

66856

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**ESNEK İMALAT SİSTEMLERİNDE
HÜCRE OLUŞTURMA PROBLEMİNE
0-1 TAMSAYILI PROGRAMLAMA İLE MODELLEME YAKLAŞIMI**

YÜKSEK LİSANS TEZİ

İşl.Müh . Süleyman Ertan TANSÖKER

66856

Tezin Enstitüye Verildiği Tarih : ŞUBAT 1997

Tezin Savunulduğu Tarih : ŞUBAT 1997

Tez Danışmanı

: Doç.Dr. Semra DURMUŞOĞLU

Diğer Jüri Üyeleri

: Doç.Dr. Burç ÜLENGİN

: Yrd.Doç.Dr. Demet BAYRAKTAR

ŞUBAT 1997

**T.C. YÜKSEKÖĞRETİM KURULU
DOKÜMANTASYON MERKEZİ**

ÖNSÖZ

Grup Teknolojisi ve Matematiksel Programlama konularında çalışmalarımı destekleyen, her türlü kaynak ve fikir yardımlarını esirgemeyen, tezin değişik aşamalarındaki inceleme ve eleştirileriyle değerli katkılarda bulunan Sayın Hocam Doç.Dr. Semra DURMUŞOĞLU'na ve bana maddi, manevi her türlü desteklerini sunan aileme en içten teşekkürlerimi sunarım.

Süleyman Ertan TANSÖKER

İSTANBUL/15.01.1997

İÇİNDEKİLER

	<u>Sayfa No</u>
ÖNSÖZ	ii
SEMBOL LİSTESİ	vi
ŞEKİL LİSTESİ	viii
TABLO LİSTESİ	ix
ÖZET	x
SUMMARY	xi
BÖLÜM 1.GİRİŞ	1
BÖLÜM 2.GRUP TEKNOLOJİSİ	3
2.1. Hücresel İmalat Sistemleri ve Grup Teknolojisi	3
2.2. İmalat Politikası	7
2.3. Grup Teknolojisinin İmalat Sistemlerindeki Baskın Maliyetlere Etkisi	8
2.4. Grup Teknolojisi'nde Kümeleme Yöntemleri	12
2.4.1.Görsel Belirleme Yöntemi	12
2.4.2.Parça Özelliklerine Dayalı Yöntemler	13
2.4.3.Üretim Prosesine Dayalı Sistemler	14
2.5.Genel Modelleme Kriterleri	15
2.6. GT 'de Üretim Akış Analizi İle Hücre Tasarımı	16
2.6.1.Benzerlik Katsayılarını Temel Alan Çalışmalar	16
2.7.Parça Makina Matrisinde Diyagonal Dizili Gruplar Oluşturan Yöntemler	20
2.7.1.Bağ Enerjisi Yöntemi	20
2.7.2.Derece Sıralamasıyla Kümelenendirme Yöntemi	22

	<u>Sayfa No</u>
2.7.3. İş Parçası ve Makina İstatistikleri Kullanarak Hücresel Tasarımı	24
BÖLÜM 3.TAMSAYILI DOĞRUSAL PROGRAMLAMA	27
3.1.Giriş	27
3.2.Balas Algoritması	28
3.3.Hücrel İmalat Sistemlerinde Makine Gruplama Problemine 0-1 Tamsayılı Programlama Yaklaşımı	41
3.3.1.Önerilen Yöntem	41
3.3.2.Matematiksel Modelleme	42
3.3.2.1.Değişkenlerin Gösterimi	42
3.3.2.2.Uyumluluk İndeksinin Oluşturulması	43
3.3.2.3.Makina Gruplama Modellerinin Formülasyonu	44
3.4.3.3.1.Model 1 - Mevcut Bir Sistemi Yeniden Organize Etme	44
3.4.3.3.2.Model 2 - Yeni Bir Sistem Kurma	46
BÖLÜM 4.LİTERATÜR TARAMASI	47
4.1.Giriş	47
4.1.1.Parça Aileleri İçin Sınıflandırma ve Kodlama Sistemleri	48
4.1.2.Makina-Parça Grup Analizi	48
4.1.3.Benzerlik Katsayısına Dayalı Kümeleme Yöntemleri	49
4.1.3.1.Grafik Teorisine Dayalı Yöntemler	50
4.1.3.2.Benzerlik Katsayısına Dayalı Matematiksel Modeller	50
4.1.4.Matematiksel ve Sezgisel Yöntemler	50
4.1.5.Bilgi Tabanlı ve Şekil Tanımaya Yönelik Yöntemler	51
4.1.6. Bulanık Kümeleme Yaklaşımı	51
BÖLÜM 5.UYGULAMA	52
5.1.Sistemin Tanımlanması	52
5.2.Modelin Kurulması	54
5.3.Parça Ailelerinin Elde Edilmesi	56

	<u>Sayfa No</u>
SONUÇLAR ve ÖNERİLER	58
KAYNAKLAR	60
EKLER	62
EK A-Model 1'e Ait Maksimizasyon Fonksiyonunun Bilgisayar Yazılımı İle Elde Edilerek LINDO'ya Aktarılan Veri Yapısı	62
EK B-Parça-Hücre Matrisi Oluşturma Programı	65
EK C-Parça-Makina Benzerlik Katsayıları Oluşturma Programı	66
EK D-Parça Ailesi Oluşturma Programı	68
EK E-Makina-Takım Verileri Oluşturma Programı	77
EK F-Parça-Takım Verileri Oluşturma Programı	79
EK G-Parça-Makina Ortak Takım Verileri Oluşturma Programı	81
EK H-Max.Amaç Fonksiyonunun Oluşturma Programı	83
ÖZGEÇMİŞ	84

SEMBOL LİSTESİ

- a_{ij} : Kısıt denklemlerindeki X_j 'lere ait teknik katsayıları
- a_{ik} : Modelin amaç fonksiyonundaki parça-hücre değişkeni
- A_j : j. makinanın mevcut miktarı
- b_i : Kaynak değerleri
- B_j : Alınabilecek max. sayıdaki j makinası
- c : İki hücre arası taşımanın ortalama maliyeti
- C_2 : Herbir hazırlığın maliyeti
- C_3 : Bir birim stoğu elde bulundurma maliyeti
- C_j : Amaç fonksiyonundaki katsayıları
- D : Yıllık iş parçası talebi
- f_j : j. makinanın yıllık sabit maliyet oranı
- G_k : k hücresindeki makina sayısı limiti
- i : Parça indeksi , $i=1,...,n$
- j : Makine indeksi , $j=1,...,m$
- k : Hücre indeksi , $k=1,...,p$
- L_i : i. parçanın birim taşıma yükü hacmi
- M_j : j. makinedeki mevcut takım sayısı
- M : İş parçasının bir biriminin hammadde maliyeti

N_i : i.parça için gereken takım sayısı

N_b : Ekonomik parti miktarı

O_{ij} : i. parça üzerinde j. makinede yapılması gereken işlem sayısı

P_i : i. parça üzerinde yapılması gereken tüm işlemlerin kümesi

q_i : i. parçanın yıllık üretim miktarı

R : Birim imalat zamanı için yüklenilmiş maliyet

S_i : Artık(Slack) değerler.

S_u : İş parçasının imalatla karşılaştığı tüm makinalar için toplam hazırlık zamanı

S_{ij} : i. parça ile j. makina arasındaki uyumluluk indeksi

t_{il} : i. parçanın l. işlemi için gereken süre

T_r : İş parçasının bir biriminin toplam işleme zamanı

T_{ij} : i. parça ile j. makine arasındaki ortak takım sayısı

X_{ii} : Sadece i makinasına uğrayan toplam parça sayısı

X_{ij} : i ve j makinalarının her ikisine de uğrayan toplam parça sayısı

X_{jj} : Sadece j makinasına uğrayan toplam parça sayısı

X_{jk} : Modelin amaç fonksiyonundaki makina-hücre değişkeni.

ŞEKİL LİSTESİ

	<u>Sayfa No</u>
Şekil 2.1. Klasik Makinalardan Yararlanma Oranları	9
Şekil 2.2. Sipariş Tipi Atölye Sisteminde Toplam İmalat Zamanının Analizi	10
Şekil 2.3. Makina-Makina Kombinasyon Matrisi	17
Şekil 2.4. Dendogram (Ağaç Diyagramı)	19
Şekil 2.5. Parça-Makina Matrisi ve Sahip Olduğu Bağ Enerjisi	20
Şekil 2.6. Sonuç Parça-Makina Matrisi ve Sahip Olduğu Bağ Enerjisi	22
Şekil 2.7. Parça-Makina Matrisinin Satırlarının İkili Sisteme Göre Değerlendirilmesi	23
Şekil 2.8. Satırların Düzenlenmesi İle Eldē Edilen İkinci Matris ve Bu Yeni Matrisin Sütunlarının İkili Sisteme Göre Değerlendirilmesi.	23
Şekil 2.9. Elde Edilen En Son Matris ve Bu Matrisin Satır ve Sütunlarının İkili Sisteme Göre Değerlendirilmesi.	24
Şekil 3.1. Başlangıç İterasyon	33
Şekil 3.2. İkinci İterasyon	34
Şekil 3.3. Balas Algoritması	41

TABLO LİSTESİ

	<u>Sayfa No</u>
Tablo 2.1. Örnek Matris İçin Her Bir Makina Çiftine Ait Benzerlik Katsayıları	17
Tablo 2.2. Satır Elemanları Seçenekleri	21
Tablo 2.3. Satır Elemanları Seçenekleri	21
Tablo 2.4. Tamsayılı Programlama Probleminin Katsayı Tablosu	32
Tablo 5.1. Parçaları İşlemek İçin Gerekli Takımlar	53
Tablo 5.2. Makinalarla İlgili Bilgiler	53
Tablo 5.3. Parça Gruplarının Kompozisyonu	57
Tablo 5.4. Modelin Çözümüne Ait Sonuçlar	57

ÖZET

Günümüzde mal ve hizmet pazarları artan bir oranda globalleşme eğilimi göstermektedir. Bu durum beraberinde artan rekabet koşullarını getirmekte; firmaların faaliyetlerini sürdürerek ayakta kalabilmeleri ancak bu zorlu rekabet koşullarına adapte olabilmelerine bağlı olmaktadır.

Bölgesel pazarların giderek daha güçlü hale gelmesi ülkemizin de Avrupa Topluluğu'na başvurusunu zorunlu kılmıştır. Bunun sonucunda 1996 yılından itibaren başlayan tam üyelik öncesi Gümrük Birliği Anlaşması ülkemiz ve Avrupa Topluluğu ülkeleri arasında imzalanmıştır. Böylece hemen hemen tüm ürünler bazındaki gümrük tarifeleri karşılıklı olarak sıfırlanmıştır. Doğaldır ki topluluk ülkeleri özellikle sanayi ürünlerinde teknoloji ve kalite açısından ülkemizden daha ileri konumdadırlar. Bu nedenle firmalarımızın rekabet edebilmeleri için kalitede dünya standartlarını yakalamaları ve maliyet açısından daha etkin çalışmaları şarttır. Sanayi ürünlerinde hammadde ve enerji girdilerinde maliyetleri azaltmak pek kolay olmamaktadır. Dolayısıyla ancak verimliliği artırarak mevcut girdilerle daha fazla ürün üretmek toplam maliyetleri etkilemese de birim maliyetleri azaltarak rekabet avantajı sağlayacaktır. Günümüzde pazar talepleri ürünlerin daha fazla çeşitli ve daha kısa sürelerde üretilmesini de gerektirmektedir.

Grup Teknolojisi bu noktada işe yaramakta; büyük imalat sistemlerini daha küçük makina ve parça aileleri olarak gruplayarak daha etkin çalışan sistemlere ulaşılmasını sağlamaktadır. Bu çalışmada grup teknolojisi ile makina-parça aileleri oluşturma problemine tamsayılı programlama ile modelleme yaklaşımı ele alınmış ve mevcut bir sistemin bu modelle yeniden oluşturulmasına çalışılmıştır. Sistemin bilgisayar destekli çalışabilmesi için gerekli bilgisayar programları üretilmiştir.

SUMMARY

MODELING APPROACH WITH 0-1 INTEGER PROGRAMMING

FOR CELL FORMATION IN

FLEXIBLE MANUFACTURING SYSTEMS

Product and service markets have recently tend to be global markets in an increased proportion. That is the reason of the increasing competition in market economy. Thus it is a must to adopt the strong market conditions for the firms to survive.

The more competitive and strong positions of regional markets around the world had forced our country to apply for the membership to The European Union. In 1996, as a result, Customs Union Agreement had been signed between Turkey and member countries of the union. Afterwards the tariffs on almost all of the products had been decreased to zero level. Since the member countries of the union have more competitive advantages on industrial products from the point of quality and technology, our firms have to reach the worldwide standards and work much more effective to catch a competitive position.

Obviously it is not easy to decrease the costs of energy and resources. Consequently the only way to acquire the competitive advantage is to increase the productivity since it provides remarkable decrease in unit costs of goods. Nowadays the changing demand of the markets requires more differentiated products with shorter production time. Thus the manufacturing systems have been forced to be organized in such a way that they will provide production with shorter time and smaller lots. The Group Technology is a powerful tool at this point to form machine-part families which work more cost and time effective.

In this thesis the modeling by 0-1 integer programming approach to form the machine-part families have been studied and a present manufacturing system reorganized by the model. The required software have been written to use the computing facilities.

In the first chapter the group technology and its effects upon costs in manufacturing systems, the application methods for the group technology, the general searching groups and assumptions have been studied. As stated in the chapter that the group technology is a kind of manufacturing philosophy. This philosophy forms the principles to benefit from the advantages of similarities in all stages of designing an manufacturing while gathering the similar parts or processes.

The results of the researches on the factors that effect the productivity of manufacturing systems are as follows:

- 1 - The effect of the improvement in manpower quality is 15%.
- 2 - The effect of the force to acquire the required capital fund is 25%.
- 3 - The effect of the improvements in manufacturing technologies is 60%.

The group technology studies take place in the last group . According to the results of a comparative productivity research including 270.000 operation selected by sampling, neither machines nor manpower can be used beneficially during 50% of the production time. That's a remarkable result. If the time of machine usage can be increased by this philosophy the processes will be more cost and time effective. The three different methods that have been based on to form the consistent machine-part families in group technology are as follows:

1-Visual inspection method that depends on analyzing the geometrical patterns of the parts to form the part families.

2-Methods depending on part characteristics which are also referred to part coding and classifying analysis.

3-Methods depending on the production process. This is a very popular method also named as Production Flow Analysis. After gathering the initial data there are three steps respectively in applying this method.

- a. Forming the part packages.
- b. Forming the matrix of production flow analysis.
- c. Analyzing the matrix to form the machine-part families.

In the second chapter the fundamental cell formation methods have been studied. They are called as follows:

1-Cell formation by production flow analysis. This analysis had been developed by Burbidge. Unfortunately it has two important problems. The first is that the analysis' depend on heuristic procedures. As the second, it not probable to reach a feasible solution while the machine-part matrix is to large. In the analysis a similarity coefficient is defined as

$$S_{ij} = X_{ij} / (X_{ii} + X_{jj} - X_{ij}) \quad (1)$$

S_{ij} : The similarity coefficient

X_{ii} : The number of parts that are operated by only i. machine.

X_{jj} : The number of parts that are operated by only j. machine.

X_{ij} : The number of parts that are operated by both i. and j.machine.

The more the value of X_{ij} the more is the similarity. According to a previously determined threshold T value the groups are formed.

2-Methods that forms diagonally ordered groups in machine-part matrix. The first method is the bond energy method. In essence this is a quadratic assignment problem . When the problem is large the counting time increases exponentially. Thus a heuristic method had been developed to reach secondary optimal solutions. The second is the rank order clustering algorithm. It was developed by King to form diagonally ordered groups. Techniques that will produce better solutions especially for the bottle-neck machines. The last one is the cell formation using machine and part statistics. In this approach first the machine families are formed, later the parts are assigned to these groups without forming the part families.

In the third chapter 0-1 integer programming approach and The Additive Algorithm or Balas Algorithm have been explained. In this algorithm the variables can only have 0 or 1 values. There are two parts as the objective function and the constraints. In the method, the objective function must be a minimization type and the constraints must be less than or equal to zero.

$$\min Z = \sum C_j X_j \quad j=1, \dots, n$$

$$\sum a_{ij} X_j + S_i = b_i \quad i=1, \dots, m$$

$$X_j=0 \text{ or } X_j=1$$

$$S_i \geq 0$$

Herein,

C_j : The coefficients of the objective function.

X_j : The 0-1 variables.

a_{ij} : The technical coefficients in the constraint matrix.

b_i : The values of the resources.

S_i : The values of the slack variables that added the model later.

The transformation techniques for incompatible problem types are provided by solving an example problem.

In the last chapter the 0-1 integer programming approach to machine grouping problem in cellular manufacturing systems and two different mathematical models have been studied. The objective is to assign the machines into previously formed part groups. The methods are applied depending on the tooling requirements of the parts, the present tools and the operating times. In the first formulation machines are grouped based on the compatibility index between the parts and machines. On the other hand, in the second formulation the transportation costs among the cells and the layout costs are aimed to minimize while grouping the machines. Both the number of machines of each type and the maximum number of machines in a group are assumed as parameters. The variables of the models are as follows:

- i : Part index , $i=1,...,n$
 j : Machine index , $j=1,...,m$
 k : Cell index , $k=1,...,p$
 N_i : The number of the tools required for the part i .
 M_j : The number of the tools available on the machine j .
 T_{ij} : The number of the common tools between parts i and j .
 O_{ij} : The number of the operations on part i machine j .
 P_i : The set of the operations that must be done on part i .
 S_{ij} : The compatibility index between part i and machine j .
 t_{il} : The required time for operation l of part i .
 q_i : The annual production amount of part i .
 f_j : The annual fixed cost ratio of machine j .
 L_i : The unit handling size of the part i .
 c : The average cost of transport between two cells.
 A_j : The available number of machine j .
 B_j : The max. number of machine j that can be procured.
 G_k : The limit of the machine number in cell k
 $a_{ik} = \begin{cases} 1 & , \text{ if part } i \text{ is in cell } k \\ 0 & , \text{ otherwise} \end{cases}$

The decision variable X is defined as

$$X_{jk} = \begin{cases} 1 & , \text{ if machine } j \text{ is in cell } k \\ 0 & , \text{ otherwise} \end{cases}$$

Two different compatibility index are defined in the models.

1-Compatibility index based on the tooling requirements.

$$S_{ij} = T_{ij} / \min \{M_j, N_i\} \quad (2)$$

This equation states the operational compatibility between a machine and a part. If there are a number of common tools between a machine and a part, that means less transportation and handling costs.

2-Compatibility index based on both the tooling requirements and operation time.

$$S_{ij} = \left(\sum_{l \in O_{ij}} t_{il} \right) / \left(\sum_{r \in P_i} t_{ir} \right) \quad (3)$$

If the machine and the part are grouped together, while the operation time is as long as possible then the working rate of the machine will be that much higher.

In this study two different models are proposed as mentioned before. These models are as follows:

Model 1-Reorganizing an available system.

$$\max Z_1 = \sum_i \sum_j \sum_k X_{jk} * a_{ik} * S_{ij} \quad (4)$$

The first objective function aims to maximize the sum of the compatibility index for all machine and parts in all cells.

$$\min Z_2 = \left(\sum_j \sum_k f_j * X_{jk} \right) - \left(\sum_i \sum_j \sum_k X_{jk} * a_{ij} * T_{ij} * C * q_i / L_i \right) \quad (5)$$

The second objective function is formed for the equilibrium between the machine assignment and handling cost among cells. The handling cost is faced when any tool of a machine in cell m is required to operate on a part in cell k. Here $m \neq k$ is assumed.

The first term in the equation is the assignment cost of a machine in a cell. The second one is the total savings from intercell movement. $T_{ij} * (q_i / L_i)$ is the number of movement between any couple of cells. Then $T_{ij} * C * q_i / L_i$ means the average cost of these movements. If these parts are grouped in a common cell, that means the savings from the cost of movements.

The constraints of the system are as follows:

$$\sum_j X_{jk} \leq G_k \quad \text{for all } k \quad (6)$$

$$\sum_k X_{jk} = A_j \quad \text{for all } j \quad (7)$$

$$X_{jk} \in (0,1) \quad \text{for all } j \text{ and } k \quad (8)$$

Model 2-Organizing a new system.

The objective function of the model is given in equation 4. The cell size constraint is the same as equation 6 and 8. But equation 7 is converted as follows:

$$X_{jk} \leq B_j \quad \text{for all } j \quad (9)$$

This constraint satisfies that the maximum number of each type of machines that can be procured will not exceed. The model does not assign a machine to a cell if it is not cost effective. It's required to invoke a constraint that at least one of each type of a machine will be assigned to any cell. Otherwise it would be impossible to do some operations on some of the parts.

$$\sum_k X_{jk} \geq 1 \quad \text{for all } j \quad (10)$$

Model 1 is applied to a manufacturing system including 10 machine and 25 different parts. The required information can be obtained from Table 5.1 and Table 5.2. The cost of an inter-cell movement is supposed TL 500.000. The unit loading size for all parts is 5. The number of machines that can be assigned to a cell is limited to 6. Meanwhile the part families are formed by a heuristic based on tooling requirements. The results are presented in Table 5.3 and Table 5.4.

As the conclusion if the proposed mathematical models presented in the thesis are applied to form machine-part families in large sized integrated flexible manufacturing systems, the layout will be more cost effective.

Key Words : Group Technology, Integer Programming, Production Flow Line

BÖLÜM 1

GİRİŞ

Grup Teknolojisi (GT) bir imalat felsefesidir. Bu felsefe benzer parça yada proseslerin belirlenip biraraya getirilerek, tasarım ve imalatın tüm aşamalarında bulunan benzerlik avantajlarından yararlanılmasını sağlayan prensipleri ortaya koymaktadır. Günümüzde bilgisayar destekli imalat sistemlerinin temelinde grup teknolojisi yatmaktadır. Gelecekte çok kısa süreli ve küçük partiler halinde, daha fazla sayıda ürün çeşidine dayanan imalat önemini daha da arttıracaktır. Çünkü gelişen imalat teknolojileri ile imalat sürelerinde kısalma, kalite düzeyinde artış sağlanmaktadır. İletişim teknolojilerinin hızla gelişimi, dünya pazarlarının büyük gruplar halinde birleşerek tekil pazarlar haline gelmesi rekabeti önleyici çeşitli kısıtlamaları ortadan kaldırmaktadır. Böylece malların dolaşımı giderek daha serbest hale gelmektedir. Bu durum dünya pazarlarında yüksek teknoloji, ucuz işgücü, ucuz ve kaliteli hammadde kullanabilen üreticilere maliyetlerde ve fiyatlamada çok büyük bir avantaj sunmaktadır. Ancak Türkiye gibi gelişmekte olan ülkelerin teknolojik avantajları çok fazla olmadığından, mevcut imalat kapasitelerinin daha etkin kullanımı, maliyet etkinliği ve verimlilik artışı sağlayabilmek açısından Grup Teknolojisi, Tam Zamanında Üretim Teknolojisi , Optimize Edilmiş Üretim Teknolojisi gibi yeni üretim felsefelerini biran önce uygulamaya koymaları, imalatın tüm aşamalarında bilgisayarlardan yararlanmaya başlamaları rekabet edebilmek için gerekmektedir.

Bu çalışmanın ilk bölümünde Grup Teknolojisi hakkında genel bir giriş yapılmış; ikinci bölümde GT'nin imalat sistemlerindeki maliyetlere etkileri, uygulanma yöntemleri, hücre tasarımı ile ilgili temel yöntemler, konu ile ilgili yapılan çalışmalardaki genel inceleme grupları ile varsayımlar ele alınmıştır. Üçüncü bölümde çalışmanın temelinde yer alan 0-1 tamsayılı programlama üzerinde durulmuş ve modelin çözümünde

kullanılan Balas Algoritması, oluşturulan hücre oluşturma modeli detaylarıyla açıklanmıştır. Dördüncü bölümde ise GT ile ilgili olarak literatürde mevcut olan çalışmalar ve özellikleri incelenmiştir. Beşinci ve son bölümde elde edilen veriler önerilen modellerden Model-1 mevcut bir sistemin yeniden organize edilmesi için uygulanmış, sonuçlar tartışılmıştır.



