdot \left(\frac{x_1}{5} - x_1^3 - x_2^5\right) \cdot e^{[-x_1^2 - x_2^2]} - \frac{1}{3} \cdot e^{[-(x_1 + 1)^2 - x_2^2]} \quad (4.7)$$

Bu fonksiyon  $x_1=0,2283$  ve  $x_2=-1,6255$  değerlerinde bütünsel enküçüğe ulaşır ve -6,5511 değerini alır. Bu değer sürekli değerler için geçerlidir. Yeni PSO yaklaşımı bütünsel enküçüğü bulmak için “peaks” fonksiyonuna uygulanmıştır. Ek-2’de parçacıkların 0-1 arama uzayındaki hareketleri, dönüşüm işlemi sonucunda aldığı kesikli değerler ve her tekrarda fonksiyonun aldığı eniyi uygunluk değeri gösterilmektedir. Dönüşüm işlemi sırasında kesikli değerleri [-5,6] aralığında değer aldığı varsayılmıştır.

Fonksiyondaki değişkenlerin kesikli değer alması durumunda, minimum değer -4,7574 olmaktadır ve bu değeri sağlayan kesikli değişkenler  $x_1=0$  iken  $x_2=2$ ’dir. Eniyi kesikli X değerleri gbest değerlerinin aldığı kesikli değerlere karşılık gelmektedir. Tekrar sayısındaki artışla birlikte parçacıklar eniyi değere ulaşmaktadır.

#### **4.4. Alternatif Rotalı Hücre Oluşturma Problemlerinin Çözümü için Melez Bir Yaklaşım**

RSA, daha önce tek rotalı hücre oluşturma problemlerinin çözümünde kullanılmıştır. Bu çalışmada, alternatif rotalı hücre oluşturma problemlerinin çözümü için PSO ve RSA’larının birlikte kullanıldığı melez bir yaklaşım önerilmiştir.



Şekil 4.8 “Peaks” fonksiyonunun grafiği

Öncelikle problemde alternatif rotası olan parçalar belirlenmektedir. Böylece parçacıklar belirlenen alternatif rotalı parça sayısı kadar bileşene sahip olmaktadır. Parçacığın her bileşen değeri hangi parçanın hangi alternatif rotayı seçtiğini gösterir. Bu bileşen değerlerine göre her parçacık, bileşenlerinde yer alan alternatif rotalara göre indirgenmiş temas matrisini temsil etmektedir. Yaklaşımında PSO alternatif rotaların seçimini yaparken RSA ise rotalara göre belirlenen temas matrisindeki parça ve makineleri gruplayarak hücreleri oluşturmaktadır.

Önerilen melez algoritmada 2 döngü vardır: dış döngüde yer alan PSO algoritması, alternatif rotaları seçer ve indirgenmiş matrisi RSA'na gönderir, iç döngüde yer alan RSA ise blok diyagonal matris formunu oluşturarak hücreleri belirler. Oluşturulan bu hücrelerin gruplama etkinlik değerleri hesaplanarak, uygunluk değeri olarak parçacıklara atanır. Parçacıklara ait hız vektörünün güncellenmesi için gerekli olan  $pbest_i$  ve  $gbest$  değerleri, parçacıkların her tekrarda elde ettiği uygunluk değerleri göz önüne alınarak belirlenmektedir. Önerilen melez yaklaşım algoritmasına ait sözde kodlar şekil 4.9'da verilmektedir.

**PSO algoritmasında başlangıç durumunun oluşturulması:** Parçacık sayısı,  $c_1$ ,  $c$  ve tekrar sayısı başlangıç parametresi olarak belirlenir. PSO algoritmasına ait parçacık konumları ( $X$ ) ve hızlar ( $V$ ) rasgele oluşturulur. Parçacık boyutu temas

---

**başla**

*Gruplama etkinliğini amaç fonksiyonu olarak belirle*

*Parçacık başlangıç konumlarını ( $X[0-1]$ ) ve hızlarını ( $V[-1,1]$ ) rasgele oluştur*

*Hız vektörünü birim vektöre dönüştür*

*Sürekli parçacık değerlerini kesikliye dönüştür*

*pbest ve gbest değerlerini belirle*

**tekrar et**

$t = t + 1$

**tekrar et** (bütün parçacıklar için)

**başla** (RSA);

$s \leftarrow 0$  ;

**tekrar et**

$s \leftarrow s + 1$ ;

*Her parçacık için amaç fonksiyonu değerini hesapla*

$s = n_{epoch}$  (RSA tekrar sayısı) **olana kadar devam et**

**sonlandır**

**sonlandır**

*pbest ve gbest değerlerini güncelle*

*Hız vektörünü güncelle ( $v_i^{t+1}$ )*

*Hız vektörünü birim vektöre dönüştür*

*Parçacıkların yeni konumlarını belirle ( $x_i^{t+1} = x_i^t + v_i^{t+1}$ )*

*Sürekli parçacık değerlerini kesikliye dönüştür*

*PSO tekrar sayısı **boyunca devam et***

*çıktı gbest*

**sonlandır**

---

**Şekil 4.9** Melez yaklaşıma ait sözde kodlar

matrisinde alternatif rotası olan parça sayısı kadardır. Her boyutta [0-1] aralığında bir değer bulunmaktadır ve her boyut için 1 ile parçanın alternatif rota sayısı kadar rassal bir değere dönüşüm uygulanmaktadır. Parçacıkların hız vektörleri [-1,1] aralığında rasgele üretildikten sonra birim vektöre dönüştürülerek, yeni değerler kullanılmaktadır.

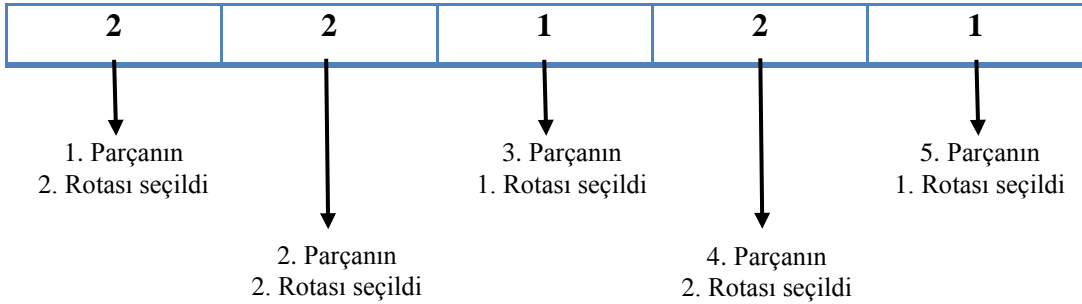
Örnek olarak Kusiak [56] tarafından Şekil 4.10'da verilen problem dikkate alınmıştır: Bu temas matrisi beş parça ve dört makineden oluşmaktadır. İlk parça üç diğerleri ise ikişer alternatif rotaya sahiptir. Parçacık boyutu, her parçanın alternatif rotası olduğu için beştir. Bu durumda her parçacığın bileşen sayısı beş

Parça No	alt. rot.	Makineler			
		1	2	3	4
1	1			1	1
1	2		1		1
1	3	1	1		
2	1		1	1	
2	2	1		1	
3	1	1			1
3	2		1		1
4	1	1			1
4	2	1		1	
5	1			1	1
5	2	1			

Şekil 4.10 Alternatif rotalara sahip bir parça-makine (temas) matrisi

olmaktadır. Şekil 4.11’de bir parçacık örnek olarak verilmektedir. Bu parçacığın her boyutu hangi parçanın hangi alternatif rotasının seçildiğini göstermektedir. Böylece parçacık sayısı kadar alternatif rotalardan arındırılmış parça-makine matrisi elde edilir ve bu matrislerin her biri rekabetçi sinir ağına gönderilir.

**Rekabetçi Sinir Ağının Çalıştırılması:** RSA için tekrar sayısı ve öğrenme oranı parametreleri başlangıçta belirlenmektedir. Alternatif rotalardan arındırılmış temas matrisinin her biri rekabetçi sinir ağı algoritmasına gelir. RSA, eş zamanlı olarak makine ve parçaları gruplar. Gelen her bir matris için hücre o-



Şekil 4.11 Örnek bir parçacık ve boyutları

luşturma işlemi tamamlandıktan sonra, oluşan diyagonal matrisin gruplama etkinlik değeri RSA'ya gelen parçacığın uygunluk değeri olarak hafızada tutulur. PSO'nun ilk tekrarında her parçacığın konumu  $pbest_i$ 'de, etkinlik değeri de  $fbest_i$ 'de tutulur. Bununla birlikte eniyi etkinlik değerine sahip parçacığın konumu  $gbest$ 'de tutulurken, eniyi değer de  $fgbest$ 'de tutulmaktadır.

**Parçacık hızlarının ve konumlarının güncellenmesi:** parçacıklara ait hız vektörleri olan  $V_i$ 'ler eşitlik 4.4 kullanılarak güncellenir. Daha sonra hız vektörleri birim vektöre dönüştürülür, önceden bahsedilen işlemler de gerçekleştirilerek parçacıkların  $[0,1]$  aralığında yeni konumları olan  $X_i$ 'ler belirlenmiş olur. Sürekli parçacık değerleri, kesikli değerlere dönüştürülerek seçilen parça alternatifleri belirlenir. Böylece RSA'a tekrar gönderilecek alternatif rotalardan arındırılmış temas matrisleri belirlenmiş olur.

**RSA'nın tekrar çalıştırılması, parçacık hız ve konumlarının yeniden güncellenmesi:** parçacık sayısı kadar alternatif rotalardan arındırılmış temas matrisi tekrar RSA'na gönderilir. RSA her biri için yeni etkinlik değerlerini hesaplar böylece PSO için gerekli uyum fonksiyonu değerleri hesaplanmış olur. Her bir parçacığın etkinlik değeri, bir önceki etkinlik değeri olan  $fbest_i$  ile karşılaştırılır, eğer parçacık daha iyi bir etkinlik değerine sahipse  $fbest_i$  güncellenir. Aynı zamanda parçacığın yeni etkinlik değerleri  $fgbest$  değeri ile karşılaştırılır ve daha iyi olması durumunda güncellenir. Yeni  $pbest_i$  ve  $gbest$  konumlarına göre parçacık hızları (4.4)'e göre tekrar güncellenir. Önceki bahsedilen işlemler tekrar edilir.