BÖLÜM 2

GRUP TEKNOLOJİSİ

2.1. HÜCRESEL İMALAT SİSTEMLERİ VE GRUP TEKNOLOJİSİ

Hücresel imalat sistemleri, bir imalat sistemi içinde benzer imalat karakteristiklerine sahip belirli bir parçalar grubunun (parça ailesi) tamamen imalatı için işlem, insan ve özellikle makina gruplarının varolduğu veya oluşturulduğu sistemlerdir. Hücredeki tüm tesis ve birimler, hücre içine giren tüm parçaları kendi kendine yeter bir seviyede imal etmek üzere organize edilmiştir. Böyle bir yaklaşım, küçük bir sistemin etkin ve denetlenebilir olma özelliğini büyük sistemlere yansıtmak amacını taşır. Bu yaklaşım büyük sistemin içinde, birbirinden alabildiğince bağımsız küçük sistemler (hücreler) oluşturma şeklinde gerçekleşir. Böylece tüm sistemin karmaşıklığı yerine, oluşturulan küçük sistemlerin problemleri ile uğraşılır.

Grup Teknolojisi (GT) ise birçok problemin temelde birbirine benzer olduğundan hareketle, benzer problemleri bularak toplayan ve böylece elde edilen gruplar için tek bir çözüm arayan bir yaklaşımdır. Böylece zaman ve çabalardan tasarruf sağlanır. Üretimde başarılı bir GT ise, öncelikle imalatta grup düzenleme ile başlamalı, ardından üretimde akış kontrolü ile devam ettirilerek tüm üretim-yönetim sistemine genişletilmelidir. [1]

GT'nin en önemli ve esas avantajı fonksiyonel düzenlemeye nazaran iş akışını basitleştirmesi ve bu arada akış tipi imalata, makina zaman verimlerinde büyük dengesizlikler olmadan imkan vermesidir. Temelde diğer avantajlar da bu iş akışının basitleşmesine dayanır. Basit iş akışı ile makina önündeki iş parçası bekleme veya kuyruk zamanları düşürülerek veya ortadan kaldırılarak proses içi stok maliyetlerinin

düşmesi ve daha kısa imalat zamanlarına ulaşılması sağlanır. Ayrıca GT, işlem için hazırlık zamanlarını da açık bir şekilde düşürür veya ortadan kaldırır. Çünkü hücre içindeki makinalar ve tertibatlar bu hücrede imal edilecek bir parça ailesi için bir tip parçadan diğerine çok hızlı geçecek şekilde yeniden tasarlanabilir. Böylece küçük parti üretimi mümkün olabilmektedir.

GT, mamul kalitesinde de bir iyileşmeye neden olur. Çünkü küçük parti üretiminin hücre düzeninde, bir işgören bir parçayı doğrudan diğer işgörenden alabilir. Böylece eğer parça hatalı ise, işlem neyin yanlış gittiğini anlamak için durdurulur. Kalite geri beslemesi derhal gerçekleştirilip yüksek kalite kolaylıkla sağlanabilir. Bundan başka postabaşılar da gruba giren parçaların tüm işlemleri grup içinde gerçekleştiğinden, parçanın bütünü için kalite sorumluluğu taşımaktadır.

Bununla beraber GT'de parça aileleri ve makina gruplarının oluşturulması (grup düzenlemesi) sırasında birçok problem ve zorluklar ortaya çıkar. Bunlardan en önemlileri şunlardır:

- a) Toplam parça ve makinaların hücrelere bölünmeleri için uygun bir yöntemin benimsenememesi.
- b) Hücre boyutunun ekonomik açıdan değerlendirilmesinin zor olması.
- c) Sayısı az makina tiplerinin, özellikle kendi sayısından daha fazla hücreye dağıtılması halinde, bazı makinaların çok fazla, bazılarının çok az yüklenmeleri, başka bir deyişle makina zaman verimlerindeki anormal değişimler ve ilave makina ihtiyaçlarının ortaya çıkması.
- d) Ürün karması düzensiz değişimlerinin hücrelerdeki makina verimleri üzerine olumsuz etkisi.
- e) Hücrelerin, mamullerin gelecekteki talep artışlarını karşılamak üzere esnek tasarımılanması gereği.
- f) Makina arızalarının üretimi uzun zamanlar aksatma eğilimi.

Bütün bu zorluklara rağmen ele alınan üretim sistemi şartlarına uygun GT prensiplerinin çok iyi tasarlanması ve sisteme uygulanması ile bu sorunlar yenilebilir veya maliyeti artırıcı bazı sorunlar ortadan kaldırılabılır.

GT'de ikinci önemli adım, üretimde stok kontrolü yerine akış kontrolünü sağlamaktır. Akış kontrolü için temelde iki yaklaşım vardır. Birincisi, satış tahminlerine dayalı geleceğe yönelik planlamanın üretimi "ittiği" İtme Sistemi'dir(İng:Push System). Burada kontrol üç aşamada gerçekleşir. Bunlar sırasıyla ;

- 1) Programlama
- 2) Sipariş verme
- 3) İşi başlatma ve grup çizelgeleme olmaktadır.

İkinci yaklaşım, bir prosesin işini bitirdiği zamanda kendinden önceki prosesden malzeme sipariş ettiği ve aldığı, başka bir ifadeyle gerekli imalat miktarlarıyla proseslerin bütünleştirildiği Çekme Sistemi'dir (İng:Pull System).

GT'de grup düzenlemesi ve üretimde akış kontrolü gerçekleştirildikten sonra son aşama, sistemin organizasyon yapısının bu değişikliklere uydurulacak bir şekilde yeniden düzenlenmesidir. [2]

Günümüzde nümerik kontrollü tezgahlardaki gelişim kısa süreli imalatı sadece uygun kılmakla kalmayıp maliyet açısından da desteklemektedir. Eğer üretilen parçalar parça aileleri olarak sınıflandırılabilir ve makinalar da gruplar halinde düzenlenebilirse üretim sırasında parçaların taşınması robotize bir iş olmaktadır. Çünkü üretim sırasında parçalar her yerde işlem görmeyecekler ve dağılımları kontrol edilebilecektir [3].

Geleneksel kitlesel üretim endüstrisinde, önce üretilecek parçalar gereken işlemlere göre ayrılır ve bir bölümden diğerine uzun yollar katederek taşınır. Örneğin bir fabrikada yapılan bir araştırma parçaların değişik proses aşamalarından geçerken 4,5 millik bir yol katettiğini ortaya çıkarmıştır. Bu durum ürün çıktı sürelerini etkilemektedir. Uzun ve belirsiz çıktı süreleri küçük parti imalatı yapan firmaların mal

tesliminde yaşadıkları gecikmelerin sebebidir. Bu küçük bir sorun değildir. Parçalar birer birer monte edildikleri için bir parçanın gecikmesi tüm montajın tamamlanmasını engellemektedir. Böyle bir sorunla karşılaşılması için de ilgili parçalardan daha fazla stok bulundurmak gerekmektedir.

Grup Teknolojisi işletmelere en fazla imalat verimliliği kazandıran teknolojik ilerlemelerden bir olarak karşımıza çıkmaktadır. Yapılan araştırmalarda imalat verimliliğini en fazla etkileyen üç faktörün aşağıdakiler olduğu sonucuna varılmıştır [4].

1 - İşgücünün kalitesindeki ilerlemenin etkisi % 15

2 - Gerektiğinde yeterli sermaye temin edebilme gücünün etkisi % 25

3 - Üretim teknolojisindeki ilerlemelerin etkisi %60. Grup Teknolojisi de üçüncü olarak belirtilen üretim teknolojilerindeki ilerlemelere katkıda bulunmaktadır.

Hollier ve Corlett adlı araştırmacılar birçok fabrikada kitlesel üretim üzerine yaptıkları çalışmalarda bir işlemin bir hafta sürdüğünü varsayarak, ortalama büyüklükte bir partiyi üretmek için gereken işlem sayısı ile ortalama çıktı süresi arasında doğrusal bir ilişkinin varolduğunu saptamışlardır. Onların ulaştıkları sonuçlar aşağıdaki gibidir [5]:

1 - Çıktı zamanı işyükünün mevcut üretken kapasiteye oranına bağlıdır. Kapasite kullanımı % 100'e yaklaştıkça tezgahlarda işlenmeyi bekleyen parçalar uzun kuyruklar oluşturmaya başlamakta, bu durum da fazla çalışmaya neden olmaktadır.

2 - Eğer partilerden birine yüksek öncelik verilirse, bu durum öncelik verilmeyen partilerde daha fazla gecikmeler oluşturmaktadır.

3 - Parçalar üretimde geçen sürenin % 90'ını sadece prosesler için bekleyerek geçirirler.

4 - Çıktı süresinde ne parti büyüklüğünün ne de proses süresinin önemli bir etkisi vardır.

Tüm bu saptamalar yüksek makina kullanım oranlarının iş miktarını çok artırdığı ve uzun çıktı süresine neden olduğu sonucunu ortaya koymaktadır. Bu durumun gerçek maliyeti ve işletmenin bütününe olan etkisi ise genelde dikkate alınmamaktadır.

Yine Hollier ve Corlett tarafından yapılan başka bir çalışmada 40 mühendis ve örnekleme yoluyla belirlenen 270.000 faaliyeti içeren karşılaştırmalı bir verimlilik araştırması yapılmıştır. Varılan sonuçlara göre toplam çıktı süresinin % 20'si yararlı bir şekilde kullanılmaktadır. Zorunlu olarak gerekli taşıma ve bekletme süreleri de dikkate alındığında çıktı süresinin en azından % 70'inin boşa geçirildiği görülmüştür. Üretim süresinin ortalama %50'den fazlasında ne makina ne işgücü yararlı olarak kullanılabilmiştir. Çalışmanın bulguladığı sonuçlara göre pahalı makinelerin ucuz olanlara göre daha etkin kullanılmadığı, makina atölyelerinin hem makina hem de işgücünü düşük düzeyde kullandığı görülmüştür.

2.2. İMALAT POLİTİKASI

Bir işletmede imalat politikasının oluşturulmasında, üretimde oluşan problemlerin doğurduğu baskıyı azaltmak için bazı kararlar alınması gerekmektedir. Skinner, dar bir faaliyet alanına sahip bir fabrikada üretime odaklaşabilmek için şu beş faktörün gerektiğini belirtmektedir [6] :

1 - Proses teknolojileri olarak iki yada üç tane genel kabul görmüş, sadece bir tane kendini kanıtlamamış teknoloji kullanmak.

2 - Pazar talebi olarak fabrika başına fiyat, kalite ve temin süresini içeren bir yada iki çeşit ürün grubu belirlemek.

3 - Ürün-hacim grubu olarak takım işleme, taşıma ve sipariş miktarlarının karşılaştırılmasında tutarlı bir yaklaşım sağlayabilmek için karşılaştırılabilir ana ürün grupları oluşturmak.

4 - Kalite seviyeleri olarak kalite kontrolünde tutarlı bir yaklaşım oluşturmak için karşılaştırılabilir seviyeler kullanmak.

5 - İmalatın görevi olarak her hangi bir anda rekabetçi olabilmek için fabrikanın mükemmelleşeceği en çok iki alan belirlemek.

Bu beş anahtar faaliyet politikası Grup Teknolojisi'ni iyi bir şekilde tamamlamaktadır.

2.3. GRUP TEKNOLOJİSİNİN İMALAT SİSTEMLERİNDEKİ BASKIN MALİYETLERE ETKİSİ

Grup Teknolojisi'nin uygulanması ile imalat sistemlerinde maliyetlerin ne yönde etkileneceği belirlenmeden önce baskın olan maliyetlerin ele alınması uygun olacaktır.

Çoğu imalat sisteminde, direkt işçilik, proses içi stok, malzeme ve sermaye baskın maliyetler arasında sayılır. Bu maliyetlerden malzeme maliyetleri satış fiyatlarının %50'sini aşmasına rağmen, düşürülmesi amacına yönelik çok etkili yöntemler bulunmamaktadır. Başka bir deyişle malzeme maliyetlerini büyük ölçüde azaltmak mümkün değildir. Sermaye kaynaklı maliyet, sistem içindeki makinalardan yetersiz yararlanma oranlarıyla belirlenir. Şekil 2.1'de bu oranlara bir ölçü olmak üzere yapılan bir istatistiksel araştırmanın sonuçları verilmektedir [7].

Şekildeki teorik kapasite, makinanın belirli bir dönemde çalışabileceği en fazla zamanı göstermektedir. Örneğin, haftada 168 saat (7 gün X 24 saat). İşgörenin ve makinayı çalıştıracak enerjinin mevcut olduğu gerçek zaman, planlanmış makina kapasitesidir ve yapılan araştırmanın sonuçlarına göre, teorik kapasitenin %66'sını oluşturmaktadır. Ancak bu kapasite de 2. ve 3. vardiyaların kullanılıp kullanılmamasına göre değişir. Planlanmış makina kapasitesi, bazı nedenler yüzünden yararlı iş üretmek için kullanılamamaktadır. Bunlar :

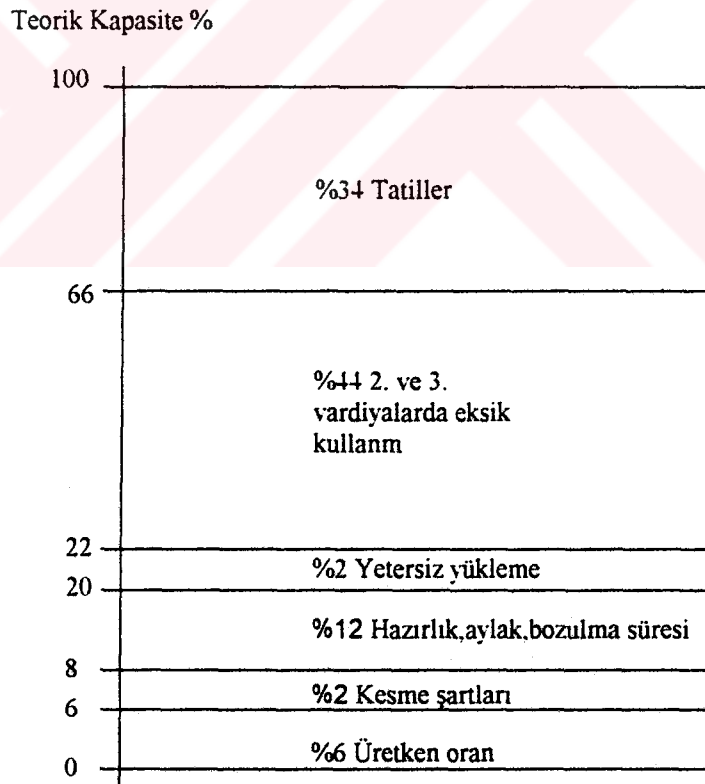
- 1) Yetersiz yükleme
- 2) Makina hazırlık zamanı
- 3) Boş geçen zaman
- 4) Bozulmalar ve kesintilerden doğan kayıp zaman

olarak sıralanabilir. Yetersiz yükleme, planlanmış makina kapasitesi ile ilgili makinaya atanmış tüm işlerin işlem zamanları arasındaki farktan ortaya çıkmaktadır. Makina aylak zamanları, malzeme, işgören veya siparişler için makina bekleme zamanlarından oluşmaktadır. Kesilmelerden doğan kayıplar ise operasyonel değişkenlerin en uygun

bir şekilde düzenlenmemesinden dolayı oluşmaktadır. Tüm bu anlatılan zamanların dışındaki zaman içerisinde, makinadan üretken olarak yararlanılabilmektedir.

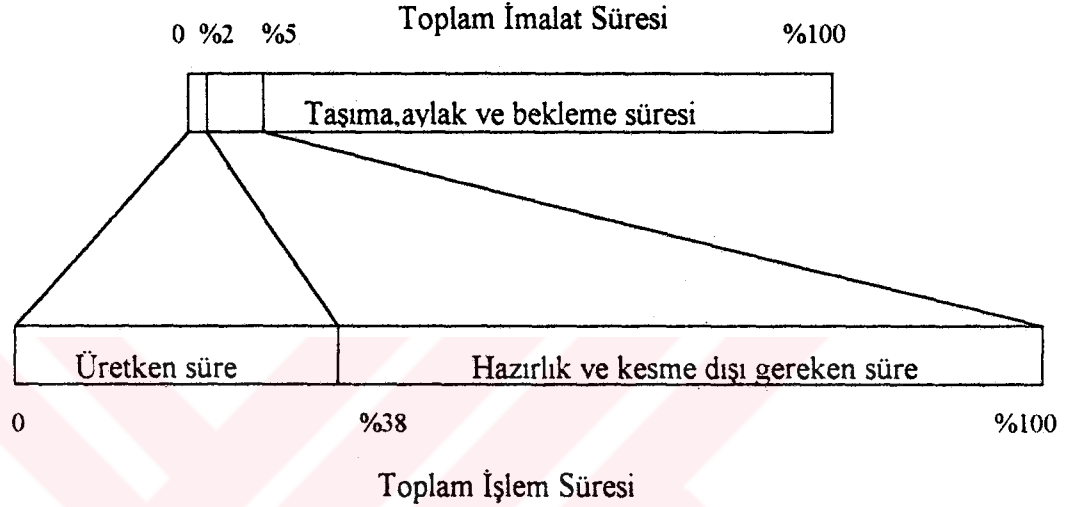
Grup Teknolojisi'nde oluşan yetersiz yükleme(daha düşük makina zaman verimleri) üretken orandan zaman çalmaktadır. Ancak klasik makinalarda toplam kapasitenin %10'u civarında olan hazırlık süresi ve makina aylak zamanlarından yapılacak yüksek oranda tasarruf, bunu fazlasıyla karşılayacaktır. Buna rağmen Grup Teknolojisi'nde sermaye kaynaklı maliyet açısından çok avantajlı bir durum söz konusu değildir. Aksine ek olarak doğacak tesis gereksinimi dolayısı ile fazladan maliyetler oluşabilir.

Sermaye kaynaklı maliyetten başka, ikinci en büyük denetlenebilir maliyet proses içi stok maliyetidir. Bu maliyet işlerin makina önünde bekleme zamanlarıyla orantılıdır. Şekil 2.2.'de sipariş tipi atölye sisteminde imal edilen bir iş parçasının toplam imalat



Şekil 2.1. Klasik Makinalardan Yararlanma Oranları

zamanı ile toplam işlem zamanı arasındaki ilişkisini göstermektedir. Şekildeki ilişkiye göre, iş parçası atölyede harcadığı zamanın %93'ünü kendisine hiçbir değer eklenmeden bekleyerek geçirmektedir. Dolayısıyla, bir iş parçasının bir makinadan diğer bir makineye hiçbir bekleme ve taşıma olmadan geçmesi, stok, yer ve bununla ilgili olarak ortaya çıkan maliyetlerde %93'lük bir düşüş sağlayacaktır. Bu hedefe yaklaşmak Grup Teknolojisi ile mümkün olabilmektedir. Bousher [7], Grup



Şekil 2.2. Sipariş Tipi Atölye Sisteminde Toplam İmalat Zamanının Analizi

Teknolojisi için ekonomik parti miktarı modeli içine proses içi stok maliyetlerini de dahil etmiştir. Bunu yaparken, proses içi stok maliyetlerini yukarıda bahsedilen %7'lik bir zamanın fonksiyonu olarak göstermiştir. Geliştirilen formül şu şekildedir.

$$N_b = \sqrt{\frac{2 \cdot C_2 \cdot D + D \cdot S_u \cdot (C_3 \cdot S_u \cdot R)}{C_3 \cdot (M + T_r \cdot R) + 2 \cdot C_3 \cdot D \cdot T_r \cdot (M + T_r \cdot R / 2)}} \quad (2.1)$$

Burada ;

N_b : Ekonomik parti miktarı

D : Yıllık iş parçası talebi

M : İş parçasının bir biriminin hammadde maliyeti

S_u : İş parçasının imalatla karşılaştığı tüm makinalar için toplam hazırlık zamanı

T_r : İş parçasının bir biriminin toplam işleme zamanı

R : Birim imalat zamanı için yüklenilmiş maliyet

C_2 : Herbir hazırlığın maliyeti

C_3 : Bir birim stoğu elde bulundurma maliyeti

Eşitliğin payındaki ilk terim hazırlık zamanından doğan maliyeti, paydadaki ilk terim ise marjinal birim imalat maliyetini göstermektedir. Eşitliğin her iki tarafındaki terimler ise, proses içi stoğun etkisini yansıtmaktadır. Bu eşitlik, sipariş tipi atölye sistemindeki proses içi stok maliyetleri için de bir göstergedir. Fakat bu sistemlerde imalat zamanları daha çok taşıma ve bekleme zamanlarından oluştuğu için, parti miktarının değişimi eşitlikteki gibi proses içi stok maliyetlerine bir etki yapmamaktadır. Diğer bir deyişle burada proses içi stoktaki önemli azalışlar, birincil dereceden taşıma ve bekleme zamanlarının azalışına bağlı kalmaktadır.

Etkin olarak tasarlanmış bir grup teknolojisi sisteminde ise taşıma ve bekleme zamanı hemen hemen ortadan kaldırıldığından, parti miktarının küçültülmesi, proses içi stok maliyetlerini eşitlikteki parametrelere bağlı olarak düşürmektedir. Ayrıca grup teknolojisinde klasik ekonomik parti miktarı yerine, yeni geliştirilmiş bu model kullanılırsa stokların maliyet minimizasyonu klasik modelle elde edilen parti miktarından çok daha az parti miktarı ile gerçekleştirilebilecektir.

Direkt işgücü maliyetlerinin durumu ise şöyledir. Bilindiği gibi genellikle her makinaya bir işgören atanmaktadır. Ancak grup teknolojisinde iş akışının basitleşmesinden dolayı çoğunlukla bir işgörenin birden fazla makinarya bakması uygun olabilmektedir. Bunun yanında endüstriyel robotlar da birden fazla makina için kullanılabilir. Bu nedenle direkt işgücü maliyeti grup teknolojisinde düşebilir. Burada ana sorun bu işgücünü makina girişimlerine meydan vermeyecek şekilde makinalara atamaktır.

Uygulamada değişkenlerin toplam maliyet değişimleri grup teknolojisi lehine gelişmektedir. Ancak bu değişimlerin olumlu veya olumsuz olması ile bunların değişim miktarları grup düzenlemesinin etkinliğine bağlı olacaktır.

2.4.GRUP TEKNOLOJİSİ'NDE KÜMELEME YÖNTEMLERİ

Grup Teknolojisi etkin üretim için kesin olarak ölçülebilir benzerliklere dayalı olarak parça ve makinaları, parça aileleri ve makina hücreleri olarak sınıflandırmayla ilgilenmektedir. Örneğin benzer parçaların gruplanması tasarım ve imalat aşamalarında benzerlik avantajından yararlanılmasını sağlayabilmektedir. Bir parça ailesi aynı makina için hazırlama, işleme, rotalama işlemlerini paylaşabilmekte; bu durum kitlesel imalatla ölçek ekonomisi sağlayabilmektedir. Benzer şekilde makinaları bir araya getirerek gruplamak hücreler arası gidiş gelişleri azaltmakta, malzeme taşıma maliyetlerinin minimize edilmesini bazen de ortadan kaldırılmasını sağlayabilmektedir. Bunlara ek olarak hazırlık ve imalat sürelerinde, tasarım değişikliğinde ve proses içi stoklarda azalma sağlanabilmektedir.

İki yada daha fazla makina, parça ailelerini işlemek üzere bir araya getirildiğinde, bu makinaların yaptığı işlemler hücresel imalat (İng: Cellular Manufacturing-CM) olarak adlandırılır ki bu terim Grup Teknolojisi ile genelde aynı anlamda kullanılmaktadır. Ancak belirtmeliyiz ki hücresel imalat, grup teknolojisinin bir uzantısı olmaktadır.