**Bütün adımların PSO için belirlenen tekrar sayısı kadar tekrarı:** ikinci tekrarda birlikte uygunluk fonksiyonundan elde edilen etkinlik değerleri kendi eniyi uygunluk değeri ve tüm parçacıklar içindeki eniyi uygunluk değerine göre karşılaştırılır ve bunlara göre hız güncellemesi yapılır. Parçacıkların her zaman  $[0,1]$  aralığında kalması için birim hız vektörlerine göre hareketler belirlenir. Sonuç itibarıyla birim hiperküp içinde kalarak sürekli değer alan parçacıklardan, seçilen alternatiflere göre tamsayı alan parçacıklara geçiş sağlanarak RSA parça makine gruplarını oluşturur. Çalışmada sonlandırma koşulu olarak PSO için geçerli tekrar sayısı alınmıştır. Alternatif rotaları seçen dış döngü yeterli sayıya



	M2	M4	M1	M3
P1(2)	1	1		
P3(2)	1	1		
P2(2)			1	1
P4(2)			1	1
P5(2)			1	0

Şekil 4.12 Alternatif rotalı hücre oluşturma probleminin çözümü

ulaştığında eldeki eniyi fgbest değerine sahip blok diyagonal matris, gruplama etkinliği eniyi olan hücre oluşumunu ifade etmektedir. Şekil 4.10'da verilen problemin çözümünü ifade eden blok köşegen matris Şekil 4.12'de verilmektedir.

Alternatif rotalı hücre oluşturma problemlerinin çözümünde başarılı sonuçlar veren PSO – RSA melez yönteminin yanında, sadece PSO veya Tavlama Benzetimi – RSA melez yönteminin kullanıldığı iki algoritma daha tasarlanmıştır.

Problem çözümü için tasarlanan PSO yaklaşımında yer alan parçacıklar, seçilen alternatif rotalarla birlikte hücrelere atanan makine ve parçaları temsil etmektedir. Şekil 4.10'da yer alan problem dikkate alındığında, çözümü ifade eden bir parçacık yapısı Şekil 4.13'deki gibi olmaktadır. Verilen parçacığın birinci bölümünde seçilen alternatif rotalar, ikinci bölümde parçalar ve üçüncü bölümde ise makineler yer almaktadır. Örneğin üçüncü parçacık ikinci alternatif rota seçilirken, dördüncü parça ikinci hücreye, ikinci makine birinci hücreye atanmıştır.

Sadece PSO yönteminin kullanıldığı algoritma PSO – RSA melez yaklaşım algoritmasına çok benzemektedir, algoritmada RSA yer almadığından, parçalar melez yaklaşımdaki gibi sadece alternatif rotaları seçmemekte aynı zamanda hücrelere atanacak parça ve makineleri de belirlemektedir. Melez yaklaşıma göre sadece PSO yaklaşımının kullanıldığı algoritmada parçacık boyutları parça ve makine sayısı kadar daha artmaktadır. Algoritma adımlarındaki işlemler, melez yaklaşımda anlatıldığı gibi birim hiperküp üzerinden gerçekleştirilir. Parçacıkların

Seçilen alternatif rotalar					Parçalar					Makineler			
1	2	3	4	5	1	2	3	4	5	1	2	3	4
2	2	2	2	2	1	2	1	2	2	2	1	2	1

Şekil 4.13 Sadece PSO yöntemi için parçacık yapısı

birim hiperküp içindeki konumları, hız vektörü güncelleştirilmesi ve kesikli değerlere geçişleri melez yaklaşımdaki gibidir.

Sofianopoulou [59] tarafından verilen P20 (EK-3’de verilmiştir) problemi için RSA – PSO melez yaklaşım algoritması 100 parçacık ve 200 tekrar için çalıştırıldığında, eniyi gruplama etkinlik değeri olan 0.4931 değerine 1405 saniye sonra ulaşmaktadır; aynı problem için sadece PSO algoritmasına dayanan yaklaşım algoritması 100 parçacık ve 500 tekrar için çalıştırıldığında etkinlik değeri 0.3526 bulunmuştur. Yaklaşım birkaç problemde daha denenmiştir, küçük boyutlu problemlerde literatürdeki eniyi sonuçlara ulaşabilen sadece PSO yaklaşımı, büyük boyutlu problemlerde elde edilen eniyi sonuçların %20 -%30 gerisinde kalmıştır. Bunun nedeni olarak da parçacıkların arama yaptığı çözüm uzayının boyutunun PSO – RSA yöntemine karşın çok daha büyük olmasından kaynaklandığı düşünülmektedir.

Alternatif rotalı hücre oluşturma problemlerinin çözümü için önerilen diğer yöntem Tavlama Benzetimi – RSA melez yaklaşımıdır. Tavlama benzetimi başlangıç parametrelerinden olan, başlangıç sıcaklığı, sıcaklık azaltma periyodu, sıcaklık azaltma oranı ve tekrar sayısı belirlendikten sonra rasgele bir çözüm üretilmektedir. Üretilen bu çözüm PSO – RSA melez yaklaşımındaki parçacık yapısı gibidir. Tavlama benzetimi alternatif rotaları seçer, böylece alternatif rotalardan arındırılmış temas matrisi RSA’ya gönderilerek hücreler oluşturulur. Algoritma adımları rasgele bir başlangıç çözüm ile başlar, sonra komşu çözüme geçiş sağlanır, komşu çözümün gruplama etkinlik değeri önceki çözüm değerinden daha iyi olması durumunda direk güncellenirken daha kötü olması durumunda Tavlama Benzetimine göre  $P = e^{-\left(\frac{\Delta}{T}\right)}$ ,  $\Delta = \frac{f(S_n) - f(S)}{f(S_n)} * 100$  olasılığı ile kabul edilir. Algoritma adımlarında komşu çözümlere geçildikçe eldeki eniyi değer ile karşılaştırma yapılır. P20 problemi için Tavlama Benzetimi – RSA melez algoritması 4000 tekrar için çalıştırıldığında 642. saniyede 0.4452 gruplama etkinlik değerine ulaşılmıştır. Bu sonuç ulaşılan eniyi değer olan 0.4931’in yaklaşık %10 altında kalmaktadır, birkaç büyük boyutlu problem için daha çözüm arandığında yaklaşık aynı oranda eniyi değerlerin altında kaldığı gözlemlenmiştir.

#### 4.5. Problem Çözümü için Tasarlanan Program

Alternatif rotalı hücre oluşturma problemlerinin çözümü için geliştirilen algoritma Excel ortamında Visual Basic for Applications (VBA) dili kullanılarak kodlanmıştır. Literatürde yer alan test problemleri bu program ile çözdürülmüştür.

Yazılan programın arayüzü şekil 4.12’de verilmektedir. Arayüzün üst bölgesinde PSO için parçacık sayısı, tekrar sayısı,  $c_1$ ,  $c_2$  parametrelerinin yanı sıra oluşturulması istenen hücre sayısı da başlangıç parametresi olarak verilir. Rekabetçi Sinir Ağına ait öğrenme oranı ve tekrar sayısı da başlangıçta belirlenen parametrelerdendir.

Arayüzün alt kısmında probleme ait temas matrisi yer almaktadır. Matrisin ilk sütununda parça numaraları, ikinci sütununda parçaların alternatif rotaları ve matrisin ilk satırında ise bu rotalarının işlem gördüğü makine numaraları gösterilmektedir.

Arayüzde “A14” hücresinden itibaren probleme ait temas matrisi oluşturulması ve yönteme ait gerekli parametrelerin girilmesi ile birlikte problem

Alter. Rota	M1	M2	M4	M3	M5	M6
P1(1)	1	1	1			
P4(2)	1	1	1			
P6(1)	1	1	0	1		
P8	0	1	1			
P2				1	1	1
P3				1	1	1
P5(1)		1		1	1	1
P7(2)				1	1	1

Şekil 4.14 Excel ortamında tasarlanan programın arayüzü

“çöz” butonu ile çalıştırılır. Girilen temas matrisini takip eden birkaç sütun sonra elde edilen çözüme ait blok diyagonal matris yazdırılmaktadır. Bloklar incelendiğinde hücrelerde yer alan makine ve parçaların hangilerinin gruplandığı ayrıca hangi rotaların izlendiği görülmektedir. İki matris arasında kalan değerler, algoritmanın dış döngüsü boyunca elde edilen eniyi gruplama etkinlik değerlerini göstermektedir. Bu değerler incelendiğinde parçacıkların daha iyi değerlere doğru yönlendiği görülmektedir.

Ayrıca yönteme ait algoritma Google Spreadsheets ortamında JavaScript dili kullanılarak da kodlanmıştır. Program arayüzü ve kullanımı yukarıda bahsedilen Excel programının aynısıdır. Şekil 4.13’de program arayüzü verilmektedir.

Google spreadsheets ortamında tasarlanan program büyük boyutlu problemleri çözememektedir. Bunun nedeni de Google’ın tekrar sayısına getirdiği limittir. İleride bu limitlemenin kaldırılması sonucu büyük boyutlu problemler de çözülebilecektir. Bu ortamda yazılan programlar diğer kullanıcılarla paylaşılabilir, bu sayede literatürde yer alan bazı problemlere erişim kolaylıkla sağlanabilecek ve bunun yanında hücre oluşturma problemlerinin çözümü için web tabanlı bir karar destek sistemi gibi kullanılabilir.

Parça No	Alter. Rota	Makine No	1	2	3	4	5	6	
1	1	1	1	1	1				0.7778
2	1	2		1	1	1	1	1	0.7778
3	1	3		1	1		1	1	0.7778
4	2	1			1		1	1	0.8148
5	3	1			1		1	1	0.8462
6	4	1	1			1			0.8462
7	4	2	1	1			1		0.8462
8	5	1		1	1		1	1	0.8462
9	5	2			1			1	0.8462
10	6	1	1	1	1				0.8462
11	6	2		1	1			1	
12	7	1			1		1	1	
13	7	2			1		1	1	
14	7	3			1			1	
15	8	1			1		1		

	M1	M2	M4	M3	M5	M6
F1(1)	1	1	1			
F4(2)	1	1	1			
F8(2)	1	1	0			1
P8	0	1	1			
P2				1	1	1
P3				1	1	1
F5(1)		1		1	1	1
F7(2)				1	1	1

Etkinlik: 0.8462

Şekil 4.15 Google Spreadsheets ortamında tasarlanan program arayüzü

Yapılan denemeler sonucunda, genelde  $c_1$ ,  $c_2$  katsayılarının 2 deęerinde, öğrenme oranında da 0.05 deęerinde iyi sonuçlar verdięi görölmüştür. Bunun yanında parçacık sayısı, PSO ve RSA tekrar sayıları problem boyutuna göre düzenlenmesi gerekmektedir.

## 5. HESAPSAL SONUÇLAR

Geliştirilen melez yaklaşım, literatürde bulunan alternatif rotalı hücre oluşturma problemlerine uygulanmıştır. Uygulama sırasında, enbüyük hücre büyüklüğü, üretim hacmi, farklı maliyetler gibi kısıtlar göz önünde bulundurulmamış, sadece alternatif rotalar dikkate alınmıştır.

İstenen hücre sayısı başlangıçta belirlenmektedir ve algoritma bu değere göre hücreleri oluşturmaktadır. Parçaların farklı alternatif rotaları arasından sadece bir tanesi seçildiğinden, rotaya göre işlem göreceği makineler belirlenmiş olur.

Yaklaşımın amacı hücreler arası taşımayı enküçükmektir, bunun için de temel alınan değerlendirme kriteri gruplama etkinliği olmuştur. Gruplama etkinliğinin “1” değerine ne kadar yaklaşırsa hücre bağımsızlıkları o kadar fazla olmaktadır. Hücre dışı istisnai elemanlar hücreleri birbirine bağımlı hale getirirken, hücre içinde boş elemanlarda makine kullanım oranını düşürmektedir.

Parçaların farklı makinelerde işlenmesi veya aynı makineden (kopya) birden fazla olması durumunda alternatif rotaların varlığından bahsedilmektedir. Çizelge 5.1’de çözüm aranan test problemlerinin listesi verilmektedir. P1-P16 numaralı problemlerdeki alternatif rotalar, parçaların farklı makinelerde işlem görmesini ele alırken, P17-P21 problemleri kopya makine olması durumunda oluşan alternatif rotaları ele almaktadır. Listede probleme ilişkin kaynak, makine çeşidi, makine sayısı, parça sayısı, rota sayısı bilgilerine yer verilerek problemin büyüklüğü gösterilmektedir. Rota sayısı arttıkça problemlerin zorluğu da kaçınılmaz olarak artmaktadır. Çizelgenin sütunlarında makine çeşidi, makine, parça ve rota sayıları verilmektedir. Makine çeşidi ve makine sayısı arasındaki fark kopya makinelerin sayısını vermektedir.

Çizelge 5.1 Test problemleri listesi

Problem No	Kaynak	Makine çeşidi	Makine sayısı	Parça sayısı	Rota sayısı
P1	Kusiak [56]	4	4	5	11
P2	Adil ve ark. [26]	4	4	5	12
P3	Won ve Kim [57]	4	4	4	8

**Çizelge 5.1 (Devam)** Test problemleri listesi

<b>P4</b>	Gupta [58]	5	5	7	11
<b>P5</b>	Gupta [58]	6	6	8	15
<b>P6</b>	Moon ve Chi [1]	6	6	6	13
<b>P7</b>	Sankaran ve Kasinlingam [1]	6	6	10	20
<b>P8</b>	Won ve Kim [57]	7	7	10	23
<b>P9</b>	Logendran [1]	7	7	14	32
<b>P10</b>	Kazerooni [27]	8	8	13	26
<b>P11</b>	Adil ve ark. [26]	10	10	10	24
<b>P12</b>	Won ve Kim [57]	11	11	10	22
<b>P13</b>	Sofianopoulou [59]	12	12	20	26
<b>P14</b>	Nagi [24]	15	15	15	27
<b>P15</b>	Won ve Kim [57]	26	26	28	71
<b>P16</b>	Kazerooni [27]	30	30	40	88
<b>P17</b>	Kasilingam and Lashkari [60]	10	13	15	55
<b>P18</b>	Harhalakis [1]	5	7	6	16
<b>P19</b>	Sofianopoulou[59]	12	14	20	45
<b>P20</b>	Sofianopoulou [59]	16	18	30	59
<b>P21</b>	Nagi [24]	17	20	20	51

Önerilen yaklaşımın başarısını test edebilmek için “gruplama etkinliği” karşılaştırma kriteri olarak alınmıştır. Saraç ve Özçelik [25], Wu ve ark. [33] ve Vin’in [1] yapmış oldukları çalışmalarda alternatif rotalı hücre oluşturma problemlerini temel almaktadır. Her bir çalışmanın ortak noktası, gruplama etkinliğini enbüyükmeyi amaçlamasıdır. Bu çalışmaların karşılaştırmaya alınma sebebi, elde ettikleri etkinlik değerlerinin yüksek olması ve test problem listelerinin geniş olmasıdır.

Çizelge 5.2’nin son ana sütunu önerilen yaklaşıma ait parametreleri, problem çözümlerine ait gruplama etkinlik değerlerini ve bu değerlere ulaşma zamanlarını saniye cinsinden vermektedir. Yaklaşıma ait algoritma, 2x2.93Ghz 6 çekirdek işlemciye sahip bir iş istasyonunda çalıştırılmıştır.

Parçaların farklı makinelerde işlenebilmesi durumunda, önerilen yaklaşımın performansını değerlendirmek amacıyla literatürde yer alan 16 test problemi (P1-

Çizelge 5.2 Alternatif rotalı hücre oluşturma problemleri için çözüm yöntemlerinin karşılaştırılması

Problem No	Altrot_GA [25]		Melez Tavlama Benzetimi [33]		Simogga [1]		Önerilen Yaklaşım					
	Hücre Sayısı	Gruplama Etkinliği	Hücre Sayısı	Gruplama Etkinliği	Hücre Sayısı	Gruplama Etkinliği	Hücre Sayısı	Gruplama Etkinliği	Parçacık Sayısı (PSO)	Tekrar Sayısı (PSO)	Tekrar Sayısı (RSA)	Süre (sn)
P1		0.9000	2	0.9000	2	0.9000	2	0.9000	5	10	7	1
P2		0.9091					2	0.9091	10	20	7	1
P3			2	1.0000	2	1.0000	2	1.0000	5	20	7	1
P4					3	0.9333	3	0.9333	5	20	7	1
P5					2	0.6667	2	0.8462 <sup>S</sup>	5	20	7	1
P6		0.8333	2	0.8333	3	0.8000	2	0.8333 <sup>S</sup>	5	20	7	1
P7		0.7222	2	0.7222	4	0.7222	2	0.7222	5	20	7	1
P8		0.8148	3	0.8148	3	0.8148	3	0.8148	20	20	7	2
P9			3	0.6944	3	0.7027	3	0.7027 <sup>T</sup>	20	30	10	4
P10					4	0.7381	3	0.8378 <sup>S</sup>	20	20	10	2
P11			3	0.8286	5	0.8333	3	0.8333 <sup>T</sup>	20	30	7	1
P12		0.8065	4	0.8065	3	0.8065	4	0.8065	20	30	7	4
P13			4	0.4947	4	0.4912	4	0.5000 <sup>T,S</sup>	20	200	10	111
P14					4	0.4915	2	0.5274 <sup>S</sup>	20	20	7	2
P15	7	0.7203	6	0.7248	7	0.5412	6	0.7248 <sup>G,S</sup>	100	200	7	335
P16					7	0.5511	7	0.7021 <sup>S</sup>	100	200	10	1415
P17	5	0.6667	3	0.619	5	0.6452	3	0.6429 <sup>T</sup>	100	200	10	35
P18					3	0.7273	3	0.7273	10	20	7	1
P19	4	0.5300	4	0.5429	3	0.5085	4	0.5300 <sup>S</sup>	100	200	10	194
P20	5	0.4626	6	0.4745	7	0.4300	6	0.4931 <sup>G,T,S</sup>	100	200	20	1405
P21		0.7952	5	0.7952	5	0.6040	5	0.7952 <sup>S</sup>	100	200	10	163

G - Altrot\_GA karşılaştırma yöntemine göre daha iyi sonuç elde edilmiştir

T - Melez Tavlama Benzetimi karşılaştırma yöntemine göre daha iyi sonuç elde edilmiştir

S - Simogga karşılaştırma yöntemine göre daha iyi sonuç elde edilmiştir



P16) çözdürülmüştür. Çizelge 5.2’de Altrot\_GA, Melez Tavlama Benzetimi ve Simogga yöntemleri önerilen yöntem ile karşılaştırılmıştır. Önerilen yaklaşım, çoğu problem için literatür değerlerini yakalamış, bazıları için de daha iyi sonuçlar elde etmiştir. Çizelgedeki gruplama etkinliği sütununda yer alan değerlerin üst indisleri, diğer yöntemlere göre elde edilen daha iyi sonuçları göstermektedir.

Aynı makineden birden fazla olması durumu için literatürde yer alan problemler (P17-P21) için temas matrisi tekrar düzenlenmiş ve çözdürülmüştür. Kopya makinelerde işlem gören her parça için alternatif rota oluşturulmuştur. Bir parçanın 1, 3, 4, 5 nolu makinelerde işlem gördüğünü ve 5. Makineden 2 tane olduğunu varsayalım. Parçanın birinci alternatif rotası 1, 3, 4,5(1) olurken, ikinci alternatif rotası da 1, 3, 4,5(2) olmaktadır. Kopya makine olması durumunda parçaların izleyecekleri alternatif rotalar da hızla artmaktadır. Melez yaklaşım algoritmasıyla iyi sonuçlar elde edilmesine rağmen, parçacık boyutunun büyüdüğü durumlarda algoritmanın eniyi çözüme ulaşmak için uzun süre çalıştırılması gerekmiştir.

Çizelge 5.2’de yer alan problemler ve çözümleri Ek-3’de verilmektedir.

## 6. SONUÇ VE ÖNERİLER

Çalışmada Grup Teknolojileri, Hücresel Üretim ve Hücre oluşturma problemleri hakkında genel bilgi verilmiştir. Literatürde yer alan çalışmalar incelendiğinde çoğunun alternatif rotaları ele almadığı görülmüştür. Bu kriterin probleme dahil edilmesi ile daha gerçekçi çözümler elde edilebilirken, problemin zorluğu da artmaktadır.

Alternatif rotası olmayan problemlerin çözümü için başarılı sonuçlar veren RSA yaklaşımı, PSO ile birleştirilerek alternatif rotalar için çözüm arayan bir yöntem geliştirilmiştir. Literatürde PSO yöntemi hücre oluşturma problemlerinin çözümünde kullanılmış olmasına rağmen, alternatif rotalar göz önünde bulundurulmamıştır, PSO ve alternatif rotalar ilk defa bu çalışmada ele alınmıştır. Bununla birlikte arama uzayı birim hiperküp düzeyinde tasarlanarak, kesikli değerlere geçiş sağlayan yeni bir yapı geliştirilmiştir.