Makina - parça ailelerini oluşturmak için bugüne değin yapılan çalışmalar üç farklı yöntemle dayanmaktadır. Bunlardan birincisi görsel yöntem, ikincisi parça özelliklerine dayalı yöntem, üçüncüsü ise üretim prosesine dayalı yöntemdir [8] .

2.4.1. Görsel Belirleme Yöntemi

Bu yöntem parçaların geometrik yapılarını inceleyerek parça ailelerinin oluşturulmasına dayanmaktadır. Burada analist, kendi tecrübelerini kullanarak parçaları incelemekte ve uygun grupları oluşturmaktadır. Bu yaklaşım maliyet olarak ucuz olsa da hataya açık, tecrübe ve uzmanlığa dayalı, oldukça az sayıda parça dışında kullanılmayan bir yöntemdir. Bu yöntemin 2000 bileşenli makina-parça aileleri gruplama problemlerinde kullanıldığı, ancak bu ifadenin oldukça iyimser olduğu bildirilmektedir.

2.4.2. Parça Özelliklerine Dayalı Yöntemler

İkinci olarak verilen bu grupta yöntemi parça ailelerini oluşturmada parça özelliklerini kullanmaktadır. Parça kodlama ve sınıflandırma analizi(İng:Part Coding and Classifying Analysis-PCA) olarak da isimlendirilen bu yaklaşım parça özelliklerine sayısal ağırlıklar vererek belirli bir sınıflandırma tablosu kullanıp parça ailelerini oluşturmaktadır. Çoğu kodlama sisteminin tasarım özelliği ve yapısından dolayı PCA'ya dayalı sistemler grup teknolojisi literatüründe pek fazla yer almamaktadır. Bu tür sistemlerin yoğun işgücü gerektirmeleri dolayısıyla benimsenmedikleri; ancak kodlama sistemlerinin otomatize edilerek bu eksikliğin giderilebileceğini belirtilmektedir. Kusiak, kodlama ve sınıflandırma sisteminin uygulamasının zor ve masraflı olduğunu bildirmekte; Hyer, Wemmerlov ve Levuhis ise tersini iddia etmektedirler [9]. Örneğin, Hyer ve Wemmerlov'un yaptıkları çalışmaya göre incelenen şirketlerin % 62'si kodlama ve sınıflandırma sistemi kullanmaktadırlar. Onlara göre bu tür sistemlerin fazla tercih edilmemesinin sebebi PCA'ya dayalı gruplama algoritmalarındaki ilerlemelerin yavaş olmasıdır. PCA'daki sayısal ağırlıklı kodların bir benzerlik ölçütüne dönüştürülebileceklerini, böylece literatürdeki gruplama algoritmalarının kullanılabilir hale geleceğini belirtmektedirler.

PCA'ya dayalı sistemler geleneksel olarak tasarım yönelimli ve geometrik şekle dayalı olduklarından bileşenlerin çeşitliliğini azaltmak için çok uygun olmaktadır. Bazı PCA'ya dayalı sistemler, örneğin Opitz sistemi, üretime dayalı kodları destekleyici kodlar olarak birleştirmekte ve üretim planlamada da kullanılabilir hale getirmektedir. 1960-70'lerde parça özelliklerine dayalı sistemler grup teknolojisinin ana araçlarını oluşturmaktaydı. Benzer parçaları tasarım özelliklerine göre tanımlama ve gruplama kavramı ilk olarak Mitrofanov tarafından ortaya atılmıştır. Opitz ve Wiendhal daha sonraları üretim hücreleri kavramını; Opitz ise iş parçaları için anlaşılabilir bir kodlama ve sınıflandırma sistemini geliştirmiştir. Bu öncü çalışmalardan sonra parça gruplamayı kolaylaştırmak için SAGT ve MICLASS gibi sistemler de geliştirilmiştir. PCA'ya dayalı sistemler monokod, polykod ve hibrid kod yapıdadırlar. Ancak burada kod yapılarının detaylarına girilmeyecektir.

2.4.3.Üretim Prosesine Dayalı Sistemler

Üçüncü ve en çok ilgi çeken yöntem üretim prosesine dayalı analizlerdir. Bu yaklaşımın amacı ortak işlem özelliklerine sahip olan parçaları belirleyerek gruplar oluşturmaktır. Üretim Akış Analizi(İng:Production Flow Analysis-PFA) olarak da bilinen bu yöntem, parçalar ve onları işleyen makineler arasındaki ilişkileri belirlemek için rota kartlarını kullanmaktadır. Yöntem onu geliştiren Burbidge tarafından popüler hale getirilmiştir. Başlangıç verilerinin toplanmasından sonra bu yöntemde sırasıyla üç işlem uygulanmaktadır:

1 - Paketlerin Oluşturulması : Benzer prosesleri gerektiren parçalar sıralanır ve her bir parçaya bir paket numarası verilir. Son zamanlarda yapılan uygulamalarda bu adımdan vazgeçilmiş, paketler yerine her parça, PFA şemalarının oluşumunda tek tek kullanılmaya başlanmıştır.

2 - Üretim Akış Analizi Matrisi : Son ürünü oluşturan parçaların imalat yöntemi incelenerek hangi parçaların hangi makinalarda işlendiği belirlenip makina-parça matrisi oluşturulur. Bu matrisin tüm elemanları 1 ve 0 rakamlarını içerir. Eğer matrise ait herhangi bir $C(i,j)$ hücresi 1 ise bu durum i makinasının j parçasını işlediğini belirtir. Eğer bu değer 0 ise j parçasının i makinasında işlemediği anlaşılır.

3 - Analiz : Makina-parça matrisi makina ve parça gruplarının belirlenmesi için yeniden düzenlenir. Bu yeniden düzenleme işlemi özellikle büyük makina-parça matrisleri için oldukça zor ve subjektif bir yöntemdir. Bu zorluklarla birlikte, analiz sonucunda oluşturulan grupların ne derece kesin olduğu konusunda belirli bir temel kriter yoktur. Bu sorunu ortadan kaldırmak için bir takım sistematik yöntemler geliştirilmektedir. Bu yöntemler içinde en çok dikkat çekenler benzerlik katsayısına, sıralamaya ve matematiksel modellere dayalı yöntemlerdir. Tüm bu zorluklara karşın bu aşama gruplama işleminin en önemli işlemidir.

2.5.GENEL MODELLEME KRİTERLERİ

Bu aşama araştırmacının çözülecek problemi nasıl algıladığını, gerçeğe uygun olup olmadığını belirleme aşamasıdır. Algılama biçimi, problem formülize edilirken ve çözülürken kullanılacak varsayımları belirler. Literatürde bulunan çalışmaların bazılarının varsayımları hücresele imalatı genel olarak kapsarken bazıları belirli modellere özgü olmaktadır. Genel olarak varsayımlar kullanılan modellere göre şekillenmektedir. Örneğin matematiksel modeller çoğunlukla oluşturulacak grup sayısının, gruplanacak toplam makina ve parça sayısının bilindiğini varsaymaktadır. Aşağıdaki varsayımlar yapılacak araştırmalara bir ön temel olması açısından genel olarak dikkate alınmaktadır:

1 - Yerleşim Düzeni

- Düz Çizgi - Dikdörtgensel - Kare - Dairesel - Rasgele

2 - Hazırlık Süresi

- Yok - Var
- Sıralamaya Bağımlı - Sıralamadan Bağımsız

3 - Makina Kümeleme

- Tamamıyla Ayrılabilir Kümeler - Kısmen Ayrılabilir Kümeler
- Çift Makina Kullanma - Parçanın Dışarıya Yaptırılması
- Her ikisi

4 - Talebin Yapısı

- Biliniyor (Zamana Bağımlı Değil) - Bilinmiyor (Zamana Bağımlı)

5 - Planlama Dönemi Sonsuz

6 - Parti Büyüklüğü

- Sabit ve Her Makina Hücresine Eşit Dağıtılmış - Değişken

7 - Üretim Akış Politikası

- Geri Dönüş İzni Verilir - Geri Dönüş İzni Verilmez

8 - Her Makina Tipinin Sayısı

- Bir - Birden Fazla

9 - İşlem Süreleri

- Biliniyor (Deterministik)

- Değişken (Probabilistik)

2.6. GT'DE ÜRETİM AKIŞ ANALİZİ İLE HÜCRE TASARIMI

Üretim Akış Analizi'ni ilk tanıtan Burbidge olmuştur. Burbidge'in bu analiz için yapmış olduğu yaklaşımda uygulama yönünden iki sakınca sözkonusu olmaktadır.:

a - Hem fabrika akış analizinin hem de grup analizinin sezgisel yöntemlere dayanmış olması .

b - Grup analizinde parça-makina matrisinin büyük olması durumunda çözüme ulaşmanın her zaman mümkün olamaması. Üretim akış analizinin bu sakıncalı yönlerini ortadan kaldırmak için birçok yöntem geliştirilmiştir.

2.6.1. Benzerlik Katsayılarını Temel Alan Çalışmalar

Benzerlik katsayısı yaklaşımı ilk olarak Sokal ve Sneath [10] tarafından geliştirilmiş ve Mc Auley [11] bu yaklaşımı üretim akış analizinde kullanmayı önermiştir. Metodun esası her bir çift makina arasındaki karşılıklı ilişkinin "Benzerliğini" ölçmek ve bu benzerlik ölçü veya katsayılarına dayanarak makinaları gruplamaktır.

Mc.Auley benzerlik ya da uyumluluk katsayısını şöyle tanımlamıştır:

“ Makina çiftlerine ait benzerlik katsayısı herhangi bir makina çiftinde her iki makinaya da uğrayan parça sayısının ,hem her iki makinaya, hem de sadece bir makinaya uğrayan parçaların sayısına ilişkin bir rakama oranıdır. “

Bu oranı formüle etmek için makina-makina kombinasyon matrisi kullanılır. Bu matriste i ve j iki ayrı makina göstermek üzere X matris elemanları;

X_{ij} : i ve j makinalarının her ikisine de uğrayan toplam parça sayısı

X_{ii} : sadece i makinasına uğrayan toplam parça sayısı

X_{ij} : sadece j makinasına uğrayan toplam parça sayısı şeklinde tanımlanır. Buna göre benzerlik katsayısının formülü ise şu şekilde ifade edilir.

$$S_{ij} = X_{ij} / (X_{ii} + X_{jj} - X_{ij}) \quad (2.2)$$

Makinaları grulamada Mc.Auley Tek hatlı Kümelendirme Analizini kullanmıştır. Bu yöntem önce en yüksek benzerlik katsayıları ile ilgili makinaları bir araya kümelendirmekte ve ardından azalan büyüklüklere göre hiyerarşik bir yapıda bu işleme devam etmektedir. Örnek olarak Şekil 2.3'deki bir makina-makina kombinasyon matrisini ele alalım.

		MAKİNALAR				
		M ₁	M ₂	M ₃	M ₄	M ₅
M ₁	2		1		2	
M ₂			2		2	
M ₃				2		2
M ₄					3	
M ₅						3

Şekil 2.3. Makina-Makina Kombinasyon Matrisi

Bu matristeki her bir makina çifti için benzerlik katsayıları hesaplanarak elde edilen değerler Tablo 2.1.'deki gibi toplanabilir.

Tablo 2.1. Örnek Matris İçin Her Bir Makina Çiftine Ait Benzerlik Katsayıları

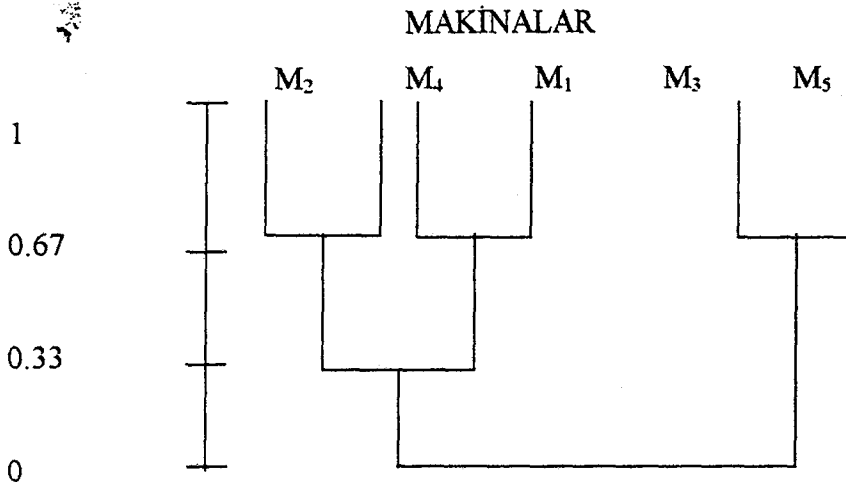
		MAKİNALAR				
		M ₁	M ₂	M ₃	M ₄	M ₅
M ₁			0.33	0	0.67	0
M ₂				0	0.67	0
M ₃					0	0.67
M ₄						0
M ₅						

Benzerlik katsayıları elde edildikten sonra tek hatlı kümelendirme analizine geçilir. Bunun için önce makinaların diyagram üzerinde gösterimi olan dendogram (ağaç diyagramı) çizilir. Şekilde örneğe ait ağaç diyagramı görülmektedir. Diyagramın sol kenarında 0'dan 1'e kadar ölçülendirilmiş benzerlik katsayısı skalası vardır. Diyagramdaki ağacın en tepe dallarında makinalar kodlanmıştır. Makina kod düzeni dalların üst üste gelmelerini engelleyici şekilde olmalıdır. Dalların birleşme yerinin benzerlik skalasına karşılık gelen değeri sözü edilen dallara ait makina grubunun benzerlik derecesini gösterir.

Örnek problemde üç alternatif mevcuttur. Birincisi 1 benzerlik katsayısıyla her makinayı bir grupta tasarlamak, ikincisi 0.67 benzerlikle M_1 , M_2 ve M_4 makinalarını bir grupta; M_3 ve M_5 makinalarını diğer bir grupta olmak üzere iki gruba ayırmaktır. Üçüncü çözüm ise tüm makinaları 0.0 benzerlikle bir arada toplamaktır. Aynı bir çözüm olarak 0.33 benzerlikle M_1 ve M_2 makinalarını bir gruba almak düşünülebilir. Ancak bu çözüm bizi 0.67 benzerlik düzeyindeki çözüme götürür ve 0.33, 0.67'ye göre daha düşük olduğundan ele alınmaz. Birinci ve üçüncü alternatifler birer uç çözüm verdiğinden, GT için üçüncü seçenek aranan çözüm olmaktadır. Bu arada uygulama yerlerine bağlı olarak birçok benzerlik katsayısı elde edilebilir. Uygun çözüm için bir alt benzerlik katsayısı -eşik değeri (T)- seçilebilir ve bu değerin üstündeki çözümler üzerinde durulur. Burada makina grupları elde edildikten sonra grup düzenleme için son aşama, iş parçalarını ilgili gruba atamaktır. Mc.Auley'in yaklaşımı bazı konularda yetersiz kalmaktadır. Bunların başlıcaları şunlardır [11] :

a) $X_{ij} \gg X_{ii}$ olması halinde, gruplamada yanlış bir sonuca gidilebilir. Örneğin, $X_{ii}=100$, $X_{ij}=100$ ve $X_{ij}=1000$ olduğu varsayılırsa, bu durumda $S_{ij}=0.1$ olarak hesaplanır. Bu değer küçük bir benzerlik katsayısıdır. Dolayısıyla i ve j makinalarının ayrı gruplarda bulunması gerektiği düşünülebilir. Oysa $X_{ii}=100$ ve $X_{ij}=100$ değerleri i ve j arasında doğrudan bir ilişki olduğunu göstermektedir.

b) Bu yaklaşım sonucu birden fazla olan bir makina tipinin, birden fazla hücreye atanması mümkün olmamaktadır.



Şekil 2.4. Dendrogram (Ağaç Diyagramı)

c) Grupların oluşturulmasında T değeri önemli olmasına rağmen ancak keyfi bir şekilde seçilmesi söz konusudur. Mc.Auley'den sonra benzerlik katsayısına dayanan diğer bir yaklaşım Rajagopalan ve Batra tarafından önerilmiştir [12]. Rajagopalan ve Batra gruplama için 'Grafik Teorisi' uygulamasını geliştirmişlerdir. Grafiğin tepe noktalarını makinalar, kenarlarını ise benzerlik katsayıları oluşturmaktadır. Burada T'nin bulunması daha sistematik bir şekilde yapılmaktadır. Büyük T değerleri için grafik seyrek ve dağınık olmakta, sonuçta hücre oluşturma işlemi basitleşmektedir. Ancak oluşacak hücrelerde, hücreler arası akış yoğunluğu artacak dolayısıyla GT'nin avantajı azalacaktır. Küçük T değerleri ise grafik yoğunluğunu arttıracığından (kenar sayısının artması) çözümü zorlaştırmaktadır. Bahsedilen yaklaşımla uygun bir T değeri uygulanan örneğe göre bilgisayar desteğinde elde edilmektedir. Yaklaşımda elde edilen grafiğin, grafik teorisine uygun olarak bölünebilmesi sayesinde çeşitli gruplar oluşturulmaktadır. Oluşturulan farklı gruplarda aynı makina tipleri bulunabilmektedir. Burada grafik oluştururken yine tek bir benzerlik katsayısı kullanıldığından, de Beer ve de Witte tarafından belirtilen sakıncalı durum ortadan kalkmamıştır.

De Beer ve de Witte çeşitli makina tiplerinin farklı gruplarda bulunmasını sağlamak amacıyla, makinaları birincil, ikincil ve üçüncül tip makinalar olmak üzere üçe ayırmışlardır. Birincil tip makina sadece bir makina mevcutsa veya aynı tipe ait makinaların sadece bir hücrede yer alması gerektiğinde; ikincil tip makina bir tipten

mevcut 2 ila 4 makina, 2 ila 4 hücreye atanacaksa; üçüncül tip makina ise tüm hücrelere tahsis edilecek yeterli sayıda makina mevcut ise ortaya çıkmaktadır. Bu yaklaşımı de Witte daha da geliştirerek gruplamada bir yerine üç farklı benzerlik katsayısı kullanmıştır. Böylece belirtilen sakıncalı durum ortadan kaldırılmaya çalışılmıştır [13].

2.7. PARÇA MAKİNA MATRİSİNDE DİYAGONAL DİZİLİ GRUPLAR OLUŞTURAN YÖNTEMLER

2.7.1. Bağı Enerjisi Yöntemi

Bağı enerjisi olarak bilinen genel bir gruplama analizi tekniği, Mc Cormick tarafından geliştirilmiştir. King bunu üretim akış analizinde uygulamıştır [14]. Bu yöntem uygulanırken makina-makina matrisi yerine parça-makina matrisinden yararlanılır. Daha önceki örnekle ilgili olarak Şekil 2.5'de parça-makina matrisi gösterilmektedir.

	MAKİNALAR					SATIR BAĞ ENERJİSİ
	M ₁	M ₂	M ₃	M ₄	M ₅	
P ₁			1		1	
P ₂		1		1		
P ₃	1	1		1		1
P ₄			1		1	
P ₅	1			5		
P ₆					1	
SÜTUN BAĞ ENERJİSİ		1		1		3

Şekil 2.5. Parça-Makina Matrisi ve Sahip Olduğu Bağ Enerjisi

Yöntemde sözü edilen bağ, parça-makina matrisinde bitişik satır ve sütun elemanlarının sayısıdır. Bu sayının artması ise bağ enerjisinin büyüklüğünün artması demektir. Matristeki elemanların değerlendirilmesinde aşağıdaki alternatifler mevcuttur.

1) Satır elemanları yan yana (X_{ij} , $X_{i,j+1}$) aşağıdaki olası seçenekleri doğurur:

Tablo 2.2. Satır Elemanları Seçenekleri

Alternatif	X_{ij}	$X_{i,j+1}$	Bağ Enerjisi
1	0	0	0
2	0	1	0
3	1	0	0
4	1	1	1

2) Sütun elemanları yan yana (X_{ij} , $X_{i-1,j}$) yine yukarıdaki örneğe benzer olası seçenekleri oluşturur.

Tablo 2.3. Satır Elemanları Seçenekleri

Alternatif	X_{ij}	$X_{i-1,j}$	Bağ Enerjisi
1	0	0	0
2	0	1	0
3	1	0	0
4	1	1	1

Şekilde elde edilen bağ enerjileri de aynı şekilde değerlendirilmiştir. Gruplamanın başarı derecesi bağ enerjisinin gücü ile vurgulanmaktadır. Örnek matriste toplam bağ enerjisi 3'tür. En iyi çözüm en yüksek bağ enerjisini elde eden çözümdür. Problem, gerçekte bir quadratik atama problemi biçimindedir. Bu nedenle problemin boyutunun büyümesi ile hesaplama zamanı üssel olarak artmaktadır. Sonuç olarak, yöntem büyük boyutlu problemler için kısıtlı kalmaktadır. Bu nedenle ayrıca bir sezgisel yöntem geliştirilmiş ve böylece ikincil en iyi çözümler elde edilebilmiştir.

Aşağıdaki Şekil 2.6'da bu yöntemin uygulanmasıyla elde edilen en yüksek bağ enerjisine sahip, en iyi çözümü gösteren parça makina matrisi verilmiştir. Görüldüğü gibi 1, 4, 6 parçaları ile M_3 , M_5 makinaları; 3, 2, 5 parçaları ile M_4 , M_2 , M_1 makinaları, parça makina matrisinde diyagonal dizili grupları oluşturmaktadır.

	MAKİNALAR					SATIR BAĞ
	M ₅	M ₃	M ₄	M ₂	M ₁	ENERJİSİ
P ₁		1	1			1
P ₄	1		1			1
P ₆	1					
P ₃			1	1	1	2
P ₂			1	1		1
P ₅			1		1	
SÜTUN						
BAĞ	2	1	2	1		11
ENERJİSİ						

Şekil 2.6. Sonuç Parça-Makina Matrisi ve Sahip Olduğu Bağ Enerjisi

2.7.2.Derece Sıralamasıyla Kümelendirme Yöntemi

Diyagonal dizili grupların türetilmesi amacıyla tasarlanan diğer bir teknik, King tarafından geliştirilmiştir [1]. King'in yöntemine ait algoritma normal olarak parça-makina matrisinden başlar. Ancak izlenen yol tekrarlı olduğundan, matrisin düzenlenmiş herhangi yeni bir durumundan başlamak da mümkündür. Bu yöntemde parça-makina matrisine ait her satır ikili numaralama sistemine göre ağırlıklandırılır. Ayrıca bu ağırlıkların ondalık değerleri hesaplanır. Elde edilen bu ondalık eşdeğerlere göre satırlar yeniden sıralanır. Daha sonra aynı işlem sütunlar için de tekrarlanır. Hem satırlarda hem de sütunlarda azalan derece sıralaması elde edilinceye kadar işlemlere devam edilir. Amaca ulaşıldığında en son olarak elde edilen satır-sütun kombinasyonuna ait parça-makina matrisi aranan matristir. Aşağıdaki Şekil 2.5.'teki örnek için uygulanan yöntem sonuçları şekil 2.7, 2.8 ve 2.9'de görülmektedir. Ancak burada şu önemli durum belirtirmelidir. Endüstriyel örnekler çok daha büyük matrisler gerektireceğinden çözüme tekrarlı olarak bu kadar çabuk ulaşılmayacağı gibi, bazı özel parçaların ayıklanıp oluşturulacak grupların dışına alınarak grup sınırlarının elde edilmesi her zaman mümkün olmamaktadır. Bununla birlikte algoritmanın geliştirilmesine yönelik, özellikle darboğaz makineler için analizin daha belirgin sonuçlar vermesini sağlayıcı çalışmalar da yapılmaktadır.