RSA ve PSO'nun birlikte çalışabileceği bir algoritma tasarlandıktan sonra problem çözümleri için MS Excel ortamında Visual Basic for Applications (VBA) dili kullanılarak bir program yazılmıştır. Literatürde yer alan problemler çözdürülerek karşılaştırma yapılmıştır. Geliştirilen yöntem sayesinde çoğu problemde eniyi değerlere ulaşılmış ya da daha iyi sonuçlar elde edilmiştir. Programın farklı bir versiyonu Google Spreadsheets ortamında da kodlanmıştır. Problem listesinde yer alan tüm problemler bu ortama aktarılmıştır. Küçük boyutlu problemlerde çözüm bulan bu ortam büyük boyutlu problemlerde Google'ın tekrar sayısına koyduğu limit dolayısıyla çözüm bulamamaktadır. Bu sistem ileride, limit sayısının kaldırılması ve kullanıcılar için servis sağlayan bir yapıya dönüştürülmesi durumunda, internet ortamında problem çözümünde kullanılan bir karar destek sistemi olarak hizmet verebilir.

Hücre oluşturma probleminde alternatif rotalar göz önüne alınmasına rağmen, gelecek çalışmalarda kapasite, talep kısıtlarının da eklenerek algoritmanın yeniden tasarlanması sonucunda daha gerçekçi çözümler elde edilebilir. Ayrıca PSO'nun yapısı gereği, uzayda arama yapılan noktaların bir listesi tutulmamaktadır. Algoritmanın daha hızlı çalışması için Yasaklı Aramada kullanılan yasak listesine benzer yaklaşımla algoritmaya dahil edilebilir.

Birim hiperküpü temel olarak arama yapan bu yeni yaklaşım sayesinde, alternatif rotalı hücre oluşturma problemlerinde başarılı sonuçlar elde edilmiştir. Yaklaşımın kolay uygulanabilirliği, diğer kombinatorik eniyileme problemlerinin çözümünde de kullanılabileceği yönünde işaretler vermektedir.

## KAYNAKLAR

- [1] Vin, E., *Genetic algorithm applied to generalized cell formation problems*, Doktora Tezi, Brüksel Üniversitesi, Fen Bilimleri Enstitüsü, Brüksel, 2010.
- [2] Askin R.G. ve Standridge C.R., *Modeling and analysis of manufacturing systems*, Wiley and Sons, New York, 1993.
- [3] Heyer, N. L. ve Wemmerlov, U., “Group technology and productivity” *Harvard Business Review*, 141-149, 1984.
- [4] Selim, H. M., Askin, R. G. ve Vakharia, A. J., “Cell formation in group technology: review, evaluation and directions for future research,” *Computers ind. Engng*, 34, 3-20, 1998.
- [5] Baalkur, A., ve Steudel, J. H “A within-cell utilization based heuristic for designing cellular manufacturing systems”, *International Journal Of Pruduction Research*, 25(5), 639-665, 1987.
- [6] James, T., Brown, E.C., ve Keeling, J.B., “A hybrid grouping genetic algorithm for the cell formation problem” , *Computers & Operations Research*, 34, 2059-2079, 2007.
- [7] King, J. R., ve Nakornchai V., “Machine component group formation in group technology review and extension”, *International Journal Of Pruduction Research*, 20, 117-133, 1982
- [8] Öztürk, G., Öztürk, Z. K. ve İşlier, A. A., “A comparison of competitive neural network with other AI techniques in manufacturing cell formation,” *Advances in Natural Computation*, 4221, 575-583, 2006.
- [9] Singh, N., *Systems approach to computer-integrated design and manufacturing*, New York, John Wiley & Sons, 1996.
- [10] Kobu, B., *Üretim yönetimi*, Beta Basım A.Ş., İstanbul, 2006.
- [11] Öztürk, Ö., *Tasarlanana hücreli üretim sisteminin mevcut üretim sistemi ile karşılaştırılması: Ford otosan fabrikası şanzuman üretim bölümü için bir uygulama*, Doktora Tezi, Anadolu Üniversitesi, Sosyal Bilimler Enstitüsü, Eskişehir, 2007.
- [12] Aktürk, M. S. ve Yayla, H. M., “Management of product variety in cellular manufacturing systems”, *International Journal of Flexible Manufacturing Systems*, 17, 93-117, 2006.

- [13] Tompkins, J. A., White, J. A., Bozer, Y. A. ve Tanchoo, J. M. A., *Facilities planning*, John Wiley & Sons, Inc., Hoboken, A.B.D., 2003.
- [14] Singh, N. “Design of cellular manufacturing systems: An invited review”, *European Journal of Operational Research*, 69, 284-291, 1993.
- [15] Mansouri, S. A., Hsseini, S. M. M. ve Newman, S. T., “A review of the modern approaches to multi-criteria cell design” *International Journal of Production Research*, 38, 1201-1218, 2000.
- [16] Groover, M. P., *Automation, production systems and computer-integrated manufacturing*, Prentice Hall, Inc., Upper Saddle River, A.B.D., 2001.
- [17] Wemmerl w, U., ve Hyer, N., “Cellular manufacturing in the U.S. industry: a survey of users”, *International Journal of Production Research*, 27, 1511-1530, 1989.
- [18] Curry, G. L., ve Feldman, R. M., *Manufacturing systems modeling and analysis 2nd edition*, Springer, Heidelberg, 2011.
- [19] Wemmerl w, U. ve Johnson, D. J., “Cellular manufacturing at 46 user plants: implementation experiences and performance improvements”, *International Journal of Production Research*, 35, 29-49, 1997.
- [20] Durmu ođlu, M. B., Kulak, O. ve Balcı, H. H., “T rkiye'de h cresel  retim uygulamalarının analizi ve deđerlendirilmesi”, *End stri M hendisliđi*, 2, 2003.
- [21] Solimanpur, M., Saeedi, S. ve Mahdavi, I., “Solving cell formation problem in cellular manufacturing using ant-colony-based optimization” *The International Journal of Advanced Manufacturing Technology*, 50, 1135-1144, 2010.
- [22] Sarker, B. R. ve Xu, Y., “Operation sequences-based cell formation methods: A critical survey” , *Production Planning & Control*, 9(8), 1998.
- [23] Papaioannou, G. ve Wilson, J. M., “The evolution of cell formation problem methodologies based on recent studies (1997–2008): Review and directions for future research”, *European Journal of Operational Research*, 206, 509–521, 2010.

- [24] Nagi, R., Harhalakis, G., ve Proth, J. M., “Multiple routings and capacity considerations in group technology applications”, *International Journal Of Production Research*, 28(12), 2243-2257, 1990.
- [25] Saraç, T. ve Özçelik, F., “Alternatif rotaların varlığında üretim hücrelerinin genetik algoritma kullanılarak oluşturulması,” *MMO Endüstri Mühendisliği Dergisi*, 17, 22-36, 2009.
- [26] Adil, G. K., Rajamani, D. ve Strong, D., “Cell formation considering alternate routings,” *International Journal of Production Research*, 34, 1361-1380, 1996.
- [27] Lee, M. K., Luong, H. S. ve Abhary, K., “A genetic algorithm based cell design considering alternative routing”, *Computer Integrated Manufacturing Systems*, 10(2), 93-107, 1997.
- [28] Lozano, S., Guerrero, F., Eguia, I. ve Omeva, L., “Cell design and loading in the presence of alternative routing”, *International Journal of Production Research*, 37(14), 3289-3304, 1999.
- [29] Caux, C., Bruniaux, R. ve Pierreval, H., “Cell formation with alternative process plans and machine capacity constraints: A new combined approach” *International Journal of Production Economics*, 64, 279-284, 2000.
- [30] Spiliopoulos, K. ve Sofianopoulou, S., “Manufacturing cell design with alternative routings in generalized group”, *International Journal of Production Research*, 45(6), 1355-1367, 2007.
- [31] Sormaz, D. N. ve Rajaraman, S. N., “Problem space search algorithm for manufacturing cell formation with alternative process plans”, *International Journal of Production Research*, 46(2), 345-369, 2008.
- [32] Susanto, S., Al-Dabass, D. ve Bhattacharya, A., “Optimised cell formation algorithm considering sequence of operations, alternative routing and part-volume”, *Third Asia International Conference on Modelling & Simulation*, 2009.
- [33] Wu, T., Chung, S., ve Chang, C., “Hybrid simulated annealing algorithm with mutation operator to the cell formation problem with alternative process routings”.. s.l. : Expert Systems with Applications, 2009, Cilt 36, s. 3652–3661.

- [34] Chung, S., Wu, T. ve Chang, C., “An efficient tabu search algorithm to the cell formation problem with alternative routings and machine reliability considerations”, s.l. : Computers & Industrial Engineering, 2011, Cilt 60, s. 7-15.
- [35] Offodile, O. F., Mehrez, A. ve Grznar, J., “Cellular manufacturing: a taxonomic review framework,” *Journal of Manufacturing*, 13, 196-220, 1994.
- [36] Liu, C., Yong Y., Kazuhiko, Y. ve Jie, L., “A heuristic algorithm for cell formation problems with consideration of multiple production factors”, *International Journal of Advanced Manufacturing Technology*, 46, 1201-1213, 2010.
- [37] İşlier, A. A., “Üretim hücrelerinin bir genetik algoritma kullanılarak oluşturulması,” *Anadolu Üniversitesi Bilim ve Teknoloji Dergisi*, 2, 137-157, 2001.
- [38] James, T. L., Brown, E. C. ve Keeling, K., “A hybrid grouping genetic algorithm for the cell formation problem”, *Computers & Operations Research*, 34, 2059–2079, 2007.
- [39] Yang, X., “Introduction to Mathematical Optimization”, *International Science Publishing*, Cambridge, UK, 2008.
- [40] Tavakkoli M.R., Rahimi, V. A. R., Ghodrathnama, A. ve Siadat A., “A simulated annealing method for solving a new mathematical model of a multi-criteria cell formation problem with capital constraints”, *Advances in Engineering Software*, 40, 268-273, 2009.
- [41] Wu, T., Yeh, J., Chang, C., "A hybrid tabu search algorithm to cell formation problem and its variants", *World Academy of Science, Engineering and Technology* 53, 2009.
- [42] İşlier, A. A. “Group technology by an ant system algorithm”, *International Journal of Production Research*, 43, 913-932, 2005.
- [43] Silahtaroglu, G., *Kavram ve algortimaları ile veri madenciliği*, Papatya Yayıncılık Eğitim, İstanbul, 2008.
- [44] Kusiak, A., *Computational intelligence in design and manufacturing*, John Wiley & Sons, New York, United States, 2000.

- [45] Kantardzic, M., *Data mining, concepts, models, methods and algorithms*, IEEE Pres, Piscataway, A.B.D., 2003.
- [46] Adenso-Díaz, B., Lozano, S., Racerob, J. ve Guerrerob, F “Machine cell formation in generalized group technology”, *Computers & Industrial Engineering*, 41,227-240, 2001.
- [47] Andres, C. ve Lozano, S., “A particle swarm optimization algorithm for part-machine grouping”, *Robotics and Computer-Integrated Manufacturing*, 22, 468–474, 2006.
- [48] Rezazadeh, H., Ghazanfari, M., Sadjadi, S. J., Aryanezhad, M. B. ve Makui, A., “Linear programming embedded particle swarm optimization for solving an extended model of dynamic virtual cellular manufacturing systems”, *Journal of Applied Research and Technology*, 7, 83-108, 2009.
- [49] Anvari, M., Mehrabad, M. S. ve Barzinpour, F., “Machine-part cell formation using a hybrid particle swarm optimization, *International Journal of Advanced Manufacturing Technology*, 47, 745-754, 2010.
- [50] Duran, O., Rodriguez, N. ve Consalter, L. A., “Collaborative particle swarm optimization with a data mining technique for manufacturing cell design”, *Expert Systems with Applications*, 37, 1563-1567, 2010.
- [51] Chandrasekharan, M.P. ve Rajagopalan, R., “An ideal seed non-hierarchical clustering algorithm for cellular manufacturing”, *International Journal of Production Research*, 24, 451-464, 1986.
- [52] Kumar, C.S. ve Chandrasekharan, M.P., “Grouping efficacy: A quantitative criterion for goodness of block diagonal forms of binary matrices in group technology, *International Journal of Production Research*, 28, 233-243, 1990.
- [53] Öztürk, G., Öztürk, Z. K., “A competitive neural network approach to manufacturing cell. Formation”, *35th International Conference on Computers and Industrial Engineering* , Istanbul, 1549-1554, 2005.
- [54] Kennedy, J. ve Eberhart, R., “Partice swarm optimization, *IEEE Int'l. Conf. on Neural Networks*, 4, Perth, Australia, 1942-1948, 1995.
- [55] Blondin, J., *Particle swarm optimization: A Tutorial*, 2009.



- [56] Kusiak, A., “The generalized group technology concept”, *International Journal Of Production Research*, 25(4), 561-569, 1987.
- [57] Won, Y.K. ve Kim, S.H., “Multiple criteria clustering algorithm for solving the group technology problem with multiple process routings”., *Computers & Industrial Engineering*, 32, 207-220, 1997..
- [58] Yin, Y. ve Yasuda, K., “Manufacturing cells’ design in consideration of various production factors”, *International Journal of Production Research*, 40(4), 885-905, 2002.
- [59] Sofianopoulou, S. “Manufacturing cells design with alternative process plans and/or replicate machines”, *International Journal of Production Research*, 37(3), 707–720, 1999.
- [60] Kasilingam, R. G.; Lashkari, R. S. “Cell formation in the presence of alternate process plans in flexible manufacturing systems”, *Production Planning and Control*, 2, 135-141, 1991.
- [61] Kennedy J. ve Eberhart, R. C., “A discrete binary version of the particle swarm algorithm”, *IEEE International Conference on Systems, Man, and Cybernetics*, 4104-4108, 1997.

## EK-1 Hücre Oluşturma Yöntemlerinin Karşılaştırılması

Kaynaklar	Amaçlar			Kısıtlar					Çözüm yaklaşımları										Karşılaştırma Sonuçları		
	O 1	O 2	O 3	C 1	C 2	C 3	C 4	C 5	M P	H	T	S	A	G	P	S S	F	H	O	D	C
Zolfaghari ve Liang (1997)	+																	+			H
Vakharia ve Chang (1997)		+	+						+		+	+								+	H
Aljaber ve ark.(1997)	+									+	+										H
Masnata ve Settineri (1997)																	+	+			E
Selim ve ark.(1998)	+	+	+	+		+			+												H
Gravel ve ark.(1998)	+						+						+								H
Lozano ve ark.(1999)	+										+									+	E
Sofianopoulou (1999)	+				+		+		+		+										H
Liang ve Zolfaghari (1999)	+				+	+												+			E
Susanto ve ark.(1999)	+								+								+				E
Mak ve ark.(2000)	+												+						+		E
Ravichveran ve Rao (2001)	+								+								+				E
Lozano ve ark.(2001)	+				+				+									+		+	E
Mukattash ve ark.(2002)	+				+	+	+			+											E
Guerrero ve ark.(2002)									+									+			E
Soleymanpour ve ark.(2002)	+								+									+			E
Lozano ve ark.(2002)	+								+								+				E
Logendran ve Karim (2003)		+				+	+		+		+									+	H
Spiliopoulos ve Sofianopoulou (2003)	+								+		+								+		H
Park ve Suresh (2003)	+				+												+	+			E
Won ve Lee (2004)	+								+										+		H
Kim ve ark.(2004)	+		+				+			+											H
Wu ve ark.(2004)	+								+		+								+	+	H
Solimanpur ve ark.(2004a)	+	+							+				+								H
Solimanpur ve ark.(2004b)	+								+								+		+		H
Tavakkoli-Moghaddam et	+	+						+	+		+	+		+						+	H



- II. Farklı modellere göre ele alınan kısıtlar  
C1:Parça/Makine işlem sıralarının dikkate alınması  
C2:Aynı tip makineden birden fazla olması  
C3:Parça/makine kullanım oranının ya da işlem süresinin dikkate alınması  
C4:Alternatif rotaların dikkate alınması  
C5:Çok dönemli zaman dilimi dikkate alınması
- III. Çözüm yaklaşımları  
MP:Matematiksel programlama, H:Sezgisel, T:Yasaklı arama, S:Tavlama Benzetimi, A:Karınca kolonisi algoritması, G:Genetik algoritma, P:Parçacık sürü optimizasyonu, SS:Dağınık arama, F:Bulanık mantık teorisi, N:Yapay sinir ağları.
- IV. Karşılaştırma sonuçları  
O:Büyük veri kümeleri için optimum değer elde edilmiştir.  
D:Optimum değerden olan yüzdelik sapma  
C:Var olan diğer yöntemlerle karşılaştırma yapılıp yapılmaması (E:Evet, H:Hayır)

## EK-2 Peaks Fonksiyonunda Enküçük Minimum için Yeni Kesikli Pso Adımları

tekrar no	parçacık no		$x_1$	$x_2$	$f(x_1, x_2)$	fpbest	fgbest
1	1	X	0.0857	0.4236			
		kX	-4	0			
		V	-0.835	0.5503			
		pbestX	0.0857	0.4236			
		pbestkX	-4	0			
		gbestX	0.0857	0.4236			
		gbestkX	-4	0	-0.000109	-0.000109	
	2	X	0.7895	0.9245			
		kX	4	6			
		V	0.3103	0.9506			
		pbestX	0.7895	0.9245			
		pbestkX	4	6			
		gbestX	0.0857	0.4236			
		gbestkX	-4	0	2.05E-18	2.046E-18	
	3	X	0.0325	0.4014			
		kX	-5	-1			
		V	-0.991	-0.132			
		pbestX	0.0325	0.4014			
		pbestkX	-5	-1			
		gbestX	0.0857	0.4236			
		gbestkX	-4	0	-1.87E-08	-1.87E-08	
	4	X	0.2014	0.8194			
		kX	-3	4			
		V	-0.978	-0.206			
pbestX		0.2014	0.8194				
pbestkX		-3	4				
gbestX		0.0857	0.4236				
gbestkX		-4	0	1.38E-07	1.379E-07		
5	X	0.5974	0.9384				
	kX	2	6				
	V	-0.508	-0.861				
	pbestX	0.5974	0.9384				
	pbestkX	2	6				
	gbestX	0.0857	0.4236				
	gbestkX	-4	0	3.31E-13	3.307E-13	<b>-0.0001</b>	
2	6	X	0.0496	0.4261			
		kX	-5	0			
		V	-0.835	0.5503			
		pbestX	0.0857	0.4236			
		pbestkX	-4	0			
		gbestX	0.4267	0.2705			
		gbestkX	0	-2	-5.42E-08	-0.000109	
	7	X	0.736	0.9514			
		kX	3	6			
		V	-0.846	0.5339			
		pbestX	0.7895	0.9245			
		pbestkX	4	6			

3		gbestX	0.4267	0.2705			
		gbestkX	0	-2	2.23E-15	2.046E-18	
	8	X	0.0171	0.4013			
		kX	-5	-1			
		V	-0.995	-0.097			
		pbestX	0.0325	0.4014			
		pbestkX	-5	-1			
		gbestX	0.4267	0.2705			
		gbestkX	0	-2	-1.87E-08	-1.87E-08	
	9	X	0.1335	0.7611			
		kX	-4	4			
		V	-0.828	-0.561			
		pbestX	0.1335	0.7611			
		pbestkX	-4	4			
		gbestX	0.4267	0.2705			
		gbestkX	0	-2	1.17E-10	1.17E-10	
	10	X	0.4267	0.2705			
		kX	0	-2			
		V	-0.563	-0.827			
		pbestX	0.4267	0.2705			
		pbestkX	0	-2			
		gbestX	0.4267	0.2705			
		gbestkX	0	-2	-4.759612	-4.759612	<b>-4.7596</b>
	11	X	0.0257	0.4297			
		kX	-5	0			
		V	-0.306	0.9522			
		pbestX	0.0857	0.4236			
		pbestkX	-4	0			
		gbestX	0.4267	0.2705			
		gbestkX	0	-2	-5.42E-08	-0.000109	
12	X	0.0701	0.8741				
	kX	-5	5				
	V	-0.865	-0.501				
	pbestX	0.7895	0.9245				
	pbestkX	4	6				
	gbestX	0.4267	0.2705				
	gbestkX	0	-2	5.27E-18	2.046E-18		
13	X	0.0124	0.3986				
	kX	-5	-1				
	V	-0.72	-0.694				
	pbestX	0.0325	0.4014				
	pbestkX	-5	-1				
	gbestX	0.4267	0.2705				
	gbestkX	0	-2	-1.87E-08	-1.87E-08		
14	X	0.1095	0.698				
	kX	-4	3				
	V	-0.743	-0.669				
	pbestX	0.1335	0.7611				
	pbestkX	-4	4				
	gbestX	0.4267	0.2705				
	gbestkX	0	-2	1.99E-08	1.17E-10		
15	X	0.2449	0.2566				