İKİLİ AĞIRLIK	2^4	2^3	2^2	2^1	2^0	ONDALIK EŞDEĞER	DERECE SIRASI
	MAKİNALAR						
	M_1	M_2	M_3	M_4	M_5		
P_1			1		1	5	4
P_2		1		1		10	3
P_3	1	1		1		26	1
P_4			1		1	5	5
P_5	1			1		18	2
P_6					1	1	6

Şekil 2.7. Parça-Makina Matrisinin Satırlarının İkili Sisteme Göre Değerlendirilmesi

İKİLİ AĞIRLIK		MAKİNALAR					ONDALIK EŞDEĞER	DERECE SIRASI
		M_4	M_1	M_2	M_5	M_4		
2^5	P_3	1	1		1			
2^4	P_5	1			1			
2^3	P_2		1		1			
2^2	P_1			1		1		
2^1	P_4			1		1		
2^0	P_6					1		
ONDALIK EŞDEĞER		48	40	6	56	7		
DERECE SIRASI		2	3	5	1	4		

Şekil 2.8. Satırların Düzenlenmesi ile Elde Edilen İkinci Matris ve Bu Yeni Matrisin Sütunlarının İkili Sisteme Göre Değerlendirilmesi.

		2^4	2^3	2^2	2^1	2^0		
İKİLİ		MAKİNALAR					ONDALIK	DERECE
AĞIRLIK		M_4	M_1	M_2	M_5	M_3	EŞDEĞER	SIRASI
2^5	P_3	1	1	1			26	1
2^4	P_5	1	1				24	2
2^3	P_2	1		1			20	3
2^2	P_1				1	1	3	4
2^1	P_4				1	1	3	5
2^0	P_6				1		2	6
ONDALIKLI								
EŞDEĞER		56	48	40	7	6		
DERECE								
SIRASI		1	2	3	4	5		

Şekil 2.9. Elde Edilen En Son Matris ve Bu Matrisin Satır ve Sütunlarının İkili Sisteme Göre Değerlendirilmesi.

2.7.3. İş Parçası ve Makina İstatistikleri Kullanarak Hücre Tasarımı

Mansour ve Leonard tarafından, hücre tasarımı için iş parçası ve makinaların bazı özelliklerine ait istatistik verilerin kullanılmasıyla çok hızlı ve pratik bir yaklaşım geliştirilmiştir. Bu yaklaşım, önce makina gruplarını oluşturmakta, daha sonra iş parçalarını makina gruplarına, parça ailesi oluşturmaya gerek duymadan atamaktadır. Genel olarak iş parçası istatistiği iki şekilde ortaya çıkar. Birincisi iş parçalarının geometrik özelliklerine dayanır. İkincisi teknolojik özellikleri ile ilgilidir. Geometrik özelliklere dayanan istatistiki bilgiler şunlar olabilir [15]:

- Dönel veya dönel olmayan parçaların tekrar sayısı
- Boyutların tekrarı
- Uzunluk/çap oranlarının tekrarı
- Kör delikli, boydan boya düz delikli, kademeli delikli gibi iş parçalarının geometrik özelliklerinin tekrarı.

Opitz tarafından Almanya'da ve Koloc[16] tarafından Çekoslavakya'da yapılan istatistiksel çalışmalar, birçok endüstriyel firmada, iş parçalarının geometrik özelliklerine göre benzer niteliklere sahip olduklarını göstermiştir. Örneğin her iki ülkede, genel olarak dönel parçaların, dönel olmayan parçalara oranı 70/30 olduğu bulgulanmıştır. Teknolojik özellikler, iş parçası çapı ve geometrisi gözönüne alınmadan, parçaların yüzde kaçına hangi işleme yöntemlerinin uygulanacağını belirlemektedir. İş parçalarının teknolojik özelliklerine ve kullanılan makinalara ait İngiltere'de metal işleme endüstrilerinde yapılan istatistiksel çalışmalarda işleme yöntemleri ile işleme tesisleri arasında ilginç benzerlik ilişkileri olduğu görülmüştür. Bu ilişkiler, Mansour ve Leonard tarafından dönel parçaların dönel olmayan parçalara oranının yaklaşık 70 / 30 olduğu firmalarda hesaplanmış ve aşağıdaki sonuçlar elde edilmiştir.

Dönel Parçalar İçin

Tornalama : Frezeleme : Matkaplama

3 : 1 : 3

Dönel Olmayan Parçalar İçin

Yüzey İşleme : Matkaplama veya İçTornalama

1 : 1

Bu sonuçlar yardımıyla özellikle genel amaçlı takım tezgahlarından freze ve matkap tezgahlarının, hangi oranlarda dönel ve dönel olmayan parçaların işlenmesine katılacağı bulunur. Örneğin, dönel parçalar için tornalamanın frezelemeye oranının 3/1 olduğu gözönüne alınarak torna ve freze tezgahları bu oranlara göre atanır. Mansour ve Leonard, bu istatistiksel sonuçları GT hücresi oluşturmak üzere, Manchester'da fonksiyonel düzenlenmiş tesislere sahip bir türbin imalat firmasına uygulamışlardır. Uygulama için firmaya ait şu bilgileri toplamışlardır.

- Takım tezgahı listesi
- Dönel parçaların dönel olmayanlara oranı
- VUOSO sınıflandırma sisteminde her bir VUOSO sınıfına giren farklı iş parçaları yüzdeleri.

Uygulamada önce yukarıda verilen istatistiksel sonuçlar yardımıyla dönel ve dönel olmayan iş parçalarına ait takım tezgahı listesi belirlenmiştir. İkinci adımda her bir VUOSO sınıfındaki iş parçası yüzdesinden hareketle, yüzde parça sayısı ile toplam takım tezgahı sayısı çarpılarak her bir sınıfa girecek takım tezgahı sayısı elde edilmiştir. Son adım, belirli VUOSO sınıflarını hücre oluşumunda esas alarak, iş parçalarına göre dağıtılmış takım tezgahları için makina hücrelerini firmaya özgü bazı düzenlemelerle oluşturmaktır.

VUOSO sınıfları için varolan iş parçası geometrik özelliklerine ait istatistiksel sonuçlar yardımıyla, yukarıda anlatılan hücre oluşturma işlemi çok daha hızlı ve etkin bir şekilde gerçekleştirilebilmektedir.

Sonuç olarak, anlatılan yöntemin başarısının, uygulamanın yapıldığı firmanın daha önceden geliştirilmiş bu istatistiksel sonuçlara uygun bir yapıya sahip olup olmamasına bağlı olduğu görülmektedir.

BÖLÜM 3

TAMSAYILI DOĞRUSAL PROGRAMLAMA

3.1.GİRİŞ

Tamsayılı Doğrusal Programlama, bilinen doğrusal programlama modelinin özel bir halidir. Burada karar değişkenlerinin sürekli değil kesikli olduğu varsayımıyla çözüme ulaşılmaya çalışılır. Yani karar değişkenlerinin tamsayı değerler alması istendiğinde Tamsayılı Doğrusal Programlama modelleri kullanılır. Bu tür problemler değişkenlerin istenen değerlerine göre "Tüm Tamsayı" yada "Karışık Tamsayı" problemleri olarak adlandırılır.

Doğrusal programlamada kullanılan klasik simplex yöntemi, atama problemleri gibi bazı özel problemlerde değişkenlerin çözüm değerlerini pozitif tamsayılar olarak verse de, diğer problem türleri için tamsayı değerler elde edilmesi kesin değildir. Simplex yöntem ile elde edilen kesirli değerlerin doğrudan tamsayı değerlerine yuvarlatılmasıyla elde edilen çözüm değerleri de çözümün optimallikten uzaklaşmasına neden olmaktadır. Bazı durumlarda bu tür bir yuvarlama sonucu elde edilen çözüm değerlerinin oluşturduğu sonucun gerçek optimal değerden sapma derecesi o kadar büyük olabilmektedir ki bu durum mümkün olmayan çözüme neden olmaktadır.

Optimal tamsayı değerlerinin elde edilmesine yönelik sistematik yöntemler bazı araştırmacıların ilgi konusunu oluşturmuştur. 1958 yılında R.E.Gomory, "Tüm Tamsayı" ve "Karışık Tamsayı" problemlerine uygulanabilecek "Kesme Düzlemi Algoritmaları" 'nı geliştirmiştir. Bu algoritmalar sınırlı sayıda iterasyon ile optimal tamsayı çözüme ulaşılmasını sağlamaktadır. Ana düşünce bilinen dual simplex algoritmasını kullanarak, çözüm alanını istenen sonuca göre belirlemeye yarayan ikincil kısıt denklemleri oluşturmaktır. Optimal olmayan çözüme ikincil kısıt denklemi

eklenmesiyle, çözüm düzlemi kesilmiş ya da sınırlandırılmış olur. Bu nedenle algoritmanın adı Kesme Düzlemi Algoritması olarak adlandırılmıştır. Optimal çözüme ulaşılan kadar ikincil kısıt denklemlerinin eklenmesiyle problem çözülmektedir. Bu çalışmada kullanılan yöntem Dal - Sınır yönteminin geliştirilmiş bir şekli olduğundan Kesme Düzlemi Algoritması'nın detaylarına inilmeyecektir.

"Tüm Tamsayı" ve "Karışık Tamsayı" problemlerinin çözümüne yönelik diğer bir yaklaşım da tüm mümkün tamsayı değerlerinin tek tek denenip, bu çözümler içinde en iyi olanın bulunmasıdır. Bu yaklaşım Dal - Sınır tekniğinin geliştirilmesini sağlamıştır. A.H.Land ve A.G.Doig, 1960 yılında "Tüm Tamsayı" ve "Karışık Tamsayı" problemlerine uygulanabilecek Dal- Sınır tekniğini geliştirmişlerdir. Egon Balas 1965 yılında sadece 0-1 değerleri alması istenen değişkenleri içeren problemlere uygulanabilecek, seçeneklerin tek tek incelemesine dayalı bir algoritma geliştirmiştir. Bu tez çalışmasında oluşturulan matematik modelin değişkenleri sadece 0-1 değerlerini alabileceği için , Egon Balas tarafından geliştirilen ve Balas Algoritması [17] olarak adlandırılan ve "Tüm Tamsayı" problemlerine uygulanabilen algoritma detaylarıyla anlatılmıştır.

3.2.BALAS ALGORİTMASI

Bu algoritma değişkenler sadece "Evet-Hayır" ya da "0-1" durumu alabildiğinde uygulanmaktadır. Örneğin bir imalat sisteminde grup teknolojisi uygulaması sırasında makina ve parça aileleri oluşturulurken j. makina k. hücreye atanırsa $X_{jk}=1$ olarak ifade edilir. Aksi halde $X_{jk}=0$ olur. Diğer bir deyişle j.makina k. hücreye atanmamıştır.

Balas Algoritması optimal fakat mümkün(feasible) olmayan bir çözüm elde edilmesiyle başlar. Bunun için amaç fonksiyonu önce minimizasyon haline, tüm kısıtlar ise sıfırdan küçük hale getirilir. Sonra artık değerlerin kısıt denklemlerine eklenmesiyle kısıtlar eşitlik haline getirilir.Başlangıç çözümde optimal ve mümkün çözüm elde ediliyorsa algoritmanın uygulanmasına gerek kalmaz. Eğer optimal çözüm elde edilse de artık değerlerden en az birinin değeri negatif olursa bu durumda mümkün(feasible) olmayan

çözüm durumu ortaya çıkar. İşte bu durumda algoritmanın adımları uygulanarak hem optimal hem de mümkün çözüme ulaşılmaya çalışılır.

Bir doğrusal programlama modelini ele alalım:

$$\max(\min) Z = \sum C_j X_j \quad j=1, \dots, n$$

$$\sum a_{ij} X_j \geq b_i \quad i=1, \dots, m$$

$$X_j=0 \text{ ya da } X_j=1$$

Görüldüğü gibi amaç fonksiyonu eğer maksimizasyon ise eşitliğin her iki tarafı da -1 ile çarpılarak fonksiyon minimizasyon haline getirilir. Kısıt fonksiyonlarında büyük eşit olan varsa bu kısıt da -1 ile çarpılarak küçük eşit haline dönüştürülür. Böylece modelimizin amaç fonksiyonu minimizasyon ve kısıtlar küçük eşit haline geldi. Şimdi her bir kısıt fonksiyonuna artık değerleri ifade eden ve değeri daima pozitif olan S değişkenleri ilave edilerek tüm kısıtlar sıfıra eşitlenir. Böylece modelimiz şu şekle dönüşür:

$$\min Z = \sum C_j X_j \quad j=1, \dots, n$$

$$\sum a_{ij} X_j + S_i = b_i \quad i=1, \dots, m$$

$$X_j=0 \text{ ya da } X_j=1$$

$$S_i \geq 0$$

Burada

C_j : Amaç fonksiyonundaki katsayılar

X_j : Karar değişkenleri

a_{ij} : Kısıt denklemlerindeki X_j 'lere ait teknik katsayılar

b_i : Kaynak değerleri

S_i : Artık(Slack) değerler anlamına gelmektedir.

Amaç fonksiyonu minimizasyon şeklinde ifade edildikten sonra, negatif C_j katsayılarına sahip X_j değişkenlerine aşağıdaki şekilde bir dönüştürme işlemi uygulanır:

$$X' = \left\{ \begin{array}{ll} X_j & , C_j \geq 0 \text{ ise} \\ 1 - X_j & , C_j < 0 \text{ ise} \end{array} \right\}$$

Örneğin amaç fonksiyonu

$$\min Z = 3X_1 - 2X_2 - 5X_3$$

ise uygun dönüştürme işlemi ile bu amaç fonksiyonu

$$\min Z = 3X'_1 - 2(1 - X'_2) - 5(1 - X'_3)$$

$$\min Z = 3X'_1 + 2X'_2 + 5X'_3 - 7$$

halini alır.

X_2 ve X_3 değişkenleri için amaç fonksiyonunda yapılan bu dönüşüm kısıt fonksiyonlarındaki değişkenlere de uygulanarak yeni bir formülasyon oluşturulur. Model çözüldükten sonra X'_j değişkenleri yeniden X_j haline dönüştürülerek gerçek değerler elde edilir.

Başlangıç seviyesinde tüm X'_j değişkenleri 0 değerinde olduğundan $Z_0 = 0$ olacaktır. Burada Z_0 başlangıç seviyesindeki amaç fonksiyonunun değerini göstermektedir. Başlangıç çözümde tüm $X'_j = 0$ ve tüm $S_i = b_i$ olacaktır. Burada b_i değerlerinden en az birinin değeri dönüştürme işlemi sonucunda negatif olursa optimal fakat mümkün olmayan çözüm oluşur, Bu mümkün olmayan çözüm durumu algoritmanın ilerlemesinde anahtar rolü oynamaktadır. Şimdi bir örnekle algoritmanın nasıl işlediğini adım adım inceleyelim:

$$\max Z = 3X'_1 + 2X'_2 - 5X'_3 - 2X'_4 + 3X'_5$$

Kısıtlar

$$X'_1 + X'_2 + X'_3 + 2X'_4 + X'_5 \leq 4$$

$$7X'_1 + 3X'_3 - 4X'_4 + 3X'_5 \leq 8$$

$$11X'_1 - 6X'_2 + 3X'_4 - 3X'_5 \geq 3$$

$$X'_j = 0 \text{ veya } X'_j = 1 \quad \text{tüm } j \text{ değerleri için}$$

Problemi önce minimizasyon haline getirilip kısıtların hepsi küçük eşit yapılacaktır. Bunun için amaç fonksiyonunu ve üçüncü kısıt fonksiyonunu -1 ile çarpalım. Böylece problem şu şekle dönüşür.

$$\min G = -3X_1 - 2X_2 + 5X_3 + 2X_4 - 3X_5$$

Kısıtlar

$$X_1 + X_2 + X_3 + 2X_4 + X_5 \leq 4$$

$$7X_1 + 3X_3 - 4X_4 + 3X_5 \leq 8$$

$$-11X_1 + 6X_2 - 3X_4 + 3X_5 \leq -3$$

$$X_j = 0 \text{ veya } X_j = 1 \quad \text{tüm } j \text{ değerleri için}$$

Amaç fonksiyonundaki X_1 , X_2 ve X_5 değişkenlerine ait C_j katsayıları negatif olduğundan

$$X_j = \begin{cases} X_j & , j = 3, 4 \\ 1 - X_j & , j = 1, 2, 5 \end{cases}$$

dönüşümü uygulanarak problem şu hale gelir.

$$\min G = -3(1-X_1) - 2(1-X_2) + 5X_3 + 2X_4 - 3(1-X_5)$$

Kısıtlar

$$(1-X_1) + (1-X_2) + X_3 + 2X_4 + (1-X_5) \leq 4$$

$$7(1-X_1) + 3X_3 - 4X_4 + 3(1-X_5) \leq 8$$

$$-11(1-X_1) + 6(1-X_2) - 3X_4 + 3(1-X_5) \leq -3$$

$$X_j = 0 \text{ veya } X_j = 1 \quad \text{tüm } j \text{ değerleri için}$$

Problem yeniden düzenlenip kısıt fonksiyonlarına artık S_j değerleri eklenerek algoritma için uygun hale getirilir ve şöyle olur:

$$\min G = 3X_1 + 2X_2 + 5X_3 + 2X_4 + 3X_5 - 8$$

Kısıtlar

$$-X_1 - X_2 + X_3 + 2X_4 - X_5 + S_1 = 1$$

$$-7X_1 + 3X_3 - 4X_4 - 3X_5 + S_2 = -2$$

$$11X_1 - 6X_2 - 3X_4 - 3X_5 + S_3 = -1$$

$$X_j = 0 \text{ veya } X_j = 1 \quad \text{tüm } j \text{ değerleri için}$$

Amaç fonksiyonundaki -8 değeri çözüm aşamasında dikkate alınmaz. Mümkün çözüm bulunduğundan sonra optimal çözüm hesaplanırken dikkate alınır. Bütün değerler tablo olarak aşağıdaki şekilde gösterilmiştir. Başlangıçta bütün X_j değerleri 0 olduğu için $\min G = 0$ olur. Ancak artık değerler $(S_1^0, S_2^0, S_3^0) = (1, -2, -1)$ olacaktır. Burada üst indis olan 0 değeri başlangıç iterasyon olduğunu göstermektedir. Problemin çözümünde n adet X değişkeninin hangilerinin 0 hangilerinin 1 olması gerektiği bulunacaktır. Yani 2^n adet çözüm olasılığının test edilmesi gerekecektir.

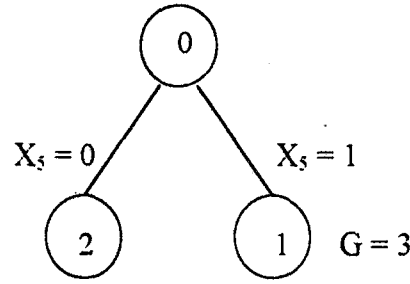
Tablo 2.4 Tamsayı Programlama Probleminin Katsayı Tablosu

X_1	X_2	X_3	X_4	X_5	S_1	S_2	S_3	
3	2	5	2	3	0	0	0	G
-1	-1	1	2	-1	1	0	0	1
-7	0	3	-4	-3	0	1	0	-2
11	-6	0	-3	-3	0	0	1	-1

Şimdi algoritmanın diğer adımlarına geçebiliriz. Tablodan görüleceği gibi X_3 eğer 1 yapılırsa, X_3 'e ait kısıt fonksiyonlarındaki katsayılar pozitif olduğundan dolayı S_i değerleri sırasıyla $S_1 = 0$, $S_2 = -5$, $S_3 = -1$ olacağından daha kötü bir sonuç elde edilir. Bu yüzden $X_3 = 0$ olacaktır. Aynı şekilde X_1 , X_2 , X_4 değişkenleri 1 yapılırsa yine tüm S_i değerleri pozitif olmaz. Ancak X_5 değişkeni 1 olduğunda S_i değerleri pozitif olmaktadır.

$$\begin{aligned}(S_1^1, S_2^1, S_3^1) &= (S_1^0 - (-1), S_2^0 - (-3), S_3^0 - (-3)) \\ &= (2, 1, 1)\end{aligned}$$

Yani $X_5 = 1$ ve $G^1 = 3$ olur. Burada $\min G^* = G^1 = 3$ değeri mevcut en iyi çözümdür. Bir sonraki aşamada mevcut değerden daha iyi bir çözüm bulmaya çalışacağız.



Şekil 3.1. Başlangıç İterasyon

Yukarıdaki şekilde başlangıç noktası olarak tüm $X_j = 0$ olduğu 0. düğüm gösterilmiştir. Bu noktadan iki dal çıkmaktadır. Bu dallar $X_5 = 0$ ve $X_5 = 1$ değerlerini ve 1. düğümde elde edilen mümkün çözümü göstermektedir. Tüm amaç fonksiyonu katsayıları pozitif olduğundan 1. düğümde sonra elde edilecek tüm dallarda daha iyi bir amaç değeri elde edilmeyeceği için $X_5 = 1$ dalı sınırlanır. Dolayısıyla $X_5 = 0$ eldeki tek dal olduğundan ona bağlı değerler dikkate alınacaktır. Bu durumda

$$(S_1^2, S_2^2, S_3^2) = (1, -2, -1), \quad G^2 = 0, \quad \text{tüm } X_j = 0 \text{ olur.}$$

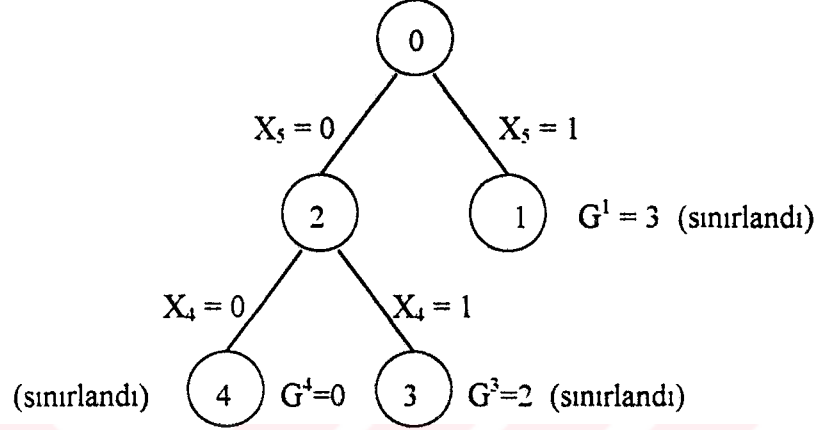
2. düğümde yalnızca X_1, X_2, X_3, X_4 değişkenleri dikkate alınacaktır. X_1 veya X_3 değişkenlerinin 1 yapılması, C_j katsayıları pozitif olduğundan amaç fonksiyonunu daha iyi bir çözüme götürmez. Bu nedenle elenirler. Geriye X_2 ve X_4 kalır. Bunlardan hangisinin seçileceği her X_j değişkeni için şu ampirik kuralla bulunur:

$$v_j = \sum \min \{ 0, S_i - a_{ij} \}$$

Testin amacı hangi değişkenin toplam mümkün olmama durumunu daha çok azalttığını bulmaktır. En küçük v_j değerine sahip olan X_j değeri 1 yapılacaktır.

$$v_2 = 0 + (-2 - 0) + 0 = -2$$

$$v_4 = (1 - 2) + 0 + 0 = -1$$



Şekil 3.2. İkinci İterasyon

$$(S_1^3, S_2^3, S_3^3) = (1 - 2, -2 + 4, -1 + 3) \\ = (-1, 2, 2) \quad , \quad G^3 = 2 \quad , \quad \text{tüm } X_j = 0 \text{ olur.}$$

3. düğümde $X_5 = 0$ ve $X_4 = 1$ olarak belirlendi. Bu daldan sonra X_1 , X_2 ve X_3 değişkenleri incelenebilir. Ancak bu değişkenlere ait C_j değerleri pozitif olduğundan amaç fonksiyonunu daha iyi yapmayacaklardır. Bu nedenle 3. dal da sınırlanır.

Geriye sadece 4. düğüm kalmıştır. Burada $X_5 = 0$ ve $X_4 = 0$ olduğundan

$$(S_1^4, S_2^4, S_3^4) = (1, -2, -1) \quad \text{ve} \quad G^4 = 0 \text{ olur. Burada da daha önceki} \\ \text{nedenlerden dolayı } X_1, X_2, X_3 \text{ elenir ve 4. düğüm de sınırlanır.}$$

Sınırlanmamış hiç bir dal kalmadığından algoritma sona erer. Çözüm düğümü 1. düğümdür. Çünkü ancak bu düğümde tüm S_i değerleri pozitifdir ve mümkün çözüme ulaşarak $G = 3$ değeri elde edilir. Bu sonuçlar orjinal değerlerine dönüştürüldüklerinde $X'_1 = X'_2 = 1$, $X'_3 = X'_4 = X'_5 = 0$ ve $Z = 5$ değerleri elde edilir.

Daha önce de belirttiğimiz gibi çözüm prosedürü 2^n adet seçeneği ele almaktadır. Bu durum örneğimizin çözümünde de gözlenmektedir. 1. düğümün sınırlandırılması $X_5=1$ durumunu içeren tüm çözümleri dikkate almıştır. Geriye $2^{5-1}=6$ çözüm kalır. 3.düğüm de kısıtlanmıştır. $X_5=0$ ve $X_4=1$ olması durumunda 3.düğümde $2^{5-2}=8$ adet çözüm dikkate alınmıştır. Son olarak 4.düğümün de sınırlandırılmasıyla $2^{5-2}=8$ çözüm dikkate alınmıştır. $16+8+8=32$ adet mümkün olan tüm çözümlerin incelendiğini göstermektedir.

Görüldüğü gibi aslında 5 adet çözüm açık olarak incelendiği halde 32 adet çözüm kapalı olarak dikkate alınmıştır. Bu nedenle bu prosedür “ Kapalı Sayma “ olarak da anılmaktadır.

Çözümün daha kolay bir şekilde elde edilmesi için 0-1 dal-sınır algoritması daha basit olarak gösterilebilir. Bunun için şu tanımlamalara ihtiyacımız vardır:

1.Serbest Değişken : Bir değişkenin değeri dal-sınır ağacının herhangi bir düğümünde. bu düğüme ulaşan dallardan herhangi birisinde sabitlenmemiş ise buna serbest değişken denir. Şekil 3.2’de 3.düğümde X_1 , X_2 ve X_3 serbest değişkenlerdir. Serbest bir değişken başlangıçta 0 seviyesindedir. Eğer problemin mümkünlik durumunu iyileştirebiliyorsa değeri 1’e yükseltilebilir.

2.Kısmi Çözüm : Bir ya da daha fazla değişkenin değerinin 0 ya da 1 olarak sabitlendiği durumdur. Kısmi çözüm aslında belirli bir sistemle elemanları sınırlanmış olan bir kümedir.

Örneğin, J^t t. düğümdeki kısmi çözüm kümesi olsun. $+j$, $X_j=1$ olduğunu , $-j$ ise $X_j=0$ olduğunu ifade etsin. Dolayısıyla J^t kümesi değeri sabitlenmiş değişkenleri $+$ ve $-$ olarak ifade edildiği bir küme olacaktır. Kümeye katılacak her yeni eleman en sağdan ilave edilecektir.

Kısmi çözümler düğümleri tanımlamak için kullanılabilir. Çünkü bir düğüme giden dal aslında bazı değişkenlere atanan 0-1 değerlerini göstermektedir. Şekil 3.2'deki düğümler aşağıdaki gibi gösterilebilir:

0 .düğüm	: $J = \emptyset$
1.düğüm	: $J = \{5\}$
2.düğüm	: $J = \{-5\}$
3.düğüm	: $J = \{-5,4\}$
4.düğüm	: $J = \{-5,-4\}$

Kısmi çözümleri en büyük kolaylığı diğer çözümlerin bir önceki çözümden türetilmesidir. Bu işlemi daha önceki örneğimiz için açıklayalım. Öncelikle düğümlerin sınırlanması için gerekli kurallar şunlardır:

1. Daha iyi bir amaç fonksiyonu değerine ulaşamama durumu.
2. Mümkün bir çözüme ulaşamama durumu.

Sınırlanmış bir düğümden sonraki çözümlerin incelenmesine gerek yoktur. Çünkü daha iyi bir çözüme ulaşamayacaktır. Önceki örneğimizde çözüm $J^0 = \emptyset$ ile başlar. Yani tüm değişkenler başlangıçta 0 seviyesindedir. Şimdi $X_5=1$ olduğunda çözüm mümkün ve $J^1 = \{5\}$ olur. Burada geriye kalan serbest değişkenlerin C_j katsayıları pozitif olduğundan bunlardan herhangi birinin 1 seviyesine yükseltilmesi çözümü daha iyi hale getiremez. Dolayısıyla J^1 sınırlanır. Bundan sonra bir sonraki kısmi çözüm, önceki çözümün yani J^1 'in en sağdaki elemanının tamamlanmasıyla elde edilir. 'Tamamlanma' terimi eleman eğer 1 ise 0'a, 0 ise 1'e dönüştürülerek yeni bir durum elde edilmesini ifade eder. Bu işlem sonucu $J^2 = \{-5\}$ çözümüne ulaşılır. Ancak yöntemin kesin kurallarını açıklamadan önce bazı temel kavramları açıklayalım:

J^2 kümesi, $X_5=1$ çözümünün elenip $X_5=0$ durumunun incelendiğini ifade etmektedir. Örneğimizin önceki çözümünde açıklanarak v_j testine göre $X_4=1$ yapılabilir. Bu yüzden J^2 sınırlanamaz. Böylece $J^3 = \{-5,4\}$ elde edilir. J^3 'e yeni değişken ilavesi daha iyi çözüm sunamaz. Çünkü C_1, C_2, C_3 pozitifdir. Bu yüzden J^3 de sınırlanır. Şimdi yine en sağdaki elemanın tamamlanmasıyla $J^4 = \{-5,-4\}$ çözümü elde edilir. Bu noktada

$X_5=0$, $X_4=1$ 'dir. Serbest değişkenlerden X_3 hem optimallik hem de mümkünlik nedeniyle X_1 ve X_2 değişkeni ise mümkünlik nedeniyle çözüme katılamazlar. Dolayısıyla J^4 de sınırlınır. Tüm elemanlar negatif olduğunda çözüm eleme aşaması sona erer. Bir sonraki kısmi çözüm elde edilmesi için en sağdaki pozitif eleman seçilir ve 'tamamlanır'. Ondan sonraki tüm negatif elemanlar silinir. Örneğin $J^4=\{1,5,-3,2,6,-7,-8\}$ sınırlanmış ise $J^{4+1}=\{1,5,-3,2,-6\}$ olacaktır. Çözüme ilave edilen değişkenler 1 seviyesinde girdikleri için negatif bir eleman bir önceki çözümün sınırlanmış bir kısmi çözümün tüm değerleri negatif ise çözüm eleme aşaması sona ermiş olur.

Şimdi kısmi çözümlerin sınırlanması ve yeni elemanların çözüme ilave edilmesi için uygulanacak testleri açıklayalım. Aşağıdaki 0- tamsayı problemlerini ele alalım:

$$\begin{aligned} \min Z &= \sum_{j=1, \dots, n} C_j X_j & \text{tüm } C_j &\geq 0 \\ \text{Kısıtlar} & & & \\ \sum_{j=1, \dots, n} a_{ij} X_j + S_i &= b_i & i=1, \dots, m & \quad j=1, \dots, n \\ X_j &= 0 \text{ ya da } X_j = 1 & & \\ S_i &\geq 0 & & \end{aligned}$$

J^t t. düğümdeki kısmi çözüm olsun. Başlangıçta $J^0=\emptyset$ ve tüm değişkenler sıfır durumundadır. Z_t t. düğümdeki Z değerini, Z^* ise mevcut en iyi üst sınırı gösterebilir. Başlangıç $Z^*=-\infty$ 'dur. Bu değer indirilmeye çalışılacaktır.

TEST 1 : Herhangi bir X_r değişkeni için $S_i^t < 0$ değeriyle ilintili olan ve tüm i değerlerine ait $a_{ir} \geq 0$ ise, X_r 'nin 1'e yükseltilmesi problemin mümkünliğini daha kötü yapar. Bu nedenle X_r çözüme alınmaz.

TEST 2 : Herhangi bir X_r serbest değişkeni için eğer $C_r + Z_t \geq Z^*$ ise daha iyi bir çözüme ulaşamayacağı için çözüme alınmaz.

TEST 3 : $S_i^t < 0$ olduğu i . kısıtı dikkate alalım.

$$a_{i1} X_1 + a_{i2} X_2 + \dots + a_{in} X_n + S_i = b_i$$

N_t , test 1 ve test 2 tarafından elenmemiş serbest değişkenler kümesini gösterebilir. Aşağıdaki koşul sağlandığında, en az bir $S_i^t < 0$ ise N_t 'deki serbest değişkenlerin hiç biri daha iyi bir çözüm sağlamaz. Bu durumda J_t sınırlandırılır.

$$\sum \min \{ 0, a_{ij} \} > S_i^t \quad j \in N_t$$

TEST 4 : Eğer $N_t \neq \emptyset$ ise seçilecek X_k değişkeni şu ilişkiye göre belirlenir:

$$v_k^t = \max \{ v_j^t \}$$

$$v_j^t = \sum \min \{ 0, S_i^t - a_{ij} \} \quad i = 1, \dots, m$$

Eğer $v_k^t = 0$ ise, $X_k = 1$ olması daha iyi bir mümkün çözüme ulaşılmasını sağlar. Bu durumda J^t 'ye $\{k\}$ ilavesiyle elde edilen J^{t+1} sınırlandırılır. Aksi halde, sınırlandırılmış kısmi çözümün tüm elemanları negatif olup eleme işlemi bitene kadar tüm testler J^{t+1} 'e uygulanır.

Şimdi örneğimizi yeniden çözelim.

İterasyon 0:

Başlangıçta $J^0 = \emptyset$, $Z^* = \infty$ 'dur.

$(S_1^0, S_2^0, S_3^0) = (1, -2, -1)$ ve $Z^0 = 0$

Test 1'e göre, S_2^0 ve S_3^0 değerleri negatif olduğundan, X_3 değişkeninin 2. ve 3. kısıt denklemlerindeki katsayılarının pozitif olması bu değişkenin elenmesini gerektirir. Çünkü S_2^0 ve S_3^0 değerleri daha negatif olur.

Test 3'e göre $N^0 = \{1, 2, 4, 5\}$ elenemez. Çünkü

$$S_2: -7 - 0 - 4 - 3 = -14 < -2$$

$$S_3: 0 - 6 - 3 - 3 = -12 < -1$$

olduğundan bu değişkenlerin çözüme alınması S_2 ve S_3 değerlerini daha pozitif yapabilir.

Test 4'e göre

$$v_1^0 = 0 + 0 + (-1 - 11) = -12$$

$$v_2^0 = 0 + (-2 - 0) + 0 = -2$$

$$v_3^0 = (1-2) + 0 + 0 = -1$$

$$v_4^0 = 0 + 0 + 0 + 0 = 0$$

olduğundan $k=5$ 'dir

İterasyon 1:

$J^1 = \{5\}$ için $Z^* = 3$ olur.

$$(S_1^1, S_2^1, S_3^1) = (1+1, -2+3, -1+3) = (2, 1, 2)$$

$$Z^1 = 3$$

Bu çözüm mümkün olduğundan $Z^* = Z^1 = 3$ olur. Dolayısıyla J^1 sınırlanır.

İterasyon 2:

$J^2 = \{-5\}$ için $Z^* = 3$ olur.

$$(S_1^2, S_2^2, S_3^2) = (1, -2, -1) = (2, 1, 2)$$

$$Z^2 = 0$$

Test 1 X_3 'ü eler. Test 2 ise X_1 ve X_3 'ü eler. Test 3 ile $N_2 = \{2, 4\}$ elenemez. Test 4'e göre $v_2^2 = -2$ ve $v_4^2 = -1$ olur. Buradan $k=4$ olur.

İterasyon 3:

$J_3 = \{-5, 4\}$ için $Z^* = 3$ olur.

$$(S_1^3, S_2^3, S_3^3) = (-1, 2, 2)$$

$$Z^3 = 2$$

Test 1 X_3 'ü eler. Test 2 X_1, X_2 ve X_3 'ü eler. $N_3 = \emptyset$ olduğundan J^3 sınırlanır.

İterasyon 4:

$J^4 = \{-5, -4\}$ için $Z^* = 3$ olur.

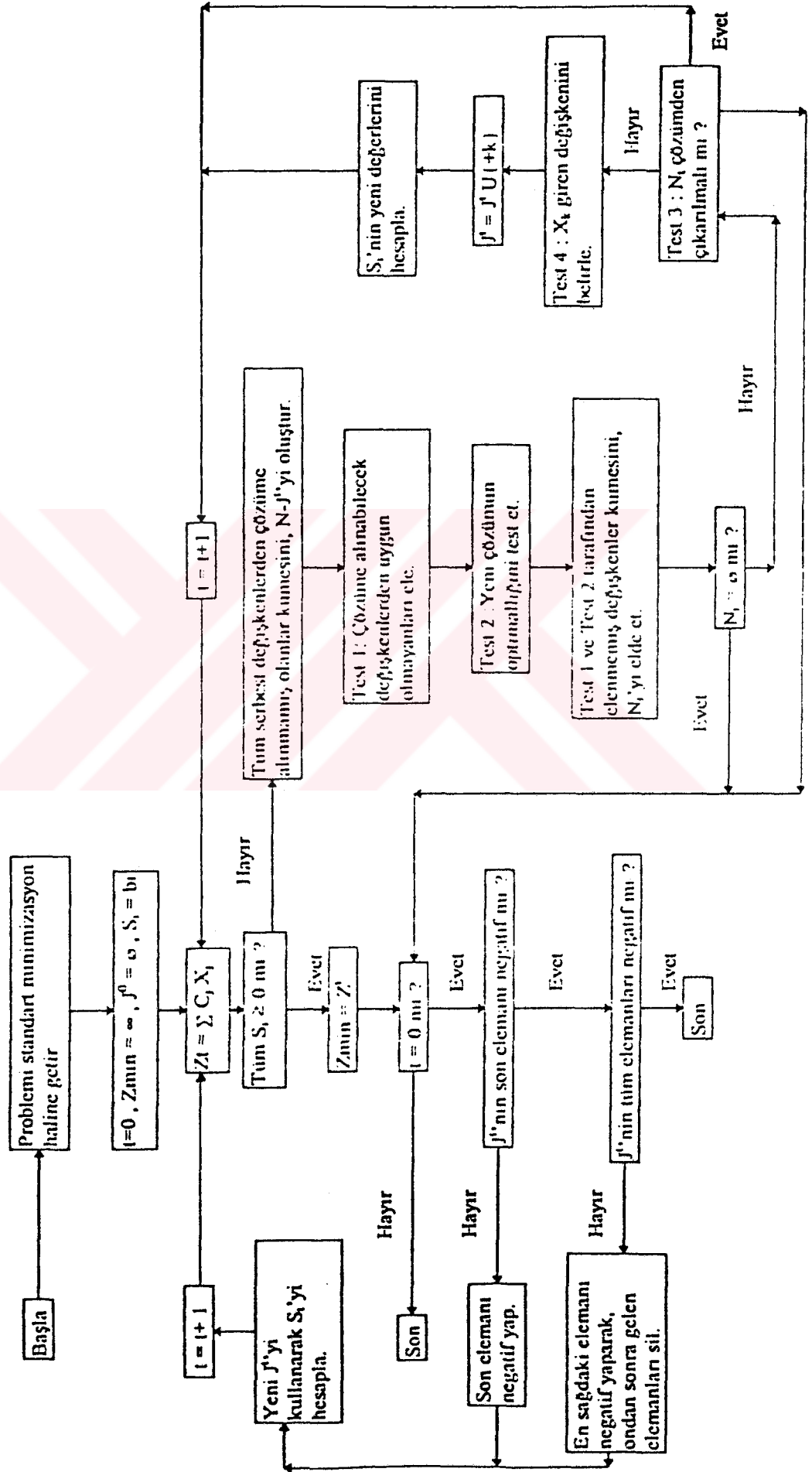
$$(S_1^4, S_2^4, S_3^4) = (1, -2, -1)$$

$$Z^4 = 0$$

Test X_3 'ü eler. Test 2 X_1 ve X_3 'ü eler. Test 3'e göre $N_4 = \{2\}$ elenmelidir. Dolayısıyla J^4 sınırlanır. J^4 'ün tüm elemanları negatif olduğundan eleme bitmiştir.

Görüldüğü gibi hem mümkün hem de optimal çözümü $J^1 = \{5\}$ vermektedir. Yani $X_1=0, X_2=0, X_3=0, X_4=0, X_5=1$ olur. Asıl max. problemi için uygun dönüştürmeler yapıldığında $X'_1=1, X'_2=1, X'_3=0, X'_4=0, X'_5=0$ elde edilir.

BALAS ALGORİTMASI



3.3. HÜCRESEL İMALAT SİSTEMLERİNDE MAKİNE GRUPLAMA PROBLEMİNE 0-1 TAMSAYILI PROGRAMLAMA YAKLAŞIMI

Hücre oluşturma ile ilgili mevcut literatürde, parça ailelerini ya da makina gruplarını bağımsız olarak tasarlayan veya makina parça ailelerini eş zamanlı olarak oluşturan yöntemler bulunmaktadır. Bu çalışma ise parça aileleri (makina grupları) oluşturulduktan sonra ilgili gruplara atanacak makina gruplarını (parça aileleri) oluşturma amacını gütmektedir. Parça ve makina sayıları çok fazla olduğunda ve alternatif rotalar mevcutsa hücre oluşturma sorunu çok karışık olmaktadır.

Çalışmamızın bu bölümünde hücresel imalat sistemlerinde makina gruplaması için bir yöntem sunulmuştur. Bu yöntem parçaların işlenmesi için gerekli takımlara, makinalar üzerindeki mevcut işleme takımlarına ve işleme sürelerine bağlı olarak uygulanır. İki tane 0-1 tamsayılı programlama formülasyonu oluşturulmuştur. Bu formülasyonlar parça ailelerinin bilindiğini varsaymaktadır. İlk formülasyonda parçaların makinalarla uyumluluğuna dayanılarak makinalar gruplanır. İkinci formülasyonda ise makinaları gruplarken makina yerleştirme ve hücreler arası hareket maliyetleri minimize edilmeye çalışılır. Her iki formülasyonda da bir gruptaki makina sayısı ve belli bir tip makinadan elde mevcut olarak kaç tane bulunduğunu dikkate alınır. Sunulan formülasyonların ilkinde ait birinci amaç fonksiyonu hipotetik olarak örneklenip modellenmiş ve uygun bilgisayar yazılımlarıyla çözümlenerek anlamlı sonuçlar elde edilmiştir.

3.3.1.Önerilen Yöntem

Literatürdeki makina gruplama yöntemleri parça ile ilgili her bir işlemin sadece bir makina tarafından yapılabileceği varsayımına dayanmaktadır. Bu bilgi genellikle her parça için makinada işlenme sırası ile ifade edilmektedir. Ancak bu varsayım hücresel imalat sistemleri için çok geçerli olmamaktadır. Hücresel imalat sistemlerinde belli bir parçanın her işleminin alternatif makinalarda yapılabileceği varsayılmaktadır. Yöntemlerin aynı zamanda darboğaz oluşturan makinalar ve istisnai parçalarla ilgili sorunları çözebilmek için özel ilavelere ve değişikliklere ihtiyacı vardır. Bu çalışmada önerilen yöntem, makinaları ilgili parçaları işleme yeteneklerine bağlı olarak

gruplamaktadır. Parçaların işlenme gereksinimleri, takım gereksinimleriyle ilişkilidir. Makina yeterliliği ise her makina üzerindeki mevcut takımlarla ifade edilmektedir. İndeksler makina ile parça arasındaki uyumluluğu tanımlamak için, parçanın takım ihtiyaçlarına, makinadaki mevcut takımlara ve işlem sürelerine bağlı olarak tanımlanmış, iki tane 0-1 tamsayılı programlama formülasyonu makina gruplama problemi için geliştirilmiştir. Bu formülasyonlarda parça ailelerinin bilindiğini varsayılmaktadır.

İlk formülasyon tüm gruptaki makina ve parça gruplarının uyumluluk indekslerinin maksimize edilmesini amaçlayarak makina ve parçaları gruplar. İkinci formülasyon, ilave makina alma maliyeti ile hücreler arası taşıma maliyetleri arasında bir seçim yapmak üzerine kurulmuştur. Bu formülasyonlar sistemdeki bir takım fiziksel kısıtları dikkate almaktadır. Bu kısıtlar, bir hücredeki makina sayısı limiti ve her makinadan elde bulunan miktardır.

3.3.2. Matematiksel Modelleme

3.3.2.1. Değişkenlerin Gösterimi

i	: Parça indeksi	, $i=1,...,n$
j	: Makina indeksi	, $j=1,...,m$
k	: Hücre indeksi	, $k=1,...,p$
N_i	: i parça için gereken takım sayısı	
M_j	: j . makinadaki mevcut takım sayısı	
T_{ij}	: i . parça ile j . makina arasındaki ortak takım sayısı	
O_{ij}	: i . parça üzerinde j . makinada yapılması gereken işlem sayısı	
P_i	: i . parça üzerinde yapılması gereken tüm işlemlerin kümesi	
S_{ij}	: i . parça ile j . makina arasındaki uyumluluk indeksi	
t_{il}	: i . parçanın l . işlemi için gereken süre	
q_i	: i . parçanın yıllık üretim miktarı	
f_j	: j . makinanın yıllık sabit maliyet oranı	
L_i	: i . parçanın birim taşıma yükü hacmi	

- c : İki hücre arası taşımanın ortalama maliyeti
 A_j : j. makinanın mevcut miktarı
 B_j : Alınabilecek max. sayıdaki j makinası
 G_k : k hücresindeki makina sayısı limiti
 a_{ik} : Parça-hücre değişkeni (Eğer i parçası k hücresine atanırsa $a_{ik} = 1$, aksi halde $a_{ik} = 0$ olacaktır.)

Karar değişkenleri ise aşağıdaki şekilde tanımlanmıştır .

- X_{jk} : Makina-hücre değişkeni (Eğer j makinası k hücresine atanırsa $X_{jk} = 1$, aksi halde $X_{jk} = 0$ olacaktır.)

3.3.2.2.Uyumluluk İndeksinin Oluşturulması

Bir makinanın bir parça ile uyumluluğunu tanımlamak için indeks oluşturulması aşağıda açıklanmıştır.

(a) Takım Gereksinimlerine Dayalı Uyumluluk İndeksi

$$S_{ij} = T_{ij} / \min \{M_j, N_i\} \text{ olarak tanımlanmıştır.} \quad (3.1)$$

Bu indeks bir makina ile bir parçanın işlem uyumluluğunu ifade etmektedir. Bir parça ile makina arasındaki çok sayıda ortak takım olması, parça ile makina aynı gruba yerleştirildiğinde hücreler arası daha az taşıma gerçekleşecektir.

(b) Takım ihtiyaçları ve işleme sürelerini temel alan uyumluluk indeksi aşağıdaki şekilde tanımlanmıştır.

$$S_{ij} = \left(\sum_{l \in O_{ij}} t_{il} \right) / \left(\sum_{r \in P_i} t_{ir} \right) \quad (3.2)$$

Bu indeks bir makina ile bir parça arasında, makinanın kullanımı kadar malzeme hareketiyle ilgili olan uyumluluğu da ifade etmektedir. Eğer parça ile makina aynı gruba yerleştirilir ve parçanın makinadaki işlem süresi ne kadar uzun olursa, o makinanın kullanımı o kadar fazla olacaktır.

3.3.2.3.Makina Gruplama Modellerinin Formülasyonu

Makina gruplama problemi iki farklı durumda ortaya çıkabilmektedir. İlk durumda mevcut bir imalat sisteminin hücresel sistem olarak yeniden organize edilmesi istenebilir. Bu durumda eldeki her tip makina bir ya da daha fazla gruba mutlaka yerleştirilecektir. Burada eldeki hiç bir makinanın elden çıkarılmayacağı varsayılmaktadır. İkinci durumda ise yeni bir hücresel imalat sistemi kurulması ele alınmaktadır. Burada sorun her makina tipinden kaçar adet gerektiğini bulmaktır. Bu durumda ise bir takım maliyetler dikkate alınmaktadır.