		kX	-3	-2				
		V	-0.563	-0.827				
		pbestX	0.4267	0.2705				
		pbestkX	0	-2				
		gbestX	0.4267	0.2705				
		gbestkX	0	-2	0.000747	-4.759612	<b>-4.7596</b>	
4	16	X	0.2671	0.5485				
		kX	-2	1				
		V	0.4461	0.895				
		pbestX	0.2671	0.5485				
		pbestkX	-2	1				
		gbestX	0.4267	0.2705				
	gbestkX	0	-2	-0.480759	-0.480759			
	17	X	0.1901	0.7515				
		kX	-3	4				
		V	0.3541	-0.935				
		pbestX	0.7895	0.9245				
		pbestkX	4	6				
		gbestX	0.4267	0.2705				
	gbestkX	0	-2	1.38E-07	2.046E-18			
	18	X	0.009	0.3821				
		kX	-5	-1				
		V	-0.565	-0.825				
		pbestX	0.0325	0.4014				
		pbestkX	-5	-1				
		gbestX	0.4267	0.2705				
	gbestkX	0	-2	-1.87E-08	-1.87E-08			
	19	X	0.0473	0.5101				
		kX	-5	1				
		V	-0.433	-0.902				
pbestX		0.0473	0.5101					
pbestkX		-5	1					
gbestX		0.4267	0.2705					
gbestkX	0	-2	-2.01E-08	-2.01E-08				
20	X	0.1702	0.2014					
	kX	-3	-3					
	V	-0.453	-0.891					
	pbestX	0.4267	0.2705					
	pbestkX	0	-2					
	gbestX	0.4267	0.2705					
gbestkX	0	-2	6.67E-05	-4.759612	<b>-4.7596</b>			
5	21	X	0.6372	0.8859				
		kX	2	5				
		V	0.6871	0.7266				
		pbestX	0.2671	0.5485				
		pbestkX	-2	1				
		gbestX	0.4267	0.2705				
	gbestkX	0	-2	7.97E-09	-0.480759			
	22	X	0.873	0.2653				
		kX	5	-2				
		V	0.7935	-0.609				
pbestX		0.7895	0.9245					

	pbestkX	4	6			
	gbestX	0.4267	0.2705			
	gbestkX	0	-2	4.79E-10	2.046E-18	
23	X	0.0003	0.3764			
	kX	-5	-1			
	V	-0.459	-0.889			
	pbestX	0.0325	0.4014			
	pbestkX	-5	-1			
	gbestX	0.4267	0.2705			
	gbestkX	0	-2	-1.87E-08	-1.87E-08	
24	X	0.0149	0.3878			
	kX	-5	-1			
	V	-0.285	-0.958			
	pbestX	0.0473	0.5101			
	pbestkX	-5	1			
	gbestX	0.4267	0.2705			
	gbestkX	0	-2	-1.87E-08	-2.01E-08	
25	X	0.1839	0.177			
	kX	-3	-3			
	V	0.093	-0.996			
	pbestX	0.4267	0.2705			
	pbestkX	0	-2			
	gbestX	0.4267	0.2705			
	gbestkX	0	-2	6.67E-05	-4.759612	<b>-4.7596</b>

X: parçacığın o andaki konumu ([0-1] aralığında)

kX: parçacığın o andaki konumu (kesikli değer)

V: parçacığın o andaki hız vektörü

Pbestx: parçacığın o ana kadar kendi eniyi konumu ([0-1] aralığında)

Pbestkx: parçacığın o ana kadar kendi eniyi konumu (kesikli değer)

Gbestx: tüm parçacıklar içinde o ana kadar elde edilen eniyi konum ([0-1] aralığında)

Gbestkx: tüm parçacıklar içinde o ana kadar elde edilen eniyi konum (kesikli değer)

$f(x_1, x_2)$ : uygunluk fonksiyonu değeri

fpbest: parçacığın o ana kadar kendi eniyi uygunluk değeri

fgbest: tüm parçacıklar içinden o ana kadar elde edilen eniyi uygunluk değeri



### EK-3 Alternatif Rotalı Hücre Oluşturma Problemleri ve Çözümleri

Parça No	Alternatif rota	Makine No			
		1	2	3	4
1	1			1	1
1	2		1		1
1	3	1	1		
2	1		1	1	
2	2	1		1	
3	1	1			1
3	2		1		1
4	1	1			1
4	2	1		1	
5	1			1	1
5	2	1			

Şekil P.1 Problem P1

	M2	M4	M1	M3
P1(2)	1	1		
P3(2)	1	1		
P2(2)			1	1
P4(2)			1	1
P5(2)			1	0

Şekil P.2 Çözüm P1

Parça No	Alt. Rota	1	2	3	4
1	1			1	1
1	2		1		1
1	3	1	1		
2	1		1	1	
2	2	1		1	
3	1	1			1
3	2		1		1
4	1	1			1
4	2	1		1	
5	1			1	1
5	2	1			
5	3	1	1	1	

Şekil P.3 Problem P2

	M1	M3	M2	M4
P2(2)	1	1		
P4(2)	1	1		
P5(3)	1	1	1	
P1(2)			1	1
P3(2)			1	1

Şekil P.4 Çözüm P2

Parça No	Alt. Rota	1	2	3	4
1	1		1	1	
1	2		1		1
2	1	1		1	
2	2			1	1
3	1	1	1		
3	2	1		1	
4	1		1		1
4	2	1			1

Şekil P.5 Problem P3

	M1	M3	M2	M4
P2(1)	1	1		
P3(2)	1	1		
P1(2)			1	1
P4(1)			1	1

Şekil P.6 Çözüm P3

Parça No	Alt. Rota	1	2	3	4	5
1	1	1			1	
1	2	1		1	1	
2	1	1			1	
3	1		1			1
3	2		1	1		1
4	1	1	1			1
4	2		1	1		1
5	1	1			1	
5	2	1		1	1	
6	1		1			1
7	1		1			1

Şekil P.7 Problem P4

	M3	M2	M5	M1	M4
P3(1)		1	1		
P4(1)		1	1	1	
P6		1	1		
P7		1	1		
P1(1)				1	1
P2				1	1
P5(1)				1	1

Şekil P.8 Çözüm P4

Parça No	Alter. Rota	1	2	3	4	5	6
1	1	1	1		1		
1	2		1	1		1	1
1	3		1	1		1	1
2	1			1		1	1
3	1			1		1	1
4	1	1			1		
4	2	1	1		1		
5	1		1	1		1	1
5	2			1			1
6	1	1	1	1			
6	2	1	1				1
7	1		1			1	1
7	2			1		1	1
7	3		1				1
8	1		1		1		

Şekil P.9 Problem P5

	M1	M2	M4	M3	M5	M6
P1(1)	1	1	1			
P4(2)	1	1	1			
P6(1)	1	1	0	1		
P8	0	1	1			
P2				1	1	1
P3				1	1	1
P5(1)		1		1	1	1
P7(2)				1	1	1

Şekil P.10 Çözüm P5

Parça No	Alt. Rota	1	2	3	4	5	6
1	1		1		1	1	
1	2		1	1		1	
1	3		1				1
2	1	1			1		1
2	2			1			1
3	1	1		1			1
3	2				1		1
4	1		1		1	1	
4	2			1		1	
5	1			1			1
5	2			1		1	
6	1						1
6	2				1	1	

Şekil P.11 Problem P6

	M1	M3	M6	M2	M4	M5
P2(2)	0	1	1			
P3(1)	1	1	1			
P5(1)	0	1	1			
P1(1)				1	1	1
P4(1)				1	1	1
P6(2)				0	1	1

Şekil P.12 Çözüm P6

Parça No	Alt. Rota	1	2	3	4	5	6
1	1		1			1	1
1	2		1	1			1
2	1		1		1	1	1
2	2		1		1		1
3	1	1			1		
4	1		1		1	1	
4	2		1		1		1
5	1	1		1	1		
5	2	1		1		1	
5	3			1	1		
5	4			1	1	1	
6	1		1		1		
6	2				1		1
7	1	1		1			
8	1		1		1	1	1
8	2	1	1		1		1
9	1		1		1		1
9	2		1			1	1
10	1			1	1		
10	2				1	1	

Şekil P.13 Problem P7

	M2	M4	M5	M6	M1	M3
P1(1)	1	0	1	1		
P2(1)	1	1	1	1		
P4(1)	1	1	1	0		
P6(1)	1	1	0	0		
P8(1)	1	1	1	1		
P9(1)	1	1	0	1		
P10(2)	0	1	1	0		
P3		1			1	0
P5(1)		1			1	1
P7					1	1

Şekil P.14 Çözüm P7

Parça No	Alt. Rota	1	2	3	4	5	6	7
1	1		1		1			
1	2	1	1					
2	1	1				1		
2	2	1		1				
3	1					1	1	
3	2					1		
4	1			1		1	1	
4	2			1			1	1
5	1	1			1		1	
5	2		1	1		1		
5	3	1	1	1				
5	4			1			1	1
6	1				1	1		
6	2		1		1			
7	1	1		1		1		
7	2	1	1					1
8	1		1	1				1
8	2					1	1	
8	3			1	1		1	1
9	1			1			1	
9	2			1	1			
10	1	1						1
10	2		1		1	1		

Şekil P.15 Problem P8

	M3	M6	M7	M1	M5	M2	M4
P4(2)	1	1	1				
P5(4)	1	1	1				
P8(3)	1	1	1				1
P9(1)	1	1	0				
P2(1)				1	1		
P3(2)				0	1		
P7(1)	1			1	1		
P1(1)						1	1
P6(2)						1	1
P10(2)					1	1	1

Şekil P.16 Çözüm P8

Parça No	Alt. Rota	1	2	3	4	5	6	7
1	1		1				1	1
1	2	1			1			
2	1	1				1		
2	2		1	1			1	
2	3				1			1
3	1				1			
3	2		1				1	
4	1		1					1
4	2	1			1		1	
5	1	1	1					1
5	2			1		1		
5	3		1		1			1
6	1			1			1	
6	2	1	1			1		
7	1		1		1			
7	2			1		1		1
8	1	1				1	1	
8	2			1	1			1
9	1	1			1			1
9	2			1	1			
9	3	1				1	1	
10	1			1				1
10	2	1			1		1	
11	1			1	1			
11	2		1			1		1
11	3	1	1				1	
12	1			1	1			
12	2		1			1		1
12	3	1	1				1	
13	1			1				
14	1	1				1		
14	2			1		1		

Şekil P.17 Problem P9

	M2	M6	M7	M3	M4	M1	M5
P1(1)	1	1	1				
P3(2)	1	1	0				
P4(1)	1	0	1				
P5(1)	1	0	1			1	
P7(1)	1			0	1		
P8(2)			1	1	1		
P9(2)				1	1		
P10(1)			1	1	0		
P11(1)				1	1		
P12(1)				1	1		
P13				1	0		
P2(1)						1	1
P6(2)	1					1	1
P14(1)						1	1