Bu bölümde sunulan makina gruplama modelleri, her iki durumu da temsil edecek şekilde aşağıdaki varsayımlar altında formülize edilmişlerdir:

1- Parça aileleri bilinmektedir. Parça ailelerinin tasarımı geometrik özelliklere ya da işleme özelliklerine dayalı olarak yapılabilir. Kümeleme analizi, tamsayılı programlama, sezgisel yöntemler bu amaçla kullanılan tekniklerden bazılarıdır. Bu çalışmada eldeki parça-takım verileri Kümeleme Analizi ile gruplanarak modellerde veri olarak kullanılmak üzere hazırlanmışlardır.

2- Takım gereksinimleri, işleme süreleri ve üretim miktarları her parça için bilinmektedir.

3- Her makinadaki mevcut takımlar, makina maliyetleri ve malzeme taşıma maliyetleri bilinmektedir.

3.4.3.3.1. Model 1 - Mevcut Bir Sistemi Yeniden Organize Etme

Bu model formülize edilirken iki mümkün amaç fonksiyonu düşünülebilir. İlk amaç fonksiyonu, bütün hücrelerdeki tüm makina ve parça uyumluluk indekslerinin toplamının maksimize edilmesi için oluşturulur.

$$\max Z_1 = \sum_i \sum_j \sum_k X_{jk} * a_{ik} * S_{ij} \quad (3.3)$$

3.4.3.3.2. Model 2 - Yeni Bir Sistem Kurma

Bu modelin amaç fonksiyonu denklem 3.1’de verilmiştir. Hücre hacmi kısıtı denklem 3.5 ve 3.7deki gibi aynı kalmaktadır. Ancak denklem 3.6 şimdi aşağıdaki şekle dönüşmektedir.

$$\sum_k X_{jk} \leq B_j \quad \text{Tüm } j\text{'ler için} \quad (3.8)$$

Yukarıdaki kısıt, alınabilecek her tip makinanın maksimum sayısının aşılmamasını sağlamaktadır. Bu model, bir makinaı herhangi bir hücreye maliyet açısından etkin değilse atamamaktadır. Yani, eğer malzeme taşıma maliyetindeki tasarrufla o makinaı yerleştirme maliyeti arasındaki fark pozitif değilse atama işlemi yapılmamaktadır. Burada, her makina tipinden en az birinin herhangi bir hücrede yer almasının sağlanması için ek bir kısıta daha ihtiyaç duyulmaktadır. Aksi halde bazı parçalardaki bazı işlemlerin yapılması mümkün olamaz [18].

$$\sum_k X_{jk} \geq 1 \quad \text{Tüm } j\text{'ler için.} \quad (3.9)$$

BÖLÜM 4

LİTERATÜR TARAMASI

4.1.GİRİŞ

Grup Teknolojisi ile hücre oluşturma konusuyla ilgili olarak çok sayıda araştırma mevcuttur. Bu araştırmalarda önerilen yöntemler aşağıdaki yaklaşımlara dayanmaktadır [19]:

- Kodlama ve sınıflandırma
- Makina parça grup analizi
- Benzerlik katsayıları
- Bilgi tabanı
- Matematiksel programlama
- Bulanık kümeleme
- Sinir ağları
- Sezgisel yöntemler

Benzerliklerle ilgili rotalama, işleme, tasarım özellikleri ve maliyetlerle ilgili operasyon, malzeme taşıma, yatırım maliyetleri ile ilgili bilgileri içeren çok farklı amaçlar bu çalışmalarda ele alınmıştır. Bir hücredeki parça ya da makina sayısı, malzeme taşıma kapasitesi sınırlamaları gibi çok sayıda kısıt dikkate alınmıştır. Hücre tasarımı herhangi bir organizasyonda geniş varsayımlara dayanan karmaşık bir problemdir. Değişen imalat teknolojisi, mevcut bilgilerin tam doğru olmaması, donanımın yeniden düzenlenmesi, değişen proses planları, iskartalar gibi imalatın gerçekleri hücre tasarımı prosesini karmaşıklasmaktadır. Aşağıdaki bölümlerde daha önce açıklanan temel yaklaşımlarla ilgili olarak yapılmış olan çalışmalarla ilgili genel bilgiler verilmiştir. Bu bilgiler için 19 numaralı kaynak temel alınmıştır.

4.1.1. Parça Aileleri İçin Sınıflandırma ve Kodlama Sistemleri

Kodlama, parçalara semboller atama prosesini ifade etmektedir. Semboller benzer niteliklere sahip parçalardan aileler oluşturmak için, parçaların niteliklerini temsil ederler. Sınıflandırma ise parçaların tasarım özellikleri, imalat özellikleri ya da her ikisine dayanarak sınıflara ayrılma prosesini ifade etmektedir. Çok sayıda kodlama ve sınıflandırma sistemi geliştirilmiştir. Kodlama sistemlerinde kullanılan temel yapılar monokod, polikod ya da hibrid kod şeklindedir. En iyi bilinen sistemlerden bazıları Opitz, CODE, CUTPLAN, DCLASS ve Multi Class kodlama sistemleridir.

Tasarım, imalat, satınalma ve maliyet tahminine uygulanan sınıflandırma ve kodlama sistemlerinin kullanımı Hyer ve Wemmerlöv(1985) tarafından detaylı olarak tanımlanmıştır. Literatürde parça ailelerini oluşturmada bu sistemleri kullanan az sayıda çalışma vardır. Kusiak(1983) parça ailelerini oluşturmak için hiyerarşik kümeleme algoritması ve p-medyan formülasyonu önermiştir. Gongaware ve Ham(1981) ile Han ve Ham(1980) parça kodlarını çok amaçlı kümeleme algoritması ile parça aileleri oluşturmak için kullanmışlardır. Xu ve Wang(1989) bulanık kümeleme yaklaşımını önermişlerdir. Parça ailelerinin oluşturulması hücre oluşturulması probleminin sadece kısmi bir çözümüdür. Parça aileleri belirlendikten sonra, bu parçaları tamamıyla işleyebilecek makina grupları oluşturulmak zorundadır.

4.1.2. Makina-Parça Grup Analizi

Makina-parça grup analizi (İng: Machine-Component Grouping Analysis, MCGA) üretim akış analizine (İng: Production Flow Analysis, PFA) dayanır. MCGA'yı temel alan yöntemlerde makina-parça grupları, 0-1 matris formu şeklinde olan makina-parça tablosunun satır ve sütunları değiştirilmek suretiyle oluşturulur. En çok kullanılan MCGA temelli algoritmalar

- Bağ Enerjisi Algoritması (Mc.Cormick)
- Derece Sıralamasıyla Kümeleme (King)
- Doğrudan Kümeleme Algoritması (Chan ve Milner)
- Değiştirilmiş Derece Sıralamasıyla Kümeleme (Chandrasekaran, Rajagopalan)

Chu ve Pan(1988) ve Chu(1989) kümeleme prosedürleriyle ilgili incelemeler de yayınlamışlardır. Bu algoritmalar basit olmalarına karşın, imalatla ilgili üretim hacmi, işlem sırası, proses süreleri gibi gerçekleri dikkate almamaktadırlar. Çoğu durumda darboğaz makinalar problemi herhangi bir makina-parça grubu belirlenmeden ele alınmak durumundadır. King ve Nakornchai(1982) darboğaz makinaların ikilenmesini önermişlerdir. Makinaların ikilenme prosedürü Chan ve Milner(1982), darboğaz makinaların ve istisnai parçaların belirlenmesi Vannelli ve Kumar(1986) tarafından geliştirilmiştir. Shafer(1992) hücrelerarası taşıma maliyetlerini, makina ikileme ve fason yaptırma faktörlerini dikkate alarak istisnai parçalarla ilgili matematiksel programlama formülasyonu sunmuştur.

4.1.3. Benzerlik Katsayısına Dayalı Kümeleme Yöntemleri

Hücre oluşturma literatüründeki çoğu çalışma benzerlik katsayılarını network, matematiksel programlama ve bilgi tabanlı yöntemlerde kullanmışlardır. Benzerlik katsayısı ilk olarak Mc.Auley (1972) tarafından önerilmiştir. Benzerlik katsayısı yönteminde temelde makinalar, takımlar, tasarım özellikleri arasındaki benzerlik ölçütlerinin tanımlanması yatmaktadır. Daha sonra bu benzerlik katsayıları parça aileleri ve makina grupları oluşturmak için şu yöntemlerle kullanılmıştır:

- Tek Bağlantılı Kümeleme (Mc.Auley,1972)
- Ortalama Bağlantılı Yöntem (Seifoddini ve Wolfe,1986)
- Tam Bağlantılı Yöntem (Mosier,1989)
- Merkezi Yöntem (Mosier,1989)
- Medyan Yöntemi (Mosier,1989)

MACE (Waghodekar ve Sahn,1984),Matematiksel Sınıflandırma (Purchek,1975) ve diğer birçok sezgisel yöntem benzerliği, gruplamada bir ölçü olarak kullanmışlardır. Srinivasan ve Narendran (1991) hiyerarşik olmayan bir kümeleme yöntemi önermişlerdir. Kusiak ve Cho(1992) ayrışabilir ve ayrışamaz makina-parça matrislerine uygulanabilen iki benzerlik katsayısı önermişlerdir.

4.1.3.1. Grafik Teorisine Dayalı Yöntemler

Literatürde grafiklerin dik eksenlerini makinalar ve geometrik yayların ağırlıkları olarak temsil eden, grafik teorisine dayalı çok sayıda çalışma vardır. Rajagopalan ve Batra(1975) yayları temsil etmek için Jaccard benzerliğini kullanmışlardır. Faber ve Carter (1986) parçaların işleme ihtiyaçlarını içeren benzerlik ölçütü tanımlamışlardır. De Witte (1980) rotalama ve makina bilgilerini içeren üç benzerlik katsayısı geliştirmiştir.

4.1.3.2. Benzerlik Katsayısına Dayalı Matematiksel Modeller

Çok sayıda çalışma matematiksel programlama formülasyonlarında amaç fonksiyonu olarak farklı tiplerde benzerlik katsayıları kullanmışlardır. Kusiak (1987) değişen proses planlarını dikkate almış ve bir p-medyan formülasyonunu hücre oluşturmak için önermiştir. Amaç fonksiyonu benzerliklerin toplamının maksimum olmasıdır. Srinivasan (1990) genel atama problemi için bir atama modeli sunmuş ve bu modelde daha az istisnai eleman elde etmiştir. Kasilingram (1989) çok sayıda benzerlik indisi, tamsayılı programlama modeli ve Lagranj Esnetme yöntemini hücre oluşturmak için geliştirmiştir.

Bu yöntemler hücre oluşturma problemini statik olarak ve kararlı imalat sistemlerinde ele alırlar. Ancak imalatın gerçekleri çok periyotlu yaklaşımları, modellerin oluşturulmasında gerekli olan bilgilerin yapısında bulunan belirsizlik ve bulanıklıkların da dikkate alınmasını zorunlu kılmaktadır.

4.1.4. Matematiksel ve Sezgisel Yöntemler

Literatürde benzerlik katsayısını açıkça kullanmayan çok sayıda matematiksel programlama ve diğer analitik, sezgisel yöntemler mevcuttur. Choobineh(1988) ilk aşamada parça ailelerini oluşturan ardışık yaklaşımı, sonra makinaları parça ailelerine yerleştiren maliyete dayalı matematiksel programlama yöntemini kullanır. Rajamani(1990) hem ardışık olarak hem de eş zamanlı olarak hücre oluşturan

tamsayı programlama modelleri geliştirmiştir. Daha sonra değişken proses planları, malzeme taşıması ile yatırım, işletme, taşıma maliyetleri gibi farklı maliyetleri dikkate alan genel bir çerçeve sunmuşlardır. Askin ve Chin (1988) stok maliyetleri, makina hazırlık ve malzeme taşıması, makina amortismanı faktörlerini içeren modeller sunmuşlardır. Hücre oluşturma problemini çözmek için bir ayrıştırma planı ve sezgisel bir yöntem oluşturulmuştur. Harhalakis(1991) normalize edilmiş hücreler arası hareketi minimize etmek için bir sezgisel yöntem geliştirmiştir. Co ve Araar(1988) hücre sayısı ve hücre oluşumunu belirlemek için üç aşamalı bir yöntem önermişlerdir. Al-Qattani(1990) bir dal-sınır yöntemi geliştirmiştir. Vohra(1990) hücrelerarası hareketleri minimize etmeyi amaçlayan ve düzeltilmiş Gomory kısımlama algoritması kullanan şebeke yöntemini önermiştir.

4.1.5. Bilgi Tabanlı ve Şekil Tanımaya Yönelik Yöntemler

Hücre oluşturmaya uygulanan yapay zeka ve şekil tanıma alanlarında yapılmış çok az sayıda çalışma vardır. Kusiak(1988) uzman sistemleri kullanan bir bilgi tabanlı sistem geliştirmiştir. El Maraghy ve Gu(1988) bilgi kurallarını ve sintaktik şekil tanıma tekniklerini içeren bir sistem geliştirmişlerdir. Singh ve Qi(1991) sintaktik şekil tanımayı kullanan çok boyutlu benzerlik katsayısı kavramını tanıtmışlar ve doğal parça ailelerinin oluşturulması için bir algoritma geliştirmişlerdir.

4.1.6. Bulanık Kümeleme Yaklaşımı

Daha önce anlatılan tüm yaklaşımlar proses maliyetleri, proses süreleri, parça talebi v.b. ile ilgili bilgilerin doğru olduğunu varsayar. Her parçanın yalnız bir aileye ait olabileceği de varsayılmıştır. Ancak türleri tam belli olmayan parçalar var olabilir. Bulanık kümeleme bu tür problemlere çözüm sağlar. Hücre oluşturma problemindeki belirsizlik sorunlarıyla ilgilenen çok az sayıda çalışma mevcuttur. Bunlar da Ding(1988), Xu ve Wang(1989), Chu ve Hayya(1991)'ya aittir [20].

BÖLÜM 5

UYGULAMA

5.1. SİSTEMİN TANIMLANMASI

Uygulama olarak 10 adet farklı makina ile 25 adet, farklı özelliklerde metal oto yedek parçası üreten bir imalat sistemi ele alınmıştır. Bu imalat sistemi makina gruplama modellerinden Model-1'e ait ilk amaç fonksiyonuna uygun olarak modellenmiştir.

Parçaların işlenmesi için gereken takım ve imalat bilgileri Tablo 5.1'de verilmiştir. Örneğin 1 numaralı parçanın yıllık talebi 600 birimdir. Bu parçayı işlemek için 10 adet takıma ihtiyaç vardır. Takımlar da özelliklerine göre harf ve rakamlarla kodlanmışlardır. Örneğin A01 kodu, A ile kodlanmış takım grubunun 01 numaralı ölçüdeki takımı ifade eder. Daha somut olarak A matkap grubunu 01 de 5mm'lik çapı gösterirse, A01 kodu bize parçaya ilk olarak 5 mm'lik matkap ile delme işlemi uygulanacağını ifade eder.

Makinalardaki mevcut takımlar, makinaların yıllık sabit maliyet oranları ve her makina tipinden kaç tane olduğu ile ilgili bilgiler Tablo 5.2'dedir. Örneğin 1 nolu makinadan elde 2 tane mevcuttur. Bu makinanın yıllık sabit maliyeti 10.000 \$'dır ve bu makineye 6 adet farklı ölçülerde aynı cins takım bağlanabilmektedir.

Hücreler arası taşımanın ortalama maliyeti 500.000 TL, tüm parçalar için birim yükleme hacmi ise 5 parça olarak varsayılmıştır. Her hücreye atanabilecek maksimum makina sayısı 6 ile sınırlandırılmıştır. Daha önceden sezgisel bir yöntemle parçalar arasındaki işlenme özellikleri dikkate alınarak oluşturulan parça aileleri Tablo 5.3'te gösterilmiştir.

Tablo 5.1.Parçaları İşlemek İçin Gerekli Takımlar Tablosu

PARÇA NO	YILLIK TALEP	PARÇALARIN İŞLENMESİ İÇİN GEREKEN TAKIMLAR																	
1	600	A01	D01	D02	D03	H01	H02	M01	M02	M03	M04								
2	800	A02	A03	A04	B01	C01	C02	M01	M05	R01	R02	R03	R04						
3	1800	A01	B01	B02	B03	B04	E02	E03	E04	F01	F02	F03	F04	G05	R02	R03	R04	R06	
4	1200	B02	D01	D02	D03	D04	H01	H02	H03	M01	M02								
5	1300	B01	E02	E03	E04	E05	F04	F05	G01	G02	H03	R01	R03	R04					
6	900	A01	A02	A05	A06	C04	M02	M03	M04	R05	R06								
7	700	A02	A04	A06	C02	C03	C04	C05	F01	G01	M01	M03	M05	R02	R04	R05	R06		
8	1600	B01	B03	E05	E06	F03	F04	F05	G01	R01	R02	R04							
9	1200	A01	B02	B03	B05	B06	E01	E03	E05	E06	E07	F05	G01	G02	R01	R02			
10	1100	D01	D03	D04	E07	H01	H02	H03	H04	M02	M03								
11	1400	A01	A03	A06	C02	C03	C04	C06	M04	M05	R01	R02	R04	R06					
12	800	A01	D01	D02	H01	H02	M01	M02	M03										
13	1300	B01	B03	D01	E01	E03	E05	F01	F03	F04	G01	G03	R01	R02	R04	R06			
14	900	C01	D02	D03	D04	H02	H03	H04	M04	M05									
15	1100	A02	A04	A06	C01	C02	C03	C05	E06	G05	M01	M05	R01	R02	R04	R05			
16	600	A01	A02	C01	C03	C04	E01	M01	M03	M04	M05	R01	R02	R03					
17	1800	A01	D05	D06	F06	G06	H03	H04	M01	M03	M04	M05							
18	1100	B01	C01	E01	E03	E04	E07	F01	F05	G03	G04	G05	G06	R01	R03	R06			
19	1400	B01	B03	B04	C01	C02	E01	F01	F02	F03	H01	R01	R02	R03	R04				
20	800	A05	D03	D04	D05	D06	H01	H02	H04	M04	M05								
21	900	A01	A02	A03	A05	C02	C03	H01	M01	M02	M03	M04	R01	R06					
22	1600	A01	A02	C01	C02	C04	H01	M01	M02	R01	R02	R03							
23	1000	A01	B01	B02	B03	C01	E01	E02	F01	F02	F03	F05	G01	G02	H01	R01	R02	R03	
24	1200	B01	B02	E02	E04	E05	F01	F03	G02	R01	R02	R04	R06						
25	600	B01	B03	B04	C01	D02	E02	E03	F01	F03	F04	G01	G02	G04	R01	R03			

Tablo 5.2.Makinalarla İlgili Bilgiler

MAKİNA NO	MEVCUT MİKTAR	YILLIK SABİT MALİYET(S)	MAKİNALARA TAKILABİLEN TAKIMLAR						
1	2	10000	A01	A02	A03	A04	A05	A06	
2	1	8000	B01	B02	B03	B04	B05	B06	
3	1	6000	C01	C02	C03	C04	C05		
4	1	7000	D01	D02	D03	D04	D05	D06	
5	1	8500	E01	E02	E03	E04	E05	E06	E07
6	1	12000	F01	F02	F03	F04	F05	F06	
7	1	11000	G01	G02	G03	G04	G05	G06	
8	1	9000	H01	H02	H03	H04			
9	2	10500	M01	M02	M03	M04	M05		
10	2	11000	R01	R02	R03	R04	R05	R06	

5.2.MODELİN KURULMASI

Modelimiz parça ailelerini veri olarak içermektedir. Bu nedenle Tablo 5.1'den elde edilen ve parçaların işlenmesi için gereken takım bilgilerini kullanarak parça ailelerini oluşturan sezgisel yönteme ait bilgisayar programı Quick Basic ile yazılmıştır. Daha sonra modelde kullanılan diğer veriler ilgili tablolardan alınarak Quick Basic programlama dilinde yazılan diğer programlarla LINDO adlı Lineer Optimizasyon programının kullanabileceği veri formatlarına dönüştürülmüştür. LINDO'ya aktarılan veriler program içerisindeki Balas Algoritması'na uygun olarak 0-1 tamsayılı programlama modülüyle çözümlenip sonuçlar elde edilmiştir.

Parça ailelerinin elde edilmesinden sonra bu aileler modelde veri alınarak, yazılan bilgisayar programlarıyla hazırlanan ve LINDO'ya aktarılacak amaç ve kısıt fonksiyonları EK-A'da verilmiştir. Amaç ve kısıt fonksiyonlarının elde edilmesi için yazılan yardımcı programlar ise EK-B ile EK-H arasında mevcuttur. Bu programlarda fonksiyonların alt bileşenleri elde edilmektedir. Ekde verilen amaç fonksiyonundaki karar değişkenlerinden örneğin X12 değişkeninin katsayı değeri olan $a_{ik} * S_{ij}$ değerinin hesaplanması aşağıdaki gibidir. Diğer değişkenlerin katsayıları da hazırlanan program tarafından aynı şekilde hesaplanmaktadır.

$$\text{MODEL 1 : } \max Z_1 = \sum_i \sum_j \sum_k X_{jk} * a_{ik} * S_{ij}$$

Amaç fonksiyonunun açılımı aşağıda verilmiştir. Aralardaki noktalar kısaltma amacıyla kullanılmıştır ve dizinin devam ettiğini göstermektedir.

$$\begin{aligned} &X11*a11*S11 + X12*a12*S11 + X13*a13*S11 - \\ &X21*a11*S12 + X22*a12*S12 + X23*a13*S12 - \\ &X31*a11*S13 + X32*a12*S13 + X33*a13*S13 - \end{aligned}$$

. . .

$$X101*a11*S110 + X102*a12*S110 + X103*a13*S110 +$$

$$\begin{aligned} &X11*a21*S21 + X12*a22*S21 + X13*a23*S21 - \\ &X21*a21*S22 + X22*a22*S22 + X23*a23*S22 - \end{aligned}$$

$$X_{31} * a_{21} * S_{23} + X_{32} * a_{22} * S_{23} + X_{33} * a_{23} * S_{23} +$$

...

$$X_{101} * a_{21} * S_{210} + X_{102} * a_{22} * S_{210} + X_{103} * a_{231} * S_{210}$$

$$X_{11} * a_{251} * S_{251} + X_{12} * a_{252} * S_{251} + X_{13} * a_{253} * S_{251} +$$

$$X_{21} * a_{251} * S_{252} + X_{22} * a_{252} * S_{252} + X_{23} * a_{253} * S_{252} +$$

$$X_{31} * a_{251} * S_{253} + X_{32} * a_{252} * S_{253} + X_{33} * a_{253} * S_{253} +$$

...

$$X_{101} * a_{251} * S_{2510} + X_{102} * a_{252} * S_{2510} + X_{103} * a_{253} * S_{2510}$$

X12 karar değişkeninin toplam katsayısı olan 4 'ün elde edilişi aşağıdaki gibidir. Amaç fonksiyonunda birden fazla aynı değişken mevcuttur. Aynı olan değişkenlerin hepsinin katsayıları toplanarak toplam katsayı elde edilmektedir.

$$X_{jk} * a_{ik} * S_{ij} \longrightarrow$$

$$= X_{12} * (a_{12} * S_{11} + a_{22} * S_{21} + a_{32} * S_{31} + a_{42} * S_{41} + a_{52} * S_{51} + a_{62} * S_{61} \\ + a_{72} * S_{71} + a_{82} * S_{81} + a_{92} * S_{91} + a_{102} * S_{101} + a_{112} * S_{111} + \\ a_{122} * S_{121} + a_{132} * S_{131} + a_{142} * S_{141} + a_{152} * S_{151} + a_{162} * S_{161} + \\ a_{172} * S_{171} + a_{182} * S_{181} + a_{192} * S_{191} + a_{202} * S_{201} + a_{212} * S_{211} + \\ a_{22} * S_{221} + a_{232} * S_{231} + a_{242} * S_{241} + a_{252} * S_{251})$$

$$= X_{12} * (0 * S_{11} + 1 * S_{21} + 0 * S_{31} + 0 * S_{41} + 0 * S_{51} + 1 * S_{61} + \\ 1 * S_{71} + 0 * S_{81} + 0 * S_{91} + 0 * S_{101} + 1 * S_{111} + 0 * S_{121} + 0 * S_{131} + \\ 0 * S_{141} + 1 * S_{151} + 1 * S_{161} + 0 * S_{171} + 0 * S_{181} + 0 * S_{191} + 0 * S_{201} + \\ 1 * S_{211} + 1 * S_{221} + 0 * S_{231} + 0 * S_{241} + 0 * S_{251})$$

$$= X_{12} * (S_{21} + S_{61} + S_{71} + S_{111} + S_{151} + S_{161} + S_{211} + S_{221})$$

$$S_{ij} = T_{ij} / \min (M_j, N_i)$$

$$S_{21} = T_{21} / \min (M_1, N_2) = 3 / \min (6, 10) = 3 / 6 = 0.5$$

$$S_{61} = T_{61} / \min (M_1, N_6) = 4 / \min (6, 10) = 4 / 6 = 0.666$$

$$S71 = T71 / \min (M1, N7) = 3 / \min (6, 16) = 3 / 6 = 0.5$$

$$S111 = T111 / \min (M1, N11) = 3 / \min (6, 13) = 3 / 6 = 0.5$$

$$S151 = T151 / \min (M1, N15) = 3 / \min (6, 15) = 3 / 6 = 0.5$$

$$S161 = T161 / \min (M1, N16) = 2 / \min (6, 13) = 2 / 6 = 0.333$$

$$S211 = T211 / \min (M1, N21) = 4 / \min (6, 13) = 4 / 6 = 0.666$$

$$S221 = T221 / \min (M1, N22) = 2 / \min (6, 11) = 2 / 6 = 0.333$$

Sij değerleri yerine konduğunda

$$= X_{12} * (0.5 + 0.666 + 0.5 + 0.5 + 0.5 + 0.333 + 0.666 + 0.333) = 4 * X_{12}$$

Görüldüğü gibi program X_{12} karar değişkenini amaç fonksiyonu değerini 4 olarak hesaplamıştır.

Modeldeki çözüm değerleri X_{jk} değişkenlerinin değerleridir.(0 ya da 1). Bu değerler j makinasının k hücresine atanıp atanmadığını ifade eder. Sonuçlar Tablo 5.4'de verilmiştir.

5.3. PARÇA AİLELERİNİN ELDE EDİLMESİ

Modelimizin veri olarak aldığı parça ailelerinin elde edilmesi için yazılan bilgisayar programının temel aldığı varsayım parçalar arasındaki takım benzerliğidir. Program tüm parçalar arasında, ortak takımlarla işlenme özelliğine benzerlik katsayıları oluşturur. Bu benzerlik katsayısı

$$SPP_{ij} = PP_{ij} / (PP_{ii} + PP_{jj} - PP_{ij}) \quad (5.10)$$

formülüne göre hesaplanır . Burada,

SPP_{ij} : i. ve j. parçalar arasındaki benzerlik katsayısını

PP_{ii} : i. parçayı işleyen takım sayısını

PP_{jj} : j. parçayı işleyen takım sayısını

PP_{ij} : i. ve j. parçaları işleyen takım sayısını ifade eder.

Elde edilen benzerlik katsayıları minimum %30 benzerlik sınırına göre gruplanmıştır. Bu katsayıya göre oluşturulan gruplar üç tanedir ve içerikleri Tablo 5.3 verilmiştir. Programın algoritması ise şu şekildedir:

- 1 - Tablo 5.1 'deki parça-takım matrisinden veriler okunur.
- 2 - Bu verilerden her parçanın işlenmesi için gereken takım sayısı hesaplanır.
- 3 -Tüm parçalar arasında, ikili gruplar halinde kaç tane ortak takımla işlendikleri bir matris formunda hesaplanır.
- 4 - Bu matrisin verilerinden, benzerlik katsayısı formülüne göre parça-parça benzerlik katsayıları matris formunda hesaplanır.
- 5 - Elde edilen matris %30 benzerlik sınırına göre işlenerek parça aileleri oluşturulur.

Tablo 5.3. Parça Ailelerinin Kompozisyonu

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

Parça aileleri elde edildikten sonra Model-1'e göre makinaların optimal atamaları yapılmış ve şu sonuçlar elde edilmiştir:

Tablo 5.4. Model-1'e Ait Sonuçlar

PARÇA AİLE NO	ATANAN MAKİNA NO				
1	4	8	9		
2	1	3	7	9	10
3	1	2	5	6	10

Tabloda görüldüğü gibi elimizde mevcut bulunan 13 adet makinanın hepsi kısıtlarda öngörüldüğü üzere parça ailelerine atanmıştır. Bu şekilde oluşturulan makina-parça hücreleriyle maksimum takım benzerliği elde edilecek, hücreler arası taşıma minimum olarak gerçekleşecektir. Dolayısıyla imalat maliyetleri, taşıma maliyetlerinde oluşan azalma nedeniyle daha düşük olacaktır. Böylece GT'nin varmak istediği sonuca ulaşmıştır.

SONUÇLAR VE ÖNERİLER

Bu çalışmada bir adet tamsayılı programlama modeli makina gruplama problemi için kullanılmıştır. Parça ailelerinin bilindiği varsayılmaktadır. Makinaların gruplanması her ailedeki parçaların işlenme ihtiyaçları ve makinalardaki mevcut takım özellikleri temel alınarak yapılmıştır. Kullanılan model, sistemde oluşturulacak hücrelerdeki parçalar ve makinalar arasındaki takım benzerliğini maksimize ederek hücreler arası parça hareketleri ile üretim maliyetlerini azaltmaya yöneliktir. Elde edilen sonuçlar, ikinci olarak önerilen modelin de uygulanmasıyla elde edilecek sonuçlarla kıyaslanabilir. İkinci model, sistemde maliyetin minimizasyonunu amaçlamaktadır. Bu çalışmaların ışığında her iki fonksiyonu birleştiren yeni bir model oluşturularak her iki amacı da eşanlı olarak gerçekleştiren sonuçların etkinliği araştırılabilir.

Küçük hacimli problemler için mevcut tamsayılı programlama kodları, modelleri çözmek için kullanılabilir. Orta ölçekli imalat sistemlerinde toplam makina sayısı ortalama 15, grup sayısı ise ortalama 4'tür. O halde, oluşturulacak modeller 60 tane kadar 0-1 tamsayı değişken üreteceklerdir. Bu büyüklükteki tamsayılı programlama problemlerinin optimal çözümüne ulaşmak için gereken sürenin uzun değildir. Modelimiz sadece 30 tane 0-1 değişken içerdiğinden optimal çözüme ulaşmak yarım dakikadan az zaman almıştır. Bu bir planlama problemi olduğundan imalat sisteminin kurulması aşamasında çözülmekte ve sık tekrar edilmemektedir. Dolayısıyla modeller orta ölçekli hücresel imalat sistemlerinde etkin olarak kullanılmakta ve mevcut IP kodlarıyla çözülebilmektedir.

1987'de 32 Amerikan imalat işletmesinde yapılan bir araştırmada göre hücre sayısı max.35, hücre hacmi ise max.8 makina civarında olduğu bulgulanmıştır. Bunun sonucu olarak, ortalama bir sistemdeki toplam makina sayısı 280 kadar olabilmektedir. Bu durumda modeller 9100 tane kadar 0-1 tamsayı değişkeni üretmektedir. Bu tür orta ölçekli problemlerde Lagranj Gevşetme(Lagrange Relaxation) ve Çarpan Düzenleme (Multiplier Adjustment) yöntemlerine dayalı çözüm yaklaşımları ya da

Subgradient Optimizasyon gibi yöntemler arařtırmacılar tarafından önerilmektedir. Çok fazla sayıda 0-1 tamsayı deęiřkeni olan problemlere uygulanan başarılı çözümler de üretilmiřtir. Sonuç olarak geleceęin üretim sistemleri olan geniş hacimli Entegre Esnek İmalat Sistemleri'nde makina-parça hücrelerinin oluşturulmasında bu çalışmada uygulanan matematiksel modellerin kullanılması maliyet olarak daha etkin düzenlemelerin oluşturulmasını sağlayabilecektir.



KAYNAKLAR

- [1] **DURMUŞOĞLU,B.**,Grup Teknolojisi İmalat Sistemi ve Bu Sistemin Tasarımına Yönelik Yeni Bir Metod,İTÜ Fen Bilimleri Enst. Doktora Tezi(1986)
- [2] **KIMURA,O. ve TERADA,H.**,Design and Analysis of Pull System, A method of Multi Stage Production Control, Int.J.Prod.Res., V:19,3,p 241-253,(1981)
- [3] **TERESKO,J.**, Can robots integrate manufacturing plants? ,Ind.Week, V 204,p 99, (1980)
- [4] **MERCHANT, M.E.**, in Anon., Mod.Mater.Handling, p 37, (1982)
- [5] **HOLLIER,R.,CORLETT,N.**, Workflow In Batch Manufacturing, HMS,London, (1966)
- [6] **SKINNER,W.**, The Decline,Fall and Renewal Of Manufacturing Plants,Ind.Engr, V:32 , (1974)
- [7] **BOUCHER,T.O.**, Lot Sizing In Group Technology Production Systems, Int. Prod. Res., V:22,p 85, (1984)
- [8] **DUDLEY,N.A.**, Comparative productivity analysis,Int.J.Prod.Res., V:8 , p 397, (1970)
- [9] **OFFODILE,O.F.,MEHREZ,A.,GRZNAR,J.**, Cellular Manufacturing:A Taxonomic Review Framework,J.Prod.Man.Sys.Vol.13 , No:3 , (1994)
- [10] **SOKAL,R.R.,SNEATH,P.H.A.**, Principles of Numerical Taxonomy , San Francisco: W.H.Freeman and Co. , (1968).
- [11] **Mc.AULEY,J.**, Machine Grouping For Efficient Production , The Production Engineer , V:51 , N:2 , pp. 53-57 , (1972)
- [12] **RAJAGOPOLAN,R.,BATRA,J.L.**, Design Of Cellular Production Systems:A Grap-Theoretic Approach , Int.J.of Prod.Res., V:13 , N:6 , pp. 567-579, (1975)
- [13] **De WITTE,J.**, The Use of Similarity Coefficients in Production Flow Analysis.Int.J.of Prod.Res., V:20 , N:4 , pp. 503-514 , (1980)

- [14] **KING,J.R.**, Machine-Component grouping in Production Flow Analysis:An Approach Using A Rank Order Clustering Algorithm, Int.J.of Prod.Res., V:18 , N:2 ,(1980)
- [15] **H.OPITZ,H.**, A Classification System to Describe Workpieces,Parts I,II.,New York-Pergomon Press,(1970)
- [16] **KOLOC,J.**, Use of Workpiece Statistics to Develop Automatic Programming for NC Machine Tools, Int.J.of Machine Tool Des.Res.,V:9,p 65,(1969)
- [17] **TAHA,A.H.**, Operational Research-Integer Programming,Mc.Millan Pres,p 258,(1990)
- [18] **GUNASINGH,K.R.,LASHKARI,R.S.**, Machine Grouping Problem In Cellular Manufacturing Systems-An Integer Programming Approach, Int. J. Prod. Res., V: 27, No:9,p 1465,(1989)
- [19] **SINGH,N.**, Design of Cellular Manufacturing Systems:An Invited Review, EJOR, V:69, p 284-291,(1993)
- [20] **BEDWORTH,D.D.**,Computer Integrated Design and Manufacturing, Mc.Graw Hill,N.Y., (1991)

EK A

MODEL 1'E AİT MAKSİMİZASYON FONKSİYONUNUN BİLGİSAYAR YAZILIMI İLE ELDE EDİLEREK LINDO'YA AKTARILAN VERİ YAPISI

BAT

INTEGER 30

MAX

.166X11+.500X41+.500X81+.800X91+
.500X12+.166X22+.333X32+.333X92+.666X102+
.166X13+.666X23+.500X53+.666X63+.166X73+.666X103+
.166X21+.666X41+.500X81+.333X91+
.166X23+.666X53+.333X63+.333X73+.166X83+.500X103+
.666X12+.166X32+.500X92+.333X102+
.500X12+.666X32+.166X62+.166X72+.500X92+.666X102+
.333X23+.333X53+.500X63+.166X73+.500X103+
.166X13+.666X23+.833X53+.166X63+.333X73+.333X103+
.500X41+.166X51+.666X81+.333X91+
.500X12+.500X32+.333X92+.666X102+
.166X11+.333X41+.333X81+.500X91+
.333X23+.166X43+.500X53+.500X63+.333X73+.666X103+
.166X31+.500X41+.500X81+.333X91+
.500X12+.666X32+.166X52+.166X72+.333X92+.666X102+
.333X12+.500X32+.166X52+.666X92+.500X102+
.166X11+.333X41+.166X61+.166X71+.333X81+.666X91+
.166X23+.166X33+.666X53+.333X63+.666X73+.500X103+
.500X23+.333X33+.166X53+.500X63+.166X83+.666X103+
.166X11+.666X41+.500X81+.333X91+
.666X12+.333X32+.166X82+.666X92+.333X102+

.333X12+.500X32+.166X82+.333X92+.500X102+
 .166X13+.500X23+.166X33+.333X53+.666X63+.333X73+.166X83+.500X103+
 .333X23+.500X53+.333X63+.166X73+.666X103+
 .500X23+.166X33+.166X43+.333X53+.500X63+.500X73+.333X103

KISITLAR

$X_{11}+X_{21}+X_{31}+X_{41}+X_{51}-X_{61}+X_{71}+X_{81}+X_{91}+X_{101} \leq 6$

$X_{12}+X_{22}+X_{32}+X_{42}+X_{52}-X_{62}+X_{72}+X_{82}+X_{92}+X_{102} \leq 6$

$X_{13}+X_{23}+X_{33}+X_{43}+X_{53}-X_{63}+X_{73}+X_{83}+X_{93}+X_{103} \leq 6$

$X_{11}+X_{12}+X_{13}=2$

$X_{21}+X_{22}+X_{23}=1$

$X_{31}+X_{32}+X_{33}=1$

$X_{41}+X_{42}+X_{43}=1$

$X_{51}+X_{52}+X_{53}=1$

$X_{61}+X_{62}+X_{63}=1$

$X_{71}+X_{72}+X_{73}=1$

$X_{81}+X_{82}+X_{83}=1$

$X_{91}+X_{92}+X_{93}=2$

$X_{101}+X_{102}+X_{103}=2$

$X_{11} \leq 1$

$X_{12} \leq 1$

$X_{13} \leq 1$

$X_{21} \leq 1$

$X_{22} \leq 1$

$X_{23} \leq 1$

$X_{31} \leq 1$

$X_{32} \leq 1$

$X_{33} \leq 1$

$X_{41} \leq 1$

$X_{42} \leq 1$

$X_{43} \leq 1$

$X_{51} \leq 1$

X52<=1

X53<=1

X61<=1

X62<=1

X63<=1

X71<=1

X72<=1

X73<=1

X81<=1

X82<=1

X83<=1

X91<=1

X92<=1

X93<=1

X101<=1

X102<=1

X103<=1

END

BAT

LEAVE

EK B

PARÇA-HÜCRE MATRİSİ OLUŞTURMA PROGRAMI

'***** AIK.BAS *****'

```
10 CLS : DIM A$(100), GRUP$(100)
20 INPUT "DUZENLENMIS DOSYA ADI "; GERGR$: OPEN "I", #1, GERGR$
25 OPEN "O", #2, "C:\ERTAN\QB\AIK.DAT"
30 IF NOT EOF(1) THEN INPUT #1, K, GRUP$(K) ELSE CLOSE #1: GOTO 55
40 PRINT K, GRUP$(K)
45 LPRINT K, GRUP$(K)
50 GOTO 30
55 GS = K
60 FOR I = 1 TO 25
70 A$(I) = STRING$(GS, "0"): NEXT I
100 FOR J = 1 TO 25
102 FOR K = 1 TO GS
103 FOR L = 1 TO LEN(GRUP$(K)) / 2
105 FOR I = 11 TO 35
120 IF I = VAL(MID$(GRUP$(K), 2 * L - 1, 2)) THEN A$(I - 10) = LEFT$(A$(I - 10), K - 1) + "1" + RIGHT$(A$(I - 10), GS - K): GOTO 140
140 NEXT I, L, K
150 PRINT J, A$(J): LPRINT J, A$(J)
152 PRINT #2, J, A$(J)
155 NEXT J
```

EK C

PARÇA-MAKİNA BENZERLİK KATSAYILARI OLUŞTURMA PROGRAMI

***** SIJ.BAS *****

```
5 DIM TS, S!(50, 50), SS(25), N%(25), M%(10), T%(50, 50)
10 CLS
15 INPUT X: ON X GOSUB 20, 500: END
20 INPUT "PARCA TAKIM DOSYASI"; PT$: OPEN "I", #1, PT$
21 INPUT "MAKINA TAKIM DOSYASI"; MT$: OPEN "I", #2, MT$
22 OPEN "I", #3, "C:\ERTAN\QB\TIJ.DAT"
23 OPEN "O", #4, "C:\ERTAN\QB\SIJ.DAT"
25 FOR I = 1 TO 25
26 SS(I) = ""
27 IF NOT EOF(1) THEN INPUT #1, PRN%, PN%, DEM%, TRP$
28 IF NOT EOF(3) THEN INPUT #3, PARNO%, T$
29 FOR J = 1 TO 10
34 IF NOT EOF(2) THEN INPUT #2, PRON%, MN%, NAV%, AFC!, TAS$
35 N%(I) = LEN(TRP$) / 3: M%(J) = LEN(TAS$) / 3
36 IF N%(I) < M%(J) THEN BOL% = N%(I) ELSE BOL% = M%(J)
37 T%(I, J) = VAL(MID$(T$, 2 * J - 1, 2))
38 S!(I, J) = T%(I, J) / BOL%
42 EK$ = MID$(STR$(S!(I, J)), 2, 4)
43 IF LEN(EK$) < 4 THEN EK$ = EK$ + STRING$((4 - LEN(EK$)), "0")
44 SS(I) = SS(I) + EK$
45 NEXT J
46 PRINT #4, I, SS(I)
50 NEXT I: RETURN
```

```
500 OPEN "I", #4, "C:\ERTAN\QB\SIJ.DAT"
510 FOR I = 1 TO 25
520 IF NOT EOF(4) THEN INPUT #4, PN%, SS(I)
525 FOR J = 1 TO 10
526 D$ = MID$(SS(I), 4 * J - 3, 4)
530 PRINT "S["; I; ", "; J; "]=", VAL(D$)
540 'LPRINT "S["; I; ", "; J; "]="; VAL(D$)
545 NEXT J, I: RETURN
```



EK D

PARÇA AİLESİ OLUŞTURMA YAZILIMI

```
2 DIM A%(100, 100), TTSP%(100), OP%(100, 100), SPP!(100, 100), SPPY!(100,
100), G$(25), AB$(50), PTOOLS$(50), TA$(50), G1$(15)
4 GOSUB 850 'DOSYALARI AC
6 CLS : PRINT "***** SECIM MENUSU *****"
7 PRINT " 1 - DOSYAYA YENI MATRIS GIRISI"
8 PRINT " 2 - DOSYAYA TABLO VERILERI AKTARMA"
9 PRINT " 3 - GIRILMIS MATRISIN ISLENMESI"
10 PRINT " 4 - PP BENZERLIK MATRISI YAZDIRMA"
11 PRINT " 5 - PARCA GRUPLARI OLUSTURMA"
12 PRINT " 6 - PARCA GRUBU YAZDIRMA"
13 PRINT " 7 - CIKIS ( Q or q )"
14 LOCATE 23, 1: INPUT " SECIMINIZ.....", SECIMS
15 IF SECIMS = "Q" OR SECIMS = "q" OR SECIMS = "7" THEN 24
17 ON VAL(SECIMS) GOSUB 45, 860, 130, 140, 141, 142
18 GOTO 6
24 CLOSE : END
40 '*****PARCA-TAKIM MATRISI GIRILIYOR*****
45 CLS
47 INPUT "PARCA SAYISI= "; PS%: PS% = PS% + 10
51 INPUT "TAKIM SAYISI= "; TS%: TS% = TS% + 10
54 PRINT "PARCA-TAKIM MATRISI GIRILIYOR"
57 FOR I = 11 TO PS%
60 FOR J = 11 TO TS%
64 PRINT "a["; I, ", "; J, "]",
67 INPUT AS(I, J)
70 IF AS(I, J) = CHR$(13) THEN AS(I, J) = "0"
```

```

71 NEXT J
75 PRINT : NEXT I
86 PRINT "MATRIS GIRISI TAMAMLANDI...TOPLAM "; (PS% - 10) * (TS% - 10); " ADET VERI GIRDINIZ"
88 GOSUB 2000
92 RETURN
130 '*****GIRILMIS MATRIS ISLENİYOR*****
131 GOSUB 1000 'DOSYADAN VERI OKUNUYOR
132 GOSUB 150 'HER PARCA ICIN GEREKEN TAKIM SAYISI HESAPLANIYOR*****
133 GOSUB 200 'PP ORTAK TAKIM MATRISI HESAPLANIYOR
134 GOSUB 300 'PP BENZERLIK MATRISI HESAPLAMA VE DOSYAYA YAZMA
140 GOSUB 370: RETURN 'PP BENZERLIK MATRISI YAZDIRMA
141 GOSUB 600: RETURN ' PARCA GRUPLARI OLUSTURMA
142 GOSUB 760: RETURN ' PARCA GRUBU YAZDIRMA
150 '*****HER PARCA ICIN GEREKEN TAKIM SAYISI HESAPLANIYOR*****
155 IF PARTTOOLS = "" THEN INPUT "PARCA TAKIM DOSYASI :"; PARTTOOLS
168 OPEN "I", #4, PARTTOOLS
170 PRINT "HER PARCA ICIN GEREKEN TAKIM SAYISI HESAPLANIYOR"
173 PS% = CVI(PSAYS) + 10
175 FOR I = 11 TO PS%
180 IF NOT EOF(4) THEN INPUT #4, PRN%, PARN%, DEM%, PTOOL1$
182 IF PN% = PRN% THEN TTSP%(I) = LEN(PTOOL1$) / 3
185 PRINT I, TTSP%(I): INPUT T: NEXT I: CLOSE #4
195 RETURN
200 '*****PARCA-PARCA ORTAK TAKIM MATRISI HESAPLANIYOR*****
207 PRINT "PARCA-PARCA ORTAK TAKIM MATRISI HESAPLANIYOR"
208 PS% = CVI(PSAYS) + 10: TS% = CVI(TSAYS) + 10

```

```

210 FOR I = 11 TO PS% - 1
220 FOR K = I + 1 TO PS%
225 OP%(I, K) = 0
230 FOR J = 11 TO TS%
231 PRINT A%(I, J), A%(K, J): INPUT T
240 IF A%(I, J) = 1 AND A%(K, J) = 1 THEN OP%(I, K) = OP%(I, K) + 1
245 NEXT J
255 PRINT "PPOT["; I, ",": K, "J"; "      "; OP%(I, K), TTSP%(I), TTSP%(K)':
INPUT T
256 'LPRINT "PPOT["; I, ",": K, "J"; "      "; OP%(I, K)
260 NEXT K
267 NEXT I
280 RETURN

300      '*****PARCA-PARCA      BENZERLIK      MATRISI
HESAPLANIYOR*****
305 PRINT "PARCA-PARCA BENZERLIK MATRISI HESAPLANIYOR"
310 FOR I = 11 TO PS% - 1
320 FOR J = (I + 1) TO PS%
325 IF OP%(I, J) = 0 THEN SPP!(I, J) = 0: GOTO 335
330 SPP!(I, J) = OP%(I, J) / (TTSP%(I) + TTSP%(J) - OP%(I, J))
335 PRINT I, J, OP%(I, J), TTSP%(I), TTSP%(J), USING "#.##"; SPP!(I, J):
INPUT T
345 NEXT J
350 NEXT I: GOSUB 1200
365 RETURN
370 '*****PARCA-PARCA BENZERLIK MATRISI YAZDIRILIYOR"
371 PRINT "PARCA-PARCA BENZERLIK MATRISI YAZDIRILIYOR"
372 INPUT "E EKRANA / Y YAZICIYA"; EYS
373 IF PN% = 0 THEN INPUT "PROJE NO="; PN%
374 PT% = 0: FOR I = 1 TO PN%
375 GET #2, PN%
376 PS% = CVI(PSAYS) + 10: TS% = CVI(TSAYS) + 10