Şekil P.18 Çözüm P9

Parça No	Alt. Rota	1	2	3	4	5	6	7	8
1	1			1		1			1
1	2			1			1		
2	1			1		1			
2	2	1	1			1			
3	1	1					1	1	
3	2	1	1			1			
4	1			1		1			1
4	2			1			1		
5	1	1	1					1	
5	2		1				1		
6	1		1		1				
6	2	1			1	1			
7	1							1	1
7	2					1			1
8	1	1			1				
8	2				1		1		
9	1			1		1			1
9	2	1	1			1			
10	1	1	1	1					
10	2		1		1		1		
11	1			1	1				1
11	2	1						1	
12	1	1	1			1			
12	2	1						1	
13	1		1		1		1		
13	2	1						1	

Şekil P.19 Problem P10

	M2	M4	M6	M3	M5	M8	M1	M7
P6(1)	1	1	0					
P8(2)	0	1	1					
P10(2)	1	1	1					
P13(1)	1	1	1					
P1(1)				1	1	1		
P2(1)				1	1	0		
P4(1)				1	1	1		
P7(2)				0	1	1		
P9(1)				1	1	1		
P3(1)			1				1	1
P5(1)	1						1	1
P11(2)							1	1
P12(2)							1	1

Şekil P.20 Çözüm P10

Parça No	Alt. Rota	1	2	3	4	5	6	7	8	9	10
1	1	1	1		1						
1	2				1	1					
1	3			1	1						1
2	1							1	1		1
2	2	1				1				1	1
3	1		1					1	1		
3	2	1				1		1			
3	3	1	1								1
4	1			1				1		1	
4	2			1				1	1		
5	1	1				1					1
5	2		1			1				1	1
6	1	1	1	1							
6	2			1				1	1		
6	3					1	1	1			
7	1		1		1	1					
7	2	1	1		1						
7	3				1			1	1		
8	1	1				1		1		1	1
8	2	1								1	1
9	1			1				1	1		1
9	2			1				1		1	
10	1	1		1					1		
10	2			1				1	1		

Şekil P.21 Problem P11

	M2	M4	M6	M1	M5	M9	M10	M3	M7	M8
P1(2)	0	1	1							
P7(1)	1	1	1							
P2(2)				1	1	1	1			
P5(1)				1	1	0	1			
P8(1)				1	1	1	1		1	
P3(1)	1							0	1	1
P4(2)								1	1	1
P6(2)								1	1	1
P9(1)							1	1	1	1
P10(2)								1	1	1

Şekil P.22 Çözüm P11



Parça No	Alt. Rota	1	2	3	4	5	6	7	8	9	10	11
1	1					1			1			
1	2		1						1		1	
2	1	1		1		1						
2	2	1	1		1							
3	1			1				1			1	
3	2	1		1		1						
4	1				1	1		1				
4	2				1		1	1				
5	1			1					1	1		
5	2	1				1					1	
6	1	1	1				1					
6	2			1							1	1
6	3		1					1	1			
7	1		1						1		1	
7	2				1	1		1				
8	1									1	1	
8	2								1			1
9	1				1	1						
9	2	1				1						
10	1		1									1
10	2							1		1		
10	3					1				1		1

Şekil P.23 Problem P12

	M1	M3	M5	M9	M11	M4	M6	M7	M2	M8	M10
P2(1)	1	1	1								
P3(2)	1	1	1								
P5(2)	1	0	1								1
P8(2)				1	1						
P10(3)			1	1	1						
P4(2)						1	1	1			
P9(1)						1	1	0			
P1(2)									1	1	1
P6(3)								1	1	1	0
P7(1)									1	1	1

Şekil P.24 Çözüm P12

Parça No	Alt. Rota	1	2	3	4	5	6	7	8	9	10	11	12
1	1			1		1	1		1			1	1
2	1					1	1	1			1	1	
2	2		1			1	1			1			
3	1	1	1		1	1					1	1	
4	1	1		1	1		1				1		
5	1		1				1						1
5	2			1				1		1			
6	1		1			1	1		1				
7	1								1				1
8	1		1		1					1			
9	1	1	1	1				1				1	1
10	1		1		1			1		1			
11	1		1	1		1			1			1	1
12	1					1			1		1	1	
12	2			1				1		1	1		
13	1					1	1				1	1	
14	1			1	1			1			1		
14	2	1				1	1		1				
15	1		1		1	1							
16	1		1	1			1	1				1	
17	1		1	1			1					1	
17	2					1			1	1	1		
17	3			1	1			1	1				
18	1				1	1			1				
19	1		1	1		1				1	1		
20	1	1				1	1						

Şekil P.25 Problem P13

	M1	M6	M2	M4	M5	M7	M9	M10	M3	M8	M11	M12
P4	1	1		1				1	1			
P6	0	1	1		1					1		
P20	1	1	1		1							
P3	1		1	1	1			1			1	
P8			1	1	0		1					
P10			1	1	0	1	1					
P15			1	1	1							
P18			0	1	1					1		
P2(1)		1			1	1	0	1			1	
P5(2)						1	1	0	1			
P12(2)						1	1	1	1			
P13					1	1	0	1			1	
P14(2)	1				1	1	1	0				
P19			1		1	0	1	1	1			
P1		1			1				1	1	1	1
P7									0	1	0	1
P9	1		1			1			1	0	1	1
P11			1		1				1	1	1	1
P16		1	1			1			1	0	1	0
P17(1)		1	1						1	0	1	0

Şekil P.26 Çözüm P13

Parça No	Alt. Rota	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1					1					1	1			1	
2	1	1		1	1	1	1		1	1				1	1	
2	2	1		1	1	1	1		1			1		1	1	
3	1											1	1			1
3	2								1				1			1
4	1		1	1	1		1					1	1		1	
4	2		1	1	1		1		1				1		1	
5	1		1	1	1	1			1	1	1	1			1	
5	2		1	1	1	1			1	1	1	1			1	
6	1			1	1				1		1	1				
7	1		1			1	1	1		1	1		1	1		
7	2		1			1	1	1		1	1	1	1	1		
8	1			1	1		1			1	1	1		1	1	
8	2			1	1		1			1	1	1		1	1	
9	1			1	1	1	1		1		1	1	1		1	
9	2			1	1	1	1		1	1	1		1		1	
10	1		1		1			1		1	1			1		
10	2		1		1			1		1	1			1		
11	1			1	1		1	1						1	1	1
12	1				1		1	1	1		1	1	1		1	1
12	2				1		1	1	1	1	1		1		1	1
13	1		1	1			1	1	1	1	1	1			1	1
13	2		1	1			1	1	1	1	1	1			1	1
14	1		1		1		1	1		1	1					
14	2		1		1		1	1		1	1					
15	1							1		1				1		
15	2							1				1		1		

Şekil P.27 Problem P14

	M1	M7	M9	M12	M13	M15	M2	M3	M4	M5	M6	M8	M10	M11	M14
P3(2)	0	0	1	1	0	1									
P10(1)	0	1	1	0	1	0	1		1				1		
P11	0	1	0	0	1	1		1	1		1				1
P12(2)	0	1	1	1	0	1			1		1	1	1		1
P15(1)	0	1	1	0	1	0									
P1							0	0	0	1	0	0	1	1	1
P2(2)	1				1		0	1	1	1	1	1	0	1	1
P4(1)				1			1	1	1	0	1	0	0	1	1
P5(1)			1				1	1	1	1	0	1	1	1	1
P6							0	1	1	0	0	1	1	1	0
P7(2)		1		1	1		1	0	0	1	1	0	1	1	0
P8(1)			1		1		0	1	1	0	1	0	1	1	1
P9(1)				1			0	1	1	1	1	1	1	1	1
P13(1)		1	1			1	1	1	0	0	1	1	1	1	1
P14(2)		1					1	0	1	0	1	0	1	1	0

Şekil P.28 Çözüm P14

Parça No	Alt. Rota	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
1	1		1						1			1				1					1			1			
1	2			1			1							1								1		1			1
1	3				1									1	1							1	1				1
2	1					1							1				1										
2	2			1		1		1					1														
3	1												1							1					1		
3	2																		1		1		1		1		
3	3														1				1				1				
4	1	1			1															1		1	1				
4	2	1						1													1				1	1	
5	1			1					1		1								1								
5	2							1									1	1									
6	1				1					1		1															
6	2									1										1							
7	1		1				1		1							1							1			1	
7	2	1	1							1			1									1		1			
7	3		1		1		1							1	1	1								1			1
8	1																1				1				1		
8	2					1																			1	1	
8	3	1	1																1		1				1		
9	1		1				1	1						1					1						1		1
9	2		1											1		1						1	1		1		1
9	3					1					1					1			1	1		1		1			
10	1	1	1					1			1										1			1			
10	2	1		1			1			1					1												
10	3							1						1					1	1						1	
10	4	1													1	1					1					1	
11	1					1					1																1
11	2					1							1				1										
12	1			1					1		1									1							
13	1			1						1		1								1							
13	2			1					1												1						
14	1	1	1				1																		1	1	
14	2		1											1		1						1		1			
15	1							1	1															1		1	
15	2	1																	1		1	1					

15	3	1					1			1			1										1	
16	1	1						1						1										
16	2			1				1								1								
17	1												1	1							1			
17	2									1						1							1	
17	3				1			1		1														
18	1											1			1					1				
18	2					1					1				1							1		
18	3					1					1				1									
19	1	1		1								1		1										1
19	2		1				1	1							1				1	1				
19	3									1		1			1				1					1
19	4		1				1					1							1	1				
20	1												1					1				1		1
20	2							1									1	1					1	
20	3	1		1				1									1						1	
21	1								1								1					1		
21	2								1	1							1							1
21	3							1			1						1							
22	1	1						1					1	1				1						
22	2					1				1													1	
23	1				1								1											
23	2								1	1					1									
24	1											1		1			1	1			1			
24	2		1				1				1							1		1			1	
25	1							1					1					1					1	
25	2	1						1			1					1	1							
26	1	1			1		1												1					
26	2															1	1							1
26	3								1							1		1				1		
27	1		1											1					1		1		1	1
27	2	1					1			1			1								1			
27	3			1				1			1													1
28	1															1			1		1			
28	2																1				1		1	1

Şekil P.29 Problem P15

	M4	M9	M11	M5	M12	M16	M3	M8	M10	M17	M1	M7	M14	M19	M25	M18	M20	M22	M24	M2	M6	M13	M15	M21	M23	M26	
P6(1)	1	1	1																								
P17(3)	1	1	1																								
P23(2)	0	1	1			1																					
P2(1)				1	1	1																					
P11(2)				1	1	1																					
P18(3)				1	1	1																					
P5(1)							1	1	1	1																	
P12							1	1	1	1																	
P13(1)							1	0	1	1																	
P16(2)							1	1	0	1																	
P21(2)							0	1	1	1																1	
P4(2)											1	1	0	1	1					1							
P10(4)											1	0	1	1	1									1			
P15(3)			1								1	1	1	0	1												
P20(3)							1				1	1	0	1	1												
P22(1)											1	1	1	1	0								1				
P25(1)											0	1	1	1	1												
P3(2)																1	1	1	1								
P8(3)											1					1	1	0	1	1							
P26(3)		1														1	1	0	1								
P28(1)																1	0	1	1								
P1(2)							1														0	1	1	0	1	1	1
P7(3)	1												1								1	1	1	1	0	1	1
P9(2)																	1				1	0	1	1	1	1	1
P14(2)																					1	0	1	1	1	1	0
P19(4)																					1	1	1	0	1	1	0
P24(2)						1															1	1	0	0	1	1	1
P27(1)															1						1	0	0	1	1	1	1

Şekil P.30 Çözüm P15

Parça No	Alt. Rota	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	1								1	1						1	1			1	1										
1	2		1						1															1							
2	1	1																					1	1							
2	2										1		1	1				1													
3	1							1					1	1			1														
3	2												1	1				1													
3	3									1				1				1							1						
4	1	1			1		1	1				1																			
4	2	1	1																				1	1							
4	3							1				1												1							
5	1	1		1	1	1																			1						
5	2					1	1					1																			
5	3							1																	1						
6	1																							1							
6	2												1		1							1	1								
7	1			1	1	1																			1						
8	1		1					1	1																1						
8	2							1																	1						
8	3		1			1				1		1													1						
9	1								1		1																				
9	2						1	1																	1						
9	3	1			1																				1						
10	1												1	1				1													
10	2													1				1		1											
10	3								1	1			1																		
11	1																			1	1			1							
11	2									1							1	1													
11	3					1							1	1				1													
12	1												1	1				1													
12	2							1	1	1			1																	1	
13	1								1	1							1				1	1									
14	1											1								1											
14	2							1					1	1			1														
15	1		1																					1		1					







	M2 1	M 3	M 4	M 5	M2 3	M2 5	M2 7	M2 9	M 7	M2 6	M2 8	M3 0	M1 0	M1 2	M1 3	M1 7	M 6	M1 1	M1 4	M1 8	M 8	M 9	M1 5	M1 6	M1 9	M2 0	M 1	M 2	M2 2	M2 4	
P6(1)	0				1																										
P8(2)	0				1				1																						
P9(3)	0		1		1																						1				
P7		1	1	1	1	0	0	0																							
P32(1)		1	1	0	1	0	0	0																							
P34(2)		1	1	1	1	1	1	0																							
P36(1)		1	1	0	1	0	1	1																							
P37(2)		1	1	1	1	1	0	1																							
P25(1)									1	1	1	1																			
P30(2)									1	0	1	1																			
P33(1)									0	1	1	1																			
P38(1)									1	1	0	1																			
P40									1	1	1	0																			
P2(2)													1	1	1	1															
P3(2)													0	1	1	1															
P10(1)													0	1	1	1															
P12(1)													0	1	1	1															
P20													1	0	1	1															
P26(4)													1	1	0	1															
P31(3)													1	1	1	1															
P5(2)				1													1	1	0	0											
P14(1)																	0	1	0	1											
P21(3)																	1	0	1	1											
P24																	0	1	1	1											
P27(2)																	1	1	1	1											
P39(2)																	1	1	0	1											
P1(1)																						1	1	1	1	1	1				
P11(2)																						0	1	1	1	0	0				
P13																						1	1	0	1	1	1				

P16(2)																						0	1	1	1	1	1																						
P18(1)																							1	0	1	1	1	1																					
P35																							0	1	1	0	1	1																					
P4(2)				1																															1	1	1	0											
P15(1)																																					0	1	1	1									
P17(3)																																							1	1	1	1							
P19																																								0	0	1	1						
P22(1)	1																																								0	1	1	1					
P23(2)																																											1	0	1	1			
P28(1)																																												0	1	1	1		
P29				1																																										1	1	1	0

Şekil P.32 Çözüm P16

Parça No	Rota No	Mak. No	Mak Çeşidi										
			1	2	3	4	5	6	7	8	9	10	
1	1						1					1	1
2	1						1					1	
2	2		1		1							1	
3	1		1				1		1				
3	2						1		1				
4	1			1			1						1
4	2			1			1						1
4	3						1					1	1
5	1			1	1							1	1
6	1		1	1				1		1			
6	2		1	1				1				1	
7	1			1	1							1	
8	1		1		1	1							
8	2		1			1							
9	1							1		1			1
10	1		1		1						1	1	
11	1		1	1				1	1				
11	2			1				1				1	
11	3			1				1	1	1			
12	1						1		1				
12	2						1		1				1
13	1				1						1	1	
14	1		1	1				1					
14	2		1	1							1		
15	1				1		1		1		1		
15	2						1		1		1	1	
15	3						1		1	1			

Şekil P.33 Problem P17

	M1	M4	M5	M11	M6	M8	M10	M13	M2	M3	M7	M9	M12
P5(4)	0	1	1	1				1					
P7(2)	0	1	1	1									
P8(1)	1	1	1	0									
P10(2)	1	1	0	1								1	
P13(2)	0	1	0	1								1	
P1(2)					1	0	1	1					
P2(1)					1	0	1	0					
P3(3)					1	1	0	0					
P4(5)					1	0	1	1					
P12(3)					1	1	0	1					
P15(4)					1	1	1	1					
P6(2)									1	1	1	1	0
P9(1)									0	0	1	1	1
P11(2)						1			1	1	1	0	0
P14(2)									1	1	1	0	0

Şekil P.34 Çözüm P17

			Ç				
			1	2	1	1	2
P	R	M	1	2	3	4	5
1	1						1
2	1						1
3	1		1		1		1
4	1		1				1
5	1						1
6	1				1		1

Şekil P.35 Problem P18

	M3	M5	M7	M1	M4	M6	M2
P1(4)	1	1	1	1			
P2(2)	0	1	1				
P3(4)	1	1	1		1		
P4(1)				1	1	0	1
P5(1)				1	1	1	
P6(1)		1		1	1	1	

Şekil P.36 Çözüm P18

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

Şekil P.37 Problem P19

	M5	M8	M12	M7	M10	M11	M14	M1	M2	M4	M9	M3	M6	M13
P1(1)	1	1	1			1						1	1	
P7	0	1	1											
P11(2)	1	1	1			1						1		1
P18(1)	1	1	0							1				
P2(2)				1	1	1	1						1	
P3(4)				0	1	1	1	1		1				1
P12(2)		1		0	1	1	1							
P13(2)				1	1	1	1							
P14				1	1	0	0			1		1		
P8(1)								0	1	1	1			
P10(1)				1				1	1	1	1			
P15(3)							1	0	1	1	0			
P4					1			1		1		1	1	0
P5(2)			1									0	1	1
P6(4)		1					1					0	1	1
P9(2)			1	1		1						1	0	1
P16(2)				1		1						1	1	1
P17(2)						1						1	1	1
P19(2)			1		1						1	1	0	1
P20(2)				1								0	1	1

Şekil P.38 Çözüm P19

			Ç	1	1	2	1	1	1	2	1	1	1	1	1	1	1	1
P	R	M	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1			1	1						1						1	
1	2					1								1		1	1	
1	3						1	1						1		1		
2	1			1					1		1						1	
3	1			1					1		1				1			
3	2				1	1					1	1						
4	1			1				1	1						1	1		
5	1						1			1			1			1		
6	1								1			1		1		1		
6	2		1					1			1	1						
7	1					1	1						1	1				1
8	1					1						1	1	1				
8	2			1			1		1	1								
8	3		1			1				1			1					
9	1		1	1						1			1		1			
10	1					1		1	1			1			1			
11	1		1					1								1	1	
11	2		1				1	1		1								
12	1					1			1	1								
12	2					1									1		1	
13	1				1						1						1	
14	1											1		1				1
15	1												1		1		1	
16	1							1			1	1						1
16	2					1	1			1		1						
17	1										1	1		1		1		
18	1			1						1	1	1					1	
19	1					1						1		1				1
20	1					1		1					1					
21	1				1	1		1									1	
22	1					1		1				1		1				
23	1			1	1							1		1		1		
24	1						1		1	1								
25	1			1										1	1		1	
26	1							1	1					1				
27	1			1		1								1				
28	1			1		1						1						1
29	1				1				1				1					
29	2					1	1		1									
30	1			1									1			1		

Şekil P.39 Problem P20

	M 7	M 17	M 1	M 5	M 8	M 11	M 14	M 2	M 9	M 3	M 13	M 15	M 4	M 10	M 12	M 16	M 6	M 18
P29(3)	1	1				1												
P5			0	1	1	1	1											
P9			1	0	1	1	0	1			1							
P11(2)			1	1	1	0	0										1	
P30			0	0	0	1	1	1										
P1(1)								1	1	1		1						
P2(2)								1	1			1						1
P3(1)	1							1	1		1							
P18					1			1	1			1		1				
P12(3)										0	1	1	1					
P13(1)									1	1	0	1						
P15						1				0	1	1						
P21(1)										1	0	1	1					1
P25										1	1	1			1			
P6(2)							1						0	1	1	0		1
P7			1			1							1	0	1	1		
P8(1)						1							1	1	1	0		
P14													0	1	1	1		
P16(1)									1				0	1	0	1	1	
P17						1		1					0	1	1	0		
P19													1	1	1	1		
P22													1	1	1	0	1	
P23(2)	1						1	1					0	1	1	0		
P27								1					1	0	1	0		
P28								1					1	1	0	1		
P4(2)							1	1			1						1	1
P10(2)											1		1	1			1	1
P20						1							1				1	0
P24(2)				1													1	1
P26(2)															1		1	1

Şekil P.40 Çözüm P20



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

Şekil P.41 Problem P21

	M	M	M	M	M	M	M	M	M	M	M	M	M	M	M	M	M	M	M	M
	3	8	11	19	1	7	9	12	10	14	17	18	4	13	15	2	5	6	16	20
P11(2)	1	1	1	1																
P12(2)	1	1	0	1																
P13(2)	1	1	1	1																
P1(2)					0	1	1	1												
P2(2)					1	1	0	1												
P3(2)					1	1	1	1												
P4(2)					1	1	0	1												
P5(2)			1		1	0	1	1												
P14(1)									1	1	1	1								
P15(1)									1	0	1	1								
P16(1)									1	1	0	1								
P17									1	1	1	0								
P18													1	1	1					
P19													1	1	1					
P20													1	0	1					
P6(1)																1	1	1	0	0
P7(3)																0	1	1	1	1
P8(1)																1	0	1	1	0
P9(3)																1	0	1	1	1
P10(1)																1	1	1	1	0

Şekil P.42 Çözüm P21