```

```

377 KN% = 1: I = 11
378 GET #3, KN%: PRON% = CVI(PRON$)
379 IF PRON% <> PN% THEN KN% = KN% + 1: GOTO 378
380 FOR J = I + 1 TO PS%
390 SPP!(I, J) = CVS(MID$(SPPSAT$, ((J - I) * 4 - 3), 4))
392 PRINT OP%(I, J), TTSP%(I), TTSP%(J)
395 PRINT "SPP("; I, ", "; J, ") = "; USING "#.##"; SPP!(I, J)
396 INPUT Y
398 NEXT J, I
400 RETURN
428 FOR I = 11 TO PS% - 1
430 FOR J = I + 1 TO PS%
432 PRINT "SPP["; I, ", "; J, "]; " "; USING "#.##"; SPP(I, J)
434 NEXT J: PRINT
436 NEXT I
438 GOSUB 2000
440 RETURN
600 '*****PARCA GRUPLARI OLUSTURULUYOR*****
601 '*****PROJEYLE İLGİLİ KAYIT BULUNUYOR
603 IF PN% = 0 THEN INPUT "PROJE NO="; PN%
605 GET #2, PN%
606 PS% = CVI(PSAYS) + 10
607 R = 1: FOR I = 11 TO PS% - 1
608 FOR J = I + 1 TO PS%
610 GET #3, R: KN% = CVI(PRON$)
611 IF PN% <> KN% THEN R = R + 1: GOTO 610
614 SPP!(I, J) = CVS(MID$(SPPSAT$, (J - I) * 4 - 3), 4))
615 NEXT J, I: PS% = CVI(PSAYS) + 10
616 '*****EN BÜYÜK SPP BULUNUYOR
618 GS% = 1: GS(1) = ""
620 FOR K = 1 TO (PS% - 11) * ((PS% - 10) / 2)
625 ENB! = 0

```

```

630 FOR I = 11 TO PS% - 1
640 FOR J = I + 1 TO PS%
650 IF SPP!(I, J) > ENB! THEN ENB! = SPP!(I, J): ENB1$ = RIGHT$(STR$(I), 2):
ENB2$ = RIGHT$(STR$(J), 2)
660 NEXT J, I
671 '*****BULUNAN PP İLGİLİ GRUBA ATANIYOR
672 IF G$(1) = "" THEN G$(1) = ENB1$ + "," + ENB2$: H% = 1: GOSUB 3500:
GOSUB 3000: GOTO 750
681 B$(1) = "0": B$(2) = "0"
682 F% = 1
683 L% = 1
685 N% = 1
688 K1$ = MIDS$(G$(L%), (3 * N%) - 2, 2)
689 K1% = VAL(K1$): ENB%(1) = VAL(ENB1$): ENB%(2) = VAL(ENB2$)
691 'PRINT K1%, ENB%(1), ENB%(2)
690 IF ENB%(F%) = K1% THEN B$(F%) = "1" + RIGHT$(STR$(10 + L%), 2):
GOTO 721
710 IF N% < ((LEN(G$(L%))) + 1) / 3 THEN N% = N% + 1: GOTO 688
720 IF L% < GS% THEN L% = L% + 1: GOTO 685
721 IF F% = 1 THEN F% = 2: GOTO 683
725 IF LEFT$(B$(1), 1) = "1" AND LEFT$(B$(2), 1) = "1" THEN PRINT "E":
GOSUB 3500: GOSUB 3000: GOTO 750
726 IF LEFT$(B$(1), 1) = "1" AND LEFT$(B$(2), 1) = "0" THEN PRINT "H":
GOSUB 4000: GOTO 750
728 IF LEFT$(B$(1), 1) = "0" AND LEFT$(B$(2), 1) = "1" THEN PRINT "H":
GOSUB 4100: GOTO 750
730 GS% = GS% + 1
740 G$(GS%) = ENB1$ + "," + ENB2$: H% = GS%: GOSUB 3500: GOSUB 3000
750 NEXT K
752 GOSUB 1300: RETURN

```

```

760 '*****PARCA GRUBU YAZDIRMA
761 CLS
762 IF PRN% = 0 THEN INPUT "PROJE NO GIRIN =", PRN%
763 IF GD$ = "" THEN INPUT "GRUP DOSYASININ ADINI GIRIN :"; GD$
766 OPEN "I", #6, GD$
767 IF NOT EOF(6) THEN INPUT #6, PN%, L%, GRUP$ ELSE CLOSE #6:
RETURN
768 IF PRN% <> PN% THEN 767
770 G$(L%) = GRUP$: PRINT PN%, L%, G$(L%): GOSUB 2500
785 GOTO 767
796 INPUT V: RETURN
850 '*****DOSYA ACMA
852 OPEN "R", #2, "C:\ERTAN\QB\PTVERI.DAT", 8
853 OPEN "R", #3, "C:\ERTAN\QB\SPP.DAT", 402
857 FIELD #2, 2 AS PROJNO$, 2 AS PSAY$, 2 AS MSAY$, 2 AS TSAY$
858 FIELD #3, 2 AS PRON$, 400 AS SPPSATS
859 RETURN
'*****TABLO VERILERI OKUMA*****
860 INPUT "PARCA-TAKIM DOSYA ADINI GIRIN :"; PARTTOOLS: INPUT
"MAKINA-TAKIM DOSYA ADINI GIRIN :"; MACTOOLS
861 OPEN "I", #4, PARTTOOLS: OPEN "I", #5, MACTOOLS: OPEN "O", #1,
"C:\ERTAN\QB\PARTAK.DAT"
862 INPUT "PROJE NO="; PN%: I = 1
863 IF NOT EOF(4) THEN INPUT #4, PRN%, PARN%, DEM%, PTOOL1$ ELSE
865
864 IF PN% = PRN% THEN PTOOLS(I) = PTOOL1$: PRINT PTOOLS(I): I = I +
1: GOTO 863
865 J = 1: TAKSAY% = 0
866 IF NOT EOF(5) THEN INPUT #5, PRON%, MN%, NAV%, AFC!, TA1$
ELSE      868
867 IF PN% = PRON% THEN TAS(J) = TA1$: PRINT TAS(J): J = J + 1:
TAKSAY% = TAKSAY% + LEN(TA1$) / 3: GOTO 866

```

```

868 PS% = I + 9: MS% = J + 9: TS% = TAKSAY% + 10: TAK = 0: PRINT PS%,
MS%, TS%: INPUT T
869 LSET PROJNO$ = MKIS(PN%): LSET PSAYS$ = MKIS(PS% - 10): LSET
MSAYS$ = MKIS(MS% - 10): LSET TSAYS$ = MKIS(TS% - 10): PUT #2, PN%
870 FOR I = 11 TO PS%: AB$(I - 10) = "": TA = 0
871 FOR J = 11 TO MS%
872 FOR T = 1 TO LEN(TAS$(J - 10)) / 3: TAK = TAK + 1: TA = TA + 1
873 FOR N = 1 TO LEN(PTOOLS$(I - 10)) / 3: TAB$ = MID$(TAS$(J - 10), 3 * T -
2, 3): NAB$ = MID$(PTOOLS$(I - 10), 3 * N - 2, 3)
877 IF TAB$ = NAB$ THEN EK$ = "1": AB$(I - 10) = AB$(I - 10) + EK$
879 NEXT N: IF LEN(AB$(I - 10)) + 57 * (I - 11) = TAK - 1 THEN EK$ = "0":
AB$(I - 10) = AB$(I - 10) + EK$
881 A%(I - 10, TA) = VAL(EK$): PRINT A%(I - 10, TA), I - 10, TA
882 NEXT T, J
887 NEXT I
888 FOR I = 11 TO PS%
889 S$ = ""
890 FOR J = 11 TO TS%
892 S$ = S$ + RIGHT$(STR$(A%(I - 10, J - 10)), 1)
894 NEXT J
895 S$ = S$ + STRING$((100 - LEN(S$)), "*")
902 PRINT S$, LEN(S$): INPUT T
910 PRINT #1, PN%, S$
930 NEXT I
940 CLOSE #1, #4, #5: RETURN
1000 '*****DOSYADAN ISLENMEK ICIN VERILERI AKTARMA*****
1001 CLS : PRINT "DOSYADAN VERI OKUNUYOR"
1003 OPEN "I", #1, "C:\ERTAN\QB\PARTAK.DAT": I = 1
1010 INPUT "PROJE NO GIRIN="; PN%
1030 GET #2, PN%
1040 PS% = CVI(PSAYS): TS% = CVI(TSAYS)
1055 PRINT PS%, TS%: INPUT T

```

```

1056 IF NOT EOF(1) THEN INPUT #1, PRON%, S1$ ELSE CLOSE #1: GOTO
1110
1070 IF PRON% <> PN% THEN 1056
1095 FOR J = 1 TO TS%
1096 A%((I + 10), (J + 10)) = VAL(MID$(S1$, J, 1)): PRINT (I + 10), (J + 10),
A%((I + 10), (J + 10)): INPUT T
1097 NEXT J: I = I + 1: GOTO 1056
1110 RETURN
1200 '*****PP BENZERLIK MATRISI DOSYAYA YAZDIRMA
1210 FOR I = 11 TO PS% - 1
1215 S1$ = ""
1220 FOR J = I + 1 TO PS%
1225 S1$ = S1$ + MKS$(SPP!(I, J)): NEXT J
1230 LSET PRON$ = MKIS$(PN%): LSET SPPSATS = S1$
1240 PUT #3, I - 10
1250 NEXT I
1260 RETURN
1300 '*****PARCA AİLESİ DOSYAYA YAZDIRMA
1301 INPUT "GRUP DOSYASININ ADINI GIRIN :"; GD$
1304 OPEN "O", #6, GD$
1305 FOR L% = 1 TO GS%: G1$(L%) = ""
1310 FOR N% = 1 TO (LEN(G$(L%)) + 1) / 3
1312 G1$(L%) = G1$(L%) + MID$(G$(L%), 3 * N% - 2, 2): NEXT N%
1313 PRINT #6, PN%, L%, G1$(L%)
1314 NEXT L%: CLOSE #6: RETURN
2000 LOCATE 24, 1: INPUT "LUTFEN BİR TUSA BASIN..."; XS
2010 IF XS = "" THEN 2000
2020 RETURN

2500 FOR T = 1 TO 5000: NEXT T: RETURN

3000 GI = VAL(ENB1$): GJ = VAL(ENB2$)
3020 SPP!(GI, GJ) = 0: RETURN

```

3100 IF F% = 1 THEN F% = 2 ELSE F% = 1: RETURN

3500 '*****PARÇA AİLESİ KONTROLÜ*****

3505 UZ% = (LEN(G\$(H%)) + 1) / 3

3510 FOR P = 1 TO UZ% - 1

3520 FOR P1 = P + 1 TO UZ%

3523 IF MID\$(G\$(H%), (3 * P) - 2, 2) = MID\$(G\$(H%), (3 * P1) - 2, 2) AND P1 = UZ% THEN G\$(H%) = LEFT\$(G\$(H%), (3 * (P1 - 1)) - 1): GOTO 3540

3530 IF MID\$(G\$(H%), (3 * P) - 2, 2) = MID\$(G\$(H%), (3 * P1) - 2, 2) THEN G\$(H%) = LEFT\$(G\$(H%), 3 * (P1 - 1)) + RIGHT\$(G\$(H%), ((UZ% - P1) * 3) - 1)

3540 NEXT P1, P: PRINT G\$(H%): RETURN

4000 H% = ((VAL(RIGHT\$(B\$(1), 2))) - 10): G\$(H%) = G\$(H%) + "," + ENB2\$: GOSUB 3500: GOSUB 3000: RETURN

4100 H% = ((VAL(RIGHT\$(B\$(2), 2))) - 10): G\$(H%) = G\$(H%) + "," + ENB1\$: GOSUB 3500: GOSUB 3000: RETURN

EK E

MAKİNA-TAKİM VERİLERİ OLUŞTURMA PROGRAMI

'***** MACTOOL.BAS *****'

10 CLS

12 INPUT "MAKİNA TAKİM VERİ DOSYASININ ADINI GİRİN :"; MT\$

13 INPUT "MAKİNA SAYISINI GİRİN :"; MAKSAY%

15 INPUT X%

17 ON X% GOSUB 100, 180: END

100 CLS

120 OPEN "O", #2, MT\$

122 FOR I = 1 TO MAKSAY%: READ PRN%, MN%, NAV%, AFC, TAS

124 PRINT #2, PRN%; MN%; NAV%; AFC!; TAS

150 PRINT PRN%: PRINT MN%: PRINT NAV%: PRINT AFC!: PRINT TAS,
(LEN(TAS) / 3)

155 INPUT T: NEXT I

170 CLOSE #2: RETURN

180 OPEN "I", #2, MT\$

181 IF EOF(2) THEN PRINT "DOSYA SONU": INPUT V: GOTO 185

182 INPUT #2, PRN%, MN%, NAV%, AFC!, TAS

183 PRINT PRN%: PRINT MN%: PRINT NAV%: PRINT AFC!: PRINT TAS,
(LEN(TAS) / 3)

184 GOTO 181

185 CLOSE #2: RETURN

'*****MACTOOL

501 DATA 02,01,4,10000,"A01A02A03A04A05A06"

502 DATA 02,02,3,08000,"B01B02B03B04B05B06"

503 DATA 02,03,1,06000,"C01C02C03C04C05"

504 DATA 02,04,2,07000,"D01D02D03D04D05D06"

505 DATA 02,05,3,08500,"E01E02E03E04E05E06E07"

506 DATA 02,06,5,12000,"F01F02F03F04F05F06"

507 DATA 02,07,1,11000,"G01G02G03G04G05G06"

508 DATA 02,08,4,09000,"H01H02H03H04"

509 DATA 02,09,1,10500,"M01M02M03M04M05"

510 DATA 02,10,3,11000,"R01R02R03R04R05R06"



EK F

PARÇA-TAKIM VERİLERİ OLUŞTURMA PROGRAMI

***** PARTTOOL.BAS *****

10 CLS

12 INPUT "PARÇA-TAKIM VERİ DOSYASININ ADI :"; PT\$

13 INPUT "PARÇA SAYISI: "; PARSAY%

15 INPUT X%

17 ON X% GOSUB 20, 80: END

20 OPEN "O", #1, PT\$

32 FOR I = 1 TO PARSAY%: READ PRN%, PN%, DEM%, TRPS

40 PRINT #1, PRN%, PN%, DEM%, TRPS

50 PRINT PRN%: PRINT PN%: PRINT DEM%: PRINT TRPS, (LEN(TRPS) / 3)

55 INPUT Y: NEXT I

70 CLOSE #1: RETURN

80 OPEN "I", #1, PT\$

81 IF EOF(1) THEN PRINT "DOSYA SONU": INPUT V: GOTO 85

82 INPUT #1, PRN%, PN%, DEM%, TRPS

83 PRINT PRN%: PRINT PN%: PRINT DEM%: PRINT TRPS, (LEN(TRPS) / 3)

84 GOTO 81

85 CLOSE #1: RETURN

***PARTTOOL

301 DATA 02,01,0600,"A01D01D02D03H01H02M01M02M03M04"

302 DATA 02,02,0800,"A02A03A04B01C01C02M01M05R01R02R03R04"

303DATA

02,03,1800,"A01B01B02B03B04E02E03E04F01F02F03F04G05R02R03R04R06"

304 DATA 02,04,1200,"B02D01D02D03D04H01H02H03M01M02"

305 DATA 02,05,1300,"B01E02E03E04E05F04F05G01G02H03R01R03R04"

306 DATA 02,06,0900,"A01A02A05A06C04M02M03M04R05R06"

307DATA

02,07,0700,"A02A04A06C02C03C04C05F01G01M01M03M05R02R04R05R06"

308 DATA 02,08,1600,"B01B03E05E06F03F04F05G01R01R02R04"

309 DATA 02,09,1200,"A01B02B03B05B06E01E03E05E06E07F05G01G02R01R02"

310 DATA 02,10,1100,"D01D03D04E07H01H02H03H04M02M03"

311 DATA 02,11,1400,"A01A03A06C02C03C04C06M04M05R01R02R04R06"

312 DATA 02,12,0800,"A01D01D02H01H02M01M02M03"

313 DATA 02,13,1300,"B01B03D01E01E03E05F01F03F04G01G03R01R02R04R06"

314 DATA 02,14,0900,"C01D02D03D04H02H03H04M04M05"

315 DATA 02,15,1100,"A02A04A06C01C02C03C05E06G05M01M05R01R02R04R05"

316 DATA 02,16,0600,"A01A02C01C03C04E01M01M03M04M05R01R02R03"

317 DATA 02,17,1800,"A01D05D06F06G06H03H04M01M03M04M05"

318 DATA 02,18,1100,"B01C01E01E03E04E07F01F05G03G04G05G06R01R03R06"

319 DATA 02,19,1400,"B01B03B04C01C02E01F01F02F03H01R01R02R03R04"

320 DATA 02,20,0800,"A05D03D04D05D06H01H02H04M04M05"

321 DATA 02,21,0900,"A01A02A03A05C02C03H01M01M02M03M04R01R06"

322 DATA 02,22,1600,"A01A02C01C02C04H01M01M02R01R02R03"

323DATA

02,23,1000,"A01B01B02B03C01E01E02F01F02F03F05G01G02H01R01R02R03"

324 DATA 02,24,1200,"B01B02E02E04E05F01F03G02R01R02R04R06"

325 DATA 02,25,0600,"B01B03B04C01D02E02E03F01F03F04G01G02G04R01R03"

326 DATA 02,26,0600,"A01A02B01B02C04E01E02M01M03M04M05R01R02R03"

327 DATA 02,27,1800,"A01B02B03D05D06F01F06G06H03H04M01M03M04M05"

328DATA

02,28,1100,"A02A03B01C01E01E03E04E07F01F05G03G04G05G06R01R03R06"

329 DATA 02,29,1400,"B01B03B04C01C02E01E02F01F02F03F06H01R01R02R03R04"

330 DATA 02,30,0800,"A02A05D03D04D05D06H01H02H04M04M05"

331 DATA 02,31,0900,"A01A02A03A05C02C03H01M01M02M03M04R01R06"

332 DATA 02,32,1600,"A01A02A04C01C02C04H01H03M01M02R01R02R03"

333DATA

2,33,1000,"A01A03B01B02B03C01E01E02F01F02F03F05G01G02H01R01R02R03"

334 DATA 02,34,1200,"B01B02E02E04E05F01F03G02R01R02R04R06"

335 DATA 02,35,0600,"B01B03B04C01D02E02E03F01F03F04G01G02G04R01R03"

EK G

PARÇA-MAKİNA ARASINDA ORTAK TAKİM VERİLERİ OLUŞTURMA PROGRAMI

```
***** TIJ.BAS *****
5 DIM TRP$(50), TA$(50), T$(50), T(50, 50)
10 CLS
15 INPUT X: ON X GOTO 20, 100, 500: END
20 INPUT "PARÇA TAKİM DOSYA ADI :"; PT$: OPEN "I", #1, PT$
30 INPUT "MAKİNA TAKİM DOSYA ADI :"; MT$: OPEN "I", #2, MT$
31 L = 1: K = 1
32 IF NOT EOF(1) THEN INPUT #1, PRN%, PN%, DEM%, TRP$: TRP$(L) =
TRP$: PRINT TRP$(L): L = L + 1
34 IF NOT EOF(2) THEN INPUT #2, PRON%, MN%, NAV%, AFC!, TA$:
TA$(K) = TA$: PRINT TA$(K): K = K + 1
36 INPUT T
70 IF EOF(1) AND EOF(2) THEN CLOSE #1, #2: GOTO 100
80 GOTO 32
100 OPEN "O", #3, "C:\ERTAN\QB\TIJ.DAT"
110 FOR I = 1 TO 25
112 T$(I) = ""
120 FOR J = 1 TO 10
125 T(I, J) = 0
130 FOR K = 1 TO LEN(TRP$(I)) / 3
140 FOR L = 1 TO LEN(TA$(J)) / 3
150 K$ = MID$(TRP$(I), 3 * K - 2, 3): L$ = MID$(TA$(J), 3 * L - 2, 3)
160 IF K$ = L$ THEN T(I, J) = T(I, J) + 1
170 NEXT L, K: INPUT H
```

```
180 IF LEN(STR$(T(I, J))) = 2 THEN T$(I) = T$(I) + "0" + RIGHT$(STR$(T(I,
J)), 1)
190 IF LEN(STR$(T(I, J))) > 2 THEN T$(I) = T$(I) + RIGHT$(STR$(T(I, J)), 2)
200 NEXT J
205 PRINT #3, I, T$(I): PRINT T$(I)
220 NEXT I: CLOSE #3

500 OPEN "I", #3, "C:\ERTAN\QB\TIJ.DAT"
510 FOR I = 1 TO 25
520 IF NOT EOF(3) THEN INPUT #3, PN%, T$(I)
530 PRINT T$(I)
540 'INPUT T
545 NEXT I
```

EK H

MAX.AMAÇ FONKSİYONUNUN OLUŞTURULMASI

```
10 CLS : DIM XAS1$(50)
20 OPEN "I", #1, "C:\ERTAN\QB\XAS.DAT"
25 FOR I = 1 TO 25
30 IF NOT EOF(1) THEN INPUT #1, XAS$
40 XAS1$(I) = XAS$
45 IF I = 25 THEN XAS1$(I) = XAS$ + "+"
47 PRINT XAS1$(I)
50 NEXT I
100 FOR J = 1 TO 10
120 FOR K = 1 TO 3
125 READ X%(J, K): PRINT "X"; J; K, X%(J, K): LPRINT "X"; J; K, X%(J, K)
130 NEXT K, J
140 CLS : TOP = 0
150 FOR J = 1 TO 10: J$ = RIGHT$(STR$(J), (LEN(STR$(J))) - 1)
160 FOR K = 1 TO 3: K$ = RIGHT$(STR$(K), (LEN(STR$(K))) - 1)
162 X$ = "X" + J$ + K$: PRINT X$
165 IF X%(J, K) = 0 THEN 215
170 FOR I = 1 TO 25: K$ = ""
171 FOR S = 1 TO LEN(XAS1$(I))
181 EK$ = MID$(XAS1$(I), S, 1)
183 IF EK$ <> "+" THEN K$ = K$ + EK$: PRINT EK$, K$, X$
184 IF EK$ = "+" THEN GOSUB 250: K$ = ""
210 NEXT S, I
215 NEXT K
220 PRINT TOP: INPUT T
230 NEXT J: END
250 IF X$ = RIGHT$(K$, ((LEN(K$)) - 4)) THEN TOP = TOP + VAL((LEFT$(K$,
5))) : PRINT K$, X$, TOP: INPUT T
260 RETURN
```

ÖZGEÇMİŞ

Süleyman Ertan TANSÖKER, 1971 yılında İstanbul'da doğdu. İlk öğrenimini Hobyarlı Ahmetpaşa İlkokulu'nda, orta öğretimini Davutpaşa Lisesi-Orta Kısımında, lise öğrenimini İstanbul Maçka Anadolu Teknik Lisesi Bilgisayar Bölümü'nde tamamladı. 1989 yılında İstanbul Teknik Üniversitesi İşletme Mühendisliği Bölümü'ne girdi. 1993 yılında mezun olduktan sonra yine aynı bölümde Yüksek Lisans eğitime başladı. 1994-1995 yılları arasında İTÜ İşletme Fakültesi Üretim Yönetimi ve Pazarlama Anabilim Dalı'nda Araştırma Görevlisi olarak çalışan ve askerlik hizmetini tamamladıktan sonra görevinden ayrılan Süleyman Ertan TANSÖKER halen KOÇBANK A.Ş.'de Kurumsal Bankacılık bölümünde çalışmaktadır.

